Virtex-II Pro System Wake-Up Solutions

UG028 (v1.1) August 13, 2007

Product Not Recommended for New Designs

![Xilinx logo]

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.

Product Not Recommended for New Designs

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
03/08/02	1.0	Initial Xilinx release.
08/13/07	1.1	• Updated format and completed other minor edits. • Removed obsolete products from the System ACE family. • Replaced mature XC18V00 and XC17V00 PROM families with Platform Flash PROM.

Product Not Recommended for New Designs

Table of Contents

XILINX ®

Preface

About This Guide

The Virtex™-II Pro Platform FPGA solution is a powerful system solution with numerous advanced features, including IBM PowerPC™ processors and RocketIO™ multi-gigabit transceiver blocks. The advanced features introduce new options that need to be considered when selecting a customized system wake-up solution. The system wake-up is the process necessary to bring the Virtex-II Pro devices to an operational stage for the system application and typically consists of two steps, configuring the FPGA logic and booting up the PowerPC processor for successful system operation.

The Virtex-II Pro PowerPC (PPC) microprocessor can be initialized as part of the system wake-up. This document briefly describes the various options available for system wake-up and highlights the two most efficient solutions for basic and advanced systems.

Guide Contents

This manual contains the following chapters:

- Chapter 1, "Managing Hardware and Software Data with Virtex-II Pro Devices," gives an overview of the different methods available to configure Virtex-II Pro devices.
- Chapter 2, "Basic and Advanced Virtex-II Pro Wake-Up Solutions," describes the wake-up setup used with other Xilinx FPGA families.

Additional Resources

To find additional documentation, see the Xilinx website at:

http://www.xilinx.com/literature.

To search the Answer Database of silicon, software, and IP questions and answers, or to create a technical support WebCase, see the Xilinx website at:

http://www.xilinx.com/support.

Conventions

This document uses the following conventions. An example illustrates each convention.

Typographical

The following typographical conventions are used in this document:

Convention	Meaning or Use	Example
`Courier font`	Messages, prompts, and program files that the system displays	`speed grade: - 100`
`Courier bold`	Literal commands that you enter in a syntactical statement	**`ngdbuild`** *design_name*
Helvetica bold	Commands that you select from a menu	**File → Open**
	Keyboard shortcuts	**Ctrl+C**
Italic font	Variables in a syntax statement for which you must supply values	**`ngdbuild`** *design_name*
	References to other manuals	See the *Development System Reference Guide* for more information.
	Emphasis in text	If a wire is drawn so that it overlaps the pin of a symbol, the two nets are *not* connected.
Square brackets [ ]	An optional entry or parameter. However, in bus specifications, such as **`bus[7:0]`**, they are required.	**`ngdbuild`** [*option_name*] *design_name*
Braces { }	A list of items from which you must choose one or more	**`lowpwr ={on|off}`**
Vertical bar |	Separates items in a list of choices	**`lowpwr ={on|off}`**

Vertical ellipsis
.
.
.
	Repetitive material that has been omitted	`IOB #1: Name = QOUT'`
		`IOB #2: Name = CLKIN'`
		`.`
		`.`
		`.`

Horizontal ellipsis ...	Repetitive material that has been omitted	**`allow block`** *block_name loc1 loc2 ... locn;*

Online Document

The following conventions are used in this document:

Convention	Meaning or Use	Example
Blue text	Cross-reference link to a location in the current document	See the section *"Additional Resources"* for details. Refer to *"Title Formats" in Chapter 1* for details.
Red text	Cross-reference link to a location in another document	See Figure 2-5 in the *Virtex-II Platform FPGA User Guide.*
Blue, underlined text	Hyperlink to a website (URL)	Go to http://www.xilinx.com for the latest speed files.

XILINX ®

Chapter 1

Managing Hardware and Software Data with Virtex-II Pro Devices

Overview

This section gives an overview of the different methods available to configure the Virtex™-II Pro device and to allow the PowerPC™ (PPC) to boot up and access memory for code storage and software execution.

