Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications,

Details	
Product Status	Obsolete
Number of LABs/CLBs	-
Number of Logic Elements/Cells	-
Total RAM Bits	110592
Number of I/O	154
Number of Gates	600000
Voltage - Supply	1.14V ~ 1.575V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	208-BFQFP
Supplier Device Package	208-PQFP (28x28)
Purchase URL	https://www.e-xfl.com/product-detail/microsemi/a3p600l-1pq208

Table of Contents

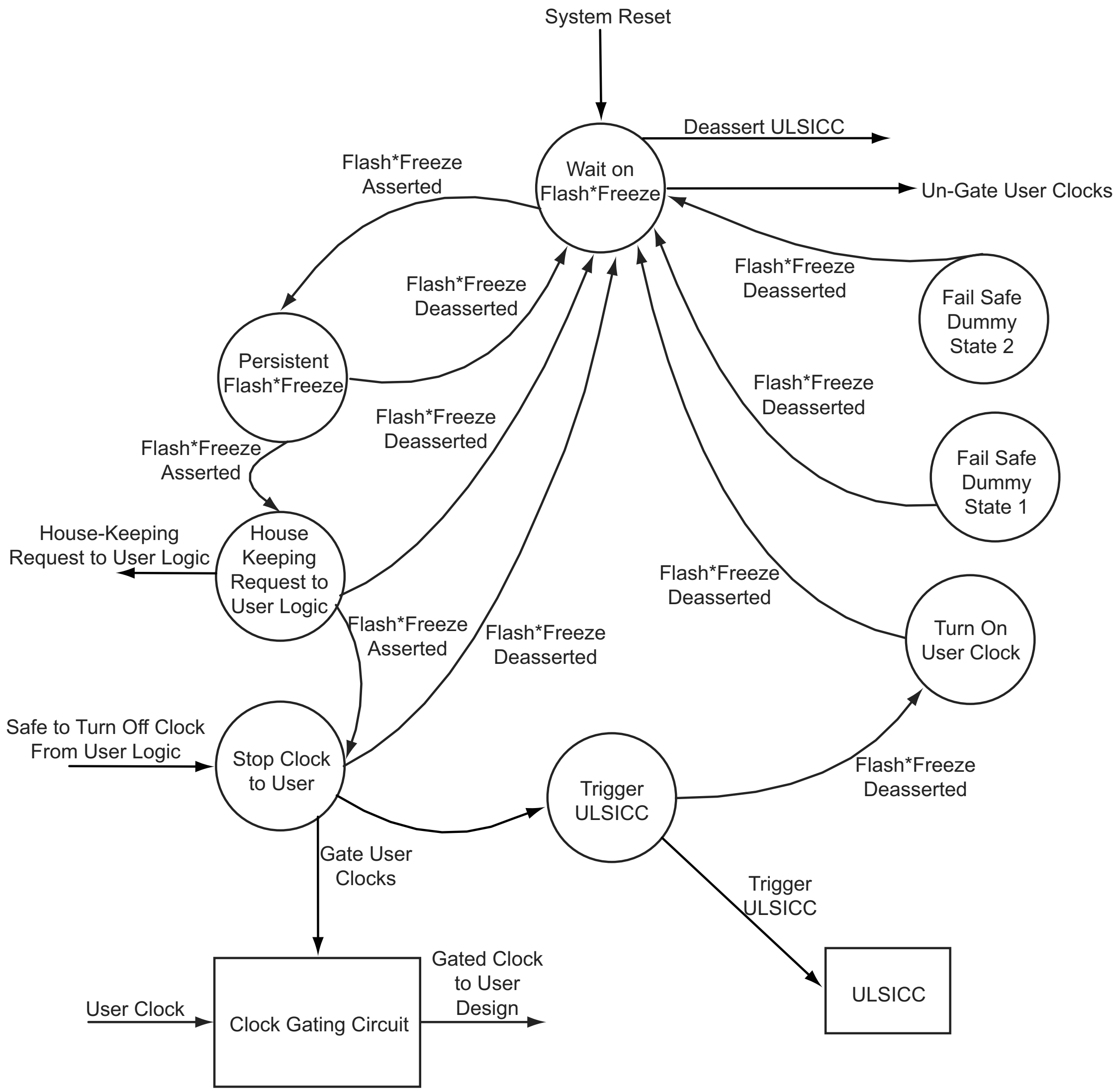

***Figure 2-11* • FSM State Diagram**

- Avoid using pull-ups and pull-downs on I/Os because these resistors draw some current. Avoid driving resistive loads or bipolar transistors, since these draw a continuous current, thereby adding to the static current.
- When partitioning the design across multiple devices, minimize I/O usage among the devices.

