Virtex-II Pro Libraries Guide for Schematic Designs

ISE 10.1

Xilinx Trademarks and Copyright Information

Xilinx is disclosing this user guide, manual, release note, and/or specification (the “Documentation”) to you solely for use in the development of designs to operate with Xilinx hardware devices. You may not reproduce, distribute, republish, download, display, post, or transmit the Documentation in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx. Xilinx expressly disclaims any liability arising out of your use of the Documentation. Xilinx reserves the right, at its sole discretion, to change the Documentation without notice at any time. Xilinx assumes no obligation to correct any errors contained in the Documentation, or to advise you of any corrections or updates. Xilinx expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information.

THE DOCUMENTATION IS DISCLOSED TO YOU “AS-IS” WITH NO WARRANTY OF ANY KIND. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS. IN NO EVENT WILL XILINX BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION.

© Copyright 2002 – 2008 Xilinx, Inc. All Rights Reserved. XILINX, the Xilinx logo, the Brand Window and other designated brands included herein are trademarks of Xilinx, Inc. All other trademarks are the property of their respective owners. The PowerPC name and logo are registered trademarks of IBM Corp., and used under license. All other trademarks are the property of their respective owners.

Table of Contents

About this Guide 13
Functional Categories 15
About Design Elements 35
- ACC16 36
- ACC4 38
- ACC8 40
- ADD16 42
- ADD4 44
- ADD8 46
- ADSU16 48
- ADSU4 50
- ADSU8 52
- AND12 54
- AND16 55
- AND2 56
- AND2B1 57
- AND2B2 58
- AND3 59
- AND3B1 60
- AND3B2 61
- AND3B3 62
- AND4 63
- AND4B1 64
- AND4B2 65
- AND4B3 66
- AND4B4 67
- AND5 68
- AND5B1 69
- AND5B2 70
- AND5B3 71
- AND5B4 72
- AND5B5 73
- AND6 74
- AND7 75
- AND8 76
- AND9 77
- BRLSHFT4 78
- BRLSHFT8 79
- BSCAN_VIRTEX2 81
- BUF 82
- BUFCF 83
- BUFE 84
- BUFE16 85
- BUFE4 86
- BUFE8 87
- BUFG 88
- BUFGCE 89
- BUFGCE_1 90
- BUFGMUX 91
- BUFGMUX_1 92
- BUFGP 93
- BUFT 94
- BUFT16 95
- BUFT4 97
- BUFT8 99
- CAPTURE_VIRTEX2 101

CB16CE....102
CB16CLE....104
CB16CLED....106
CB16RE....108
CB2CE....110
CB2CLE....112
CB2CLED....114
CB2RE....116
CB4CE....118
CB4CLE....120
CB4CLED....122
CB4RE....124
CB8CE....126
CB8CLE....128
CB8CLED....130
CB8RE....132
CC16CE....134
CC16CLE....136
CC16CLED....138
CC16RE....140
CC8CE....142
CC8CLE....144
CC8CLED....146
CC8RE....148
CD4CE....150
CD4CLE....152
CD4RE....154
CD4RLE....156
CJ4CE....158
CJ4RE....159
CJ5CE....160
CJ5RE....161
CJ8CE....162
CJ8RE....163
CLKDLL....164
CLKDLLE....166
CLKDLLHF....168
COMP16....170
COMP2....171
COMP4....172
COMP8....173
COMPM16....174
COMPM2....176
COMPM4....177
COMPM8....178
COMPMC16....180
COMPMC8....182
CR16CE....184
CR8CE....185
D2_4E....186
D3_8E....187
D4_16E....188
DCM....189
DEC_CC16....193
DEC_CC4....195
DEC_CC8....196
DECODE16....197
DECODE32....198
DECODE4....199

DECODE64 200
DECODE8 201
FD 202
FD_1 203
FD16CE 204
FD16RE 205
FD4CE 206
FD4RE 208
FD8CE 210
FD8RE 211
FDC 212
FDC_1 213
FDCE 214
FDCE_1 215
FDCP 216
FDCP_1 218
FDCPE 219
FDCPE_1 222
FDDRCPE 224
FDDRRSE 226
FDE 228
FDE_1 229
FDP 230
FDP_1 232
FDPE 233
FDPE_1 234
FDR 235
FDR_1 236
FDRE 237
FDRE_1 238
FDRS 239
FDRS_1 241
FDRSE 242
FDRSE_1 244
FDS 246
FDS_1 247
FDSE 248
FDSE_1 250
FJKC 251
FJKCE 252
FJKP 253
FJKPE 254
FJKRSE 255
FJKSRE 257
FMAP 259
FTC 260
FTCE 261
FTCLE 262
FTCLEX 263
FTP 264
FTPE 265
FTPLE 266
FTRSE 267
FTRSLE 268
FTSRE 270
FTSRLE 271
GND 273
GT_AURORA_1 274
GT_AURORA_2 275

GT_AURORA_4 276
GT_CUSTOM 277
GT_ETHERNET_1 278
GT_ETHERNET_2 279
GT_ETHERNET_4 280
GT_FIBRE_CHAN_1 281
GT_FIBRE_CHAN_2 282
GT_FIBRE_CHAN_4 283
GT_INFINIBAND_1 284
GT_INFINIBAND_2 285
GT_INFINIBAND_4 286
GT_XAUI_1 287
GT_XAUI_2 288
GT_XAUI_4 289
GT10_10GE_4 290
GT10_10GE_8 292
GT10_10GFC_4 294
GT10_10GFC_8 296
GT10_AURORA_1 298
GT10_AURORA_2 300
GT10_AURORA_4 302
GT10_AURORAX_4 304
GT10_AURORAX_8 306
GT10_CUSTOM 308
GT10_INFINIBAND_1 310
GT10_INFINIBAND_2 312
GT10_INFINIBAND_4 314
GT10_OC192_4 316
GT10_OC192_8 318
GT10_OC48_1 320
GT10_OC48_2 322
GT10_OC48_4 324
GT10_PCI_EXPRESS_1 326
GT10_PCI_EXPRESS_2 328
GT10_PCI_EXPRESS_4 330
GT10_XAUI_1 332
GT10_XAUI_2 334
GT10_XAUI_4 336
IBUF 338
IBUF16 340
IBUF4 341
IBUF8 342
IBUFDS 343
IBUFDS_DIFF_OUT 345
IBUFG 346
IBUFGDS 347
ICAP_VIRTEX2 350
IFD 351
IFD_1 352
IFD16 353
IFD4 354
IFD8 355
IFDDRCPE 356
IFDDRRSE 358
IFDI 360
IFDI_1 361
IFDX 362
IFDX_1 363
IFDX16 364

IFDX4 365
IFDX8 366
IFDXI 367
IFDXI_1 368
ILD 369
ILD_1 370
ILD16 371
ILD4 372
ILD8 373
ILDI 374
ILDI_1 375
ILDX 376
ILDX_1 377
ILDX16 378
ILDX4 379
ILDX8 380
ILDXI 381
ILDXI_1 382
INV 383
INV16 384
INV4 385
INV8 386
IOBUF 387
IOBUFDS 389
JTAGPPC 391
KEEPER 392
LD 393
LD_1 394
LD16 395
LD16CE 396
LD4 398
LD4CE 399
LD8 401
LD8CE 402
LDC 404
LDC_1 405
LDCE 406
LDCE_1 407
LDCP 409
LDCP_1 411
LDCPE 413
LDCPE_1 415
LDE 417
LDE_1 418
LDP 419
LDP_1 420
LDPE 421
LDPE_1 422
LUT1 424
LUT1_D 426
LUT1_L 428
LUT2 430
LUT2_D 432
LUT2_L 434
LUT3 436
LUT3_D 438
LUT3_L 440
LUT4 442
LUT4_D 444

LUT4_L....446
M16_1E....448
M2_1....450
M2_1B1....451
M2_1B2....452
M2_1E....453
M4_1E....454
M8_1E....455
MULT_AND....457
MULT18X18....458
MULT18X18S....459
MUXCY....460
MUXCY_D....461
MUXCY_L....462
MUXF5....463
MUXF5_D....464
MUXF5_L....465
MUXF6....466
MUXF6_D....467
MUXF6_L....468
MUXF7....469
MUXF7_D....470
MUXF7_L....471
MUXF8....472
MUXF8_D....473
MUXF8_L....474
NAND12....475
NAND16....476
NAND2....477
NAND2B1....478
NAND2B2....479
NAND3....480
NAND3B1....481
NAND3B2....482
NAND3B3....483
NAND4....484
NAND4B1....485
NAND4B2....486
NAND4B3....487
NAND4B4....488
NAND5....489
NAND5B1....490
NAND5B2....491
NAND5B3....492
NAND5B4....493
NAND5B5....494
NAND6....495
NAND7....496
NAND8....497
NAND9....498
NOR12....499
NOR16....500
NOR2....501
NOR2B1....502
NOR2B2....503
NOR3....504
NOR3B1....505
NOR3B2....506
NOR3B3....507

NOR4 ... 508
NOR4B1 ... 509
NOR4B2 ... 510
NOR4B3 ... 511
NOR4B4 ... 512
NOR5 ... 513
NOR5B1 ... 514
NOR5B2 ... 515
NOR5B3 ... 516
NOR5B4 ... 517
NOR5B5 ... 518
NOR6 ... 519
NOR7 ... 520
NOR8 ... 521
NOR9 ... 522
OBUF ... 523
OBUF16 ... 524
OBUF4 ... 525
OBUF8 ... 526
OBUFDS ... 527
OBUFE ... 528
OBUFE16 ... 529
OBUFE4 ... 530
OBUFE8 ... 531
OBUFT ... 532
OBUFT16 ... 534
OBUFT4 ... 535
OBUFT8 ... 537
OBUFTDS ... 538
OFD ... 540
OFD_1 ... 541
OFD16 ... 542
OFD4 ... 543
OFD8 ... 544
OFDDRCPE ... 545
OFDDRRSE ... 546
OFDDRTCPE ... 548
OFDDRTRSE ... 550
OFDE ... 552
OFDE_1 ... 553
OFDE16 ... 554
OFDE4 ... 555
OFDE8 ... 556
OFDI ... 557
OFDI_1 ... 558
OFDT ... 559
OFDT_1 ... 560
OFDT16 ... 561
OFDT4 ... 562
OFDT8 ... 563
OFDX ... 564
OFDX_1 ... 565
OFDX16 ... 566
OFDX4 ... 567
OFDX8 ... 568
OFDXI ... 569
OFDXI_1 ... 570
OR12 ... 571
OR16 ... 572

OR2 573
OR2B1 574
OR2B2 575
OR3 576
OR3B1 577
OR3B2 578
OR3B3 579
OR4 580
OR4B1 581
OR4B2 582
OR4B3 583
OR4B4 584
OR5 585
OR5B1 586
OR5B2 587
OR5B3 588
OR5B4 589
OR5B5 590
OR6 591
OR7 592
OR8 593
OR9 594
ORCY 595
PPC405 596
PULLDOWN 599
PULLUP 600
RAM128X1S 601
RAM128X1S_1 603
RAM16X1D 605
RAM16X1D_1 607
RAM16X1S 609
RAM16X1S_1 611
RAM16X2D 613
RAM16X2S 615
RAM16X4D 617
RAM16X4S 619
RAM16X8D 621
RAM16X8S 623
RAM32X1D 625
RAM32X1D_1 627
RAM32X1S 629
RAM32X1S_1 631
RAM32X2S 633
RAM32X4S 635
RAM32X8S 637
RAM64X1D 639
RAM64X1D_1 641
RAM64X1S 643
RAM64X1S_1 645
RAM64X2S 647
RAMB16_S1 649
RAMB16_S1_S1 652
RAMB16_S1_S18 660
RAMB16_S1_S2 668
RAMB16_S1_S36 676
RAMB16_S1_S4 684
RAMB16_S1_S9 692
RAMB16_S18 700
RAMB16_S18_S18 703

RAMB16_S18_S36 709
RAMB16_S2 717
RAMB16_S2_S18 720
RAMB16_S2_S2 728
RAMB16_S2_S36 736
RAMB16_S2_S4 745
RAMB16_S2_S9 753
RAMB16_S36 761
RAMB16_S36_S36 764
RAMB16_S4 772
RAMB16_S4_S18 775
RAMB16_S4_S36 783
RAMB16_S4_S4 791
RAMB16_S4_S9 799
RAMB16_S9 807
RAMB16_S9_S18 810
RAMB16_S9_S36 818
RAMB16_S9_S9 826
RAMB4_S1 834
RAMB4_S1_S1 836
RAMB4_S1_S16 838
RAMB4_S1_S2 840
RAMB4_S1_S4 844
RAMB4_S1_S8 846
RAMB4_S16 848
RAMB4_S16_S16 850
RAMB4_S2 852
RAMB4_S2_S16 854
RAMB4_S2_S2 857
RAMB4_S2_S4 859
RAMB4_S2_S8 861
RAMB4_S4 863
RAMB4_S4_S16 865
RAMB4_S4_S4 868
RAMB4_S4_S8 871
RAMB4_S8 873
RAMB4_S8_S16 875
RAMB4_S8_S8 878
ROM128X1 881
ROM16X1 883
ROM256X1 885
ROM32X1 887
ROM64X1 889
SOP3 891
SOP3B1A 892
SOP3B1B 893
SOP3B2A 894
SOP3B2B 895
SOP3B3 896
SOP4 897
SOP4B1 898
SOP4B2A 899
SOP4B2B 900
SOP4B3 901
SOP4B4 902
SR16CE 903
SR16CLE 905
SR16CLED 907
SR16RE 909

SR16RLE 911
SR16RLED 913
SR4CE 915
SR4CLE 917
SR4CLED 919
SR4RE 921
SR4RLE 923
SR4RLED 925
SR8CE 927
SR8CLE 929
SR8CLED 931
SR8RE 933
SR8RLE 935
SR8RLED 937
SRL16 939
SRL16_1 941
SRL16E 943
SRL16E_1 945
SRLC16 947
SRLC16_1 949
SRLC16E 951
SRLC16E_1 953
STARTBUF_VIRTEX2 955
STARTUP_VIRTEX2 956
VCC 957
XNOR2 958
XNOR3 959
XNOR4 960
XNOR5 961
XNOR6 962
XNOR7 963
XNOR8 964
XNOR9 965
XOR2 966
XOR3 967
XOR4 968
XOR5 969
XOR6 970
XOR7 971
XOR8 972
XOR9 973
XORCY 974
XORCY_D 975
XORCY_L 976

About this Guide

This HDL guide is part of the ISE documentation collection. A separate version of this guide is available if you prefer to work with schematics.

This guide contains the following:

- Introduction.
- A list of design elements supported in this architecture, organized by functional categories.
- Individual descriptions of each available primitive.

About Design Elements

This version of the Libraries Guide describes design elements available for this architecture. There are several categories of design elements:

- **Primitives -** The simplest design elements in the Xilinx libraries. Primitives are the design element "atoms." Examples of Xilinx primitives are the simple buffer, BUF, and the D flip-flop with clock enable and clear, FDCE.
- **Macros -** The design element "molecules" of the Xilinx libraries. Macros can be created from the design element primitives or macros. For example, the FD4CE flip-flop macro is a composite of 4 FDCE primitives.

Xilinx maintains software libraries with hundreds of functional design elements (macros and primitives) for different device architectures. New functional elements are assembled with each release of development system software. This guide is one in a series of architecture-specific libraries.

Functional Categories

This section categorizes, by function, the circuit design elements described in detail later in this guide. The elements (*primitives* and *macros*) are listed in alphanumeric order under each functional category.

Arithmetic
Buffer
Carry Logic
Comparator
Counter
DDR Flip Flop
Decoder
Flip Flop
General
GigaBit 10 IO
GigaBit IO/Processor
IO
IO FlipFlop
IO Latch
Latch
Logic
LUT
Map
Memory
Mux
Shift Register
Shifter

Arithmetic

Design Element	Description
ACC16	Macro: 16-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset
ACC4	Macro: 4-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset
ACC8	Macro: 8-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset
ADD16	Macro: 16-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow
ADD4	Macro: 4-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow
ADD8	Macro: 8-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow
ADSU16	Macro: 16-Bit Cascadable Adder/Subtracter with Carry-In, Carry-Out, and Overflow
ADSU4	Macro: 4-Bit Cascadable Adder/Subtracter with Carry-In, Carry-Out, and Overflow
ADSU8	Macro: 8-Bit Cascadable Adder/Subtracter with Carry-In, Carry-Out, and Overflow
MULT18X18	Primitive: 18 x 18 Signed Multiplier
MULT18X18S	Primitive: 18 x 18 Signed Multiplier -- Registered Version

Buffer

Design Element	Description
BUF	Primitive: General Purpose Buffer
BUFCF	Primitive: Fast Connect Buffer
BUFE	Primitive: Internal 3-State Buffer with Active High Enable
BUFE16	Macro: 16-Bit Internal 3-State Buffer with Active High Enable
BUFE4	Macro: 4-BitInternal 3-State Buffer with Active High Enable

Design Element	Description
BUFE8	Macro: 8-Bit Internal 3-State Buffer with Active High Enable
BUFG	Primitive: Global Clock Buffer
BUFGCE	Primitive: Global Clock Buffer with Clock Enable
BUFGCE_1	Primitive: Global Clock Buffer with Clock Enable and Output State 1
BUFGMUX	Primitive: Global Clock MUX Buffer
BUFGMUX_1	Primitive: Global Clock MUX Buffer with Output State 1
BUFGP	Primitive: Primary Global Buffer for Driving Clocks or Longlines (Four per PLD Device)
BUFT	Primitive: Internal 3-State Buffer with Active Low Enable
BUFT16	Macro: 16-Bit Internal 3-State Buffers with Active Low Enable
BUFT4	Macro: 4-Bit Internal 3-State Buffers with Active Low Enable
BUFT8	Macro: 8-Bit Internal 3-State Buffers with Active Low Enable

Carry Logic

Design Element	Description
MUXCY	Primitive: 2-to-1 Multiplexer for Carry Logic with General Output
MUXCY_D	Primitive: 2-to-1 Multiplexer for Carry Logic with Dual Output
MUXCY_L	Primitive: 2-to-1 Multiplexer for Carry Logic with Local Output
ORCY	Primitive: OR with Carry Logic
XORCY	Primitive: XOR for Carry Logic with General Output
XORCY_D	Primitive: XOR for Carry Logic with Dual Output
XORCY_L	Primitive: XOR for Carry Logic with Local Output

Comparator

Design Element	Description
COMP16	Macro: 16-Bit Identity Comparator
COMP2	Macro: 2-Bit Identity Comparator
COMP4	Macro: 4-Bit Identity Comparator
COMP8	Macro: 8-Bit Identity Comparator
COMPM16	Macro: 16-Bit Magnitude Comparator
COMPM2	Macro: 2-Bit Magnitude Comparator
COMPM4	Macro: 4-Bit Magnitude Comparator
COMPM8	Macro: 8-Bit Magnitude Comparator
COMPMC16	Macro: 16-Bit Magnitude Comparator
COMPMC8	Macro: 8-Bit Magnitude Comparator

Counter

Design Element	Description
CB16CE	Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB16CLE	Macro: 16-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear
CB16CLED	Macro: 16-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear
CB16RE	Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB2CE	Macro: 2-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB2CLE	Macro: 2-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear
CB2CLED	Macro: 2-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear
CB2RE	Macro: 2-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB4CE	Macro: 4-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB4CLE	Macro: 4-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear
CB4CLED	Macro: 4-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear
CB4RE	Macro: 4-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB8CE	Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB8CLE	Macro: 8-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear
CB8CLED	Macro: 8-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear
CB8RE	Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CC16CE	Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC16CLE	Macro: 16-Bit Loadable Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC16CLED	Macro: 16-Bit Loadable Cascadable Bidirectional Binary Counter with Clock Enable and Asynchronous Clear
CC16RE	Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CC8CE	Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC8CLE	Macro: 8-Bit Loadable Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC8CLED	Macro: 8-Bit Loadable Cascadable Bidirectional Binary Counter with Clock Enable and Asynchronous Clear
CC8RE	Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CD4CE	Macro: 4-Bit Cascadable BCD Counter with Clock Enable and Asynchronous Clear
CD4CLE	Macro: 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Asynchronous Clear
CD4RE	Macro: 4-Bit Cascadable BCD Counter with Clock Enable and Synchronous Reset
CD4RLE	Macro: 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Synchronous Reset

Design Element	Description
CJ4CE	Macro: 4-Bit Johnson Counter with Clock Enable and Asynchronous Clear
CJ4RE	Macro: 4-Bit Johnson Counter with Clock Enable and Synchronous Reset
CJ5CE	Macro: 5-Bit Johnson Counter with Clock Enable and Asynchronous Clear
CJ5RE	Macro: 5-Bit Johnson Counter with Clock Enable and Synchronous Reset
CJ8CE	Macro: 8-Bit Johnson Counter with Clock Enable and Asynchronous Clear
CJ8RE	Macro: 8-Bit Johnson Counter with Clock Enable and Synchronous Reset
CR16CE	Macro: 16-Bit Negative-Edge Binary Ripple Counter with Clock Enable and Asynchronous Clear
CR8CE	Macro: 8-Bit Negative-Edge Binary Ripple Counter with Clock Enable and Asynchronous Clear

DDR Flip Flop

Design Element	Description
FDDRCPE	Primitive: Dual Data Rate D Flip-Flop with Clock Enable and Asynchronous Preset and Clear
FDDRRSE	Primitive: Dual Data Rate D Flip-Flop with Clock Enable and Synchronous Reset and Set
IFDDRCPE	Primitive: Dual Data Rate Input D Flip-Flop with Clock Enable and Asynchronous Preset and Clear
IFDDRRSE	Primitive: Dual Data Rate Input D Flip-Flop with Synchronous Reset and Set and Clock Enable
OFDDRCPE	Primitive: Dual Data Rate Output D Flip-Flop with Clock Enable and Asynchronous Preset and Clear
OFDDRRSE	Primitive: Dual Data Rate Output D Flip-Flop with Synchronous Reset and Set and Clock Enable
OFDDRTCPE	Primitive: Dual Data Rate D Flip-Flop with Active-Low 3--State Output Buffer, Clock Enable, and Asynchro-nous Preset and Clear
OFDDRTRSE	Primitive: Dual Data Rate D Flip-Flop with Active -Low 3-State Output Buffer, Synchronous Reset and Set, and Clock Enable

Decoder

Design Element	Description
D2_4E	Macro: 2- to 4-Line Decoder/Demultiplexer with Enable
D3_8E	Macro: 3- to 8-Line Decoder/Demultiplexer with Enable
D4_16E	Macro: 4- to 16-Line Decoder/Demultiplexer with Enable
DEC_CC16	Macro: 16-Bit Active Low Decoder
DEC_CC4	Macro: 4-Bit Active Low Decoder
DEC_CC8	Macro: 8-Bit Active Low Decoder
DECODE16	Macro: 16-Bit Active-Low Decoder
DECODE32	Macro: 32-Bit Active-Low Decoder

Design Element	Description
DECODE4	Macro: 4-Bit Active-Low Decoder
DECODE64	Macro: 64-Bit Active-Low Decoder
DECODE8	Macro: 8-Bit Active-Low Decoder

Flip Flop

Design Element	Description
FD	Primitive: D Flip-Flop
FD_1	Primitive: D Flip-Flop with Negative-Edge Clock
FD16CE	Macro: 16-Bit Data Register with Clock Enable and Asynchronous Clear
FD16RE	Macro: 16-Bit Data Register with Clock Enable and Synchronous Reset
FD4CE	Macro: 4-Bit Data Register with Clock Enable and Asynchronous Clear
FD4RE	Macro: 4-Bit Data Register with Clock Enable and Synchronous Reset
FD8CE	Macro: 8-Bit Data Register with Clock Enable and Asynchronous Clear
FD8RE	Macro: 8-Bit Data Register with Clock Enable and Synchronous Reset
FDC	Primitive: D Flip-Flop with Asynchronous Clear
FDC_1	Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Clear
FDCE	Primitive: D Flip-Flop with Clock Enable and Asynchronous Clear
FDCE_1	Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Clear
FDCP	Primitive: D Flip-Flop with Asynchronous Preset and Clear
FDCP_1	Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Preset and Clear
FDCPE	Primitive: D Flip-Flop with Clock Enable and Asynchronous Preset and Clear
FDCPE_1	Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset and Clear
FDE	Primitive: D Flip-Flop with Clock Enable
FDE_1	Primitive: D Flip-Flop with Negative-Edge Clock and Clock Enable
FDP	Primitive: D Flip-Flop with Asynchronous Preset
FDP_1	Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Preset
FDPE	Primitive: D Flip-Flop with Clock Enable and Asynchronous Preset
FDPE_1	Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset
FDR	Primitive: D Flip-Flop with Synchronous Reset
FDR_1	Primitive: D Flip-Flop with Negative-Edge Clock and Synchronous Reset
FDRE	Primitive: D Flip-Flop with Clock Enable and Synchronous Reset
FDRE_1	Primitive: D Flip-Flop with Negative-Clock Edge, Clock Enable, and Synchronous Reset
FDRS	Primitive: D Flip-Flop with Synchronous Reset and Set
FDRS_1	Primitive: D Flip-Flop with Negative-Clock Edge and Synchronous Reset and Set

Design Element	Description
FDRSE	Primitive: D Flip-Flop with Synchronous Reset and Set and Clock Enable
FDRSE_1	Primitive: D Flip-Flop with Negative-Clock Edge, Synchronous Reset and Set, and Clock Enable
FDS	Primitive: D Flip-Flop with Synchronous Set
FDS_1	Primitive: D Flip-Flop with Negative-Edge Clock and Synchronous Set
FDSE	Primitive: D Flip-Flop with Clock Enable and Synchronous Set
FDSE_1	Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Synchronous Set
FJKC	Macro: J-K Flip-Flop with Asynchronous Clear
FJKCE	Macro: J-K Flip-Flop with Clock Enable and Asynchronous Clear
FJKP	Macro: J-K Flip-Flop with Asynchronous Preset
FJKPE	Macro: J-K Flip-Flop with Clock Enable and Asynchronous Preset
FJKRSE	Macro: J-K Flip-Flop with Clock Enable and Synchronous Reset and Set
FJKSRE	Macro: J-K Flip-Flop with Clock Enable and Synchronous Set and Reset
FTC	Macro: Toggle Flip-Flop with Asynchronous Clear
FTCE	Macro: Toggle Flip-Flop with Clock Enable and Asynchronous Clear
FTCLE	Macro: Toggle/Loadable Flip-Flop with Clock Enable and Asynchronous Clear
FTCLEX	Macro: Toggle/Loadable Flip-Flop with Clock Enable and Asynchronous Clear
FTP	Macro: Toggle Flip-Flop with Asynchronous Preset
FTPE	Macro: Toggle Flip-Flop with Clock Enable and Asynchronous Preset
FTPLE	Macro: Toggle/Loadable Flip-Flop with Clock Enable and Asynchronous Preset
FTRSE	Macro: Toggle Flip-Flop with Clock Enable and Synchronous Reset and Set
FTRSLE	Macro: Toggle/Loadable Flip-Flop with Clock Enable and Synchronous Reset and Set
FTSRE	Macro: Toggle Flip-Flop with Clock Enable and Synchronous Set and Reset
FTSRLE	Macro: Toggle/Loadable Flip-Flop with Clock Enable and Synchronous Set and Reset

General

Design Element	Description
BSCAN_VIRTEX2	Primitive: Virtex2 Boundary Scan Logic Control Circuit
CAPTURE_VIRTEX2	Primitive: Virtex4 Register State Capture for Bitstream Readback
CLKDLL	Primitive: Clock Delay Locked Loop
CLKDLLE	Primitive: Clock Delay Locked Loop with Expanded Output
CLKDLLHF	Primitive: High Frequency Clock Delay Locked Loop
DCM	Primitive: Digital Clock Manager
GND	Primitive: Ground-Connection Signal Tag
ICAP_VIRTEX2	Primitive: User Interface to Virtex-II Internal Configuration Access Port
KEEPER	Primitive: KEEPER Symbol
PULLDOWN	Primitive: Resistor to GND for Input Pads, Open-Drain, and 3-State Outputs

Design Element	Description
PULLUP	Primitive: Resistor to VCC for Input PADs, Open-Drain, and 3-State Outputs
STARTBUF_VIRTEX2	Primitive: VHDL Simulation of FPGA Designs
STARTUP_VIRTEX2	Primitive: Virtex-II, Virtex-II Pro, and Virtex-II User Interface to Global Clock, Reset, and 3-State Controls
VCC	Primitive: VCC-Connection Signal Tag

GigaBit 10 IO

Design Element	Description
GT10_10GE_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_10GE_8	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_10GFC_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_10GFC_8	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_AURORA_1	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_AURORA_2	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_AURORA_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_AURORAX_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_AURORAX_8	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_CUSTOM	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_INFINIBAND_1	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_INFINIBAND_2	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_INFINIBAND_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_OC192_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_OC192_8	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_OC48_1	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_OC48_2	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_OC48_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_PCI_EXPRESS_1	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_PCI_EXPRESS_2	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_PCI_EXPRESS_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_XAUI_1	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_XAUI_2	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT10_XAUI_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O

GigaBit IO/Processor

Design Element	Description
GT_AURORA_1	Primitive: Gigabit Transceiver for High-Speed I/O

Design Element	Description
GT_AURORA_2	Primitive: Gigabit Transceiver for High-Speed I/O
GT_AURORA_4	Primitive: Gigabit Transceiver for High-Speed I/O
GT_CUSTOM	Primitive: Gigabit Transceiver for High-Speed I/O
GT_ETHERNET_1	Primitive: Gigabit Transceiver for High-Speed I/O
GT_ETHERNET_2	Primitive: Gigabit Transceiver for High-Speed I/O
GT_ETHERNET_4	Primitive: Gigabit Transceiver for High-Speed I/O
GT_FIBRE_CHAN_1	Primitive: Gigabit Transceiver for High-Speed I/O
GT_FIBRE_CHAN_2	Primitive: Gigabit Transceiver for High-Speed I/O
GT_FIBRE_CHAN_4	Primitive: Gigabit Transceiver for High-Speed I/O
GT_INFINIBAND_1	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT_INFINIBAND_2	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT_INFINIBAND_4	Primitive: 10-Gigabit Transceiver for High-Speed I/O
GT_XAUI_1	Primitive: Gigabit Transceiver for High-Speed I/O
GT_XAUI_2	Primitive: Gigabit Transceiver for High-Speed I/O
GT_XAUI_4	Primitive: Gigabit Transceiver for High-Speed I/O
JTAGPPC	Primitive: JTAG Primitive for the PowerPC
PPC405	Primitive: Primitive for the PowerPC Core

IO

Design Element	Description
IBUF	Primitive: Input Buffer
IBUF16	Macro: 16-Bit Input Buffer
IBUF4	Macro: 4-Bit Input Buffer
IBUF8	Macro: 8-Bit Input Buffer
IBUFDS	Primitive: Differential Signaling Input Buffer with Optional Delay
IBUFDS_DIFF_OUT	Primitive: Differential I/O Input Buffer with Differential Outputs
IBUFG	Primitive: Dedicated Input Clock Buffer
IBUFGDS	Primitive: Differential Signaling Dedicated Input Clock Buffer and Optional Delay
IOBUF	Primitive: Bi-Directional Buffer
IOBUFDS	Primitive: 3-State Differential Signaling I/O Buffer with Active Low Output Enable
OBUF	Primitive: Output Buffer
OBUF16	Macro: 16-Bit Output Buffer
OBUF4	Macro: 4-Bit Output Buffer
OBUF8	Macro: 8-Bit Output Buffer
OBUFDS	Primitive: Differential Signaling Output Buffer
OBUFE	Macro: 3-State Output Buffer with Active-High Output Enable
OBUFE16	Macro: 16-Bit 3-State Output Buffer with Active-High Output Enable

Design Element	Description
OBUFE4	Macro: 4-Bit 3-State Output Buffer with Active-High Output Enable
OBUFE8	Macro: 8-Bit 3-State Output Buffer with Active-High Output Enable
OBUFT	Primitive: 3-State Output Buffer with Active Low Output Enable
OBUFT16	Macro: 16-Bit 3-State Output Buffer with Active Low Output Enable
OBUFT4	Macro: 4-Bit 3-State Output Buffers with Active-Low Output Enable
OBUFT8	Macro: 8-Bit 3-State Output Buffers with Active-Low Output Enable
OBUFTDS	Primitive: 3-State Output Buffer with Differential Signaling, Active-Low Output Enable

IO FlipFlop

Design Element	Description
IFD	Macro: Input D Flip-Flop
IFD_1	Macro: Input D Flip-Flop with Inverted Clock (Asynchronous Preset)
IFD16	Macro: 16-Bit Input D Flip-Flop
IFD4	Macro: 4-Bit Input D Flip-Flop
IFD8	Macro: 8-Bit Input D Flip-Flop
IFDI	Macro: Input D Flip-Flop (Asynchronous Preset)
IFDI_1	Macro: Input D Flip-Flop with Inverted Clock (Asynchronous Preset)
IFDX	Macro: Input D Flip-Flop with Clock Enable
IFDX_1	Macro: Input D Flip-Flop with Inverted Clock and Clock Enable
IFDX16	Macro: 16-Bit Input D Flip-Flops with Clock Enable
IFDX4	Macro: 4-Bit Input D Flip-Flop with Clock Enable
IFDX8	Macro: 8-Bit Input D Flip-Flop with Clock Enable
IFDXI	Macro: Input D Flip-Flop with Clock Enable (Asynchronous Preset)
IFDXI_1	Macro: Input D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)
OFD	Macro: Output D Flip-Flop
OFD_1	Macro: Output D Flip-Flop with Inverted Clock
OFD16	Macro: 16-Bit Output D Flip-Flop
OFD4	Macro: 4-Bit Output D Flip-Flop
OFD8	Macro: 8-Bit Output D Flip-Flop
OFDE	Macro: D Flip-Flop with Active-High Enable Output Buffers
OFDE_1	Macro: D Flip-Flop with Active-High Enable Output Buffer and Inverted Clock
OFDE16	Macro: 16-Bit D Flip-Flop with Active-High Enable Output Buffers
OFDE4	Macro: 4-Bit D Flip-Flop with Active-High Enable Output Buffers
OFDE8	Macro: 8-Bit D Flip-Flop with Active-High Enable Output Buffers
OFDI	Macro: Output D Flip-Flop (Asynchronous Preset)
OFDI_1	Macro: Output D Flip-Flop with Inverted Clock (Asynchronous Preset)

Design Element	Description
OFDT	Macro: D Flip-Flop with Active-Low 3-State Output Buffer
OFDT_1	Macro: D Flip-Flop with Active-Low 3-State Output Buffer and Inverted Clock
OFDT16	Macro: 16-Bit D Flip-Flop with Active-Low 3-State Output Buffers
OFDT4	Macro: 4-Bit D Flip-Flop with Active-Low 3-State Output Buffers
OFDT8	Macro: 8-Bit D Flip-Flop with Active-Low 3-State Output Buffers
OFDX	Macro: Output D Flip-Flop with Clock Enable
OFDX_1	Macro: Output D Flip-Flop with Inverted Clock and Clock Enable
OFDX16	Macro: 16-Bit Output D Flip-Flop with Clock Enable
OFDX4	Macro: 4-Bit Output D Flip-Flop with Clock Enable
OFDX8	Macro: 8-Bit Output D Flip-Flop with Clock Enable
OFDXI	Macro: Output D Flip-Flop with Clock Enable (Asynchronous Preset)
OFDXI_1	Macro: Output D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)

IO Latch

Design Element	Description
ILD	Macro: Transparent Input Data Latch
ILD_1	Macro: Transparent Input Data Latch with Inverted Gate
ILD16	Macro: Transparent Input Data Latch
ILD4	Macro: Transparent Input Data Latch
ILD8	Macro: Transparent Input Data Latch
ILDI	Macro: Transparent Input Data Latch (Asynchronous Preset)
ILDI_1	Macro: Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)
ILDX	Macro: Transparent Input Data Latch
ILDX_1	Macro: Transparent Input Data Latch with Inverted Gate
ILDX16	Macro: Transparent Input Data Latch
ILDX4	Macro: Transparent Input Data Latch
ILDX8	Macro: Transparent Input Data Latch
ILDXI	Macro: Transparent Input Data Latch (Asynchronous Preset)
ILDXI_1	Macro: Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)

Latch

Design Element	Description
LD	Primitive: Transparent Data Latch
LD_1	Primitive: Transparent Data Latch with Inverted Gate
LD16	Macro: Multiple Transparent Data Latch

Design Element	Description
LD16CE	Macro: Transparent Data Latch with Asynchronous Clear and Gate Enable
LD4	Macro: Multiple Transparent Data Latch
LD4CE	Macro: Transparent Data Latch with Asynchronous Clear and Gate Enable
LD8	Macro: Multiple Transparent Data Latch
LD8CE	Macro: Transparent Data Latch with Asynchronous Clear and Gate Enable
LDC	Primitive: Transparent Data Latch with Asynchronous Clear
LDC_1	Primitive: Transparent Data Latch with Asynchronous Clear and Inverted Gate
LDCE	Primitive: Transparent Data Latch with Asynchronous Clear and Gate Enable
LDCE_1	Primitive: Transparent Data Latch with Asynchronous Clear, Gate Enable, and Inverted Gate
LDCP	Primitive: Transparent Data Latch with Asynchronous Clear and Preset
LDCP_1	Primitive: Transparent Data Latch with Asynchronous Clear and Preset and Inverted Gate
LDCPE	Primitive: Transparent Data Latch with Asynchronous Clear and Preset and Gate Enable
LDCPE_1	Primitive: Transparent Data Latch with Asynchronous Clear and Preset, Gate Enable, and Inverted Gate
LDE	Primitive: Transparent Data Latch with Gate Enable
LDE_1	Primitive: Transparent Data Latch with Gate Enable and Inverted Gate
LDP	Primitive: Transparent Data Latch with Asynchronous Preset
LDP_1	Primitive: Transparent Data Latch with Asynchronous Preset and Inverted Gate
LDPE	Primitive: Transparent Data Latch with Asynchronous Preset and Gate Enable
LDPE_1	Primitive: Transparent Data Latch with Asynchronous Preset, Gate Enable, and Inverted Gate

Logic

Design Element	Description
AND12	Macro: 12- Input AND Gate with Non-Inverted Inputs
AND16	Macro: 16- Input AND Gate with Non-Inverted Inputs
AND2	Primitive: 2-Input AND Gate with Non-Inverted Inputs
AND2B1	Primitive: 2-Input AND Gate with 1 Inverted and 1 Non-Inverted Inputs
AND2B2	Primitive: 2-Input AND Gate with Inverted Inputs
AND3	Primitive: 3-Input AND Gate with Non-Inverted Inputs
AND3B1	Primitive: 3-Input AND Gate with 1 Inverted and 2 Non-Inverted Inputs
AND3B2	Primitive: 3-Input AND Gate with 2 Inverted and 1 Non-Inverted Inputs
AND3B3	Primitive: 3-Input AND Gate with Inverted Inputs
AND4	Primitive: 4-Input AND Gate with Non-Inverted Inputs
AND4B1	Primitive: 4-Input AND Gate with 1 Inverted and 3 Non-Inverted Inputs
AND4B2	Primitive: 4-Input AND Gate with 2 Inverted and 2 Non-Inverted Inputs

Design Element	Description
AND4B3	Primitive: 4-Input AND Gate with 3 Inverted and 1 Non-Inverted Inputs
AND4B4	Primitive: 4-Input AND Gate with Inverted Inputs
AND5	Primitive: 5-Input AND Gate with Non-Inverted Inputs
AND5B1	Primitive: 5-Input AND Gate with 1 Inverted and 4 Non-Inverted Inputs
AND5B2	Primitive: 5-Input AND Gate with 2 Inverted and 3 Non-Inverted Inputs
AND5B3	Primitive: 5-Input AND Gate with 3 Inverted and 2 Non-Inverted Inputs
AND5B4	Primitive: 5-Input AND Gate with 4 Inverted and 1 Non-Inverted Inputs
AND5B5	Primitive: 5-Input AND Gate with Inverted Inputs
AND6	Macro: 6-Input AND Gate with Non-Inverted Inputs
AND7	Macro: 7-Input AND Gate with Non-Inverted Inputs
AND8	Macro: 8-Input AND Gate with Non-Inverted Inputs
AND9	Macro: 9-Input AND Gate with Non-Inverted Inputs
INV	Primitive: Inverter
INV16	Macro: 16 Inverters
INV4	Macro: Four Inverters
INV8	Macro: Eight Inverters
MULT_AND	Primitive: Fast Multiplier AND
NAND12	Macro: 12- Input NAND Gate with Non-Inverted Inputs
NAND16	Macro: 16- Input NAND Gate with Non-Inverted Inputs
NAND2	Primitive: 2-Input NAND Gate with Non-Inverted Inputs
NAND2B1	Primitive: 2-Input NAND Gate with 1 Inverted and 1 Non-Inverted Inputs
NAND2B2	Primitive: 2-Input NAND Gate with Inverted Inputs
NAND3	Primitive: 3-Input NAND Gate with Non-Inverted Inputs
NAND3B1	Primitive: 3-Input NAND Gate with 1 Inverted and 2 Non-Inverted Inputs
NAND3B2	Primitive: 3-Input NAND Gate with 2 Inverted and 1 Non-Inverted Inputs
NAND3B3	Primitive: 3-Input NAND Gate with Inverted Inputs
NAND4	Primitive: 4-Input NAND Gate with Non-Inverted Inputs
NAND4B1	Primitive: 4-Input NAND Gate with 1 Inverted and 3 Non-Inverted Inputs
NAND4B2	Primitive: 4-Input NAND Gate with 2 Inverted and 2 Non-Inverted Inputs
NAND4B3	Primitive: 4-Input NAND Gate with 3 Inverted and 1 Non-Inverted Inputs
NAND4B4	Primitive: 4-Input NAND Gate with Inverted Inputs
NAND5	Primitive: 5-Input NAND Gate with Non-Inverted Inputs
NAND5B1	Primitive: 5-Input NAND Gate with 1 Inverted and 4 Non-Inverted Inputs
NAND5B2	Primitive: 5-Input NAND Gate with 2 Inverted and 3 Non-Inverted Inputs
NAND5B3	Primitive: 5-Input NAND Gate with 3 Inverted and 2 Non-Inverted Inputs
NAND5B4	Primitive: 5-Input NAND Gate with 4 Inverted and 1 Non-Inverted Inputs
NAND5B5	Primitive: 5-Input NAND Gate with Inverted Inputs
NAND6	Macro: 6-Input NAND Gate with Non-Inverted Inputs

Design Element	Description
NAND7	Macro: 7-Input NAND Gate with Non-Inverted Inputs
NAND8	Macro: 8-Input NAND Gate with Non-Inverted Inputs
NAND9	Macro: 9-Input NAND Gate with Non-Inverted Inputs
NOR12	Macro: 12-Input NOR Gate with Non-Inverted Inputs
NOR16	Macro: 16-Input NOR Gate with Non-Inverted Inputs
NOR2	Primitive: 2-Input NOR Gate with Non-Inverted Inputs
NOR2B1	Primitive: 2-Input NOR Gate with 1 Inverted and 1 Non-Inverted Inputs
NOR2B2	Primitive: 2-Input NOR Gate with Inverted Inputs
NOR3	Primitive: 3-Input NOR Gate with Non-Inverted Inputs
NOR3B1	Primitive: 3-Input NOR Gate with 1 Inverted and 2 Non-Inverted Inputs
NOR3B2	Primitive: 3-Input NOR Gate with 2 Inverted and 1 Non-Inverted Inputs
NOR3B3	Primitive: 3-Input NOR Gate with Inverted Inputs
NOR4	Primitive: 4-Input NOR Gate with Non-Inverted Inputs
NOR4B1	Primitive: 4-Input NOR Gate with 1 Inverted and 3 Non-Inverted Inputs
NOR4B2	Primitive: 4-Input NOR Gate with 2 Inverted and 2 Non-Inverted Inputs
NOR4B3	Primitive: 4-Input NOR Gate with 3 Inverted and 1 Non-Inverted Inputs
NOR4B4	Primitive: 4-Input NOR Gate with Inverted Inputs
NOR5	Primitive: 5-Input NOR Gate with Non-Inverted Inputs
NOR5B1	Primitive: 5-Input NOR Gate with 1 Inverted and 4 Non-Inverted Inputs
NOR5B2	Primitive: 5-Input NOR Gate with 2 Inverted and 3 Non-Inverted Inputs
NOR5B3	Primitive: 5-Input NOR Gate with 3 Inverted and 2 Non-Inverted Inputs
NOR5B4	Primitive: 5-Input NOR Gate with 4 Inverted and 1 Non-Inverted Inputs
NOR5B5	Primitive: 5-Input NOR Gate with Inverted Inputs
NOR6	Macro: 6-Input NOR Gate with Non-Inverted Inputs
NOR7	Macro: 7-Input NOR Gate with Non-Inverted Inputs
NOR8	Macro: 8-Input NOR Gate with Non-Inverted Inputs
NOR9	Macro: 9-Input NOR Gate with Non-Inverted Inputs
OR12	Macro: 12-Input OR Gate with Non-Inverted Inputs
OR16	Macro: 16-Input OR Gate with Non-Inverted Inputs
OR2	Primitive: 2-Input OR Gate with Non-Inverted Inputs
OR2B1	Primitive: 2-Input OR Gate with 1 Inverted and 1 Non-Inverted Inputs
OR2B2	Primitive: 2-Input OR Gate with Inverted Inputs
OR3	Primitive: 3-Input OR Gate with Non-Inverted Inputs
OR3B1	Primitive: 3-Input OR Gate with 1 Inverted and 2 Non-Inverted Inputs
OR3B2	Primitive: 3-Input OR Gate with 2 Inverted and 1 Non-Inverted Inputs
OR3B3	Primitive: 3-Input OR Gate with Inverted Inputs
OR4	Primitive: 4-Input OR Gate with Non-Inverted Inputs
OR4B1	Primitive: 4-Input OR Gate with 1 Inverted and 3 Non-Inverted Inputs

Design Element	Description
OR4B2	Primitive: 4-Input OR Gate with 2 Inverted and 2 Non-Inverted Inputs
OR4B3	Primitive: 4-Input OR Gate with 3 Inverted and 1 Non-Inverted Inputs
OR4B4	Primitive: 4-Input OR Gate with Inverted Inputs
OR5	Primitive: 5-Input OR Gate with Non-Inverted Inputs
OR5B1	Primitive: 5-Input OR Gate with 1 Inverted and 4 Non-Inverted Inputs
OR5B2	Primitive: 5-Input OR Gate with 2 Inverted and 3 Non-Inverted Inputs
OR5B3	Primitive: 5-Input OR Gate with 3 Inverted and 2 Non-Inverted Inputs
OR5B4	Primitive: 5-Input OR Gate with 4 Inverted and 1 Non-Inverted Inputs
OR5B5	Primitive: 5-Input OR Gate with Inverted Inputs
OR6	Macro: 6-Input OR Gate with Non-Inverted Inputs
OR7	Macro: 7-Input OR Gate with Non-Inverted Inputs
OR8	Macro: 8-Input OR Gate with Non-Inverted Inputs
OR9	Macro: 9-Input OR Gate with Non-Inverted Inputs
SOP3	Macro: Sum of Products
SOP3B1A	Macro: Sum of Products
SOP3B1B	Macro: Sum of Products
SOP3B2A	Macro: Sum of Products
SOP3B2B	Macro: Sum of Products
SOP3B3	Macro: Sum of Products
SOP4	Macro: Sum of Products
SOP4B1	Macro: Sum of Products
SOP4B2A	Macro: Sum of Products
SOP4B2B	Macro: Sum of Products
SOP4B3	Macro: Sum of Products
SOP4B4	Macro: Sum of Products
XNOR2	Primitive: 2-Input XNOR Gate with Non-Inverted Inputs
XNOR3	Primitive: 3-Input XNOR Gate with Non-Inverted Inputs
XNOR4	Primitive: 4-Input XNOR Gate with Non-Inverted Inputs
XNOR5	Primitive: 5-Input XNOR Gate with Non-Inverted Inputs
XNOR6	Macro: 6-Input XNOR Gate with Non-Inverted Inputs
XNOR7	Macro: 7-Input XNOR Gate with Non-Inverted Inputs
XNOR8	Macro: 8-Input XNOR Gate with Non-Inverted Inputs
XNOR9	Macro: 9-Input XNOR Gate with Non-Inverted Inputs
XOR2	Primitive: 2-Input XOR Gate with Non-Inverted Inputs
XOR3	Primitive: 3-Input XOR Gate with Non-Inverted Inputs
XOR4	Primitive: 4-Input XOR Gate with Non-Inverted Inputs
XOR5	Primitive: 5-Input XOR Gate with Non-Inverted Inputs
XOR6	Macro: 6-Input XOR Gate with Non-Inverted Inputs

Design Element	Description
XOR7	Macro: 7-Input XOR Gate with Non-Inverted Inputs
XOR8	Macro: 8-Input XOR Gate with Non-Inverted Inputs
XOR9	Macro: 9-Input XOR Gate with Non-Inverted Inputs

LUT

Design Element	Description
LUT1	Primitive: 1-Bit Look-Up-Table with General Output
LUT1_D	Primitive: 1-Bit Look-Up-Table with Dual Output
LUT1_L	Primitive: 1-Bit Look-Up-Table with Local Output
LUT2	Primitive: 2-Bit Look-Up-Table with General Output
LUT2_D	Primitive: 2-Bit Look-Up-Table with Dual Output
LUT2_L	Primitive: 2-Bit Look-Up-Table with Local Output
LUT3	Primitive: 3-Bit Look-Up-Table with General Output
LUT3_D	Primitive: 3-Bit Look-Up-Table with Dual Output
LUT3_L	Primitive: 3-Bit Look-Up-Table with Local Output
LUT4	Primitive: 4-Bit Look-Up-Table with General Output
LUT4_D	Primitive: 4-Bit Look-Up-Table with Dual Output
LUT4_L	Primitive: 4-Bit Look-Up-Table with Local Output

Map

Design Element	Description
FMAP	Primitive: F Function Generator Partitioning Control Symbol

Memory

Design Element	Description
RAM128X1S	Primitive: 128-Deep by 1-Wide Static Synchronous RAM
RAM128X1S_1	Primitive: 128-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM16X1D	Primitive: 16-Deep by 1-Wide Static Dual Port Synchronous RAM
RAM16X1D_1	Primitive: 16-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock
RAM16X1S	Primitive: 16-Deep by 1-Wide Static Synchronous RAM
RAM16X1S_1	Primitive: 16-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM16X2D	Macro: 16-Deep by 2-Wide Static Dual Port Synchronous RAM
RAM16X2S	Primitive: 16-Deep by 2-Wide Static Synchronous RAM
RAM16X4D	Macro: 16-Deep by 4-Wide Static Dual Port Synchronous RAM

Design Element	Description
RAM16X4S	Primitive: 16-Deep by 4-Wide Static Synchronous RAM
RAM16X8D	Macro: 16-Deep by 8-Wide Static Dual Port Synchronous RAM
RAM16X8S	Primitive: 16-Deep by 8-Wide Static Synchronous RAM
RAM32X1D	Primitive: 32-Deep by 1-Wide Static Dual Port Synchronous RAM
RAM32X1D_1	Primitive: 32-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock
RAM32X1S	Primitive: 32-Deep by 1-Wide Static Synchronous RAM
RAM32X1S_1	Primitive: 32-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM32X2S	Primitive: 32-Deep by 2-Wide Static Synchronous RAM
RAM32X4S	Primitive: 32-Deep by 4-Wide Static Synchronous RAM
RAM32X8S	Primitive: 32-Deep by 8-Wide Static Synchronous RAM
RAM64X1D	Primitive: 64-Deep by 1-Wide Dual Port Static Synchronous RAM
RAM64X1D_1	Primitive: 64-Deep by 1-Wide Dual Port Static Synchronous RAM with Negative-Edge Clock
RAM64X1S	Primitive: 64-Deep by 1-Wide Static Synchronous RAM
RAM64X1S_1	Primitive: 64-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM64X2S	Primitive: 64-Deep by 2-Wide Static Synchronous RAM
RAMB16_S1	Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 1-bit Port
RAMB16_S1_S1	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit Ports
RAMB16_S1_S18	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 18-bit Ports
RAMB16_S1_S2	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 2-bit Ports
RAMB16_S1_S36	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 36-bit Ports
RAMB16_S1_S4	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 4-bit Ports
RAMB16_S1_S9	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 9-bit Ports
RAMB16_S18	Primitive: 16K-bit Data + 2K-bit Parity Memory, Single-Port Synchronous Block RAM with 18-bit Port
RAMB16_S18_S18	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit Ports
RAMB16_S18_S36	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit and 36-bit Ports
RAMB16_S2	Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 2-bit Port
RAMB16_S2_S18	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 18-bit Ports
RAMB16_S2_S2	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit Ports

Design Element	Description
RAMB16_S2_S36	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 36-bit Ports
RAMB16_S2_S4	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 4-bit Ports
RAMB16_S2_S9	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 9-bit Ports
RAMB16_S36	Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 36-bit Port
RAMB16_S36_S36	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with Two 36-bit Ports
RAMB16_S4	Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 4-bit Port
RAMB16_S4_S18	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 18-bit Ports
RAMB16_S4_S36	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 36-bit Ports
RAMB16_S4_S4	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit Ports
RAMB16_S4_S9	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 9-bit Ports
RAMB16_S9	Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 9-bit Port
RAMB16_S9_S18	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 18-bit Ports
RAMB16_S9_S36	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 36-bit Ports
RAMB16_S9_S9	Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit Ports
RAMB4_S1	Primitive: 4K-bit Single-Port Synchronous Block RAM with Port Width Configured to 1 Bit
RAMB4_S1_S1	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit
RAMB4_S1_S16	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 16-bits
RAMB4_S1_S2	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 2-bits
RAMB4_S1_S4	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 4-bits
RAMB4_S1_S8	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 8-bits
RAMB4_S16	Primitive: 4096-Bit Single-Port Synchronous Block RAM with Port Width Configured to 16 Bits
RAMB4_S16_S16	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 16-bits
RAMB4_S2	Primitive: 4K-bit Single-Port Synchronous Block RAM with Port Width Configured to 2-bits
RAMB4_S2_S16	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 16-bits

Design Element	Description
RAMB4_S2_S2	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 2-bits
RAMB4_S2_S4	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 4-bits
RAMB4_S2_S8	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 8-bits
RAMB4_S4	Primitive: 4k-bit Single-Port Synchronous Block RAM with Port Width Configured to 4-bits
RAMB4_S4_S16	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 4-bits and 16-bits
RAMB4_S4_S4	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 4-bits and 4-bits
RAMB4_S4_S8	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 4-bits and 8-bits
RAMB4_S8	Primitive: 4k-bit Single-Port Synchronous Block RAM with Port Width Configured to 8-bits
RAMB4_S8_S16	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 8-bits and 16-bits
RAMB4_S8_S8	Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 8-bits
ROM128X1	Primitive: 128-Deep by 1-Wide ROM
ROM16X1	Primitive: 16-Deep by 1-Wide ROM
ROM256X1	Primitive: 256-Deep by 1-Wide ROM
ROM32X1	Primitive: 32-Deep by 1-Wide ROM
ROM64X1	Primitive: 64-Deep by 1-Wide ROM

Mux

Design Element	Description
M16_1E	Macro: 16-to-1 Multiplexer with Enable
M2_1	Macro: 2-to-1 Multiplexer
M2_1B1	Macro: 2-to-1 Multiplexer with D0 Inverted
M2_1B2	Macro: 2-to-1 Multiplexer with D0 and D1 Inverted
M2_1E	Macro: 2-to-1 Multiplexer with Enable
M4_1E	Macro: 4-to-1 Multiplexer with Enable
M8_1E	Macro: 8-to-1 Multiplexer with Enable
MUXF5	Primitive: 2-to-1 Look-Up Table Multiplexer with General Output
MUXF5_D	Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output
MUXF5_L	Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output
MUXF6	Primitive: 2-to-1 Look-Up Table Multiplexer with General Output
MUXF6_D	Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output
MUXF6_L	Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output

Design Element	Description
MUXF7	Primitive: 2-to-1 Look-Up Table Multiplexer with General Output
MUXF7_D	Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output
MUXF7_L	Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output
MUXF8	Primitive: 2-to-1 Look-Up Table Multiplexer with General Output
MUXF8_D	Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output
MUXF8_L	Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output

Shift Register

Design Element	Description
SR16CE	Macro: 16-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR16CLE	Macro: 16-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR16CLED	Macro: 16-Bit Shift Register with Clock Enable and Asynchronous Clear
SR16RE	Macro: 16-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR16RLE	Macro: 16-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR16RLED	Macro: 16-Bit Shift Register with Clock Enable and Synchronous Reset
SR4CE	Macro: 4-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR4CLE	Macro: 4-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR4CLED	Macro: 4-Bit Shift Register with Clock Enable and Asynchronous Clear
SR4RE	Macro: 4-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR4RLE	Macro: 4-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR4RLED	Macro: 4-Bit Shift Register with Clock Enable and Synchronous Reset
SR8CE	Macro: 8-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR8CLE	Macro: 8-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR8CLED	Macro: 8-Bit Shift Register with Clock Enable and Asynchronous Clear
SR8RE	Macro: 8-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR8RLE	Macro: 8-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR8RLED	Macro: 8-Bit Shift Register with Clock Enable and Synchronous Reset
SRL16	Primitive: 16-Bit Shift Register Look-Up-Table (LUT)
SRL16_1	Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock
SRL16E	Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Clock Enable

Design Element	Description
SRL16E_1	Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock and Clock Enable
SRLC16	Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry
SRLC16_1	Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry and Negative-Edge Clock
SRLC16E	Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry and Clock Enable
SRLC16E_1	Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry, Negative-Edge Clock, and Clock Enable

Shifter

Design Element	Description
BRLSHFT4	Macro: 4-Bit Barrel Shifter
BRLSHFT8	Macro: 8-Bit Barrel Shifter

About Design Elements

This section describes the design elements that can be used with this architecture. The design elements are organized alphabetically.

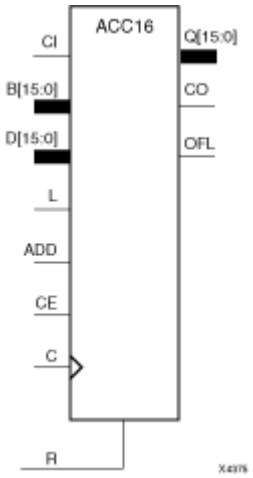
The following information is provided for each design element, where applicable:

- Name of element
- Brief description
- Schematic symbol (if any)
- Logic Table (if any)
- Port Descriptions (if any)
- Design Entry Method
- Available Attributes (if any)
- For more information

You can find examples of VHDL and Verilog instantiation code in the ISE software (in the main menu, select **Edit > Language Templates** or in the *Libraries Guide for HDL Designs* for this architecture.

ACC16

Macro: 16-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset



Introduction

This design element can add or subtract a 16-bit unsigned-binary, respectively or twos-complement word to or from the contents of a 16-bit data register and store the results in the register. The register can be loaded with the 16-bit word.

When the load input (L) is High, CE is ignored and the data on the D inputs is loaded into the register during the Low-to-High clock (C) transition. ACC16 loads the data on inputs D15 – D0 into the 16-bit register.

This design element operates on either 16-bit unsigned binary numbers or 16-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is how they determine when "overflow" occurs. Unsigned binary uses carry-out (CO), while twos complement uses OFL to determine when "overflow" occurs.

- For unsigned binary operation, ACC16 can represent numbers between 0 and 15, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds. The carry-out (CO) is not registered synchronously with the data outputs. CO always reflects the accumulation of the B inputs (B15 – B0 for ACC16). This allows the cascading of ACC16s by connecting CO of one stage to CI of the next stage. An unsigned binary "overflow" that is always active-High can be generated by gating the ADD signal and CO as follows:

  ```
  unsigned overflow = CO XOR ADD
  ```

  Ignore OFL in unsigned binary operation.

- For twos-complement operation, ACC16 represents numbers between -8 and +7, inclusive. If an addition or subtraction operation result exceeds this range, the OFL output goes High. The overflow (OFL) is not registered synchronously with the data outputs. OFL always reflects the accumulation of the B inputs (B15 – B0 for ACC16) and the contents of the register, which allows cascading of ACC4s by connecting OFL of one stage to CI of the next stage.

  Ignore CO in twos-complement operation.

The synchronous reset (R) has priority over all other inputs, and when set to High, causes all outputs to go to logic level zero during the Low-to-High clock (C) transition. Clock (C) transitions are ignored when clock enable (CE) is Low.

This design element is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Input						Output
R	**L**	**CE**	**ADD**	**D**	**C**	**Q**
1	x	x	x	x	↑	0
0	1	x	x	Dn	↑	Dn
0	0	1	1	x	↑	Q0+Bn+CI
0	0	1	0	x	↑	Q0-Bn-CI
0	0	0	x	x	↑	No Change

Q0: Previous value of Q

Bn: Value of Data input B

CI: Value of input CI

Design Entry Method

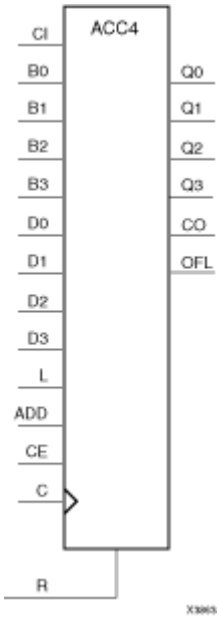
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ACC4

Macro: 4-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset



Introduction

This design element can add or subtract a 4-bit unsigned-binary, respectively or twos-complement word to or from the contents of a 4-bit data register and store the results in the register. The register can be loaded with the 4-bit word.

When the load input (L) is High, CE is ignored and the data on the D inputs is loaded into the register during the Low-to-High clock (C) transition. ACC4 loads the data on inputs D3 – D0 into the 4-bit register.

This design element operates on either 4-bit unsigned binary numbers or 4-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is how they determine when "overflow" occurs. Unsigned binary uses carry-out (CO), while twos complement uses OFL to determine when "overflow" occurs.

- For unsigned binary operation, ACC4 can represent numbers between 0 and 15, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds. The carry-out (CO) is not registered synchronously with the data outputs. CO always reflects the accumulation of the B inputs (B3 – B0 for ACC4). This allows the cascading of ACC4s by connecting CO of one stage to CI of the next stage. An unsigned binary "overflow" that is always active-High can be generated by gating the ADD signal and CO as follows:

  ```
  unsigned overflow = CO XOR ADD
  ```

  Ignore OFL in unsigned binary operation.

- For twos-complement operation, ACC4 represents numbers between -8 and +7, inclusive. If an addition or subtraction operation result exceeds this range, the OFL output goes High. The overflow (OFL) is not registered synchronously with the data outputs. OFL always reflects the accumulation of the B inputs (B3 – B0 for ACC4) and the contents of the register, which allows cascading of ACC4s by connecting OFL of one stage to CI of the next stage.

Ignore CO in twos-complement operation.

The synchronous reset (R) has priority over all other inputs, and when set to High, causes all outputs to go to logic level zero during the Low-to-High clock (C) transition. Clock (C) transitions are ignored when clock enable (CE) is Low.

This design element is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Input						Output
R	**L**	**CE**	**ADD**	**D**	**C**	**Q**
1	x	x	x	x	↑	0
0	1	x	x	Dn	↑	Dn
0	0	1	1	x	↑	Q0+Bn+CI
0	0	1	0	x	↑	Q0-Bn-CI
0	0	0	x	x	↑	No Change
Q0: Previous value of Q Bn: Value of Data input B CI: Value of input CI						

Design Entry Method

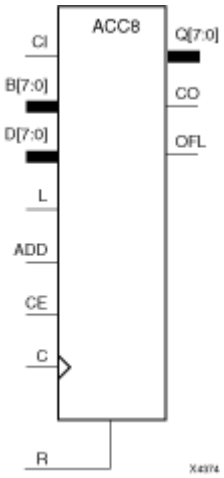
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ACC8

Macro: 8-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset



Introduction

This design element can add or subtract a 8-bit unsigned-binary, respectively or twos-complement word to or from the contents of a 8-bit data register and store the results in the register. The register can be loaded with the 8-bit word.

When the load input (L) is High, CE is ignored and the data on the D inputs is loaded into the register during the Low-to-High clock (C) transition. ACC8 loads the data on inputs D7 – D0 into the 8-bit register.

This design element operates on either 8-bit unsigned binary numbers or 8-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is how they determine when "overflow" occurs. Unsigned binary uses carry-out (CO), while twos complement uses OFL to determine when "overflow" occurs.

- For unsigned binary operation, ACC8 can represent numbers between 0 and 255, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds. The carry-out (CO) is not registered synchronously with the data outputs. CO always reflects the accumulation of the B inputs (B3 – B0 for ACC4). This allows the cascading of ACC8s by connecting CO of one stage to CI of the next stage. An unsigned binary "overflow" that is always active-High can be generated by gating the ADD signal and CO as follows:

  ```
  unsigned overflow = CO XOR ADD
  ```

  Ignore OFL in unsigned binary operation.

- For twos-complement operation, ACC8 represents numbers between -128 and +127, inclusive. If an addition or subtraction operation result exceeds this range, the OFL output goes High. The overflow (OFL) is not registered synchronously with the data outputs. OFL always reflects the accumulation of the B inputs (B3 – B0 for ACC8) and the contents of the register, which allows cascading of ACC8s by connecting OFL of one stage to CI of the next stage.

  Ignore CO in twos-complement operation.

The synchronous reset (R) has priority over all other inputs, and when set to High, causes all outputs to go to logic level zero during the Low-to-High clock (C) transition. Clock (C) transitions are ignored when clock enable (CE) is Low.

This design element is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Input						Output
R	**L**	**CE**	**ADD**	**D**	**C**	**Q**
1	x	x	x	x	↑	0
0	1	x	x	Dn	↑	Dn
0	0	1	1	x	↑	Q0+Bn+CI
0	0	1	0	x	↑	Q0-Bn-CI
0	0	0	x	x	↑	No Change
Q0: Previous value of Q Bn: Value of Data input B CI: Value of input CI						

Design Entry Method

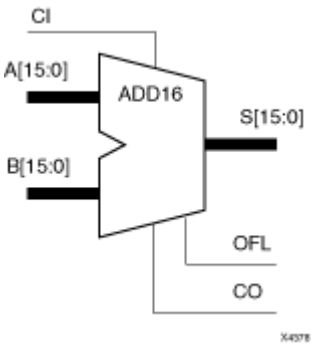
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ADD16

Macro: 16-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow



Introduction

This design element adds two words and a carry-in (CI), producing a sum output and carry-out (CO) or overflow (OFL). The factors added are A15 – A0, B15 – B0 and CI, producing the sum output S15 – S0 and CO (or OFL).

Logic Table

Input		Output
A	**B**	**S**
An	Bn	An+Bn+CI
CI: Value of input CI.		

Unsigned Binary Versus Twos Complement

This design element can operate on either 16-bit unsigned binary numbers or 16-bit twos-complement numbers, respectively. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is the way they determine when “overflow” occurs. Unsigned binary uses CO, while twos-complement uses OFL to determine when “overflow” occurs. To interpret the inputs as unsigned binary, follow the CO output. To interpret the inputs as twos complement, follow the OFL output.

Unsigned Binary Operation

For unsigned binary operation, this element represents numbers between 0 and 65535, inclusive. OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, this element can represent numbers between -32768 and +32767, inclusive. OFL is active (High) when the sum exceeds the bounds of the adder. CO is ignored in twos-complement operation.

Design Entry Method

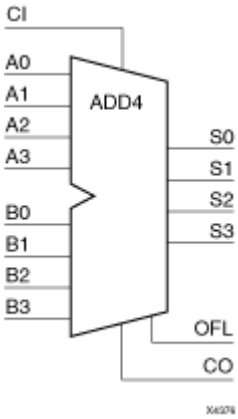
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ADD4

Macro: 4-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow



Introduction

This design element adds two words and a carry-in (CI), producing a sum output and carry-out (CO) or overflow (OFL). The factors added are A3 – A0, B3 – B0, and CI producing the sum output S3 – S0 and CO (or OFL).

Logic Table

Input		Output
A	**B**	**S**
An	Bn	An+Bn+CI
CI: Value of input CI.		

Unsigned Binary Versus Twos Complement

This design element can operate on either 4-bit unsigned binary numbers or 4-bit twos-complement numbers, respectively. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is the way they determine when "overflow" occurs. Unsigned binary uses CO, while twos-complement uses OFL to determine when "overflow" occurs. To interpret the inputs as unsigned binary, follow the CO output. To interpret the inputs as twos complement, follow the OFL output.

