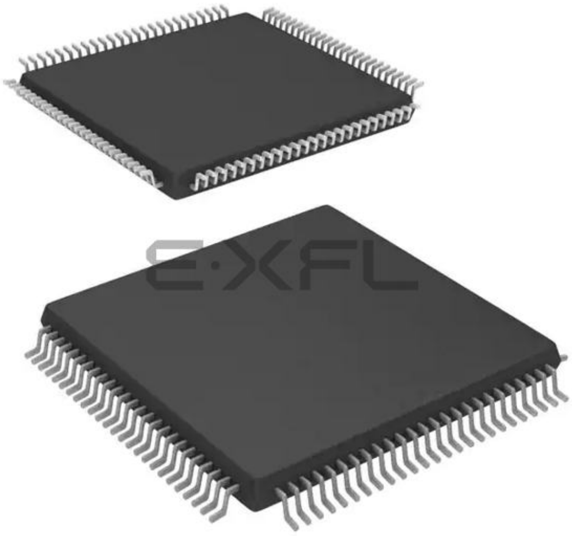


Welcome to E-XFL.COM

## 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 | 36864 |
| Number of I/O | 68 |
| Number of Gates | 250000 |
| Voltage - Supply | 1.425V ~ 1.575V |
| Mounting Type | Surface Mount |
| Operating Temperature | -40°C ~ 100°C (TJ) |
| Package / Case | 100-TQFP |
| Supplier Device Package | 100-VQFP (14x14) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/a3pn250-z2vq100i |