Conclusion

Microsemi IGLOO, IGLOO nano, IGLOO PLUS, ProASIC3L, and RT ProASIC3 family architectures are designed to achieve ultra-low power consumption based on enhanced nonvolatile and live-at-power-up flash-based technology. Power consumption can be reduced further by using Flash*Freeze, Static (Idle), Sleep, and Shutdown power modes. All these features result in a low power, cost-effective, single-chip solution designed specifically for power-sensitive and battery-operated electronics applications.

Related Documents

Application Notes

Embedded SRAM Initialization Using External Serial EEPROM

http://www.microsemi.com/soc/documents/EmbeddedSRAMInit_AN.pdf

List of Changes

The following table lists critical changes that were made in each version of the chapter.

Date	Changes	Page
July 2010	This chapter is no longer published separately with its own part number and version but is now part of several FPGA fabric user's guides.	N/A
v2.3 (November 2009)	The "Sleep Mode" section was revised to state the VJTAG and VPUMP, as well as VCC, are grounded during Sleep mode (SAR 22517).	32
	Figure 2-6 • Controlling Power-On/-Off State Using Microprocessor and Power FET and Figure 2-7 • Controlling Power-On/-Off State Using Microprocessor and Voltage Regulator were revised to show that VJTAG and VPUMP are powered off during Sleep mode.	33
v2.2 (December 2008)	IGLOO nano devices were added as a supported family.	N/A
	The "Prototyping for IGLOO and ProASIC3L Devices Using ProASIC3" section was removed, as these devices are now in production.	N/A
	The "Additional Power Conservation Techniques" section was revised to add RT ProASIC3 devices.	41
v2.0 (October 2008)	The "Flash*Freeze Management FSM" section was updated with the following information: The FSM also asserts Flash_Freeze_Enabled whenever the device enters Flash*Freeze mode. This occurs after all housekeeping and clock gating functions have completed.	37

Software Configuration

SmartGen automatically generates the desired CCC functional block by configuring the control bits, and allows the user to select two CCC modes: Static PLL and Delayed Clock (CLKDLY).

Static PLL Configuration

The newly implemented Visual PLL Configuration Wizard feature provides the user a quick and easy way to configure the PLL with the desired settings (Figure 4-23). The user can invoke SmartGen to set the parameters and generate the netlist file with the appropriate flash configuration bits set for the CCCs. As mentioned in "PLL Macro Block Diagram" on page 85, the input reference clock CLKA can be configured to be driven by Hardwired I/O, External I/O, or Core Logic. The user enters the desired settings for all the parameters (output frequency, output selection, output phase adjustment, clock delay, feedback delay, and system delay). Notice that the actual values (divider values, output frequency, delay values, and phase) are shown to aid the user in reaching the desired design frequency in real time. These values are typical-case data. Best- and worst-case data can be observed through static timing analysis in SmartTime within Designer.

For dynamic configuration, the CCC parameters are defined using either the external JTAG port or an internally defined serial interface via the built-in dynamic shift register. This feature provides the ability to compensate for changes in the external environment.