Unsigned Binary Operation

For unsigned binary operation, this element represents numbers from 0 to 15, inclusive. OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, this element can represent numbers between -8 and +7, inclusive. OFL is active (High) when the sum exceeds the bounds of the adder. CO is ignored in twos-complement operation.

Design Entry Method

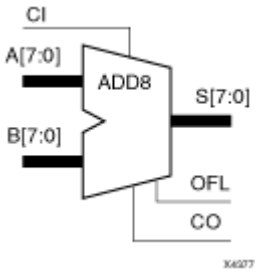
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ADD8

Macro: 8-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow



Introduction

This design element adds two words and a carry-in (CI), producing a sum output and carry-out (CO) or overflow (OFL). The factors added are A7 – A0, B7 – B0, and CI, producing the sum output S7 – S0 and CO (or OFL).

Logic Table

Input		Output
A	**B**	**S**
An	Bn	An+Bn+CI
CI: Value of input CI.		

Unsigned Binary Versus Twos Complement

This design element can operate on either 8-bit unsigned binary numbers or 8-bit twos-complement numbers, respectively. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is the way they determine when “overflow” occurs. Unsigned binary uses CO, while twos-complement uses OFL to determine when “overflow” occurs. To interpret the inputs as unsigned binary, follow the CO output. To interpret the inputs as twos complement, follow the OFL output.

Unsigned Binary Operation

For unsigned binary operation, this element represents numbers between 0 and 255, inclusive. OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, this element can represent numbers between -128 and +127, inclusive. OFL is active (High) when the sum exceeds the bounds of the adder. CO is ignored in twos-complement operation.

Design Entry Method

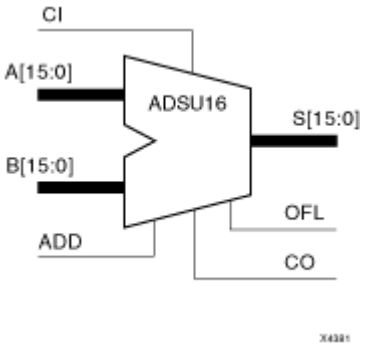
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ADSU16

Macro: 16-Bit Cascadable Adder/Subtracter with Carry-In, Carry-Out, and Overflow



Introduction

When the ADD input is High, this element adds two 16-bit words (A15 – A0 and B15 – B0) and a carry-in (CI), producing a 16-bit sum output (S15 – S0) and carry-out (CO) or overflow (OFL).

When the ADD input is Low, this element subtracts B15 – B0 from A15– A0, producing a difference output and a carry-out (CO) or an overflow (OFL).

In add mode, CO and CI are active-High. In subtract mode, CO and CI are active-Low. OFL is active-High in add and subtract modes.

Logic Table

Input			Output
ADD	**A**	**B**	**S**
1	An	Bn	An+Bn+CI*
0	An	Bn	An-Bn-CI*
CI*: ADD = 0, CI, CO active LOW CI*: ADD = 1, CI, CO active HIGH			

Unsigned Binary Versus Twos Complement

This design element can operate on either 16-bit unsigned binary numbers or 16-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is the way they determine when “overflow” occurs. Unsigned binary uses CO, while twos complement uses OFL to determine when “overflow” occurs.

With adder/subtracters, either unsigned binary or twos-complement operations cause an overflow. If the result crosses the overflow boundary, an overflow is generated. Similarly, when the result crosses the carry-out boundary, a carry-out is generated.

Unsigned Binary Operation

For unsigned binary operation, this element can represent numbers between 0 and 65535, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds.

An unsigned binary “overflow” that is always active-High can be generated by gating the ADD signal and CO as follows:

```
unsigned overflow = CO XOR ADD
```

OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, this element can represent numbers between -32768 and +32767, inclusive.

If an addition or subtraction operation result exceeds this range, the OFL output goes High. CO is ignored in twos-complement operation.

Design Entry Method

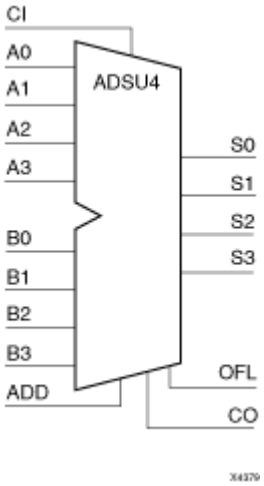
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ADSU4

Macro: 4-Bit Cascadable Adder/Subtracter with Carry-In, Carry-Out, and Overflow



Introduction

When the ADD input is High, this element adds two 4-bit words (A3 – A0 and B3 – B0) and a carry-in (CI), producing a 4-bit sum output (S3 – S0) and a carry-out (CO) or an overflow (OFL).

When the ADD input is Low, this element subtracts B3 – B0 from A3– A0, producing a 4-bit difference output (S3 – S0) and a carry-out (CO) or an overflow (OFL).

In add mode, CO and CI are active-High. In subtract mode, CO and CI are active-Low. OFL is active-High in add and subtract modes.

Logic Table

Input			Output
ADD	**A**	**B**	**S**
1	An	Bn	An+Bn+CI*
0	An	Bn	An-Bn-CI*
CI*: ADD = 0, CI, CO active LOW CI*: ADD = 1, CI, CO active HIGH			

Unsigned Binary Versus Twos Complement

This design element can operate on either 4-bit unsigned binary numbers or 4-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is the way they determine when "overflow" occurs. Unsigned binary uses CO, while twos complement uses OFL to determine when "overflow" occurs.

With adder/subtracters, either unsigned binary or twos-complement operations cause an overflow. If the result crosses the overflow boundary, an overflow is generated. Similarly, when the result crosses the carry-out boundary, a carry-out is generated.

Unsigned Binary Operation

For unsigned binary operation, ADSU4 can represent numbers between 0 and 15, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds.

An unsigned binary "overflow" that is always active-High can be generated by gating the ADD signal and CO as follows:

```
unsigned overflow = CO XOR ADD
```

OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, this element can represent numbers between -8 and +7, inclusive.

If an addition or subtraction operation result exceeds this range, the OFL output goes High. CO is ignored in twos-complement operation.

Design Entry Method

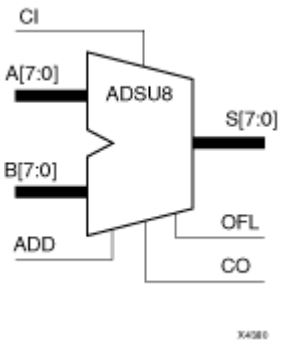
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ADSU8

Macro: 8-Bit Cascadable Adder/Subtracter with Carry-In, Carry-Out, and Overflow



Introduction

When the ADD input is High, this element adds two 8-bit words (A7 – A0 and B7 – B0) and a carry-in (CI), producing, an 8-bit sum output (S7 – S0) and carry-out (CO) or an overflow (OFL).

When the ADD input is Low, this element subtracts B7 – B0 from A7 – A0, producing an 8-bit difference output (S7 – S0) and a carry-out (CO) or an overflow (OFL).

In add mode, CO and CI are active-High. In subtract mode, CO and CI are active-Low. OFL is active-High in add and subtract modes.

Logic Table

Input			Output
ADD	**A**	**B**	**S**
1	An	Bn	An+Bn+CI*
0	An	Bn	An-Bn-CI*
CI*: ADD = 0, CI, CO active LOW CI*: ADD = 1, CI, CO active HIGH			

Unsigned Binary Versus Twos Complement

This design element can operate on either 8-bit unsigned binary numbers or 8-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is the way they determine when “overflow” occurs. Unsigned binary uses CO, while twos complement uses OFL to determine when “overflow” occurs.

With adder/subtracters, either unsigned binary or twos-complement operations cause an overflow. If the result crosses the overflow boundary, an overflow is generated. Similarly, when the result crosses the carry-out boundary, a carry-out is generated.

Unsigned Binary Operation

For unsigned binary operation, this element can represent numbers between 0 and 255, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds.

An unsigned binary “overflow” that is always active-High can be generated by gating the ADD signal and CO as follows:

```
unsigned overflow = CO XOR ADD
```

OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, this element can represent numbers between -128 and +127, inclusive.

If an addition or subtraction operation result exceeds this range, the OFL output goes High. CO is ignored in twos-complement operation.

Design Entry Method

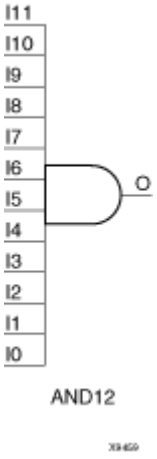
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND12

Macro: 12- Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

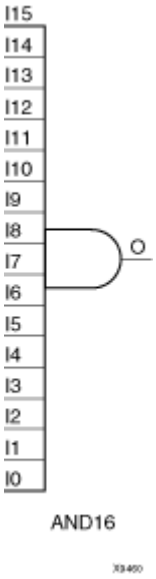
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND16

Macro: 16- Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND2

Primitive: 2-Input AND Gate with Non-Inverted Inputs

Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

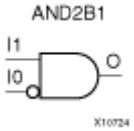
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND2B1

Primitive: 2-Input AND Gate with 1 Inverted and 1 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

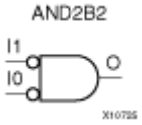
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND2B2

Primitive: 2-Input AND Gate with Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND3

Primitive: 3-Input AND Gate with Non-Inverted Inputs

Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

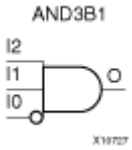
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND3B1

Primitive: 3-Input AND Gate with 1 Inverted and 2 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

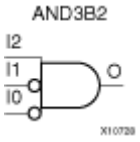
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND3B2

Primitive: 3-Input AND Gate with 2 Inverted and 1 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

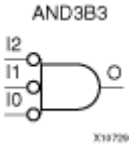
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND3B3

Primitive: 3-Input AND Gate with Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

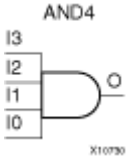
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND4

Primitive: 4-Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

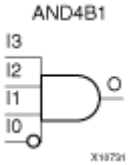
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND4B1

Primitive: 4-Input AND Gate with 1 Inverted and 3 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

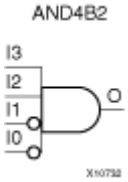
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND4B2

Primitive: 4-Input AND Gate with 2 Inverted and 2 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

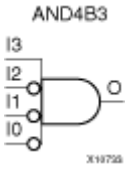
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND4B3

Primitive: 4-Input AND Gate with 3 Inverted and 1 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

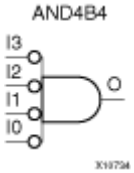
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND4B4

Primitive: 4-Input AND Gate with Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

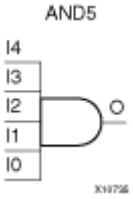
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND5

Primitive: 5-Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

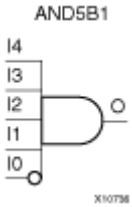
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND5B1

Primitive: 5-Input AND Gate with 1 Inverted and 4 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

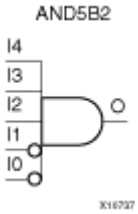
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND5B2

Primitive: 5-Input AND Gate with 2 Inverted and 3 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

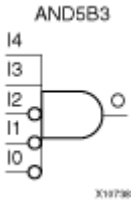
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND5B3

Primitive: 5-Input AND Gate with 3 Inverted and 2 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

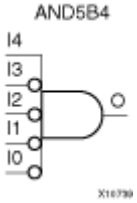
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND5B4

Primitive: 5-Input AND Gate with 4 Inverted and 1 Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

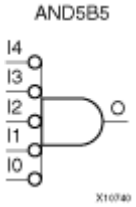
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND5B5

Primitive: 5-Input AND Gate with Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

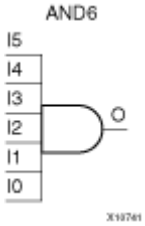
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND6

Macro: 6-Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

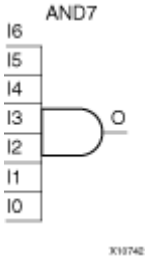
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND7

Macro: 7-Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

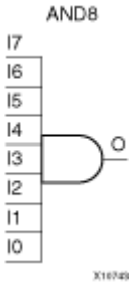
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND8

Macro: 8-Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

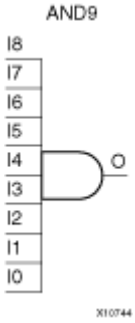
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

AND9

Macro: 9-Input AND Gate with Non-Inverted Inputs



Introduction

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs, 12 inputs, and 16 inputs are available with noninverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the appropriate number of inputs.

Design Entry Method

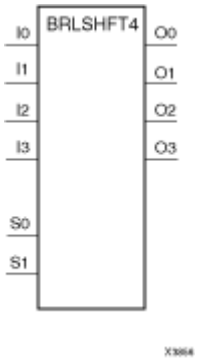
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BRLSHFT4

Macro: 4-Bit Barrel Shifter



Introduction

This design element is a 4-bit barrel shifter that can rotate four inputs (I3 – I0) up to four places. The control inputs (S1 and S0) determine the number of positions, from one to four, that the data is rotated. The four outputs (O3 – O0) reflect the shifted data inputs.

Logic Table

Inputs						Outputs			
S1	**S0**	**I0**	**I1**	**I2**	**I3**	**O0**	**O1**	**O2**	**O3**
0	0	a	b	c	d	a	b	c	d
0	1	a	b	c	d	b	c	d	a
1	0	a	b	c	d	c	d	a	b
1	1	a	b	c	d	d	a	b	c

Design Entry Method

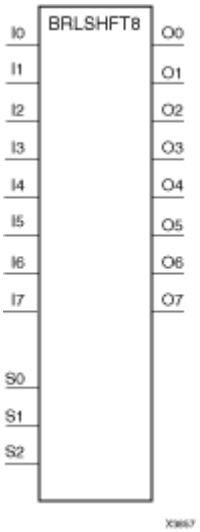
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BRLSHFT8

Macro: 8-Bit Barrel Shifter



Introduction

This design element is an 8-bit barrel shifter, can rotate the eight inputs (I7 – I0) up to eight places. The control inputs (S2 – S0) determine the number of positions, from one to eight, that the data is rotated. The eight outputs (O7 – O0) reflect the shifted data inputs.

Logic Table

Inputs											Outputs							
S2	S1	S0	I0	I1	I2	I3	I4	I5	I6	I7	O0	O1	O2	O3	O4	O5	O6	O7
0	0	0	a	b	c	d	e	f	g	h	a	b	c	d	e	f	g	h
0	0	1	a	b	c	d	e	f	g	h	b	c	d	e	f	g	h	a
0	1	0	a	b	c	d	e	f	g	h	c	d	e	f	g	h	a	b
0	1	1	a	b	c	d	e	f	g	h	d	e	f	g	h	a	b	c
1	0	0	a	b	c	d	e	f	g	h	e	f	g	h	a	b	c	d
1	0	1	a	b	c	d	e	f	g	h	f	g	h	a	b	c	d	e
1	1	0	a	b	c	d	e	f	g	h	g	h	a	b	c	d	e	f
1	1	1	a	b	c	d	e	f	g	h	h	a	b	c	d	e	f	g

Design Entry Method

This design element is only for use in schematics.

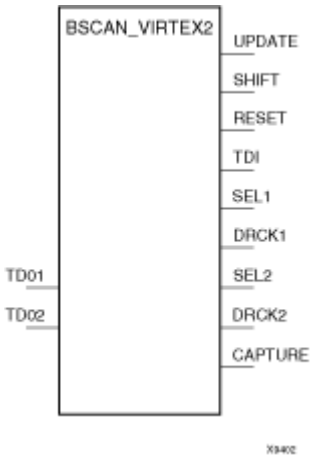
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

BSCAN_VIRTEX2

Primitive: Virtex2 Boundary Scan Logic Control Circuit



Introduction

This design element provides access to the BSCAN sites on a Virtex-II, Virtex-II Pro, or Virtex-II Pro X device. It is used to create internal boundary scan chains. The 4-pin JTAG interface (TDI, TDO, TCK, and TMS) are dedicated pins in Virtex-II, Virtex-II Pro, and Virtex-II Pro X. To use normal JTAG for boundary scan purposes, just hook up the JTAG pins to the port and go. The pins on the this design element do not need to be connected, unless those special functions are needed to drive an internal scan chain.

A signal on the TDO1 input is passed to the external TDO output when the USER1 instruction is executed; the SEL1 output goes High to indicate that the USER1 instruction is active. The DRCK1 output provides USER1 access to the data register clock (generated by the TAP controller). The TDO2 and SEL2 pins perform a similar function for the USER2 instruction and the DRCK2 output provides USER2 access to the data register clock (generated by the TAP controller). The RESET, UPDATE, SHIFT, and CAPTURE pins represent the decoding of the corresponding state of the boundary scan internal state machine. The TDI pin provides access to the TDI signal of the JTAG port in order to shift data into an internal scan chain.

Note For specific information on boundary scan for an architecture, see *The Programmable Logic Data Sheets*

Design Entry Method

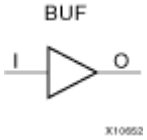
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUF

Primitive: General Purpose Buffer



Introduction

This is a general-purpose, non-inverting buffer.

This element is not necessary and is removed by the partitioning software (MAP).

Design Entry Method

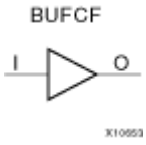
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFCF

Primitive: Fast Connect Buffer



Introduction

This design element is a single fast connect buffer used to connect the outputs of the LUTs and some dedicated logic directly to the input of another LUT. Using this buffer implies CLB packing. No more than four LUTs may be connected together as a group.

Design Entry Method

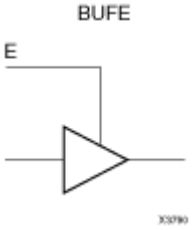
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFE

Primitive: Internal 3-State Buffer with Active High Enable



Introduction

This design element is a single, 3-state buffer with input I and output O, and an active-High output enable (E). When E is High, data on the input of the buffer is transferred to the corresponding output. When E is Low, the output is high impedance (Z state or Off). The outputs of the buffers are connected to horizontal longlines in FPGA architectures.

The outputs of separate symbols for this entity can be tied together to form a bus or a multiplexer. Make sure that only one E is High at any one time. If none of the E inputs is active-High, a "weak-keeper" circuit keeps the output bus from floating but does not guarantee that the bus remains at the last value driven onto it. For certain CPLD devices, output from nets assume the High logic level when all connected BUFE/BUFT buffers are disabled. For FPGA devices, elements need a PULLUP element connected to their output. NGDBuild inserts a PULLUP element if one is not connected.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

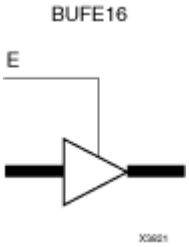
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFE16

Macro: 16-Bit Internal 3-State Buffer with Active High Enable



Introduction

This design element is a multiple 3-state buffer with inputs of I15 – I0 and outputs of O15 – O0 and an active-High output enable (E). When E is High, data on the inputs of the buffers is transferred to the corresponding outputs.

When E is Low, the output is high impedance (Z state or Off). The outputs of the buffers are connected to horizontal longlines in FPGA architectures. The outputs of separate BUFE elements can be tied together to form a bus or a multiplexer. Make sure that only one E is High at any one time. If none of the E inputs is active-High, a "weak-keeper" circuit keeps the output bus from floating but does not guarantee that the bus remains at the last value driven onto it.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

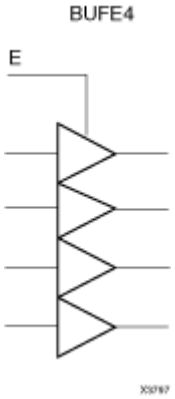
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFE4

Macro: 4-BitInternal 3-State Buffer with Active High Enable



Introduction

This design element is a multiple 3-state buffer with inputs of I3 – I0 and outputs of O3 – O0 and an active-High output enable (E). When E is High, data on the inputs of the buffers is transferred to the corresponding outputs.

When E is Low, the output is high impedance (Z state or Off). The outputs of the buffers are connected to horizontal longlines in FPGA architectures. The outputs of separate BUFE elements can be tied together to form a bus or a multiplexer. Make sure that only one E is High at any one time. If none of the E inputs is active-High, a "weak-keeper" circuit keeps the output bus from floating but does not guarantee that the bus remains at the last value driven onto it.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

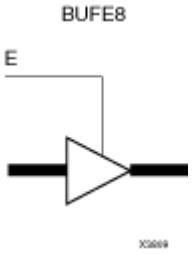
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFE8

Macro: 8-Bit Internal 3-State Buffer with Active High Enable



Introduction

This design element is a multiple 3-state buffer with inputs of I7 – I0 and outputs of O7 – O0 and an active-High output enable (E). When E is High, data on the inputs of the buffers is transferred to the corresponding outputs.

When E is Low, the output is high impedance (Z state or Off). The outputs of the buffers are connected to horizontal longlines in FPGA architectures. The outputs of separate BUFE elements can be tied together to form a bus or a multiplexer. Make sure that only one E is High at any one time. If none of the E inputs is active-High, a "weak-keeper" circuit keeps the output bus from floating but does not guarantee that the bus remains at the last value driven onto it.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFG

Primitive: Global Clock Buffer

Introduction

This design element is a high-fanout buffer that connects signals to the global routing resources for low skew distribution of the signal. BUFGs are typically used on clock nets as well other high fanout nets like sets/resets and clock enables.

Design Entry Method

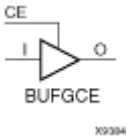
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFGCE

Primitive: Global Clock Buffer with Clock Enable



Introduction

This design element is a global clock buffer with a single gated input. Its O output is "0" when clock enable (CE) is Low (inactive). When clock enable (CE) is High, the I input is transferred to the O output.

Logic Table

Inputs		Outputs
I	**CE**	**O**
X	0	0
I	1	I

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFGCE_1

Primitive: Global Clock Buffer with Clock Enable and Output State 1

Introduction

This design element is a multiplexed global clock buffer with a single gated input. Its O output is High (1) when clock enable (CE) is Low (inactive). When clock enable (CE) is High, the I input is transferred to the O output.

Logic Table

Inputs		Outputs
I	**CE**	**O**
X	0	1
I	1	I

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFGMUX

Primitive: Global Clock MUX Buffer

Introduction

BUFGMUX is a multiplexed global clock buffer, based off of the BUFGCTRL, that can select between two input clocks: I0 and I1. When the select input (S) is Low, the signal on I0 is selected for output (O). When the select input (S) is High, the signal on I1 is selected for output.

BUFGMUX and BUFGMUX_1 are distinguished by the state the output assumes when that output switches between clocks in response to a change in its select input. BUGFMUX assumes output state 0 and BUFGMUX_1 assumes output state 1.

Note BUFGMUX guarantees that when S is toggled, the state of the output remains in the inactive state until the next active clock edge (either I0 or I1) occurs.

Logic Table

Inputs			Outputs
I0	**I1**	**S**	**O**
I0	X	0	I0
X	I1	1	I1
X	X	↑	0
X	X	↓	0

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFGMUX_1

Primitive: Global Clock MUX Buffer with Output State 1

Introduction

This design element is a multiplexed global clock buffer that can select between two input clocks: I0 and I1. When the select input (S) is Low, the signal on I0 is selected for output (0). When the select input (S) is High, the signal on I1 is selected for output.

This design element is distinguished from BUFGMUX by the state the output assumes when that output switches between clocks in response to a change in its select input. BUFGMUX assumes output state 0 and BUFGMUX_1 assumes output state 1.

Logic Table

Inputs			Outputs
I0	**I1**	**S**	**O**
I0	X	0	I0
X	I1	1	I1
X	X	↑	1
X	X	↓	1

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFGP

Primitive: Primary Global Buffer for Driving Clocks or Longlines (Four per PLD Device)

Introduction

This design element is a primary global buffer that is used to distribute high fan-out clock or control signals throughout in FPGA devices. It is equivalent to an IBUFG driving a BUFG.

This design element provides direct access to Configurable Logic Block (CLB) and Input Output Block (IOB) clock pins and limited access to other CLB inputs. The input to a BUFGP comes only from a dedicated IOB. Because of its structure, this element can always access a clock pin directly. However, it can access only one of the F3, G1, C3, or C1 pins, depending on the corner in which the BUFGP is placed. When the required pin cannot be accessed directly from the vertical line, PAR feeds the signal through another CLB and uses general purpose routing to access the load pin.

Design Entry Method

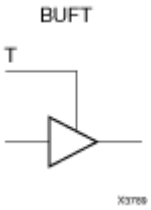
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFT

Primitive: Internal 3-State Buffer with Active Low Enable



Introduction

This design element is a single 3-state buffer with input I and an output of O and active-Low output enable (T). When T is Low, data on the input of the buffer is transferred to the corresponding output. When T is High, the output is high impedance (Z state or off). The output of the buffer is connected to a horizontal longline in FPGA architectures.

The output of separate BUFT symbols can be tied together to form a bus or a multiplexer. Make sure that only one T is Low at one time. For CPLD devices, BUFT output nets assume the High logic level when all connected BUFE/BUFT buffers are disabled. For FPGAs, when all BUFTs on a net are disabled, the net is High. For correct simulation of this effect, a PULLUP element must be connected to the net. NGDBuild inserts a PULLUP element if one is not connected so that back-annotation simulation reflects the true state of the device.

Logic Table

Inputs		Outputs
T	**I**	**O**
1	X	Z
0	1	1
0	0	0

Design Entry Method

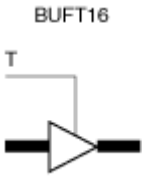
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFT16

Macro: 16-Bit Internal 3-State Buffers with Active Low Enable



Introduction

This design element is a multiple 3-state buffer with inputs I15 – 10 and outputs O15 – O0 and active-Low output enable (T). When T is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When T is High, the output is high impedance (Z state or off). The outputs of the buffers are connected to horizontal longlines in FPGA architectures.

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Logic Table

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Design Entry Method

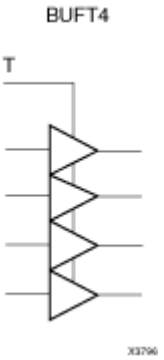
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFT4

Macro: 4-Bit Internal 3-State Buffers with Active Low Enable



Introduction

This design element is a multiple 3-state buffer with inputs I3 – I0 and outputs O3 – O0 and active-Low output enable (T). When T is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When T is High, the output is high impedance (Z state or off). The outputs of the buffers are connected to horizontal longlines in FPGA architectures.

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Logic Table

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Design Entry Method

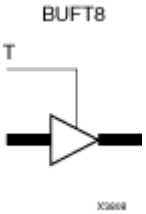
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

BUFT8

Macro: 8-Bit Internal 3-State Buffers with Active Low Enable



Introduction

This design element is a multiple 3-state buffer with inputs I7 – I0 and outputs O7 – O0 and active-Low output enable (T). When T is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When T is High, the output is high impedance (Z state or off). The outputs of the buffers are connected to horizontal longlines in FPGA architectures.

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Logic Table

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Design Entry Method

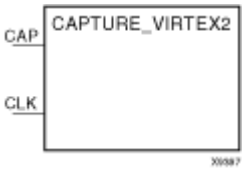
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CAPTURE_VIRTEX2

Primitive: Virtex4 Register State Capture for Bitstream Readback



Introduction

This element provides user control and synchronization over when and how the capture register (flip-flop and latch) information task is requested. The readback function is provided through dedicated configuration port instructions. However, without this element, the readback data is synchronized to the configuration clock. Only register (flip-flop and latch) states can be captured. Although LUT RAM, SRL, and block RAM states are readback, they cannot be captured.

An asserted high CAP signal indicates that the registers in the device are to be captured at the next Low-to-High clock transition. By default, data is captured after every trigger when transition on CLK while CAP is asserted. To limit the readback operation to a single data capture, add the ONESHOT=TRUE attribute to this element.

Port Descriptions

Port	Direction	Width	Function
CAP	Input	1	Readback capture trigger
CLK	Input	1	Readback capture clock

Design Entry Method

This design element can be used in schematics.

Connect all inputs and outputs to the design in order to ensure proper operation.

Available Attributes

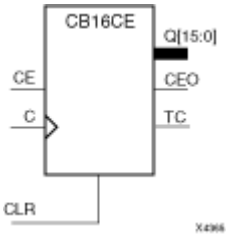
Attribute	Type	Allowed Values	Default	Description
ONESHOT	Boolean	TRUE, FALSE	TRUE	Specifies the procedure for performing single readback per CAP trigger.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB16CE

Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is an asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
CLR	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	X	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE					

Design Entry Method

This design element is only for use in schematics.

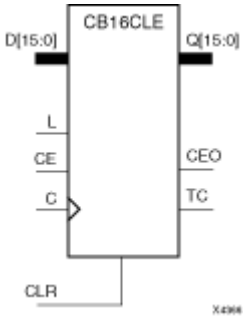
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

CB16CLE

Macro: 16-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This element is a synchronously loadable, asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock transition, independent of the state of clock enable (CE). The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs		
CLR	**L**	**CE**	**C**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No change	No change	0
0	0	1	↑	X	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE							

Design Entry Method

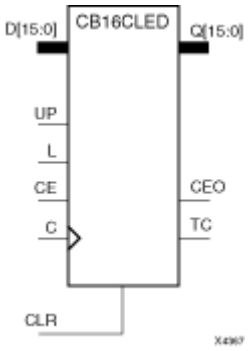
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB16CLED

Macro: 16-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable, bidirectional binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs decrement when CE is High and UP is Low during the Low-to- High clock transition. The Q outputs increment when CE and UP are High. The counter ignores clock transitions when CE is Low.

For counting up, the TC output is High when all Q outputs and UP are High. For counting down, the TC output is High when all Q outputs and UP are Low.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, UP, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

For CPLD parts, see “CB2X1”, “CB4X1”, “CB8X1”, “CB16X1” for high-performance cascadable, bidirectional counters.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	X	0	0	0
0	1	X	↑	X	Dn	Dn	TC	CEO
0	0	0	X	X	X	No change	No change	0
0	0	1	↑	1	X	Inc	TC	CEO

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
0	0	1	↑	0	X	Dec	TC	CEO
z = bit width - 1 TC = (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) + (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) CEO = TC•CE								

Design Entry Method

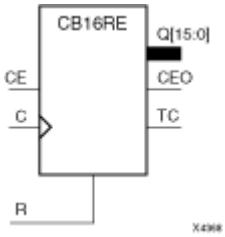
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB16RE

Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a synchronous, resettable, cascadable binary counter. The synchronous reset (R), when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to zero on the Low-to-High clock transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when both Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
R	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	↑	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0) CEO = TC•CE					

Design Entry Method

This design element is only for use in schematics.

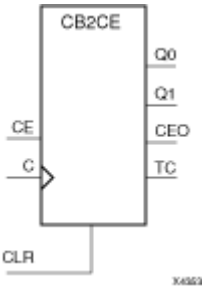
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

CB2CE

Macro: 2-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is an asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**		
CLR	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	X	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE					

Design Entry Method

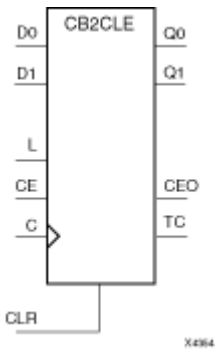
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB2CLE

Macro: 2-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This element is a synchronously loadable, asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock transition, independent of the state of clock enable (CE). The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs		
CLR	**L**	**CE**	**C**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No change	No change	0
0	0	1	↑	X	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE							

Design Entry Method

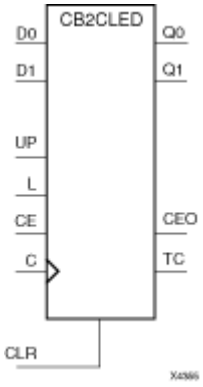
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB2CLED

Macro: 2-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable, bidirectional binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs decrement when CE is High and UP is Low during the Low-to- High clock transition. The Q outputs increment when CE and UP are High. The counter ignores clock transitions when CE is Low.

For counting up, the TC output is High when all Q outputs and UP are High. For counting down, the TC output is High when all Q outputs and UP are Low.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, UP, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

For CPLD parts, see “CB2X1”, “CB4X1”, “CB8X1”, “CB16X1” for high-performance cascadable, bidirectional counters.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	X	0	0	0
0	1	X	↑	X	Dn	Dn	TC	CEO
0	0	0	X	X	X	No change	No change	0
0	0	1	↑	1	X	Inc	TC	CEO

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
0	0	1	↑	0	X	Dec	TC	CEO
z = bit width - 1 TC = (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) + (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) CEO = TC•CE								

Design Entry Method

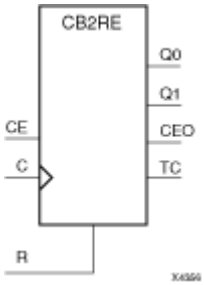
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB2RE

Macro: 2-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a synchronous, resettable, cascadable binary counter. The synchronous reset (R), when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to zero on the Low-to-High clock transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when both Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
R	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	↑	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0) CEO = TC•CE					

Design Entry Method

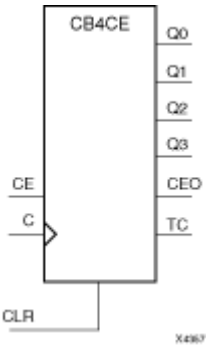
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB4CE

Macro: 4-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is an asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
CLR	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	X	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE					

Design Entry Method

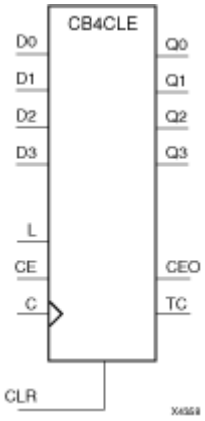
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB4CLE

Macro: 4-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This element is a synchronously loadable, asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock transition, independent of the state of clock enable (CE). The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					**Outputs**		
CLR	**L**	**CE**	**C**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No change	No change	0
0	0	1	↑	X	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE							

Design Entry Method

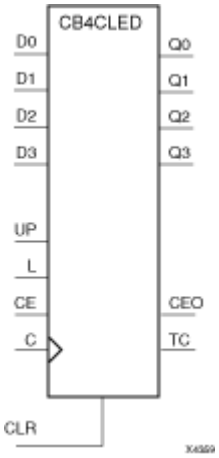
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB4CLED

Macro: 4-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable, bidirectional binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs decrement when CE is High and UP is Low during the Low-to- High clock transition. The Q outputs increment when CE and UP are High. The counter ignores clock transitions when CE is Low.

For counting up, the TC output is High when all Q outputs and UP are High. For counting down, the TC output is High when all Q outputs and UP are Low.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, UP, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

For CPLD parts, see “CB2X1”, “CB4X1”, “CB8X1”, “CB16X1” for high-performance cascadable, bidirectional counters.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	X	0	0	0
0	1	X	↑	X	Dn	Dn	TC	CEO
0	0	0	X	X	X	No change	No change	0

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
0	0	1	↑	1	X	Inc	TC	CEO
0	0	1	↑	0	X	Dec	TC	CEO
z = bit width - 1 TC = (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) + (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) CEO = TC•CE								

Design Entry Method

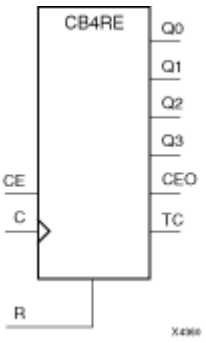
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB4RE

Macro: 4-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a synchronous, resettable, cascadable binary counter. The synchronous reset (R), when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to zero on the Low-to-High clock transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when both Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
R	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	↑	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0) CEO = TC•CE					

Design Entry Method

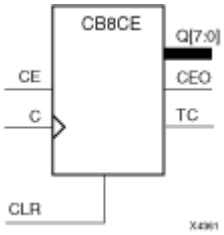
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB8CE

Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is an asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
CLR	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	X	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE					

Design Entry Method

This design element is only for use in schematics.

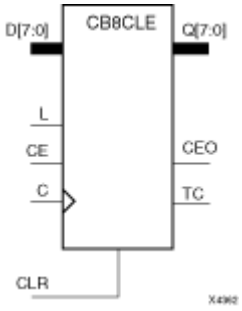
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

CB8CLE

Macro: 8-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This element is a synchronously loadable, asynchronously clearable, cascadable binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock transition, independent of the state of clock enable (CE). The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs		
CLR	**L**	**CE**	**C**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No change	No change	0
0	0	1	↑	X	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE							

Design Entry Method

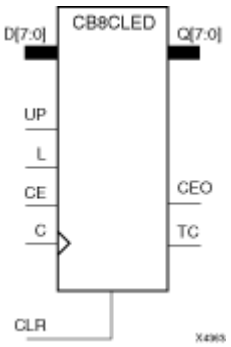
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB8CLED

Macro: 8-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable, bidirectional binary counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs decrement when CE is High and UP is Low during the Low-to- High clock transition. The Q outputs increment when CE and UP are High. The counter ignores clock transitions when CE is Low.

For counting up, the TC output is High when all Q outputs and UP are High. For counting down, the TC output is High when all Q outputs and UP are Low.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, UP, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

For CPLD parts, see “CB2X1”, “CB4X1”, “CB8X1”, “CB16X1” for high-performance cascadable, bidirectional counters.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	X	0	0	0
0	1	X	↑	X	Dn	Dn	TC	CEO
0	0	0	X	X	X	No change	No change	0
0	0	1	↑	1	X	Inc	TC	CEO

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
0	0	1	↑	0	X	Dec	TC	CEO
z = bit width - 1 TC = (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) + (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) CEO = TC•CE								

Design Entry Method

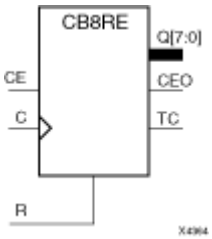
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CB8RE

Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a synchronous, resettable, cascadable binary counter. The synchronous reset (R), when High, overrides all other inputs and forces the Q outputs, terminal count (TC), and clock enable out (CEO) to zero on the Low-to-High clock transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when both Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
R	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	↑	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0) CEO = TC•CE					

Design Entry Method

This design element is only for use in schematics.

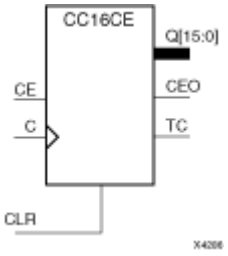
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

CC16CE

Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is an asynchronously clearable, cascadable binary counter. It is implemented using carry logic with relative location constraints to ensure efficient logic placement. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
CLR	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	X	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE					

Design Entry Method

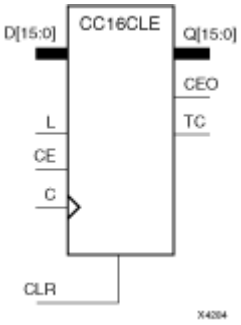
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CC16CLE

Macro: 16-Bit Loadable Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable binary counter. It is implemented using carry logic with relative location constraints to ensure efficient logic placement. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					**Outputs**		
CLR	**L**	**CE**	**C**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No change	No change	0
0	0	1	↑	X	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE							

Design Entry Method

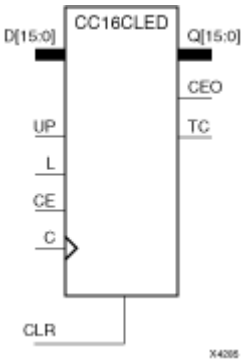
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CC16CLED

Macro: 16-Bit Loadable Cascadable Bidirectional Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable, bidirectional binary counter. It is implemented using carry logic with relative location constraints, which assures most efficient logic placement. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs decrement when CE is High and UP is Low during the Low-to-High clock transition. The Q outputs increment when CE and UP are High. The counter ignores clock transitions when CE is Low.

For counting up, the TC output is High when all Q outputs and UP are High. For counting down, the TC output is High when all Q outputs and UP are Low.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, UP, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	X	0	0	0
0	1	X	↑	X	Dn	Dn	TC	CEO
0	0	0	X	X	X	No change	No change	0
0	0	1	↑	1	X	Inc	TC	CEO

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
0	0	1	↑	0	X	Dec	TC	CEO
z = bit width - 1 TC = (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) + (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) CEO = TC•CE								

Design Entry Method

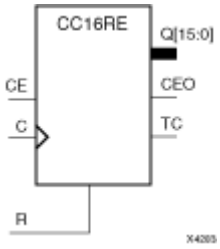
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CC16RE

Macro: 16-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a synchronous resettable, cascadable binary counter. These counters are implemented using carry logic with relative location constraints to ensure efficient logic placement. The synchronous reset (R) is the highest priority input. When R is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero on the Low-to-High clock (C) transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs and CE are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
R	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	↑	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0) CEO = TC•CE					

Design Entry Method

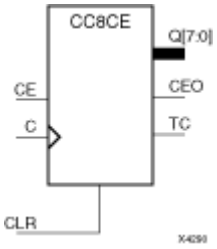
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CC8CE

Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is an asynchronously clearable, cascadable binary counter. It is implemented using carry logic with relative location constraints to ensure efficient logic placement. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**		
CLR	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	X	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE					

Design Entry Method

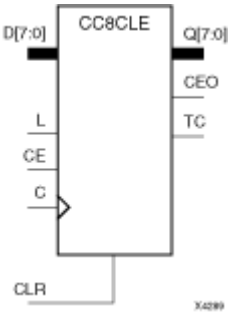
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CC8CLE

Macro: 8-Bit Loadable Cascadable Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable binary counter. It is implemented using carry logic with relative location constraints to ensure efficient logic placement. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					**Outputs**		
CLR	**L**	**CE**	**C**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No change	No change	0
0	0	1	↑	X	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0 CEO = TC•CE							

Design Entry Method

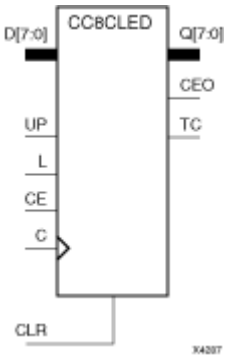
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CC8CLED

Macro: 8-Bit Loadable Cascadable Bidirectional Binary Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a synchronously loadable, asynchronously clearable, cascadable, bidirectional binary counter. It is implemented using carry logic with relative location constraints, which assures most efficient logic placement. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs decrement when CE is High and UP is Low during the Low-to-High clock transition. The Q outputs increment when CE and UP are High. The counter ignores clock transitions when CE is Low.

For counting up, the TC output is High when all Q outputs and UP are High. For counting down, the TC output is High when all Q outputs and UP are Low.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, UP, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
1	X	X	X	X	X	0	0	0
0	1	X	↑	X	Dn	Dn	TC	CEO
0	0	0	X	X	X	No change	No change	0
0	0	1	↑	1	X	Inc	TC	CEO

Inputs						Outputs		
CLR	**L**	**CE**	**C**	**UP**	**Dz – D0**	**Qz – Q0**	**TC**	**CEO**
0	0	1	↑	0	X	Dec	TC	CEO
z = bit width - 1 TC = (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) + (Qz•Q(z-1)•Q(z-2)•...•Q0•UP) CEO = TC•CE								

Design Entry Method

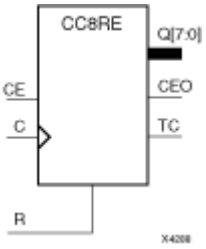
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CC8RE

Macro: 8-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a synchronous resettable, cascadable binary counter. These counters are implemented using carry logic with relative location constraints to ensure efficient logic placement. The synchronous reset (R) is the highest priority input. When R is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero on the Low-to-High clock (C) transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs and CE are High.

Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. .For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs		
R	**CE**	**C**	**Qz-Q0**	**TC**	**CEO**
1	X	↑	0	0	0
0	0	X	No change	No change	0
0	1	↑	Inc	TC	CEO
z = bit width - 1 TC = Qz•Q(z-1)•Q(z-2)•...•Q0) CEO = TC•CE					

Design Entry Method

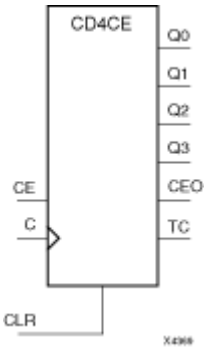
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CD4CE

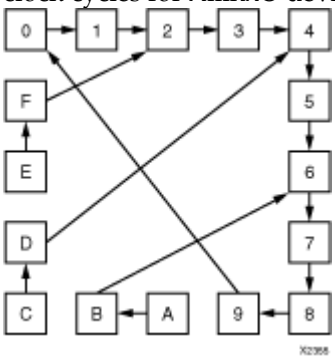
Macro: 4-Bit Cascadable BCD Counter with Clock Enable and Asynchronous Clear



Introduction

CD4CE is a 4-bit (stage), asynchronous clearable, cascadable binary-coded-decimal (BCD) counter. The asynchronous clear input (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The Q outputs increment when clock enable (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles for Xilinx® devices, as shown in the following state diagram:



Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs					
CLR	**CE**	**C**	**Q3**	**Q2**	**Q1**	**Q0**	**TC**	**CEO**
1	X	X	0	0	0	0	0	0

Inputs			Outputs					
CLR	**CE**	**C**	**Q3**	**Q2**	**Q1**	**Q0**	**TC**	**CEO**
0	1	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	X	No Change	No Change	No Change	No Change	TC	0
0	1	X	1	0	0	1	1	1
TC = Q3•!Q2•!Q1•Q0 CEO = TC•CE								

Design Entry Method

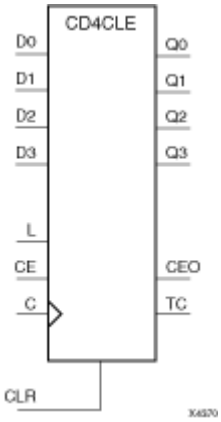
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CD4CLE

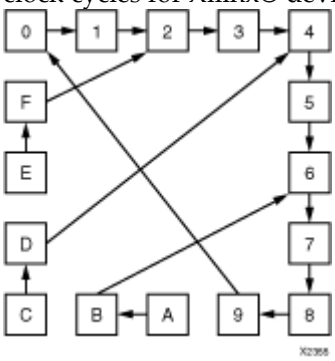
Macro: 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Asynchronous Clear



Introduction

CD4CLE is a 4-bit (stage), synchronously loadable, asynchronously clearable, binarycoded- decimal (BCD) counter. The asynchronous clear input (CLR) is the highest priority input. When (CLR) is High, all other inputs are ignored; the (Q) outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the (D) inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition. The (Q) outputs increment when clock enable input (CE) is High during the Low- to-High clock transition. The counter ignores clock transitions when (CE) is Low. The (TC) output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles for Xilinx® devices, as shown in the following state diagram:



Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs					
CLR	**L**	**CE**	**D3 – D0**	**C**	**Q3**	**Q2**	**Q1**	**Q0**	**TC**	**CEO**
1	X	X	X	X	0	0	0	0	0	0
0	1	X	D3 – D0	↑	D3	D2	D1	D0	TC	CEO
0	0	1	X	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	0	X	X	No Change	No Change	No Change	No Change	TC	0
0	0	1	X	X	1	0	0	1	1	1
TC = Q3•!Q2•!Q1•Q0 CEO = TC•CE										

Design Entry Method

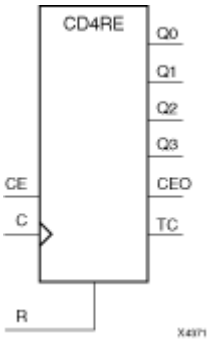
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CD4RE

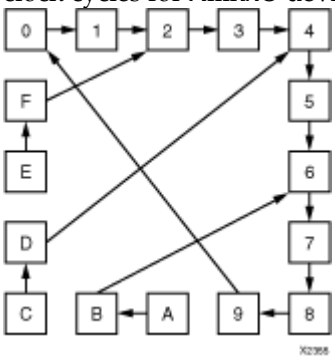
Macro: 4-Bit Cascadable BCD Counter with Clock Enable and Synchronous Reset



Introduction

CD4RE is a 4-bit (stage), synchronous resettable, cascadable binary-coded-decimal (BCD) counter. The synchronous reset input (R) is the highest priority input. When (R) is High, all other inputs are ignored; the (Q) outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero on the Low-to-High clock (C) transition. The (Q) outputs increment when the clock enable input (CE) is High during the Low-to- High clock transition. The counter ignores clock transitions when (CE) is Low. The (TC) output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles for Xilinx® devices, as shown in the following state diagram:



Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs					
R	**CE**	**C**	**Q3**	**Q2**	**Q1**	**Q0**	**TC**	**CEO**
1	X	↑	0	0	0	0	0	0

Inputs			Outputs					
R	**CE**	**C**	**Q3**	**Q2**	**Q1**	**Q0**	**TC**	**CEO**
0	1	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	X	No Change	No Change	No Change	No Change	TC	0
0	1	X	1	0	0	1	1	1
TC = Q3•!Q2•!Q1•Q0 CEO = TC•CE								

Design Entry Method

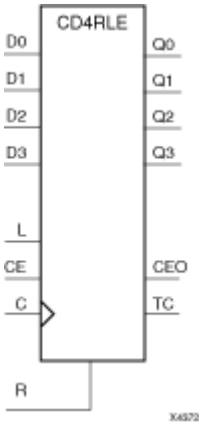
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CD4RLE

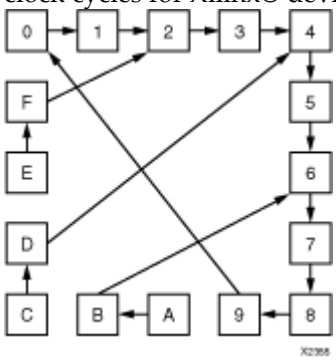
Macro: 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Synchronous Reset



Introduction

CD4RLE is a 4-bit (stage), synchronous loadable, resettable, binary-coded-decimal (BCD) counter. The synchronous reset input (R) is the highest priority input. When R is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero on the Low-to-High clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles for Xilinx® devices, as shown in the following state diagram:



Create larger counters by connecting the CEO output of each stage to the CE input of the next stage and connecting the C, L, and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than n ($t_{CE\text{-}TC}$), where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input or use the TC output if it does not.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs					
R	**L**	**CE**	**D3 – D0**	**C**	**Q3**	**Q2**	**Q1**	**Q0**	**TC**	**CEO**
1	X	X	X	↑	0	0	0	0	0	0
0	1	X	D3 – D0	↑	D3	D	D	D0	TC	CEO
0	0	1	X	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	0	X	X	No Change	No Change	No Change	No Change	TC	0
0	0	1	X	X	1	0	0	1	1	1
TC = Q3•!Q2•!Q1•Q0 CEO = TC•CE										

Design Entry Method

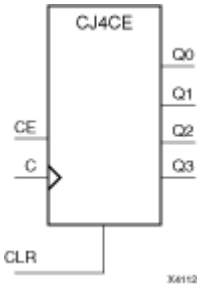
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CJ4CE

Macro: 4-Bit Johnson Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a clearable Johnson/shift counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the data (Q) outputs to logic level zero, independent of clock (C) transitions. The counter increments (shifts Q0 to Q1, Q1 to Q2,and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when (CE) is Low.

The Q3 output is inverted and fed back to input Q0 to provide continuous counting operation.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs	
CLR	**CE**	**C**	**Q0**	**Q1 through Q3**
1	X	X	0	0
0	0	X	No change	No change
0	1	↑	!q3	q0 through q2
q = state of referenced output one setup time prior to active clock transition				

Design Entry Method

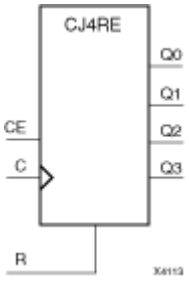
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CJ4RE

Macro: 4-Bit Johnson Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a resettable Johnson/shift counter. The synchronous reset (R) input, when High, overrides all other inputs and forces the data (Q) outputs to logic level zero during the Low-to-High clock (C) transition. The counter increments (shifts Q0 to Q1, Q1 to Q2, and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when CE is Low.

The Q3 output is inverted and fed back to input Q0 to provide continuous counting operation.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**	
R	**CE**	**C**	**Q0**	**Q1 through Q3**
1	X	↑	0	0
0	0	X	No change	No change
0	1	↑	!q3	q0 through q2
q = state of referenced output one setup time prior to active clock transition				

Design Entry Method

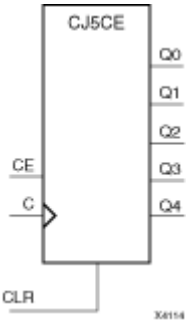
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CJ5CE

Macro: 5-Bit Johnson Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a clearable Johnson/shift counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the data (Q) outputs to logic level zero, independent of clock (C) transitions. The counter increments (shifts Q0 to Q1, Q1 to Q2,and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when (CE) is Low.

The Q4 output is inverted and fed back to input Q0 to provide continuous counting operation.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs	
CLR	**CE**	**C**	**Q0**	**Q1 through Q4**
1	X	X	0	0
0	0	X	No change	No change
0	1	↑	!q4	q0 through q3
q = state of referenced output one setup time prior to active clock transition				

Design Entry Method

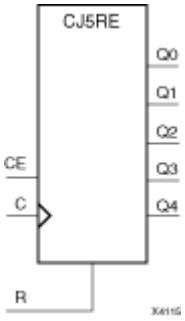
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CJ5RE

Macro: 5-Bit Johnson Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a resettable Johnson/shift counter. The synchronous reset (R) input, when High, overrides all other inputs and forces the data (Q) outputs to logic level zero during the Low-to-High clock (C) transition. The counter increments (shifts Q0 to Q1, Q1 to Q2, and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when CE is Low.

The Q4 output is inverted and fed back to input Q0 to provide continuous counting operation.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**	
R	**CE**	**C**	**Q0**	**Q1 through Q4**
1	X	↑	0	0
0	0	X	No change	No change
0	1	↑	!q4	q0 through q3
q = state of referenced output one setup time prior to active clock transition				

Design Entry Method

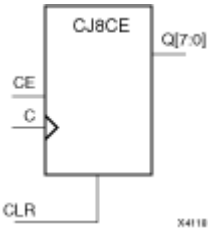
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CJ8CE

Macro: 8-Bit Johnson Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a clearable Johnson/shift counter. The asynchronous clear (CLR) input, when High, overrides all other inputs and forces the data (Q) outputs to logic level zero, independent of clock (C) transitions. The counter increments (shifts Q0 to Q1, Q1 to Q2,and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when (CE) is Low.

The Q7 output is inverted and fed back to input Q0 to provide continuous counting operation.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**	
CLR	**CE**	**C**	**Q0**	**Q1 through Q8**
1	X	X	0	0
0	0	X	No change	No change
0	1	↑	!q7	q0 through q7
q = state of referenced output one setup time prior to active clock transition				

Design Entry Method

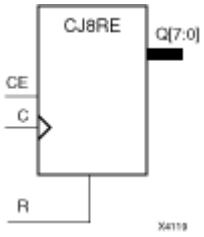
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CJ8RE

Macro: 8-Bit Johnson Counter with Clock Enable and Synchronous Reset



Introduction

This design element is a resettable Johnson/shift counter. The synchronous reset (R) input, when High, overrides all other inputs and forces the data (Q) outputs to logic level zero during the Low-to-High clock (C) transition. The counter increments (shifts Q0 to Q1, Q1 to Q2, and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when CE is Low.

The Q7 output is inverted and fed back to input Q0 to provide continuous counting operation.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs	
R	**CE**	**C**	**Q0**	**Q1 through Q7**
1	X	↑	0	0
0	0	X	No change	No change
0	1	↑	!q7	q0 through q6
q = state of referenced output one setup time prior to active clock transition				

Design Entry Method

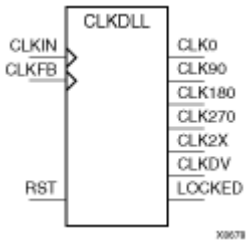
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CLKDLL

Primitive: Clock Delay Locked Loop



Introduction

This design element is a clock delay locked loop used to minimize clock skew. It synchronizes the clock signal at the feedback clock input (CLKFB) to the clock signal at the input clock (CLKIN). The locked output (LOCKED) is high when the two signals are in phase. The signals are considered to be in phase when their rising edges are within a specific range of each other (see The Programmable Logic Data Sheets for the most current value).

The frequency of the clock signal at the CLKIN input must be in a specific range depending on speed grade (see The Programmable Logic Data Sheets for the most current values). The CLKIN pin must be driven by an IBUFG or a BUFG. If phase alignment is not required, CLKIN can also be driven by IBUF.

On-chip synchronization is achieved by connecting the CLKFB input to a point on the global clock network driven by a BUFG, a global clock buffer. The BUFG connected to the CLKFB input of the CLKDLL must be sourced from either the CLK0 or CLK2X outputs of the same CLKDLL. The CLKIN input should be connected to the output of an IBUFG, with the IBUFG input connected to a pad driven by the system clock.

Off-chip synchronization is achieved by connecting the CLKFB input to the output of an IBUFG, with the IBUFG input connected to a pad. Either the CLK0 or CLK2X output can be used but not both. The CLK0 or CLK2X must be connected to the input of OBUF, an output buffer.

The duty cycle of the CLK0 output is 50-50 unless the DUTY_CYCLE_CORRECTION attribute is set to FALSE, in which case the duty cycle is the same as that of the CLKIN input. The duty cycle of the phase shifted outputs (CLK90, CLK180, and CLK270) is the same as that of the CLK0 output. The duty cycle of the CLK2X and CLKDV outputs is always 50-50. The frequency of the CLKDV output is determined by the value assigned to the CLKDV_DIVIDE attribute.

The master reset input (RST) resets CLKDLL to its initial (power-on) state. The signal at the RST input is asynchronous and must be held High for just 2ns.

Port Descriptions

Port	Function
CLK0	Clock at 1x CLKIN frequency
CLK180	Clock at 1x CLKIN frequency, shifted 180 º with regards to CLK0
CLK270	Clock at 1x CLKIN frequency, shifted 270 º with regards to CLK0
CLK2X	Clock at 2x CLKIN frequency, in phase with CLK0
CLK90	Clock at 1x CLKIN frequency, shifted 90 º with regards to CLK0
CLKDV	Clock at (1/n)x CLKIN frequency, n=CLKDV_DIVIDE value. CLKDV is in phase with CLK0.
LOCKED	CLKDLL locked

Note See the "PERIOD Specifications on CLKDLLs and DCM" in the *Constraints Guide* for additional information on using the TNM, TNM_NET, and PERIOD attributes with CLKDLL components.

Design Entry Method

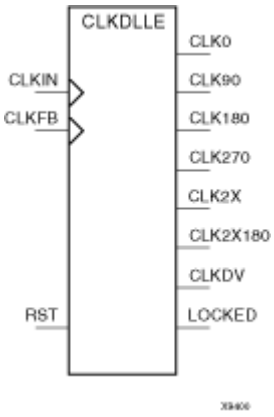
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CLKDLLE

Primitive: Clock Delay Locked Loop with Expanded Output



Introduction

This design element is a clock delay locked loop used to minimize clock skew. It synchronizes the clock signal at the feedback clock input (CLKFB) to the clock signal at the input clock (CLKIN). The locked output (LOCKED) is high when the two signals are in phase. The signals are considered to be in phase when their rising edges are within a specific range of each other (see The Programmable Logic Data Sheets for the most current value).

The frequency of the clock signal at the CLKIN input must be in a specific range depending on speed grade (see The Programmable Logic Data Sheets for the most current values). The CLKIN pin must be driven by an IBUFG or a BUFG.

On-chip synchronization is achieved by connecting the CLKFB input to a point on the global clock network driven by a BUFG, a global clock buffer. The BUFG input can only be connected to the CLK0 or CLK2X output of CLKDLLE. The BUFG connected to the CLKFB input of the CLKDLLE must be sourced from either the CLK0 or CLK2X outputs of the same CLKDLLE. The CLKIN input should be connected to the output of an IBUFG, with the IBUFG input connected to a pad driven by the system clock.

Off-chip synchronization is achieved by connecting the CLKFB input to the output of an IBUFG, with the IBUFG input connected to a pad. Either the CLK0 or CLK2X output can be used but not both. The CLK0 or CLK2X must be connected to the input of OBUF, an output buffer.

The duty cycle of the CLK0 output is 50-50 unless the DUTY_CYCLE_CORRECTION attribute is set to FALSE, in which case the duty cycle is the same as that of the CLKIN input. The duty cycle of the phase shifted outputs (CLK90, CLK180, and CLK270) is the same as that of the CLK0 output. The duty cycle of the CLK2X and CLKDV outputs is always 50-50. The frequency of the CLKDV output is determined by the value assigned to the CLKDV_DIVIDE attribute.

The master reset input (RST) resets CLKDLL to its initial (power-on) state. The signal at the RST input is asynchronous and must be held High for just 2ns.

Port Descriptions

Port	Function
CLK0	Clock at 1x CLKIN frequency
CLK180	Clock at 1x CLK0 frequency, shifted 180 º with regards to CLK0
CLK270	Clock at 1x CLK0 frequency, shifted 270 º with regards to CLK0
CLK2X	Clock at 2x CLK0 frequency, in phase with CLK0

Port	Function
CLK2X180	Clock at 1x CLK2X frequency shifted 180 o with regards to CLK2X
CLK90	Clock at 1x CLK0 frequency, shifted 90 o with regards to CLK0
CLKDV	Clock at (1/n) x CLK0 frequency, where n=CLKDV_DIVIDE value. CLKDV is in phase with CLK0.
LOCKED	CLKDLLE locked. CLKIN and CLKFB synchronized.

Note See the "PERIOD Specifications on CLKDLLs and DCM" in the Constraints Guide for additional information on using the TNM, TNM_NET, and PERIOD attributes with CLKDLLE components.

Design Entry Method

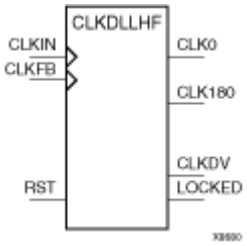
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CLKDLLHF

Primitive: High Frequency Clock Delay Locked Loop



Introduction

This design element is a high frequency clock delay locked loop used to minimize clock skew. It synchronizes the clock signal at the feedback clock input (CLKFB) to the clock signal at the input clock (CLKIN). The locked output (LOCKED) is high when the two signals are in phase. The signals are considered to be in phase when their rising edges are within a specific range of each other (see The Programmable Logic Data Sheets for the most current value).

The frequency of the clock signal at the CLKIN input must be in a specific range depending on speed grade (see The Programmable Logic Data Sheets for the most current values). The CLKIN pin must be driven by an IBUFG or a BUFG.

On-chip synchronization is achieved by connecting the CLKFB input to a point on the global clock network driven by a BUFG, a global clock buffer. The BUFG input can only be connected to the CLK0 output of CLKDLLHF. The BUFG connected to the CLKFB input of the CLKDLLHF must be sourced from the CLK0 output of the same CLKDLLHF. The CLKIN input should be connected to the output of an IBUFG, with the IBUFG input connected to a pad driven by the system clock.

Off-chip synchronization is achieved by connecting the CLKFB input to the output of an IBUFG, with the IBUFG input connected to a pad. Only the CLK0 output can be used. CLK0 must be connected to the input of OBUF, an output buffer.

The duty cycle of the CLK0 output is 50-50 unless the DUTY_CYCLE_CORRECTION attribute is set to FALSE, in which case the duty cycle is the same as that of the CLKIN input. The duty cycle of the phase shifted output (CLK180) is the same as that of the CLK0 output. The frequency of the CLKDV output is determined by the value assigned to the CLKDV_DIVIDE attribute.

The master reset input (RST) resets CLKDLL to its initial (power-on) state. The signal at the RST input is asynchronous and must be held High for just 2ns.

Port Descriptions

Output	Function
CLK0	Clock at 1x CLKIN frequency
CLK180	Clock at 1x CLKIN frequency, shifted 180o with regards to CLK0
CLKDV	Clock at (1/n)x CLKIN frequency, n=CLKDV_DIVIDE value. CLKDV is in phase with CLK0.
LOCKED	CLKDLLHF locked

Note See the "PERIOD Specifications on CLKDLLs and DCM" section of the "Xilinx Constraints P" chapter in the Constraints Guide for additional information on using the TNM, TNM_NET, and PERIOD attributes with CLKDLLHF components.

Design Entry Method

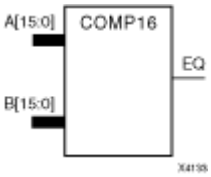
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMP16

Macro: 16-Bit Identity Comparator



Introduction

This design element is a 16-bit identity comparator. The equal output (EQ) is high when A15 – A0 and B15 – B0 are equal.

Equality is determined by a bit comparison of the two words. When any two of the corresponding bits from each word are not the same, the EQ output is Low.

Design Entry Method

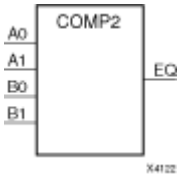
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMP2

Macro: 2-Bit Identity Comparator



Introduction

This design element is a 2-bit identity comparator. The equal output (EQ) is High when the two words A1 – A0 and B1 – B0 are equal.

Equality is determined by a bit comparison of the two words. When any two of the corresponding bits from each word are not the same, the EQ output is Low.

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMP4

Macro: 4-Bit Identity Comparator

Introduction

This design element is a 4-bit identity comparator. The equal output (EQ) is high when A3 – A0 and B3 – B0 are equal.

Equality is determined by a bit comparison of the two words. When any two of the corresponding bits from each word are not the same, the EQ output is Low.

Design Entry Method

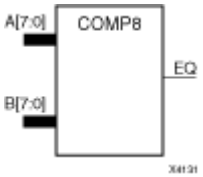
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMP8

Macro: 8-Bit Identity Comparator



Introduction

This design element is an 8-bit identity comparator. The equal output (EQ) is high when A7 – A0 and B7 – B0 are equal.

Equality is determined by a bit comparison of the two words. When any two of the corresponding bits from each word are not the same, the EQ output is Low.

Design Entry Method

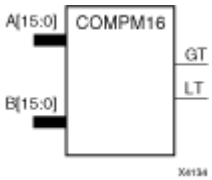
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMPM16

Macro: 16-Bit Magnitude Comparator



Introduction

This design element is a 16-bit magnitude comparator that compare two positive Binary-weighted words. It compares A15 – A0 and B15 – B0, where A15 and B15 are the most significant bits.

The greater-than output (GT) is High when A > B, and the less-than output (LT) is High when A < B When the two words are equal, both GT and LT are Low. Equality can be measured with this macro by comparing both outputs with a NOR gate.

Logic Table

Inputs								Outputs	
A7, B7	**A6, B6**	**A5, B5**	**A4, B4**	**A3, B3**	**A2, B2**	**A1, B1**	**A0, B0**	**GT**	**LT**
A7>B7	X	X	X	X	X	X	X	1	0
A7<B7	X	X	X	X	X	X	X	0	1
A7=B7	A6>B6	X	X	X	X	X	X	1	0
A7=B7	A6<B6	X	X	X	X	X	X	0	1
A7=B7	A6=B6	A5>B5	X	X	X	X	X	1	0
A7=B7	A6=B6	A5<B5	X	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4>B4	X	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4<B4	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3>B3	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3<B3	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2>B2	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2<B2	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1>B1	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1<B1	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0>B0	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0<B0	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0=B0	0	0

Design Entry Method

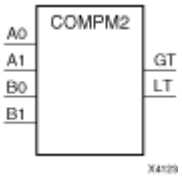
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMPM2

Macro: 2-Bit Magnitude Comparator



Introduction

This design element is a 2-bit magnitude comparator that compare two positive Binary-weighted words. It compares A1 – A0 and B1 – B0, where A1 and B1 are the most significant bits.

The greater-than output (GT) is High when A > B, and the less-than output (LT) is High when A < B When the two words are equal, both GT and LT are Low. Equality can be measured with this macro by comparing both outputs with a NOR gate.

Logic Table

Inputs				Outputs	
A1	**B1**	**A0**	**B0**	**GT**	**LT**
0	0	0	0	0	0
0	0	1	0	1	0
0	0	0	1	0	1
0	0	1	1	0	0
1	1	0	0	0	0
1	1	1	0	1	0
1	1	0	1	0	1
1	1	1	1	0	0
1	0	X	X	1	0
0	1	X	X	0	1

Design Entry Method

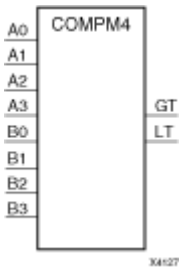
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMPM4

Macro: 4-Bit Magnitude Comparator



Introduction

This design element is a 4-bit magnitude comparator that compare two positive Binary-weighted words. It compares A3 – A0 and B3 – B0, where A3 and B3 are the most significant bits.

The greater-than output (GT) is High when A > B, and the less-than output (LT) is High when A < B When the two words are equal, both GT and LT are Low. Equality can be measured with this macro by comparing both outputs with a NOR gate.