The three system wake-up solutions for Virtex-II Pro Platform FPGAs are:

1. System ACE™ CF – CompactFlash-based configuration management
2. XILINX PROM – Programmable Read-Only Memory
3. External Non-volatile Memory – Flash, Read-Only Memory, and other equivalent solutions.

Table 1-1 provides the available solutions for the Virtex-II Pro system wake-up.

Table 1-1:   **Solutions Available for Virtex-II Pro System Wake-Up**

Solution	Density	Mode Support	Configuration Speed	Technology	I/O Voltage Tolerance	Feature Support
System ACE CF CompactFlash	64 Mb up to 8 Gb	JTAG (IEEE 1149.1, IEEE 1532)	Up to 30 Mb/s	CompactFlash-based solution	Down to 2.5V	• MPU Interface (PPC compatible) • Configuration management – multiple Design set support • Network reconfigurable
Platform Flash PROM[2]	Up to 32 Mb (cascadable for higher density)	Master-Serial, Slave-SelectMAP	Up to 264 Mb/s	Flash	1.5V – 3.3V	• Devices are in-system, reprogrammable via IEEE 1149.1 (JTAG)

Notes:

1. The XC2VP2 device which does not contain a CPU can be configured in the same manner as a Virtex-II device with a similar density.
2. See DS123, *Platform Flash PROM Data Sheet*, for the features and specifications of each family member.

System ACE CF (CompactFlash) Solution

The System ACE CF is a pre-engineered interface solution between a CompactFlash or one-inch Microdrive disk drive and a Virtex-II Pro FPGA. It features a built-in microprocessor interface and can support the configuration of multiple devices, boards, and bitstreams (multiple designs), as well as processor core initialization and software storage. This drop-in configuration solution delivers a dense, flexible, pre-engineered solution that reduces design time while supporting the complex needs of a microprocessor-based solution.   For more information, see:

http://www.xilinx.com/products/silicon_solutions/proms/system_ace/

When the System ACE CF solution is used with the Virtex-II Pro Platform FPGA, the wake-up sequence is as follows:

1. Configure the Virtex-II Pro FPGA logic from the CompactFlash (CF) via JTAG.

2. Transfer boot code from CompactFlash to block SelectRAM™ (block RAM) or external RAM (DRAM or SRAM) via JTAG.

3. The PPC boots up from block RAM or RAM.

4. PPC can access the CompactFlash memory via the MPU port of the System ACE CF Controller for code storage and software execution. The PPC can also transfer software from the CompactFlash to block RAM, RAM, or a combination of both for faster execution.
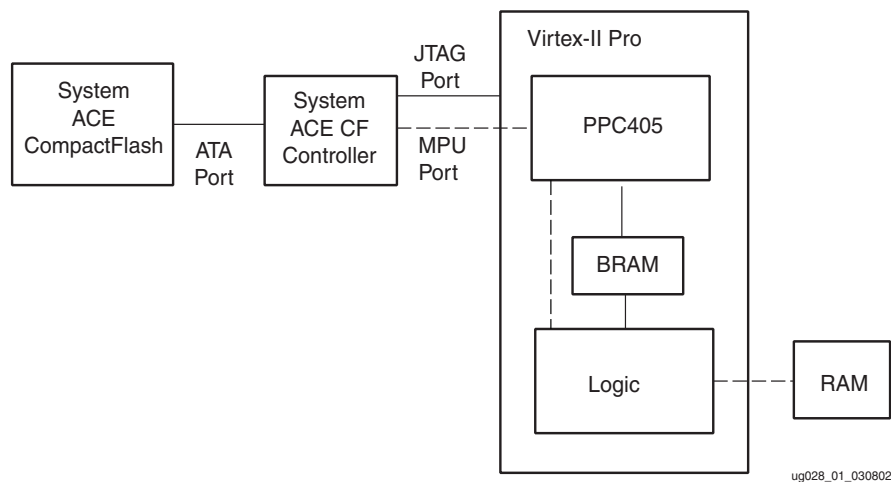

ug028_01_030802

Figure 1-1: **System ACE CF Solution for Virtex-II Pro Wake-Up**

Xilinx PROMs

Xilinx PROMs provide an ease-to-use, high-performance configuration solution. In addition, Xilinx PROMs are in-system programmable through a standard IEEE 1149.1 (JTAG) port.