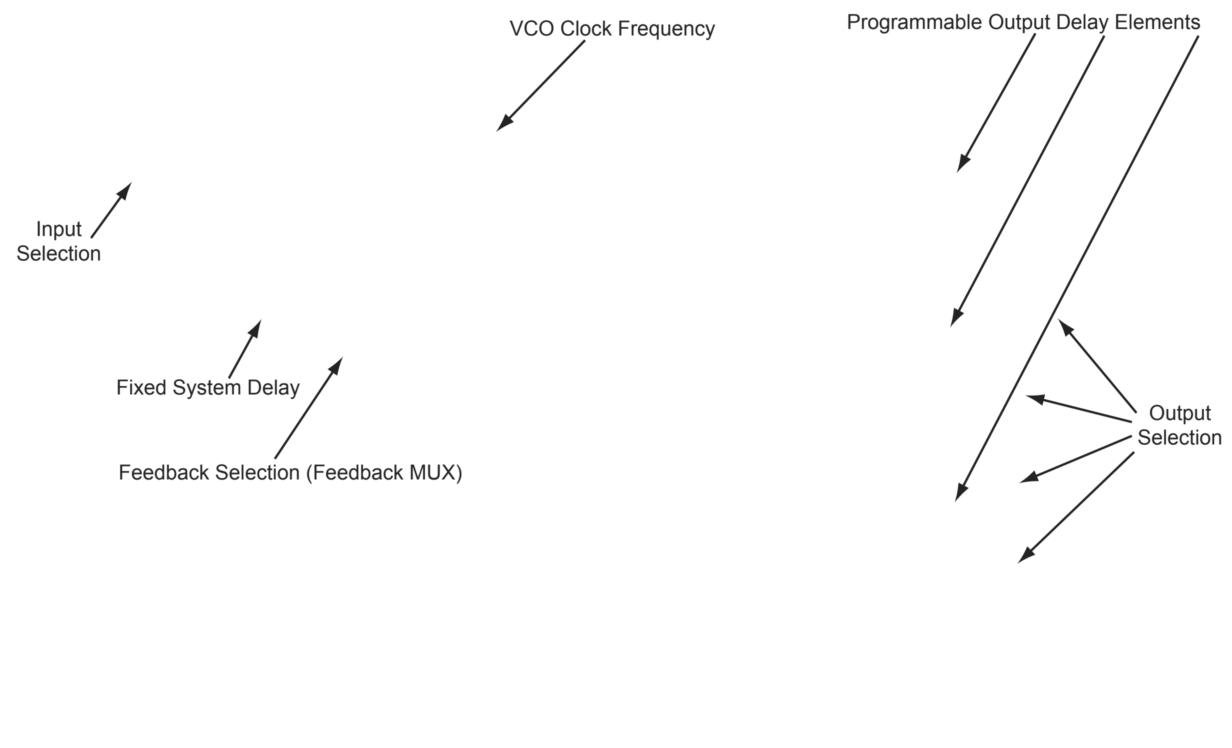

Figure 4-23 **• Visual PLL Configuration Wizard**

Recommended Board-Level Considerations

The power to the PLL core is supplied by VCCPLA/B/C/D/E/F (VCCPLx), and the associated ground connections are supplied by VCOMPLA/B/C/D/E/F (VCOMPLx). When the PLLs are not used, the Designer place-and-route tool automatically disables the unused PLLs to lower power consumption. The user should tie unused VCCPLx and VCOMPLx pins to ground. Optionally, the PLL can be turned on/off during normal device operation via the POWERDOWN port (see Table 4-3 on page 84).

PLL Power Supply Decoupling Scheme

The PLL core is designed to tolerate noise levels on the PLL power supply as specified in the datasheets. When operated within the noise limits, the PLL will meet the output peak-to-peak jitter specifications specified in the datasheets. User applications should always ensure the PLL power supply is powered from a noise-free or low-noise power source.

However, in situations where the PLL power supply noise level is higher than the tolerable limits, various decoupling schemes can be designed to suppress noise to the PLL power supply. An example is provided in Figure 4-38. The VCCPLx and VCOMPLx pins correspond to the PLL analog power supply and ground.

Microsemi strongly recommends that two ceramic capacitors (10 nF in parallel with 100 nF) be placed close to the power pins (less than 1 inch away). A third generic 10 µF electrolytic capacitor is recommended for low-frequency noise and should be placed farther away due to its large physical size. Microsemi recommends that a 6.8 µH inductor be placed between the supply source and the capacitors to filter out any low-/medium- and high-frequency noise. In addition, the PCB layers should be controlled so the VCCPLx and VCOMPLx planes have the minimum separation possible, thus generating a good-quality RF capacitor.

For more recommendations, refer to the *Board-Level Considerations* application note.

Recommended 100 nF capacitor:

- Producer BC Components, type X7R, 100 nF, 16 V
- BC Components part number: 0603B104K160BT
- Digi-Key part number: BC1254CT-ND
- Digi-Key part number: BC1254TR-ND

Recommended 10 nF capacitor:

- Surface-mount ceramic capacitor
- Producer BC Components, type X7R, 10 nF, 50 V
- BC Components part number: 0603B103K500BT
- Digi-Key part number: BC1252CT-ND
- Digi-Key part number: BC1252TR-ND

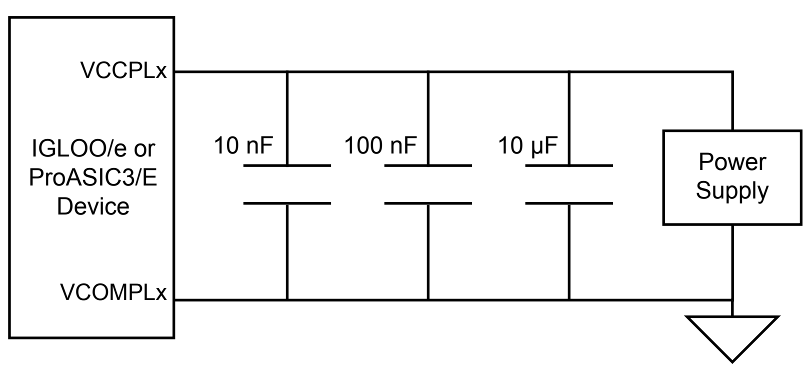

Figure 4-38 **• Decoupling Scheme for One PLL (should be replicated for each PLL used)**

Temporary overshoots are allowed according to the overshoot and undershoot table in the datasheet.