Logic Table

Inputs				Outputs	
A3, B3	**A2, B2**	**A1, B1**	**A0, B0**	**GT**	**LT**
A3>B3	X	X	X	1	0
A3<B3	X	X	X	0	1
A3=B3	A2>B2	X	X	1	0
A3=B3	A2<B2	X	X	0	1
A3=B3	A2=B2	A1>B1	X	1	0
A3=B3	A2=B2	A1<B1	X	0	1
A3=B3	A2=A2	A1=B1	A0>B0	1	0
A3=B3	A2=B2	A1=B1	A0<B0	0	1
A3=B3	A2=B2	A1=B1	A0=B0	0	0

Design Entry Method

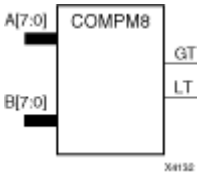
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMPM8

Macro: 8-Bit Magnitude Comparator



Introduction

This design element is an 8-bit magnitude comparator that compare two positive Binary-weighted words. It compares A7 – A0 and B7 – B0, where A7 and B7 are the most significant bits.

The greater-than output (GT) is High when A > B, and the less-than output (LT) is High when A < B When the two words are equal, both GT and LT are Low. Equality can be measured with this macro by comparing both outputs with a NOR gate.

Logic Table

Inputs								Outputs	
A7, B7	**A6, B6**	**A5, B5**	**A4, B4**	**A3, B3**	**A2, B2**	**A1, B1**	**A0, B0**	**GT**	**LT**
A7>B7	X	X	X	X	X	X	X	1	0
A7<B7	X	X	X	X	X	X	X	0	1
A7=B7	A6>B6	X	X	X	X	X	X	1	0
A7=B7	A6<B6	X	X	X	X	X	X	0	1
A7=B7	A6=B6	A5>B5	X	X	X	X	X	1	0
A7=B7	A6=B6	A5<B5	X	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4>B4	X	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4<B4	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3>B3	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3<B3	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2>B2	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2<B2	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1>B1	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1<B1	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0>B0	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0<B0	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0=B0	0	0

Design Entry Method

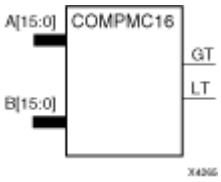
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMPMC16

Macro: 16-Bit Magnitude Comparator



Introduction

This design element is a 16-bit, magnitude comparator that compares two positive Binary weighted words A15 – A0 and B15 – B0, where A15 and B15 are the most significant bits.

This comparator is implemented using carry logic with relative location constraints to ensure efficient logic placement.

The greater-than output (GT) is High when A>B, and the less-than output (LT) is High when A<B. When the two words are equal, both GT and LT are Low. Equality can be flagged with this macro by connecting both outputs to a NOR gate.

Logic Table

Inputs								Outputs	
A7, B7	**A6, B6**	**A5, B5**	**A4, B4**	**A3, B3**	**A2, B2**	**A1, B1**	**A0, B0**	**GT**	**LT**
A7>B7	X	X	X	X	X	X	X	1	0
A7<B7	X	X	X	X	X	X	X	0	1
A7=B7	A6>B6	X	X	X	X	X	X	1	0
A7=B7	A6<B6	X	X	X	X	X	X	0	1
A7=B7	A6=B6	A5>B5	X	X	X	X	X	1	0
A7=B7	A6=B6	A5<B5	X	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4>B4	X	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4<B4	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3>B3	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3<B3	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2>B2	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2<B2	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1>B1	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1<B1	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0>B0	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0<B0	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0=B0	0	0

Design Entry Method

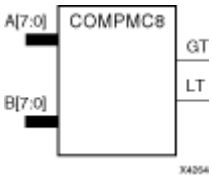
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

COMPMC8

Macro: 8-Bit Magnitude Comparator



Introduction

This design element is an 8-bit, magnitude comparator that compares two positive Binaryweighted words A7 – A0 and B7 – B0, where A7 and B7 are the most significant bits.

This comparator is implemented using carry logic with relative location constraints to ensure efficient logic placement.

The greater-than output (GT) is High when A>B, and the less-than output (LT) is High when A<B. When the two words are equal, both GT and LT are Low. Equality can be flagged with this macro by connecting both outputs to a NOR gate.

Logic Table

Inputs								Outputs	
A7, B7	**A6, B6**	**A5, B5**	**A4, B4**	**A3, B3**	**A2, B2**	**A1, B1**	**A0, B0**	**GT**	**LT**
A7>B7	X	X	X	X	X	X	X	1	0
A7<B7	X	X	X	X	X	X	X	0	1
A7=B7	A6>B6	X	X	X	X	X	X	1	0
A7=B7	A6<B6	X	X	X	X	X	X	0	1
A7=B7	A6=B6	A5>B5	X	X	X	X	X	1	0
A7=B7	A6=B6	A5<B5	X	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4>B4	X	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4<B4	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3>B3	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3<B3	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2>B2	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2<B2	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1>B1	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1<B1	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0>B0	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0<B0	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0=B0	0	0

Design Entry Method

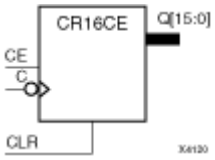
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CR16CE

Macro: 16-Bit Negative-Edge Binary Ripple Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is a 16-bit cascadable, clearable, binary ripple counter with clock enable and asynchronous clear.

Larger counters can be created by connecting the last Q output of the first stage to the clock input of the next stage. CLR and CE inputs are connected in parallel. The clock period is not affected by the overall length of a ripple counter. The overall clock-to-output propagation is n(t_{C-Q}), where n is the number of stages and the time t_{C-Q} is the C-to-Qz propagation delay of each stage.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CLR	**CE**	**C**	**Qz – Q0**
1	X	X	0
0	0	X	No Change
0	1	↓	Inc
z = bit width - 1			

Design Entry Method

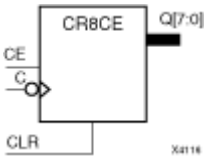
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

CR8CE

Macro: 8-Bit Negative-Edge Binary Ripple Counter with Clock Enable and Asynchronous Clear



Introduction

This design element is an 8-bit cascadable, clearable, binary, ripple counter with clock enable and asynchronous clear.

The asynchronous clear (CLR), when High, overrides all other inputs and causes the Q outputs to go to logic level zero. The counter increments when the clock enable input (CE) is High during the High-to-Low clock (C) transition. The counter ignores clock transitions when CE is Low.

Larger counters can be created by connecting the last Q output of the first stage to the clock input of the next stage. CLR and CE inputs are connected in parallel. The clock period is not affected by the overall length of a ripple counter. The overall clock-to-output propagation is n(t_{C-Q}), where n is the number of stages and the time t_{C-Q} is the C-to-Qz propagation delay of each stage.

This counter is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CLR	**CE**	**C**	**Qz – Q0**
1	X	X	0
0	0	X	No Change
0	1	↓	Inc
z = bit width - 1			

Design Entry Method

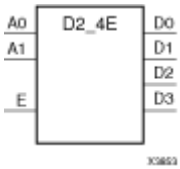
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

D2_4E

Macro: 2- to 4-Line Decoder/Demultiplexer with Enable



Introduction

This design element is a decoder/demultiplexer. When the enable (E) input of this element is High, one of four active-High outputs (D3 – D0) is selected with a 2-bit binary address (A1 – A0) input. The non-selected outputs are Low. Also, when the E input is Low, all outputs are Low. In demultiplexer applications, the E input is the data input.

Logic Table

Inputs			Outputs			
A1	**A0**	**E**	**D3**	**D2**	**D1**	**D0**
X	X	0	0	0	0	0
0	0	1	0	0	0	1
0	1	1	0	0	1	0
1	0	1	0	1	0	0
1	1	1	1	0	0	0

Design Entry Method

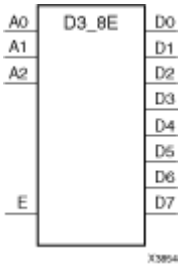
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

D3_8E

Macro: 3- to 8-Line Decoder/Demultiplexer with Enable



Introduction

When the enable (E) input of the D3_8E decoder/demultiplexer is High, one of eight active-High outputs (D7 – D0) is selected with a 3-bit binary address (A2 – A0) input. The non-selected outputs are Low. Also, when the E input is Low, all outputs are Low. In demultiplexer applications, the E input is the data input.

Logic Table

Inputs				Outputs							
A2	**A1**	**A0**	**E**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
X	X	X	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1
0	0	1	1	0	0	0	0	0	0	1	0
0	1	0	1	0	0	0	0	0	1	0	0
0	1	1	1	0	0	0	0	1	0	0	0
1	0	0	1	0	0	0	1	0	0	0	0
1	0	1	1	0	0	1	0	0	0	0	0
1	1	0	1	0	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0

Design Entry Method

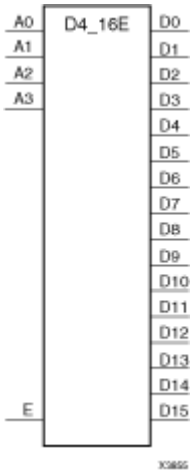
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

D4_16E

Macro: 4- to 16-Line Decoder/Demultiplexer with Enable



Introduction

This design element is a decoder/demultiplexer. When the enable (E) input of this design element is High, one of 16 active-High outputs (D15 – D0) is selected with a 4-bit binary address (A3 – A0) input. The non-selected outputs are Low. Also, when the E input is Low, all outputs are Low. In demultiplexer applications, the E input is the data input.

Design Entry Method

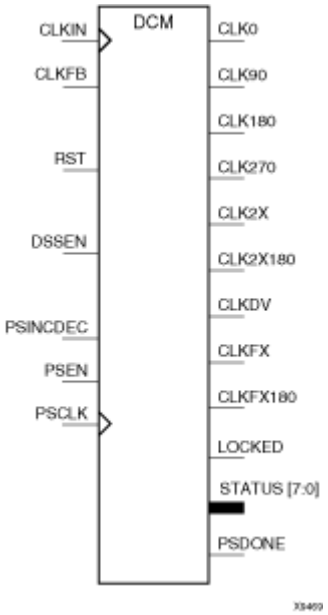
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DCM

Primitive: Digital Clock Manager



Introduction

This design element is a digital clock manager that provides multiple functions. It can implement a clock delay locked loop, a digital frequency synthesizer, digital phase shifter, and a digital spread spectrum.

Note All unused inputs must be driven Low. The program will automatically tie the inputs Low if they are unused.

Clock Delay Locked Loop (DLL)

DCM includes a clock delay locked loop used to minimize clock skew for Spartan-3, Virtex-II, Virtex-II Pro, and Virtex-II Pro X devices. DCM synchronizes the clock signal at the feedback clock input (CLKFB) to the clock signal at the input clock (CLKIN). The locked output (LOCKED) is high when the two signals are in phase. The signals are considered to be in phase when their rising edges are within a specified time (ps) of each other.

DCM supports two frequency modes for the DLL. By default, the DLL_FREQUENCY_MODE attribute is set to Low and the frequency of the clock signal at the CLKIN input must be in the Low (DLL_CLKIN_MIN_LF to DLL_CLKIN_MAX_LF) frequency range (MHz). In Low frequency mode, the CLK0, CLK90, CLK180, CLK270, CLK2X, CLKDV, and CLK2X180 outputs are available.

For up to and including Virtex-II Pro, you get only CLK0, CLK180, CLKDV, CLKFX and CLKFX180 in the HF mode. In Virtex-4, you get all outputs.

When the DLL_FREQUENCY_MODE attribute is set to High, the frequency of the clock signal at the CLKIN input must be in the High (DLL_CLKIN_MIN_HF to DLL_CLKIN_MAX_HF) frequency range (MHz). In High frequency mode, only the CLK0, CLK180, and CLKDV outputs are available.

On-chip synchronization is achieved by connecting the CLKFB input to a point on the global clock network driven by a BUFG, a global clock buffer. The BUFG connected to the CLKFB input of the DCM must be sourced from either the CLK0 or CLK2X outputs of the same DCM. The CLKIN input should be connected to the output of an IBUFG, with the IBUFG input connected to a pad driven by the system clock. Off-chip synchronization is achieved by connecting the CLKFB input to the output of an IBUFG, with the IBUFG input connected to a pad. Either the CLK0 or CLK2X output can be used but not both. The CLK0 or CLK2X must be connected to the input of OBUF, an output buffer. The CLK_FEEDBACK attribute controls whether the CLK0 output, the default, or the CLK2X output is the source of the CLKFB input.

The duty cycle of the CLK0 output is 50-50 unless the DUTY_CYCLE_CORRECTION attribute is set to FALSE, in which case the duty cycle is the same as that of the CLKIN input. The duty cycle of the phase shifted outputs (CLK90, CLK180, and CLK270) is the same as that of the CLK0 output. The duty cycle of the CLK2X, CLK2X180, and CLKDV outputs is 50-50 unless CLKDV_DIVIDE is a non-integer and the DLL_FREQUENCY_MODE is High (see CLKDV_DIVIDE, in the Constraints Guide for details). The frequency of the CLKDV output is determined by the value assigned to the CLKDV_DIVIDE attribute

DCM Clock Delay Lock Loop Outputs

Output	Description
CLK0	Clock at 1x CLKIN frequency
CLK180	Clock at 1x CLK0 frequency, shifted 180o with regards to CLK0
CLK270*	Clock at 1x CLK0 frequency, shifted 270o with regards to CLK0
CLK2X*	Clock at 2x CLK0 frequency, in phase with CLK0
CLK2X180*	Clock at 2x CLK0 frequency shifted 180o with regards to CLK2X
CLK90*	Clock at 1x CLK0 frequency, shifted 90o with regards to CLK0
CLKDV	Clock at (1/n) x CLK0 frequency, where n=CLKDV_DIVIDE value. CLKDV is in phase with CLK0.
LOCKED	All enabled DCM features locked.
* The CLK90, CLK270, CLK2X, and CLK2X180 outputs are not available if the DLL_FREQUENCY_MODE is set to High.	

Digital Frequency Synthesizer (DFS)

The CLKFX and CLKFX180 outputs in conjunction with the CLKFX_MULTIPLY and CLKFX_DIVIDE attributes provide a frequency synthesizer that can be any multiple or division of CLKIN. CLKFX and CLKIN are in phase every CLKFX_MULTIPLY cycles of CLKFX and every CLKFX_DIVIDE cycles of CLKIN when a feedback is provided to the CLKFB input of the DLL. The frequency of CLKFX is defined by the following equation.
FrequencyCLKFX= (CLKFX_MULTIPLY_value/CLKFX_DIVIDE_value) * FrequencyCLKIN

Both the CLKFX or CLKFX180 output can be used simultaneously. CLKFX180 is 1x the CLKFX frequency, shifted 180o with regards to CLKFX. CLKFX and CLKFX180 always have a 50/50 duty cycle. The DFS_FREQUENCY_MODE attribute specifies the allowable input clock and output clock frequency ranges. The CLK_FEEDBACK attribute set to NONE will cause the DCM to be in the Digital Frequency Synthesizer mode. The CLKFX and CLKFX180 will be generated without phase correction with respect to CLKIN. The DSSEN input pin for the DCM is no longer recommended for use and should remain unconnected in the design.

Digital Phase Shifter (DPS)

The phase shift (skew) between the rising edges of CLKIN and CLKFB may be configured as a fraction of the CLKIN period with the PHASE_SHIFT attribute. This allows the phase shift to remain constant as ambient conditions change. The CLKOUT_PHASE_SHIFT attribute controls the use of the PHASE_SHIFT value. By default, the CLKOUT_PHASE_SHIFT attribute is set to NONE and the PHASE_SHIFT attribute has no effect.

By creating skew between CLKIN and CLKFB, all DCM output clocks are phase shifted by the amount of the skew. When the CLKOUT_PHASE_SHIFT attribute is set to FIXED, the skew set by the PHASE_SHIFT attribute is used at configuration for the rising edges of CLKIN and CLKFB. The skew remains constant. When the CLKOUT_PHASE_SHIFT attribute is set to VARIABLE, the skew set at configuration is used as a starting point and the skew value can be changed dynamically during operation using the PS* signals. This digital phase shifter feature is controlled by a synchronous interface. The inputs PSEN (phase shift enable) and PSINCDEC (phase shift increment/decrement) are set up to the rising edge of PSCLK (phase shift clock). The PSDONE (phase shift done) output is clocked with the rising edge of PSCLK (the phase shift clock). PSDONE must be connected to implement the complete synchronous interface. The rising-edge skew between CLKIN and CLKFB may be dynamically adjusted after the LOCKED output goes High. The PHASE_SHIFT attribute value specifies the initial phase shift amount when the device is configured. Then the PHASE_SHIFT value is changed one unit when PSEN is activated for one period of PSCLK. The PHASE_SHIFT value is incremented when PSINCDEC is High and decremented when PSINCDEC is Low during the period that PSEN is High.

When the DCM completes an increment or decrement operation, the PSDONE output goes High for a single PSCLK cycle to indicate the operation is complete. At this point the next change may be made. When RST (reset) is High, the PHASE_SHIFT attribute value is reset to the skew value set at configuration. If CLKOUT_PHASE_SHIFT is FIXED or NONE, the PSEN, PSINCDEC, and PSCLK inputs must be tied to GND. The program will automatically tie the inputs to GND if they are not connected by the user.

Additional Status Bits

The STATUS output bits return the following information:

Bit	Description
0	Phase Shift Overflow*
1 = |PHASE_SHIFT| > 255	
1	DLL CLKIN stopped**
1 = CLKIN stopped toggling	
2	DLL CLKFX stopped
1 = CLKFX stopped toggling	
3	No
4	No
5	No
6	No
7	No
* Phase Shift Overflow will also go high if the end of the phase shift delay line is reached (see the product data sheet for the most current value of the maximum shifting delay).	
** If only the DFS outputs are used (CLKFX & CLKFX180), this status bit will not go high if CLKIN stops.	

LOCKED

When LOCKED is high, all enabled signals are locked.

RST

The master reset input (RST) resets DCM to its initial (power-on) state. The signal at the RST input is asynchronous and must be held High for 2ns.

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
SIM_MODE	String	"SAFE" or "FAST"	"SAFE"	This is a simulation only attribute. It will direct the simulation model to run in performance-oriented mode when set to "FAST." Please see the *Synthesis and Simulation Design Guide* for more information.

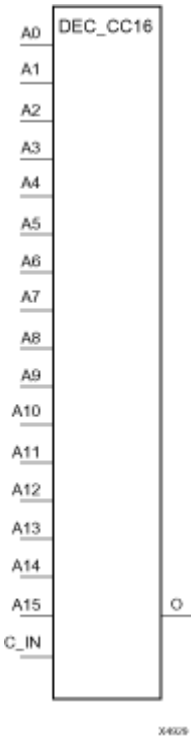
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

DEC_CC16

Macro: 16-Bit Active Low Decoder



Introduction

This design element is a 16-bit decoder that is used to build wide-decoder functions. It is implemented by cascading CY_MUX elements driven by Look-Up Tables (LUTs). The C_IN pin can only be driven by the output (O) of a previous decode stage. When one or more of the inputs (A) are Low, the output is Low. When all the inputs are High and the C_IN input is High, the output is High. You can decode patterns by adding inverters to inputs.

Logic Table

Inputs					Outputs
A0	**A1**	**…**	**Az**	**C_IN**	**O**
1	1	1	1	1	1
X	X	X	X	0	0
0	X	X	X	X	0
X	0	X	X	X	0
X	X	X	0	X	0
z = 3 for DEC_CC4; z = 7 for DEC_CC8; z = 15 for DEC_CC16					

Design Entry Method

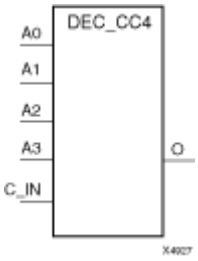
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DEC_CC4

Macro: 4-Bit Active Low Decoder



Introduction

This design element is a 4-bit decoder that is used to build wide-decoder functions. It is implemented by cascading CY_MUX elements driven by Look-Up Tables (LUTs). The C_IN pin can only be driven by the output (O) of a previous decode stage. When one or more of the inputs (A) are Low, the output is Low. When all the inputs are High and the C_IN input is High, the output is High. You can decode patterns by adding inverters to inputs.

Logic Table

Inputs					Outputs
A0	**A1**	**…**	**Az**	**C_IN**	**O**
1	1	1	1	1	1
X	X	X	X	0	0
0	X	X	X	X	0
X	0	X	X	X	0
X	X	X	0	X	0
z = 3 for DEC_CC4; z = 7 for DEC_CC8; z = 15 for DEC_CC16					

Design Entry Method

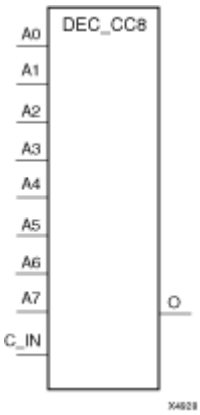
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DEC_CC8

Macro: 8-Bit Active Low Decoder



Introduction

This design element is a 8-bit decoder that is used to build wide-decoder functions. It is implemented by cascading CY_MUX elements driven by Look-Up Tables (LUTs). The C_IN pin can only be driven by the output (O) of a previous decode stage. When one or more of the inputs (A) are Low, the output is Low. When all the inputs are High and the C_IN input is High, the output is High. You can decode patterns by adding inverters to inputs.

Logic Table

Inputs					Outputs
A0	**A1**	**...**	**Az**	**C_IN**	**O**
1	1	1	1	1	1
X	X	X	X	0	0
0	X	X	X	X	0
X	0	X	X	X	0
X	X	X	0	X	0
z = 3 for DEC_CC4; z = 7 for DEC_CC8; z = 15 for DEC_CC16					

Design Entry Method

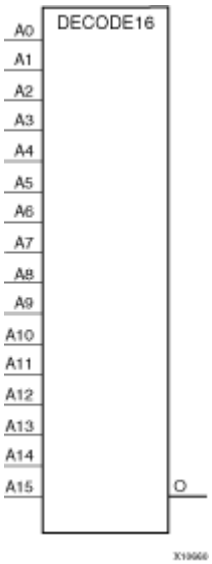
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DECODE16

Macro: 16-Bit Active-Low Decoder



Introduction

This design element is a 4-bit, active-low decoder that is implemented using combinations of LUTs and MUXCYs.

Logic Table

Inputs				Outputs*
A0	**A1**	**...**	**Az**	**O**
1	1	1	1	1
0	X	X	X	0
X	0	X	X	0
X	X	X	0	0
z = bitwidth -1 *A pull-up resistor must be connected to the output to establish High-level drive current.				

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DECODE32

Macro: 32-Bit Active-Low Decoder

Introduction

This design element is a 32-bit active-low decoder that is implemented using combinations of LUTs and MUXCYs.

Logic Table

Inputs				Outputs
A0	**A1**	**…**	**Az**	**O**
1	1	1	1	1
0	X	X	X	0
X	0	X	X	0
X	X	X	0	0
z = 31 for DECODE32, z = 63 for DECODE64				

Design Entry Method

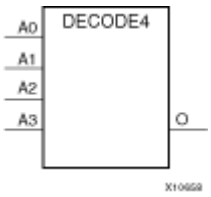
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DECODE4

Macro: 4-Bit Active-Low Decoder



Introduction

This design element is a 4-bit, active-low decoder that is implemented using combinations of LUTs and MUXCYs.

Logic Table

Inputs				Outputs*
A0	**A1**	**…**	**Az**	**O**
1	1	1	1	1
0	X	X	X	0
X	0	X	X	0
X	X	X	0	0

z = bitwidth -1

*A pull-up resistor must be connected to the output to establish High-level drive current.

Design Entry Method

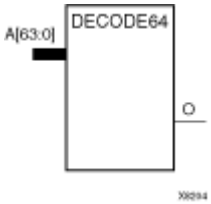
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DECODE64

Macro: 64-Bit Active-Low Decoder



Introduction

This design element is a 64-bit active-low decoder that is implemented using combinations of LUTs and MUXCYs.

Logic Table

Inputs				Outputs
A0	**A1**	**…**	**Az**	**O**
1	1	1	1	1
0	X	X	X	0
X	0	X	X	0
X	X	X	0	0
z = 31 for DECODE32, z = 63 for DECODE64				

Design Entry Method

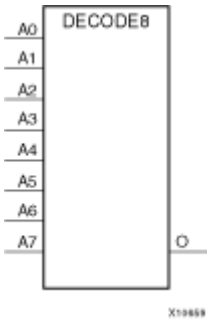
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

DECODE8

Macro: 8-Bit Active-Low Decoder



Introduction

This design element is a 8-bit, active-low decoder that is implemented using combinations of LUTs and MUXCY's.

Logic Table

Inputs				Outputs*
A0	**A1**	**…**	**Az**	**O**
1	1	1	1	1
0	X	X	X	0
X	0	X	X	0
X	X	X	0	0
z = bitwidth -1 *A pull-up resistor must be connected to the output to establish High-level drive current.				

Design Entry Method

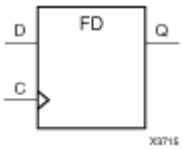
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD

Primitive: D Flip-Flop



Introduction

This design element is a D-type flip-flop with data input (D) and data output (Q). The data on the D inputs is loaded into the flip-flop during the Low-to-High clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
0	↑	0
1	↑	1

Design Entry Method

This design element is only for use in schematics.

Available Attributes

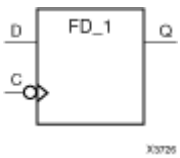
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0, 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD_1

Primitive: D Flip-Flop with Negative-Edge Clock



Introduction

This design element is a single D-type flip-flop with data input (D) and data output (Q). The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	C	Q
0	X	0
1	↓	1

Design Entry Method

This design element is only for use in schematics.

Available Attributes

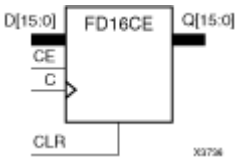
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0, 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD16CE

Macro: 16-Bit Data Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a 16-bit data register with clock enable and asynchronous clear. When clock enable (CE) is High and asynchronous clear (CLR) is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q) during the Low-to-High clock (C) transition. When CLR is High, it overrides all other inputs and resets the data outputs (Q) Low. When CE is Low, clock transitions are ignored.

This register is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				**Outputs**
CLR	**CE**	**Dz – D0**	**C**	**Qz – Q0**
1	X	X	X	0
0	0	X	X	No Change
0	1	Dn	↑	Dn
z = bit-width - 1				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

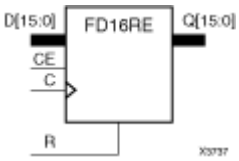
Attribute	**Type**	**Allowed Values**	**Default**	**Description**
INIT	Binary	Any 16-bit Value	All zeros	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD16RE

Macro: 16-Bit Data Register with Clock Enable and Synchronous Reset



Introduction

This design element is a 16-bit data registers. When the clock enable (CE) input is High, and the synchronous reset (R) input is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q0) during the Low-to-High clock (C) transition. When R is High, it overrides all other inputs and resets the data outputs (Q) Low on the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

This register is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				**Outputs**
R	**CE**	**Dz – D0**	**C**	**Qz – Q0**
1	X	X	↑	0
0	0	X	X	No Change
0	1	Dn	↑	Dn
z = bit-width - 1				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

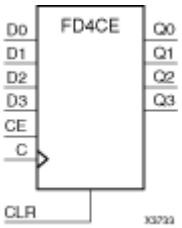
Attribute	**Type**	**Allowed Values**	**Default**	**Description**
INIT	Binary	Any 16-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD4CE

Macro: 4-Bit Data Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a 4-bit data register with clock enable and asynchronous clear. When clock enable (CE) is High and asynchronous clear (CLR) is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q) during the Low-to-High clock (C) transition. When CLR is High, it overrides all other inputs and resets the data outputs (Q) Low. When CE is Low, clock transitions are ignored.

This register is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				**Outputs**
CLR	**CE**	**Dz – D0**	**C**	**Qz – Q0**
1	X	X	X	0
0	0	X	X	No Change
0	1	Dn	↑	Dn
z = bit-width - 1				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

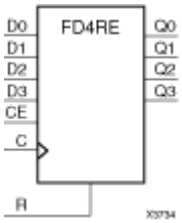
Attribute	**Type**	**Allowed Values**	**Default**	**Description**
INIT	Binary	Any 4-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD4RE

Macro: 4-Bit Data Register with Clock Enable and Synchronous Reset



Introduction

This design element is a 4-bit data registers. When the clock enable (CE) input is High, and the synchronous reset (R) input is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q0) during the Low-to-High clock (C) transition. When R is High, it overrides all other inputs and resets the data outputs (Q) Low on the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

This register is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				**Outputs**
R	**CE**	**Dz – D0**	**C**	**Qz – Q0**
1	X	X	↑	0
0	0	X	X	No Change
0	1	Dn	↑	Dn
z = bit-width - 1				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

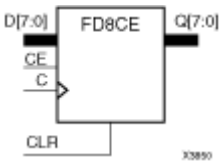
Attribute	**Type**	**Allowed Values**	**Default**	**Description**
INIT	Binary	Any 4-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD8CE

Macro: 8-Bit Data Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a 8-bit data register with clock enable and asynchronous clear. When clock enable (CE) is High and asynchronous clear (CLR) is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q) during the Low-to-High clock (C) transition. When CLR is High, it overrides all other inputs and resets the data outputs (Q) Low. When CE is Low, clock transitions are ignored.

This register is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				**Outputs**
CLR	**CE**	**Dz – D0**	**C**	**Qz – Q0**
1	X	X	X	0
0	0	X	X	No Change
0	1	Dn	↑	Dn
z = bit-width - 1				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

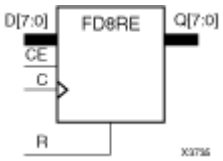
Attribute	**Type**	**Allowed Values**	**Default**	**Description**
INIT	Binary	Any 8-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FD8RE

Macro: 8-Bit Data Register with Clock Enable and Synchronous Reset



Introduction

This design element is an 8-bit data register. When the clock enable (CE) input is High, and the synchronous reset (R) input is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q0) during the Low-to-High clock (C) transition. When R is High, it overrides all other inputs and resets the data outputs (Q) Low on the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

This register is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				**Outputs**
R	**CE**	**Dz – D0**	**C**	**Qz – Q0**
1	X	X	↑	0
0	0	X	X	No Change
0	1	Dn	↑	Dn
z = bit-width - 1				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

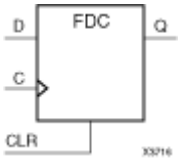
Attribute	**Type**	**Allowed Values**	**Default**	**Description**
INIT	Binary	Any 8-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDC

Primitive: D Flip-Flop with Asynchronous Clear



Introduction

This design element is a single D-type flip-flop with data (D) and asynchronous clear (CLR) inputs and data output (Q). The asynchronous CLR, when High, overrides all other inputs and sets the (Q) output Low. The data on the (D) input is loaded into the flip-flop when CLR is Low on the Low-to-High clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CLR	**D**	**C**	**Q**
1	X	X	0
0	D	↑	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

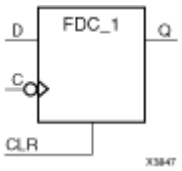
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0, 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDC_1

Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Clear



Introduction

FDC_1 is a single D-type flip-flop with data input (D), asynchronous clear input (CLR), and data output (Q). The asynchronous CLR, when active, overrides all other inputs and sets the (Q) output Low. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CLR	**D**	**C**	**Q**
1	X	X	0
0	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

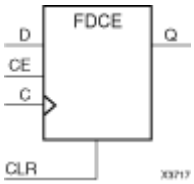
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0, 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDCE

Primitive: D Flip-Flop with Clock Enable and Asynchronous Clear



Introduction

This design element is a single D-type flip-flop with clock enable and asynchronous clear. When clock enable (CE) is High and asynchronous clear (CLR) is Low, the data on the data input (D) of this design element is transferred to the corresponding data output (Q) during the Low-to-High clock (C) transition. When CLR is High, it overrides all other inputs and resets the data output (Q) Low. When CE is Low, clock transitions are ignored.

For XC9500XL and XC9500XV devices, logic connected to the clock enable (CE) input may be implemented using the clock enable product term (p-term) in the macrocell, provided the logic can be completely implemented using the single p-term available for clock enable without requiring feedback from another macrocell. Only FDCE and FDPE flip-flops may take advantage of the clock-enable p-term.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**CE**	**D**	**C**	**Q**
1	X	X	X	0
0	0	X	X	No Change
0	1	D	↑	D

Design Entry Method

This design element can be used in schematics.

Available Attributes

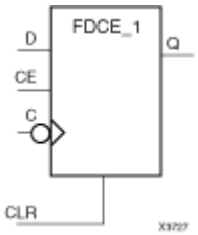
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0,1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDCE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Clear



Introduction

This design element is a single D-type flip-flop with data (D), clock enable (CE), asynchronous clear (CLR) inputs, and data output (Q). The asynchronous CLR input, when High, overrides all other inputs and sets the Q output Low. The data on the (D) input is loaded into the flip-flop when CLR is Low and CE is High on the High-to-Low clock (C) transition. When CE is Low, the clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**CE**	**D**	**C**	**Q**
1	X	X	X	0
0	0	X	?	No Change
0	1	1	?	1
0	1	0	?	0

Design Entry Method

This design element can be used in schematics.

Available Attributes

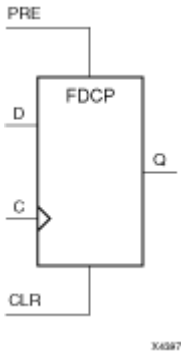
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0,1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDCP

Primitive: D Flip-Flop with Asynchronous Preset and Clear



Introduction

This design element is a single D-type flip-flop with data (D), asynchronous preset (PRE) and clear (CLR) inputs, and data output (Q). The asynchronous PRE, when High, sets the (Q) output High; CLR, when High, resets the output Low. Data on the (D) input is loaded into the flip-flop when PRE and CLR are Low on the Low-to-High clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**PRE**	**D**	**C**	**Q**
1	X	X	X	0
0	1	X	X	1
0	0	D	↑	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

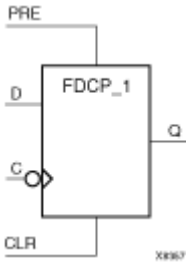
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

FDCP_1

Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Preset and Clear



Introduction

This design element is a single D-type flip-flop with data (D), asynchronous preset (PRE) and clear (CLR) inputs, and data output (Q). The asynchronous PRE, when High, sets the (Q) output High; CLR, when High, resets the output Low. Data on the (D) input is loaded into the flip-flop when PRE and CLR are Low on the High-to-Low clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**PRE**	**D**	**C**	**Q**
1	X	X	X	0
0	1	X	X	1
0	0	0	↓	0
0	0	1	↓	1

Design Entry Method

This design element is only for use in schematics.

Available Attributes

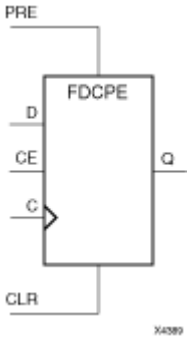
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0,1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDCPE

Primitive: D Flip-Flop with Clock Enable and Asynchronous Preset and Clear



Introduction

This design element is a single D-type flip-flop with data (D), clock enable (CE), asynchronous preset (PRE), and asynchronous clear (CLR) inputs. The asynchronous active high PRE sets the Q output High; that active high CLR resets the output Low and has precedence over the PRE input. Data on the D input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High clock (C) transition. When CE is Low, the clock transitions are ignored and the previous value is retained. The FDCPE is generally implemented as a slice or IOB register within the device.

For FPGA devices, upon power-up, the initial value of this component is specified by the INIT attribute. If a subsequent GSR (Global Set/Reset) is asserted, the flop is asynchronously set to the INIT value.

Note While this device supports the use of asynchronous set and reset, it is not generally recommended to be used for in most cases. Use of asynchronous signals pose timing issues within the design that are difficult to detect and control and also have an adverse affect on logic optimization causing a larger design that can consume more power than if a synchronous set or reset is used.

Logic Table

Inputs					Outputs
CLR	**PRE**	**CE**	**D**	**C**	**Q**
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	D	↑	D

Port Descriptions

Port	Direction	Width	Function
Q	Output	1	Data output
C	Input	1	Clock input
CE	Input	1	Clock enable input
CLR	Input	1	Asynchronous clear input

Port	Direction	Width	Function
D	Input	1	Data input
PRE	Input	1	Asynchronous set input

Design Entry Method

This design element can be used in schematics.

Available Attributes

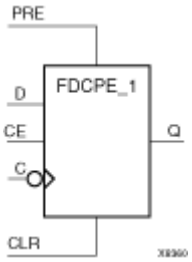
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0,1	0	Sets the initial value of Q output after configuration and on GSR.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDCPE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset and Clear



Introduction

FDCPE_1 is a single D-type flip-flop with data (D), clock enable (CE), asynchronous preset (PRE), and asynchronous clear (CLR) inputs and data output (Q). The asynchronous PRE, when High, sets the (Q) output High; CLR, when High, resets the output Low. Data on the (D) input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the High-to-Low clock (C) transition. When CE is Low, the clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					**Outputs**
CLR	**PRE**	**CE**	**D**	**C**	**Q**
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	D	↓	D

Port Descriptions

Port	**Direction**	**Width**	**Function**
Q	Output	1	Data output
C	Input	1	Clock input
CE	Input	1	Clock enable input
CLR	Input	1	Asynchronous clear input
D	Input	1	Data input
PRE	Input	1	Asynchronous set input

Design Entry Method

This design element can be used in schematics.

Available Attributes

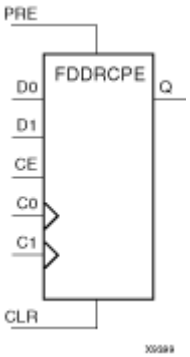
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0,1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDDRCPE

Primitive: Dual Data Rate D Flip-Flop with Clock Enable and Asynchronous Preset and Clear



Introduction

FDDRCPE is a Dual Data Rate (DDR) D flip-flop with two separate clocks (C0 and C1) phase shifted 180 degrees that allow selection of two separate data inputs (D0 and D1). It also has clock enable (CE), asynchronous preset (PRE), and asynchronous clear (CLR) inputs and data output (Q). The asynchronous PRE, when High, sets the Q output High; CLR, when High, resets the output Low. Data on the D0 input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High C0 clock transition. Data on the D1 input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High C1 clock transition. When CE is Low, the clock transitions are ignored.

Use the INIT attribute to initialize FDDRCPE during configuration.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs							Outputs
C0	**C1**	**CE**	**D0**	**D1**	**CLR**	**PRE**	**Q**
X	X	X	X	X	1	0	0
X	X	X	X	X	0	1	1
X	X	X	X	X	1	1	0
X	X	0	X	X	0	0	No Change
↑	X	1	D0	X	0	0	D0
X	↑	1	X	D1	0	0	D1

Design Entry Method

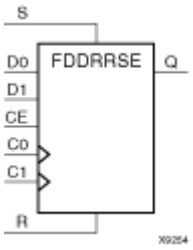
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDDRRSE

Primitive: Dual Data Rate D Flip-Flop with Clock Enable and Synchronous Reset and Set



Introduction

FDDRRSE is a Dual Data Rate (DDR) D flip-flop with two separate clocks (C0 and C1) phase shifted 180 degrees that allow selection of two separate data inputs (D0 and D1). It also has synchronous reset (R), synchronous set (S), and clock enable (CE) inputs and data output (Q). The reset (R) input, when High, overrides all other inputs and resets the Q output Low during any Low-to-High clock transition (C0 or C1). (Reset has precedence over Set.) When the S input is High and R is Low, the flip-flop is set, output High, during a Low-to-High clock transition (C0 or C1). Data on the D0 input is loaded into the flip-flop when R and S are Low and CE is High during the Low-to-High C0 clock transition. Data on the D1 input is loaded into the flip-flop when R and S are Low and CE is High during the Low-to-High C1 clock transition.

Use the INIT attribute to initialize FDDRRSE during configuration.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs							Outputs
C0	**C1**	**CE**	**D0**	**D1**	**R**	**S**	**Q**
↑	X	X	X	X	1	0	0
↑	X	X	X	X	0	1	1
↑	X	X	X	X	1	1	0
X	↑	X	X	X	1	0	0
X	↑	X	X	X	0	1	1
X	↑	X	X	X	1	1	0
X	X	0	X	X	0	0	No Change
↑	X	1	D0	X	0	0	D0
X	↑	1	X	D1	0	0	D1

Design Entry Method

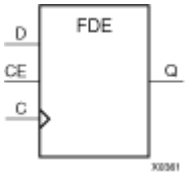
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDE

Primitive: D Flip-Flop with Clock Enable



Introduction

This design element is a single D-type flip-flop with data input (D), clock enable (CE), and data output (Q). When clock enable is High, the data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CE	**D**	**C**	**Q**
0	X	X	No Change
1	0	↑	0
1	1	↑	1

Design Entry Method

This design element is only for use in schematics.

Available Attributes

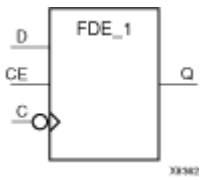
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDE_1

Primitive: D Flip-Flop with Negative-Edge Clock and Clock Enable



Introduction

This design element is a single D-type flip-flop with data input (D), clock enable (CE), and data output (Q). When clock enable is High, the data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CE	**D**	**C**	**Q**
0	X	X	No Change
1	0	↓	0
1	1	↓	1

Design Entry Method

This design element is only for use in schematics.

Available Attributes

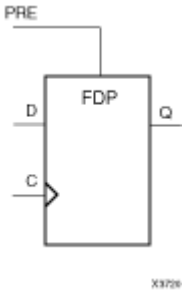
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDP

Primitive: D Flip-Flop with Asynchronous Preset



Introduction

This design element is a single D-type flip-flop with data (D) and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and presets the (Q) output High. The data on the (D) input is loaded into the flip-flop when PRE is Low on the Low-to-High clock (C) transition.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
PRE	**C**	**D**	**Q**
1	X	X	1
0	↑	D	D
0	↑	0	0

Design Entry Method

This design element is only for use in schematics.

Available Attributes

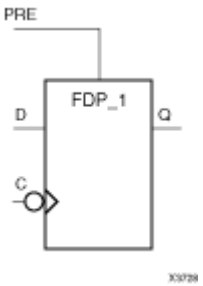
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDP_1

Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Preset



Introduction

This design element is a single D-type flip-flop with data (D) and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and presets the Q output High. The data on the D input is loaded into the flip-flop when PRE is Low on the High-to-Low clock (C) transition.

This flip-flop is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
PRE	**C**	**D**	**Q**
1	X	X	1
0	↓	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

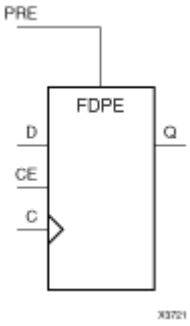
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDPE

Primitive: D Flip-Flop with Clock Enable and Asynchronous Preset



Introduction

This design element is a single D-type flip-flop with data (D), clock enable (CE), and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and sets the (Q) output High. Data on the (D) input is loaded into the flip-flop when PRE is Low and CE is High on the Low-to-High clock (C) transition. When CE is Low, the clock transitions are ignored.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
PRE	**CE**	**D**	**C**	**Q**
1	X	X	X	1
0	0	X	X	No Change
0	1	D	↑	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

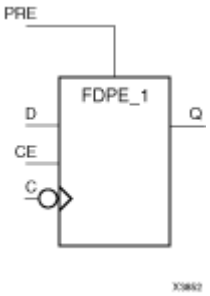
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDPE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset



Introduction

This design element is a single D-type flip-flop with data (D), clock enable (CE), and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and sets the (Q) output High. Data on the (D) input is loaded into the flip-flop when PRE is Low and CE is High on the High-to-Low clock (C) transition. When CE is Low, the clock transitions are ignored.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
PRE	**CE**	**D**	**C**	**Q**
1	X	X	X	1
0	0	X	X	No Change
0	1	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

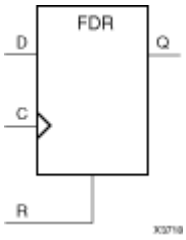
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDR

Primitive: D Flip-Flop with Synchronous Reset



Introduction

This design element is a single D-type flip-flop with data (D) and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the Low-to-High clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low during the Low-to- High clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
R	**D**	**C**	**Q**
1	X	↑	0
0	D	↑	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

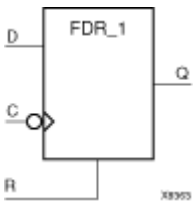
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDR_1

Primitive: D Flip-Flop with Negative-Edge Clock and Synchronous Reset



Introduction

This design element is a single D-type flip-flop with data (D) and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the High-to-Low clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low during the High-to- Low clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
R	**D**	**C**	**Q**
1	X	↓	0
0	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

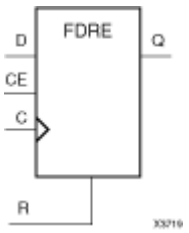
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDRE

Primitive: D Flip-Flop with Clock Enable and Synchronous Reset



Introduction

This design element is a single D-type flip-flop with data (D), clock enable (CE), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the Low-to-High clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low and CE is High during the Low-to-High clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
R	**CE**	**D**	**C**	**Q**
1	X	X	↑	0
0	0	X	X	No Change
0	1	D	↑	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

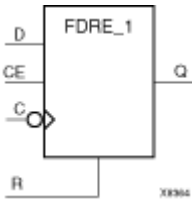
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDRE_1

Primitive: D Flip-Flop with Negative-Clock Edge, Clock Enable, and Synchronous Reset



Introduction

FDRE_1 is a single D-type flip-flop with data (D), clock enable (CE), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the High-to-Low clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low and CE is High during the High-to-Low clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
R	**CE**	**D**	**C**	**Q**
1	X	X	↓	0
0	0	X	X	No Change
0	1	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

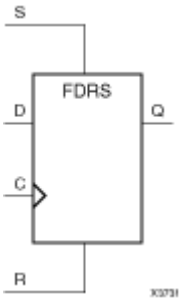
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDRS

Primitive: D Flip-Flop with Synchronous Reset and Set



Introduction

FDRS is a single D-type flip-flop with data (D), synchronous set (S), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low during the Low-to-High clock (C) transition. (Reset has precedence over Set.) When S is High and R is Low, the flip-flop is set, output High, during the Low-to-High clock transition. When R and S are Low, data on the (D) input is loaded into the flip-flop during the Low-to-High clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
R	**S**	**D**	**C**	**Q**
1	X	X	↓	0
0	1	X	↓	1
0	0	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

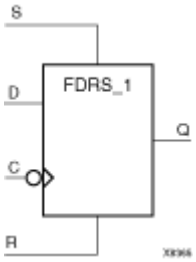
Attribute	Type	Allowed Values	Default	Description
IN	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDRS_1

Primitive: D Flip-Flop with Negative-Clock Edge and Synchronous Reset and Set



Introduction

FDRS_1 is a single D-type flip-flop with data (D), synchronous set (S), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low during the High-to-Low clock (C) transition. (Reset has precedence over Set.) When S is High and R is Low, the flip-flop is set, output High, during the High-to-Low clock transition. When R and S are Low, data on the (D) input is loaded into the flip-flop during the High-to-Low clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
R	**S**	**D**	**C**	**Q**
1	X	X	↓	0
0	1	X	↓	1
0	0	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

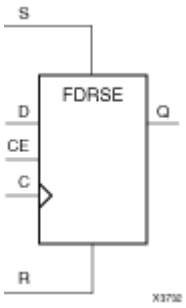
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDRSE

Primitive: D Flip-Flop with Synchronous Reset and Set and Clock Enable



Introduction

FDRSE is a single D-type flip-flop with synchronous reset (R), synchronous set (S), clock enable (CE) inputs. The reset (R) input, when High, overrides all other inputs and resets the Q output Low during the Low-to-High clock transition. (Reset has precedence over Set.) When the set (S) input is High and R is Low, the flip-flop is set, output High, during the Low-to-High clock (C) transition. Data on the D input is loaded into the flip-flop when R and S are Low and CE is High during the Low-to-High clock transition.

Upon power-up, the initial value of this component is specified by the INIT attribute. If a subsequent GSR (Global Set/Reset) is asserted, the flop is asynchronously set to the INIT value.

Logic Table

Inputs					Outputs
R	**S**	**CE**	**D**	**C**	**Q**
1	X	X	X	↑	0
0	1	X	X	↑	1
0	0	0	X	X	No Change
0	0	1	1	↑	1
0	0	1	0	↑	0

Design Entry Method

This design element can be used in schematics.

Available Attributes

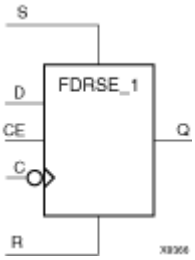
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration and on GSR.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDRSE_1

Primitive: D Flip-Flop with Negative-Clock Edge, Synchronous Reset and Set, and Clock Enable



Introduction

FDRSE_1 is a single D-type flip-flop with synchronous reset (R), synchronous set (S), and clock enable (CE) inputs and data output (Q). The reset (R) input, when High, overrides all other inputs and resets the (Q) output Low during the High-to-Low clock transition. (Reset has precedence over Set.) When the set (S) input is High and R is Low, the flip-flop is set, output High, during the High-to-Low clock (C) transition. Data on the (D) input is loaded into the flip-flop when (R) and (S) are Low and (CE) is High during the High-to-Low clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs
R	**S**	**CE**	**D**	**C**	**Q**
1	X	X	X	↓	0
0	1	X	X	↓	1
0	0	0	X	X	No Change
0	0	1	D	↓	D

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration and on GSR.

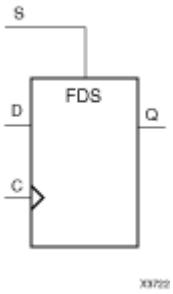
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

FDS

Primitive: D Flip-Flop with Synchronous Set



Introduction

FDS is a single D-type flip-flop with data (D) and synchronous set (S) inputs and data output (Q). The synchronous set input, when High, sets the Q output High on the Low-to-High clock (C) transition. The data on the D input is loaded into the flip-flop when S is Low during the Low-to-High clock (C) transition.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
S	**D**	**C**	**Q**
1	X	↑	1
0	D	↑	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

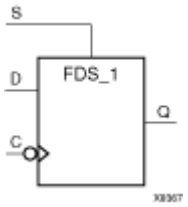
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDS_1

Primitive: D Flip-Flop with Negative-Edge Clock and Synchronous Set



Introduction

FDS is a single D-type flip-flop with data (D) and synchronous set (S) inputs and data output (Q). The synchronous set input, when High, sets the Q output High on the Low-to-High clock (C) transition. The data on the D input is loaded into the flip-flop when S is Low during the Low-to-High clock (C) transition.

This flip-flop is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
S	**D**	**C**	**Q**
1	X	↓	1
0	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

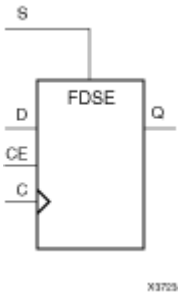
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDSE

Primitive: D Flip-Flop with Clock Enable and Synchronous Set



Introduction

FDSE is a single D-type flip-flop with data (D), clock enable (CE), and synchronous set (S) inputs and data output (Q). The synchronous set (S) input, when High, overrides the clock enable (CE) input and sets the Q output High during the Low-to-High clock (C) transition. The data on the D input is loaded into the flip-flop when S is Low and CE is High during the Low-to-High clock (C) transition.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
S	**CE**	**D**	**C**	**Q**
1	X	X	↑	1
0	0	X	X	No Change
0	1	D	↑	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

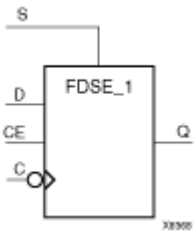
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FDSE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Synchronous Set



Introduction

FDSE_1 is a single D-type flip-flop with data (D), clock enable (CE), and synchronous set (S) inputs and data output (Q). The synchronous set (S) input, when High, overrides the clock enable (CE) input and sets the Q output High during the High-to-Low clock (C) transition. The data on the D input is loaded into the flip-flop when S is Low and CE is High during the High-to-Low clock (C) transition.

This flip-flop is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
S	**CE**	**D**	**C**	**Q**
1	X	X	↓	1
0	0	X	X	No Change
0	1	D	↓	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

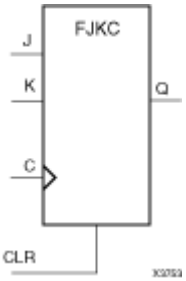
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FJKC

Macro: J-K Flip-Flop with Asynchronous Clear



Introduction

This design element is a single J-K-type flip-flop with J, K, and asynchronous clear (CLR) inputs and data output (Q). The asynchronous clear (CLR) input, when High, overrides all other inputs and resets the Q output Low. When CLR is Low, the output responds to the state of the J and K inputs, as shown in the following logic table, during the Low-to-High clock (C) transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**J**	**K**	**C**	**Q**
1	X	X	X	0
0	0	0	↑	No Change
0	0	1	↑	0
0	1	0	↑	1
0	1	1	↑	Toggle

Design Entry Method

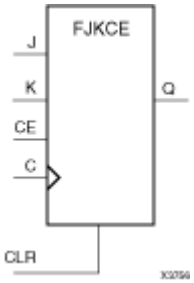
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FJKCE

Macro: J-K Flip-Flop with Clock Enable and Asynchronous Clear



Introduction

This design element is a single J-K-type flip-flop with J, K, clock enable (CE), and asynchronous clear (CLR) inputs and data output (Q). The asynchronous clear (CLR), when High, overrides all other inputs and resets the Q output Low. When CLR is Low and CE is High, Q responds to the state of the J and K inputs, as shown in the following logic table, during the Low-to-High clock transition. When CE is Low, the clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					**Outputs**
CLR	**CE**	**J**	**K**	**C**	**Q**
1	X	X	X	X	0
0	0	X	X	X	No Change
0	1	0	0	X	No Change
0	1	0	1	↑	0
0	1	1	0	↑	1
0	1	1	1	↑	Toggle

Design Entry Method

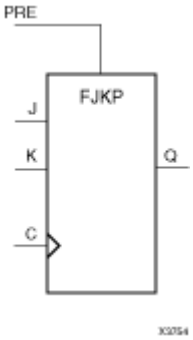
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FJKP

Macro: J-K Flip-Flop with Asynchronous Preset



Introduction

This design element is a single J-K-type flip-flop with J, K, and asynchronous preset (PRE) inputs and data output (Q). The asynchronous preset (PRE) input, when High, overrides all other inputs and sets the (Q) output High. When (PRE) is Low, the (Q) output responds to the state of the J and K inputs, as shown in the following logic table, during the Low-to-High clock transition.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
PRE	**J**	**K**	**C**	**Q**
1	X	X	X	1
0	0	0	X	No Change
0	0	1	↑	0
0	1	0	↑	1
0	1	1	↑	Toggle

Design Entry Method

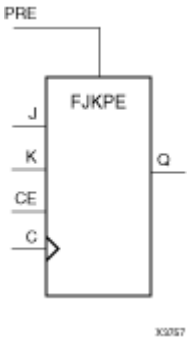
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FJKPE

Macro: J-K Flip-Flop with Clock Enable and Asynchronous Preset



Introduction

This design element is a single J-K-type flip-flop with J, K, clock enable (CE), and asynchronous preset (PRE) inputs and data output (Q). The asynchronous preset (PRE), when High, overrides all other inputs and sets the (Q) output High. When (PRE) is Low and (CE) is High, the (Q) output responds to the state of the J and K inputs, as shown in the logic table, during the Low-to-High clock (C) transition. When (CE) is Low, clock transitions are ignored.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs
PRE	**CE**	**J**	**K**	**C**	**Q**
1	X	X	X	X	1
0	0	X	X	X	No Change
0	1	0	0	X	No Change
0	1	0	1	↑	0
0	1	1	0	↑	1
0	1	1	1	↑	Toggle

Design Entry Method

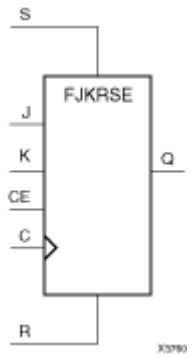
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FJKRSE

Macro: J-K Flip-Flop with Clock Enable and Synchronous Reset and Set



Introduction

This design element is a single J-K-type flip-flop with J, K, synchronous reset (R), synchronous set (S), and clock enable (CE) inputs and data output (Q). When synchronous reset (R) is High during the Low-to-High clock (C) transition, all other inputs are ignored and output (Q) is reset Low. When synchronous set (S) is High and (R) is Low, output (Q) is set High. When (R) and (S) are Low and (CE) is High, output (Q) responds to the state of the J and K inputs, according to the following logic table, during the Low-to-High clock (C) transition. When (CE) is Low, clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs
R	**S**	**CE**	**J**	**K**	**C**	**Q**
1	X	X	X	X	↑	0
0	1	X	X	X	↑	1
0	0	0	X	X	X	No Change
0	0	1	0	0	X	No Change
0	0	1	0	1	↑	0
0	0	1	1	1	↑	Toggle
0	0	1	1	0	↑	1

Design Entry Method

This design element is only for use in schematics.

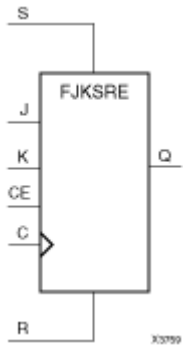
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

FJKSRE

Macro: J-K Flip-Flop with Clock Enable and Synchronous Set and Reset



Introduction

This design element is a single J-K-type flip-flop with J, K, synchronous set (S), synchronous reset (R), and clock enable (CE) inputs and data output (Q). When synchronous set (S) is High during the Low-to-High clock (C) transition, all other inputs are ignored and output (Q) is set High. When synchronous reset (R) is High and (S) is Low, output (Q) is reset Low. When (S) and (R) are Low and (CE) is High, output (Q) responds to the state of the J and K inputs, as shown in the following logic table, during the Low-to-High clock (C) transition. When (CE) is Low, clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						**Outputs**
S	**R**	**CE**	**J**	**K**	**C**	**Q**
1	X	X	X	X	↑	1
0	1	X	X	X	↑	0
0	0	0	X	X	X	No Change
0	0	1	0	0	X	No Change
0	0	1	0	1	↑	0
0	0	1	1	0	↑	1
0	0	1	1	1	↑	Toggle

Design Entry Method

This design element is only for use in schematics.

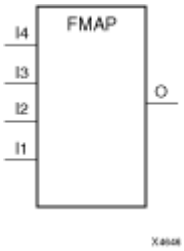
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

FMAP

Primitive: F Function Generator Partitioning Control Symbol



Introduction

The FMAP symbol is used to map logic to the function generator of a slice. See the appropriate CAE tool interface user guide for information about specifying this attribute in your schematic design editor.

The MAP= *type* parameter can be used with the FMAP symbol to further define how much latitude you want to give the mapping program. The following table shows MAP option characters and their meanings

MAP Option Character	Function
P	Pins.
C	Closed — Adding logic to or removing logic from the CLB is not allowed.
L	Locked — Locking CLB pins.
O	Open — Adding logic to or removing logic from the CLB is allowed.
U	Unlocked — No locking on CLB pins.

Possible types of MAP parameters for FMAP are MAP=PUC, MAP=PLC, MAP=PLO, and MAP=PUO. The default parameter is PUO. If one of the “open” parameters is used (PLO or PUO), only the output signals must be specified.

Note Currently, only PUC and PUO are observed. PLC and PLO are translated into PUC and PUO, respectively.

The FMAP symbol can be assigned to specific CLB locations using LOC attributes.

Design Entry Method

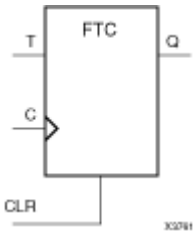
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTC

Macro: Toggle Flip-Flop with Asynchronous Clear



Introduction

This design element is a synchronous, resettable toggle flip-flop. The asynchronous clear (CLR) input, when High, overrides all other inputs and resets the data output (Q) Low. The (Q) output toggles, or changes state, when the toggle enable (T) input is High and (CLR) is Low during the Low-to-High clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CLR	**T**	**C**	**Q**
1	X	X	0
0	0	X	No Change
0	1	↑	Toggle

Design Entry Method

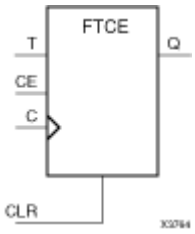
You can instantiate this element when targeting a CPLD, but not when you are targeting an FPGA.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTCE

Macro: Toggle Flip-Flop with Clock Enable and Asynchronous Clear



Introduction

This design element is a toggle flip-flop with toggle and clock enable and asynchronous clear. When the asynchronous clear (CLR) input is High, all other inputs are ignored and the data output (Q) is reset Low. When CLR is Low and toggle enable (T) and clock enable (CE) are High, Q output toggles, or changes state, during the Low-to-High clock (C) transition. When CE is Low, clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				**Outputs**
CLR	**CE**	**T**	**C**	**Q**
1	X	X	X	0
0	0	X	X	No Change
0	1	0	X	No Change
0	1	1	↑	Toggle

Design Entry Method

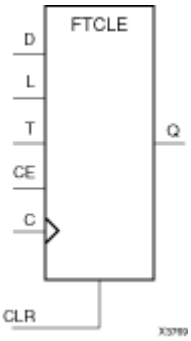
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTCLE

Macro: Toggle/Loadable Flip-Flop with Clock Enable and Asynchronous Clear



Introduction

This design element is a toggle/loadable flip-flop with toggle and clock enable and asynchronous clear. When the asynchronous clear input (CLR) is High, all other inputs are ignored and output Q is reset Low. When load enable input (L) is High and CLR is Low, clock enable (CE) is overridden and the data on data input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition. When toggle enable (T) and CE are High and L and CLR are Low, output Q toggles, or changes state, during the Low- to-High clock transition. When CE is Low, clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs
CLR	**L**	**CE**	**T**	**D**	**C**	**Q**
1	X	X	X	X	X	0
0	1	X	X	D	↑	D
0	0	0	X	X	X	No Change
0	0	1	0	X	X	No Change
0	0	1	1	X	↑	Toggle

Design Entry Method

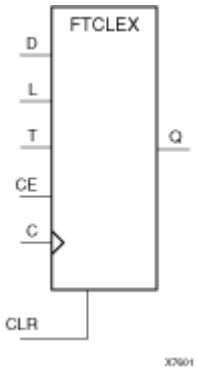
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTCLEX

Macro: Toggle/Loadable Flip-Flop with Clock Enable and Asynchronous Clear



Introduction

This design element is a toggle/loadable flip-flop with toggle and clock enable and asynchronous clear. When the asynchronous clear input (CLR) is High, all other inputs are ignored and output Q is reset Low. When load enable input (L) is High, CLR is Low, and CE is High, the data on data input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition. When toggle enable (T) and CE are High and L and CLR are Low, output Q toggles, or changes state, during the Low- to-High clock transition. When CE is Low, clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs
CLR	**L**	**CE**	**T**	**D**	**C**	**Q**
1	X	X	X	X	X	0
0	1	X	X	D	↑	D
0	0	0	X	X	X	No Change
0	0	1	0	X	X	No Change
0	0	1	1	X	↑	Toggle

Design Entry Method

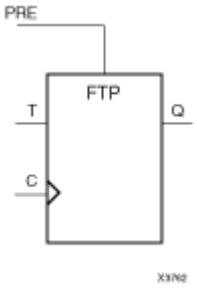
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTP

Macro: Toggle Flip-Flop with Asynchronous Preset



Introduction

This design element is a toggle flip-flop with toggle enable and asynchronous preset. When the asynchronous preset (PRE) input is High, all other inputs are ignored and output (Q) is set High. When toggle-enable input (T) is High and (PRE) is Low, output (Q) toggles, or changes state, during the Low-to-High clock (C) transition.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
PRE	**T**	**C**	**Q**
1	X	X	1
0	0	X	No Change
0	1	↑	Toggle

Design Entry Method

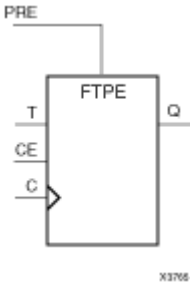
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTPE

Macro: Toggle Flip-Flop with Clock Enable and Asynchronous Preset



Introduction

This design element is a toggle flip-flop with toggle and clock enable and asynchronous preset. When the asynchronous preset (PRE) input is High, all other inputs are ignored and output (Q) is set High. When the toggle enable input (T) is High, clock enable (CE) is High, and (PRE) is Low, output (Q) toggles, or changes state, during the Low-to-High clock transition. When (CE) is Low, clock transitions are ignored.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
PRE	**CE**	**T**	**C**	**Q**
1	X	X	X	1
0	0	X	X	No Change
0	1	0	X	No Change
0	1	1	↑	Toggle

Design Entry Method

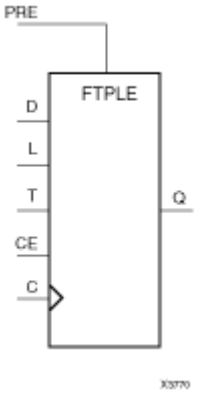
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTPLE

Macro: Toggle/Loadable Flip-Flop with Clock Enable and Asynchronous Preset



Introduction

This design element is a toggle/loadable flip-flop with toggle and clock enable and asynchronous preset. When the asynchronous preset input (PRE) is High, all other inputs are ignored and output (Q) is set High. When the load enable input (L) is High and (PRE) is Low, the clock enable (CE) is overridden and the data (D) is loaded into the flip-flop during the Low-to-High clock transition. When L and PRE are Low and toggle-enable input (T) and (CE) are High, output (Q) toggles, or changes state, during the Low-to-High clock transition. When (CE) is Low, clock transitions are ignored.

For FPGA devices, this flip-flop is asynchronously preset, output High, when power is applied. Power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs
PRE	**L**	**CE**	**T**	**D**	**C**	**Q**
1	X	X	X	X	X	1
0	1	X	X	D	↑	D
0	0	0	X	X	X	No Change
0	0	1	0	X	X	No Change
0	0	1	1	X	↑	Toggle

Design Entry Method

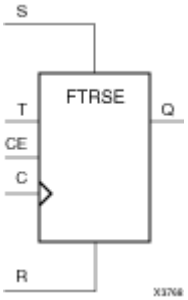
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTRSE

Macro: Toggle Flip-Flop with Clock Enable and Synchronous Reset and Set



Introduction

This design element is a toggle flip-flop with toggle and clock enable and synchronous reset and set. When the synchronous reset input (R) is High, it overrides all other inputs and the data output (Q) is reset Low. When the synchronous set input (S) is High and (R) is Low, clock enable input (CE) is overridden and output (Q) is set High. (Reset has precedence over Set.) When toggle enable input (T) and (CE) are High and (R) and (S) are Low, output (Q) toggles, or changes state, during the Low-to-High clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs
R	**S**	**CE**	**T**	**C**	**Q**
1	X	X	X	↑	0
0	1	X	X	↑	1
0	0	0	X	X	No Change
0	0	1	0	X	No Change
0	0	1	1	↑	Toggle

Design Entry Method

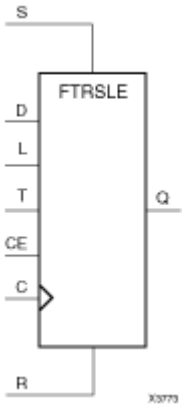
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTRSLE

Macro: Toggle/Loadable Flip-Flop with Clock Enable and Synchronous Reset and Set



Introduction

This design element is a toggle/loadable flip-flop with toggle and clock enable and synchronous reset and set. The synchronous reset input (R), when High, overrides all other inputs and resets the data output (Q) Low. (Reset has precedence over Set.) When R is Low and synchronous set input (S) is High, the clock enable input (CE) is overridden and output Q is set High. When R and S are Low and load enable input (L) is High, CE is overridden and data on data input (D) is loaded into the flip-flop during the Low-to-High clock transition. When R, S, and L are Low, CE is High and T is High, output Q toggles, or changes state, during the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs							Outputs
R	**S**	**L**	**CE**	**T**	**D**	**C**	**Q**
1	0	X	X	X	X	↑	0
0	1	X	X	X	X	↑	1
0	0	1	X	X	1	↑	1
0	0	1	X	X	0	↑	0
0	0	0	0	X	X	X	No Change
0	0	0	1	0	X	X	No Change
0	0	0	1	1	X	↑	Toggle

Design Entry Method

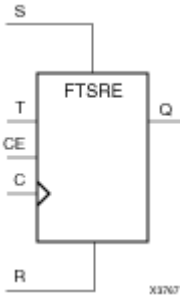
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTSRE

Macro: Toggle Flip-Flop with Clock Enable and Synchronous Set and Reset



Introduction

This design element is a toggle flip-flop with toggle and clock enable and synchronous set and reset. The synchronous set input, when High, overrides all other inputs and sets data output (Q) High. (Set has precedence over Reset.) When synchronous reset input (R) is High and S is Low, clock enable input (CE) is overridden and output Q is reset Low. When toggle enable input (T) and CE are High and S and R are Low, output Q toggles, or changes state, during the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs
S	**R**	**CE**	**T**	**C**	**Q**
1	X	X	X	↑	1
0	1	X	X	↑	0
0	0	0	X	X	No Change
0	0	1	0	X	No Change
0	0	1	1	↑	Toggle

Design Entry Method

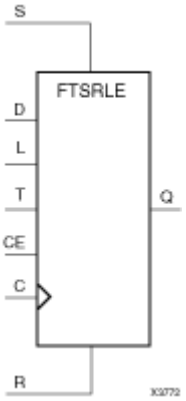
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

FTSRLE

Macro: Toggle/Loadable Flip-Flop with Clock Enable and Synchronous Set and Reset



Introduction

This design element is a toggle/loadable flip-flop with toggle and clock enable and synchronous set and reset. The synchronous set input (S), when High, overrides all other inputs and sets data output (Q) High. (Set has precedence over Reset.) When synchronous reset (R) is High and (S) is Low, clock enable input (CE) is overridden and output (Q) is reset Low. When load enable input (L) is High and S and R are Low, CE is overridden and data on data input (D) is loaded into the flip-flop during the Low-to-High clock transition. When the toggle enable input (T) and (CE) are High and (S), (R), and (L) are Low, output (Q) toggles, or changes state, during the Low-to-High clock transition. When (CE) is Low, clock transitions are ignored.