When a Xilinx Configuration PROM is used with a Virtex-II Pro device (Figure 1-2), the wake-up sequence is as follows:

1. Configure the FPGA portion of Virtex-II Pro device and transfer boot code from PROM via SelectMap or Serial to block RAM.

2. The block RAM boots up the PPC.

3. The PPC can access the external non-volatile memory, like standard Flash, for code storage and software execution. The PPC can transfer software from the external non-volatile memory to block RAM, RAM, or a combination of both for faster execution.
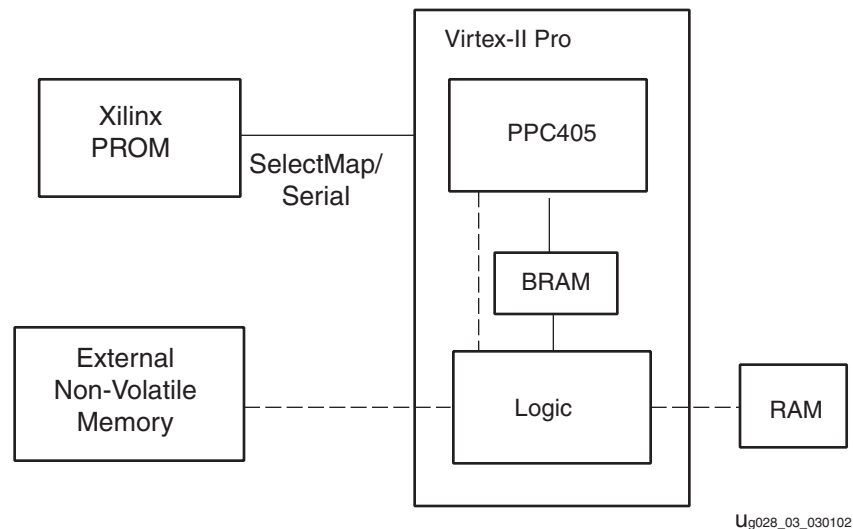

Figure 1-2: **Xilinx PROM Solution for Virtex-II Pro System Wake-Up**
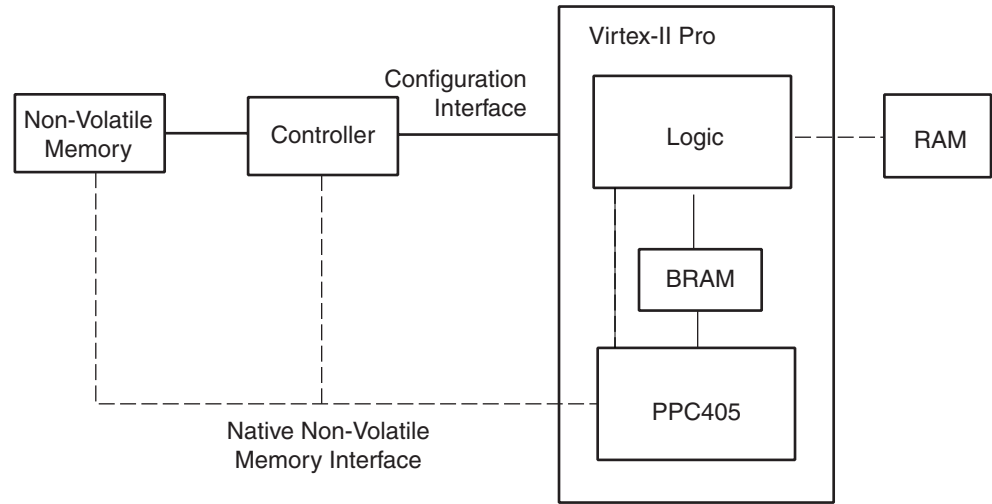
External Non-Volatile Memory

External non-volatile memory can be used to configure a Virtex-II Pro device including Flash-based memory (such as standard Flash, CF cards, SD cards, etc.), as well as and hard disk drives. Using an external non-volatile memory requires a controller to configure the Virtex-II Pro device on wake-up. For the controller design, refer to the following application notes:

- [XAPP058](), *Xilinx In-System Programming Using an Embedded Microcontroller*
- [XAPP079](), *Configuring Xilinx FPGAs Using an XC9500 CPLD and Parallel PROM*
- [XAPP424](), *Embedded JTAG ACE Player*
- [XAPP502](), *Using a Microprocessor to Configure Xilinx FPGAs via Slave Serial or SelectMAP Mode*

The sequence of wake-up for a Virtex-II Pro device using an external non-volatile memory is as follows:

1. Configure the FPGA portion of the Virtex-II Pro device and transfer the boot code from external non-volatile memory to either block RAM or RAM.
2. Either block RAM or RAM boots up the PPC.

The PPC can access external non-volatile memory, via the non-volatile memory interface, for code storage and software execution. The PPC can transfer software from the external non-volatile memory to block RAM, RAM, or a combination of both for faster execution.

Figure 1-3: **Non-Volatile Memory Solution for Virtex-II Pro Wake-Up**

Basic and Advanced Virtex-II Pro Wake-Up Solutions

Basic Virtex-II Pro Wake-Up Solution

Overview

The first method described in this section is a well-known setup used with other Xilinx FPGA families. The setup requires very little external hardware and minimal connections. The only external device required for this setup is a PROM storage device. The PROM is used to store the traditional configuration data for the Virtex™-II Pro device.

The new level of complexity that must be addressed is the storage location of the Virtex-II Pro PowerPC™ boot-code. In this basic setup, the Virtex-II Pro FPGA block SelectRAM (also referred to as block RAM) is utilized for the PPC boot code storage. The block RAM contents are loaded in the configuration data that is stored in the PROM, thereby eliminating the need for an additional external RAM device to store the PPC contents.

Basic System Architecture

Figure 2-1 shows the connections to the Xilinx PROM required for the Master Serial configuration mode. Figure 2-1 also shows a simplified block diagram of the Virtex-II Pro device before configuration. In this setup, both the FPGA logic and block RAM load on configuration from the PROM and the PPC boot-ups from the block RAM.

ug028_05_022502
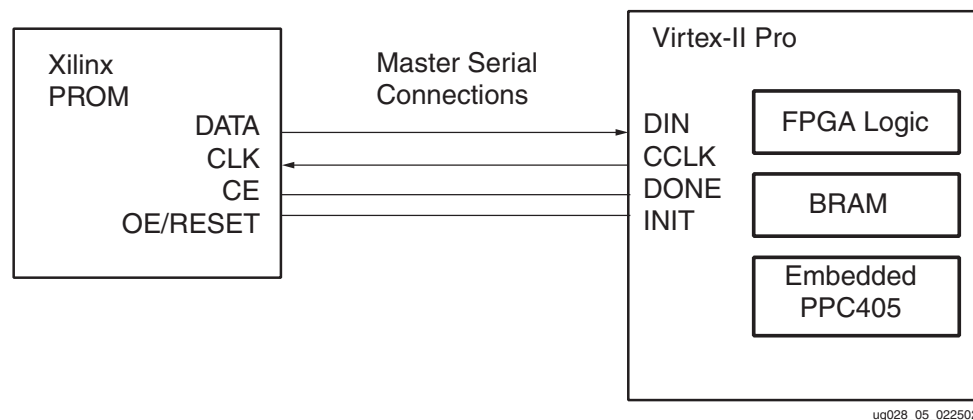
Notes:

1. Other modes can also be used in a similar manner with additional hardware connections. Refer to the appropriate Xilinx PROM data sheet for guidelines and the required VCC/GND pull-up/pull-down and optional connections

Figure 2-1: **Virtex-II Pro Basic System Wake-Up Solution**

This basic setup requires less hardware and external connections, but the maximum storage space available for the boot-code and software is limited by the block RAM. Each available block RAM contains 18 Kbits. Table 2-1 displays the total amount of block RAM memory available for each member of the Virtex-II Pro family.