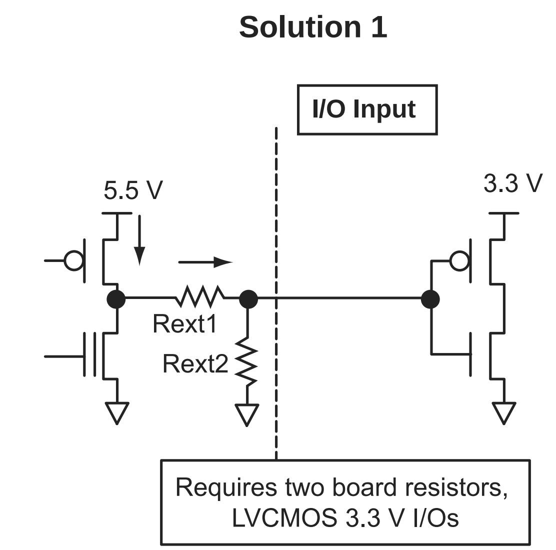

Figure 7-9 • **Solution 1**

Solution 2

The board-level design must ensure that the reflected waveform at the pad does not exceed the voltage overshoot/undershoot limits provided in the datasheet. This is a requirement to ensure long-term reliability.

This scheme will also work for a 3.3 V PCI/PCI-X configuration, but the internal diode should not be used for clamping, and the voltage must be limited by the external resistors and Zener, as shown in Figure 7-10. Relying on the diode clamping would create an excessive pad DC voltage of 3.3 V + 0.7 V = 4 V.

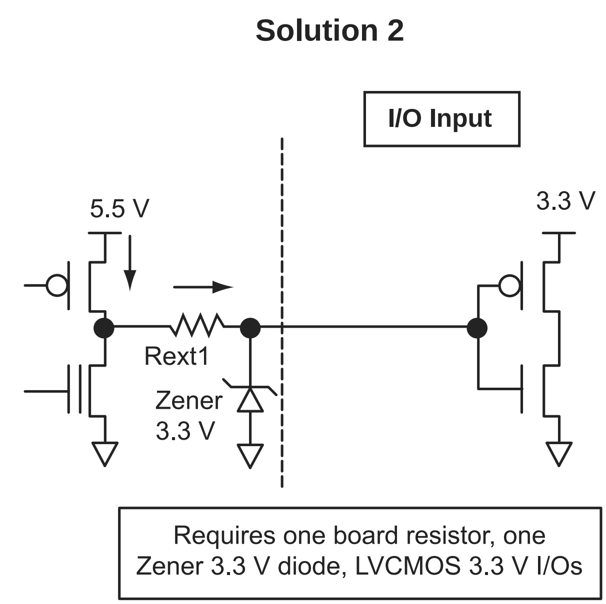

Figure 7-10 • **Solution 2**

Table 7-19 shows some high-level interfacing examples using low power flash devices.

***Table 7-19* • High-Level Interface Examples**

Interface	Clock		I/O			
	Type	**Frequency**	**Type**	**Signals In**	**Signals Out**	**Data I/O**
GM	Src Sync	125 MHz	LVTTL	8	8	125 Mbps
TBI	Src Sync	125 MHz	LVTTL	10	10	125 Mbps
XSBI	Src Sync	644 MHz	LVDS	16	16	644 Mbps
XGMI	Src Sync DDR	156 MHz	HSTL1	32	32	312 Mbps
FlexBus 3	Sys Sync	104 MHz	LVTTL	≤ 32	≤ 32	≤ 104
Pos-PHY3/SPI-3	Sys Sync	104	LVTTL	8, 16, 32	8, 16, 32	≤ 104 Mbps
FlexBus 4/SPI-4.1	Src Sync	200 MHz	HSTL1	16,64	16,64	200 Mbps
Pos-PHY4/SPI-4.2	Src Sync DDR	≥ 311 MHz	LVDS	16	16	≥ 622 Mbps
SFI-4.1	Src Sync	622 MHz	LVDS	16	16	622 Mbps
CSIX L1	Sys Sync	≤ 250 MHz	HSTL1	32,64,96,128	32,64,96,128	≤ 250 Mbps
Hyper Transport	Sys Sync DDR	≤ 800 MHz	LVDS	2,4,8,16	2,4,8,16	≤ 1.6 Gbps
Rapid I/O Parallel	Sys Sync DDR	250 MHz – 1 GHz	LVDS	8,16	8,16	≤ 2 Gbps
Star Fabric	CDR		LVDS	4	4	622 Mbps

Note: Sys Sync = System Synchronous Clocking, Src Sync = Source Synchronous Clocking, and CDR = Clock and Data Recovery.