For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs							Outputs
S	**R**	**L**	**CE**	**T**	**D**	**C**	**Q**
1	X	X	X	X	X	↑	1
0	1	X	X	X	X	↑	0
0	0	1	X	X	1	↑	1
0	0	1	X	X	0	↑	0
0	0	0	0	X	X	X	No Change
0	0	0	1	0	X	X	No Change
0	0	0	1	1	X	↑	Toggle

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GND

Primitive: Ground-Connection Signal Tag

Introduction

The GND signal tag, or parameter, forces a net or input function to a Low logic level. A net tied to GND cannot have any other source.

When the logic-trimming software or fitter encounters a net or input function tied to GND, it removes any logic that is disabled by the GND signal. The GND signal is only implemented when the disabled logic cannot be removed.

Design Entry Method

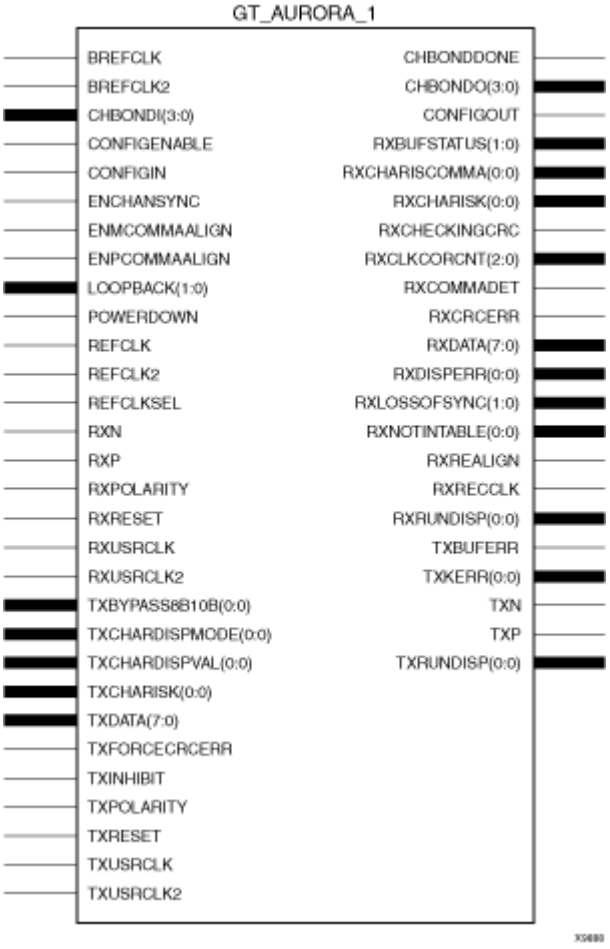
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_AURORA_1

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

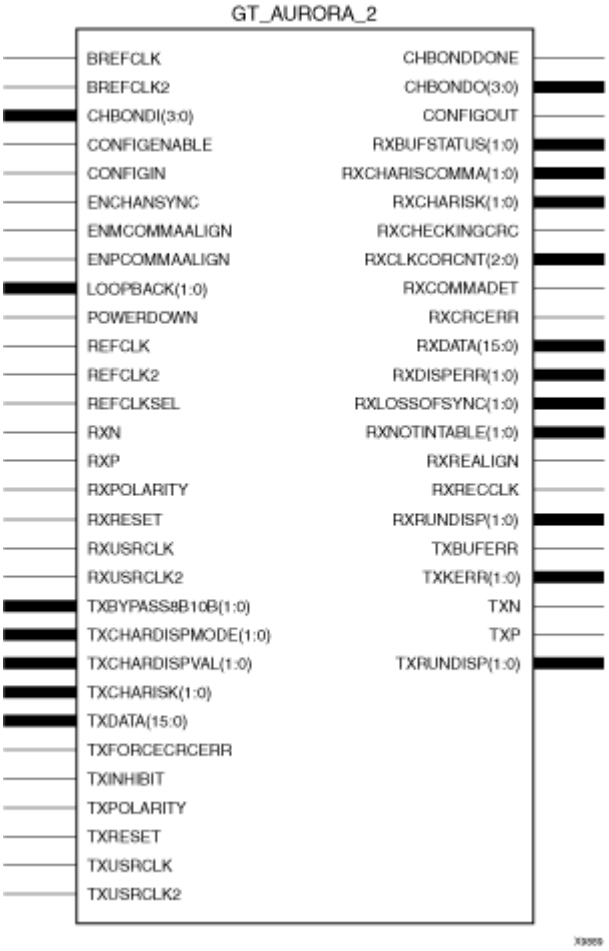
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_AURORA_2

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

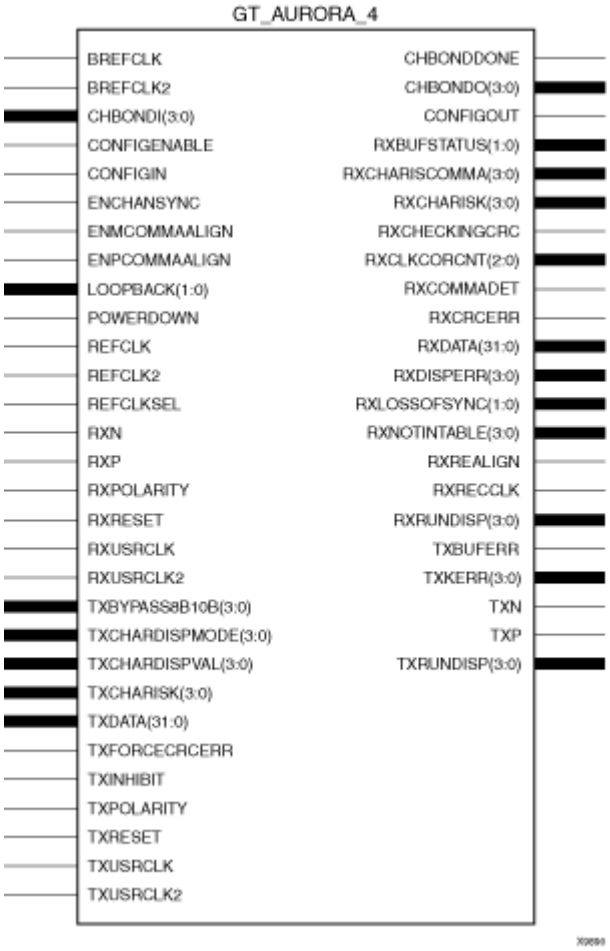
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_AURORA_4

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

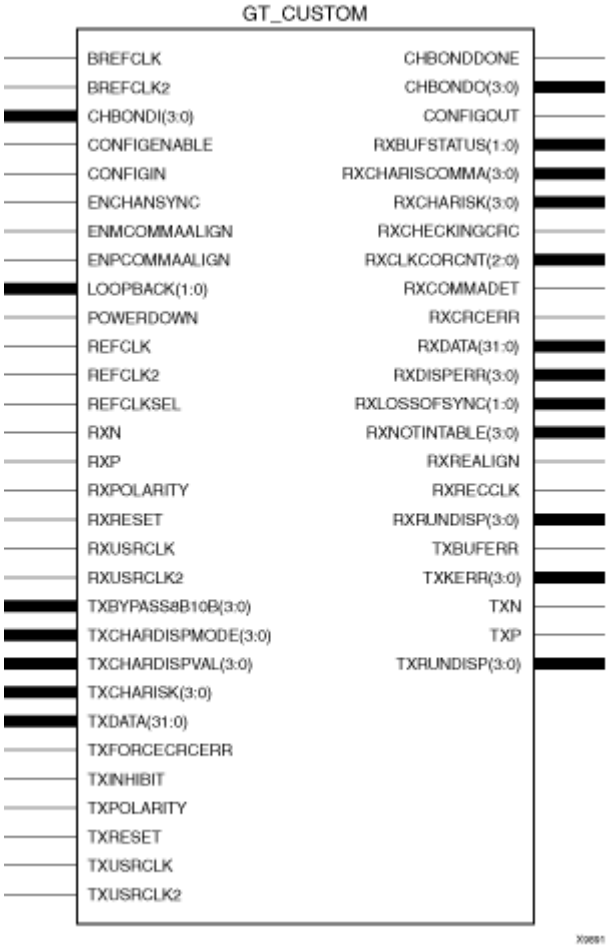
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_CUSTOM

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This gigabit transceiver is fully customizable. You can set attributes for the primitives. You can also set attributes for the primitives.

Design Entry Method

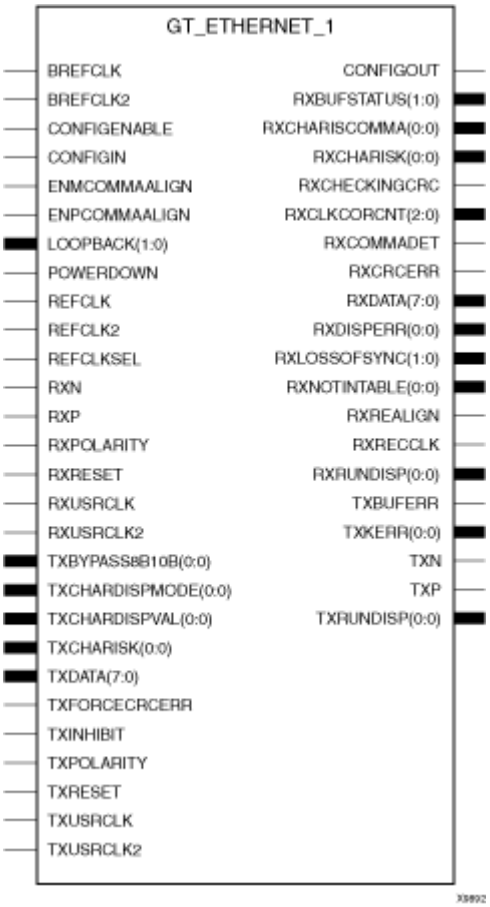
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_ETHERNET_1

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

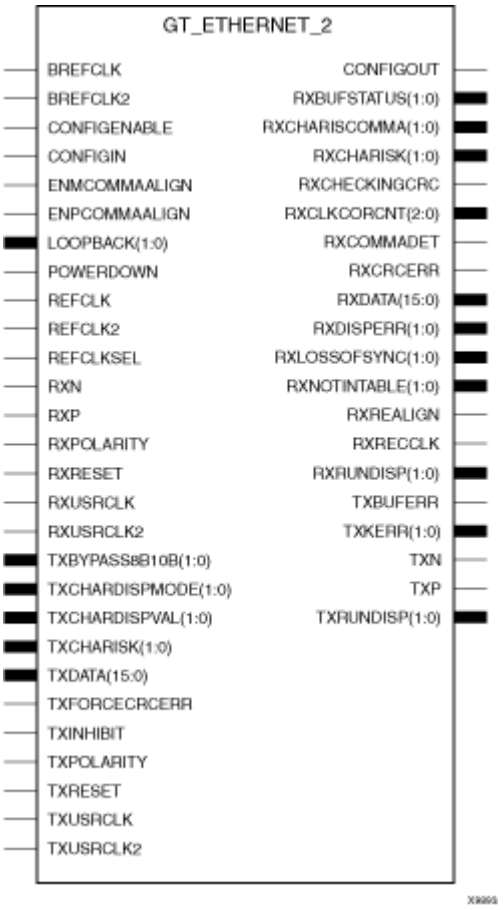
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_ETHERNET_2

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

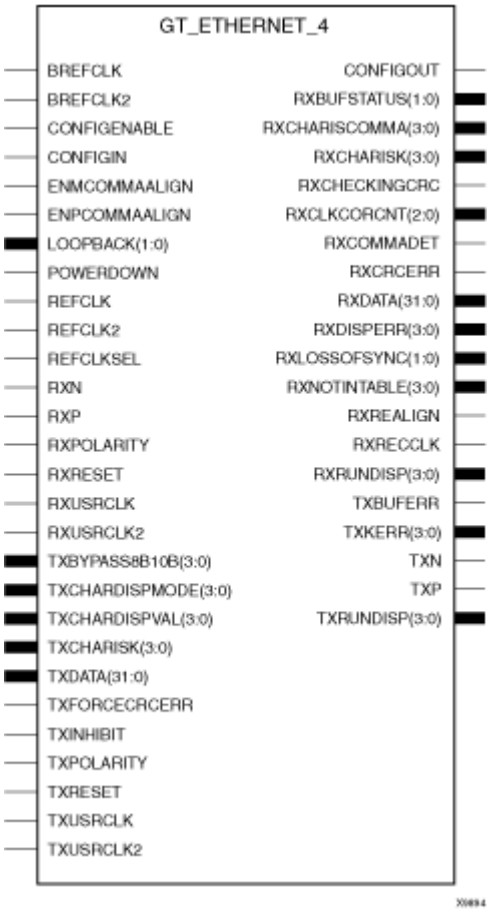
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_ETHERNET_4

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

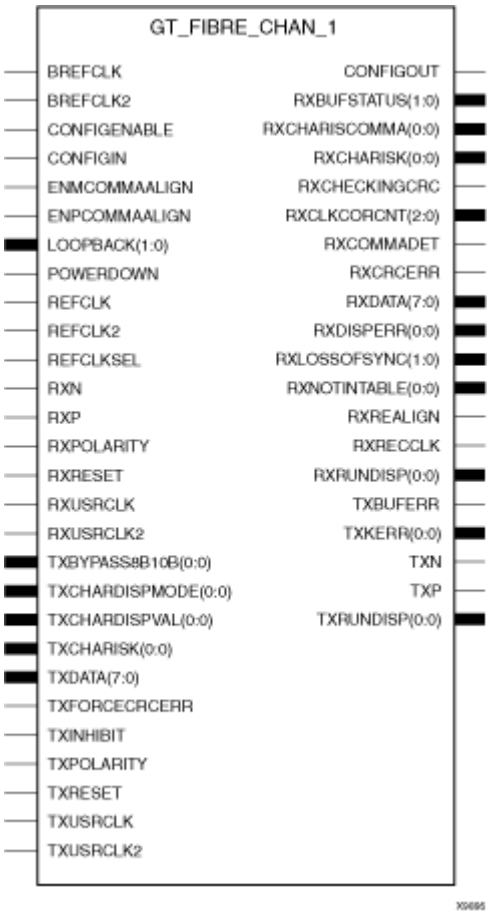
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_FIBRE_CHAN_1

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This Fibre Channel gigabit transceiver supports 1-byte data paths. You can also set attributes for the primitives.

Design Entry Method

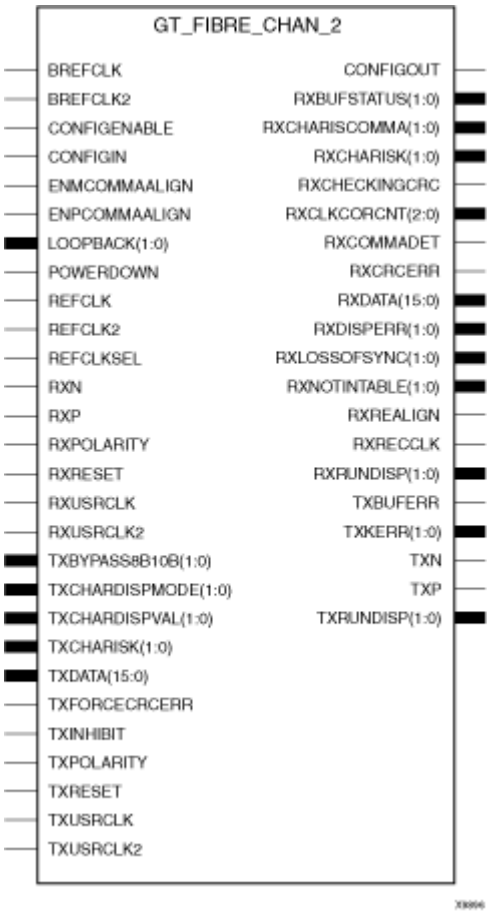
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_FIBRE_CHAN_2

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This Fibre Channel gigabit transceiver supports 2-byte data paths. You can also set attributes for the primitives.

Design Entry Method

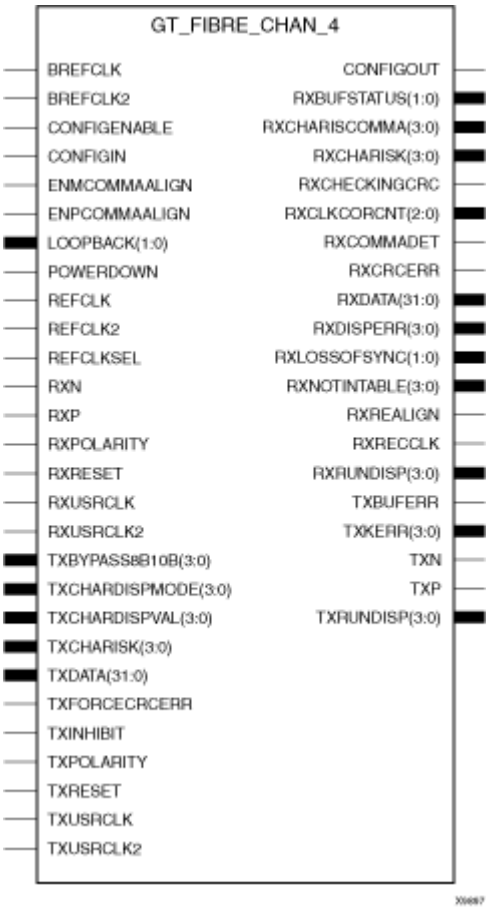
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_FIBRE_CHAN_4

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This Fibre Channel gigabit transceiver supports 4-byte data paths. You can also set attributes for the primitives.

Design Entry Method

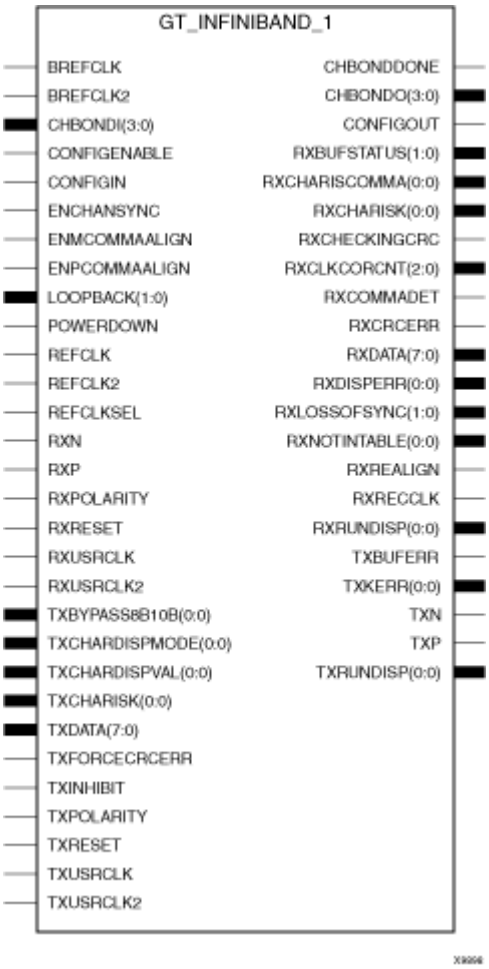
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_INFINIBAND_1

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

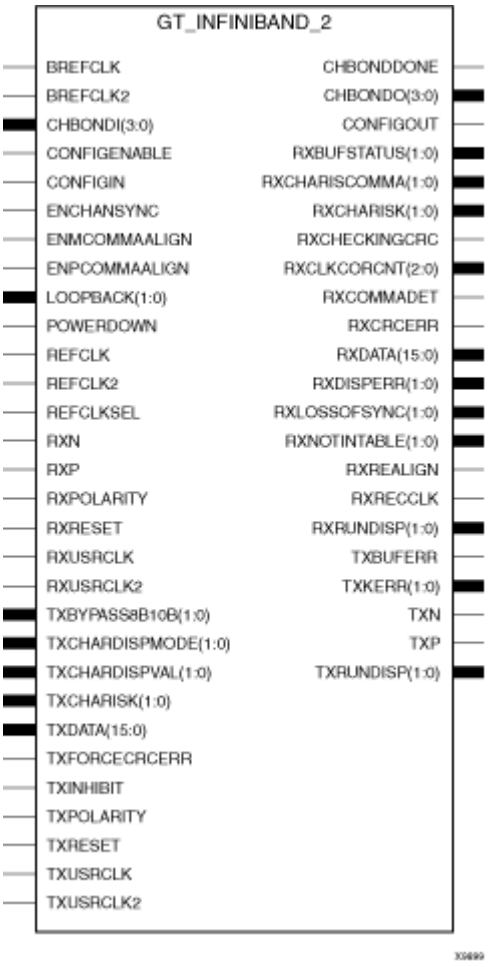
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_INFINIBAND_2

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

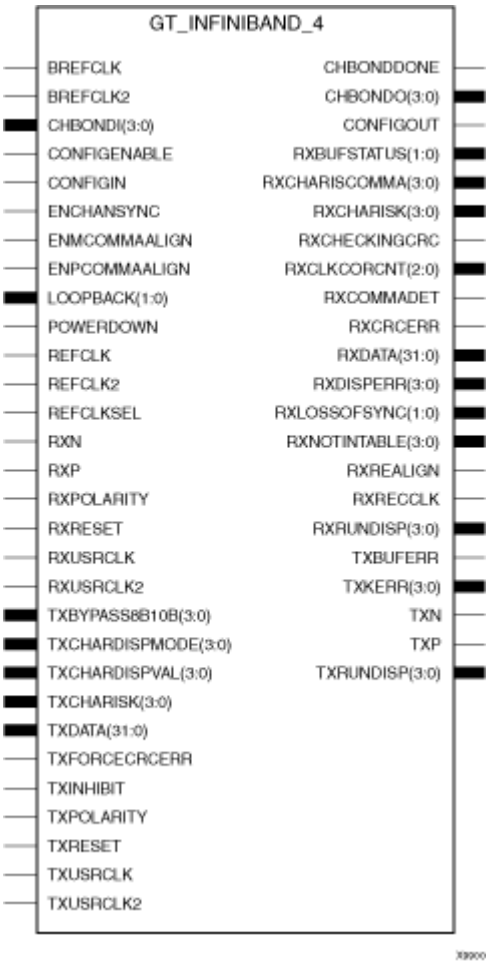
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_INFINIBAND_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

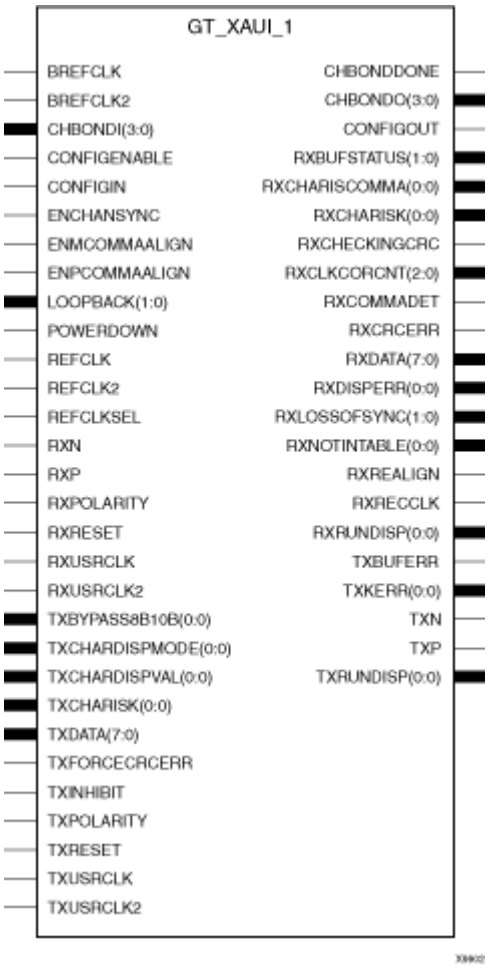
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_XAUI_1

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

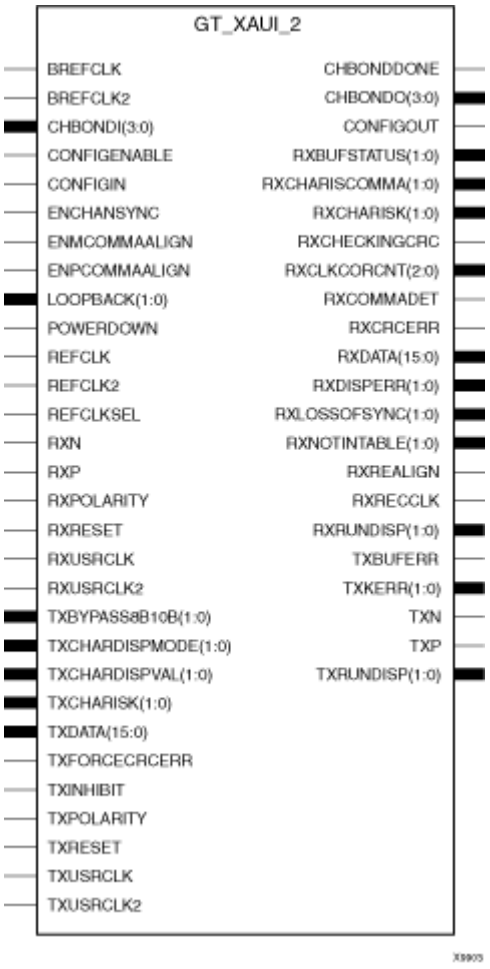
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_XAUI_2

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

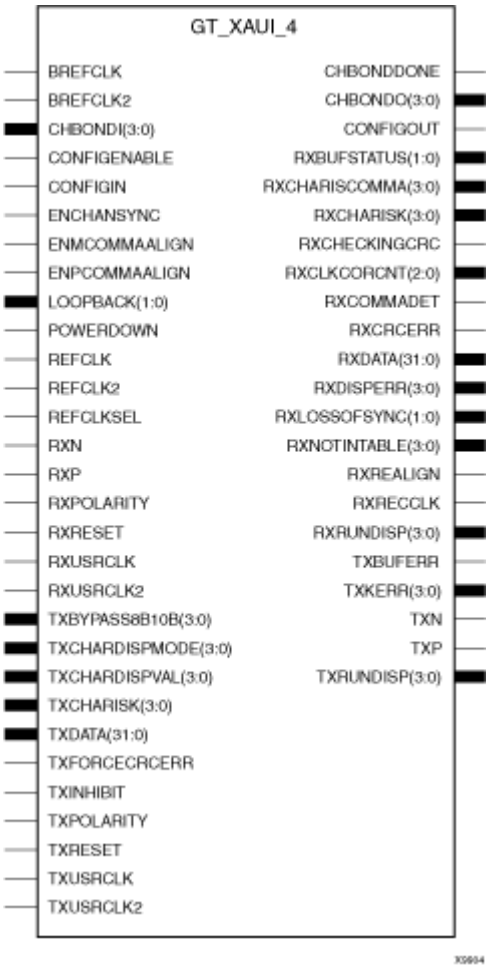
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT_XAUI_4

Primitive: Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

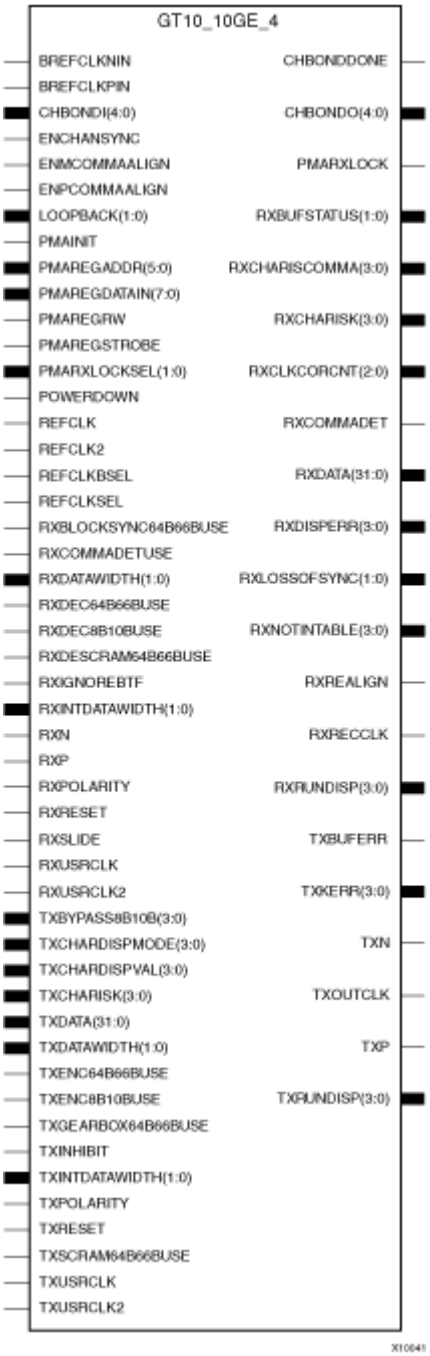
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_10GE_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports 4-byte data paths. You can also set attributes for the primitives.

Design Entry Method

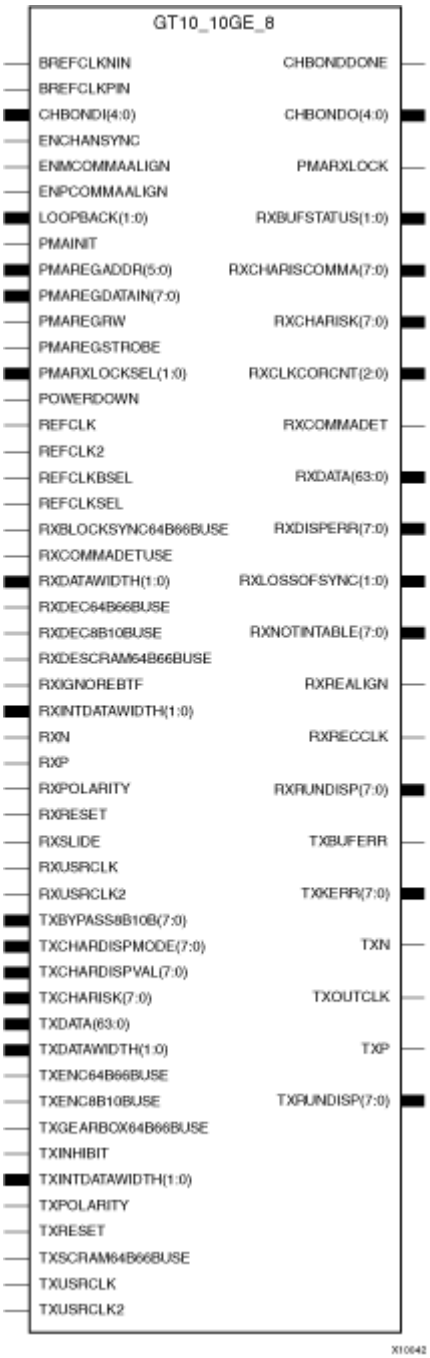
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_10GE_8

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver support an 8-byte data path. You can also set attributes for the primitives.

Design Entry Method

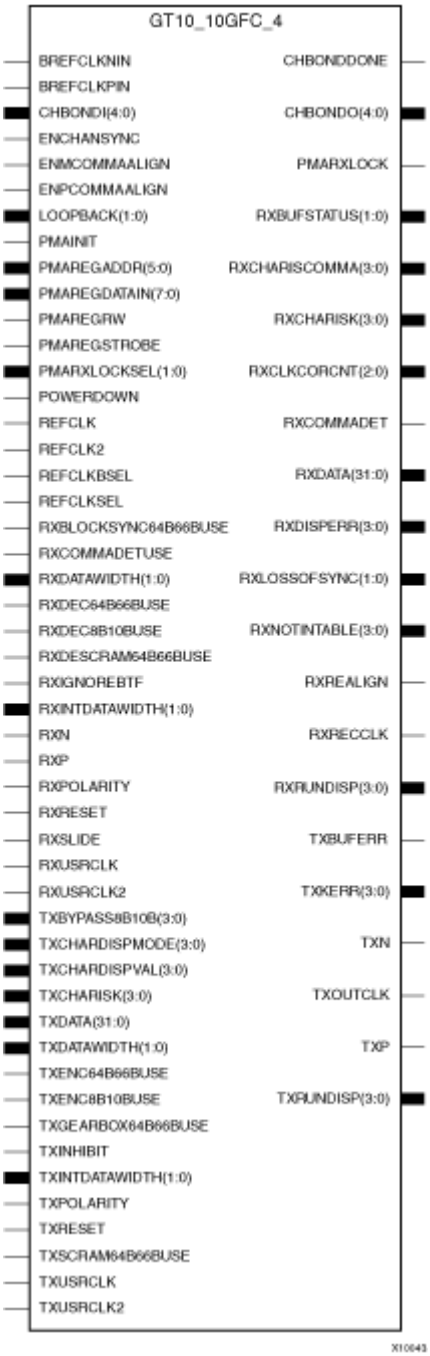
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_10GFC_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports 4-byte data paths. You can also set attributes for the primitives.

Design Entry Method

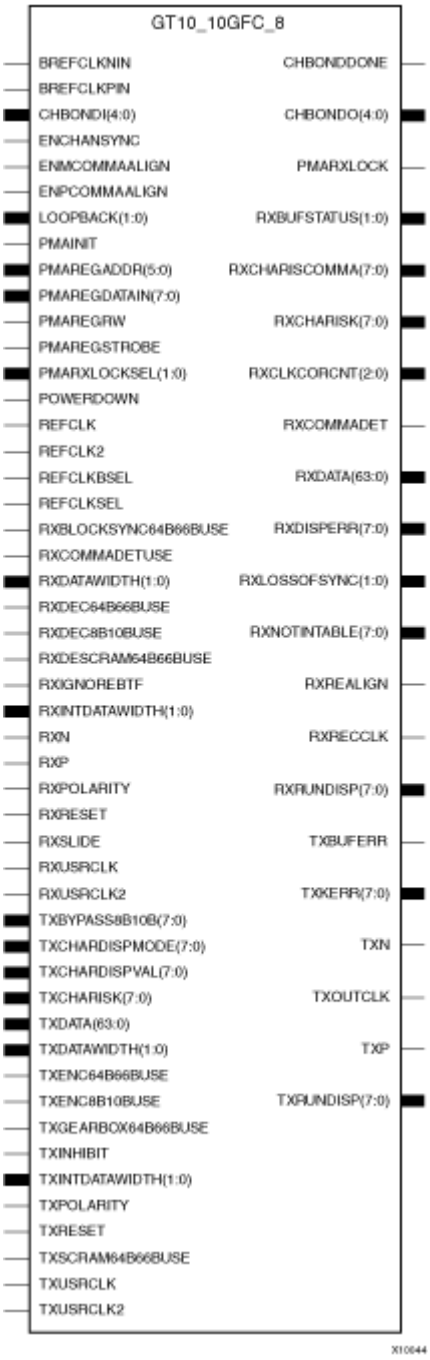
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_10GFC_8

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports an 8-byte data path. You can also set attributes for the primitives.

Design Entry Method

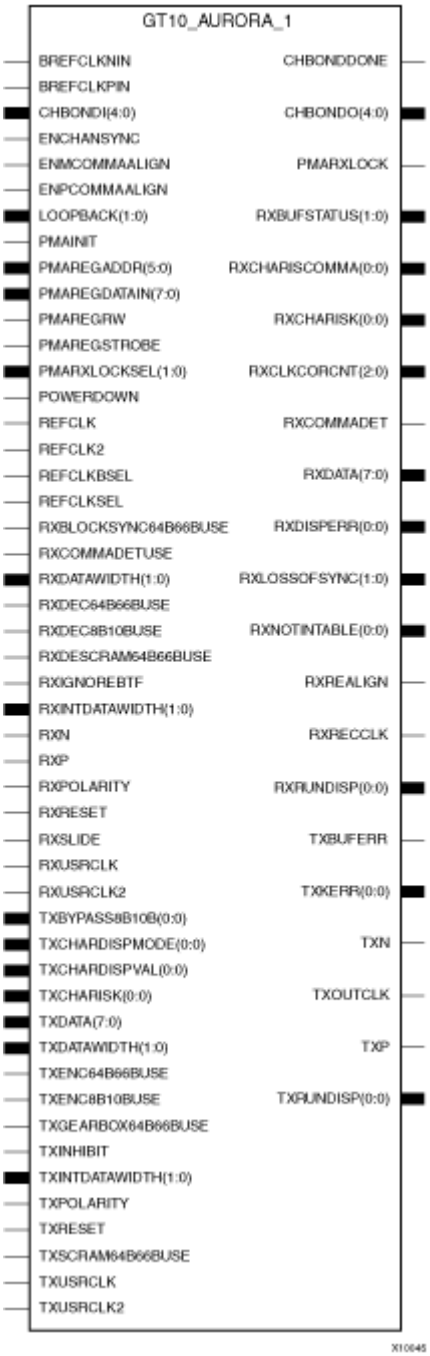
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_AURORA_1

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports 1-byte data path. You can also set attributes for the primitives.

Design Entry Method

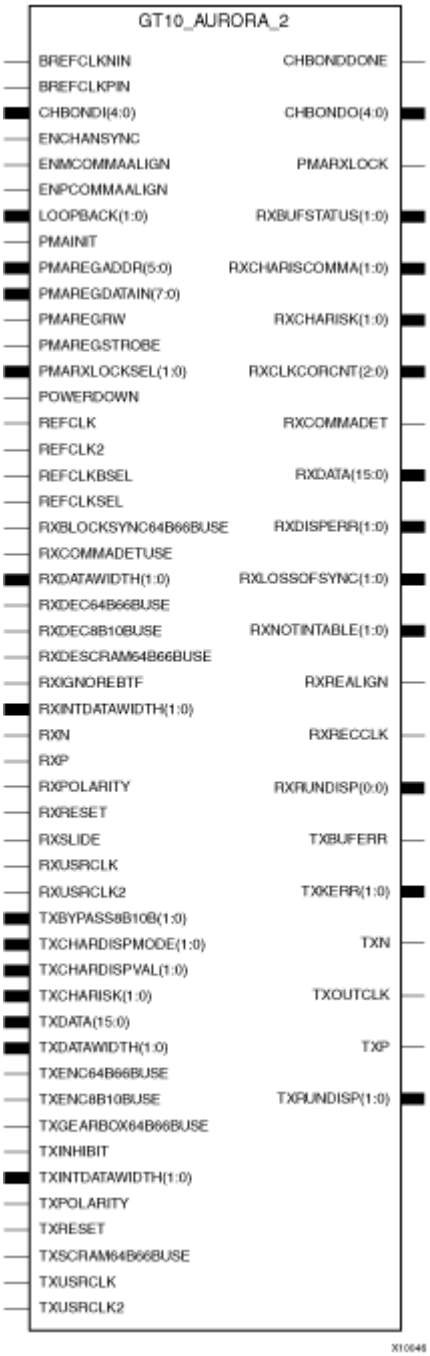
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_AURORA_2

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports a 2--byte data path. You can also set attributes for the primitives.

Design Entry Method

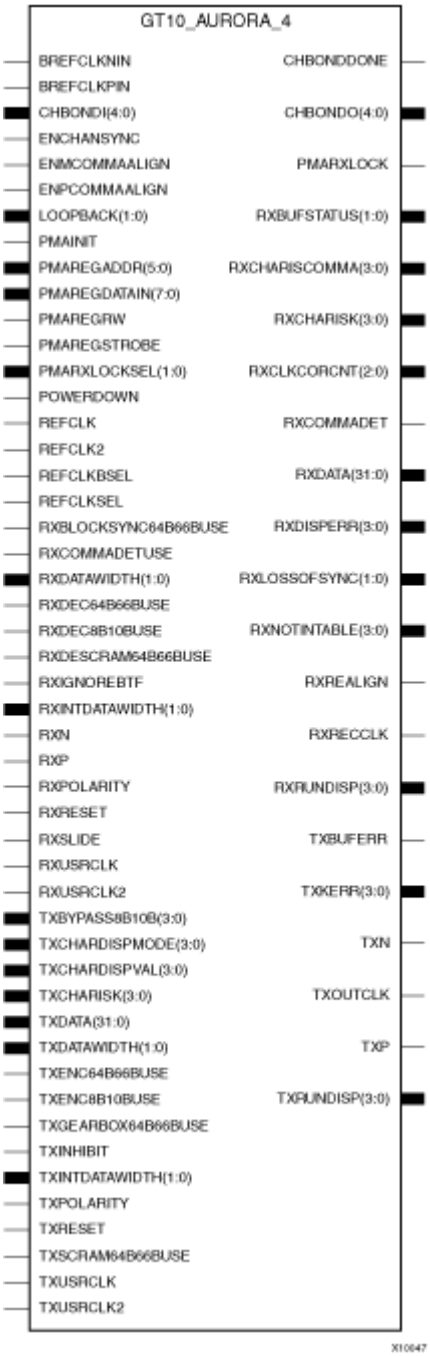
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_AURORA_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver support a 4-byte data path. You can also set attributes for the primitives.

Design Entry Method

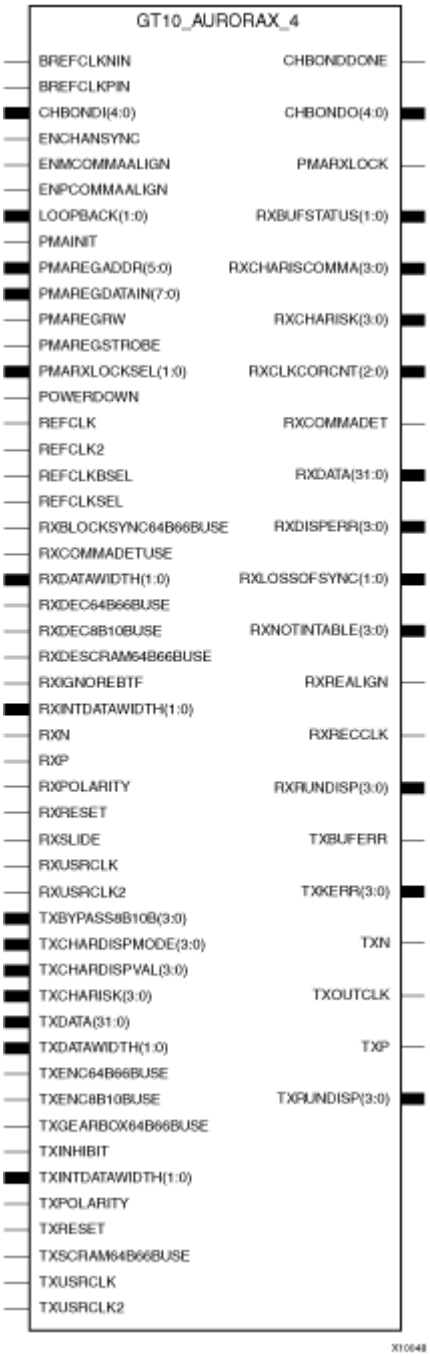
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_AURORAX_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports 4-byte data paths. You can also set attributes for the primitives.

Design Entry Method

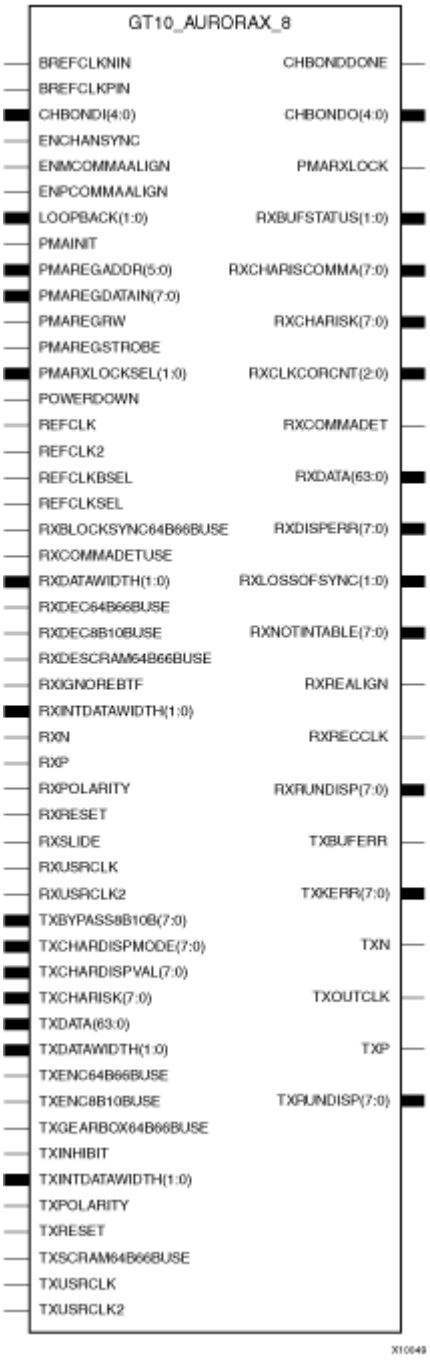
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_AURORAX_8

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports an 8-byte data path. You can also set attributes for the primitives.

Design Entry Method

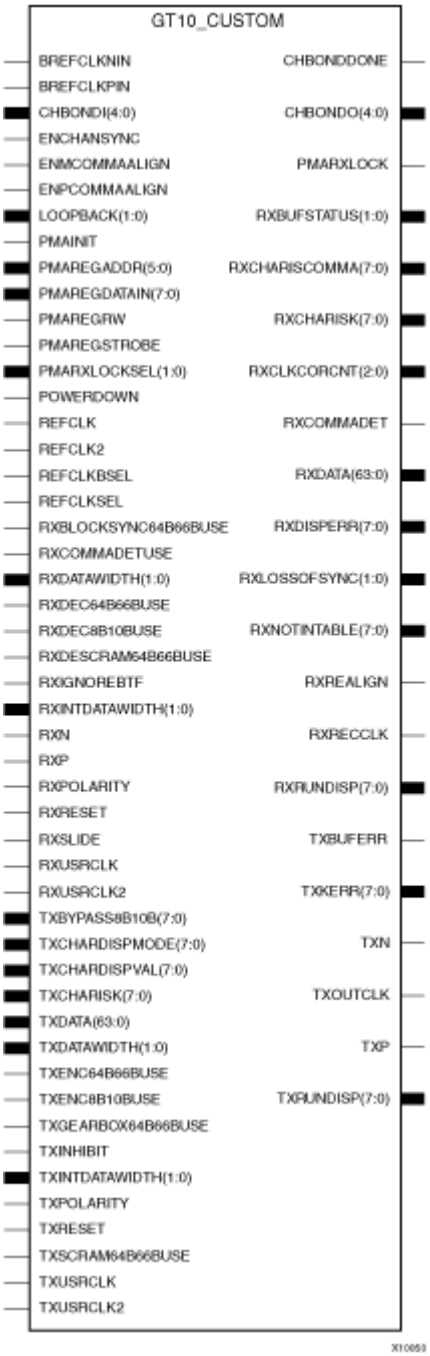
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_CUSTOM

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This 10-gigabit transceiver is fully customizable. You can also set attributes for the primitives.

Design Entry Method

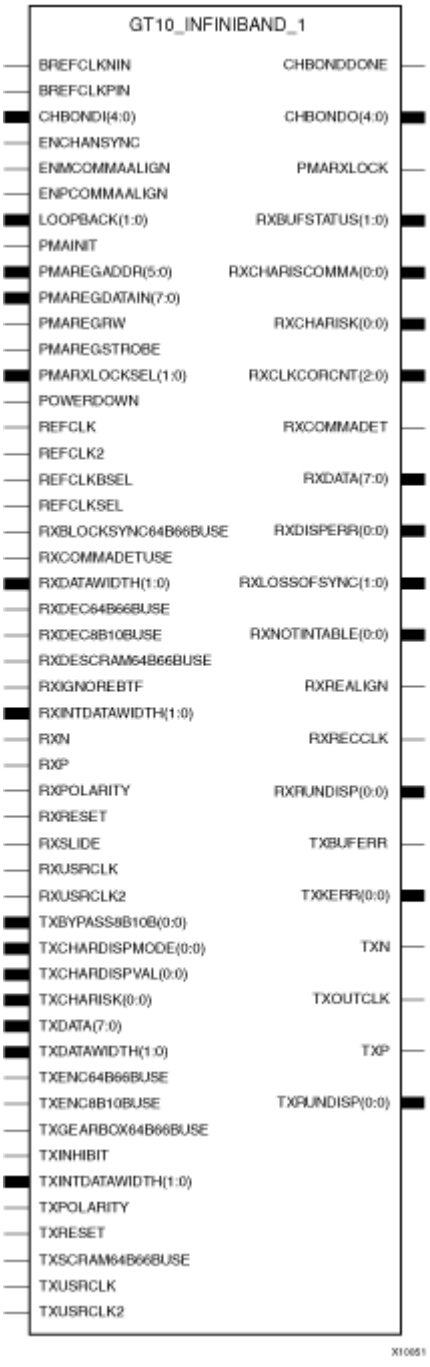
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_INFINIBAND_1

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

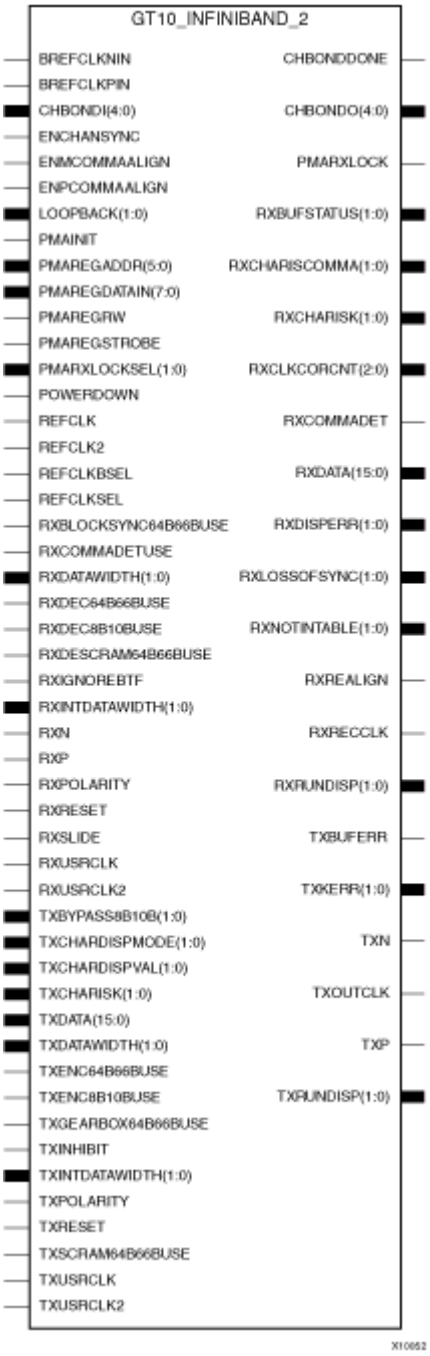
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_INFINIBAND_2

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

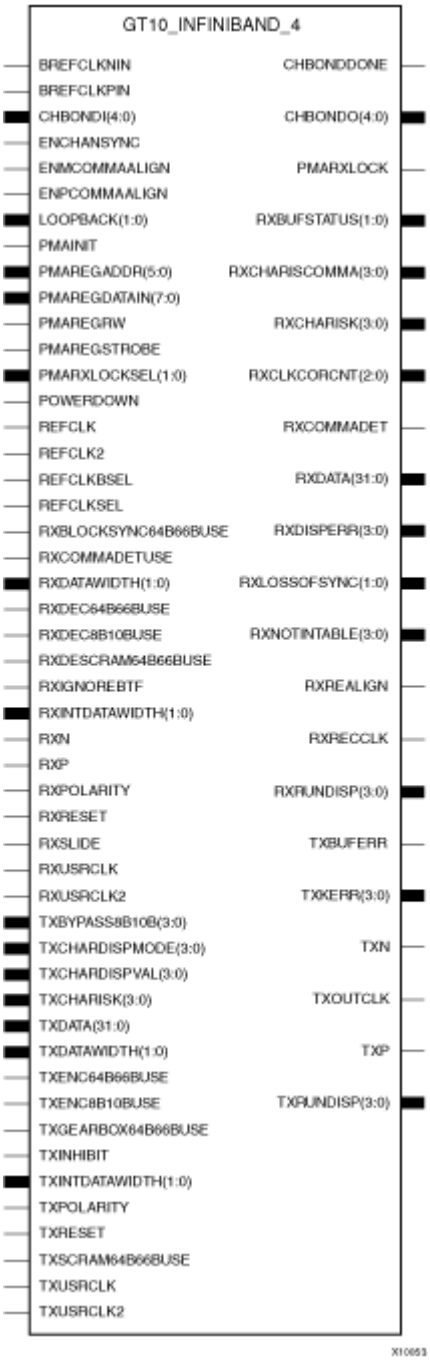
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_INFINIBAND_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

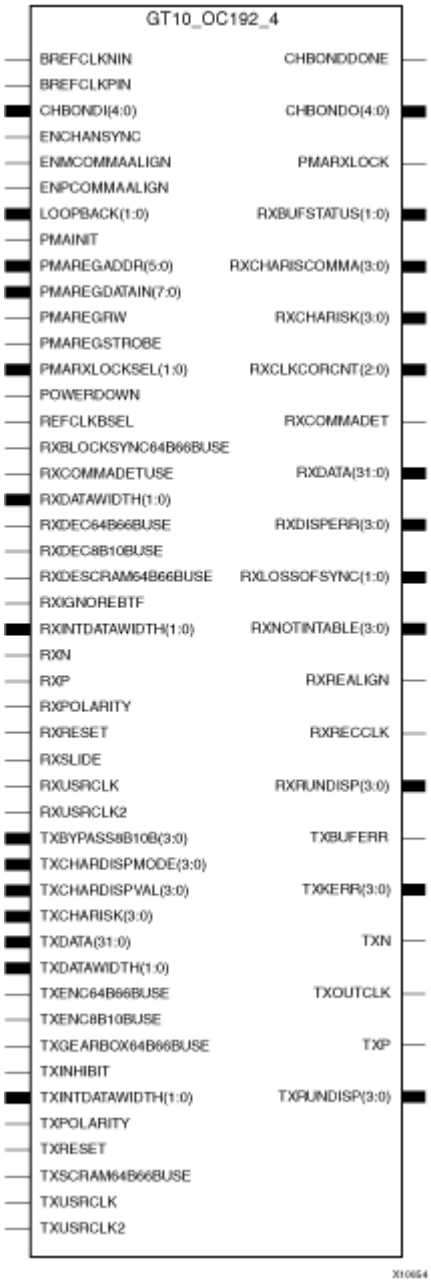
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_OC192_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports 4-byte data paths. You can also set attributes for the primitives.

Design Entry Method

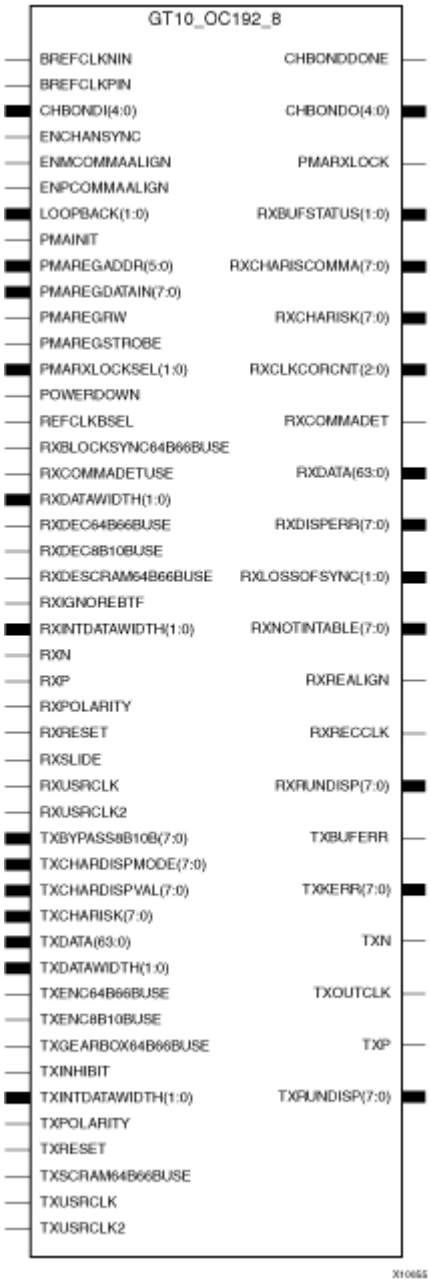
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_OC192_8

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports 8-byte data paths. You can also set attributes for the primitives.

Design Entry Method

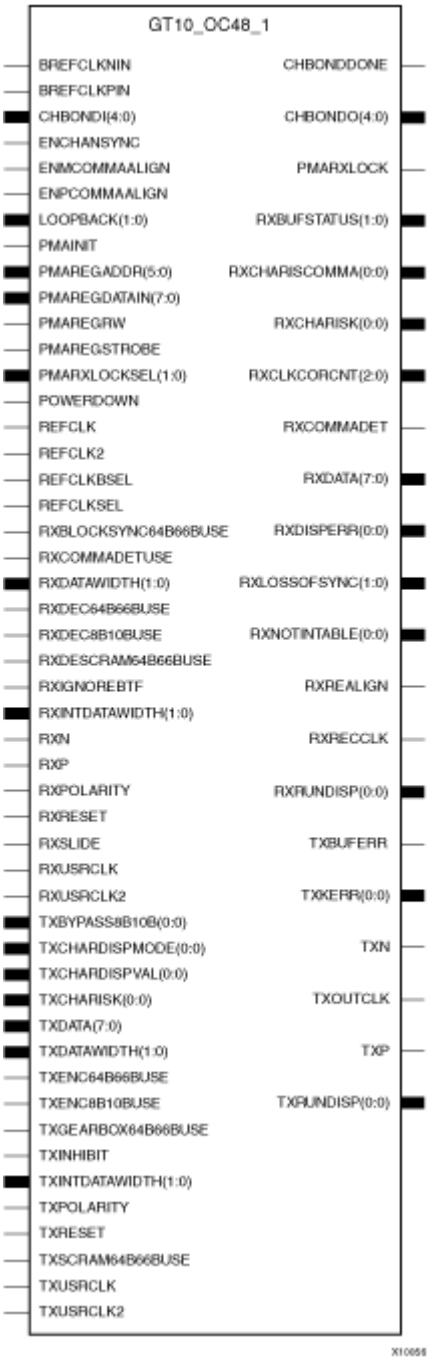
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_OC48_1

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

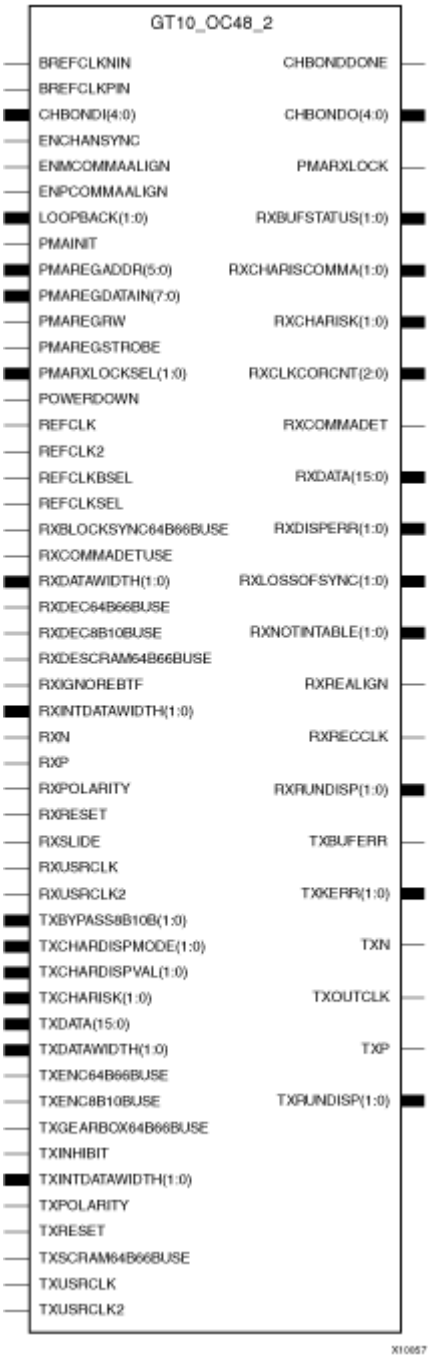
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_OC48_2

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

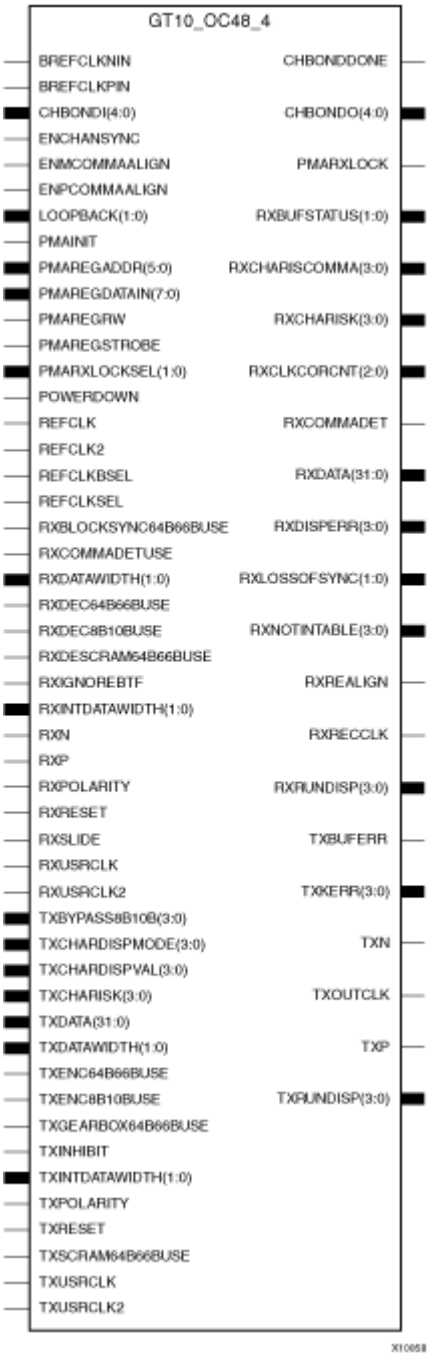
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_OC48_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This design element supports *n*-byte data paths as indicated in *design_element_name_n*. For more information, see the *RocketIO Transceiver User Guide*.

Design Entry Method

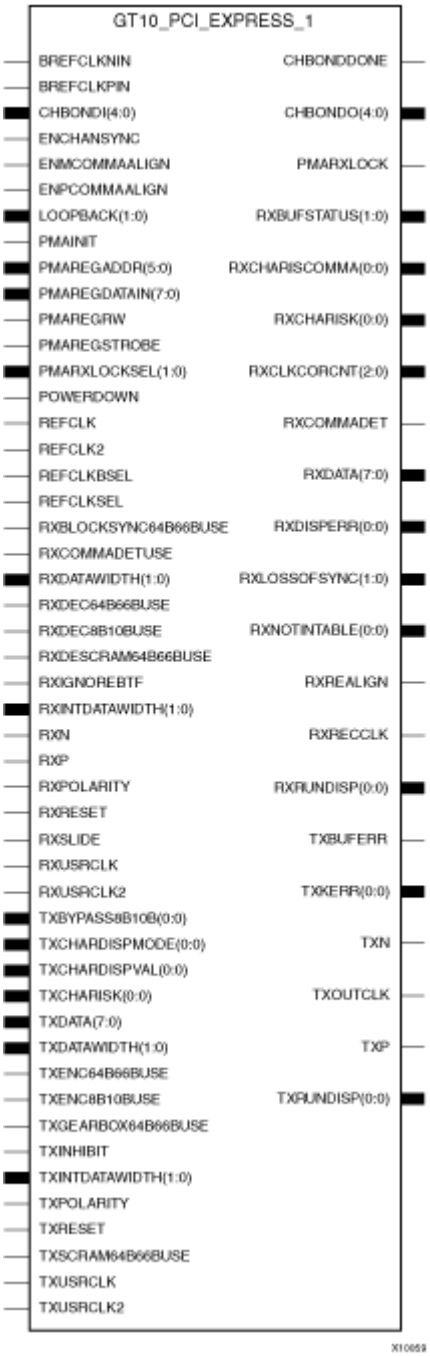
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_PCI_EXPRESS_1

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports a 1-byte data path. You can also set attributes for the primitives.

Design Entry Method

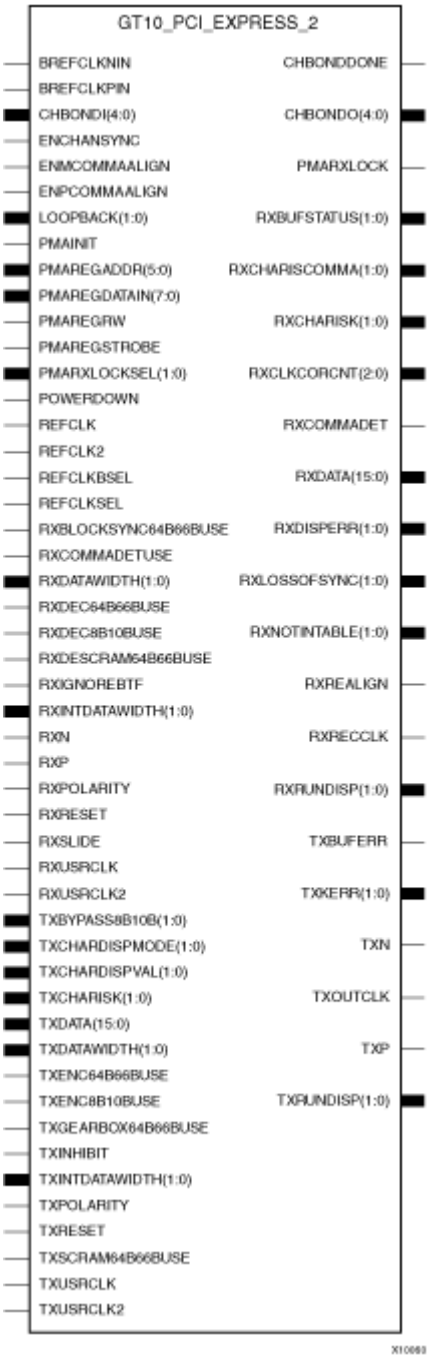
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_PCI_EXPRESS_2

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports a 2-byte data path. You can also set attributes for the primitives.