Table 2-1: **Sample Virtex-II Pro Platform FPGA Block RAM and Microprocessor Cores**

Device	Columns	No. of PPC Cores	Total SelectRAM[1,2] Memory		
			Blocks	**In Kbits**	**In Bits**
XC2VP4	4	1	28	504	516,096
XC2VP7	6	1	44	792	811,008
XC2VP20	8	2	88	1,584	1,622,016
XC2VP50	12	4	216	3,888	3,981,312

Notes:

1. The XC2VP2 does not contain PPC cores, so standard Virtex-II configuration solutions can be utilized.
2. 16 Kbits of the block RAM are available for direct interfacing with the Virtex-II Pro PPC.

Basic Solution Wake-Up Flow

Step 1. Configuration	Step 2. Startup	Step 3. PPC Boot

Power-on Reset (POR)   MPU Wake-Up
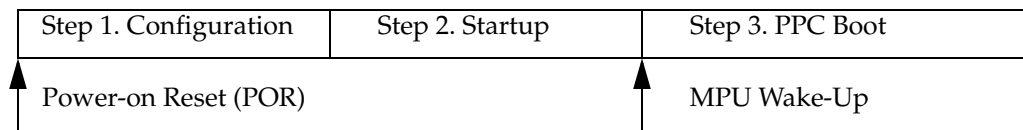
Step 1

The Configuration stage is similar to other Virtex FPGA families. During this stage, the Virtex-II Pro device is powered on and completes three major phases:

- Clearing configuration memory
- Initialization
- Loading the bitstream

Phase 1 and 2 simply prepare the device to receive the configuration data from the PROM. The FPGA's memory is cleared and the mode pin settings are sampled to determine the correct download mode. The appropriate pins become active to signify the loading of the configuration data. The loading of the bitstream defines the functional operation of the internal blocks (for example function look-up tables (LUTs), flip-flops (FFs), multiplexers, buffers), I/O (defining pull-up/pull-down, slew rate, etc.), and their interconnections.

Step 2

Device startup is a transition stage from the configuration mode to the normal programmed device operation. The startup sequence performs the following tasks:

1. Release the DONE pin
2. Negate Global Tristate Signal (GTS) which activates all I/Os
3. Assert Global Write Enable (GWE) which allows all RAMS and flip-flops to change state
4. Assert End-Of-Startup (EOS) internal flag

Step 3

At this stage, the FPGA has completed configuration and the PPC starts to execute code from the reset vector.

Basic Software Flow

This section describes the software flows necessary to generate a valid programming bitstream with the Virtex-II Pro PPC boot code. It also explains how to convert this bitstream into a PROM file required for this solution.

The flow through the Xilinx software tools to create a valid bitstream for the Virtex-II Pro device in general is the same as it is for any other Xilinx FPGA. The standard flow takes a design originating from HDL or schematic source code through the design front-end tools to create a standard netlist. This netlist is taken through the Xilinx implementation tool suite to physically map, then place and route the design into the desired Virtex-II Pro architecture. For further details on these software tools, refer to the on-line ISE™ Software Manuals.

The deviation from the most standard flow for Xilinx FPGA families to a flow targeting the Virtex-II Pro device is in the setup required to allow the PPC to execute code from the block RAM. Two additional files are utilized to enable this functionality, the block SelectRAM Memory Map file (.bmm) and the software code file (.elf). For additional information on the creation and usage of these two files, refer to the Embedded Development Kit (EDK) User Guides.
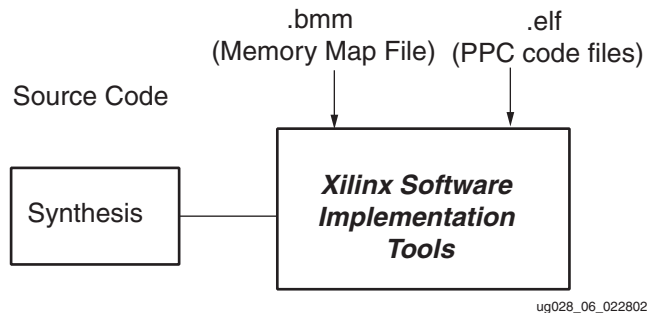

ug028_06_022802

Figure 2-2: **Xilinx Tool Software Flow for Bitstream Generation with Block RAM Initialization**

The last step in this software flow is the creation of the PROM file for programming the PROM. The PROM File is created in iMPACT software. The merged bitstream created from the Data2BRAM flow is used as the input file for iMPACT. The user targets the appropriate PROM density and .mcs format to create a formatted PROM file for the target solution. For a detailed flow, refer to the on-line iMPACT Software Manual.