Conclusion

IGLOO and ProASIC3 support for multiple I/O standards minimizes board-level components and makes possible a wide variety of applications. The Microsemi Designer software, integrated with Libero SoC, presents a clear visual display of I/O assignments, allowing users to verify I/O and board-level design requirements before programming the device. The IGLOO and ProASIC3 device I/O features and functionalities ensure board designers can produce low-cost and low power FPGA applications fulfilling the complexities of contemporary design needs.

Date	Change	Page
June 2011 (continued)	The following sentence was removed from the "LVCMOS (Low-Voltage CMOS)" section (SAR 22634): "All these versions use a 3.3 V–tolerant CMOS input buffer and a push-pull output buffer."	184
	Hot-insertion was changed to "No" for other IGLOO and all ProASIC3 devices in Table 7-12 • I/O Hot-Swap and 5 V Input Tolerance Capabilities in IGLOO and ProASIC3 Devices (SAR 24526).	193
	The "Electrostatic Discharge Protection" section was revised to remove references to tolerances (refer to the *Reliability Report* for tolerances). The Machine Model (MM) is not supported and was deleted from this section (SAR 24385).	192
	The "I/O Interfacing" section was revised to state that low power flash devices are 5 V–input– and 5 V–output–tolerant if certain I/O standards are selected, removing "without adding any extra circuitry," which was incorrect (SAR 21404).	208
July 2010	This chapter is no longer published separately with its own part number and version but is now part of several FPGA fabric user's guides.	N/A
v1.4 (December 2008)	The terminology in the "Low Power Flash Device I/O Support" section was revised.	176
v1.3 (October 2008)	The "Low Power Flash Device I/O Support" section was revised to include new families and make the information more concise.	176
v1.2 (June 2008)	The following changes were made to the family descriptions in Table 7-1 • Flash-Based FPGAs: • ProASIC3L was updated to include 1.5 V. • The number of PLLs for ProASIC3E was changed from five to six.	176
v1.1 (March 2008)	Originally, this document contained information on all IGLOO and ProASIC3 families. With the addition of new families and to highlight the differences between the features, the document has been separated into 3 documents: This document contains information specific to IGLOO, ProASIC3, and ProASIC3L. "I/O Structures in IGLOOe and ProASIC3E Devices" in the *ProASIC3E FPGA Fabric User's Guide* contains information specific to IGLOOe, ProASIC3E, and ProASIC3EL I/O features. "I/O Structures in IGLOO PLUS Devices" in the *IGLOO PLUS FPGA Fabric User's Guide* contains information specific to IGLOO PLUS I/O features.	N/A

Table 9-3 **• PDC I/O Constraints (continued)**

Command	Action	Example	Comment
I/O Attribute Constraint			

set_io	Sets the attributes of an I/O	`set_io portname`
		`[-pinname value]`
		`[-fixed value]`
		`[-iostd value]`
		`[-out_drive value]`
		`[-slew value]`
		`[-res_pull value]`
		`[-schmitt_trigger value]`
		`[-in_delay value]`
		`[-skew value]`
		`[-out_load value]`
		`[-register value]`
		
		`set_io IN2 -pinname 28`
		`-fixed yes -iostd LVCMOS15`
		`-out_drive 12 -slew high`
		`-RES_PULL None`
		`-SCHMITT_TRIGGER Off`
		`-IN_DELAY Off –skew off`
		`-REGISTER No`
			If the I/O macro is generic (e.g., INBUF) or technology-specific (INBUF_LVCMOS25), then all I/O attributes can be assigned using this constraint.
			If the netlist has an I/O macro that specifies one of its attributes, that attribute cannot be changed using this constraint, though other attributes can be changed.
			Example: OUTBUF_S_24 (low slew, output drive 24 mA)
			Slew and output drive cannot be changed.