Design Entry Method

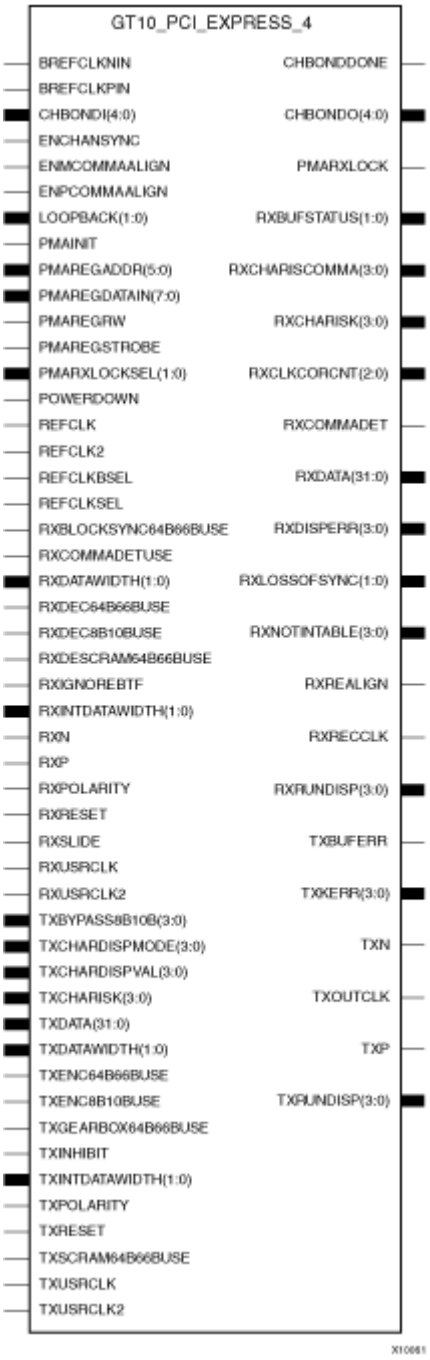
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_PCI_EXPRESS_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This Xilinx protocol 10-gigabit transceiver supports a 4-byte data path. You can also set attributes for the primitives.

Design Entry Method

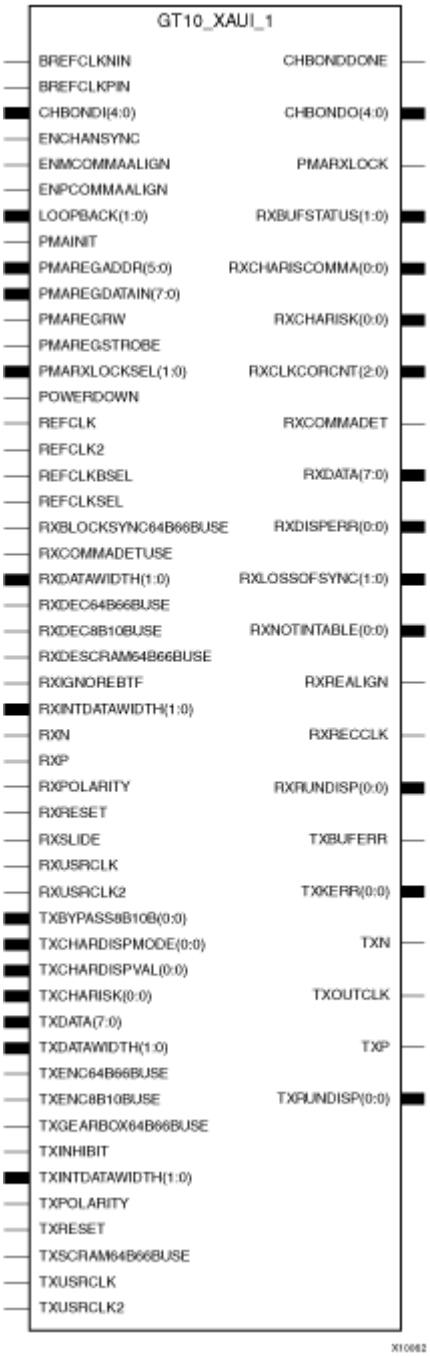
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_XAUI_1

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This XAUI 10-gigabit transceiver supports a 1-byte data path. You can also set attributes for the primitives.

Design Entry Method

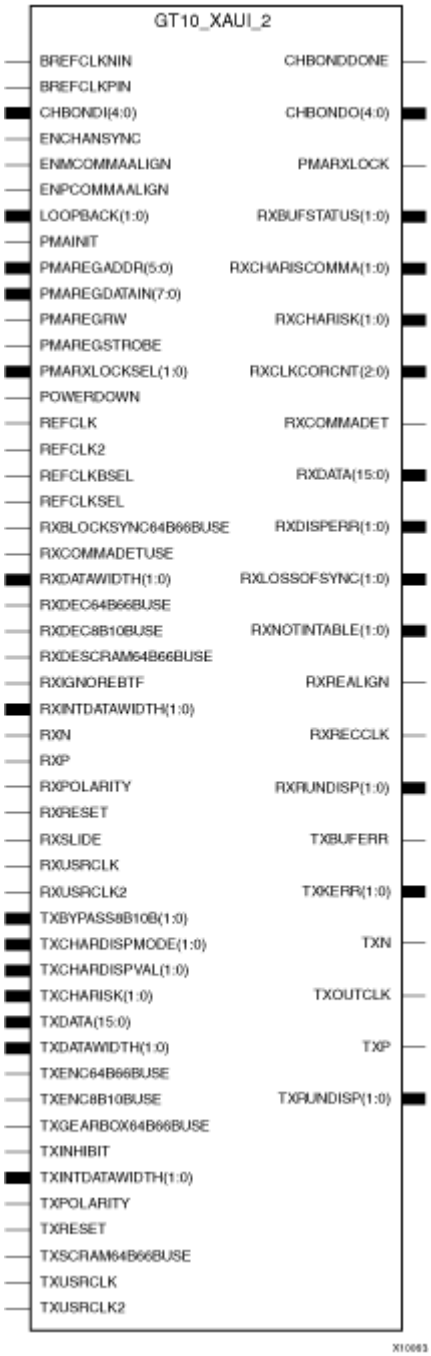
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_XAUI_2

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This XAUI 10-gigabit transceiver supports a 2-byte data path. You can also set attributes for the primitives.

Design Entry Method

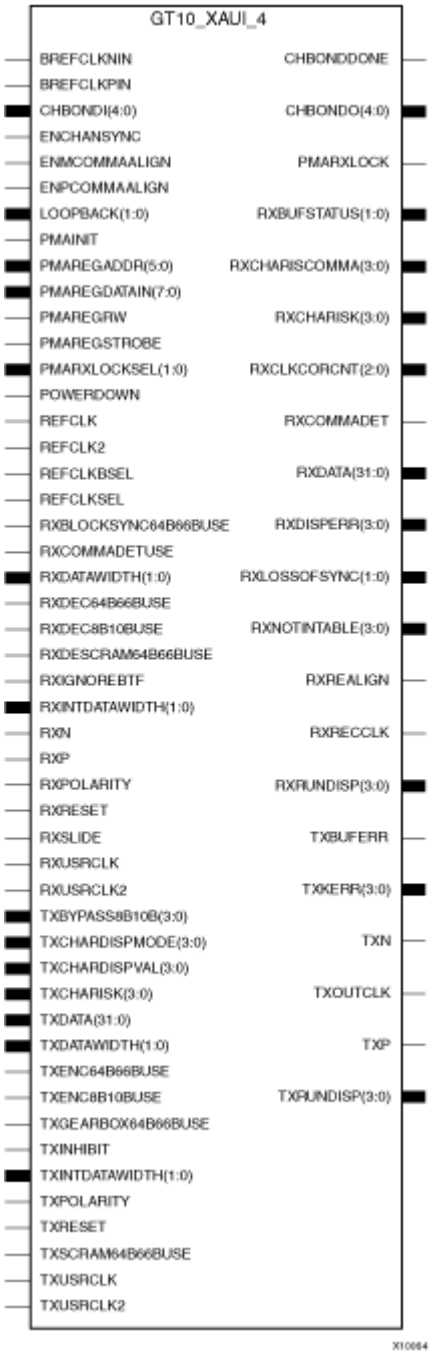
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

GT10_XAUI_4

Primitive: 10-Gigabit Transceiver for High-Speed I/O



Introduction

This XAUI 10-gigabit transceiver supports a 4-byte data path. You can also set attributes for the primitives.

Design Entry Method

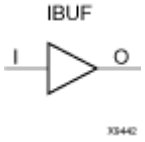
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUF

Primitive: Input Buffer



Introduction

This design element is automatically inserted (inferred) by the synthesis tool to any signal directly connected to a top-level input or in-out port of the design. You should generally let the synthesis tool infer this buffer. However, it can be instantiated into the design if required. In order to do so, connect the input port (I) directly to the associated top-level input or in-out port, and connect the output port (O) to the logic sourced by that port. Modify any necessary generic maps (VHDL) or named parameter value assignment (Verilog) in order to change the default behavior of the component.

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Buffer input
I	Input	1	Buffer output

Design Entry Method

This design element can be used in schematics.

In general, this element is inferred by the synthesis tool for any specified top-level input port to the design. It is generally not necessary to specify them in the source code however if desired, they be manually instantiated by either copying the instantiation code from the ISE Libraries Guide HDL Template and paste it into the top-level entity/module of your code. It is recommended to always put all I/O components on the top-level of the design to help facilitate hierarchical design methods. Connect the I port directly to the top-level input port of the design and the O port to the logic in which this input is to source. Specify the desired generic/defparam values in order to configure the proper behavior of the buffer.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUF16

Macro: 16-Bit Input Buffer

Introduction

Input Buffers isolate the internal circuit from the signals coming into the chip. This design element is contained in input/output blocks (IOBs) and allows the specification of the particular I/O Standard to configure the I/O. In general, an this element should be used for all single-ended data input or bidirectional pins.

Design Entry Method

This design element can be used in schematics.

In general, this element is inferred by the synthesis tool for any specified top-level input port to the design. It is generally not necessary to specify them in the source code however if desired, they be manually instantiated by either copying the instantiation code from the ISE Libraries Guide HDL Template and paste it into the top-level entity/module of your code. It is recommended to always put all I/O components on the top-level of the design to help facilitate hierarchical design methods. Connect the I port directly to the top-level input port of the design and the O port to the logic in which this input is to source. Specify the desired generic/defparam values in order to configure the proper behavior of the buffer.

Available Attributes

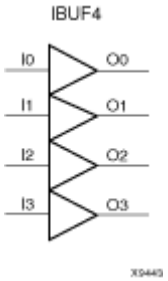
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUF4

Macro: 4-Bit Input Buffer



Introduction

Input Buffers isolate the internal circuit from the signals coming into the chip. This design element is contained in input/output blocks (IOBs) and allows the specification of the particular I/O Standard to configure the I/O. In general, an this element should be used for all single-ended data input or bidirectional pins.

Design Entry Method

This design element can be used in schematics.

In general, this element is inferred by the synthesis tool for any specified top-level input port to the design. It is generally not necessary to specify them in the source code however if desired, they be manually instantiated by either copying the instantiation code from the ISE Libraries Guide HDL Template and paste it into the top-level entity/module of your code. It is recommended to always put all I/O components on the top-level of the design to help facilitate hierarchical design methods. Connect the I port directly to the top-level input port of the design and the O port to the logic in which this input is to source. Specify the desired generic/defparam values in order to configure the proper behavior of the buffer.

Available Attributes

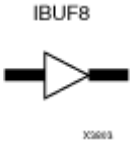
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUF8

Macro: 8-Bit Input Buffer



Introduction

Input Buffers isolate the internal circuit from the signals coming into the chip. This design element is contained in input/output blocks (IOBs) and allows the specification of the particular I/O Standard to configure the I/O. In general, an this element should be used for all single-ended data input or bidirectional pins.

Design Entry Method

This design element can be used in schematics.

In general, this element is inferred by the synthesis tool for any specified top-level input port to the design. It is generally not necessary to specify them in the source code however if desired, they be manually instantiated by either copying the instantiation code from the ISE Libraries Guide HDL Template and paste it into the top-level entity/module of your code. It is recommended to always put all I/O components on the top-level of the design to help facilitate hierarchical design methods. Connect the I port directly to the top-level input port of the design and the O port to the logic in which this input is to source. Specify the desired generic/defparam values in order to configure the proper behavior of the buffer.

Available Attributes

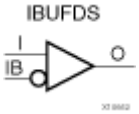
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUFDS

Primitive: Differential Signaling Input Buffer with Optional Delay



Introduction

This design element is an input buffer that supports low-voltage, differential signaling. In IBUFDS, a design level interface signal is represented as two distinct ports (I and IB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET_P and MYNET_N). Optionally, a programmable differential termination feature is available to help improve signal integrity and reduce external components. Also available is a programmable delay to assist in the capturing of incoming data to the device.

Logic Table

Inputs		Outputs
I	IB	O
0	0	No Change
0	1	0
1	0	1
1	1	No Change

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Diff_p Buffer Input
IB	Input	1	Diff_n Buffer Input
I	Input	1	Buffer Output

Design Entry Method

This design element can be used in schematics.

Put all I/O components on the top-level of the design to help facilitate hierarchical design methods. Connect the I port directly to the top-level "master" input port of the design, the IB port to the top-level "slave" input port, and the O port to the logic in which this input is to source. Specify the desired generic/defparam values in order to configure the proper behavior of the buffer.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUFDS_DIFF_OUT

Primitive: Differential I/O Input Buffer with Differential Outputs

Introduction

This design element is a differential I/O input buffer with differential outputs. The differential output pair (O&OB) maintains the relation of its differential input pair

Design Entry Method

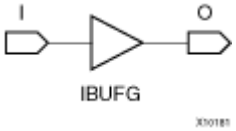
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUFG

Primitive: Dedicated Input Clock Buffer



Introduction

The IBUFG is a dedicated input to the device which should be used to connect incoming clocks to the FPGA to the global clock routing resources. The IBUFG provides dedicated connections to the DCM_SP and BUFG providing the minimum amount of clock delay and jitter to the device. The IBUFG input can only be driven by the global clock pins. The IBUFG output can drive CLKIN of a DCM_SP, BUFG, or your choice of logic. The IBUFG can be routed to your choice of logic to allow the use of the dedicated clock pins for general logic.

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Clock Buffer input
I	Input	1	Clock Buffer output

Design Entry Method

This design element can be used in schematics.

Available Attributes

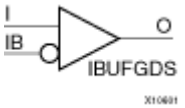
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IBUFGDS

Primitive: Differential Signaling Dedicated Input Clock Buffer and Optional Delay



Introduction

This design element is a dedicated differential signaling input buffer for connection to the clock buffer (BUFG) or DCM. In IBUFGDS, a design-level interface signal is represented as two distinct ports (I and IB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET_P and MYNET_N). Optionally, a programmable differential termination feature is available to help improve signal integrity and reduce external components. Also available is a programmable delay is to assist in the capturing of incoming data to the device.

Logic Table

Inputs		Outputs
I	**IB**	**O**
0	0	No Change
0	1	0
1	0	1
1	1	No Change

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Diff_p Clock Buffer Input
IB	Input	1	Diff_n Clock Buffer Input
I	Input	1	Clock Buffer output

Design Entry Method

This design element can be used in schematics.

Put all I/O components on the top-level of the design to help facilitate hierarchical design methods. Connect the I port directly to the top-level "master" input port of the design, the IB port to the top-level "slave" input port and the O port to a DCM, BUFG or logic in which this input is to source. Some synthesis tools infer the BUFG automatically if necessary, when connecting an IBUFG to the clock resources of the FPGA. Specify the desired generic/defparam values in order to configure the proper behavior of the buffer.

Available Attributes

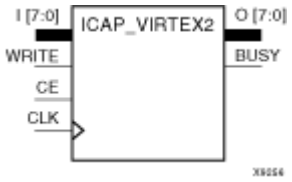
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ICAP_VIRTEX2

Primitive: User Interface to Virtex-II Internal Configuration Access Port



Introduction

This design element provides user access to the Virtex-II, Virtex-II Pro, and Virtex-II Pro X internal configuration access port (ICAP).

Design Entry Method

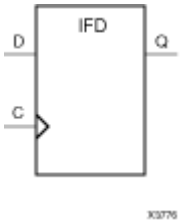
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFD

Macro: Input D Flip-Flop



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
Dn	↑	Dn

Design Entry Method

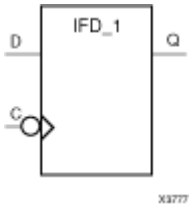
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFD_1

Macro: Input D Flip-Flop with Inverted Clock (Asynchronous Preset)



Introduction

This design element is a D-type flip flop which is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The D input also provides data input for the flip-flop, which synchronizes data entering the chip. The D input data is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
0	↓	0
1	↓	1

Design Entry Method

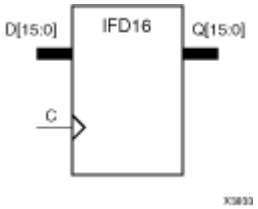
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFD16

Macro: 16-Bit Input D Flip-Flop



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
Dn	↑	Dn

Design Entry Method

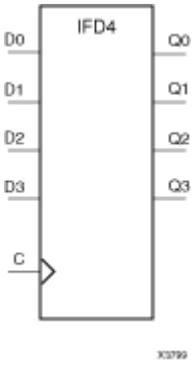
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFD4

Macro: 4-Bit Input D Flip-Flop



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
Dn	↑	Dn

Design Entry Method

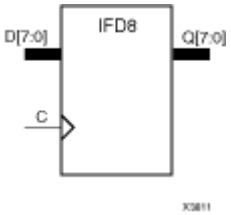
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFD8

Macro: 8-Bit Input D Flip-Flop



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
Dn	↑	Dn

Design Entry Method

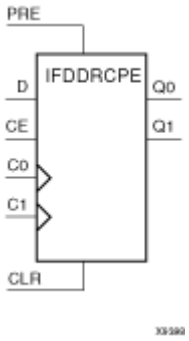
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDDRCPE

Primitive: Dual Data Rate Input D Flip-Flop with Clock Enable and Asynchronous Preset and Clear



Introduction

This design element is a dual data rate (DDR) input D flip-flop with clock enable (CE) and asynchronous preset (PRE) and clear (CLR). It consists of one input buffer and two identical flip-flops (FDCPE).

When the asynchronous PRE is High and CLR is Low, both the Q0 and Q1 outputs are set High. When CLR is High, both outputs are reset Low. When PRE and CLR are Low and CE is High, data on the D input is loaded into the Q0 output on the Low-to High C0 clock transition, and into the Q1 output on the Low-to-High C1 clock transition.

The INIT attribute does not apply to this design elements components.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs	
C0	**C1**	**CE**	**D**	**CLR**	**PRE**	**Q0**	**Q1**
X	X	X	X	1	0	0	0
X	X	X	X	0	1	1	1
X	X	X	X	1	1	0	0
X	X	0	X	0	0	No Change	No Change
↑	X	1	D	0	0	D	No Change
X	↑	1	D	0	0	No Change	D

Design Entry Method

This design element can be used in schematics.

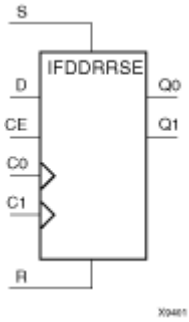
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

IFDDRRSE

Primitive: Dual Data Rate Input D Flip-Flop with Synchronous Reset and Set and Clock Enable



Introduction

This design element is a dual data rate (DDR) input D flip-flop with synchronous reset (R), synchronous set (S), and clock enable (CE). It consists of one input buffer and two identical flip-flops (FDRSE).

For the C0 input and Q0 output, reset (R) has precedence. The R input, when High, resets the Q0 output Low during the Low-to-High C0 clock transition. When S is High and R is Low, the Q0 output is set High during the Low-to-High C0 clock transition. For the C1 input and Q1 output, set (S) has precedence. The R input, when High, resets the Q1 output Low during the Low-to-High C1 clock transition. When S is High and R is Low, the Q0 output is set to High during the Low-to-High C1 clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

The INIT attribute does not apply to this element.

Logic Table

Inputs						Outputs	
C0	**C1**	**CE**	**D**	**R**	**S**	**Q0**	**Q1**
↑	X	X	X	1	0	0	No Change
↑	X	X	X	0	1	1	No Change
↑	X	X	X	1	1	0	No Change
X	↑	X	X	1	0	No Change	0
X	↑	X	X	0	1	No Change	1
X	↑	X	X	1	1	No Change	0
X	X	0	X	0	0	No Change	No Change
↑	X	1	D	0	0	D	No Change
X	↑	1	D	0	0	No Change	D

Design Entry Method

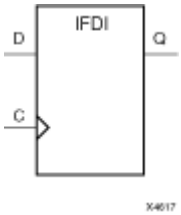
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDI

Macro: Input D Flip-Flop (Asynchronous Preset)



Introduction

This design element is a D-type flip-flop which is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The D input provides data input for the flip-flop, which synchronizes data entering the chip. The D input data is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↑	D

Design Entry Method

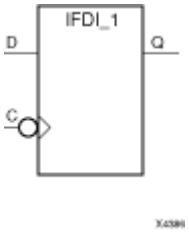
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDI_1

Macro: Input D Flip-Flop with Inverted Clock (Asynchronous Preset)



Introduction

The design element is a D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
0	↓	D

Design Entry Method

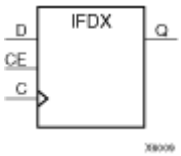
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDX

Macro: Input D Flip-Flop with Clock Enable



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The D input provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the Low-to- High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When CE is Low, flip-flop outputs do not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**Dn**	**C**	**Qn**
1	Dn	↑	Dn
0	X	X	No Change

Design Entry Method

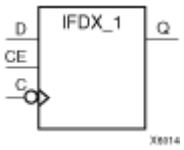
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDX_1

Macro: Input D Flip-Flop with Inverted Clock and Clock Enable



Introduction

This design element is a D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The D input also provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When the CE pin is Low, the output (Q) does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	D	C	Q
1	D	↓	D
0	X	X	No Change

Design Entry Method

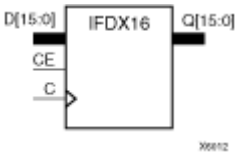
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDX16

Macro: 16-Bit Input D Flip-Flops with Clock Enable



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The D input provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the Low-to- High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When CE is Low, flip-flop outputs do not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**Dn**	**C**	**Qn**
1	Dn	↑	Dn
0	X	X	No Change

Design Entry Method

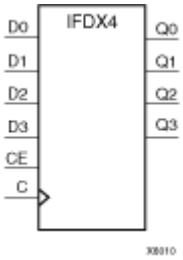
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDX4

Macro: 4-Bit Input D Flip-Flop with Clock Enable



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The D input provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the Low-to- High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When CE is Low, flip-flop outputs do not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**Dn**	**C**	**Qn**
1	Dn	↑	Dn
0	X	X	No Change

Design Entry Method

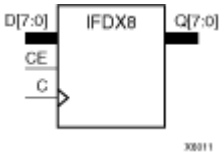
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDX8

Macro: 8-Bit Input D Flip-Flop with Clock Enable



Introduction

This D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The D input provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the Low-to- High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When CE is Low, flip-flop outputs do not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**Dn**	**C**	**Qn**
1	Dn	↑	Dn
0	X	X	No Change

Design Entry Method

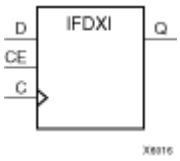
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDXI

Macro: Input D Flip-Flop with Clock Enable (Asynchronous Preset)



Introduction

The design element is a D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The D input provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When the CE pin is Low, the output (Q) does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CE	**D**	**C**	**Q**
1	D	↑	D
0	X	X	No Change

Design Entry Method

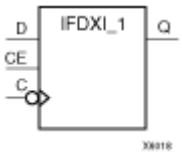
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IFDXI_1

Macro: Input D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)



Introduction

The design element is a D-type flip-flop that is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. When (CE) is High, the data on input (D) is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When the (CE) pin is Low, the output (Q) does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**D**	**C**	**Q**
1	D	↓	D
0	X	X	No Change

Design Entry Method

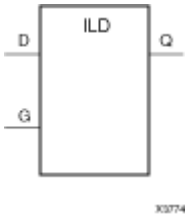
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILD

Macro: Transparent Input Data Latch



Introduction

This design element is a single, transparent data latch that holds transient data entering a chip. This latch is contained in an input/output block (IOB). The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF). When the gate input (G) is High, data on the input (D) appears on the output (Q). Data on the D input during the High-to-Low G transition is stored in the latch.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Output
G	**D**	**Q**
1	D	D
0	X	No Change
Ø	D	D

Design Entry Method

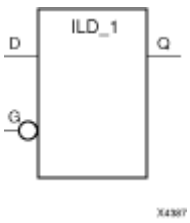
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILD_1

Macro: Transparent Input Data Latch with Inverted Gate



Introduction

This design element is a transparent data latch that holds transient data entering a chip. When the gate input (G) is Low, data on the data input (D) appears on the data output (Q). Data on (D) during the Low-to-High (G) transition is stored in the latch.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
0	D	D
1	X	D
↑	D	D

Design Entry Method

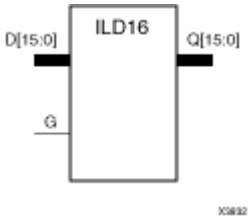
This component is inside the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILD_1, you would infer an ILD_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option **`-pri 0`** to pack all input registers into the IOBs.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILD16

Macro: Transparent Input Data Latch



Introduction

These design elements are multiple transparent data latches that hold transient data entering a chip. The ILD latch is contained in an input/output block (IOB). The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF). When the gate input (G) is High, data on the inputs (D) appears on the outputs (Q). Data on the D inputs during the High-to-Low G transition is stored in the latch.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	D	D

Design Entry Method

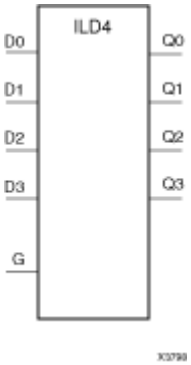
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILD4

Macro: Transparent Input Data Latch



Introduction

These design elements are multiple transparent data latches that hold transient data entering a chip. The ILD latch is contained in an input/output block (IOB). The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF). When the gate input (G) is High, data on the inputs (D) appears on the outputs (Q). Data on the D inputs during the High-to-Low G transition is stored in the latch.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	D	D

Design Entry Method

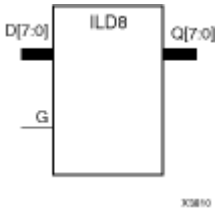
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILD8

Macro: Transparent Input Data Latch



Introduction

These design elements are multiple transparent data latches that hold transient data entering a chip. The ILD latch is contained in an input/output block (IOB). The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF). When the gate input (G) is High, data on the inputs (D) appears on the outputs (Q). Data on the D inputs during the High-to-Low G transition is stored in the latch.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	D	D

Design Entry Method

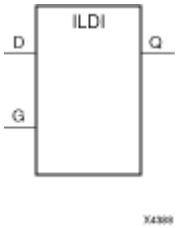
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDI

Macro: Transparent Input Data Latch (Asynchronous Preset)



Introduction

This design element is a transparent data latch that holds transient data entering a chip. When the gate input (G) is High, data on the input (D) appears on the output (Q). Data on the D input during the High-to-Low G transition is stored in the latch.

The ILDI is the input flip-flop master latch. It is possible to access two different outputs from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDI) corresponds to a falling edge-triggered flip-flop (IFDI_1). Similarly, a transparent Low latch (ILDI_1) corresponds to a rising edge-triggered flip-flop (IFDI).

The latch is asynchronously preset, output High, when power is applied.

For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	D
↓	D	D

Design Entry Method

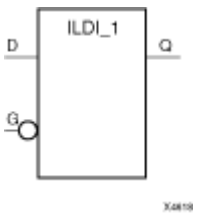
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDI_1

Macro: Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)



Introduction

This design element is a transparent data latch that holds transient data entering a chip. When the gate input (G) is Low, data on the data input (D) appears on the data output (Q). Data on D during the Low-to-High G transition is stored in the latch.

The latch is asynchronously preset, output High, when power is applied.

For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
0	1	1
0	0	0
1	X	D
↑	D	D

Design Entry Method

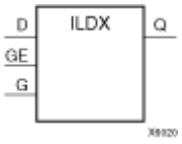
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDX

Macro: Transparent Input Data Latch



Introduction

This design element is single or multiple transparent data latches that holds transient data entering a chip. The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF).

The ILDX is the input flip-flop master latch. Two outputs can be accessed from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDX) corresponds to a falling edge-triggered flip-flop (IFDX_1). Similarly, a transparent Low latch (ILDX_1) corresponds to a rising edge-triggered flip-flop (IFDX)

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	0	X	No Change
1	1	D	D

Design Entry Method

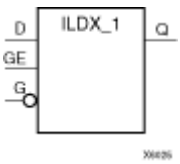
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDX_1

Macro: Transparent Input Data Latch with Inverted Gate



Introduction

This design element is a transparent data latch that holds transient data entering a chip. When the gate input (G) is Low, data on the data input (D) appears on the data output (Q). Data on D during the Low-to-High G transition is stored in the latch.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	1	X	No Change
1	0	1	1
1	0	0	0
1	↑	D	D

Design Entry Method

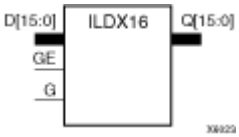
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDX16

Macro: Transparent Input Data Latch



Introduction

This design element is single or multiple transparent data latches that holds transient data entering a chip. The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF).

The ILDX is the input flip-flop master latch. Two outputs can be accessed from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDX) corresponds to a falling edge-triggered flip-flop (IFDX_1). Similarly, a transparent Low latch (ILDX_1) corresponds to a rising edge-triggered flip-flop (IFDX)

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	0	X	No Change
1	1	D	D

Design Entry Method

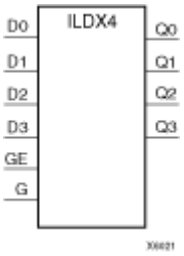
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDX4

Macro: Transparent Input Data Latch



Introduction

This design element is single or multiple transparent data latches that holds transient data entering a chip. The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF).

The ILDX is the input flip-flop master latch. Two outputs can be accessed from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDX) corresponds to a falling edge-triggered flip-flop (IFDX_1). Similarly, a transparent Low latch (ILDX_1) corresponds to a rising edge-triggered flip-flop (IFDX)

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	0	X	No Change
1	1	D	D

Design Entry Method

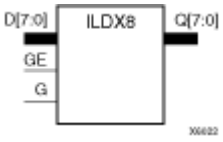
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDX8

Macro: Transparent Input Data Latch



Introduction

This design element is single or multiple transparent data latches that holds transient data entering a chip. The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF).

The ILDX is the input flip-flop master latch. Two outputs can be accessed from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDX) corresponds to a falling edge-triggered flip-flop (IFDX_1). Similarly, a transparent Low latch (ILDX_1) corresponds to a rising edge-triggered flip-flop (IFDX)

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	0	X	No Change
1	1	D	D

Design Entry Method

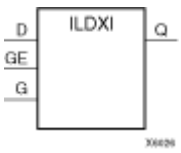
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDXI

Macro: Transparent Input Data Latch (Asynchronous Preset)



Introduction

This design element is a transparent data latch that holds transient data entering a chip. When the gate input (G) is High, data on the input (D) appears on the output (Q). Data on the (D) input during the High-to-Low (G) transition is stored in the latch.

The ILDXI is the input flip-flop master latch. Two outputs can be accessed from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDXI) corresponds to a falling edge-triggered flip-flop (IFDXI_1). Similarly, a transparent Low latch (ILDXI_1) corresponds to a rising edge-triggered flip-flop (IFDXI).

The latch is asynchronously preset, output High, when power is applied.

For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	0	X	No Change
1	1	D	D
1	↓	D	D

Design Entry Method

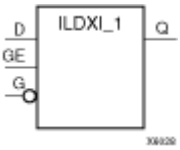
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ILDXI_1

Macro: Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)



Introduction

This design element is a transparent data latch that holds transient data entering a chip.

The latch is asynchronously preset, output High, when power is applied.

For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	1	X	No Change
1	0	D	D
1	↑	D	D

Design Entry Method

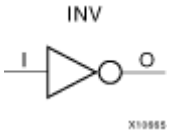
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

INV

Primitive: Inverter



Introduction

This design element is a single inverter that identifies signal inversions in a schematic.

Design Entry Method

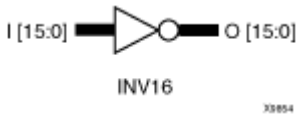
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

INV16

Macro: 16 Inverters



Introduction

This design element is a multiple inverter that identifies signal inversions in a schematic.

Design Entry Method

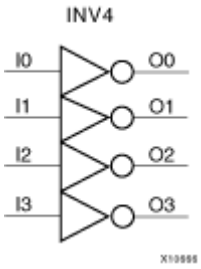
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

INV4

Macro: Four Inverters



Introduction

This design element is a multiple inverter that identifies signal inversions in a schematic.

Design Entry Method

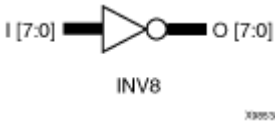
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

INV8

Macro: Eight Inverters



Introduction

This design element is a multiple inverter that identifies signal inversions in a schematic.

Design Entry Method

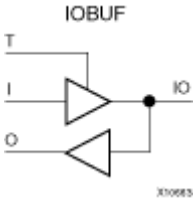
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IOBUF

Primitive: Bi-Directional Buffer



Introduction

The design element is a bidirectional single-ended I/O Buffer used to connect internal logic to an external bidirectional pin.

Logic Table

Inputs		Bidirectional	Outputs
T	**I**	**IO**	**O**
1	X	Z	X
0	1	1	1
0	0	0	0

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Buffer output
IO	Inout	1	Buffer inout
I	Input	1	Buffer input
T	Input	1	3-State enable input

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
DRIVE	Integer	2, 4, 6, 8, 12, 16, 24	12	Selects output drive strength (mA) for the SelectIO buffers that use the LVTTL, LVCMOS12, LVCMOS15, LVCMOS18, LVCMOS25, or LVCMOS33 interface I/O standard.

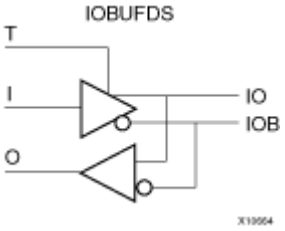
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.
SLEW	String	"SLOW", "FAST"	"SLOW"	Sets the output rise and fall time. See the Data Sheet for recommendations of the best setting for this attribute.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

IOBUFDS

Primitive: 3-State Differential Signaling I/O Buffer with Active Low Output Enable



Introduction

The design element is a bidirectional buffer that supports low-voltage, differential signaling. For the IOBUFDS, a design level interface signal is represented as two distinct ports (IO and IOB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET_P and MYNET_N). Optionally, a programmable differential termination feature is available to help improve signal integrity and reduce external components. Also available is a programmable delay is to assist in the capturing of incoming data to the device.

Logic Table

Inputs		Bidirectional		Outputs
I	**T**	**IO**	**IOB**	**O**
X	1	Z	Z	No Change
0	0	0	1	0
I	0	1	0	1

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Buffer output
IO	Inout	1	Diff_p inout
IOB	Inout	1	Diff_n inout
I	Input	1	Buffer input
T	Input	1	3-state enable input

Design Entry Method

This design element can be used in schematics.

Available Attributes

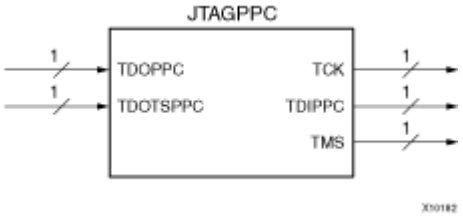
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

JTAGPPC

Primitive: JTAG Primitive for the PowerPC®



Introduction

This design element connects the JTAG logic in the PPC405 core to the JTAG logic of the FPGA device in which the core resides. The connections are made through programmable routing, so the connection only exists after configuration.

Port Descriptions

Inputs	Outputs
TDOPPC	**TCK**
TDOTSPPC	TDIPPC
	TMS

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

KEEPER

Primitive: KEEPER Symbol

Introduction

The design element is a weak keeper element that retains the value of the net connected to its bidirectional O pin. For example, if a logic 1 is being driven onto the net, KEEPER drives a weak/resistive 1 onto the net. If the net driver is then 3-stated, KEEPER continues to drive a weak/resistive 1 onto the net.

Port Descriptions

Name	Direction	Width	Function
O	Output	1-Bit	Keeper output

Design Entry Method

This design element can be used in schematics.

This element can be connected to a net in the following locations on a top-level schematic file:

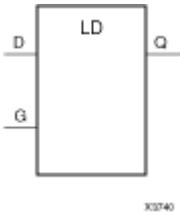
- A net connected to an input IO Marker
- A net connected to both an output IO Marker and 3-statable IO element, such as an OBUFT.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD

Primitive: Transparent Data Latch



Introduction

LD is a transparent data latch. The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	D	D

Design Entry Method

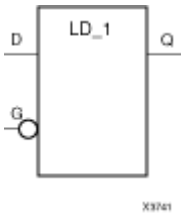
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD_1

Primitive: Transparent Data Latch with Inverted Gate



Introduction

This design element is a transparent data latch with an inverted gate. The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
0	D	D
1	X	No Change
↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

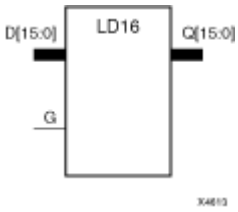
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	Any 1-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD16

Macro: Multiple Transparent Data Latch



Introduction

This design element has 16 transparent data latches with a common gate enable (G). The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	Dn	Dn

Design Entry Method

This design element is only for use in schematics.

Available Attributes

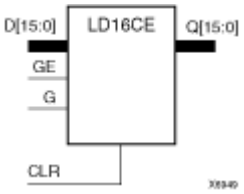
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	Any 16-Bit Value	All zeros	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD16CE

Macro: Transparent Data Latch with Asynchronous Clear and Gate Enable



Introduction

This design element has 16 transparent data latches with asynchronous clear and gate enable. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) outputs Low. (Q) reflects the data (D) inputs while the gate (G) and gate enable (GE) are High, and (CLR) is Low. If (GE) is Low, data on (D) cannot be latched. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**GE**	**G**	**Dn**	**Qn**
1	X	X	X	0
0	0	X	X	No Change
0	1	1	Dn	Dn
0	1	0	X	No Change
0	1	↓	Dn	Dn
Dn = referenced input, for example, D0, D1, D2 Qn = referenced output, for example, Q0, Q1, Q2				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Binary	Any 16-Bit Value	All zeros	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD4

Macro: Multiple Transparent Data Latch

Introduction

This design element has four transparent data latches with a common gate enable (G). The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	Dn	Dn

Design Entry Method

This design element is only for use in schematics.

Available Attributes

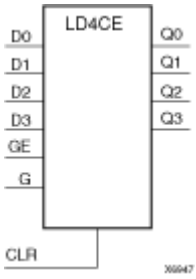
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	Any 4-Bit Value	All zeros	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD4CE

Macro: Transparent Data Latch with Asynchronous Clear and Gate Enable



Introduction

This design element has 4 transparent data latches with asynchronous clear and gate enable. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) outputs Low. (Q) reflects the data (D) inputs while the gate (G) and gate enable (GE) are High, and (CLR) is Low. If (GE) is Low, data on (D) cannot be latched. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**GE**	**G**	**Dn**	**Qn**
1	X	X	X	0
0	0	X	X	No Change
0	1	1	Dn	Dn
0	1	0	X	No Change
0	1	↓	Dn	Dn
Dn = referenced input, for example, D0, D1, D2 Qn = referenced output, for example, Q0, Q1, Q2				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

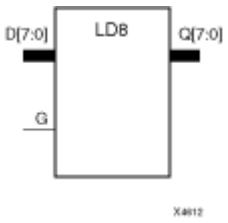
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	Any 4-Bit Value	All zeros	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD8

Macro: Multiple Transparent Data Latch



Introduction

This design element has 8 transparent data latches with a common gate enable (G). The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	Dn	Dn

Design Entry Method

This design element is only for use in schematics.

Available Attributes

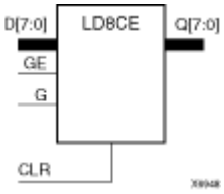
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	Any 8-Bit Value	All zeros	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LD8CE

Macro: Transparent Data Latch with Asynchronous Clear and Gate Enable



Introduction

This design element has 8 transparent data latches with asynchronous clear and gate enable. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) outputs Low. (Q) reflects the data (D) inputs while the gate (G) and gate enable (GE) are High, and (CLR) is Low. If (GE) is Low, data on (D) cannot be latched. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**GE**	**G**	**Dn**	**Qn**
1	X	X	X	0
0	0	X	X	No Change
0	1	1	Dn	Dn
0	1	0	X	No Change
0	1	↓	Dn	Dn
Dn = referenced input, for example, D0, D1, D2 Qn = referenced output, for example, Q0, Q1, Q2				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

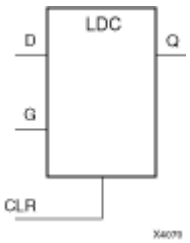
Attribute	Type	Allowed Values	Default	Description
INIT	Binary	Any 8-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDC

Primitive: Transparent Data Latch with Asynchronous Clear



Introduction

This design element is a transparent data latch with asynchronous clear. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. (Q) reflects the data (D) input while the gate enable (G) input is High and (CLR) is Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CLR	**G**	**D**	**Q**
1	X	X	0
0	1	D	D
0	0	X	No Change
0	↓	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

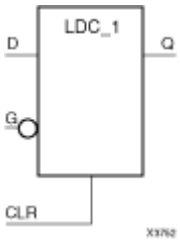
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDC_1

Primitive: Transparent Data Latch with Asynchronous Clear and Inverted Gate



Introduction

This design element is a transparent data latch with asynchronous clear and inverted gate. When the asynchronous clear input (CLR) is High, it overrides the other inputs (D and G) and resets the data (Q) output Low. (Q) reflects the data (D) input while the gate enable (G) input and CLR are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CLR	**G**	**D**	**Q**
1	X	X	0
0	0	D	D
0	1	X	No Change
0	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

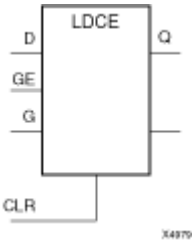
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDCE

Primitive: Transparent Data Latch with Asynchronous Clear and Gate Enable



Introduction

This design element is a transparent data latch with asynchronous clear and gate enable. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. Q reflects the data (D) input while the gate (G) input and gate enable (GE) are High and CLR is Low. If (GE) is Low, data on (D) cannot be latched. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**GE**	**G**	**D**	**Q**
1	X	X	X	0
0	0	X	X	No Change
0	1	1	D	D
0	1	0	X	No Change
0	1	↓	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

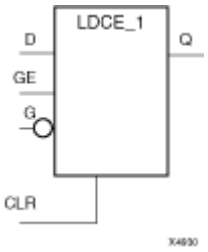
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDCE_1

Primitive: Transparent Data Latch with Asynchronous Clear, Gate Enable, and Inverted Gate



Introduction

This design element is a transparent data latch with asynchronous clear, gate enable, and inverted gate. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. (Q) reflects the data (D) input while the gate (G) input and (CLR) are Low and gate enable (GE) is High. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High or (GE) remains Low

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**GE**	**G**	**D**	**Q**
1	X	X	X	0
0	0	X	X	No Change
0	1	0	D	D
0	1	1	X	No Change
0	1	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Sets the initial value of Q output after configuration.

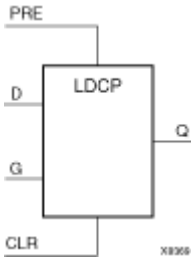
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

LDCP

Primitive: Transparent Data Latch with Asynchronous Clear and Preset



Introduction

The design element is a transparent data latch with data (D), asynchronous clear (CLR) and preset (PRE) inputs. When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When PRE is High and (CLR) is low, it presets the data (Q) output High. (Q) reflects the data (D) input while the gate (G) input is High and (CLR) and PRE are Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**PRE**	**G**	**D**	**Q**
1	X	X	X	0
0	1	X	X	1
0	0	1	D	D
0	0	0	X	No Change
0	0	↓	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

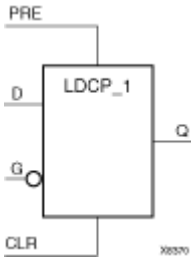
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

LDCP_1

Primitive: Transparent Data Latch with Asynchronous Clear and Preset and Inverted Gate



Introduction

This design element is a transparent data latch with data (D), asynchronous clear (CLR), preset (PRE) inputs, and inverted gate (G). When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When (PRE) is High and (CLR) is Low, it presets the data (Q) output High. (Q) reflects the data (D) input while gate (G) input, (CLR), and (PRE) are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
CLR	**PRE**	**G**	**D**	**Q**
1	X	X	X	0
0	1	X	X	1
0	0	0	D	D
0	0	1	X	No Change
0	0	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

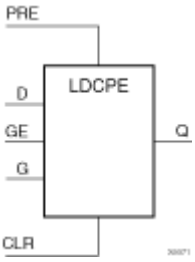
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

LDCPE

Primitive: Transparent Data Latch with Asynchronous Clear and Preset and Gate Enable



Introduction

This design element is a transparent data latch with data (D), asynchronous clear (CLR), asynchronous preset (PRE), and gate enable (GE). When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When (PRE) is High and (CLR) is Low, it presets the data (Q) output High. Q reflects the data (D) input while the gate (G) input and gate enable (GE) are High and (CLR) and PRE are Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the Q output remains unchanged as long as (G) or (GE) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs
CLR	**PRE**	**GE**	**G**	**D**	**Q**
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	1	0	0
0	0	1	1	1	1
0	0	1	0	X	No Change
0	0	1	↓	D	D

Port Descriptions

Port	Direction	Width	Function
Q	Output	1	Data Output
CLR	Input	1	Asynchronous clear/reset input
D	Input	1	Data Input
G	Input	1	Gate Input
GE	Input	1	Gate Enable Input
PRE	Input	1	Asynchronous preset/set input

Design Entry Method

This design element can be used in schematics.

Available Attributes

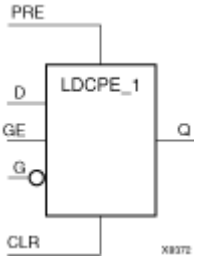
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDCPE_1

Primitive: Transparent Data Latch with Asynchronous Clear and Preset, Gate Enable, and Inverted Gate



Introduction

This design element is a transparent data latch with data (D), asynchronous clear (CLR), asynchronous preset (PRE), gate enable (GE), and inverted gate (G). When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When PRE is High and (CLR) is Low, it presets the data (Q) output High. (Q) reflects the data (D) input while gate enable (GE) is High and gate (G), (CLR), and (PRE) are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) is High or (GE) is Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs					Outputs
CLR	**PRE**	**GE**	**G**	**D**	**Q**
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	0	D	D
0	0	1	1	X	No Change
0	0	1	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

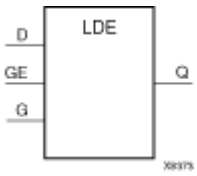
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDE

Primitive: Transparent Data Latch with Gate Enable



Introduction

This design element is a transparent data latch with data (D) and gate enable (GE) inputs. Output (Q) reflects the data (D) while the gate (G) input and gate enable (GE) are High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	1	D	D
1	0	X	No Change
1	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

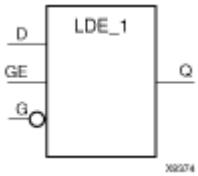
Attribute	Type	Allowed Values	Default	Description
INIT	INTEGER	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDE_1

Primitive: Transparent Data Latch with Gate Enable and Inverted Gate



Introduction

This design element is a transparent data latch with data (D), gate enable (GE), and inverted gate (G). Output (Q) reflects the data (D) while the gate (G) input is Low and gate enable (GE) is High. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) is High or (GE) is Low.

This latch is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
GE	**G**	**D**	**Q**
0	X	X	No Change
1	0	D	D
1	1	X	No Change
1	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

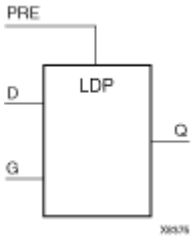
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDP

Primitive: Transparent Data Latch with Asynchronous Preset



Introduction

This design element is a transparent data latch with asynchronous preset (PRE). When the (PRE) input is High, it overrides the other inputs and presets the data (Q) output High. (Q) reflects the data (D) input while gate (G) input is High and (PRE) is Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

The latch is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
PRE	**G**	**D**	**Q**
1	X	X	1
0	1	0	0
0	1	1	1
0	0	X	No Change
0	↓	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

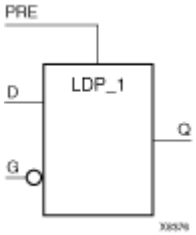
Attribute	Type	Allowed Values	Default	Description
INIT	INTEGER	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDP_1

Primitive: Transparent Data Latch with Asynchronous Preset and Inverted Gate



Introduction

This design element is a transparent data latch with asynchronous preset (PRE) and inverted gate (G). When the (PRE) input is High, it overrides the other inputs and presets the data (Q) output High. (Q) reflects the data (D) input while gate (G) input and (PRE) are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

The latch is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
PRE	**G**	**D**	**Q**
1	X	X	1
0	0	D	D
0	1	X	No Change
0	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

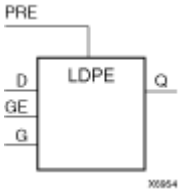
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDPE

Primitive: Transparent Data Latch with Asynchronous Preset and Gate Enable



Introduction

This design element is a transparent data latch with asynchronous preset and gate enable. When the asynchronous preset (PRE) is High, it overrides the other input and presets the data (Q) output High. Q reflects the data (D) input while the gate (G) input and gate enable (GE) are High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

The latch is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
PRE	**GE**	**G**	**D**	**Q**
1	X	X	X	1
0	0	X	X	No Change
0	1	1	D	D
0	1	0	X	No Change
0	1	↓	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

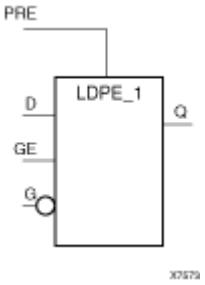
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LDPE_1

Primitive: Transparent Data Latch with Asynchronous Preset, Gate Enable, and Inverted Gate



Introduction

This design element is a transparent data latch with asynchronous preset, gate enable, and inverted gate. When the asynchronous preset (PRE) is High, it overrides the other input and presets the data (Q) output High. (Q) reflects the data (D) input while the gate (G) and (PRE) are Low and gate enable (GE) is High. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High or (GE) remains Low.

The latch is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs				Outputs
PRE	**GE**	**G**	**D**	**Q**
1	X	X	X	1
0	0	X	X	No Change
0	1	0	D	D
0	1	1	X	No Change
0	1	↑	D	D

Design Entry Method

This design element is only for use in schematics.

Available Attributes

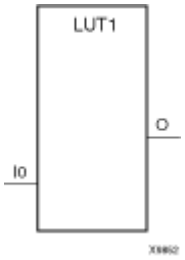
Attribute	Type	Allowed Values	Default	Description
INIT	Integer	0 or 1	0	Specifies the initial value upon power-up or the assertion of GSR for the (Q) port.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT1

Primitive: 1-Bit Look-Up-Table with General Output



Introduction

This design element is a 1-bit look-up-tables (LUT) with general output (O).

An INIT attribute with an appropriate number of hexadecimal digits for the number of inputs must be attached to the LUT to specify its function. This element provides a look-up-table version of a buffer or inverter. These elements are the basic building blocks. Two LUTs are available in each CLB slice; four LUTs are available in each CLB. Multiple variants of LUTs accommodate additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs	Outputs
I0	**O**
0	INIT[0]
1	INIT[1]
INIT = Binary number assigned to the INIT attribute	

Design Entry Method

This design element can be used in schematics.

Available Attributes

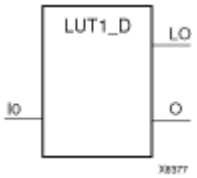
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 2-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT1_D

Primitive: 1-Bit Look-Up-Table with Dual Output



Introduction

This design element is a 1-bit look-up-table (LUT) with two functionally identical outputs, O and LO. It provides a look-up-table version of a buffer or inverter.

The O output is a general interconnect. The LO output is used to connect to another output within the same CLB slice and to the fast connect buffer. A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs	Outputs	
I0	**O**	**LO**
0	INIT[0]	INIT[0]
1	INIT[1]	INIT[1]
INIT = Binary number assigned to the INIT attribute		

Design Entry Method

This design element can be used in schematics.

Available Attributes

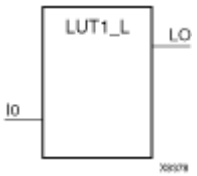
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 2-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT1_L

Primitive: 1-Bit Look-Up-Table with Local Output



Introduction

This design element is a 1- bit look-up-tables (LUTs) with a local output (LO) that is used to connect to another output within the same CLB slice and to the fast connect buffer. It provides a look-up-table version of a buffer or inverter.

A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs	Outputs
I0	**LO**
0	INIT[0]
1	INIT[1]
INIT = Binary number assigned to the INIT attribute	

Design Entry Method

This design element can be used in schematics.

Available Attributes

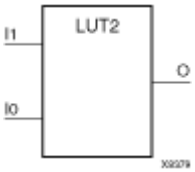
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 2-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT2

Primitive: 2-Bit Look-Up-Table with General Output



Introduction

This design element is a 2-bit look-up-table (LUT) with general output (O).

An INIT attribute with an appropriate number of hexadecimal digits for the number of inputs must be attached to the LUT to specify its function. This element provides a look-up-table version of a buffer or inverter. These elements are the basic building blocks. Two LUTs are available in each CLB slice; four LUTs are available in each CLB. Multiple variants of LUTs accommodate additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs		Outputs
I1	**I0**	**O**
0	0	INIT[0]
0	1	INIT[1]
1	0	INIT[2]
1	1	INIT[3]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute		

Design Entry Method

This design element can be used in schematics.

Available Attributes

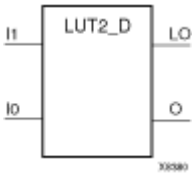
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 4-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT2_D

Primitive: 2-Bit Look-Up-Table with Dual Output



Introduction

This design element is a 2-bit look-up-tables (LUTs) with two functionally identical outputs, O and LO.

The O output is a general interconnect. The LO output is used to connect to another output within the same CLB slice and to the fast connect buffer. A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs		Outputs	
I1	**I0**	**O**	**LO**
0	0	INIT[0]	INIT[0]
0	1	INIT[1]	INIT[1]
1	0	INIT[2]	INIT[2]
1	1	INIT[3]	INIT[3]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute			

Design Entry Method

This design element can be used in schematics.

Available Attributes

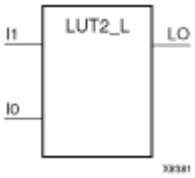
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 4-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT2_L

Primitive: 2-Bit Look-Up-Table with Local Output



Introduction

This design element is a 2- bit look-up-tables (LUTs) with a local output (LO) that is used to connect to another output within the same CLB slice and to the fast connect buffer. It provides a look-up-table version of a buffer or inverter.

A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs		Outputs
I1	**I0**	**LO**
0	0	INIT[0]
0	1	INIT[1]
1	0	INIT[2]
1	1	INIT[3]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute		

Design Entry Method

This design element can be used in schematics.

Available Attributes

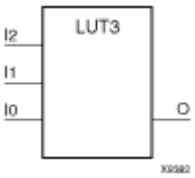
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 4-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT3

Primitive: 3-Bit Look-Up-Table with General Output



Introduction

This design element is a 3-bit look-up-table (LUT) with general output (O). A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

An INIT attribute with an appropriate number of hexadecimal digits for the number of inputs must be attached to the LUT to specify its function. This element provides a look-up-table version of a buffer or inverter. These elements are the basic building blocks. Two LUTs are available in each CLB slice; four LUTs are available in each CLB. Multiple variants of LUTs accommodate additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Logic Table

Inputs			Outputs
I2	**I1**	**I0**	**O**
0	0	0	INIT[0]
0	0	1	INIT[1]
0	1	0	INIT[2]
0	1	1	INIT[3]
1	0	0	INIT[4]
1	0	1	INIT[5]
1	1	0	INIT[6]
1	1	1	INIT[7]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute			

Design Entry Method

This design element can be used in schematics.

Available Attributes

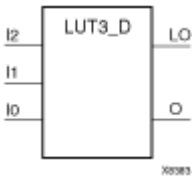
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 8-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT3_D

Primitive: 3-Bit Look-Up-Table with Dual Output



Introduction

This design element is a 3-bit look-up-tables (LUTs) with two functionally identical outputs, O and LO.

The O output is a general interconnect. The LO output is used to connect to another output within the same CLB slice and to the fast connect buffer. A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Logic Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary logic table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs			Outputs	
I2	**I1**	**I0**	**O**	**LO**
0	0	0	INIT[0]	INIT[0]
0	0	1	INIT[1]	INIT[1]
0	1	0	INIT[2]	INIT[2]
0	1	1	INIT[3]	INIT[3]
1	0	0	INIT[4]	INIT[4]
1	0	1	INIT[5]	INIT[5]
1	1	0	INIT[6]	INIT[6]
1	1	1	INIT[7]	INIT[7]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute				

Design Entry Method

This design element can be used in schematics.

Available Attributes

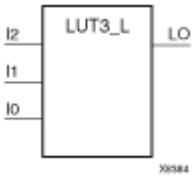
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 8-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT3_L

Primitive: 3-Bit Look-Up-Table with Local Output



Introduction

This design element is a 3- bit look-up-tables (LUTs) with a local output (LO) that is used to connect to another output within the same CLB slice and to the fast connect buffer. It provides a look-up-table version of a buffer or inverter.

A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs			Outputs
I2	**I1**	**I0**	**LO**
0	0	0	INIT[0]
0	0	1	INIT[1]
0	1	0	INIT[2]
0	1	1	INIT[3]
1	0	0	INIT[4]
1	0	1	INIT[5]
1	1	0	INIT[6]
1	1	1	INIT[7]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute			

Design Entry Method

This design element can be used in schematics.

Available Attributes

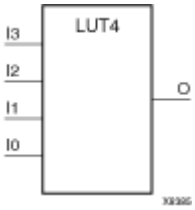
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 8-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT4

Primitive: 4-Bit Look-Up-Table with General Output



Introduction

This design element is a 4-bit look-up-tables (LUT) with general output (O).

An INIT attribute with an appropriate number of hexadecimal digits for the number of inputs must be attached to the LUT to specify its function. This element provides a look-up-table version of a buffer or inverter. These elements are the basic building blocks. Two LUTs are available in each CLB slice; four LUTs are available in each CLB. Multiple variants of LUTs accommodate additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs				Outputs
I3	**I2**	**I1**	**I0**	**O**
0	0	0	0	INIT[0]
0	0	0	1	INIT[1]
0	0	1	0	INIT[2]
0	0	1	1	INIT[3]
0	1	0	0	INIT[4]
0	1	0	1	INIT[5]
0	1	1	0	INIT[6]
0	1	1	1	INIT[7]
1	0	0	0	INIT[8]
1	0	0	1	INIT[9]
1	0	1	0	INIT[10]

Inputs				Outputs
I3	**I2**	**I1**	**I0**	**O**
1	0	1	1	INIT[11]
1	1	0	0	INIT[12]
1	1	0	1	INIT[13]
1	1	1	0	INIT14]
1	1	1	1	INIT[15]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute				

Design Entry Method

This design element can be used in schematics.

Available Attributes

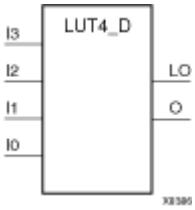
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT4_D

Primitive: 4-Bit Look-Up-Table with Dual Output



Introduction

This design element is a 4-bit look-up-tables (LUTs) with two functionally identical outputs, O and LO

The O output is a general interconnect. The LO output is used to connect to another output within the same CLB slice and to the fast connect buffer. A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs				Outputs	
I3	**I2**	**I1**	**I0**	**O**	**LO**
0	0	0	0	INIT[0]	INIT[0]
0	0	0	1	INIT[1]	INIT[1]
0	0	1	0	INIT[2]	INIT[2]
0	0	1	1	INIT[3]	INIT[3]
0	1	0	0	INIT[4]	INIT[4]
0	1	0	1	INIT[5]	INIT[5]
0	1	1	0	INIT[6]	INIT[6]
0	1	1	1	INIT[7]	INIT[7]
1	0	0	0	INIT[8]	INIT[8]
1	0	0	1	INIT[9]	INIT[9]
1	0	1	0	INIT[10]	INIT[10]
1	0	1	1	INIT[11]	INIT[11]
1	1	0	0	INIT[12]	INIT[12]

Inputs				Outputs	
I3	**I2**	**I1**	**I0**	**O**	**LO**
1	1	0	1	INIT[13]	INIT[13]
1	1	1	0	INIT14]	INIT14]
1	1	1	1	INIT[15]	INIT[15]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute					

Design Entry Method

This design element can be used in schematics.

Available Attributes

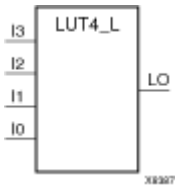
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

LUT4_L

Primitive: 4-Bit Look-Up-Table with Local Output



Introduction

This design element is a 4- bit look-up-tables (LUTs) with a local output (LO) that is used to connect to another output within the same CLB slice and to the fast connect buffer. It provides a look-up-table version of a buffer or inverter.

A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

The INIT parameter for the FPGA LUT primitive is what gives the LUT its logical value. By default, this value is zero, thus driving the output to a zero regardless of the input values (acting as a ground). However, in most cases a new INIT value must be determined in order to specify the logic function for the LUT primitive. There are at least two methods by which the LUT value can be determined:

The Truth Table Method -A common method to determine the desired INIT value for a LUT is using a truth table. To do so, simply create a binary truth table of all possible inputs, specify the desired logic value of the output and then create the INIT string from those output values.

The Equation Method -Another method to determine the LUT value is to define parameters for each input to the LUT that correspond to their listed truth value and use those to build the logic equation you are after. This method is easier to understand once you have grasped the concept and more self-documenting that the above method however does require the code to first specify the appropriate parameters.

Logic Table

Inputs				Outputs
I3	**I2**	**I1**	**I0**	**LO**
0	0	0	0	INIT[0]
0	0	0	1	INIT[1]
0	0	1	0	INIT[2]
0	0	1	1	INIT[3]
0	1	0	0	INIT[4]
0	1	0	1	INIT[5]
0	1	1	0	INIT[6]
0	1	1	1	INIT[7]
1	0	0	0	INIT[8]
1	0	0	1	INIT[9]
1	0	1	0	INIT[10]
1	0	1	1	INIT[11]
1	1	0	0	INIT[12]

Inputs				Outputs
I3	**I2**	**I1**	**I0**	**LO**
1	1	0	1	INIT[13]
1	1	1	0	INIT14]
1	1	1	1	INIT[15]
INIT = Binary equivalent of the hexadecimal number assigned to the INIT attribute				

Design Entry Method

This design element can be used in schematics.

Available Attributes

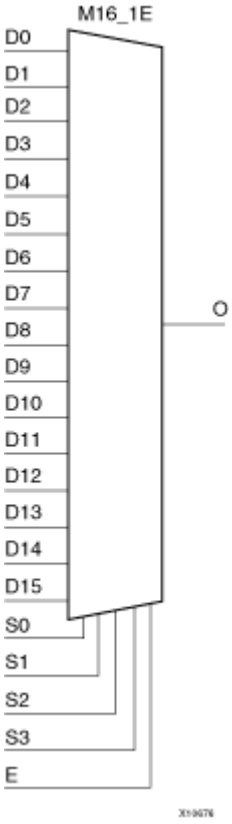
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Initializes look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

M16_1E

Macro: 16-to-1 Multiplexer with Enable



Introduction

This design element is a 16-to-1 multiplexer with enable. When the enable input (E) is High, the M16_1E multiplexer chooses one data bit from 16 sources (D15 – D0) under the control of the select inputs (S3 – S0). The output (O) reflects the state of the selected input as shown in the logic table. When (E) is Low, the output is Low.

Logic Table

Inputs						Outputs
E	**S3**	**S2**	**S1**	**S0**	**D15-D0**	**O**
0	X	X	X	X	X	0
1	0	0	0	0	D0	D0
1	0	0	0	1	D1	D1
1	0	0	1	0	D2	D2
1	0	0	1	1	D3	D3
. . .	. . .	. . .	. . .	. . .	. . .	. . .
1	1	1	0	0	D12	D12

Inputs						Outputs
E	**S3**	**S2**	**S1**	**S0**	**D15-D0**	**O**
1	1	1	0	1	D13	D13
1	1	1	1	0	D14	D14
1	1	1	1	1	D15	D15

Design Entry Method

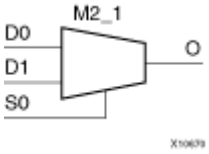
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

M2_1

Macro: 2-to-1 Multiplexer



Introduction

This design element chooses one data bit from two sources (D1 or D0) under the control of the select input (S0). The output (O) reflects the state of the selected data input. When Low, S0 selects D0 and when High, S0 selects D1.

Logic Table

Inputs			Outputs
S0	**D1**	**D0**	**O**
1	D1	X	D1
0	X	D0	D0

Design Entry Method

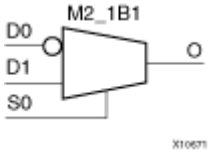
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

M2_1B1

Macro: 2-to-1 Multiplexer with D0 Inverted



Introduction

This design element chooses one data bit from two sources (D1 or D0) under the control of select input (S0). When S0 is Low, the output (O) reflects the inverted value of (D0). When S0 is High, (O) reflects the state of D1.

Logic Table

Inputs			Outputs
S0	**D1**	**D0**	**O**
1	1	X	1
1	0	X	0
0	X	1	0
0	X	0	1

Design Entry Method

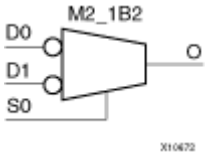
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

M2_1B2

Macro: 2-to-1 Multiplexer with D0 and D1 Inverted



Introduction

This design element chooses one data bit from two sources (D1 or D0) under the control of select input (S0). When S0 is Low, the output (O) reflects the inverted value of D0. When S0 is High, O reflects the inverted value of D1.

Logic Table

Inputs			Outputs
S0	**D1**	**D0**	**O**
1	1	X	0
1	0	X	1
0	X	1	0
0	X	0	1

Design Entry Method

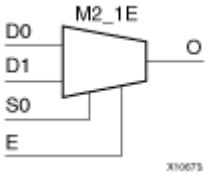
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

M2_1E

Macro: 2-to-1 Multiplexer with Enable



Introduction

This design element is a 2-to-1 multiplexer with enable. When the enable input (E) is High, the M2_1E chooses one data bit from two sources (D1 or D0) under the control of select input (S0). When Low, S0 selects D0 and when High, S0 selects D1. When (E) is Low, the output is Low.

Logic Table

Inputs				Outputs
E	**S0**	**D1**	**D0**	**O**
0	X	X	X	0
1	0	X	1	1
1	0	X	0	0
1	1	1	X	1
1	1	0	X	0

Design Entry Method

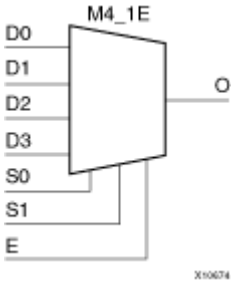
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

M4_1E

Macro: 4-to-1 Multiplexer with Enable



Introduction

This design element is a 4-to-1 multiplexer with enable. When the enable input (E) is High, the M4_1E multiplexer chooses one data bit from four sources (D3, D2, D1, or D0) under the control of the select inputs (S1 – S0). The output (O) reflects the state of the selected input as shown in the logic table. When (E) is Low, the output is Low.

Logic Table

Inputs							Outputs
E	**S1**	**S0**	**D0**	**D1**	**D2**	**D3**	**O**
0	X	X	X	X	X	X	0
1	0	0	D0	X	X	X	D0
1	0	1	X	D1	X	X	D1
1	1	0	X	X	D2	X	D2
1	1	1	X	X	X	D3	D3

Design Entry Method

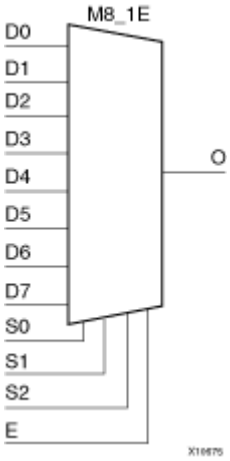
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

M8_1E

Macro: 8-to-1 Multiplexer with Enable



Introduction

This design element is an 8-to-1 multiplexer with enable. When the enable input (E) is High, the M8_1E multiplexer chooses one data bit from eight sources (D7 – D0) under the control of the select inputs (S2 – S0). The output (O) reflects the state of the selected input as shown in the logic table. When (E) is Low, the output is Low.

Logic Table

Inputs					Outputs
E	**S2**	**S1**	**S0**	**D7-D0**	**O**
0	X	X	X	X	0
1	0	0	0	D0	D0
1	0	0	1	D1	D1
1	0	1	0	D2	D2
1	0	1	1	D3	D3
1	1	0	0	D4	D4
1	1	0	1	D5	D5
1	1	1	0	D6	D6
1	1	1	1	D7	D7

Design Entry Method

This design element is only for use in schematics.