Advanced Virtex-II Pro Wake-Up Solution

Overview

The second method discussed is for advanced users who want to take advantage of the configuration management and auxiliary storage capabilities of the System ACE CompactFlash (CF) solution.

The System ACE CF can manage up to eight separate designs for the Virtex-II Pro device. By default, one design is delivered at power-up to configure the Virtex-II Pro device and boot the PPC. On command, the System ACE CF can reconfigure the Virtex-II Pro FPGA with any of the eight possible designs stored in the CF.

The System ACE CF controller has a microprocessor (MPU) port that provides access to the CompactFlash. The Virtex-II Pro PPC can be connected to the System ACE CF MPU port to access data in the standard file structure within the CF.

The System ACE CF provides a centralized solution for system wake-up and the storage of auxiliary application code and data.

Advanced System Architecture

The advanced system contains the CompactFlash storage device, the System ACE CompactFlash controller, the Virtex-II Pro device, and external RAM. The System ACE CF controller acts as the master controller for the CompactFlash device. At power-up or on command, the System ACE CF controller extracts a selected configuration file from the CompactFlash and delivers the configuration data to the Virtex-II Pro device via the JTAG Test Access Port (TAP). The System ACE CF configuration process includes the delivery of both the FPGA fabric configuration bits and the initialization of the external RAM with the code and data. See below for details on the configuration delivery scheme.

The optional MPU port connection between the Virtex-II Pro device and the System ACE CF controller gives the PPC access to the CompactFlash file structure for reading and writing additional application code and data.
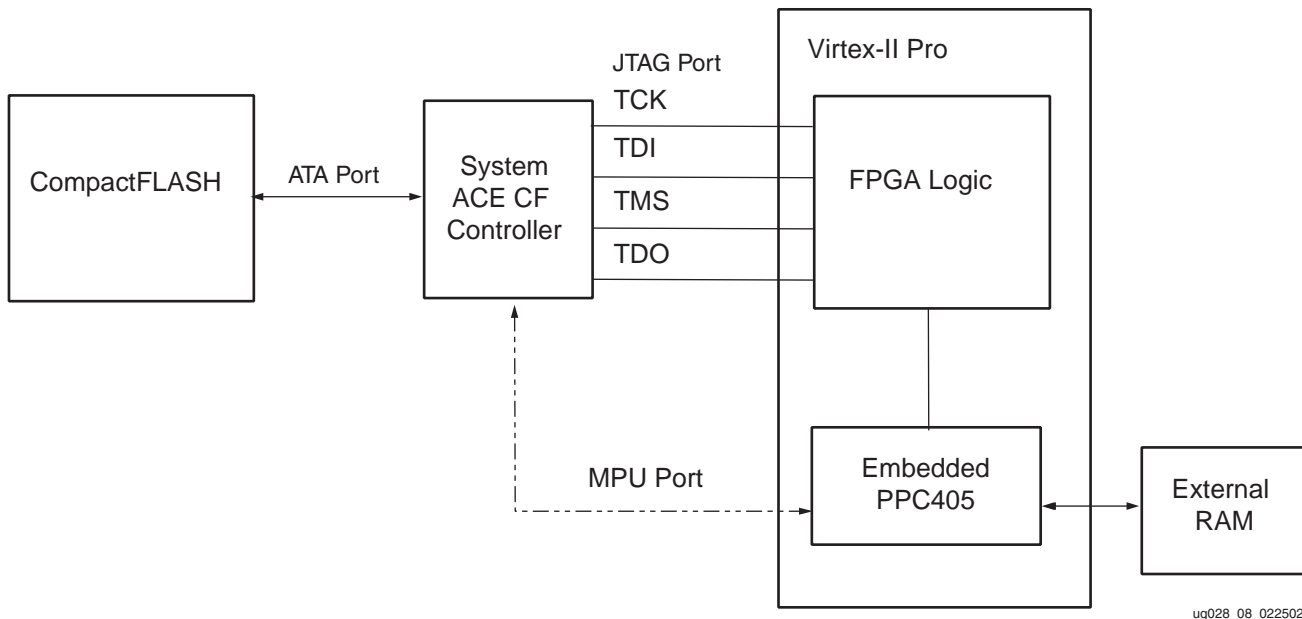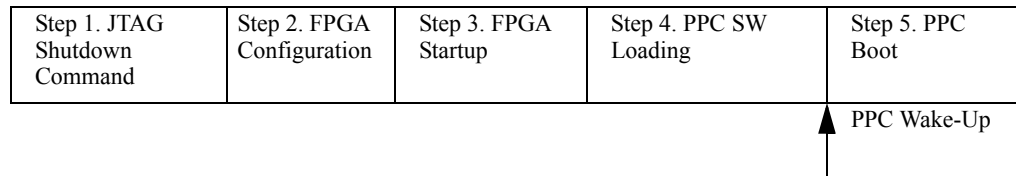