I/O Region Placement Constraints			

define_region	Defines either a rectangular region or a rectilinear region	`define_region`
		`-name [region_name]`
		`-type [region_type] x1 y1 x2 y2`
		
		`define_region -name test`
		`-type inclusive 0 15 2 29`
			If any number of I/Os must be assigned to a particular I/O region, such a region can be created with this constraint.

assign_region	Assigns a set of macros to a specified region	`assign_region [region name]`
		`[macro_name...]`
		
		`assign_region test U12`
			This constraint assigns I/O macros to the I/O regions. When assigning an I/O macro, PDC naming conventions must be followed if the macro name contains special characters; e.g., if the macro name is \\$1I19\\, the correct use of escape characters is \\\\$1I19\\\\.

Note: Refer to the Libero SoC User's Guide *for detailed rules on PDC naming and syntax conventions.*

List of Changes

The following table lists critical changes that were made in each revision of the chapter.

Date	Changes	Page
July 2010	This chapter is no longer published separately with its own part number and version but is now part of several FPGA fabric user's guides.	N/A
	Notes were added where appropriate to point out that IGLOO nano and ProASIC3 nano devices do not support differential inputs (SAR 21449).	N/A
v1.4 (December 2008)	IGLOO nano and ProASIC3 nano devices were added to Table 10-1 • Flash-Based FPGAs.	272
	The "I/O Cell Architecture" section was updated with information applicable to nano devices.	273
	The output buffer (OUTBUF_SSTL3_I) input was changed to D, instead of Q, in Figure 10-1 • DDR Support in Low Power Flash Devices, Figure 10-3 • DDR Output Register (SSTL3 Class I), Figure 10-6 • DDR Output Register (SSTL3 Class I), Figure 10-7 • DDR Tristate Output Register, LOW Enable, 8 mA, Pull-Up (LVTTL), and the output from the DDR_OUT macro was connected to the input of the TRIBUFF macro in Figure 10-7 • DDR Tristate Output Register, LOW Enable, 8 mA, Pull-Up (LVTTL).	271, 275, 278, 279
v1.3 (October 2008)	The "Double Data Rate (DDR) Architecture" section was updated to include mention of the AFS600 and AFS1500 devices.	271
	The "DDR Support in Flash-Based Devices" section was revised to include new families and make the information more concise.	272
v1.2 (June 2008)	The following changes were made to the family descriptions in Table 10-1 • Flash-Based FPGAs: • ProASIC3L was updated to include 1.5 V. • The number of PLLs for ProASIC3E was changed from five to six.	272
v1.1 (March 2008)	The "IGLOO Terminology" section and "ProASIC3 Terminology" section are new.	272

***Table 13-4* • Programming Header Pin Numbers and Description**

Pin	Signal	Source	Description
1	TCK	Programmer	JTAG Clock
2	GND[1]	–	Signal Reference
3	TDO	Target Board	Test Data Output
4	NC	–	No Connect (FlashPro3/3X); Prog_Mode (FlashPro4). See note associated with Figure 13-5 on page 335 regarding Prog_Mode on FlashPro4.
5	TMS	Programmer	Test Mode Select
6	VJTAG	Target Board	JTAG Supply Voltage
7	VPUMP[2]	Programmer/Target Board	Programming Supply Voltage
8	nTRST	Programmer	JTAG Test Reset (Hi-Z with 10 kΩ pull-down, HIGH, LOW, or toggling)
9	TDI	Programmer	Test Data Input
10	GND[1]	–	Signal Reference

Notes:

1. *Both GND pins must be connected.*
2. *FlashPro4/3/3X can provide VPUMP if there is only one device on the target board.*

2. VCC rises to 1.5 V before programming begins.

Figure 14-3 • **Programming Algorithm**

The oscilloscope plot in Figure 14-3 shows a wider time interval for the programming algorithm and includes the TDI and TMS signals from the FlashPro3. These signals carry the programming information that is programmed into the device and should only start toggling after the V_{CC} core voltage reaches 1.5 V. Again, TRST from FlashPro3 and the V_{CC} core voltage of the IGLOO device are labeled. As shown in Figure 14-3, TDI and TMS are floating initially, and the core voltage is 1.2 V. When a programming command on the FlashPro3 is executed, TRST is driven HIGH and TDI is momentarily driven to ground. In response to the HIGH TRST signal, the circuit responds and pulls the core voltage to 1.5 V. After 100 ms, TRST is briefly driven LOW by the FlashPro software. This is expected behavior that ensures the device JTAG state machine is in Reset prior to programming. TRST remains HIGH for the duration of the programming. It can be seen in Figure 14-3 that the VCC core voltage signal remains at 1.5 V for approximately 50 ms before information starts passing through on TDI and TMS. This confirms that the voltage switching circuit drives the VCC core supply voltage to 1.5 V prior to programming.