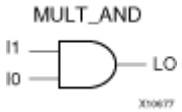
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

MULT_AND

Primitive: Fast Multiplier AND



Introduction

The design element is an AND component located within the slice where the two inputs are shared with the 4-input LUT and the output drives into the carry logic. This added logic is especially useful for building fast and smaller multipliers however be used for other purposes as well. The I1 and I0 inputs must be connected to the I1 and I0 inputs of the associated LUT. The LO output must be connected to the DI input of the associated MUXCY, MUXCY_D, or MUXCY_L.

Logic Table

Inputs		Outputs
I1	**I0**	**LO**
0	0	0
0	1	0
1	0	0
1	1	1

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MULT18X18

Primitive: 18 x 18 Signed Multiplier

Introduction

MULT18X18 is a combinational signed 18-bit by 18-bit multiplier. The value represented in the 18-bit input A is multiplied by the value represented in the 18-bit input B. Output P is the 36-bit product of A and B.

Logic Table

Inputs		Output
A	**B**	**P**
A	B	A x B
A, B, and P are two's complement.		

Design Entry Method

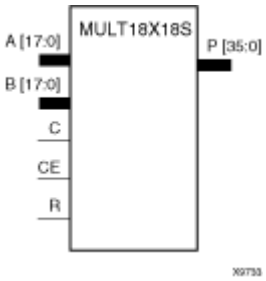
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MULT18X18S

Primitive: 18 x 18 Signed Multiplier – Registered Version



Introduction

MULT18X18S is the registered version of the 18 x 18 signed multiplier with output P and inputs A, B, C, CE, and R. The registers are initialized to 0 after the GSR pulse.

The value represented in the 18-bit input A is multiplied by the value represented in the 18-bit input B. Output P is the 36-bit product of A and B.

Logic Table

Inputs					Output
C	**CE**	**Am**	**Bn**	**R**	**P**
↑	X	X	X	1	0
↑	1	Am	Bn	0	A x B
X	0	X	X	0	No Change
A, B, and P are two's complement.					

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXCY

Primitive: 2-to-1 Multiplexer for Carry Logic with General Output

Introduction

The direct input (DI) of a slice is connected to the (DI) input of the MUXCY. The carry in (CI) input of an LC is connected to the CI input of the MUXCY. The select input (S) of the MUXCY is driven by the output of the Look-Up Table (LUT) and configured as a MUX function. The carry out (O) of the MUXCY reflects the state of the selected input and implements the carry out function of each LC. When Low, S selects DI; when High, S selects CI.

The variants “MUXCY_D” and “MUXCY_L” provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Logic Table

Inputs			Outputs
S	**DI**	**CI**	**O**
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Design Entry Method

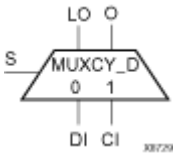
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXCY_D

Primitive: 2-to-1 Multiplexer for Carry Logic with Dual Output



Introduction

This design element implements a 1-bit, high-speed carry propagate function. One such function can be implemented per logic cell (LC), for a total of 4-bits per configurable logic block (CLB). The direct input (DI) of an LC is connected to the DI input of the MUXCY_D. The carry in (CI) input of an LC is connected to the CI input of the MUXCY_D. The select input (S) of the MUX is driven by the output of the Look-Up Table (LUT) and configured as an XOR function. The carry out (O and LO) of the MUXCY_D reflects the state of the selected input and implements the carry out function of each LC. When Low, S selects DI; when High, S selects CI.

Outputs O and LO are functionally identical. The O output is a general interconnect. See also “MUXCY” and “MUXCY_L”.

Logic Table

Inputs			Outputs	
S	**DI**	**CI**	**O**	**LO**
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

Design Entry Method

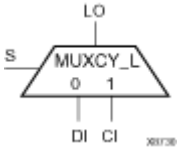
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXCY_L

Primitive: 2-to-1 Multiplexer for Carry Logic with Local Output



Introduction

This design element implements a 1-bit high-speed carry propagate function. One such function is implemented per logic cell (LC), for a total of 4-bits per configurable logic block (CLB). The direct input (DI) of an LC is connected to the DI input of the MUXCY_L. The carry in (CI) input of an LC is connected to the CI input of the MUXCY_L. The select input (S) of the MUXCY_L is driven by the output of the Look-Up Table (LUT) and configured as an XOR function. The carry out (LO) of the MUXCY_L reflects the state of the selected input and implements the carry out function of each (LC). When Low, (S) selects DI; when High, (S) selects (CI).

See also “MUXCY” and “MUXCY_D.”

Logic Table

Inputs			Outputs
S	**DI**	**CI**	**LO**
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Design Entry Method

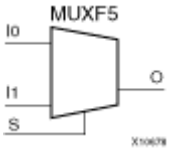
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF5

Primitive: 2-to-1 Look-Up Table Multiplexer with General Output



Introduction

This design element provides a multiplexer function in a CLB slice for creating a function-of-5 lookup table or a 4-to-1 multiplexer in combination with the associated lookup tables. The local outputs (LO) from the two lookup tables are connected to the I0 and I1 inputs of the MUXF5. The (S) input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The variants, “MUXF5_D” and “MUXF5_L”, provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Logic Table

Inputs			Outputs
S	**I0**	**I1**	**O**
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Design Entry Method

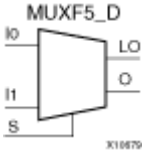
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF5_D

Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output



Introduction

This design element provides a multiplexer function in a CLB slice for creating a function-of-5 lookup table or a 4-to-1 multiplexer in combination with the associated lookup tables. The local outputs (LO) from the two lookup tables are connected to the I0 and I1 inputs of the MUXF5. The S input is driven from any internal net. When Low, S selects I0. When High, S selects I1.

Outputs O and LO are functionally identical. The O output is a general interconnect. The LO output connects to other inputs in the same CLB slice. See also “MUXF5” and “MUXF5_L”

Logic Table

Inputs			Outputs	
S	**I0**	**I1**	**O**	**LO**
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

Design Entry Method

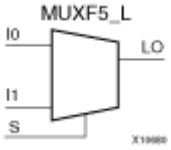
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF5_L

Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output



Introduction

This design element provides a multiplexer function in a CLB slice for creating a function-of-5 lookup table or a 4-to-1 multiplexer in combination with the associated lookup tables. The local outputs (LO) from the two lookup tables are connected to the I0 and I1 inputs of the MUXF5. The S input is driven from any internal net. When Low, S selects I0. When High, S selects I1.

The LO output connects to other inputs in the same CLB slice.

See also “MUXF5” and “MUXF5_D”

Logic Table

Inputs			Output
S	**I0**	**I1**	**LO**
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Design Entry Method

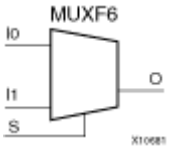
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF6

Primitive: 2-to-1 Look-Up Table Multiplexer with General Output



Introduction

This design element provides a multiplexer function in two slices for creating a function-of-6 lookup table or an 8-to-1 multiplexer in combination with the associated four lookup tables and two MUXF5s. The local outputs (LO) from the two MUXF5s in the CLB are connected to the I0 and I1 inputs of the MUXF6. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The variants, “MUXF6_D” and “MUXF6_L”, provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Logic Table

Inputs			Outputs
S	**I0**	**I1**	**O**
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Design Entry Method

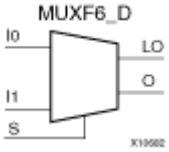
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF6_D

Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output



Introduction

This design element provides a multiplexer function in a two slices for creating a function-of-6 lookup table or an 8-to-1 multiplexer in combination with the associated four lookup tables and two MUXF5s. The local outputs (LO) from the two MUXF5s in the CLB are connected to the I0 and I1 inputs of the MUXF6. The (S) input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

Outputs (O) and (LO) are functionally identical. The (O) output is a general interconnect. The (LO) output connects to other inputs in the same CLB slice.

Logic Table

Inputs			Outputs	
S	**I0**	**I1**	**O**	**LO**
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

Design Entry Method

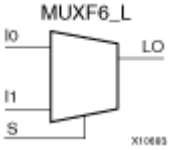
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF6_L

Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output



Introduction

This design element provides a multiplexer function for use in creating a function-of-6 lookup table or an 8-to-1 multiplexer in combination with the associated four lookup tables and two MUXF5s. The local outputs (LO) from the two MUXF5s in the (CLB) are connected to the I0 and I1 inputs of the MUXF6. The (S) input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The LO output connects to other inputs in the same CLB slice.

Logic Table

Inputs			Output
S	**I0**	**I1**	**LO**
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Design Entry Method

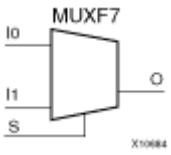
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF7

Primitive: 2-to-1 Look-Up Table Multiplexer with General Output



Introduction

This design element provides a multiplexer function for use in creating a function-of-7 Look-Up Table or a 16-to-1 multiplexer in combination with the associated Look-Up Tables. Local outputs (LO) of MUXF6 are connected to the I0 and I1 inputs of the MUXF7. The (S) input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The variants, “MUXF7_D” and “MUXF7_L”, provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Logic Table

Inputs			Outputs
S	**I0**	**I1**	**O**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Output of MUX to general routing
I0	Input	1	Input (tie to MUXF6 LO out)
I1	Input	1	Input (tie to MUXF6 LO out)
S	Input	1	Input select to MUX

Design Entry Method

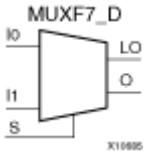
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF7_D

Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output



Introduction

This design element provides a multiplexer function for use in creating a function-of-7 Look-Up Table or a 16-to-1 multiplexer in combination with the associated Look-Up Tables. Local outputs (LO) of MUXF6 are connected to the I0 and I1 inputs of the MUXF7. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

Outputs O and LO are functionally identical. The O output is a general interconnect. The LO output connects to other inputs in the same CLB slice.

Logic Table

Inputs			Outputs	
S	**I0**	**I1**	**O**	**LO**
0	I0	X	I0	I0
1	X	I1	I1	I1
X	0	0	0	0
X	1	1	1	1

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Output of MUX to general routing
LO	Output	1	Output of MUX to local routing
I0	Input	1	Input (tie to MUXF6 LO out)
I1	Input	1	Input (tie to MUXF6 LO out)
S	Input	1	Input select to MUX

Design Entry Method

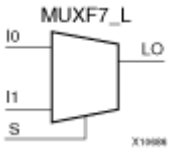
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF7_L

Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output



Introduction

This design element provides a multiplexer function for use in creating a function-of-7 Look-Up Table or a 16-to-1 multiplexer in combination with the associated Look-Up Tables. Local outputs (LO) of MUXF6 are connected to the I0 and I1 inputs of the MUXF7. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The LO output connects to other inputs in the same CLB slice.

Logic Table

Inputs			Output
S	**I0**	**I1**	**LO**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Port Descriptions

Port	Direction	Width	Function
LO	Output	1	Output of MUX to local routing
I0	Input	1	Input (tie to MUXF6 LO out)
I1	Input	1	Input (tie to MUXF6 LO out)
S	Input	1	Input select to MUX

Design Entry Method

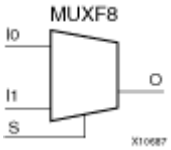
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF8

Primitive: 2-to-1 Look-Up Table Multiplexer with General Output



Introduction

This design element provides a multiplexer function in eight slices for creating a function-of-8 Look-Up Table or a 32-to-1 multiplexer in combination with the associated Look-Up Tables, MUXF5s, MUXF6s, and MUXF7s. Local outputs (LO) of MUXF7 are connected to the I0 and I1 inputs of the MUXF8. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

Logic Table

Inputs			Outputs
S	**I0**	**I1**	**O**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Output of MUX to general routing
I0	Input	1	Input (tie to MUXF7 LO out)
I1	Input	1	Input (tie to MUXF7 LO out)
S	Input	1	Input select to MUX

Design Entry Method

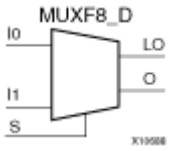
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF8_D

Primitive: 2-to-1 Look-Up Table Multiplexer with Dual Output



Introduction

This design element provides a multiplexer function in eight slices for creating a function-of-8 Look-Up Table or a 32-to-1 multiplexer in combination with the associated four Look-Up Tables and two MUXF8s. Local outputs (LO) of MUXF7 are connected to the I0 and I1 inputs of the MUXF8. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

Outputs O and LO are functionally identical. The O output is a general interconnect. The LO output connects to other inputs in the same CLB slice.

Logic Table

Inputs			Outputs	
S	**I0**	**I1**	**O**	**LO**
0	I0	X	I0	I0
1	X	I1	I1	I1
X	0	0	0	0
X	1	1	1	1

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Output of MUX to general routing
LO	Output	1	Output of MUX to local routing
I0	Input	1	Input (tie to MUXF7 LO out)
I1	Input	1	Input (tie to MUXF7 LO out)
S	Input	1	Input select to MUX

Design Entry Method

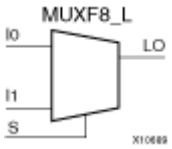
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

MUXF8_L

Primitive: 2-to-1 Look-Up Table Multiplexer with Local Output



Introduction

This design element provides a multiplexer function in eight slices for creating a function-of-8 Look-Up Table or a 32-to-1 multiplexer in combination with the associated four Look-Up Tables and two MUXF8s. Local outputs (LO) of MUXF7 are connected to the I0 and I1 inputs of the MUXF8. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The LO output connects to other inputs in the same CLB slice.

Logic Table

Inputs			Output
S	**I0**	**I1**	**LO**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Port Descriptions

Port	Direction	Width	Function
LO	Output	1	Output of MUX to local routing
I0	Input	1	Input (tie to MUXF7 LO out)
I1	Input	1	Input (tie to MUXF7 LO out)
S	Input	1	Input select to MUX

Design Entry Method

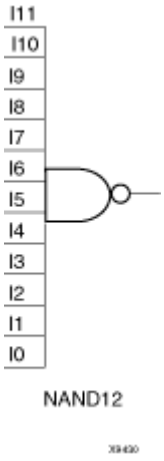
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND12

Macro: 12- Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

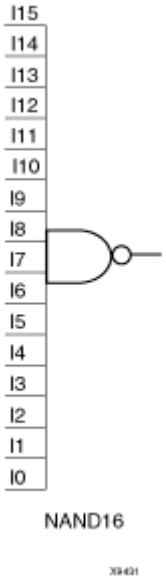
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND16

Macro: 16- Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

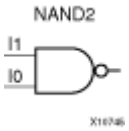
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND2

Primitive: 2-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND2B1

Primitive: 2-Input NAND Gate with 1 Inverted and 1 Non-Inverted Inputs

Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND2B2

Primitive: 2-Input NAND Gate with Inverted Inputs

Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

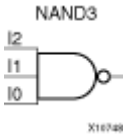
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND3

Primitive: 3-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

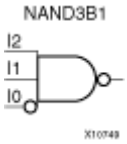
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND3B1

Primitive: 3-Input NAND Gate with 1 Inverted and 2 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

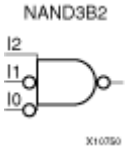
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND3B2

Primitive: 3-Input NAND Gate with 2 Inverted and 1 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

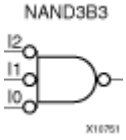
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND3B3

Primitive: 3-Input NAND Gate with Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

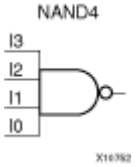
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND4

Primitive: 4-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

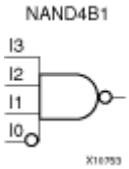
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND4B1

Primitive: 4-Input NAND Gate with 1 Inverted and 3 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

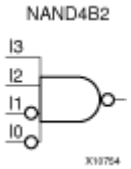
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND4B2

Primitive: 4-Input NAND Gate with 2 Inverted and 2 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

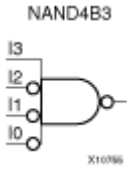
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND4B3

Primitive: 4-Input NAND Gate with 3 Inverted and 1 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

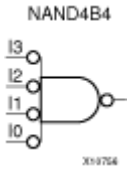
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND4B4

Primitive: 4-Input NAND Gate with Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

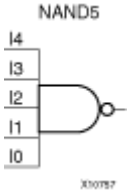
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND5

Primitive: 5-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

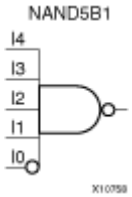
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND5B1

Primitive: 5-Input NAND Gate with 1 Inverted and 4 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

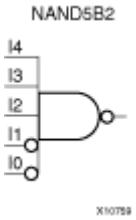
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND5B2

Primitive: 5-Input NAND Gate with 2 Inverted and 3 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

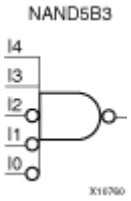
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND5B3

Primitive: 5-Input NAND Gate with 3 Inverted and 2 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

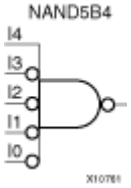
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND5B4

Primitive: 5-Input NAND Gate with 4 Inverted and 1 Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

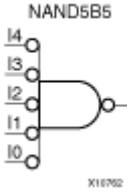
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND5B5

Primitive: 5-Input NAND Gate with Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

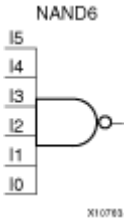
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND6

Macro: 6-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

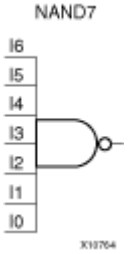
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND7

Macro: 7-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

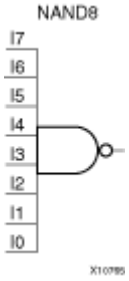
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND8

Macro: 8-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

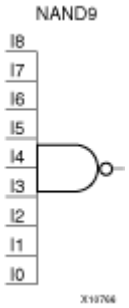
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NAND9

Macro: 9-Input NAND Gate with Non-Inverted Inputs



Introduction

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

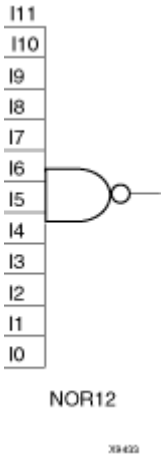
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR12

Macro: 12-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

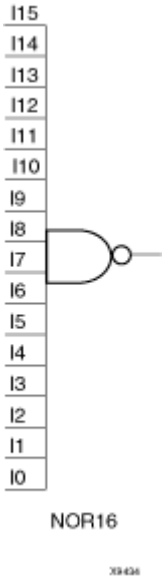
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR16

Macro: 16-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

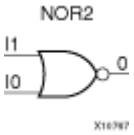
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR2

Primitive: 2-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

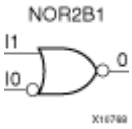
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR2B1

Primitive: 2-Input NOR Gate with 1 Inverted and 1 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

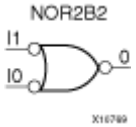
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR2B2

Primitive: 2-Input NOR Gate with Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

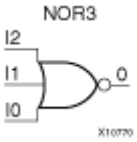
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR3

Primitive: 3-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

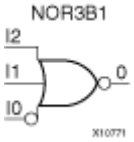
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR3B1

Primitive: 3-Input NOR Gate with 1 Inverted and 2 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

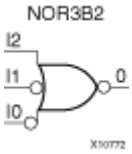
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR3B2

Primitive: 3-Input NOR Gate with 2 Inverted and 1 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

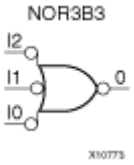
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR3B3

Primitive: 3-Input NOR Gate with Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

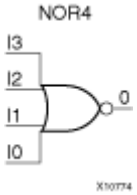
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR4

Primitive: 4-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

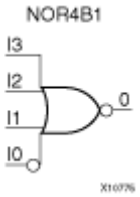
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR4B1

Primitive: 4-Input NOR Gate with 1 Inverted and 3 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

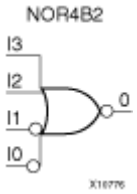
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR4B2

Primitive: 4-Input NOR Gate with 2 Inverted and 2 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

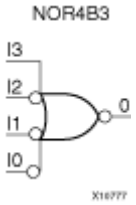
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR4B3

Primitive: 4-Input NOR Gate with 3 Inverted and 1 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

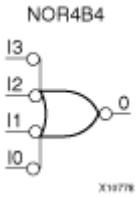
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR4B4

Primitive: 4-Input NOR Gate with Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

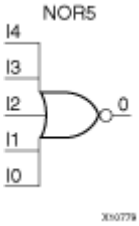
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR5

Primitive: 5-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

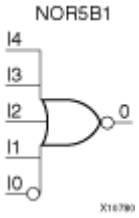
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR5B1

Primitive: 5-Input NOR Gate with 1 Inverted and 4 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

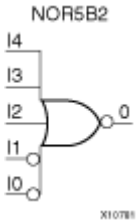
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR5B2

Primitive: 5-Input NOR Gate with 2 Inverted and 3 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

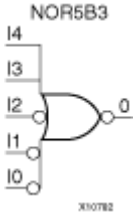
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR5B3

Primitive: 5-Input NOR Gate with 3 Inverted and 2 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

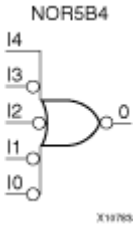
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR5B4

Primitive: 5-Input NOR Gate with 4 Inverted and 1 Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

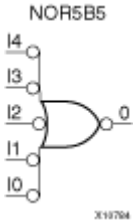
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR5B5

Primitive: 5-Input NOR Gate with Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

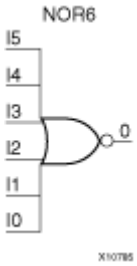
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR6

Macro: 6-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

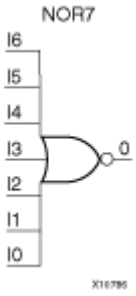
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR7

Macro: 7-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

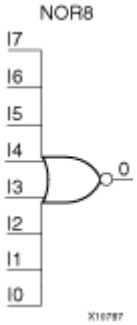
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR8

Macro: 8-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

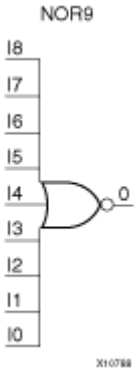
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

NOR9

Macro: 9-Input NOR Gate with Non-Inverted Inputs



Introduction

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs, 12 inputs, and 16 inputs are available only with non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUF

Primitive: Output Buffer

Introduction

This design element is a simple output buffer used to drive output signals to the FPGA device pins that do not need to be 3-stated (constantly driven). Either an OBUF, OBUFT, OBUFDS, or OBUFTDS must be connected to every output port in the design.

This element isolates the internal circuit and provides drive current for signals leaving a chip. It exists in input/output blocks (IOB). Its output (O) is connected to an OPAD or an IOPAD. The interface standard used by this element is LVTTL. Also, this element has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Output of OBUF to be connected directly to top-level output port.
I	Input	1	Input of OBUF. Connect to the logic driving the output port.

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUF16

Macro: 16-Bit Output Buffer

Introduction

This design element is a multiple output buffer.

This element isolates the internal circuit and provides drive current for signals leaving a chip. It exists in input/output blocks (IOB). Its output (O) is connected to an OPAD or an IOPAD. The interface standard used by this element is LVTTL. Also, this element has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

Design Entry Method

This design element can be used in schematics.

Available Attributes

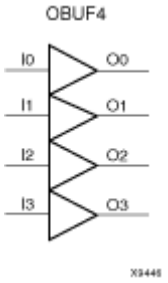
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUF4

Macro: 4-Bit Output Buffer



Introduction

This design element is a multiple output buffer.

This element isolates the internal circuit and provides drive current for signals leaving a chip. It exists in input/output blocks (IOB). Its output (O) is connected to an OPAD or an IOPAD. The interface standard used by this element is LVTTL. Also, this element has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUF8

Macro: 8-Bit Output Buffer

Introduction

This design element is a multiple output buffer.

This element isolates the internal circuit and provides drive current for signals leaving a chip. It exists in input/output blocks (IOB). Its output (O) is connected to an OPAD or an IOPAD. The interface standard used by this element is LVTTL. Also, this element has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

Design Entry Method

This design element can be used in schematics.

Available Attributes

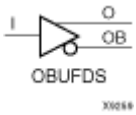
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFDS

Primitive: Differential Signaling Output Buffer



Introduction

This design element is a single output buffer that supports low-voltage, differential signaling (1.8 v CMOS). OBUFDS isolates the internal circuit and provides drive current for signals leaving the chip. Its output is represented as two distinct ports (O and OB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET and MYNETB).

Logic Table

Inputs		Outputs
I	**O**	**OB**
0	0	1
1	1	0

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Diff_p output (connect directly to top level port)
OB	Input	1	Diff_n output (connect directly to top level port)
I	Input	1	Buffer input

Design Entry Method

This design element can be used in schematics.

Available Attributes

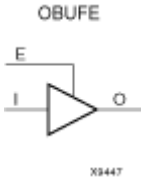
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFE

Macro: 3-State Output Buffer with Active-High Output Enable



Introduction

This design element is a 3-state buffer with input I, output O, and active-High output enable (E).

When E is High, data on the inputs of the buffers is transferred to the corresponding outputs. When E is Low, the output is High impedance (off or Z state). This design element isolates the internal circuit and provides drive current for signals leaving a chip. It is connected to an OPAD or an IOPAD, and its input is connected to the internal circuit.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

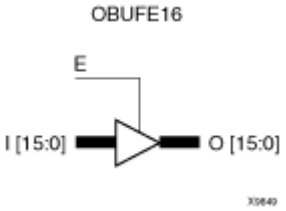
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFE16

Macro: 16-Bit 3-State Output Buffer with Active-High Output Enable



Introduction

This design element is a 3-state buffer with input I15-I0, output O15-O0, and active-High output enable (E).

When E is High, data on the inputs of the buffers is transferred to the corresponding outputs. When E is Low, the output is High impedance (off or Z state). This design element isolates the internal circuit and provides drive current for signals leaving a chip. It is connected to an OPAD or an IOPAD, and its input is connected to the internal circuit.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

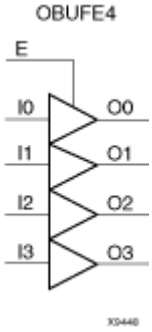
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFE4

Macro: 4-Bit 3-State Output Buffer with Active-High Output Enable



Introduction

This design element is a 3-state buffer with input I3-I0, output O3-O0, and active-High output enable (E).

When E is High, data on the inputs of the buffers is transferred to the corresponding outputs. When E is Low, the output is High impedance (off or Z state). This design element isolates the internal circuit and provides drive current for signals leaving a chip. It is connected to an OPAD or an IOPAD, and its input is connected to the internal circuit.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

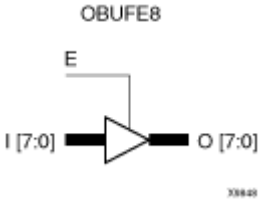
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFE8

Macro: 8-Bit 3-State Output Buffer with Active-High Output Enable



Introduction

This design element is a 3-state buffer with input I7-I0, output O7-O0, and active-High output enable (E).

When E is High, data on the inputs of the buffers is transferred to the corresponding outputs. When E is Low, the output is High impedance (off or Z state). This design element isolates the internal circuit and provides drive current for signals leaving a chip. It is connected to an OPAD or an IOPAD, and its input is connected to the internal circuit.

Logic Table

Inputs		Outputs
E	**I**	**O**
0	X	Z
1	1	1
1	0	0

Design Entry Method

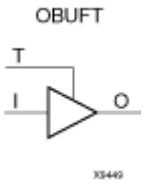
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFT

Primitive: 3-State Output Buffer with Active Low Output Enable



Introduction

This design element is a single, 3-state output buffer with input I, output O, and active-Low output enables (T). This element uses the LVTTL standard and has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

When T is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When T is High, the output is high impedance (off or Z state). OBUFTs are generally used when a single-ended output is needed with a 3-state capability, such as the case when building bidirectional I/O.

Logic Table

Inputs		Outputs
T	**I**	**O**
1	X	Z
0	I	F

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Buffer output (connect directly to top-level port)
I	Input	1	Buffer input
T	Input	1	3-state enable input

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

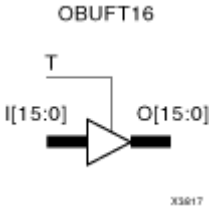
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

OBUFT16

Macro: 16-Bit 3-State Output Buffer with Active Low Output Enable



Introduction

This design element is a multiple, 3-state output buffer with input I, output O, and active-Low output enables (T). This element uses the LVTTL standard and has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

When T is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When T is High, the output is high impedance (off or Z state). OBUFTs are generally used when a single-ended output is needed with a 3-state capability, such as the case when building bidirectional I/O.

Logic Table

Inputs		Outputs
T	**I**	**O**
1	X	Z
0	I	F

Design Entry Method

This design element is only for use in schematics.

Available Attributes

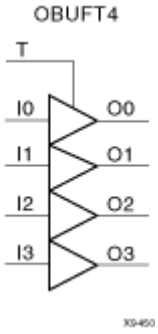
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFT4

Macro: 4-Bit 3-State Output Buffers with Active-Low Output Enable



Introduction

This design element is a multiple, 3-state output buffer with input I, output O, and active-Low output enables (T). This element uses the LVTTL standard and has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

When T is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When T is High, the output is high impedance (off or Z state). OBUFTs are generally used when a single-ended output is needed with a 3-state capability, such as the case when building bidirectional I/O.

Logic Table

Inputs		Outputs
T	**I**	**O**
1	X	Z
0	I	F

Design Entry Method

This design element is only for use in schematics.

Available Attributes

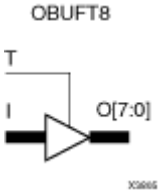
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFT8

Macro: 8-Bit 3-State Output Buffers with Active-Low Output Enable



Introduction

This design element is a multiple, 3-state output buffer with input I, output O, and active-Low output enables (T). This element uses the LVTTL standard and has selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

When T is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When T is High, the output is high impedance (off or Z state). OBUFTs are generally used when a single-ended output is needed with a 3-state capability, such as the case when building bidirectional I/O.

Logic Table

Inputs		Outputs
T	**I**	**O**
1	X	Z
0	I	F

Design Entry Method

This design element is only for use in schematics.

Available Attributes

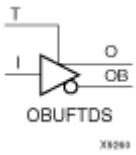
Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OBUFTDS

Primitive: 3-State Output Buffer with Differential Signaling, Active-Low Output Enable



Introduction

This design element is an output buffer that supports low-voltage, differential signaling. For the OBUFTDS, a design level interface signal is represented as two distinct ports (O and OB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET_P and MYNET_N).

Logic Table

Inputs		Outputs	
I	**T**	**O**	**OB**
X	1	Z	Z
0	0	0	1
1	0	1	0

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Diff_p output (connect directly to top level port)
OB	Output	1	Diff_n output (connect directly to top level port)
I	Input	1	Buffer input
T	Input	1	3-state enable input

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
IOSTANDARD	String	See Data Sheet	"DEFAULT"	Assigns an I/O standard to the element.

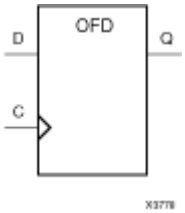
For More Information

- See the *Virtex-II Pro User Guide* .

- See the *Virtex-II Pro Data Sheets* .

OFD

Macro: Output D Flip-Flop



Introduction

This design element is a single output D flip-flop.

The outputs are connected to OPADs or IOPADs. The data on the (D) inputs is loaded into the flip-flops during the Low-to-High clock (C) transition and appears on the (Q) outputs.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↑	D

Design Entry Method

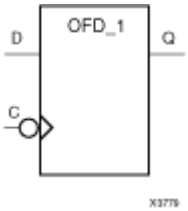
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFD_1

Macro: Output D Flip-Flop with Inverted Clock



Introduction

The design element is located in an input/output block (IOB). The output (Q) of the D flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the (Q) output.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↓	D

Design Entry Method

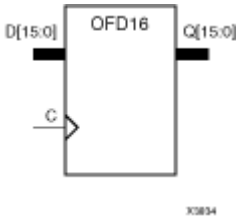
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFD16

Macro: 16-Bit Output D Flip-Flop



Introduction

This design element is a multiple output D flip-flop.

The outputs are connected to OPADs or IOPADs. The data on the (D) inputs is loaded into the flip-flops during the Low-to-High clock (C) transition and appears on the (Q) outputs.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↑	D

Design Entry Method

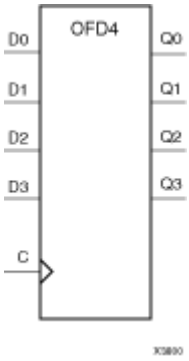
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFD4

Macro: 4-Bit Output D Flip-Flop



Introduction

This design element is a multiple output D flip-flop.

The outputs are connected to OPADs or IOPADs. The data on the (D) inputs is loaded into the flip-flops during the Low-to-High clock (C) transition and appears on the (Q) outputs.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↑	D

Design Entry Method

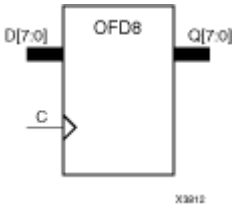
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFD8

Macro: 8-Bit Output D Flip-Flop



Introduction

This design element is a multiple output D flip-flop.

The outputs are connected to OPADs or IOPADs. The data on the (D) inputs is loaded into the flip-flops during the Low-to-High clock (C) transition and appears on the (Q) outputs.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↑	D

Design Entry Method

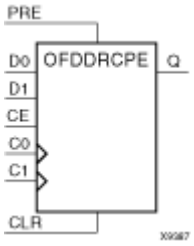
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDDRCPE

Primitive: Dual Data Rate Output D Flip-Flop with Clock Enable and Asynchronous Preset and Clear



Introduction

This design element is a dual data rate (DDR) output D flip-flop with clock enable (CE) and asynchronous preset (PRE) and clear (CLR). It consists of one output buffer and one dual data rate flip-flop (FDDRCPE). When the asynchronous PRE is High and CLR is Low, the Q output is preset High.

When CLR is High, Q is set Low. Data on the D0 input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High C0 clock transition. Data on the D1 input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High C1 clock transition. The INIT attribute does not apply to OFDDRCPE components.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs							Outputs
C0	**C1**	**CE**	**D0**	**D1**	**CLR**	**PRE**	**Q**
X	X	X	X	X	1	0	0
X	X	X	X	X	0	1	1
X	X	X	X	X	1	1	0
X	X	0	X	X	0	0	No Change
↑	X	1	D0	X	0	0	D0
X	↑	1	X	D1	0	0	D1

Design Entry Method

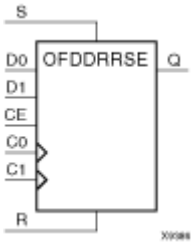
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDDRRSE

Primitive: Dual Data Rate Output D Flip-Flop with Synchronous Reset and Set and Clock Enable



Introduction

This design element is a dual data rate (DDR) output D flip-flop with synchronous reset (R) and set (S) and clock enable (CE). It consists of one output buffer and one dual data rate flip-flop (FDDRRSE).

On a Low-to-High clock transition (C0 or C1), a High R input resets the Q output Low; a Low R input with a High S input sets Q High. When both R and S are Low and clock enable is High, data on the D0 input is loaded into the flip-flop on a Low-to-High C0 clock transition and data on the D1 input is loaded into the flip-flop on a Low-to-High C1 clock transition.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

The INIT attribute does not apply to this design element.

Logic Table

Inputs							Outputs
C0	**C1**	**CE**	**D0**	**D1**	**R**	**S**	**Q**
↑	X	X	X	X	1	0	0
↑	X	X	X	X	0	1	1
↑	X	X	X	X	1	1	0
X	↑	X	X	X	1	0	0
X	↑	X	X	X	0	1	1
X	↑	X	X	X	1	1	0
X	X	0	X	X	0	0	No Change
↑	X	1	D0	X	0	0	D0
X	↑	1	X	D1	0	0	D1

Design Entry Method

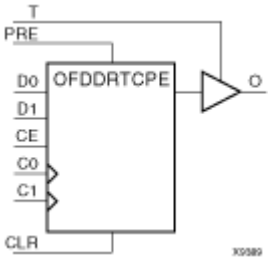
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDDRTCPE

Primitive: Dual Data Rate D Flip-Flop with Active-Low 3–State Output Buffer, Clock Enable, and Asynchro-nous Preset and Clear



Introduction

This design element is a dual data rate (DDR) D flip-flop with clock enable (CE) and asynchronous preset and clear whose output is enabled by a 3-state buffer. It consists of a dual data rate flip-flop (FDDRCPE) and a 3-state output buffer (OBUFT). The data output (O) of the flip-flop is connected to the input of the output buffer (OBUFT). The output of the OBUFT is connected to an OPAD or IOPAD.

When the active-Low enable input (T) is Low, output is enabled and the data on the flip-flop's Q output appears on the OBUFT's O output. When the asynchronous PRE is High and CLR is Low, the O output is preset High. When CLR is High, O is set Low. Data on the D0 input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High C0 clock transition. Data on the D1 input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High C1 clock transition.

When T is High, outputs are high impedance (Off). When CE is Low and T is Low, the outputs do not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

The INIT attribute does not apply to this design element.

Logic Table

Inputs								Outputs
C0	**C1**	**CE**	**D0**	**D1**	**CLR**	**PRE**	**T**	**O**
X	X	X	X	X	X	X	1	Z
X	X	X	X	X	1	0	0	0
X	X	X	X	X	0	1	0	1
X	X	X	X	X	1	1	0	0
X	X	0	X	X	0	0	0	No Change
↑	X	1	D0	X	0	0	0	D0
X	↑	1	X	D1	0	0	0	D1

Design Entry Method

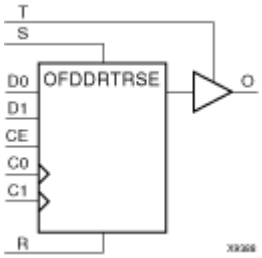
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDDRTRSE

Primitive: Dual Data Rate D Flip-Flop with Active -Low 3-State Output Buffer, Synchronous Reset and Set, and Clock Enable



Introduction

This design element is a dual data rate (DDR) D flip-flop with clock enable (CE) and synchronous reset and set whose output is enabled by a 3-state buffer. It consists of a dual data rate flip-flop (FDDRRSE) and a 3-state output buffer (OBUFT). The data output (O) of the flip-flop is connected to the input of the output buffer (OBUFT). The output of the OBUFT is connected to an OPAD or IOPAD.

When the active-Low enable input (T) is Low, output is enabled and the data on the flip-flop's Q output appears on the OBUFT's O output. On a Low-to-High clock transition (C0 or C1), a High R input resets the Q output Low; a Low R input with a High S input sets O High. When both R and S are Low and clock enable is High, data on the D0 input is loaded into the flip-flop on a Low-to-High C0 clock transition and data on the D1 input is loaded into the flip-flop on a Low-to-High C1 clock transition.

When T is High, outputs are high impedance (Off). When CE is Low and T is Low, the outputs do not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

The INIT attribute does not apply to this design elements components

Logic Table

Inputs								Outputs
C0	**C1**	**CE**	**D0**	**D1**	**R**	**S**	**T**	**O**
X	X	X	X	X	X	X	1	Z
↑	X	X	X	X	1	0	0	0
↑	X	X	X	X	0	1	0	1
↑	X	X	X	X	1	1	0	0
X	↑	X	X	X	1	0	0	0
X	↑	X	X	X	0	1	0	1
X	↑	X	X	X	1	1	0	0
X	X	0	X	X	0	0	0	No Change
↑	X	1	D0	X	0	0	0	D0
X	↑	1	X	D1	0	0	0	D1

Design Entry Method

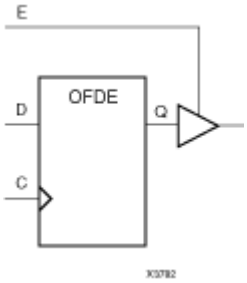
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDE

Macro: D Flip-Flop with Active-High Enable Output Buffers



Introduction

This is a single D flip-flop whose output is enabled by a 3-state buffer. The flip-flop data output (Q) is connected to the input of output buffer (OBUFE). The OBUFE output (O) is connected to an OPAD or IOPAD. The data on the data input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition. When the active-High enable input (E) is High, the data on the flip-flop output (Q) appears on the OBUFE (O) output. When (E) is Low, the output is high impedance (Z state or Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Output
E	**D**	**C**	**O**
0	X	X	Z
1	Dn	↑	Dn

Design Entry Method

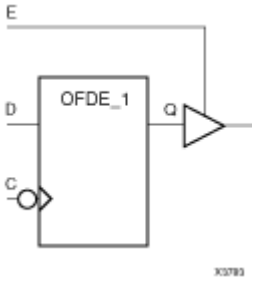
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDE_1

Macro: D Flip-Flop with Active-High Enable Output Buffer and Inverted Clock



Introduction

This design element and its output buffer are located in an input/output block (IOB). The data output of the flip-flop (Q) is connected to the input of an output buffer or OBUFE. The output of the OBUFE is connected to an OPAD or an IOPAD. The data on the data input (D) is loaded into the flip-flop on the High-to-Low clock (C) transition. When the active-High enable input (E) is High, the data on the flip-flop output (Q) appears on the (O) output. When (E) is Low, the output is high impedance (Z state or Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
E	**D**	**C**	**O**
0	X	X	Z
1	D	↓	D

Design Entry Method

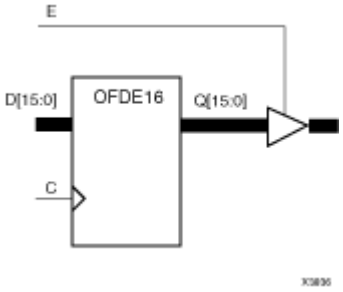
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDE16

Macro: 16-Bit D Flip-Flop with Active-High Enable Output Buffers



Introduction

This is a multiple D flip-flop whose outputs are enabled by 3-state buffers. The flip-flop data outputs (Q) are connected to the inputs of output buffers (OBUFE). The OBUFE outputs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-High enable inputs (E) are High, the data on the flip-flop outputs (Q) appears on the OBUFE outputs (O). When (E) is Low, outputs are high impedance (Z state or Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
E	**D**	**C**	**O**
0	X	X	Z
1	Dn	↑	Dn

Design Entry Method

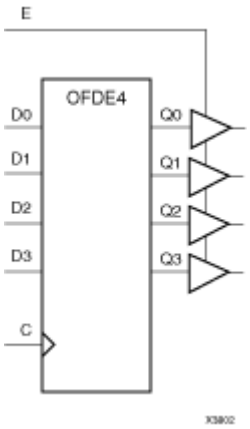
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDE4

Macro: 4-Bit D Flip-Flop with Active-High Enable Output Buffers



Introduction

This is a multiple D flip-flop whose outputs are enabled by 3-state buffers. The flip-flop data outputs (Q) are connected to the inputs of output buffers (OBUFE). The OBUFE outputs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-High enable inputs (E) are High, the data on the flip-flop outputs (Q) appears on the OBUFE outputs (O). When (E) is Low, outputs are high impedance (Z state or Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
E	**D**	**C**	**O**
0	X	X	Z
1	Dn	↑	Dn

Design Entry Method

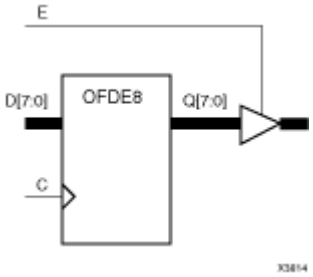
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDE8

Macro: 8-Bit D Flip-Flop with Active-High Enable Output Buffers



Introduction

This is a multiple D flip-flop whose outputs are enabled by 3-state buffers. The flip-flop data outputs (Q) are connected to the inputs of output buffers (OBUFE). The OBUFE outputs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-High enable inputs (E) are High, the data on the flip-flop outputs (Q) appears on the OBUFE outputs (O). When (E) is Low, outputs are high impedance (Z state or Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
E	**D**	**C**	**O**
0	X	X	Z
1	Dn	↑	Dn

Design Entry Method

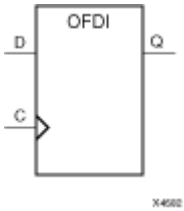
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDI

Macro: Output D Flip-Flop (Asynchronous Preset)



Introduction

The design element is contained in an input/output block (IOB). The output (Q) of the (D) flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q).

This flip-flop is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↑	D

Design Entry Method

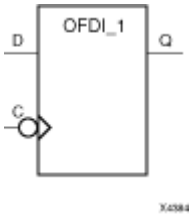
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDI_1

Macro: Output D Flip-Flop with Inverted Clock (Asynchronous Preset)



Introduction

This design element exists in an input/output block (IOB). The (D) flip-flop output (Q) is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the (Q) output.

This flip-flop is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs		Outputs
D	**C**	**Q**
D	↓	D

Design Entry Method

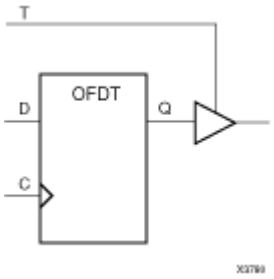
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDT

Macro: D Flip-Flop with Active-Low 3-State Output Buffer



Introduction

This design element is a single D flip-flops whose output is enabled by a 3-state buffer.

The data outputs (Q) of the flip-flops are connected to the inputs of output buffers (OBUFT). The outputs of the OBUFTs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-Low enable inputs (T) are Low, the data on the flip-flop outputs (Q) appears on the (O) outputs. When (T) is High, outputs are high impedance (Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
T	**D**	**C**	**O**
1	X	X	Z
0	D	↑	D

Design Entry Method

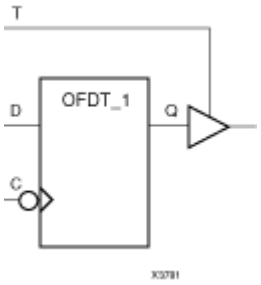
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDT_1

Macro: D Flip-Flop with Active-Low 3-State Output Buffer and Inverted Clock



Introduction

The design element and its output buffer are located in an input/output block (IOB). The flip-flop data output (Q) is connected to the input of an output buffer (OBUFT). The OBUFT output is connected to an OPAD or an IOPAD. The data on the data input (D) is loaded into the flip-flop on the High-to-Low clock (C) transition. When the active-Low enable input (T) is Low, the data on the flip-flop output (Q) appears on the (O) output. When (T) is High, the output is high impedance (Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
T	**D**	**C**	**O**
1	X	X	Z
0	D	↓	D

Design Entry Method

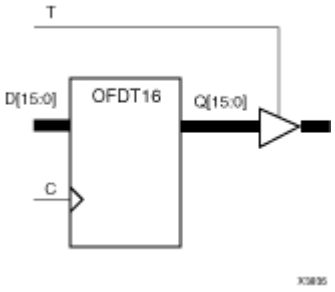
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDT16

Macro: 16-Bit D Flip-Flop with Active-Low 3-State Output Buffers



Introduction

This design element is a multiple D flip-flop whose output are enabled by 3-state buffers.

The data outputs (Q) of the flip-flops are connected to the inputs of output buffers (OBUFT). The outputs of the OBUFTs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-Low enable inputs (T) are Low, the data on the flip-flop outputs (Q) appears on the (O) outputs. When (T) is High, outputs are high impedance (Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
T	**D**	**C**	**O**
1	X	X	Z
0	D	↑	D

Design Entry Method

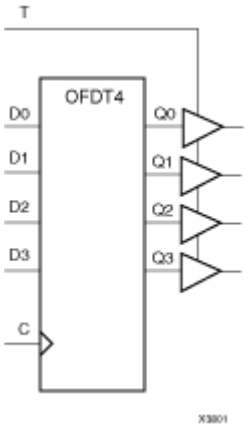
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDT4

Macro: 4-Bit D Flip-Flop with Active-Low 3-State Output Buffers



Introduction

This design element is a multiple D flip-flop whose output are enabled by 3-state buffers.

The data outputs (Q) of the flip-flops are connected to the inputs of output buffers (OBUFT). The outputs of the OBUFTs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-Low enable inputs (T) are Low, the data on the flip-flop outputs (Q) appears on the (O) outputs. When (T) is High, outputs are high impedance (Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
T	**D**	**C**	**O**
1	X	X	Z
0	D	↑	D

Design Entry Method

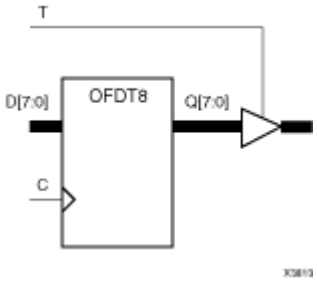
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDT8

Macro: 8-Bit D Flip-Flop with Active-Low 3-State Output Buffers



Introduction

This design element is a multiple D flip-flop whose output are enabled by 3-state buffers.

The data outputs (Q) of the flip-flops are connected to the inputs of output buffers (OBUFT). The outputs of the OBUFTs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-Low enable inputs (T) are Low, the data on the flip-flop outputs (Q) appears on the (O) outputs. When (T) is High, outputs are high impedance (Off).

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
T	**D**	**C**	**O**
1	X	X	Z
0	D	↑	D

Design Entry Method

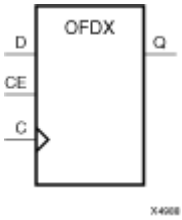
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDX

Macro: Output D Flip-Flop with Clock Enable



Introduction

This design element is a single output D flip-flop. The (Q) output is connected to OPAD or IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears on the (Q) output. When (CE) is Low, the flip-flop output does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CE	**D**	**C**	**Q**
1	Dn	↑	Dn
0	X	X	No change

Design Entry Method

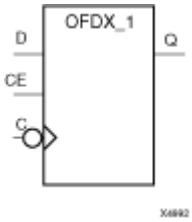
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDX_1

Macro: Output D Flip-Flop with Inverted Clock and Clock Enable



Introduction

The design element is located in an input/output block (IOB). The output (Q) of the (D) flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the (Q) output. When the (CE) pin is Low, the output (Q) does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**D**	**C**	**Q**
1	D	↓	D
0	X	X	No Change

Design Entry Method

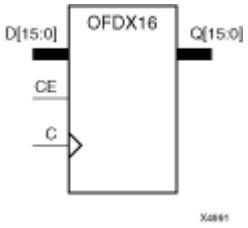
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDX16

Macro: 16-Bit Output D Flip-Flop with Clock Enable



Introduction

This design element is a multiple output D flip-flop. The (Q) output is connected to OPAD or IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears on the (Q) output. When (CE) is Low, the flip-flop output does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**D**	**C**	**Q**
1	Dn	↑	Dn
0	X	X	No change

Design Entry Method

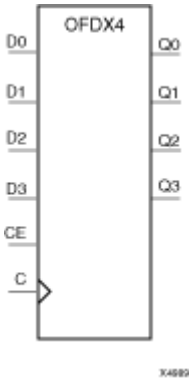
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDX4

Macro: 4-Bit Output D Flip-Flop with Clock Enable



Introduction

This design element is a multiple output D flip-flop. The (Q) output is connected to OPAD or IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears on the (Q) output. When (CE) is Low, the flip-flop output does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			**Outputs**
CE	**D**	**C**	**Q**
1	Dn	↑	Dn
0	X	X	No change

Design Entry Method

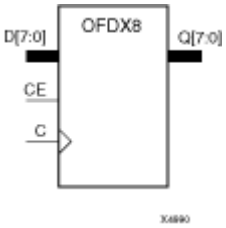
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDX8

Macro: 8-Bit Output D Flip-Flop with Clock Enable



Introduction

This design element is a multiple output D flip-flop. The (Q) output is connected to OPAD or IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears on the (Q) output. When (CE) is Low, the flip-flop output does not change.

This flip-flop is asynchronously cleared, outputs Low, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CE	**D**	**C**	**Q**
1	Dn	↑	Dn
0	X	X	No change

Design Entry Method

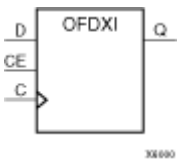
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDXI

Macro: Output D Flip-Flop with Clock Enable (Asynchronous Preset)



Introduction

The design element is contained in an input/output block (IOB). The output (Q) of the D flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). When(CE) is Low, the output does not change

This flip-flop is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CE	**D**	**C**	**Q**
1	D	↑	D
0	X	X	No Change

Design Entry Method

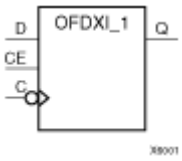
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OFDXI_1

Macro: Output D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)



Introduction

The design element is located in an input/output block (IOB). The D flip-flop output (Q) is connected to an OPAD or an IOPAD. The data on the D input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the Q output. When CE is Low, the output (Q) does not change.

This flip-flop is asynchronously preset, output High, when power is applied. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs			Outputs
CE	**D**	**C**	**Q**
1	D	↓	D
0	X	X	No Change

Design Entry Method

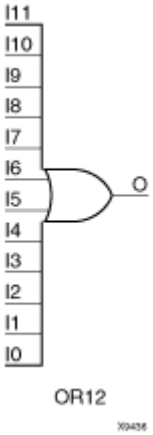
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR12

Macro: 12-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

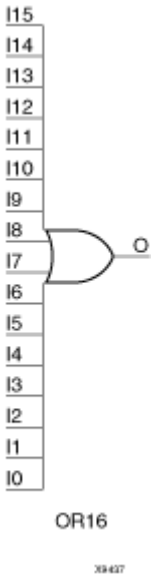
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR16

Macro: 16-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

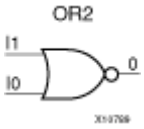
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR2

Primitive: 2-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

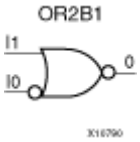
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR2B1

Primitive: 2-Input OR Gate with 1 Inverted and 1 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

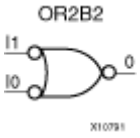
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR2B2

Primitive: 2-Input OR Gate with Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

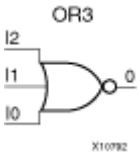
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR3

Primitive: 3-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

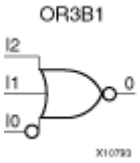
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR3B1

Primitive: 3-Input OR Gate with 1 Inverted and 2 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

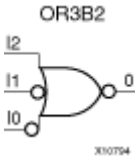
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR3B2

Primitive: 3-Input OR Gate with 2 Inverted and 1 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

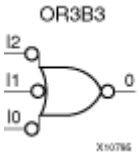
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR3B3

Primitive: 3-Input OR Gate with Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

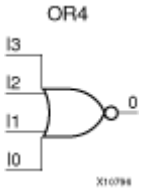
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR4

Primitive: 4-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

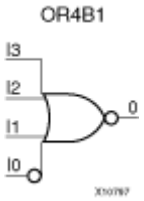
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR4B1

Primitive: 4-Input OR Gate with 1 Inverted and 3 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

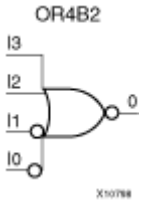
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR4B2

Primitive: 4-Input OR Gate with 2 Inverted and 2 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

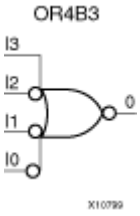
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR4B3

Primitive: 4-Input OR Gate with 3 Inverted and 1 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

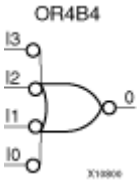
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR4B4

Primitive: 4-Input OR Gate with Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

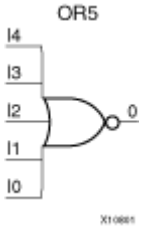
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR5

Primitive: 5-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

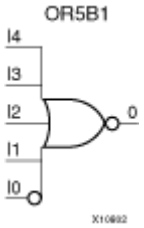
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR5B1

Primitive: 5-Input OR Gate with 1 Inverted and 4 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

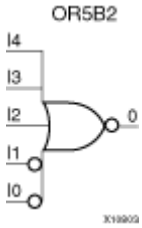
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR5B2

Primitive: 5-Input OR Gate with 2 Inverted and 3 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

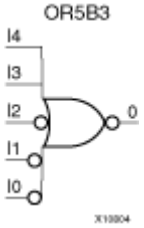
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR5B3

Primitive: 5-Input OR Gate with 3 Inverted and 2 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

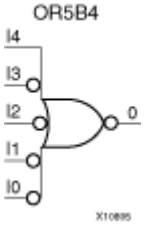
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR5B4

Primitive: 5-Input OR Gate with 4 Inverted and 1 Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

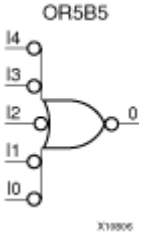
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR5B5

Primitive: 5-Input OR Gate with Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

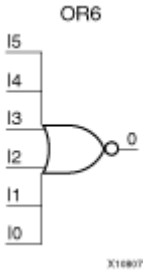
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR6

Macro: 6-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

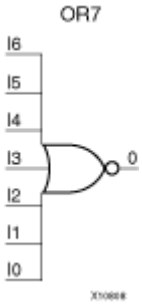
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR7

Macro: 7-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

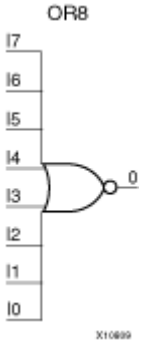
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR8

Macro: 8-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

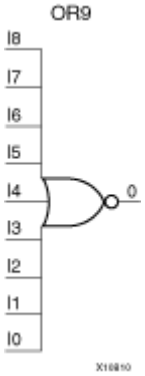
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

OR9

Macro: 9-Input OR Gate with Non-Inverted Inputs



Introduction

OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs, 12 inputs, and 16 inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

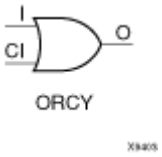
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ORCY

Primitive: OR with Carry Logic



Introduction

This element is a special OR with general O output that generates faster and smaller arithmetic functions.

Each Virtex-II, Virtex-II Pro, and Virtex-II Pro X slice contains a dedicated 2-input OR gate that ORs together carry out values for a series of horizontally adjacent carry chains. The OR gate gets one input external to the slice and the other input from the output of the high order carry mux. The OR gate's output drives the next slice's OR gate horizontally across the die.

Only MUXCY outputs can drive the signal on the CI pin. Only this design elements outputs or logic zero can drive the I pin.

Design Entry Method

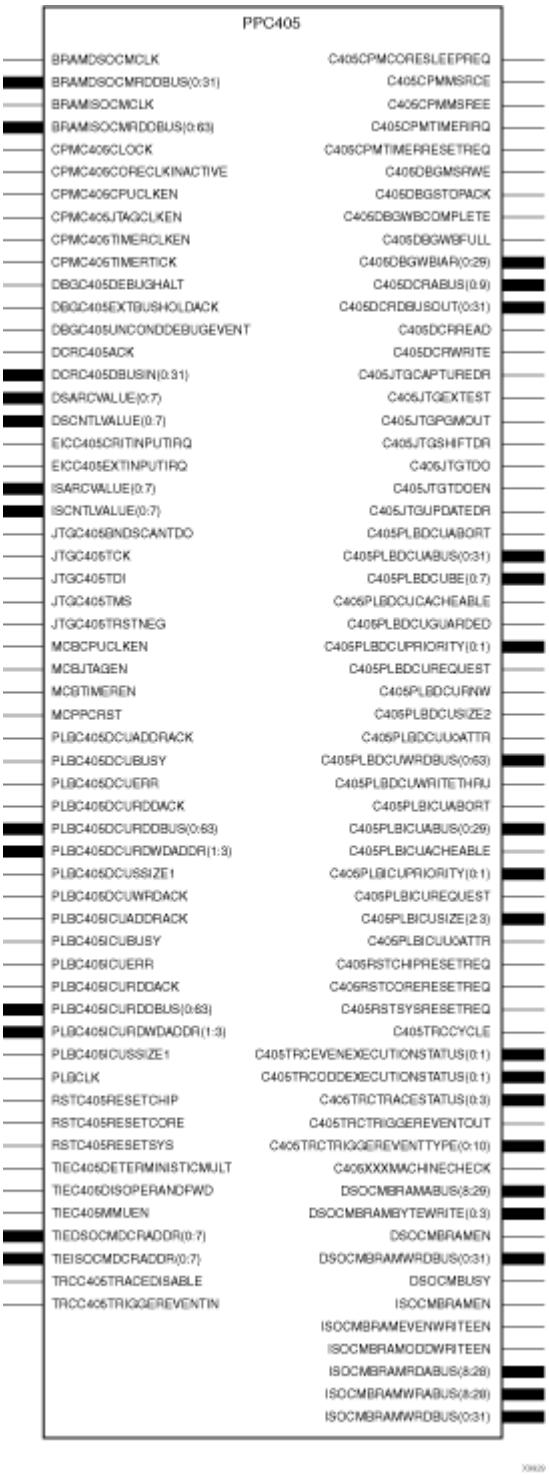
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

PPC405

Primitive: Primitive for the PowerPC® Core



Introduction

This design element is an embedded core that is a 32-bit RISC core integrating a PowerPC® 405 CPU, separate instruction and data caches, a JTAG port, trace FIFO, multiple timers, and a memory management unit (MMU). Integrated on-chip memory (OCM) controllers provide dedicated interfaces between Block SelectRAM memory and the processor core instruction and data paths for high-speed access. The PowerPC® 405 core implements the PowerPC® User Instruction Set.

Logic Table

Inputs	Outputs
BRAMDSOCMCLK	C405CPMCORESLEEPREQ
BRAMDSOCMRDDBUS [0:31]	C405CPMMSRCE
BRAMISOCMCLK	C405CPMMSREE
BRAMISOCMRDDBUS [0:63]	C405CPMTIMERIRQ
CPMC405CLOCK	C405CPMTIMERRESETREQ
CPMC405CORECLKINACTIVE	C405DBGMSRWE
CPMC405CPUCLKEN	C405DBGSTOPACK
CPMC405JTAGCLKEN	C405DBGWBCOMPLETE
CPMC405TIMERCLKEN	C405DBGWBFULL
CPMC405TIMERTICK	C405DBGWBIAR[0:29]
DBGC405DEBUGHALT	C405DCRABUS [0:9]
DBGC405EXTBUSHOLDACK	C405DCRDBUSOUT [0:31]
DBGC405UNCONDDEBUGEVENT	C405DCRREAD
DCRC405ACK	C405DCRWRITE
DCRC405DBUSIN [0:31]	C405JTGCAPTUREDR
DSARCVALUE [0:7]	C405JTGEXTEST
DSCNTLVALUE [0:7]	C405JTGPGMOUT
EICC405CRITINPUTIRQ	C405JTGSHIFTDR
EICC405EXTINPUTIRQ	C405JTGTDO
ISARCVALUE [0:7]	C405JTGTDOEN
ISCNTLVALUE [0:7]	C405JTGUPDATEDR
JTGC405BNDSCANTDO	C405PLBDCUABORT
JTGC405TCK	C405PLBDCUABUS [0:31]
JTGC405TDI	C405PLBDCUBE [0:7]
JTGC405TMS	C405PLBDCUCACHEABLE
JTGC405TRSTNEG	C405PLBDCUGUARDED
MCBCPUCLKEN	C405PLBDCUPRIORITY [0:1]
MCBJTAGEN	C405PLBDCUREQUEST
MCBTIMEREN	C405PLBDCURNW
MCPPCRST	C405PLBDCUSIZE2
PLBC405DCUADDRACK	C405PLBDCUU0ATTR

Inputs	Outputs
PLBC405DCUBUSY	C405PLBDCUWRDBUS [0:63]
PLBC405DCUERR	C405PLBDCUWRITETHRU
PLBC405DCURDDACK	C405PLBICUABORT
PLBC405DCURDDBUS [0:63]	C405PLBICUABUS [0:29]
PLBC405DCURDWDADDR [1:3]	C405PLBICUCACHEABLE
PLBC405DCUSSIZE1	C405PLBICUPRIORITY [0:1]
PLBC405DCUWRDACK	C405PLBICUREQUEST
PLBC405ICUADDRACK	C405PLBICUSIZE [2:3]
PLBC405ICUBUSY	C405PLBICUU0ATTR
PLBC405ICUERR	C405RSTCHIPRESETREQ
PLBC405ICURDDACK	C405RSTCORERESETREQ
PLBC405ICURDDBUS [0:63]	C405RSTSYSRESETREQ
PLBC405ICURDWDADDR [1:3]	C405TRCCYCLE
PLBC405ICUSSIZE1	C405TRCEVENEXECUTIONSTATUS [0:1]
PLBCLK	C405TRCODDEXECUTIONSTATUS [0:1]
RSTC405RESETCHIP	C405TRCTRACESTATUS [0:3]
RSTC405RESETCORE	C405TRCTRIGGEREVENTOUT
RSTC405RESETSYS	C405TRCTRIGGEREVENTTYPE [0:10]
TIEC405DETERMINISTICMULT	C405XXXMACHINECHECK
TIEC405DISOPERANDFWD	DSOCMBRAMABUS [8:29]
TIEC405MMUEN	DSOCMBRAMBYTEWRITE [0:3]
TIEDSOCMDCRADDR [0:7]	DSOCMBRAMEN
TIEISOCMDCRADDR [0:7]	DSOCMBRAMWRDBUS [0:31]
TRCC405TRACEDISABLE	DSOCMBUSY
TRCC405TRIGGEREVENTIN	ISOCMBRAMEN
	ISOCMBRAMEVENWRITEEN
	ISOCMBRAMODDWRITEEN
	ISOCMBRAMRDABUS [8:28]
	ISOCMBRAMWRABUS [8:28]
	ISOCMBRAMWRDBUS [0:31]

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

PULLDOWN

Primitive: Resistor to GND for Input Pads, Open-Drain, and 3-State Outputs

Introduction

This resistor element is connected to input, output, or bidirectional pads to guarantee a logic Low level for nodes that might float.

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Pulldown output (connect directly to top level port)

Design Entry Method

This design element can be used in schematics.

This element can be connected to a net in the following locations on a top-level schematic file:

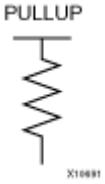
- A net connected to an input IO Marker
- A net connected to both an output IO Marker and 3-statable IO element, such as an OBUFT.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

PULLUP

Primitive: Resistor to VCC for Input PADs, Open-Drain, and 3-State Outputs



Introduction

This design element allows for an input, 3-state output or bi-directional port to be driven to a weak high value when not being driven by an internal or external source. This element establishes a High logic level for open-drain elements and macros when all the drivers are off.

Port Descriptions

Port	Direction	Width	Function
O	Output	1	Pullup output (connect directly to top level port)

Design Entry Method

This design element can be used in schematics.

This element can be connected to a net in the following locations on a top-level schematic file:

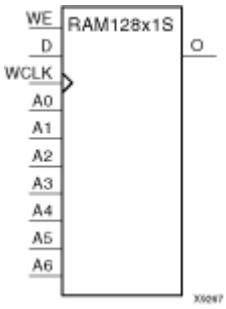
- A net connected to an input IO Marker
- A net connected to both an output IO Marker and 3-statable IO element, such as an OBUFT.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM128X1S

Primitive: 128-Deep by 1-Wide Static Synchronous RAM



Introduction

This element is a 128-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D) into the word selected by the 7-bit address (A6 - A0). This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM128X1S during configuration using the INIT attribute.

Logic Table

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D	D
1 (read)	↓	X	Data
Data = word addressed by bits A6 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

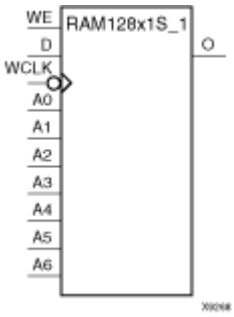
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 128-Bit Value	All zeros	Initializes ROMs, RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM128X1S_1

Primitive: 128-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



Introduction

This element is a 128-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any negative transition on WCLK loads the data on the data input (D) into the word selected by the 7-bit address (A6 – A0). For predictable performance, address and data inputs must be stable before a High-to-Low WCLK transition. This RAM block assumes an active-Low WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM128X1S_1 during configuration using the INIT attribute.

Logic Table

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data
Data = word addressed by bits A6 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

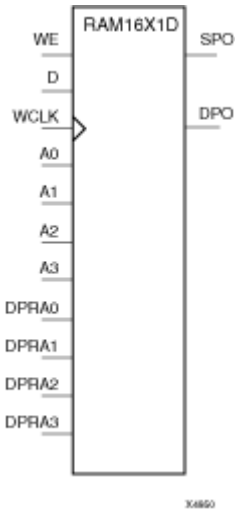
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 128-Bit Value	All zeros	Initializes ROMs, RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X1D

Primitive: 16-Deep by 1-Wide Static Dual Port Synchronous RAM



Introduction

This element is a 16-word by 1-bit static dual port random access memory with synchronous write capability. The device has two address ports: the read address (DPRA3–DPRA0) and the write address (A3–A0). These two address ports are asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected.

When WE is High, any positive transition on (WCLK) loads the data on the data input (D) into the word selected by the 4-bit write address. For predictable performance, write address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK). (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The SPO output reflects the data in the memory cell addressed by A3 – A0. The DPO output reflects the data in the memory cell addressed by DPRA3–DPRA0.