ug028_08_022502

Figure 2-3: **Advanced System Wake-Up Solution**

Advanced Solution Wake-Up Flow

The Virtex-II Pro JTAG configuration flow is designed to support reconfiguration as well as initial configuration. Thus, the Virtex-II Pro wake-up flow begins with a JTAG shutdown command sequence that disables the active logic in the FPGA. The FPGA configuration and startup stages are the same as in the Basic setup. However, two extra stages are required to load the PPC code and wake up the PPC.

Step 1. JTAG Shutdown Command	Step 2. FPGA Configuration	Step 3. FPGA Startup	Step 4. PPC SW Loading	Step 5. PPC Boot

PPC Wake-Up

The five stages of the wake-up flow are accomplished through JTAG command sequences from the System ACE CF.

Step 1

The shutdown stage halts the PPC and disables the active logic in the FPGA. This stage prepares the FPGA to receive new configuration data. In the case of a reconfiguration, this stage eliminates the danger of contention between the new configuration and the previously loaded configuration.

Step 2

The configuration stage is similar to the other Virtex FPGA families' configuration stages. During this stage, the new bitstream is downloaded to the Virtex-II Pro device. The bitstream defines the functional operation of the internal logic blocks, the I/O pins, and the PPC interconnects to internal and peripheral devices, such as the external RAM.

Step 3

Startup is a transition stage from the configuration mode to the normal programmed device operation. The startup sequence performs the following tasks:

1. Releases the DONE pin
2. Negates Global Tristate Signal (GTS) which activates all I/Os
3. Asserts Global Write Enable (GWE) which allows all RAMS and flip-flops to change state
4. Assert End-Of-Startup (EOS) internal flag

At the completion of this stage, the FPGA fabric is active including the PPC interconnects.

Step 4

The PPC software loading stage fills the external RAM with the PPC boot code and data through the PPC control logic.

Step 5

The PPC boot stage consists of the setting of the PPC program counter to the starting location of the boot code and releasing the PPC into its normal run mode.

Virtex-II Pro JTAG Test Access Port (TAP)

The JTAG TAP in the Virtex-II Pro device is tightly integrated with the FPGA configuration logic as well as the PPC control logic. The JTAG TAP and FPGA configuration logic integration allows the FPGA to be configured through the JTAG TAP. The JTAG TAP and PPC control logic integration allows the PPC to be controlled through the JTAG TAP.

JTAG control of the PPC is the key to the memory initialization and PPC wake-up scheme. The System ACE CF can instruct the PPC through the JTAG TAP to write the initial memory image. The System ACE CF can also instruct the PPC to start from a specific code location through the JTAG TAP.

Advanced Software Flow

A single System ACE CF configuration (`.ace`) file contains all the JTAG commands that are necessary to configure the Virtex-II Pro device and wake the PPC in the advanced system. The System ACE CF configuration file is created from two sources: the FPGA configuration bitstream (`.bit`) file and the PPC code and data (`.elf`) file. See the *Embedded Development Kit (EDK) User Guides* for descriptions of the flows that create the `.bit` and `.elf` files.