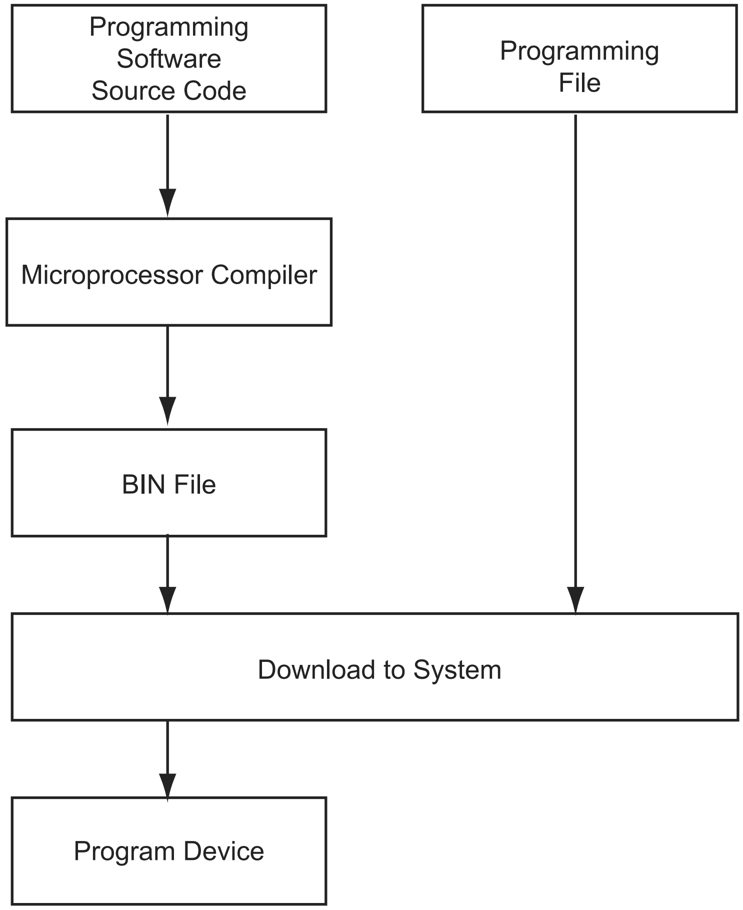

Figure 15-3 • **MCU FPGA Programming Model**

FlashROM

Microsemi low power flash devices have 1 kbit of user-accessible, nonvolatile, FlashROM on-chip. This nonvolatile FlashROM can be programmed along with the core or on its own using the standard IEEE 1532 JTAG programming interface.

The FlashROM is architected as eight pages of 128 bits. Each page can be individually programmed (erased and written). Additionally, on-chip AES security decryption can be used selectively to load data securely into the FlashROM (e.g., over public or private networks, such as the Internet). Refer to the "FlashROM in Microsemi's Low Power Flash Devices" section on page 133.

Microsemi's Flash Devices Support the JTAG Feature

The flash-based FPGAs listed in Table 16-1 support the JTAG feature and the functions described in this document.

***Table 16-1* • Flash-Based FPGAs**

Series	Family*	Description
IGLOO	IGLOO	Ultra-low power 1.2 V to 1.5 V FPGAs with Flash*Freeze technology
	IGLOOe	Higher density IGLOO FPGAs with six PLLs and additional I/O standards
	IGLOO nano	The industry's lowest-power, smallest-size solution
	IGLOO PLUS	IGLOO FPGAs with enhanced I/O capabilities
ProASIC3	ProASIC3	Low power, high-performance 1.5 V FPGAs
	ProASIC3E	Higher density ProASIC3 FPGAs with six PLLs and additional I/O standards
	ProASIC3 nano	Lowest-cost solution with enhanced I/O capabilities
	ProASIC3L	ProASIC3 FPGAs supporting 1.2 V to 1.5 V with Flash*Freeze technology
	RT ProASIC3	Radiation-tolerant RT3PE600L and RT3PE3000L
	Military ProASIC3/EL	Military temperature A3PE600L, A3P1000, and A3PE3000L
	Automotive ProASIC3	ProASIC3 FPGAs qualified for automotive applications
Fusion	Fusion	Mixed signal FPGA integrating ProASIC®3 FPGA fabric, programmable analog block, support for ARM® Cortex™-M1 soft processors, and flash memory into a monolithic device

*Note: *The device names link to the appropriate datasheet, including product brief, DC and switching characteristics, and packaging information.*

IGLOO Terminology

In documentation, the terms IGLOO series and IGLOO devices refer to all of the IGLOO devices as listed in Table 16-1. Where the information applies to only one product line or limited devices, these exclusions will be explicitly stated.