Note The write process is not affected by the address on the read address port.

You can use the INIT attribute to directly specify an initial value. The value must be a hexadecimal number, for example, INIT=ABAC. If the INIT attribute is not specified, the RAM is initialized with all zeros.

Logic Table

Mode selection is shown in the following logic table:

Inputs			Outputs	
WE (mode)	**WCLK**	**D**	**SPO**	**DPO**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↑	D	D	data_d

Inputs			Outputs	
WE (mode)	**WCLK**	**D**	**SPO**	**DPO**
1 (read)	↓	X	data_a	data_d
data_a = word addressed by bits A3-A0 data_d = word addressed by bits DPRA3-DPRA0				

Design Entry Method

This design element can be used in schematics.

Available Attributes

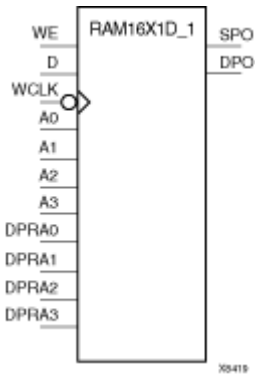
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros.	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X1D_1

Primitive: 16-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock



Introduction

This is a 16-word by 1-bit static dual port random access memory with synchronous write capability and negative-edge clock. The device has two separate address ports: the read address (DPRA3–DPRA0) and the write address (A3–A0). These two address ports are asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction.

When the write enable (WE) is set to Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 4-bit write address. For predictable performance, write address and data inputs must be stable before a High-to-Low WCLK transition. This RAM block assumes an active-High (WCLK). (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

You can initialize RAM16X1D_1 during configuration using the INIT attribute.

The SPO output reflects the data in the memory cell addressed by A3–A0. The DPO output reflects the data in the memory cell addressed by DPRA3–DPRA0.

Note The write process is not affected by the address on the read address port.

Logic Table

Mode selection is shown in the following logic table:

Inputs			Outputs	
WE (mode)	**WCLK**	**D**	**SPO**	**DPO**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↓	D	D	data_d
1 (read)	↑	X	data_a	data_d
data_a = word addressed by bits A3 – A0 data_d = word addressed by bits DPRA3-DPRA0				

Port Descriptions

Port	Direction	Width	Function
DPO	Output	1	Read-only 1-Bit data output
SPO	Output	1	R/W 1-Bit data output
A0	Input	1	R/W address[0] input
A1	Input	1	R/W address[1] input
A2	Input	1	R/W address[2] input
A3	Input	1	R/W address[3] input
D	Input	1	Write 1-Bit data input
DPRA0	Input	1	Read-only address[0] input
DPRA1	Input	1	Read-only address[1] input
DPRA2	Input	1	Read-only address[2] input
DPRA3	Input	1	Read-only address[3] input
WCLK	Input	1	Write clock input
WE	Input	1	Write enable input

Design Entry Method

This design element can be used in schematics.

Available Attributes

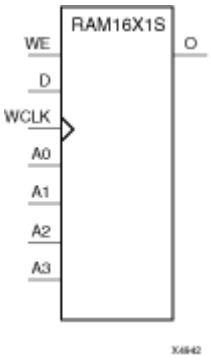
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X1S

Primitive: 16-Deep by 1-Wide Static Synchronous RAM



Introduction

This element is a 16-word by 1-bit static random access memory with synchronous write capability. When the write enable (WE) is set Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is set High, any positive transition on WCLK loads the data on the data input (D) into the word selected by the 4-bit address (A3 – A0). This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins. You can initialize RAM16X1S during configuration using the INIT attribute.

Logic Table

Inputs			Outputs
WE(mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D	D
1 (read)	↓	X	Data
Data = word addressed by bits A3 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

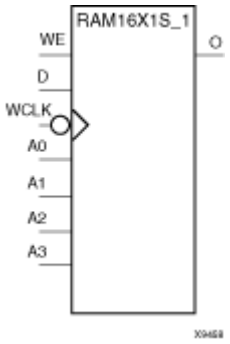
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Specifies initial contents of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X1S_1

Primitive: 16-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



Introduction

This element is a 16-word by 1-bit static random access memory with synchronous write capability and negative-edge clock. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 4-bit address (A3 – A0). For predictable performance, address and data inputs must be stable before a High-to-Low WCLK transition. This RAM block assumes an active-Low (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize this element during configuration using the INIT attribute.

Logic Table

Inputs			Outputs
WE(mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data
Data = word addressed by bits A3 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

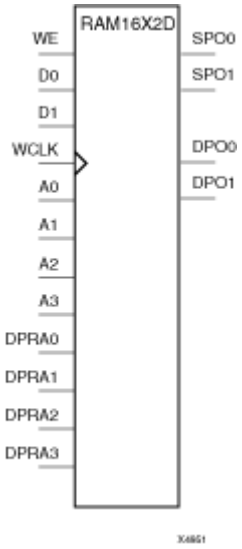
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Specifies initial contents of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X2D

Macro: 16-Deep by 2-Wide Static Dual Port Synchronous RAM



Introduction

This element is a 16-word by 2-bit static dual port random access memory with synchronous write capability. The device has two separate address ports: the read address (DPRA3 – DPRA0) and the write address (A3 – A0). These two address ports are completely asynchronous. The read address controls the location of data driven out of the output pin (DPO1 – DPO0), and the write address controls the destination of a valid write transaction.

When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D1 – D0) into the word selected by the 4-bit write address. For predictable performance, write address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The initial contents of this element cannot be specified directly.

The SPO output reflects the data in the memory cell addressed by A3 – A0. The DPO output reflects the data in the memory cell addressed by DPRA3 – DPRA0.

Note The write process is not affected by the address on the read address port.

Logic Table

Inputs			Outputs	
WE (mode)	**WCLK**	**D1-D0**	**SPO1-SPO0**	**DPO1-DPO0**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↑	D1 – D0	D1 – D0	data_d

Inputs			Outputs	
WE (mode)	**WCLK**	**D1-D0**	**SPO1-SPO0**	**DPO1-DPO0**
1 (read)	↓	X	data_a	data_d
data_a = word addressed by bits A3 – A0 data_d = word addressed by bits DPRA3-DPRA0				

Design Entry Method

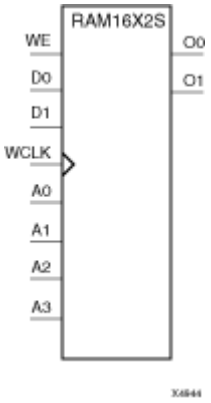
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X2S

Primitive: 16-Deep by 2-Wide Static Synchronous RAM



Introduction

This element is a 16-word by 2-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D1–D0) into the word selected by the 4-bit address (A3–A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O1–O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can use the INIT_xx properties to specify the initial contents of a Virtex-4 wide RAM. INIT_00 initializes the RAM cells corresponding to the O0 output, INIT_01 initializes the cells corresponding to the O1 output, etc. For example, a RAM16X2S instance is initialized by INIT_00 and INIT_01 containing 4 hex characters each. A RAM16X8S instance is initialized by eight properties INIT_00 through INIT_07 containing 4 hex characters each. A RAM64x2S instance is completely initialized by two properties INIT_00 and INIT_01 containing 16 hex characters each.

Except for Virtex-4 devices, the initial contents of this element cannot be specified directly.

Logic Table

Inputs			**Outputs**
WE (mode)	**WCLK**	**D1-D0**	**O1-O0**
0 (read)	X	X	Data
1(read)	0	X	Data
1(read)	1	X	Data
1(write)	↑	D1-D0	D1-D0
1 (read)	↓	X	Data
Data = word addressed by bits A3 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

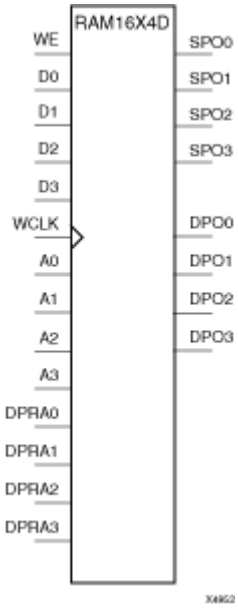
Attribute	Type	Allowed Values	Default	Description
INIT_00 to INIT_01	Hexadecimal	Any 16-Bit Value	All zeros	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X4D

Macro: 16-Deep by 4-Wide Static Dual Port Synchronous RAM



Introduction

This element is a 16-word by 4-bit static dual port random access memory with synchronous write capability. The device has two separate address ports: the read address (DPRA3 – DPRA0) and the write address (A3 – A0). These two address ports are completely asynchronous. The read address controls the location of data driven out of the output pin (DPO3 – DPO0), and the write address controls the destination of a valid write transaction.

When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D3 – D0) into the word selected by the 4-bit write address. For predictable performance, write address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The SPO output reflects the data in the memory cell addressed by A3 – A0. The DPO output reflects the data in the memory cell addressed by DPRA3 – DPRA0.

Note The write process is not affected by the address on the read address port.

The initial contents of this element cannot be specified directly.

Logic Table

Inputs			Outputs	
WE (mode)	**WCLK**	**D3-D0**	**SPO3-SPO0**	**DPO3-DPO0**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↑	D3-D0	D3-D0	data_d

Inputs			Outputs	
WE (mode)	**WCLK**	**D3-D0**	**SPO3-SPO0**	**DPO3-DPO0**
1 (read)	↓	X	data_a	data_d
data_a = word addressed by bits A3-A0 data_d = word addressed by bits DPRA3-DPRA0				

Design Entry Method

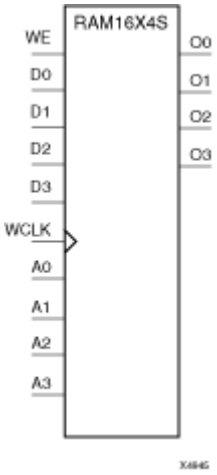
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X4S

Primitive: 16-Deep by 4-Wide Static Synchronous RAM



Introduction

This element is a 16-word by 4-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D3 – D0) into the word selected by the 4-bit address (A3 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O3 – O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

Logic Table

Inputs			Outputs
WE (mode)	**WCLK**	**D3 – D0**	**O3 – O0**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D3-D0	D3-D0
1 (read)	↓	X	Data
Data = word addressed by bits A3 – A0.			

Design Entry Method

This design element can be used in schematics.

Available Attributes

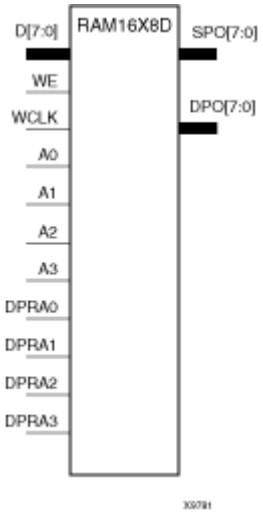
Attribute	Type	Allowed Values	Default	Description
INIT_00 to INIT_03	Hexadecimal	Any 16-Bit Value	All zeros	INIT for bit 0 of RAM

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X8D

Macro: 16-Deep by 8-Wide Static Dual Port Synchronous RAM



Introduction

This element is a 16-word by 8-bit static dual port random access memory with synchronous write capability. The device has two separate address ports: the read address (DPRA3 – DPRA0) and the write address (A3 – A0). These two address ports are completely asynchronous. The read address controls the location of data driven out of the output pin (DPO7 – DPO0), and the write address controls the destination of a valid write transaction. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected.

When WE is High, any positive transition on WCLK loads the data on the data input (D7 – D0) into the word selected by the 4-bit write address (A3 – A0). For predictable performance, write address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The SPO output reflects the data in the memory cell addressed by A3 – A0. The DPO output reflects the data in the memory cell addressed by DPRA3 – DPRA0.

The write process is not affected by the address on the read address port.

The initial contents of this element cannot be specified directly.

Logic Table

Inputs			Outputs	
WE (mode)	**WCLK**	**D7-D0**	**SP7-SPO0**	**DPO7-DPO0**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↑	D7-D0	D7-D0	data_d

Inputs			Outputs	
WE (mode)	**WCLK**	**D7-D0**	**SP7-SPO0**	**DPO7-DPO0**
1 (read)	↓	X	data_a	data_d
data_a = word addressed by bits A3-A0 data_d = word addressed by bits DPRA3-DPRA0				

Design Entry Method

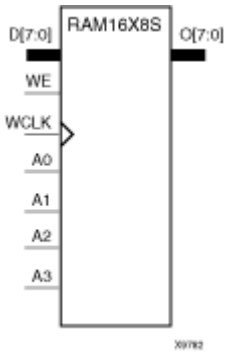
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM16X8S

Primitive: 16-Deep by 8-Wide Static Synchronous RAM



Introduction

This element is a 16-word by 8-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on data inputs (D7–D0) into the word selected by the 4-bit address (A3–A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O7–O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

Logic Table

Inputs			Outputs
WE (mode)	**WCLK**	**D7-D0**	**O7-O0**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D7-D0	D7-D0
1 (read)	↓	X	Data
Data = word addressed by bits A3–A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

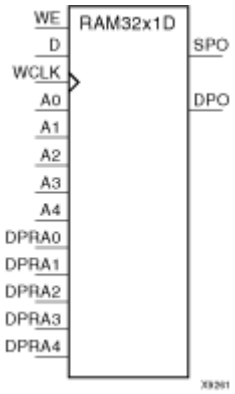
Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_07	Hexadecimal	Any 16-Bit Value	All zeros	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM32X1D

Primitive: 32-Deep by 1-Wide Dual Static Port Synchronous RAM



Introduction

The design element is a 32-word by 1-bit static dual port random access memory with synchronous write capability. The device has two separate address ports: the read address (DPRA4 – DPRA0) and the write address (A4 – A0). These two address ports are completely asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D) into the word selected by the 5-bit write address. For predictable performance, write address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block. You can initialize RAM32X1D during configuration using the INIT attribute. Mode selection is shown in the following logic table.

The SPO output reflects the data in the memory cell addressed by A4 – A0. The DPO output reflects the data in the memory cell addressed by DPRA4 – DPRA0. The write process is not affected by the address on the read address port.

Logic Table

Inputs			Outputs	
WE (Mode)	**WCLK**	**D**	**SPO**	**DPO**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↑	D	D	data_d
1 (read)	↓	X	data_a	data_d

Design Entry Method

This design element can be used in schematics.

Available Attributes

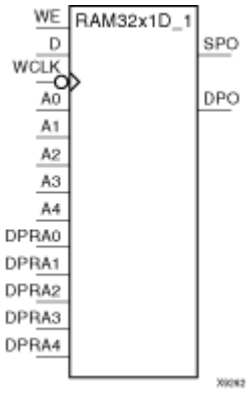
Attribute	Type	Allowed Values	Default	Descriptions
INIT	Hexadecimal	Any 32-Bit Value	All Zeros	Initializes ROMs, RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM32X1D_1

Primitive: 32-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock



Introduction

This design element is a 32-word by 1-bit static dual port random access memory with synchronous write capability and a negative-edge clock. The device has two separate address ports: the read address (DPRA4 – DPRA0) and the write address (A4 – A0). These two address ports are completely asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction.

When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any negative transition on WCLK loads the data on the data input (D) into the word selected by the 5-bit write address. For predictable performance, write address and data inputs must be stable before a High-to-Low WCLK transition. This RAM block assumes an active-Low WCLK. WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

You can initialize this design element during configuration using the INIT attribute.

The SPO output reflects the data in the memory cell addressed by A4 – A0. The DPO output reflects the data in the memory cell addressed by DPRA4 – DPRA0.

Note The write process is not affected by the address on the read address port.

Logic Table

Inputs			Outputs	
WE (mode)	**WCLK**	**D**	**SPO**	**DPO**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↓	D	D	data_d
1 (read)	↑	X	data_a	data_d
data_a = word addressed by bits A4-A0 data_d = word addressed by bits DPRA4-DPRA0				

Design Entry Method

This design element is only for use in schematics.

Available Attributes

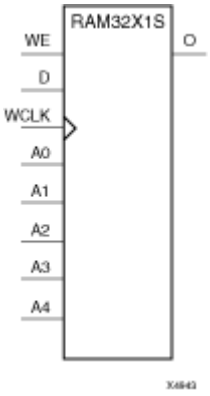
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 32-Bit Value	All zeros	Initializes ROMs, RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM32X1S

Primitive: 32-Deep by 1-Wide Static Synchronous RAM



Introduction

The design element is a 32-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any positive transition on (WCLK) loads the data on the data input (D) into the word selected by the 5-bit address (A4-A0). For predictable performance, address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins. You can initialize RAM32X1S during configuration using the INIT attribute.

Logic Table

Inputs			Outputs
WE (Mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data

Design Entry Method

This design element can be used in schematics.

Available Attributes

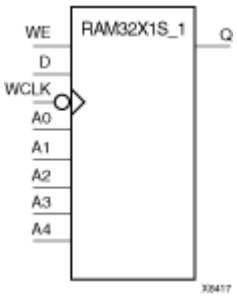
Attribute	Type	Allowed Values	Default	Descriptions
INIT	Hexadecimal	Any 32-Bit Value	All zeros	Specifies initial contents of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM32X1S_1

Primitive: 32-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



Introduction

The design element is a 32-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 5-bit address (A4-A0). For predictable performance, address and data inputs must be stable before a High-to-Low (WCLK) transition. This RAM block assumes an active-Low (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins. You can initialize RAM32X1S_1 during configuration using the INIT attribute.

Logic Table

Inputs			Outputs
WE (Mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data
Data = word addressed by bits A4 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

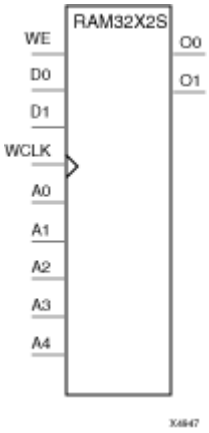
Attribute	Type	Allowed Values	Default	Descriptions
INIT	Hexadecimal	Any 32-Bit Value	0	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM32X2S

Primitive: 32-Deep by 2-Wide Static Synchronous RAM



Introduction

The design element is a 32-word by 2-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any positive transition on (WCLK) loads the data on the data input (D1-D0) into the word selected by the 5-bit address (A4-A0). For predictable performance, address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block. The signal output on the data output pins (O1-O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can use the INIT_00 and INIT_01 properties to specify the initial contents of RAM32X2S.

Logic Table

Inputs			Outputs
WE (Mode)	**WCLK**	**D**	**O0-O1**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D1-D0	D1-D0
1 (read)	↓	X	Data
Data = word addressed by bits A4 A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

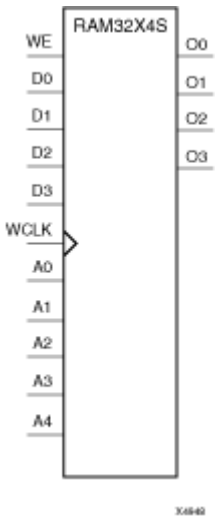
Attribute	Type	Allowed Values	Default	Descriptions
INIT_00	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 0 of RAM.
INIT_01	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 1 of RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM32X4S

Primitive: 32-Deep by 4-Wide Static Synchronous RAM



Introduction

This design element is a 32-word by 4-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data inputs (D3-D0) into the word selected by the 5-bit address (A4-A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O3-O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

Logic Table

Inputs			Outputs
WE	**WCLK**	**D3-D0**	**O3-O0**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D3-D0	D3-D0
1 (read)	↓	X	Data
Data = word addressed by bits A4-A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

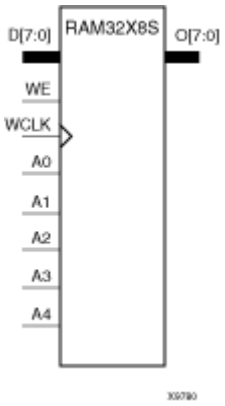
Attribute	Type	Allowed Values	Default	Description
INIT_00	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 0 of RAM.
INIT_01	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 1 of RAM.
INIT_02	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 2 of RAM.
INIT_03	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 3 of RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM32X8S

Primitive: 32-Deep by 8-Wide Static Synchronous RAM



Introduction

This design element is a 32-word by 8-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data inputs (D7 – D0) into the word selected by the 5-bit address (A4 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O7 – O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

Logic Table

Inputs			Outputs
WE (mode)	**WCLK**	**D7-D0**	**O7-O0**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D7-D0	D7-D0
1 (read)	↓	X	Data
Data = word addressed by bits A4 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 0 of RAM.

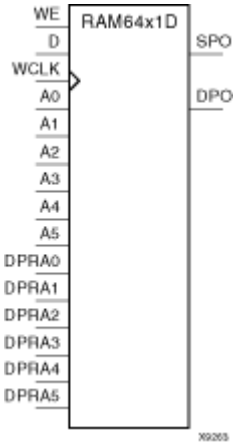
Attribute	Type	Allowed Values	Default	Description
INIT_01	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 1 of RAM.
INIT_02	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 2 of RAM.
INIT_03	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 3 of RAM.
INIT_04	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 4 of RAM.
INIT_05	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 5 of RAM.
INIT_06	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 6 of RAM.
INIT_07	Hexadecimal	Any 32-Bit Value	All zeros	INIT for bit 7 of RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM64X1D

Primitive: 64-Deep by 1-Wide Dual Port Static Synchronous RAM



Introduction

This design element is a 64-word by 1-bit static dual port random access memory with synchronous write capability. The device has two separate address ports: the read address (DPRA5–DPRA0) and the write address (A5–A0). These two address ports are completely asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected.

When WE is High, any positive transition on WCLK loads the data on the data input (D) into the word selected by the 6-bit (A0–A5) write address. For predictable performance, write address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The SPO output reflects the data in the memory cell addressed by A5–A0. The DPO output reflects the data in the memory cell addressed by DPRA5–DPRA0.

Note The write process is not affected by the address on the read address port.

Logic Table

Inputs			Outputs	
WE (mode)	**WCLK**	**D**	**SPO**	**DPO**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↑	D	D	data_d
1 (read)	↓	X	data_a	data_d
data_a = word addressed by bits A5–A0				
data_d = word addressed by bits DPRA5–DPRA0				

Design Entry Method

This design element can be used in schematics.

Available Attributes

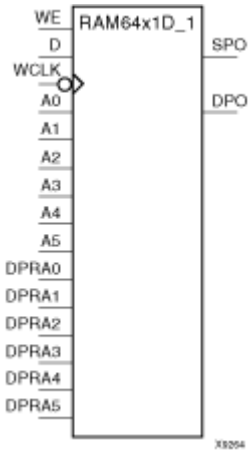
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 64-Bit Value	All zeros	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM64X1D_1

Primitive: 64-Deep by 1-Wide Dual Port Static Synchronous RAM with Negative-Edge Clock



Introduction

This design element is a 64-word by 1-bit static dual port random access memory with synchronous write capability and a negative-edge clock. The device has two separate address ports: the read address (DPRA5–DPRA0) and the write address (A5–A0). These two address ports are completely asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction.

When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any negative transition on WCLK loads the data on the data input (D) into the word selected by the 6-bit (A0–A5) write address. For predictable performance, write address and data inputs must be stable before a High-to-Low WCLK transition. This RAM block assumes an active-Low WCLK. WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

You can initialize this design element during configuration using the INIT attribute. The SPO output reflects the data in the memory cell addressed by A5–A0. The DPO output reflects the data in the memory cell addressed by DPRA5–DPRA0.

Note The write process is not affected by the address on the read address port.

Logic Table

Inputs			Outputs	
WE (mode)	**WCLK**	**D**	**SPO**	**DPO**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↓	D	D	data_d
1 (read)	↑	X	data_a	data_d
data_a = word addressed by bits A5–A0 data_d = word addressed by bits DPRA5–DPRA0				

Design Entry Method

This design element can be used in schematics.

Available Attributes

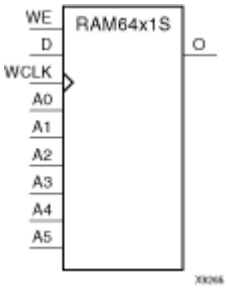
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 64-Bit Value	All zeros	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM64X1S

Primitive: 64-Deep by 1-Wide Static Synchronous RAM



Introduction

This design element is a 64-word by 1-bit static random access memory (RAM) with synchronous write capability. When the write enable is set Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is set High, any positive transition on WCLK loads the data on the data input (D) into the word selected by the 6-bit address (A5 - A0). This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize this element during configuration using the INIT attribute.

Logic Table

Mode selection is shown in the following logic table

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D	D
1 (read)	↓	X	Data
Data = word addressed by bits A5 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

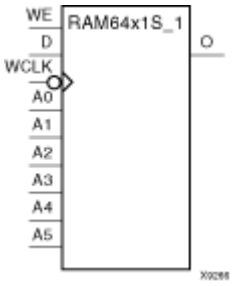
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 64-Bit Value	All zeros	Initializes ROMs, RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM64X1S_1

Primitive: 64-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



Introduction

This design element is a 64-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 6-bit address (A5 – A0). For predictable performance, address and data inputs must be stable before a High-to-Low (WCLK) transition. This RAM block assumes an active-Low (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize this element during configuration using the INIT attribute.

Logic Table

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data
Data = word addressed by bits A5 – A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

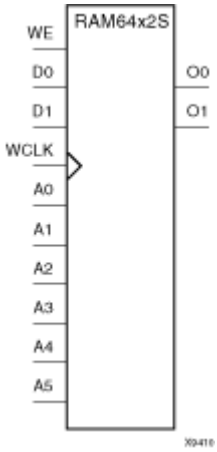
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 64-Bit Value	All zeros	Initializes ROMs, RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAM64X2S

Primitive: 64-Deep by 2-Wide Static Synchronous RAM



Introduction

This design element is a 64-word by 2-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D1–D0) into the word selected by the 6-bit address (A5–A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O1–O0) is the data that is stored in the RAM at the location defined by the values on the address pins. You can use the INIT_00 and INIT_01 properties to specify the initial contents of this design element.

Logic Table

Inputs			Outputs
WE (mode)	WCLK	D0–D1	O0–O1
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D1-D0	D1-D0
1 (read)	↓	X	Data
Data = word addressed by bits A5–A0			

Design Entry Method

This design element can be used in schematics.

Available Attributes

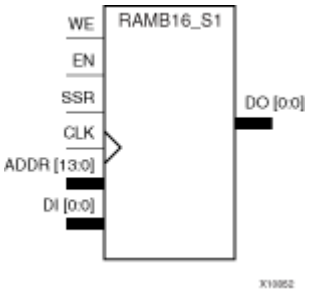
Attribute	Type	Allowed Values	Default	Description
INIT_00	Hexadecimal	Any 64-Bit Value	All zeros	Initializes RAMs, registers, and look-up tables.
INIT_01	Hexadecimal	Any 64-Bit Value	All zeros	Initializes RAMs, registers, and look-up tables.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S1

Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 1-bit Port



Introduction

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Data Cells		Parity Cells				
Depth	**Width**	**Depth**	**Width**	**Address Bus**	**Data Bus**	**Parity Bus**
16384	1	-	-	(13:0)	(0:0)	-

The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The above description assumes an active High EN, WE, SSR, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Logic Table

Inputs								Outputs			
GSR	**EN**	**SSR**	**WE**	**CLK**	**ADDR**	**DI**	**DIP**	**DO**	**DOP**	**RAM Contents**	
										Data RAM	**Parity RAM**
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Initialization

Initializing Memory Contents

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16 is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

Any INIT_xx or INITP_xx attribute that is not specified is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register

In Spartan-3A, each bit in the output register can be initialized at power on to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Two types of properties control initialization of the output register for a single-port RAMB16: INIT and SRVAL. The INIT attribute specifies the output register value at power on. You can use the SRVAL attribute to define the state resulting from assertion of the SSR (set/reset) input.

The INIT and SRVAL attributes specify the initialization value as a hexadecimal String containing one bit for each bit in the output port. For example, for a RAMB16_S1 with port width equal to 1, the output register contains 1 bit. Therefore, the INIT or SRVAL value can only be specified as a 1 or 0. For RAMB16_S4 with port width equal to 4, the output register contains 4 bits. In this case, you can specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT or SRVAL value.

Selecting Write Mode

The WRITE_MODE attribute controls RAMB16 memory and output contents. By default, the WRITE_MODE is set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the WRITE_MODE to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the WRITE_MODE to NO_CHANGE to have the input written to memory without changing the output.

Design Entry Method

This design element can be used in schematics.

Available Attributes

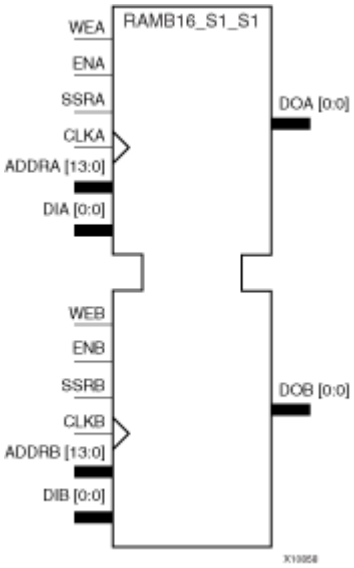
Attribute	Type	Allowed Values	Default	Description
INIT	Binary/ Hexadecimal	Any Hex Value	All zeros	Identifies the initial value of the DO output port after completing configuration. The bit width is dependent on the width of the A or B port of the RAM.
INIT_00 - INIT_3F	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the data portion of the RAM array.
INITP_00 - INITP_07	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SRVAL	Binary/ Hexadecimal	Any Hex Value	All zeros	Allows the individual selection of whether the DO output port sets (go to a one) or reset (go to a zero) upon the assertion of the SSR pin. The bit width is dependent on the width of the A or B port of the RAM.
WRITE_MODE	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DO port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S1_S1

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Component	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S1_S1	16384 x 1	-	(13:0)	(0:0)	-	16384 x 1	-	(13:0)	(0:0)	-

(a) Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.

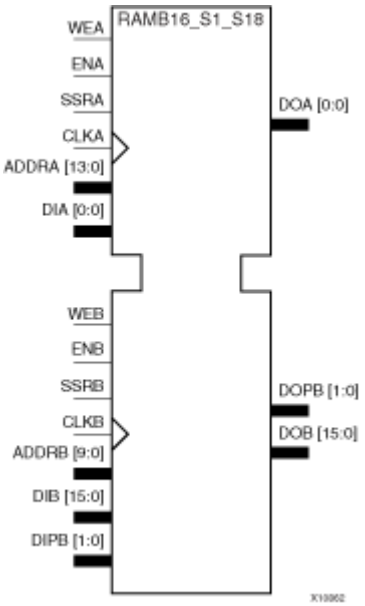
Attribute	Type	Allowed Values	Default	Description
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S1_S18

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 18-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Component	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S1_S18	16384 x 1	-	(13:0)	(0:0)	-	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a) Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

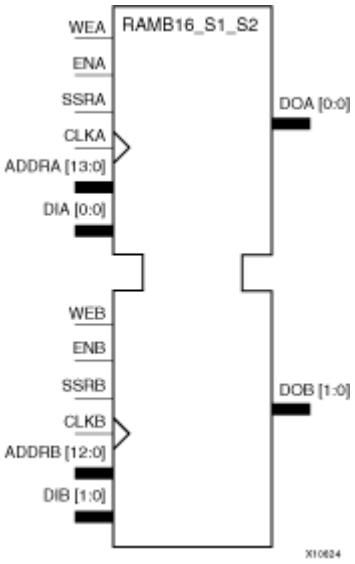
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S1_S2

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 2-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S1_S2	16384 x 1	-	(13:0)	(0:0)	-	8192 x 2	-	(12:0)	(1:0)	-

(a)Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables show address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.

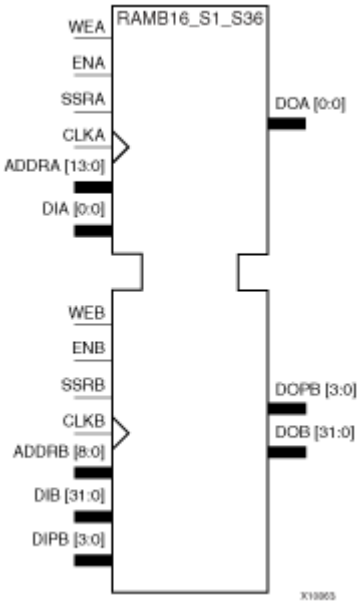
Attribute	Type	Allowed Values	Default	Description
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S1_S36

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 36-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Component	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S1_S36	16384 x 1	-	(13:0)	(0:0)	-	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a) Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables show address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

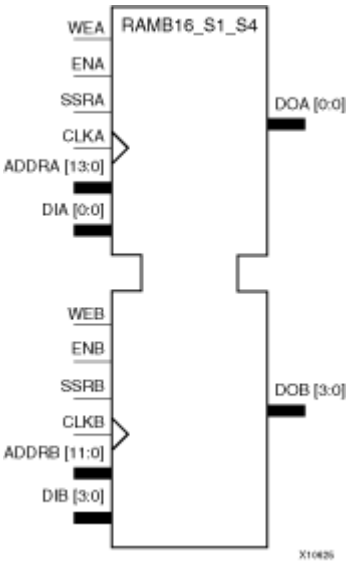
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S1_S4

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 4-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S1_S4	16384 x 1	-	(13:0)	(0:0)	-	4096 x 4	-	(11:0)	(3:0)	-

(a)Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", "NONE", "WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested

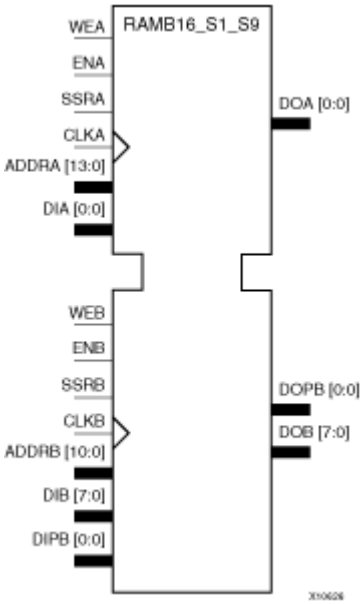
Attribute	Type	Allowed Values	Default	Description
				mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S1_S9

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 1-bit and 9-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	**ENB**	**SSRB**	**WEB**	**CLKB**	**ADDRB**	**DIB**	**DIPB**	**DOB**	**DOPB**	**RAM Contents**	
										Data RAM	**Parity RAM**
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

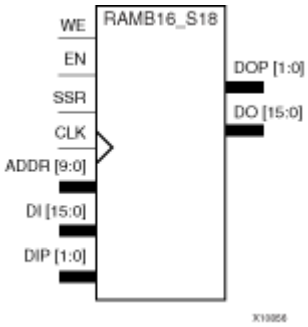
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S18

Primitive: 16K-bit Data + 2K-bit Parity Memory, Single-Port Synchronous Block RAM with 18-bit Port



Introduction

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Data Cells		Parity Cells				
Depth	**Width**	**Depth**	**Width**	**Address Bus**	**Data Bus**	**Parity Bus**
1024	16	1024	2	(9:0)	(15:0)	(1:0)

The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The above description assumes an active High EN, WE, SSR, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Logic Table

Inputs								Outputs			
GSR	**EN**	**SSR**	**WE**	**CLK**	**ADDR**	**DI**	**DIP**	**DO**	**DOP**	**RAM Contents**	
										Data RAM	**Parity RAM**
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Initialization

Initializing Memory Contents

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16 is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

Any INIT_xx or INITP_xx attribute that is not specified is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register

In Spartan-3A, each bit in the output register can be initialized at power on to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Two types of properties control initialization of the output register for a single-port RAMB16: INIT and SRVAL. The INIT attribute specifies the output register value at power on. You can use the SRVAL attribute to define the state resulting from assertion of the SSR (set/reset) input.

The INIT and SRVAL attributes specify the initialization value as a hexadecimal String containing one bit for each bit in the output port. For example, for a RAMB16_S1 with port width equal to 1, the output register contains 1 bit. Therefore, the INIT or SRVAL value can only be specified as a 1 or 0. For RAMB16_S4 with port width equal to 4, the output register contains 4 bits. In this case, you can specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT or SRVAL value.

Selecting Write Mode

The WRITE_MODE attribute controls RAMB16 memory and output contents. By default, the WRITE_MODE is set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the WRITE_MODE to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the WRITE_MODE to NO_CHANGE to have the input written to memory without changing the output.

Design Entry Method

This design element can be used in schematics.

Available Attributes

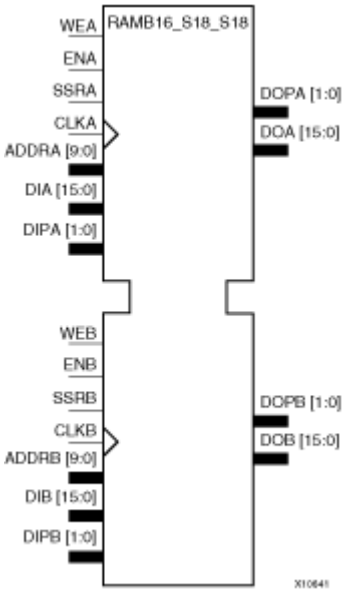
Attribute	Type	Allowed Values	Default	Description
INIT	Binary/ Hexadecimal	Any Hex Value	All zeros	Identifies the initial value of the DO output port after completing configuration. The bit width is dependent on the width of the A or B port of the RAM.
INIT_00 - INIT_3F	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the data portion of the RAM array.
INITP_00 - INITP_07	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SRVAL	Binary/ Hexadecimal	Any Hex Value	All zeros	Allows the individual selection of whether the DO output port sets (go to a one) or reset (go to a zero) upon the assertion of the SSR pin. The bit width is dependent on the width of the A or B port of the RAM.
WRITE_MODE	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DO port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S18_S18

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S18_S18	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables show address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S18_S36

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 18-bit and 36-bit Ports

Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S18_S36	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)

(a)Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

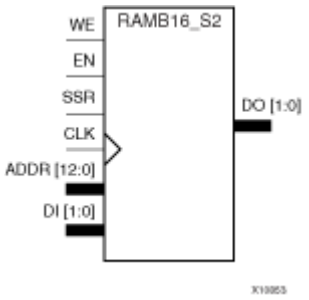
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S2

Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 2-bit Port



Introduction

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Data Cells		Parity Cells				
Depth	**Width**	**Depth**	**Width**	**Address Bus**	**Data Bus**	**Parity Bus**
8192	2	-	-	(12:0)	(1:0)	-

The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The above description assumes an active High EN, WE, SSR, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Logic Table

Inputs								Outputs			
GSR	**EN**	**SSR**	**WE**	**CLK**	**ADDR**	**DI**	**DIP**	**DO**	**DOP**	**RAM Contents**	
										Data RAM	**Parity RAM**
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Initialization

Initializing Memory Contents

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16 is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

Any INIT_xx or INITP_xx attribute that is not specified is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register

In Spartan-3A, each bit in the output register can be initialized at power on to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Two types of properties control initialization of the output register for a single-port RAMB16: INIT and SRVAL. The INIT attribute specifies the output register value at power on. You can use the SRVAL attribute to define the state resulting from assertion of the SSR (set/reset) input.

The INIT and SRVAL attributes specify the initialization value as a hexadecimal String containing one bit for each bit in the output port. For example, for a RAMB16_S1 with port width equal to 1, the output register contains 1 bit. Therefore, the INIT or SRVAL value can only be specified as a 1 or 0. For RAMB16_S4 with port width equal to 4, the output register contains 4 bits. In this case, you can specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT or SRVAL value.

Selecting Write Mode

The WRITE_MODE attribute controls RAMB16 memory and output contents. By default, the WRITE_MODE is set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the WRITE_MODE to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the WRITE_MODE to NO_CHANGE to have the input written to memory without changing the output.

Design Entry Method

This design element can be used in schematics.

Available Attributes

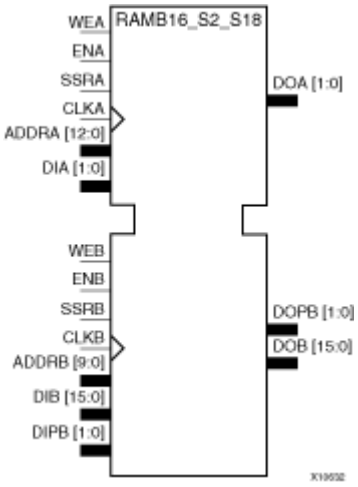
Attribute	Type	Allowed Values	Default	Description
INIT	Binary/ Hexadecimal	Any Hex Value	All zeros	Identifies the initial value of the DO output port after completing configuration. The bit width is dependent on the width of the A or B port of the RAM.
INIT_00 - INIT_3F	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the data portion of the RAM array.
INITP_00 - INITP_07	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SRVAL	Binary/ Hexadecimal	Any Hex Value	All zeros	Allows the individual selection of whether the DO output port sets (go to a one) or reset (go to a zero) upon the assertion of the SSR pin. The bit width is dependent on the width of the A or B port of the RAM.
WRITE_MODE	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DO port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S2_S18

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 18-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S2_S18	8192 x 2	-	(12:0)	(1:0)	-	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a)Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

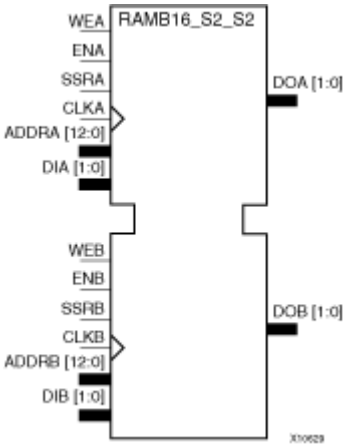
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S2_S2

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S2_S2	8192 x 2	-	(12:0)	(1:0)	-	8192 x 2	-	(12:0)	(1:0)	-
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_CHECK	String	"ALL", "NONE", "WARNING", or "GENERATE_X_ ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested

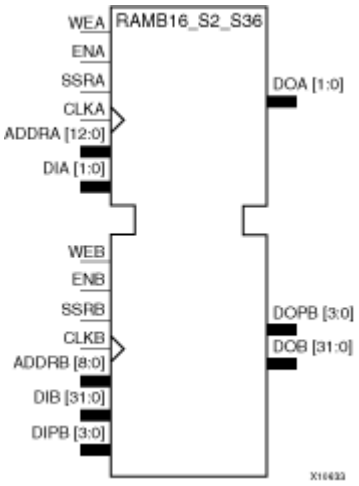
Attribute	Type	Allowed Values	Default	Description
				mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S2_S36

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 36-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S2_S36	8192 x 2	-	(12:0)	(1:0)	-	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the “Port Address Mapping for Data” table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in “Port Address Mapping for Parity” table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.

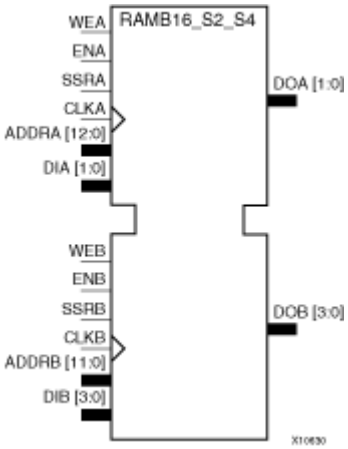
Attribute	Type	Allowed Values	Default	Description
SIM_COLLISION_ CHECK	String	"ALL", "NONE", "WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S2_S4

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 4-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells (a)	Parity Cells (a)	Address Bus	Data Bus	Parity Bus	Data Cells (a)	Parity Cells (a)	Address Bus	Data Bus	Parity Bus
RAMB16_S2_S4	8192 x 2	-	(12:0)	(1:0)	-	4096 x 4	-	(11:0)	(3:0)	-
(a) Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the “Port Address Mapping for Data” table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in “Port Address Mapping for Parity” table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", "NONE", "WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.

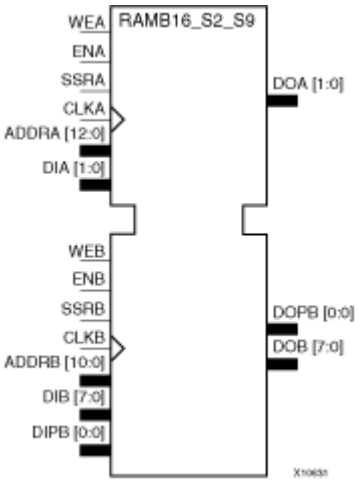
Attribute	Type	Allowed Values	Default	Description
SRVAL_A	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S2_S9

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 2-bit and 9-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S2_S9	8192 x 2	-	(12:0)	(1:0)	-	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the “Port Address Mapping for Data” table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in “Port Address Mapping for Parity” table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	**Port Data Addresses**																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_CHECK	String	"ALL", "NONE", "WARNING", or "GENERATE_X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.

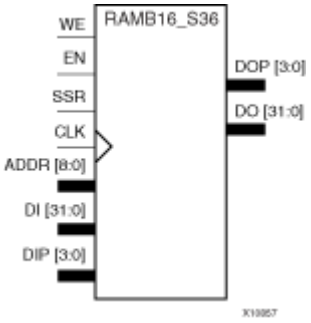
Attribute	Type	Allowed Values	Default	Description
SRVAL_A	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S36

Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 36-bit Port



Introduction

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Data Cells		Parity Cells				
Depth	**Width**	**Depth**	**Width**	**Address Bus**	**Data Bus**	**Parity Bus**
512	32	512	4	(8:0)	(31:0)	(3:0)

The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The above description assumes an active High EN, WE, SSR, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Logic Table

Inputs								Outputs			
GSR	**EN**	**SSR**	**WE**	**CLK**	**ADDR**	**DI**	**DIP**	**DO**	**DOP**	**RAM Contents**	
										Data RAM	**Parity RAM**
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Initialization

Initializing Memory Contents

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16 is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

Any INIT_xx or INITP_xx attribute that is not specified is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register

In Spartan-3A, each bit in the output register can be initialized at power on to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Two types of properties control initialization of the output register for a single-port RAMB16: INIT and SRVAL. The INIT attribute specifies the output register value at power on. You can use the SRVAL attribute to define the state resulting from assertion of the SSR (set/reset) input.

The INIT and SRVAL attributes specify the initialization value as a hexadecimal String containing one bit for each bit in the output port. For example, for a RAMB16_S1 with port width equal to 1, the output register contains 1 bit. Therefore, the INIT or SRVAL value can only be specified as a 1 or 0. For RAMB16_S4 with port width equal to 4, the output register contains 4 bits. In this case, you can specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT or SRVAL value.

Selecting Write Mode

The WRITE_MODE attribute controls RAMB16 memory and output contents. By default, the WRITE_MODE is set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the WRITE_MODE to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the WRITE_MODE to NO_CHANGE to have the input written to memory without changing the output.

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Binary/ Hexadecimal	Any Hex Value	All zeros	Identifies the initial value of the DO output port after completing configuration. The bit width is dependent on the width of the A or B port of the RAM.
INIT_00 - INIT_3F	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the data portion of the RAM array.
INITP_00 - INITP_07	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SRVAL	Binary/ Hexadecimal	Any Hex Value	All zeros	Allows the individual selection of whether the DO output port sets (go to a one) or reset (go to a zero) upon the assertion of the SSR pin. The bit width is dependent on the width of the A or B port of the RAM.
WRITE_MODE	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DO port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S36_S36

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with Two 36-bit Ports

Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S36_S36	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

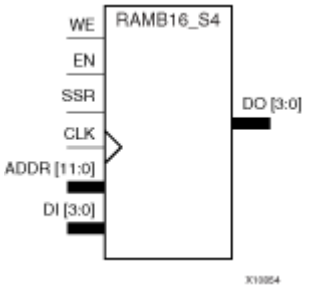
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S4

Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 4-bit Port



Introduction

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Data Cells		Parity Cells				
Depth	**Width**	**Depth**	**Width**	**Address Bus**	**Data Bus**	**Parity Bus**
4096	4	-	-	(11:0)	(3:0)	-

The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The above description assumes an active High EN, WE, SSR, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Logic Table

Inputs								Outputs			
GSR	**EN**	**SSR**	**WE**	**CLK**	**ADDR**	**DI**	**DIP**	**DO**	**DOP**	**RAM Contents**	
										Data RAM	**Parity RAM**
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Initialization

Initializing Memory Contents

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16 is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

Any INIT_xx or INITP_xx attribute that is not specified is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register

In Spartan-3A, each bit in the output register can be initialized at power on to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Two types of properties control initialization of the output register for a single-port RAMB16: INIT and SRVAL. The INIT attribute specifies the output register value at power on. You can use the SRVAL attribute to define the state resulting from assertion of the SSR (set/reset) input.

The INIT and SRVAL attributes specify the initialization value as a hexadecimal String containing one bit for each bit in the output port. For example, for a RAMB16_S1 with port width equal to 1, the output register contains 1 bit. Therefore, the INIT or SRVAL value can only be specified as a 1 or 0. For RAMB16_S4 with port width equal to 4, the output register contains 4 bits. In this case, you can specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT or SRVAL value.

Selecting Write Mode

The WRITE_MODE attribute controls RAMB16 memory and output contents. By default, the WRITE_MODE is set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the WRITE_MODE to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the WRITE_MODE to NO_CHANGE to have the input written to memory without changing the output.

Design Entry Method

This design element can be used in schematics.

Available Attributes

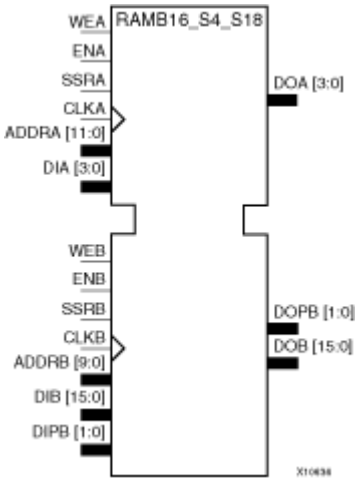
Attribute	Type	Allowed Values	Default	Description
INIT	Binary/ Hexadecimal	Any Hex Value	All zeros	Identifies the initial value of the DO output port after completing configuration. The bit width is dependent on the width of the A or B port of the RAM.
INIT_00 - INIT_3F	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the data portion of the RAM array.
INITP_00 - INITP_07	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SRVAL	Binary/ Hexadecimal	Any Hex Value	All zeros	Allows the individual selection of whether the DO output port sets (go to a one) or reset (go to a zero) upon the assertion of the SSR pin. The bit width is dependent on the width of the A or B port of the RAM.
WRITE_MODE	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DO port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S4_S18

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 18-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S4_S18	4096 x 4	-	(11:0)	(3:0)	-	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a)Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

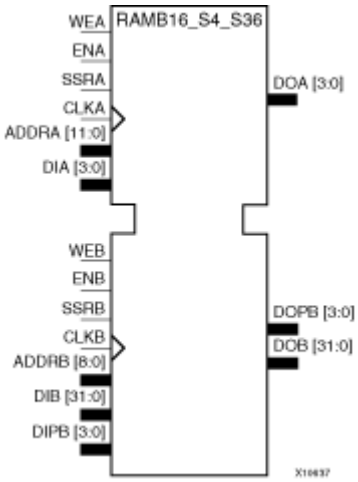
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S4_S36

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 36-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S4_S36	4096 x 4	-	(11:0)	(3:0)	-	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0			
8	2048	<--	3								2								1								0							
16	1024	<--	1																0															
32	512	<--	0																															

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

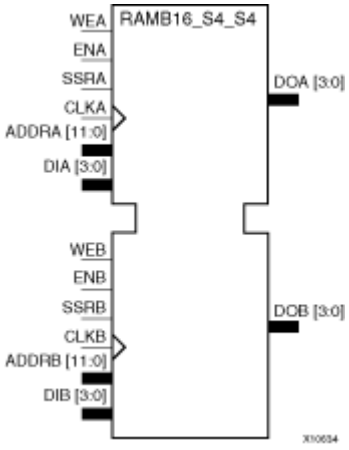
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S4_S4

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S4_S4	4096 x 4	-	(11:0)	(3:0)	-	4096 x 4	-	(11:0)	(3:0)	-
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

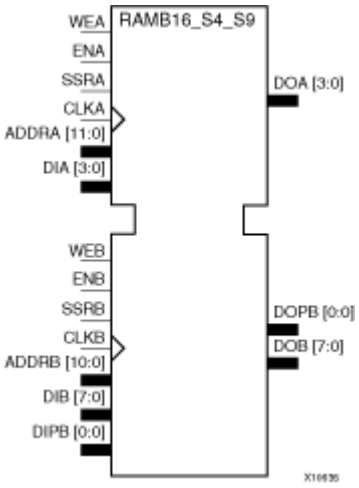
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S4_S9

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 4-bit and 9-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S4_S9	4096 x 4	-	(11:0)	(3:0)	-	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

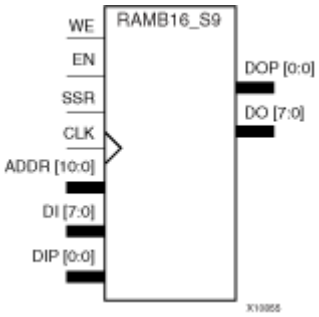
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S9

Primitive: 16K-bit Data and 2K-bit Parity Single-Port Synchronous Block RAM with 9-bit Port



Introduction

This design element is a dedicated random access memory blocks with synchronous write capability. The block RAM port has 16384 bits of data memory. The cell configuration for this element is listed in the following table.

Data Cells		Parity Cells				
Depth	**Width**	**Depth**	**Width**	**Address Bus**	**Data Bus**	**Parity Bus**
2048	8	2048	1	(10:0)	(7:0)	(0:0)

The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The above description assumes an active High EN, WE, SSR, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Logic Table

Inputs								Outputs			
GSR	**EN**	**SSR**	**WE**	**CLK**	**ADDR**	**DI**	**DIP**	**DO**	**DOP**	**RAM Contents**	
										Data RAM	**Parity RAM**
1	X	X	X	X	X	X	X	INIT	INIT	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL	SRVAL	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL	SRVAL	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	EN	SSR	WE	CLK	ADDR	DI	DIP	DO	DOP	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change(a) RAM (addr)(b) data(c)	No Change(a) RAM (addr)(b) pdata(c)	RAM (addr)=>data	RAM (addr)=>pdata

GSR=Global Set Reset signal

INIT=Value specified by the INIT attribute for data memory. Default is all zeros.

SRVAL=Value after assertion of SSR as specified by the SRVAL attribute.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

(a) WRITE_MODE=NO_CHANGE

(b) WRITE_MODE=READ_FIRST

(c) WRITE_MODE=WRITE_FIRST

Initialization

Initializing Memory Contents

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16 is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

Any INIT_xx or INITP_xx attribute that is not specified is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register

In Spartan-3A, each bit in the output register can be initialized at power on to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Two types of properties control initialization of the output register for a single-port RAMB16: INIT and SRVAL. The INIT attribute specifies the output register value at power on. You can use the SRVAL attribute to define the state resulting from assertion of the SSR (set/reset) input.

The INIT and SRVAL attributes specify the initialization value as a hexadecimal String containing one bit for each bit in the output port. For example, for a RAMB16_S1 with port width equal to 1, the output register contains 1 bit. Therefore, the INIT or SRVAL value can only be specified as a 1 or 0. For RAMB16_S4 with port width equal to 4, the output register contains 4 bits. In this case, you can specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT or SRVAL value.

Selecting Write Mode

The WRITE_MODE attribute controls RAMB16 memory and output contents. By default, the WRITE_MODE is set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the WRITE_MODE to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the WRITE_MODE to NO_CHANGE to have the input written to memory without changing the output.

Design Entry Method

This design element can be used in schematics.

Available Attributes

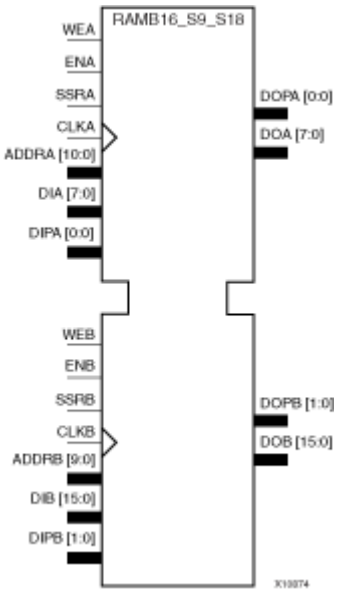
Attribute	Type	Allowed Values	Default	Description
INIT	Binary/ Hexadecimal	Any Hex Value	All zeros	Identifies the initial value of the DO output port after completing configuration. The bit width is dependent on the width of the A or B port of the RAM.
INIT_00 - INIT_3F	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the data portion of the RAM array.
INITP_00 - INITP_07	Binary/ Hexadecimal	Any Hex Value	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SRVAL	Binary/ Hexadecimal	Any Hex Value	All zeros	Allows the individual selection of whether the DO output port sets (go to a one) or reset (go to a zero) upon the assertion of the SSR pin. The bit width is dependent on the width of the A or B port of the RAM.
WRITE_MODE	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DO port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S9_S18

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 18-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Component	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S9_S18	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)	1024 x 16	1024 x 2	(9:0)	(15:0)	(1:0)

(a) Depth x Width

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables show address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

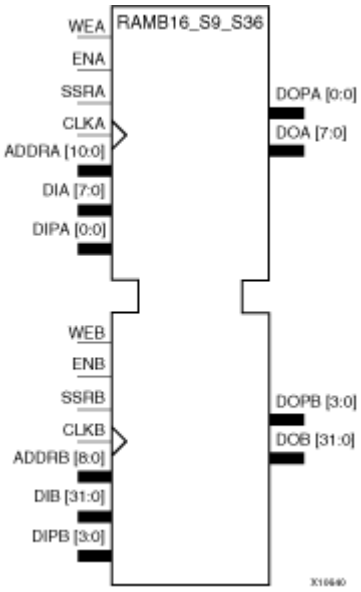
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S9_S36

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit and 36-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S9_S36	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)	512 x 32	512 x 4	(8:0)	(31:0)	(3:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

The following tables shows address mapping for each port width.

Port Address Mapping for Data

Data Width	Port Data Addresses																																
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses															
1	2048	<-----	3				2				1				0	
2	1024	<-----	1								0					
4	512	<-----	0													

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

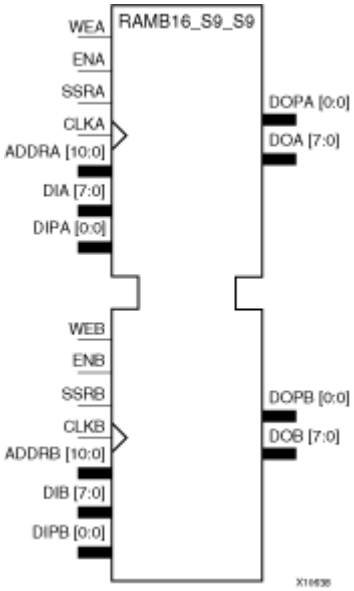
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB16_S9_S9

Primitive: 16K-bit Data and 2K-bit Parity Dual-Port Synchronous Block RAM with 9-bit Ports



Introduction

This design element is a dual-ported dedicated random access memory block with synchronous write capability. Each block RAM port has 16384 bits of data memory. Ports configured as 9, 18, or 36-bits wide have an additional 2048 bits of parity memory. Each port is independent of the other while accessing the same set of 16384 data memory cells. Each port is independently configured to a specific data width. The possible port and cell configurations for this element are listed under "Port Descriptions."

Logic Table

Truth Table A

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_A	INIT_A	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_A	SRVAL_A	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_A	SRVAL_A	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change

Inputs								Outputs			
GSR	ENA	SSRA	WEA	CLKA	ADDRA	DIA	DIPA	DOA	DOPA	RAM Contents	
										Data RAM	Parity RAM
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr)2, data3	No Change1, RAM(addr)2, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset

INIT_A=Value specified by the INIT_A attribute for output register. Default is all zeros.

SRVAL_A=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_A=NO_CHANGE.

2WRITE_MODE_A=READ_FIRST.

3WRITE_MODE_A=WRITE_FIRST.

Truth Table B

Inputs								Outputs			
GSR	ENB	SSRB	WEB	CLKB	ADDRB	DIB	DIPB	DOB	DOPB	RAM Contents	
										Data RAM	Parity RAM
1	X	X	X	X	X	X	X	INIT_B	INIT_B	No Change	No Change
0	0	X	X	X	X	X	X	No Change	No Change	No Change	No Change
0	1	1	0	↑	X	X	X	SRVAL_B	SRVAL_B	No Change	No Change
0	1	1	1	↑	addr	data	pdata	SRVAL_B	SRVAL_B	RAM(addr) =>data	RAM(addr) =>pdata
0	1	0	0	↑	addr	X	X	RAM(addr)	RAM(addr)	No Change	No Change
0	1	0	1	↑	addr	data	pdata	No Change1, RAM(addr) =>data, data3	No Change1, RAM(addr) =>data, pdata3	RAM(addr) =>data	RAM(addr) =>pdata

GSR=Global Set Reset.

INIT_B=Value specified by the INIT_B attribute for output registers. Default is all zeros.

SRVAL_B=register value.

addr=RAM address.

RAM(addr)=RAM contents at address ADDR.

data=RAM input data.

pdata=RAM parity data.

1WRITE_MODE_B=NO_CHANGE.

2WRITE_MODE_B=READ_FIRST.

3WRITE_MODE_B=WRITE_FIRST.

Port Descriptions

Port A						Port B				
Design Element	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus	Data Cells(a)	Parity Cells(a)	Address Bus	Data Bus	Parity Bus
RAMB16_S9_S9	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)	2048 x 8	2048 x 1	(10:0)	(7:0)	(0:0)
(a)Depth x Width										

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB. The enable ENA pin controls read, write, and reset for Port A. When ENA is Low, no data is written and the outputs (DOA and DOPA) retain the last state. When ENA is High and reset (SSRA) is High, DOA and DOPA are set to SRVAL_A during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA and DIPA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. By default, WRITE_MODE_A=WRITE_FIRST, when ENA and WEA are High, the data on the data inputs (DIA and DIPA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data outputs (DOA and DOPA) reflect the selected (addressed) word.

The enable ENB pin controls read, write, and reset for Port B. When ENB is Low, no data is written and the outputs (DOB and DOPB) retain the last state. When ENB is High and reset (SSRB) is High, DOB and DOPB are set to SRVAL_B during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB and DIPB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. By default, WRITE_MODE_B=WRITE_FIRST, when ENB and WEB are High, the data on the data inputs (DIB and PB) are loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data outputs (DOB and DOPB) reflect the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, SSRA, CLKA, ENB, WEB, SSRB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource.

Address Mapping

Each port accesses the same set of 18432 memory cells using an addressing scheme that is dependent on the width of the port. For all port widths, 16384 memory cells are available for data as shown in the "Port Address Mapping for Data" table below. For 9-, 18-, and 36-bit wide ports, 2408 parity memory cells are also available as shown in "Port Address Mapping for Parity" table below. The physical RAM location that is addressed for a particular width is determined from the following formula.

Start=((ADDR port+1)*(Widthport)) -1

End=(ADDRport)*(Widthport)

Port Address Mapping for Data

Data Width	Port Data Addresses																															
1	16384	<--	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	8192	<--	15		14		13		12		11		10		9		8		7		6		5		4		3		2		1		0	
4	4096	<--	7				6				5				4				3				2				1				0		
8	2048	<--	3								2								1								0						
16	1024	<--	1																0														
32	512	<--	0																														

Port Address Mapping for Parity

Parity Width	Port Parity Addresses														
1	2048	<-----	3				2				1				0
2	1024	<-----	1								0				
4	512	<-----	0												

Initializing Memory Contents of a Dual-Port RAMB16

You can use the INIT_xx attributes to specify an initialization value for the memory contents of a RAMB16 during device configuration. The initialization of each RAMB16_Sm_Sn is set by 64 initialization attributes (INIT_00 through INIT_3F) of 64 hex values for a total of 16384 bits.

You can use the INITP_xx attributes to specify an initial value for the parity memory during device configuration or assertion. The initialization of the parity memory for ports configured for 9, 18, or 36 bits is set by 8 initialization attributes (INITP_00 through INITP_07) of 64 hex values for a total of 2048 bits.

If any INIT_xx or INITP_xx attribute is not specified, it is configured as zeros. Partial Strings are padded with zeros to the left.

Initializing the Output Register of a Dual-Port RAMB16

In Spartan-3A, each bit in an output register can be initialized at power on (when GSR is high) to either a 0 or 1. In addition, the initial state specified for power on can be different than the state that results from assertion of a set/reset. Four properties control initialization of the output register for a dual-port RAMB16: INIT_A, INIT_B, SRVAL_A, and SRVAL_B. The INIT_A attribute specifies the output register value at power on for Port A and the INIT_B attribute specifies the value for Port B. You can use the SRVAL_A attribute to define the state resulting from assertion of the SSR (set/reset) input on Port A. You can use the SRVAL_B attribute to define the state resulting from assertion of the SSR input on Port B.

The INIT_A, INIT_B, SRVAL_A, and SRVAL_B attributes specify the initialization value as a hexadecimal String. The value is dependent upon the port width. For example, for a RAMB16_S1_S4 with Port A width equal to 1 and Port B width equal to 4, the Port A output register contains 1 bit and the Port B output register contains 4 bits. Therefore, the INIT_A or SRVAL_A value can only be specified as a 1 or 0. For Port B, the output register contains 4 bits. In this case, you can use INIT_B or SRVAL_B to specify a hexadecimal value from 0 through F to initialize the 4 bits of the output register.

For those ports that include parity bits, the parity portion of the output register is specified in the high order bit position of the INIT_A, INIT_B, SRVAL_A, or SRVAL_B value.

The INIT and SRVAL attributes default to zero if they are not set by you.

Write Mode Selection

The WRITE_MODE_A attribute controls the memory and output contents of Port A for a dual-port RAMB16. The WRITE_MODE_B attribute does the same for Port B. By default, both WRITE_MODE_A and WRITE_MODE_B are set to WRITE_FIRST. This means that input is read, written to memory, and then passed to output. You can set the write mode for Port A and Port B to READ_FIRST to read the memory contents, pass the memory contents to the outputs, and then write the input to memory. Or, you can set the write mode to NO_CHANGE to have the input written to memory without changing the output. The “Port A and Port B Conflict Resolution” section describes how read/write conflicts are resolved when both Port A and Port B are attempting to read/write to the same memory cells.

Port A and Port B Conflict Resolution

Spartan-3A block SelectRAM is True Dual-Port RAM that allows both ports to simultaneously access the same memory cell. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window.

The following tables summarize the collision detection behavior of the dual-port RAMB16 based on the WRITE_MODE_A and WRITE_MODE_B settings.

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=NO_CHANGE

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	No Change	X	No Change	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	No Change	No Change	No Change	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	X	X

WRITE_MODE_A= WRITE_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	DIA	X	DIPA	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	X	X	X	X	X

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=READ_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIB	DIPB

WRITE_MODE_A=NO_CHANGE and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	No Change	X	No Change	X	X	X

WRITE_MODE_A=READ_FIRST and WRITE_MODE_B=WRITE_FIRST

WEA	WEB	CLKA	CLKB	DIA	DIB	DIPA	DIPB	DOA	DOB	DOPA	DOPB	Data RAM	Parity Ram
0	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	No Change	No Change
1	0	↑	↑	DIA	DIB	DIPA	DIPB	RAM	RAM	RAM	RAM	DIA	DIPA
0	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIB	DIPB
1	1	↑	↑	DIA	DIB	DIPA	DIPB	X	DIB	X	DIPB	DIA	DIPA

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.
INIT_A	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexadecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexadecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", ¨NONE", ¨WARNING", or "GENERATE_ X_ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary /Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexadecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.

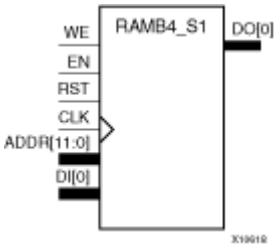
Attribute	Type	Allowed Values	Default	Description
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S1

Primitive: 4K-Bit Single-Port Synchronous Block RAM with Port Width Configured to 1 Bit



Introduction

This design element is a dedicated, random access memory block with synchronous write capability. It provides the capability for fast, discrete, large blocks of RAM in each device. This element is configured as indicated in the following table:

Design Element	Depth	Width	Address Bus	Data Bus
RAMB4_S1	4096	1	(11:0)	(0:0)

The enable (EN) pin controls read, write, and reset. When EN is Low, no data is written and the output (DO) retains the last state. When EN is High and reset (RST) is High, DO is cleared during the Low-to-High clock (CLK) transition; if write enable (WE) is High, the memory contents reflect the data at DI. When EN is High and WE is Low, the data stored in the RAM address (ADDR) is read during the Low-to-High clock transition. When EN and WE are High, the data on the data input (DI) is loaded into the word selected by the write address (ADDR) during the Low-to-High clock transition and the data output (DO) reflects the selected (addressed) word. The above description assumes an active High EN, WE, RST, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

This element can be initialized during configuration. Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs	
EN	**RST**	**WE**	**CLK**	**ADDR**	**DI**	**DO**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address. RAM(addr)=RAM contents at address ADDR. data=RAM input data.							