Because the target Virtex-II Pro device can be in a JTAG scan chain with an arbitrary number of JTAG devices, the System ACE CF configuration file preparation process must understand the JTAG scan chain layout. The System ACE CF solution delivers the configuration and code through serial JTAG vectors. These serial JTAG vectors are saved in an intermediate *de facto* standard Serial Vector Format (SVF) file.
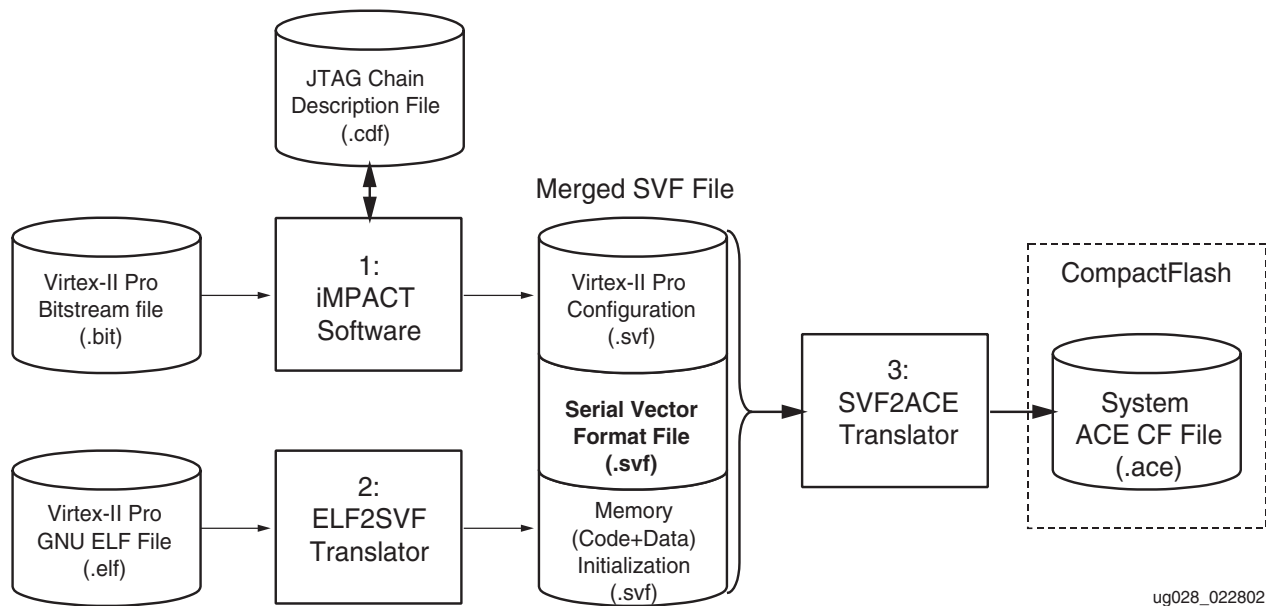
Figure 2-4: **Advanced Virtex-II Pro Software Flow**

Transformation of the `.bit` and `.elf` files into the unified `.ace` file requires three software steps:

Step 1

The iMPACT software generates the JTAG vectors for delivering the bitstream to the target Virtex-II Pro within the JTAG scan chain. The result is an SVF file with vectors to shut down, configure, and start up the FPGA logic. (See the *iMPACT Software Manual* for detailed instructions on defining the JTAG scan chain and creating an SVF file to program the Virtex-II Pro device.)

Step 2

The ELF2SVF translates the PPC code and data (`.elf`) file into JTAG instructions for the PPC to fill the appropriate memory locations. In addition, the ELF2SVF translator appends JTAG instructions to preset the PPC program counter to the starting location of the boot code and to put the PPC into its run state with the preset program counter value. The resulting SVF is appended to the bitstream configuration SVF file.

Step 3

Convert the combined SVF file into the corresponding System ACE CompactFlash (ACE) file.

The ACE file is copied to the CompactFlash unit for system deployment (see DS080, *System ACE CF Data Sheet* for detailed instructions on the System ACE CompactFlash operation and file structure).

In the target system, the System ACE CF controller executes the ACE file from the CompactFlash to configure and wake up the PPC.