ProASIC3 Terminology

In documentation, the terms ProASIC3 series and ProASIC3 devices refer to all of the ProASIC3 devices as listed in Table 16-1. Where the information applies to only one product line or limited devices, these exclusions will be explicitly stated.

To further understand the differences between the IGLOO and ProASIC3 devices, refer to the *Industry's Lowest Power FPGAs Portfolio*.

SRAM Initialization

Users can also initialize embedded SRAMs of the low power flash devices. The initialization of the embedded SRAM blocks of the design can be done using UJTAG tiles, where the initialization data is imported using the TAP Controller. Similar functionality is available in ProASICPLUS devices using JTAG. The guidelines for implementation and design examples are given in the *RAM Initialization and ROM Emulation in ProASICPLUS Devices* application note.

SRAMs are volatile by nature; data is lost in the absence of power. Therefore, the initialization process should be done at each power-up if necessary.

FlashROM Read-Back Using JTAG

The low power flash architecture contains a dedicated nonvolatile FlashROM block, which is formatted into eight 128-bit pages. For more information on FlashROM, refer to the "FlashROM in Microsemi's Low Power Flash Devices" section on page 133. The contents of FlashROM are available to the VersaTiles during normal operation through a read operation. As a result, the UJTAG macro can be used to provide the FlashROM contents to the JTAG port during normal operation. Figure 17-7 illustrates a simple block diagram of using UJTAG to read the contents of FlashROM during normal operation.

The FlashROM read address can be provided from outside the FPGA through the TDI input or can be generated internally using the core logic. In either case, data serialization logic is required (Figure 17-7) and should be designed using the VersaTile core logic. FlashROM contents are read asynchronously in parallel from the flash memory and shifted out in a synchronous serial format to TDO. Shifting the serial data out of the serialization block should be performed while the TAP is in UDRSH mode. The coordination between TCK and the data shift procedure can be done using the TAP state machine by monitoring UDRSH, UDRCAP, and UDRUPD.

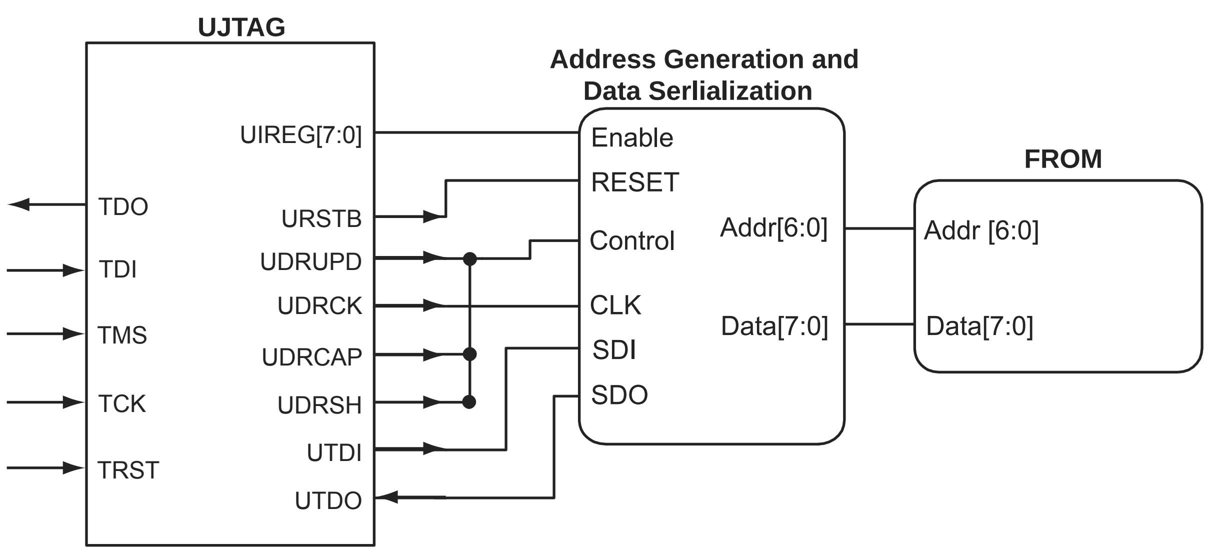

***Figure 17-7* • Block Diagram of Using UJTAG to Read FlashROM Contents**