Specifying Initial Contents of a Block RAM -

You can use the INIT_xx attributes to specify an initial value during device configuration. The initialization of each of these elements is set by 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Design Entry Method

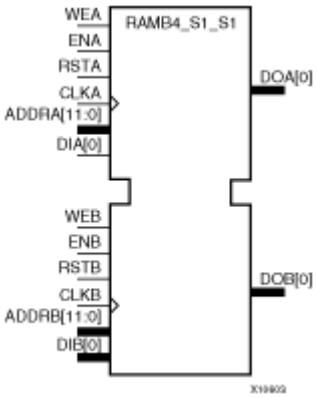
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S1_S1

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S1_S1	4096	1	(11:0)	(0:0)	4096	1	(11:0)	(0:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
1	4096	<-----	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

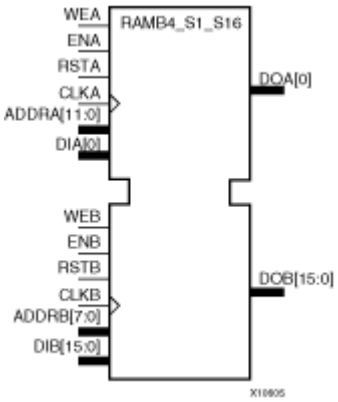
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S1_S16

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 16-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S1_S16	4096	1	(11:0)	(0:0)	256	16	(7:0)	(15:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
1	4096	<-----	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
16	256	<-----	0															

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

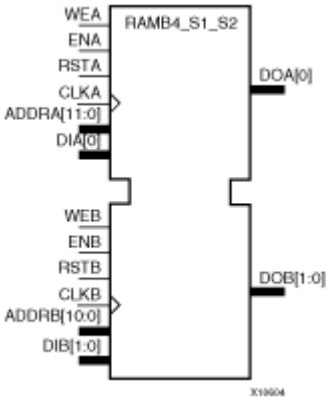
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S1_S2

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 2-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S1_S2	4096	1	(11:0)	(0:0)	2048	2	(10:0)	(1:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
1	4096	<-----	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	2048	<-----	7		6		5		4		3		2		1		0	

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT_00 To INIT_3F	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the data portion of the RAM array.

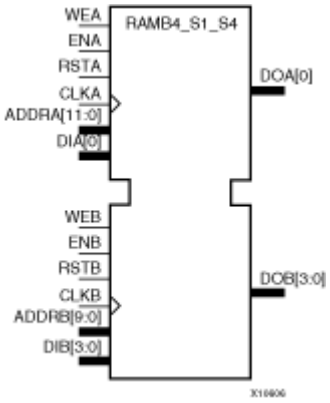
Attribute	Type	Allowed Values	Default	Description
INIT_A	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INIT_B	Binary/ Hexidecimal	Any	All zeros	Identifies the initial value of the DOA/DOB output port after completing configuration. For Type, the bit width is dependent on the width of the A or B port of the RAM.
INITP_00 To INITP_07	Binary/ Hexidecimal	Any	All zeros	Specifies the initial contents of the parity portion of the RAM array.
SIM_COLLISION_ CHECK	String	"ALL", "NONE", "WARNING", or "GENERATE_X_ ONLY"	"ALL"	Specifies the behavior during simulation in the event of a data collision (data being read or written to the same address from both ports of the Ram simultaneously. "ALL" issues a warning to simulator console and generate an X or all unknown data due to the collision. This is the recommended setting. "WARNING" generates a warning only and "GENERATE_X_ONLY" generates an X for unknown data but won't output the occurrence to the simulation console. "NONE" completely ignores the error. It is suggested to only change this attribute if you can ensure the data generated during a collision is discarded.
SRVAL_A	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTA pin. For Type, the bit width is dependent on the width of the A port of the RAM.
SRVAL_B	Binary/ Hexidecimal	Any	All zeros	Allows the individual selection of whether the DOA/DOB output port sets (go to a one) or reset (go to a zero) upon the assertion of the RSTB pin. For Type, the bit width is dependent on the width of the B port of the RAM.
WRITE_MODE_A	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM
WRITE_MODE_B	String	"WRITE_FIRST", "READ_FIRST" or "NO_CHANGE"	"WRITE_ FIRST"	Specifies the behavior of the DOA/DOB port upon a write command to the respected port. If set to "WRITE_FIRST", the same port that is written to displays the contents of the written data to the outputs upon completion of the operation. "READ_FIRST" displays the prior contents of the RAM to the output port prior to writing the new data. "NO_CHANGE" keeps the previous value on the output port and won't update the output port upon a write command. This is the suggested mode if not using the read data from a particular port of the RAM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S1_S4

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 4-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S1_S4	4096	1	(11:0)	(0:0)	1024	4	(9:0)	(3:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
1	4096	<-----	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
4	1024	<-----	3				2				1				0			

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

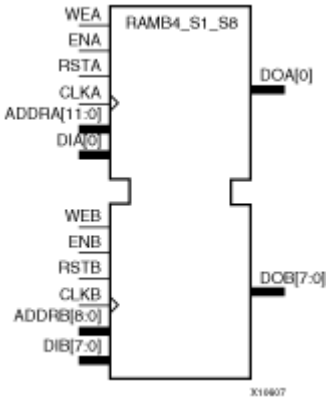
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S1_S8

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 1-bit and 8-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S1_S8	4096	1	(11:0)	(0:0)	512	8	(8:0)	(7:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word. The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
1	4096	<-----	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
8	512	<-----	1								0							

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

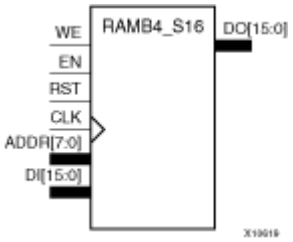
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S16

Primitive: 4096-Bit Single-Port Synchronous Block RAM with Port Width Configured to 16 Bits



Introduction

This design element is a dedicated, random access memory block with synchronous write capability. It provides the capability for fast, discrete, large blocks of RAM in each device. This element is configured as indicated in the following table:

Design Element	Depth	Width	Address Bus	Data Bus
RAMB4_S16	256	16	(7:0)	(15:0)

The enable (EN) pin controls read, write, and reset. When EN is Low, no data is written and the output (DO) retains the last state. When EN is High and reset (RST) is High, DO is cleared during the Low-to-High clock (CLK) transition; if write enable (WE) is High, the memory contents reflect the data at DI. When EN is High and WE is Low, the data stored in the RAM address (ADDR) is read during the Low-to-High clock transition. When EN and WE are High, the data on the data input (DI) is loaded into the word selected by the write address (ADDR) during the Low-to-High clock transition and the data output (DO) reflects the selected (addressed) word. The above description assumes an active High EN, WE, RST, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

This element can be initialized during configuration. Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs	
EN	**RST**	**WE**	**CLK**	**ADDR**	**DI**	**DO**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address. RAM(addr)=RAM contents at address ADDR. data=RAM input data.							

Specifying Initial Contents of a Block RAM -

You can use the INIT_xx attributes to specify an initial value during device configuration. The initialization of each of these elements is set by 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Design Entry Method

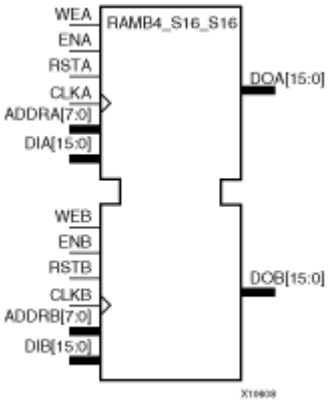
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S16_S16

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 16-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S16_S16	256	16	(7:0)	(15:0)	256	16	(7:0)	(15:0)
ADDR=address bus for the port DI=data input bus for the port								

All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
16	256	<-----	0															

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

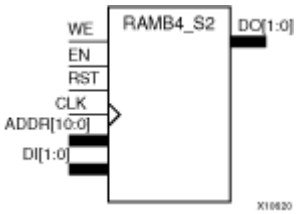
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S2

Primitive: 4K-bit Single-Port Synchronous Block RAM with Port Width Configured to 2-bits



Introduction

This design element is a dedicated, random access memory block with synchronous write capability. It provides the capability for fast, discrete, large blocks of RAM in each device. This element is configured as indicated in the following table:

Design Element	Depth	Width	Address Bus	Data Bus
RAMB4_S2	2048	2	(10:0)	(1:0)

The enable (EN) pin controls read, write, and reset. When EN is Low, no data is written and the output (DO) retains the last state. When EN is High and reset (RST) is High, DO is cleared during the Low-to-High clock (CLK) transition; if write enable (WE) is High, the memory contents reflect the data at DI. When EN is High and WE is Low, the data stored in the RAM address (ADDR) is read during the Low-to-High clock transition. When EN and WE are High, the data on the data input (DI) is loaded into the word selected by the write address (ADDR) during the Low-to-High clock transition and the data output (DO) reflects the selected (addressed) word. The above description assumes an active High EN, WE, RST, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

This element can be initialized during configuration. Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs	
EN	**RST**	**WE**	**CLK**	**ADDR**	**DI**	**DO**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address. RAM(addr)=RAM contents at address ADDR. data=RAM input data.							

Specifying Initial Contents of a Block RAM -

You can use the INIT_xx attributes to specify an initial value during device configuration. The initialization of each of these elements is set by 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Design Entry Method

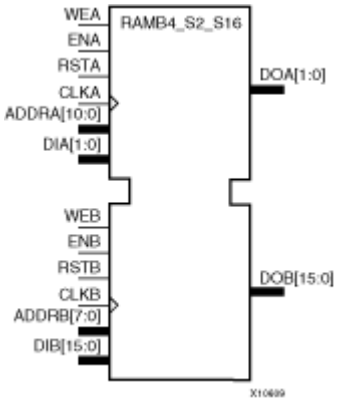
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S2_S16

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 16-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S2_S16	2048	2	(10:0)	(1:0)	256	16	(7:0)	(15:0)
ADDR=address bus for the port. DI=data input bus for the port.								

All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
2	2048	<-----	7		6		5		4		3		2		1		0	
16	256	<-----	0															

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

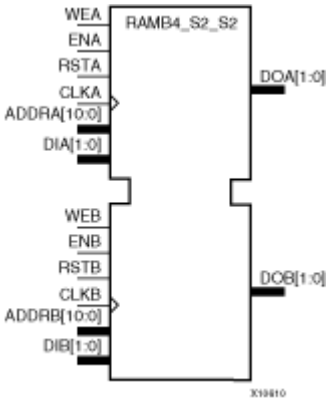
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S2_S2

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 2-bits



Introduction

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S2_S2	2048	2	(10:0)	(1:0)	2048	2	(10:0)	(1:0)
ADDR=address bus for the port DI=data input bus for the port								

All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	RST(A/B)	WE(A/B)	CLK(A/B)	ADDR(A/B)	DI(A/B)	DO(A/B)	RAM Contents
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
2	2048	<-----	7		6		5		4		3		2		1		0	

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

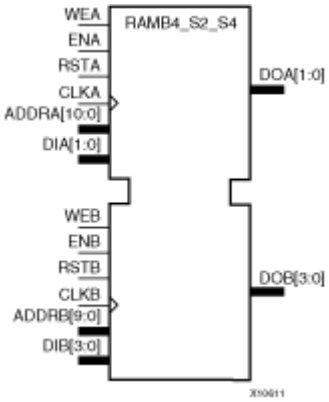
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S2_S4

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 4-bits



Introduction

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S2_S4	2048	2	(10:0)	(1:0)	1024	4	(9:0)	(3:0)
ADDR=address bus for the port. DI=data input bus for the port.								

All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

This component can be initialized during configuration. See the logic table below.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																
2	2048	<-----	7		6		5		4		3		2		1		0
4	1024	<-----	3				2				1				0		

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

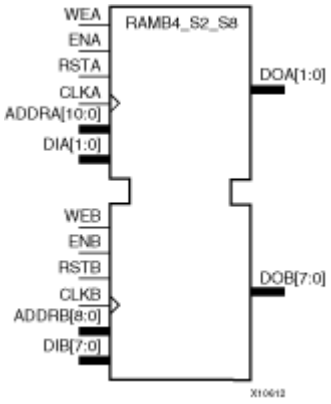
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S2_S8

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 2-bits and 8-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S2_S8	2048	2	(10:0)	(1:0)	512	8	(8:0)	(7:0)
ADDR=address bus for the port. DI=data input bus for the port.								

All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																
2	2048	<-----	7		6		5		4		3		2		1		0	
8	512	<-----	1								0							

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

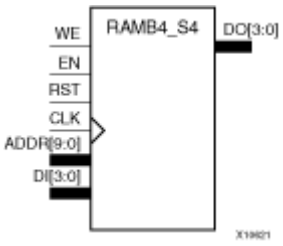
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S4

Primitive: 4k-bit Single-Port Synchronous Block RAM with Port Width Configured to 4-bits



Introduction

This design element is a dedicated, random access memory block with synchronous write capability. It provides the capability for fast, discrete, large blocks of RAM in each device. This element is configured as indicated in the following table:

Design Element	Depth	Width	Address Bus	Data Bus
RAMB4_S4	1024	4	(9:0)	(3:0)

The enable (EN) pin controls read, write, and reset. When EN is Low, no data is written and the output (DO) retains the last state. When EN is High and reset (RST) is High, DO is cleared during the Low-to-High clock (CLK) transition; if write enable (WE) is High, the memory contents reflect the data at DI. When EN is High and WE is Low, the data stored in the RAM address (ADDR) is read during the Low-to-High clock transition. When EN and WE are High, the data on the data input (DI) is loaded into the word selected by the write address (ADDR) during the Low-to-High clock transition and the data output (DO) reflects the selected (addressed) word. The above description assumes an active High EN, WE, RST, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

This element can be initialized during configuration. Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs	
EN	**RST**	**WE**	**CLK**	**ADDR**	**DI**	**DO**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address. RAM(addr)=RAM contents at address ADDR. data=RAM input data.							

Specifying Initial Contents of a Block RAM -

You can use the INIT_xx attributes to specify an initial value during device configuration. The initialization of each of these elements is set by 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Design Entry Method

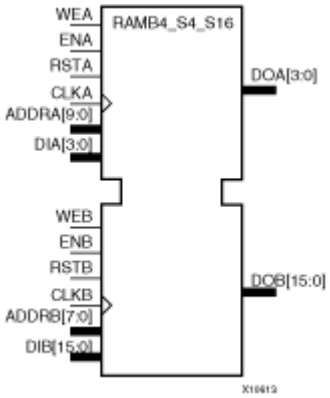
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S4_S16

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 4-bits and 16-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S4_S16	1024	4	(9:0)	(3:0)	256	16	(7:0)	(15:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
4	1024	<-----	3				2				1				0			
16	256	<-----	0															

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

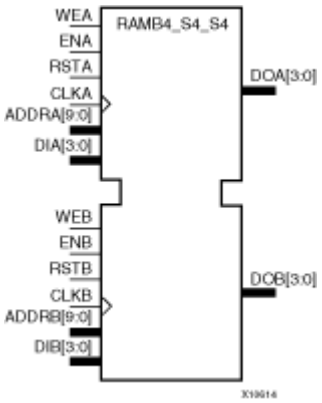
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S4_S4

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 4-bits and 4-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S4_S4	1024	4	(9:0)	(3:0)	1024	4	(9:0)	(3:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
4	1024	<-----	3				2				1				0			

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

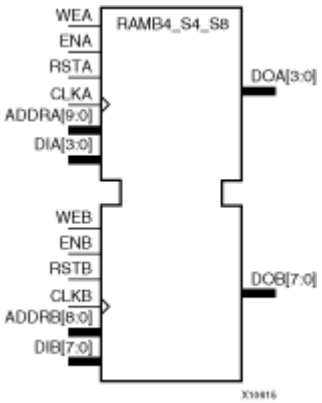
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S4_S8

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 4-bits and 8-bits



Introduction

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S4_S8	1024	4	(9:0)	(3:0)	512	8	(8:0)	(7:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
4	1024	<-----	3				2				1				0			
8	512	<-----	1								0							

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

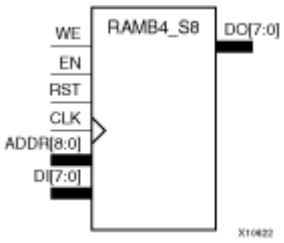
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S8

Primitive: 4k-bit Single-Port Synchronous Block RAM with Port Width Configured to 8-bits



Introduction

This design element is a dedicated, random access memory block with synchronous write capability. It provides the capability for fast, discrete, large blocks of RAM in each device. This element is configured as indicated in the following table:

Design Element	Depth	Width	Address Bus	Data Bus
RAMB4_S8	512	8	(8:0)	(7:0)

The enable (EN) pin controls read, write, and reset. When EN is Low, no data is written and the output (DO) retains the last state. When EN is High and reset (RST) is High, DO is cleared during the Low-to-High clock (CLK) transition; if write enable (WE) is High, the memory contents reflect the data at DI. When EN is High and WE is Low, the data stored in the RAM address (ADDR) is read during the Low-to-High clock transition. When EN and WE are High, the data on the data input (DI) is loaded into the word selected by the write address (ADDR) during the Low-to-High clock transition and the data output (DO) reflects the selected (addressed) word. The above description assumes an active High EN, WE, RST, and CLK. However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

This element can be initialized during configuration. Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

Logic Table

Inputs						Outputs	
EN	**RST**	**WE**	**CLK**	**ADDR**	**DI**	**DO**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address. RAM(addr)=RAM contents at address ADDR. data=RAM input data.							

Specifying Initial Contents of a Block RAM -

You can use the INIT_xx attributes to specify an initial value during device configuration. The initialization of each of these elements is set by 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Design Entry Method

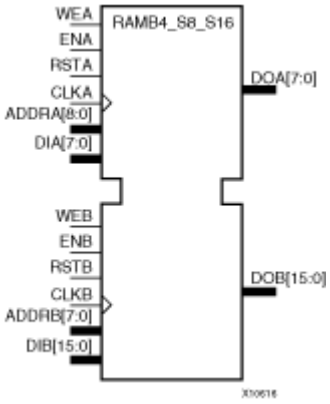
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S8_S16

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 8-bits and 16-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S8_S16	512	8	(8:0)	(7:0)	256	16	(7:0)	(15:0)
ADDR=address bus for the port. DI=data input bus for the port.								

All Port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All Port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																	
8	512	<-----	1								0							
16	256	<-----	0															

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

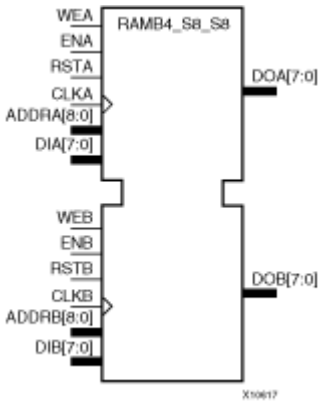
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

RAMB4_S8_S8

Primitive: 4K-bit Dual-Port Synchronous Block RAM with Port Widths Configured to 8-bits



Introduction

This design element is a 4096-bit dual-ported dedicated random access memory block with synchronous write capability. Each port is independent of the other while accessing the same set of 4096 memory cells. Each port is independently configured to a specific data width, as expressed in the following table:

Design Element	Port A Depth	Port A Width	Port A ADDR	Port A DI	Port B Depth	Port B Width	Port B ADDR	Port B DI
RAMB4_S8_S8	512	8	(8:0)	(7:0)	512	8	(8:0)	(7:0)
ADDR=address bus for the port DI=data input bus for the port								

Each port is fully synchronous with independent clock pins. All port A input pins have setup time referenced to the CLKA pin and its data output bus DOA has a clock-to-out time referenced to the CLKA. All port B input pins have setup time referenced to the CLKB pin and its data output bus DOB has a clock-to-out time referenced to the CLKB.

The enable ENA pin controls read, write, and reset for port A. When ENA is Low, no data is written and the output (DOA) retains the last state. When ENA is High and reset (RSTA) is High, DOA is cleared during the Low-to-High clock (CLKA) transition; if write enable (WEA) is High, the memory contents reflect the data at DIA. When ENA is High and WEA is Low, the data stored in the RAM address (ADDRA) is read during the Low-to-High clock transition. When ENA and WEA are High, the data on the data input (DIA) is loaded into the word selected by the write address (ADDRA) during the Low-to-High clock transition and the data output (DOA) reflects the selected (addressed) word.

The enable ENB pin controls read, write, and reset for port B. When ENB is Low, no data is written and the output (DOB) retains the last state. When ENB is High and reset (RSTB) is High, DOB is cleared during the Low-to-High clock (CLKB) transition; if write enable (WEB) is High, the memory contents reflect the data at DIB. When ENB is High and WEB is Low, the data stored in the RAM address (ADDRB) is read during the Low-to-High clock transition. When ENB and WEB are High, the data on the data input (DIB) is loaded into the word selected by the write address (ADDRB) during the Low-to-High clock transition and the data output (DOB) reflects the selected (addressed) word.

The above descriptions assume active High control pins (ENA, WEA, RSTA, CLKA, ENB, WEB, RSTB, and CLKB). However, the active level can be changed by placing an inverter on the port. Any inverter placed on a RAMB4 port is absorbed into the block and does not use a CLB resource.

Block RAM output registers are asynchronously cleared, output Low, when power is applied. The initial contents of the block RAM are not altered. For FPGA devices, power-on conditions are simulated when global set/reset (GSR) is active. GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the appropriate STARTUP_*architecture* symbol.

You can use the INIT_0x attributes to specify an initial value during device configuration. There are 16 initialization attributes (INIT_00 through INIT_0F) of 64 hex values for a total of 4096 bits. If any INIT_0x attribute is not specified, it is configured as zeros. Partial initialization strings are padded with zeros to the left.

Logic Table

Inputs						Outputs	
EN(A/B)	**RST(A/B)**	**WE(A/B)**	**CLK(A/B)**	**ADDR(A/B)**	**DI(A/B)**	**DO(A/B)**	**RAM Contents**
0	X	X	X	X	X	No Change	No Change
1	1	0	↑	X	X	0	No Change
1	1	1	↑	addr	data	0	RAM(addr) =>data
1	0	0	↑	addr	X	RAM(addr)	No Change
1	0	1	↑	addr	data	data	RAM(addr) =>data
addr=RAM address of port A/B RAM(addr)=RAM contents at address ADDRA/ADDRB data=RAM input data at pins DIA/DIB							

Port Descriptions

Address Mapping - Each port accesses the same set of 4096 memory cells using an addressing scheme that is dependent on the width of the port. The physical RAM location that is addressed for a particular width is determined from the following formula.

- Start=((ADDR port+1)*(Widthport)) -1
- End=(ADDRport)*(Widthport)

Port Width	Port Addresses																
8	512	<-----	1								0						

Port Conflict resolution - This design element is a true dual-ported RAM in that it allows simultaneous reads of the same memory cell. When one port is performing a write to a given memory cell, the other port should not address that memory cell (for a write or a read) within the clock-to-clock setup window.

- If both ports write to the same memory cell simultaneously, violating the clock-to-setup requirement, the data stored will be invalid.
- If one port attempts to read from the same memory cell that the other is simultaneously writing to, violating the clock setup requirement, the write will be successful but the data read will be invalid.

Design Entry Method

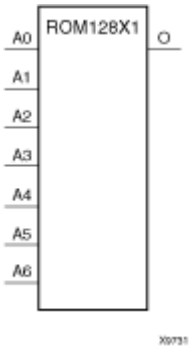
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ROM128X1

Primitive: 128-Deep by 1-Wide ROM



Introduction

This design element is a 128-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 7-bit address (A6 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of 32 hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H. An error occurs if the INIT=value is not specified.

Logic Table

Input				Output
I0	**I1**	**I2**	**I3**	**O**
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

Design Entry Method

This design element can be used in schematics.

Available Attributes

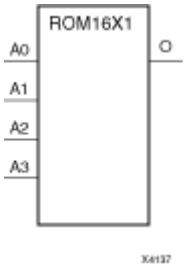
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 128-Bit Value	All zeros	Specifies the contents of the ROM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ROM16X1

Primitive: 16-Deep by 1-Wide ROM



Introduction

This design element is a 16-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 4-bit address (A3 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of four hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H. For example, the INIT=10A7 parameter produces the data stream: 0001 0000 1010 0111 An error occurs if the INIT=value is not specified.

Logic Table

Input				Output
I0	**I1**	**I2**	**I3**	**O**
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

Design Entry Method

This design element can be used in schematics.

Available Attributes

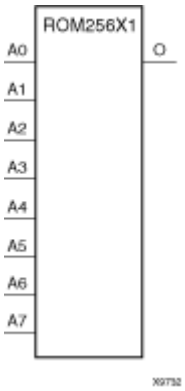
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Specifies the contents of the ROM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ROM256X1

Primitive: 256-Deep by 1-Wide ROM



Introduction

This design element is a 256-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 8-bit address (A7– A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of 64 hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H.

An error occurs if the INIT=value is not specified.

Logic Table

Input				Output
I0	**I1**	**I2**	**I3**	**O**
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

Design Entry Method

This design element can be used in schematics.

Available Attributes

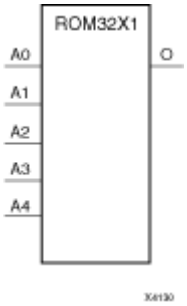
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 256-Bit Value	All zeros	Specifies the contents of the ROM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ROM32X1

Primitive: 32-Deep by 1-Wide ROM



Introduction

This design element is a 32-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 5-bit address (A4 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of eight hexadecimal digits that are written into the ROM from the most-significant digit A=1FH to the least-significant digit A=00H.

For example, the INIT=10A78F39 parameter produces the data stream: 0001 0000 1010 0111 1000 1111 0011 1001 An error occurs if the INIT=value is not specified.

Logic Table

Input				Output
I0	**I1**	**I2**	**I3**	**O**
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

Design Entry Method

This design element can be used in schematics.

Available Attributes

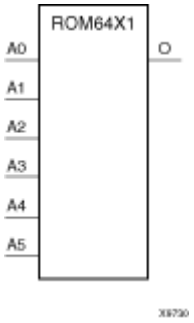
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 32-Bit Value	All zeros	Specifies the contents of the ROM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

ROM64X1

Primitive: 64-Deep by 1-Wide ROM



Introduction

This design element is a 64-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 6-bit address (A5 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of 16 hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H. An error occurs if the INIT=value is not specified.

Logic Table

Input				Output
I0	**I1**	**I2**	**I3**	**O**
0	0	0	0	INIT(0)
0	0	0	1	INIT(1)
0	0	1	0	INIT(2)
0	0	1	1	INIT(3)
0	1	0	0	INIT(4)
0	1	0	1	INIT(5)
0	1	1	0	INIT(6)
0	1	1	1	INIT(7)
1	0	0	0	INIT(8)
1	0	0	1	INIT(9)
1	0	1	0	INIT(10)
1	0	1	1	INIT(11)
1	1	0	0	INIT(12)
1	1	0	1	INIT(13)
1	1	1	0	INIT(14)
1	1	1	1	INIT(15)

Design Entry Method

This design element can be used in schematics.

Available Attributes

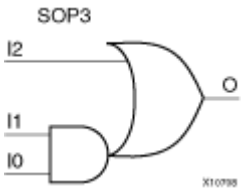
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 64-Bit Value	All zeros	Specifies the contents of the ROM.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP3

Macro: Sum of Products



Introduction

Three input Sum of Products (SOP) macros provide common logic functions by OR gating the output of one AND function with one direct input. Variations of inverting and non-inverting inputs are available.

Design Entry Method

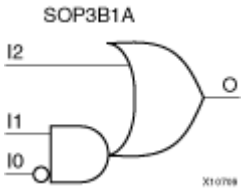
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP3B1A

Macro: Sum of Products



Introduction

Three input Sum of Products (SOP) macros provide common logic functions by OR gating the output of one AND function with one direct input. Variations of inverting and non-inverting inputs are available.

Design Entry Method

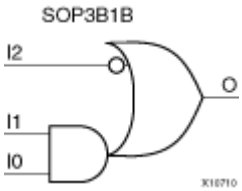
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP3B1B

Macro: Sum of Products



Introduction

Three input Sum of Products (SOP) macros provide common logic functions by OR gating the output of one AND function with one direct input. Variations of inverting and non-inverting inputs are available.

Design Entry Method

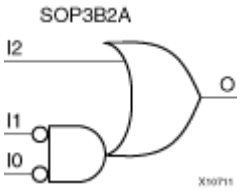
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP3B2A

Macro: Sum of Products



Introduction

Three input Sum of Products (SOP) macros provide common logic functions by OR gating the output of one AND function with one direct input. Variations of inverting and non-inverting inputs are available.

Design Entry Method

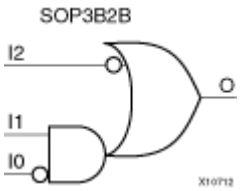
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP3B2B

Macro: Sum of Products



Introduction

Three input Sum of Products (SOP) macros provide common logic functions by OR gating the output of one AND function with one direct input. Variations of inverting and non-inverting inputs are available.

Design Entry Method

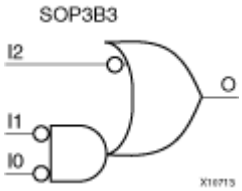
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP3B3

Macro: Sum of Products



Introduction

Three input Sum of Products (SOP) macros provide common logic functions by OR gating the output of one AND function with one direct input. Variations of inverting and non-inverting inputs are available.

Design Entry Method

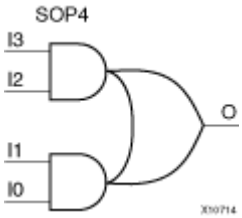
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP4

Macro: Sum of Products



Introduction

Four input Sum of Products (SOP) macros provide common logic functions by OR gating the outputs of two AND functions. Variations of inverting and non-inverting inputs are available.

Design Entry Method

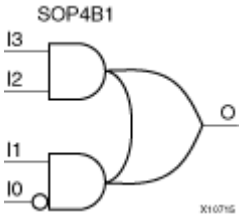
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP4B1

Macro: Sum of Products



Introduction

Four input Sum of Products (SOP) macros provide common logic functions by OR gating the outputs of two AND functions. Variations of inverting and non-inverting inputs are available.

Design Entry Method

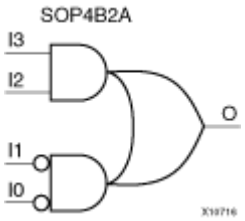
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP4B2A

Macro: Sum of Products



Introduction

Four input Sum of Products (SOP) macros provide common logic functions by OR gating the outputs of two AND functions. Variations of inverting and non-inverting inputs are available.

Design Entry Method

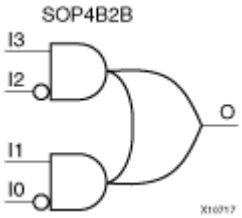
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP4B2B

Macro: Sum of Products



Introduction

Four input Sum of Products (SOP) macros provide common logic functions by OR gating the outputs of two AND functions. Variations of inverting and non-inverting inputs are available.

Design Entry Method

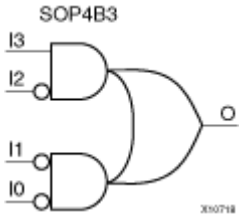
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP4B3

Macro: Sum of Products



Introduction

Four input Sum of Products (SOP) macros provide common logic functions by OR gating the outputs of two AND functions. Variations of inverting and non-inverting inputs are available.

Design Entry Method

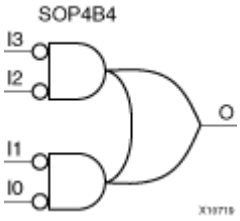
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SOP4B4

Macro: Sum of Products



Introduction

Four input Sum of Products (SOP) macros provide common logic functions by OR gating the outputs of two AND functions. Variations of inverting and non-inverting inputs are available.

Design Entry Method

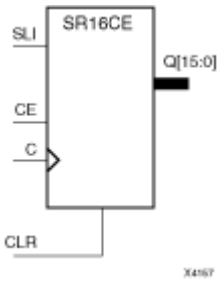
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR16CE

Macro: 16-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with a shift-left serial input (SLI), parallel outputs (Q), and clock enable (CE) and asynchronous clear (CLR) inputs. The (CLR) input, when High, overrides all other inputs and resets the data outputs (Q) Low. When (CE) is High and (CLR) is Low, the data on the SLI input is loaded into the first bit of the shift register during the Low-to- High clock (C) transition and appears on the (Q0) output. During subsequent Low-to- High clock transitions, when (CE) is High and (CLR) is Low, data shifts to the next highest bit position as new data is loaded into (Q0) (SLI→Q0, Q0→Q1, Q1→Q2, and so forth). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), and (CLR) in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs				Outputs	
CLR	**CE**	**SLI**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1
z = bit width - 1 qn-1 = state of referenced output one setup time prior to active clock transition					

Design Entry Method

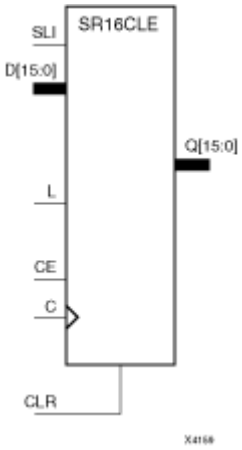
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR16CLE

Macro: 16-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with a shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and asynchronous clear (CLR) . The register ignores clock transitions when (L) and (CE) are Low. The asynchronous (CLR), when High, overrides all other inputs and resets the data outputs (Q) Low. When (L) is High and (CLR) is Low, data on the Dn – D0 inputs is loaded into the corresponding Qn – (Q0) bits of the register.

When (CE) is High and (L) and (CLR) are Low, data on the SLI input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent clock transitions, when (CE) is High and (L) and (CLR) are Low, the data shifts to the next highest bit position as new data is loaded into (Q)0 (for example, SLI→Q0, Q0→Q1, and Q1→Q2).

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (CLR) inputs in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs						Outputs	
CLR	**L**	**CE**	**SLI**	**Dn – D0**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	X	X	0	0
0	1	X	X	Dn – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change
z = bitwidth -1							

qn-1 = state of referenced output one setup time prior to active clock transition

Design Entry Method

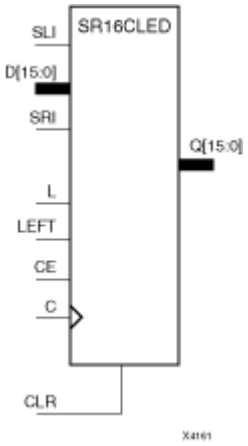
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR16CLED

Macro: 16-Bit Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q), and four control inputs: clock enable (CE), load enable (L), shift left/right (LEFT), and asynchronous clear (CLR). The register ignores clock transitions when (CE) and (L) are Low. The asynchronous clear, when High, overrides all other inputs and resets the data outputs (Qn) Low.

When (L) is High and (CLR) is Low, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register. When (CE) is High and (L) and (CLR) are Low, data is shifted right or left, depending on the state of the LEFT input. If LEFT is High, data on the SLI is loaded into (Q0) during the Low-to-High clock transition and shifted left (for example, to Q1 or Q2) during subsequent clock transitions. If LEFT is Low, data on the SRI is loaded into the last (Q) output during the Low-to-High clock transition and shifted right during subsequent clock transitions. The logic tables indicate the state of the (Q) outputs under all input conditions.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs								Outputs		
CLR	**L**	**CE**	**LEFT**	**SLI**	**SRI**	**D15 – D0**	**C**	**Q0**	**Q15**	**Q14 – Q1**
1	X	X	X	X	X	X	X	0	0	0
0	1	X	X	X	X	D15 – D0	↑	D0	D15	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q14	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1
qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition.										

Design Entry Method

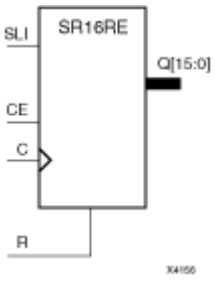
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR16RE

Macro: 16-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left serial input (SLI), parallel outputs (Qn), clock enable (CE), and synchronous reset (R) inputs. The R input, when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low.

When (CE) is High and (R) is Low, the data on the (SLI) is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent Low-to-High clock transitions, when (CE) is High and R is Low, data shifts to the next highest bit position as new data is loaded into (Q0) (for example, SLI→Q0, Q0→Q1, and Q1→Q2). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), and (R) in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs				Outputs	
R	**CE**	**SLI**	**C**	**Q0**	**Qz – Q1**
1	X	X	↑	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1
z = bitwidth -1 qn-1 = state of referenced output one setup time prior to active clock transition					

Design Entry Method

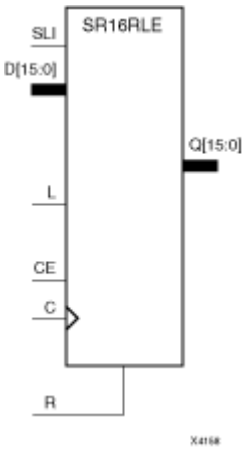
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR16RLE

Macro: 16-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and synchronous reset (R). The register ignores clock transitions when (L) and (CE) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, data on the (D) inputs is loaded into the corresponding Q bits of the register.

When (CE) is High and (L) and (R) are Low, data on the (SLI) input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the Q0 output. During subsequent clock transitions, when (CE) is High and (L) and (R) are Low, the data shifts to the next highest bit position as new data is loaded into Q0.

Registers can be cascaded by connecting the last Q output of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (R) inputs in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs						Outputs	
R	**L**	**CE**	**SLI**	**Dz – D0**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	X	↑	0	0
0	1	X	X	Dz – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change
z = bitwidth -1 qn-1 = state of referenced output one setup time prior to active clock transition							

Design Entry Method

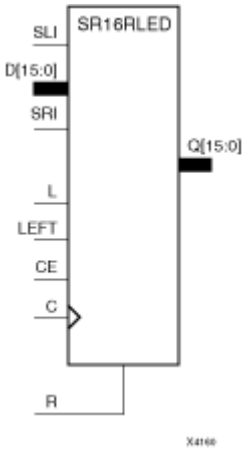
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR16RLED

Macro: 16-Bit Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q) and four control inputs — clock enable (CE), load enable (L), shift left/right (LEFT), and synchronous reset (R). The register ignores clock transitions when (CE) and (L) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register.

When (CE) is High and (L) and (R) are Low, data shifts right or left, depending on the state of the LEFT input. If LEFT is High, data on (SLI) is loaded into (Q0) during the Low-to-High clock transition and shifted left (for example, to Q1 and Q2) during subsequent clock transitions. If LEFT is Low, data on the (SRI) is loaded into the last (Q) output during the Low-to-High clock transition and shifted right) during subsequent clock transitions. The logic tables below indicates the state of the (Q) outputs under all input conditions.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs								**Outputs**		
R	**L**	**CE**	**LEFT**	**SLI**	**SRI**	**D15 – D0**	**C**	**Q0**	**Q15**	**Q14 – Q1**
1	X	X	X	X	X	X	↑	0	0	0
0	1	X	X	X	X	D15 – D0	↑	D0	D15	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q14	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1
qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition										

Design Entry Method

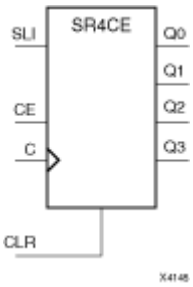
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR4CE

Macro: 4-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with a shift-left serial input (SLI), parallel outputs (Q), and clock enable (CE) and asynchronous clear (CLR) inputs. The (CLR) input, when High, overrides all other inputs and resets the data outputs (Q) Low. When (CE) is High and (CLR) is Low, the data on the SLI input is loaded into the first bit of the shift register during the Low-to- High clock (C) transition and appears on the (Q0) output. During subsequent Low-to- High clock transitions, when (CE) is High and (CLR) is Low, data shifts to the next highest bit position as new data is loaded into (Q0) (SLI→Q0, Q0→Q1, Q1→Q2, and so forth). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), and (CLR) in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs				Outputs	
CLR	**CE**	**SLI**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1

z = bit width - 1

qn-1 = state of referenced output one setup time prior to active clock transition

Design Entry Method

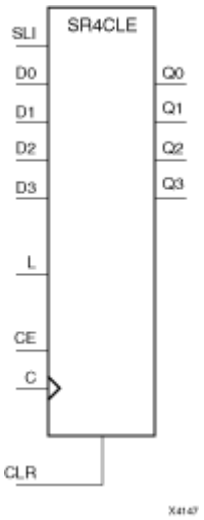
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR4CLE

Macro: 4-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with a shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and asynchronous clear (CLR) . The register ignores clock transitions when (L) and (CE) are Low. The asynchronous (CLR), when High, overrides all other inputs and resets the data outputs (Q) Low. When (L) is High and (CLR) is Low, data on the Dn – D0 inputs is loaded into the corresponding Qn – (Q0) bits of the register.

When (CE) is High and (L) and (CLR) are Low, data on the SLI input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent clock transitions, when (CE) is High and (L) and (CLR) are Low, the data shifts to the next highest bit position as new data is loaded into (Q)0 (for example, SLI→Q0, Q0→Q1, and Q1→Q2).

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (CLR) inputs in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs						Outputs	
CLR	**L**	**CE**	**SLI**	**Dn – D0**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	X	X	0	0
0	1	X	X	Dn – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change

z = bitwidth -1

qn-1 = state of referenced output one setup time prior to active clock transition

Design Entry Method

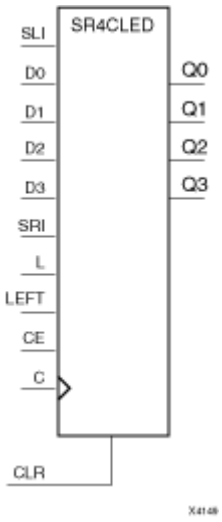
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR4CLED

Macro: 4-Bit Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q), and four control inputs: clock enable (CE), load enable (L), shift left/right (LEFT), and asynchronous clear (CLR). The register ignores clock transitions when (CE) and (L) are Low. The asynchronous clear, when High, overrides all other inputs and resets the data outputs (Qn) Low.

When (L) is High and (CLR) is Low, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register. When (CE) is High and (L) and (CLR) are Low, data is shifted right or left, depending on the state of the LEFT input. If LEFT is High, data on the SLI is loaded into (Q0) during the Low-to-High clock transition and shifted left (for example, to Q1 or Q2) during subsequent clock transitions. If LEFT is Low, data on the SRI is loaded into the last (Q) output during the Low-to-High clock transition and shifted right during subsequent clock transitions. The logic tables indicate the state of the (Q) outputs under all input conditions.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs								**Outputs**		
CLR	**L**	**CE**	**LEFT**	**SLI**	**SRI**	**D3 – D0**	**C**	**Q0**	**Q3**	**Q2 – Q1**
1	X	X	X	X	X	X	X	0	0	0
0	1	X	X	X	X	D3– D0	↑	D0	D3	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q2	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1
qn-1 and qn+1 = state of referenced output one setup time prior to active clock transition.										

Design Entry Method

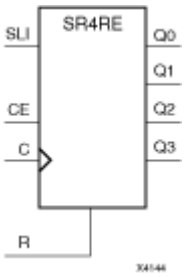
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR4RE

Macro: 4-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left serial input (SLI), parallel outputs (Qn), clock enable (CE), and synchronous reset (R) inputs. The R input, when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low.

When (CE) is High and (R) is Low, the data on the (SLI) is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent Low-to-High clock transitions, when (CE) is High and R is Low, data shifts to the next highest bit position as new data is loaded into (Q0) (for example, SLI→Q0, Q0→Q1, and Q1→Q2). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), and (R) in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs				Outputs	
R	**CE**	**SLI**	**C**	**Q0**	**Qz – Q1**
1	X	X	↑	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1
z = bitwidth -1 qn-1 = state of referenced output one setup time prior to active clock transition					

Design Entry Method

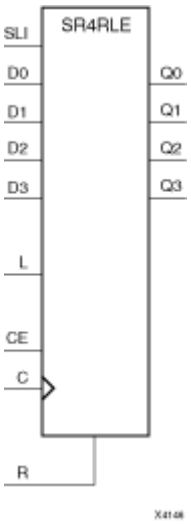
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR4RLE

Macro: 4-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and synchronous reset (R). The register ignores clock transitions when (L) and (CE) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, data on the (D) inputs is loaded into the corresponding Q bits of the register.

When (CE) is High and (L) and (R) are Low, data on the (SLI) input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the Q0 output. During subsequent clock transitions, when (CE) is High and (L) and (R) are Low, the data shifts to the next highest bit position as new data is loaded into Q0.

Registers can be cascaded by connecting the last Q output of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (R) inputs in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs						Outputs	
R	**L**	**CE**	**SLI**	**Dz – D0**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	X	↑	0	0
0	1	X	X	Dz – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change
z = bitwidth -1 qn-1 = state of referenced output one setup time prior to active clock transition							

Design Entry Method

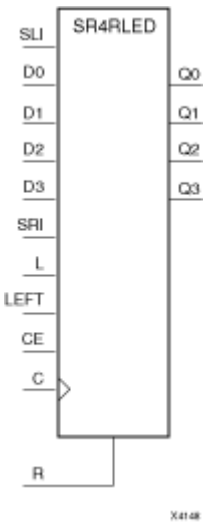
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR4RLED

Macro: 4-Bit Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q) and four control inputs — clock enable (CE), load enable (L), shift left/right (LEFT), and synchronous reset (R). The register ignores clock transitions when (CE) and (L) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register.

When (CE) is High and (L) and (R) are Low, data shifts right or left, depending on the state of the LEFT input. If LEFT is High, data on (SLI) is loaded into (Q0) during the Low-to-High clock transition and shifted left (for example, to Q1 and Q2) during subsequent clock transitions. If LEFT is Low, data on the (SRI) is loaded into the last (Q) output during the Low-to-High clock transition and shifted right) during subsequent clock transitions. The logic tables below indicates the state of the (Q) outputs under all input conditions.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs								Outputs		
R	**L**	**CE**	**LEFT**	**SLI**	**SRI**	**D3 – D0**	**C**	**Q0**	**Q3**	**Q2 – Q1**
1	X	X	X	X	X	X	↑	0	0	0
0	1	X	X	X	X	D3 – D0	↑	D0	D3	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q2	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1
qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition										

Design Entry Method

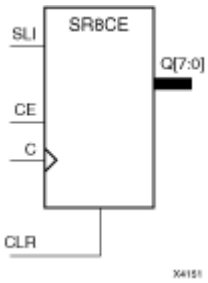
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR8CE

Macro: 8-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with a shift-left serial input (SLI), parallel outputs (Q), and clock enable (CE) and asynchronous clear (CLR) inputs. The (CLR) input, when High, overrides all other inputs and resets the data outputs (Q) Low. When (CE) is High and (CLR) is Low, the data on the SLI input is loaded into the first bit of the shift register during the Low-to- High clock (C) transition and appears on the (Q0) output. During subsequent Low-to- High clock transitions, when (CE) is High and (CLR) is Low, data shifts to the next highest bit position as new data is loaded into (Q0) (SLI→Q0, Q0→Q1, Q1→Q2, and so forth). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), and (CLR) in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs				Outputs	
CLR	**CE**	**SLI**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1
z = bit width - 1 qn-1 = state of referenced output one setup time prior to active clock transition					

Design Entry Method

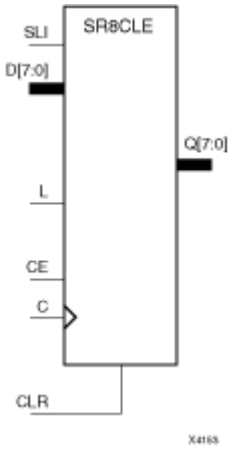
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR8CLE

Macro: 8-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with a shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and asynchronous clear (CLR) . The register ignores clock transitions when (L) and (CE) are Low. The asynchronous (CLR), when High, overrides all other inputs and resets the data outputs (Q) Low. When (L) is High and (CLR) is Low, data on the Dn – D0 inputs is loaded into the corresponding Qn – (Q0) bits of the register.

When (CE) is High and (L) and (CLR) are Low, data on the SLI input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent clock transitions, when (CE) is High and (L) and (CLR) are Low, the data shifts to the next highest bit position as new data is loaded into (Q)0 (for example, SLI→Q0, Q0→Q1, and Q1→Q2).

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (CLR) inputs in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs						Outputs	
CLR	**L**	**CE**	**SLI**	**Dn – D0**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	X	X	0	0
0	1	X	X	Dn – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change
z = bitwidth -1							

qn-1 = state of referenced output one setup time prior to active clock transition

Design Entry Method

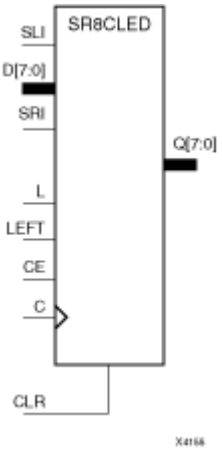
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR8CLED

Macro: 8-Bit Shift Register with Clock Enable and Asynchronous Clear



Introduction

This design element is a shift register with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q), and four control inputs: clock enable (CE), load enable (L), shift left/right (LEFT), and asynchronous clear (CLR). The register ignores clock transitions when (CE) and (L) are Low. The asynchronous clear, when High, overrides all other inputs and resets the data outputs (Qn) Low.

When (L) is High and (CLR) is Low, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register. When (CE) is High and (L) and (CLR) are Low, data is shifted right or left, depending on the state of the LEFT input. If LEFT is High, data on the SLI is loaded into (Q0) during the Low-to-High clock transition and shifted left (for example, to Q1 or Q2) during subsequent clock transitions. If LEFT is Low, data on the SRI is loaded into the last (Q) output during the Low-to-High clock transition and shifted right during subsequent clock transitions. The logic tables indicate the state of the (Q) outputs under all input conditions.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs								**Outputs**		
CLR	**L**	**CE**	**LEFT**	**SLI**	**SRI**	**D7 – D0**	**C**	**Q0**	**Q7**	**Q6 – Q1**
1	X	X	X	X	X	X	X	0	0	0
0	1	X	X	X	X	D7 – D0	↑	D0	D7	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q6	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1
qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition.										

Design Entry Method

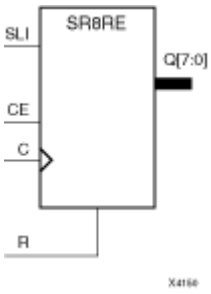
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR8RE

Macro: 8-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left serial input (SLI), parallel outputs (Qn), clock enable (CE), and synchronous reset (R) inputs. The R input, when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low.

When (CE) is High and (R) is Low, the data on the (SLI) is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent Low-to-High clock transitions, when (CE) is High and R is Low, data shifts to the next highest bit position as new data is loaded into (Q0) (for example, SLI→Q0, Q0→Q1, and Q1→Q2). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output of one stage to the SLI input of the next stage and connecting clock, (CE), and (R) in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs				Outputs	
R	**CE**	**SLI**	**C**	**Q0**	**Qz – Q1**
1	X	X	↑	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1
z = bitwidth -1 qn-1 = state of referenced output one setup time prior to active clock transition					

Design Entry Method

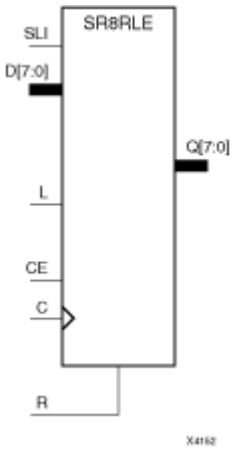
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR8RLE

Macro: 8-Bit Loadable Serial/Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and synchronous reset (R). The register ignores clock transitions when (L) and (CE) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, data on the (D) inputs is loaded into the corresponding Q bits of the register.

When (CE) is High and (L) and (R) are Low, data on the (SLI) input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the Q0 output. During subsequent clock transitions, when (CE) is High and (L) and (R) are Low, the data shifts to the next highest bit position as new data is loaded into Q0.

Registers can be cascaded by connecting the last Q output of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (R) inputs in parallel.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs						**Outputs**	
R	**L**	**CE**	**SLI**	**Dz – D0**	**C**	**Q0**	**Qz – Q1**
1	X	X	X	X	↑	0	0
0	1	X	X	Dz – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change
z = bitwidth -1 qn-1 = state of referenced output one setup time prior to active clock transition							

Design Entry Method

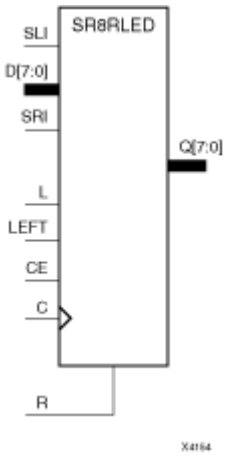
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SR8RLED

Macro: 8-Bit Shift Register with Clock Enable and Synchronous Reset



Introduction

This design element is a shift register with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q) and four control inputs — clock enable (CE), load enable (L), shift left/right (LEFT), and synchronous reset (R). The register ignores clock transitions when (CE) and (L) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register.

When (CE) is High and (L) and (R) are Low, data shifts right or left, depending on the state of the LEFT input. If LEFT is High, data on (SLI) is loaded into (Q0) during the Low-to-High clock transition and shifted left (for example, to Q1 and Q2) during subsequent clock transitions. If LEFT is Low, data on the (SRI) is loaded into the last (Q) output during the Low-to-High clock transition and shifted right) during subsequent clock transitions. The logic tables below indicates the state of the (Q) outputs under all input conditions.

This register is asynchronously cleared, outputs Low, when power is applied.

Logic Table

Inputs								Outputs		
R	**L**	**CE**	**LEFT**	**SLI**	**SRI**	**D7– D0**	**C**	**Q0**	**Q7**	**Q6 – Q1**
1	X	X	X	X	X	X	↑	0	0	0
0	1	X	X	X	X	D7 – D0	↑	D0	D7	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q6	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1
qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition										

Design Entry Method

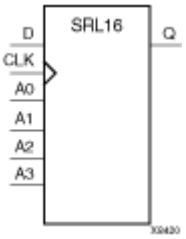
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRL16

Primitive: 16-Bit Shift Register Look-Up-Table (LUT)



Introduction

This design element is a shift register look-up table (LUT). The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. During subsequent Low-to-High clock transitions data shifts to the next highest bit position while new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

Logic Table

Inputs			Output
Am	**CLK**	**D**	**Q**
Am	X	X	Q(Am)
Am	↑	D	Q(Am - 1)
m= 0, 1, 2, 3			

Design Entry Method

This design element can be used in schematics.

Available Attributes

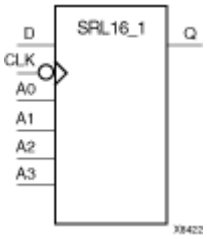
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of Q output after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRL16_1

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock



Introduction

This design element is a shift register look-up table (LUT). The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the High-to-Low clock (CLK) transition. During subsequent High-to-Low clock transitions data shifts to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

Logic Table

Inputs			Output
Am	**CLK**	**D**	**Q**
Am	X	X	Q(Am)
Am	↓	D	Q(Am - 1)
m= 0, 1, 2, 3			

Design Entry Method

This design element can be used in schematics.

Available Attributes

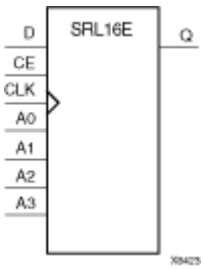
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of Q output after configuration

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRL16E

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Clock Enable



Introduction

This design element is a shift register look-up table (LUT). The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

When CE is High, the data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. During subsequent Low-to-High clock transitions, when CE is High, data shifts to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached. When CE is Low, the register ignores clock transitions.

Logic Table

Inputs				Output
Am	**CE**	**CLK**	**D**	**Q**
Am	0	X	X	Q(Am)
Am	1	↑	D	Q(Am - 1)
m= 0, 1, 2, 3				

Design Entry Method

This design element can be used in schematics.

Available Attributes

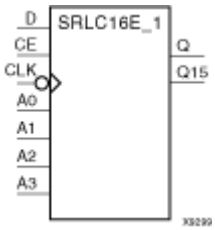
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of content and output of shift register after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRL16E_1

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock and Clock Enable



Introduction

This design element is a shift register look up table (LUT) with clock enable (CE). The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

When CE is High, the data (D) is loaded into the first bit of the shift register during the High-to-Low clock (CLK) transition. During subsequent High-to-Low clock transitions, when CE is High, data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached. When CE is Low, the register ignores clock transitions.

Logic Table

Inputs				Output
Am	**CE**	**CLK**	**D**	**Q**
Am	0	X	X	Q(Am)
Am	1	↓	D	Q(Am - 1)
m= 0, 1, 2, 3				

Design Entry Method

This design element can be used in schematics.

Available Attributes

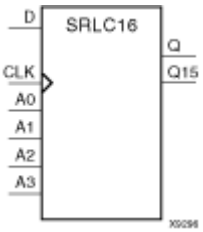
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of content and output of shift register after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRLC16

Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry



Introduction

This design element is a shift register look-up table (LUT) with Carry. The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. During subsequent Low-to-High clock transitions data shifts to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached. The Q15 output is available for you in cascading to multiple shift register LUTs to create larger shift registers.

Logic Table

Inputs			Output
Am	**CLK**	**D**	**Q**
Am	X	X	Q(Am)
Am	↑	D	Q(Am - 1)
m= 0, 1, 2, 3			

Design Entry Method

This design element can be used in schematics.

Available Attributes

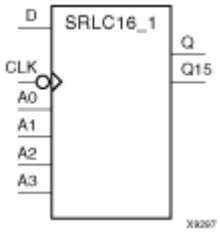
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of content and output of shift register after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRLC16_1

Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry and Negative-Edge Clock



Introduction

This design element is a shift register look-up table (LUT) with carry and a negative-edge clock. The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The Q15 output is available for your use in cascading multiple shift register LUTs to create larger shift registers.

Logic Table

Inputs			Output	
Am	**CLK**	**D**	**Q**	**Q15**
Am	X	X	Q(Am)	No Change
Am	↓	D	Q(Am - 1)	Q14
m= 0, 1, 2, 3				

Design Entry Method

This design element can be used in schematics.

Available Attributes

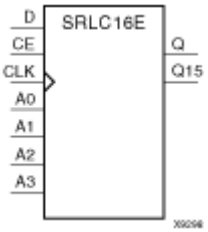
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of content and output of shift register after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRLC16E

Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry and Clock Enable



Introduction

This design element is a shift register look-up table (LUT) with carry and clock enable. The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. When CE is High, during subsequent Low-to-High clock transitions, data shifts to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

The Q15 output is available for you in cascading to multiple shift register LUTs to create larger shift registers.

Logic Table

Inputs				Output	
Am	**CLK**	**CE**	**D**	**Q**	**Q15**
Am	X	0	X	Q(Am)	Q(15)
Am	X	1	X	Q(Am)	Q(15)
Am	↑	1	D	Q(Am - 1)	Q15
m= 0, 1, 2, 3					

Design Entry Method

This design element can be used in schematics.

Available Attributes

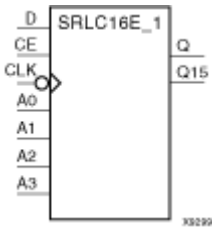
Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of content and output of shift register after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

SRLC16E_1

Primitive: 16-Bit Shift Register Look-Up Table (LUT) with Carry, Negative-Edge Clock, and Clock Enable



Introduction

This design element is a shift register look-up table (LUT) with carry, clock enable, and negative-edge clock. The inputs A3, A2, A1, and A0 select the output length of the shift register.

The shift register can be of a fixed, static length or it can be dynamically adjusted.

- **To create a fixed-length shift register -** Drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits, as determined by the following formula: Length = (8 x A3) +(4 x A2) + (2 x A1) + A0 +1 If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.
- **To change the length of the shift register dynamically -** Change the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits. Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

When CE is High, the data (D) is loaded into the first bit of the shift register during the High-to-Low clock (CLK) transition. During subsequent High-to-Low clock transitions data shifts to the next highest bit position as new data is loaded when CE is High. The data appears on the Q output when the shift register length determined by the address inputs is reached.

The Q15 output is available for your use in cascading multiple shift register LUTs to create larger shift registers.

Logic Table

Inputs				Output	
Am	**CE**	**CLK**	**D**	**Q**	**Q15**
Am	0	X	X	Q(Am)	No Change
Am	1	X	X	Q(Am)	No Change
Am	1	↓	D	Q(Am -1)	Q14
m= 0, 1, 2, 3					

Design Entry Method

This design element can be used in schematics.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	Hexadecimal	Any 16-Bit Value	All zeros	Sets the initial value of content and output of shift register after configuration.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

STARTBUF_VIRTEX2

Primitive: VHDL Simulation of FPGA Designs

Introduction

This design element is used for VHDL simulation of FPGA designs that require the use of the STARTUP block. The difference between the STARTBUF and the STARTUP block is that the STARTBUF contains output ports which may be connected to all register set/resets in the design (GSROUT) or to all I/O three-state controls (GTSOUT) so that these functions may be functionally simulated. This design element should not be used for Verilog or schematic entry. In order to use the STARTBUF, the desired input(s) should be connected to a top-level port in the design and the corresponding output(s) must be connected to either the three-state control signal for all inferred and instantiated output buffers in the design (GTSOUT) or all inferred or instantiated register set/resets in the design.

During simulation, the inputs to the STARTBUF can be toggled by the testbench in order to activate the global three-state or global set/reset signal in the design. This should be done at the beginning of the simulation to simulate the behavior of the registers and I/O during configuration. It may also be applied during simulation to simulate a reconfiguration (ProG pin high) of the device. During synthesis and implementation, this component will be treated as a STARTUP block. The connected input ports to this component should remain in the design and be connected to the correct corresponding global resource.

The value at port GSROUT will be always the be value at port GSRIN. The value at port GTSOUT will always be the value at port GTSIN. CLKIN has no effect on simulation.

Design Entry Method

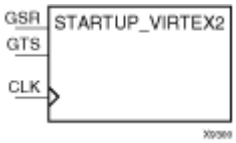
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

STARTUP_VIRTEX2

Primitive: Virtex-II, Virtex-II Pro, and Virtex-II User Interface to Global Clock, Reset, and 3-State Controls



Introduction

This design element is used for Global Set/Reset, global 3-state control, and the user configuration clock. The Global Set/Reset (GSR) input, when High, sets or resets all flip-flops, all latches, and every block RAMB16 output register in the device, depending on the initialization state (INIT=1 or 0) of the component. For Virtex and Virtex-E, see "STARTUP_VIRTEX".

Note Block RAM content, LUT RAMs, the Digital Clock Manager (DCM), and shift register LUTs (SRL16, SRL16_1, SRL16E, SRL16E_1, SRLC16, SRLC16_1, SRLC16E, and SRLC16E_1) are not set/reset.

Following configuration, the global 3-state control (GTS), when High—and BSCAN is not enabled and executing an EXTEST instruction—forces all the IOB outputs into high impedance mode, which isolates the device outputs from the circuit but leaves the inputs active.

Note GTS= Global 3-State

Including the STARTUP_VIRTEX2 symbol in a design is optional. You must include the symbol under the following conditions.

- To exert external control over global set/reset, connect the GSR pin to a top level port and an IBUF.
- To exert external control over global 3-state, connect the GTS pin to a top level port and IBUF.
- To synchronize startup to a user clock, connect the user clock signal to the CLK input. Furthermore, "user clock" must be selected in the BitGen program.

You can use location constraints to specify the pin from which GSR or GTS (or both) is accessed.

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

VCC

Primitive: VCC-Connection Signal Tag

Introduction

This design element serves as a signal tag, or parameter, that forces a net or input function to a logic High level. A net tied to this element cannot have any other source.

When the placement and routing software encounters a net or input function tied to this element, it removes any logic that is disabled by the Vcc signal, which is only implemented when the disabled logic cannot be removed.

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR2

Primitive: 2-Input XNOR Gate with Non-Inverted Inputs

Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR3

Primitive: 3-Input XNOR Gate with Non-Inverted Inputs

Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

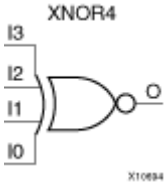
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR4

Primitive: 4-Input XNOR Gate with Non-Inverted Inputs



Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

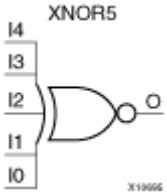
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR5

Primitive: 5-Input XNOR Gate with Non-Inverted Inputs



Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

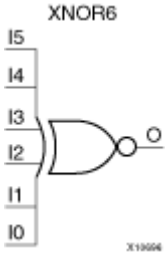
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR6

Macro: 6-Input XNOR Gate with Non-Inverted Inputs



Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

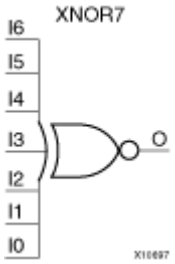
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR7

Macro: 7-Input XNOR Gate with Non-Inverted Inputs



Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

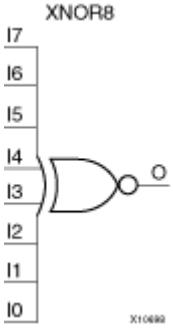
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR8

Macro: 8-Input XNOR Gate with Non-Inverted Inputs



Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

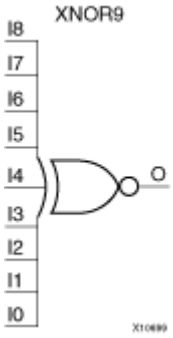
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XNOR9

Macro: 9-Input XNOR Gate with Non-Inverted Inputs



Introduction

XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Logic Table

Input	Output
I0 ... Iz	O
Odd number of 1	0
Even number of 1	1

Design Entry Method

This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR2

Primitive: 2-Input XOR Gate with Non-Inverted Inputs

Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

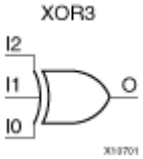
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR3

Primitive: 3-Input XOR Gate with Non-Inverted Inputs



Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

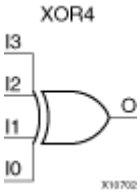
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR4

Primitive: 4-Input XOR Gate with Non-Inverted Inputs



Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

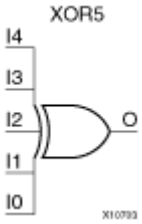
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR5

Primitive: 5-Input XOR Gate with Non-Inverted Inputs



Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

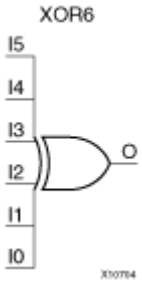
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR6

Macro: 6-Input XOR Gate with Non-Inverted Inputs



Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

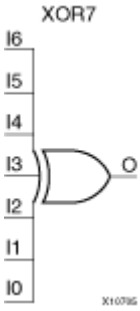
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR7

Macro: 7-Input XOR Gate with Non-Inverted Inputs



Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

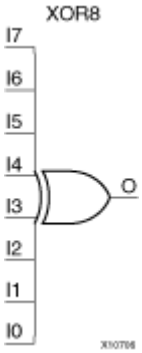
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR8

Macro: 8-Input XOR Gate with Non-Inverted Inputs



Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

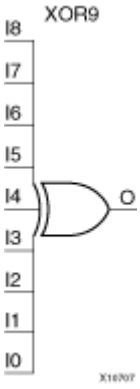
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XOR9

Macro: 9-Input XOR Gate with Non-Inverted Inputs



Introduction

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.

Design Entry Method

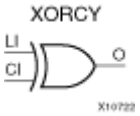
This design element is only for use in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XORCY

Primitive: XOR for Carry Logic with General Output



Introduction

This design element is a special XOR with general O output that generates faster and smaller arithmetic functions. The XORCY primitive is a dedicated XOR function within the carry-chain logic of the slice. It allows for fast and efficient creation of arithmetic (add/subtract) or wide logic functions (large AND/OR gate).

Logic Table

Input		Output
LI	CI	O
0	0	0
0	1	1
1	0	1
1	1	0

Design Entry Method

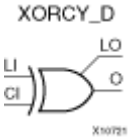
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XORCY_D

Primitive: XOR for Carry Logic with Dual Output



Introduction

This design element is a special XOR that generates faster and smaller arithmetic functions.

Logic Table

Input		Output
LI	CI	O and LO
0	0	0
0	1	1
1	0	1
1	1	0

Design Entry Method

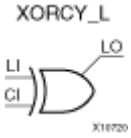
This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .

XORCY_L

Primitive: XOR for Carry Logic with Local Output



Introduction

This design element is a special XOR with local LO output that generates faster and smaller arithmetic functions.

Logic Table

Input		Output
LI	CI	LO
0	0	0
0	1	1
1	0	1
1	1	0

Design Entry Method

This design element can be used in schematics.

For More Information

- See the *Virtex-II Pro User Guide* .
- See the *Virtex-II Pro Data Sheets* .