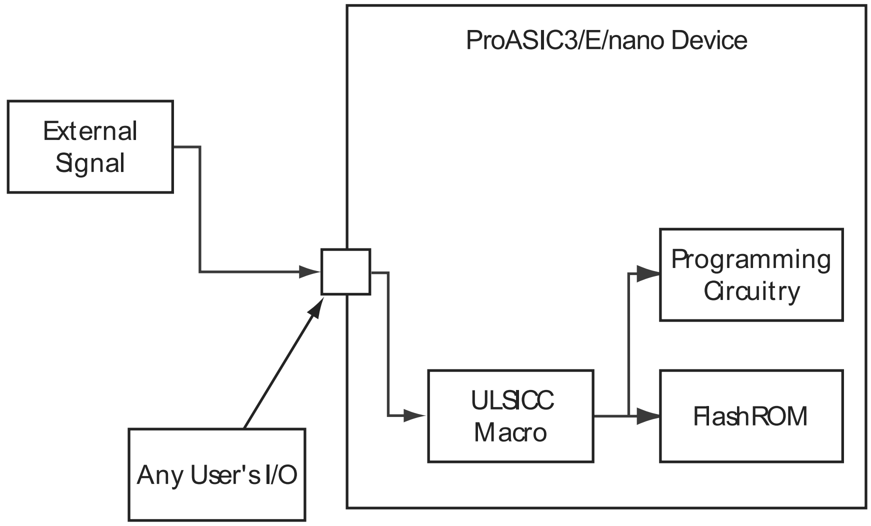



**Figure 2-3 • User Low Static (Idle) Mode Application—External Control Signal**

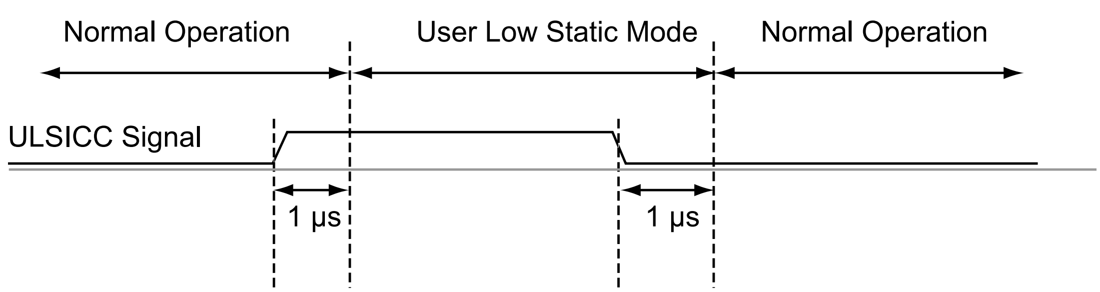



**Figure 2-4 • User Low Static (Idle) Mode Timing Diagram**

# Sleep Mode

ProASIC3/E and ProASIC3 nano FPGAs support Sleep mode when device functionality is not required. In Sleep mode, the VCC (core voltage), VJTAG (JTAG DC voltage), and VPUMP (programming voltage) are grounded, resulting in the FPGA core being turned off to reduce power consumption. While the ProASIC3/E device is in Sleep mode, the rest of the system is still operating and driving the input buffers of the ProASIC3/E device. The driven inputs do not pull up power planes, and the current draw is limited to a minimal leakage current.

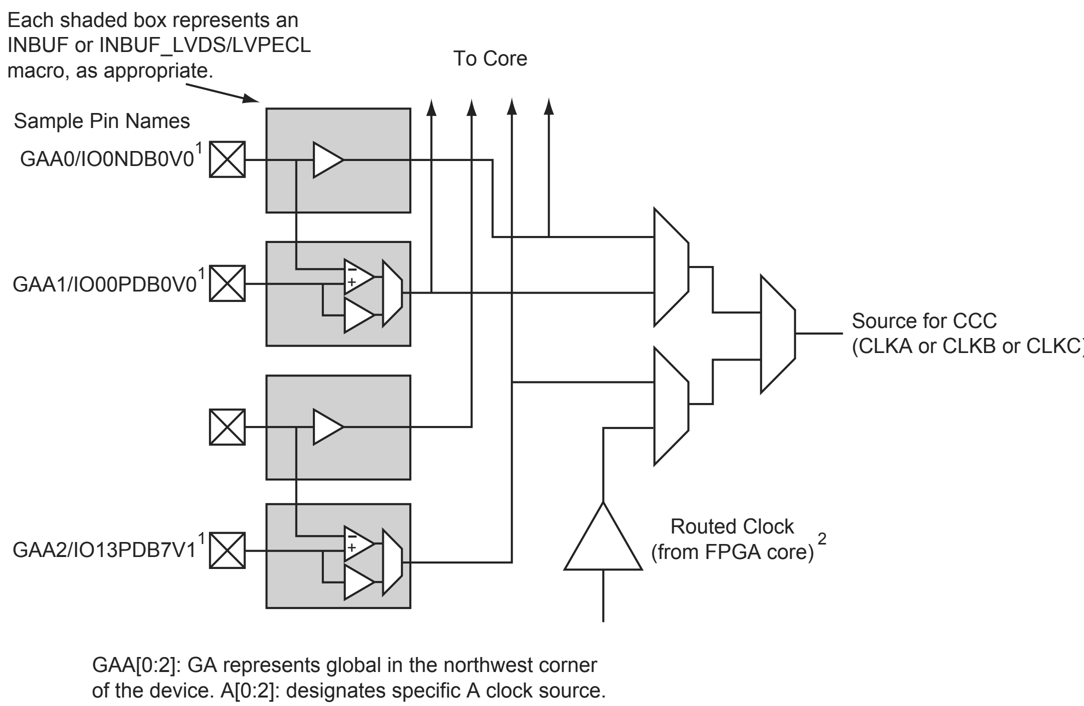
Table 2-3 shows the status of the power supplies in Sleep mode. When a power supply is powered off, the corresponding power pin can be left floating or grounded.

***Table 2-3* • Sleep Mode—Power Supply Requirements for ProASIC3/E/nano Devices**

| Power Supplies | ProASIC3/E/nano Device |
|---|---|
| VCC | Powered off |
| VCCI = VMV | Powered on |
| VJTAG | Powered off |
| VPUMP | Powered off |

Figure 3-5 shows more detailed global input connections. It shows the global input pins connection to the northwest quadrant global networks. Each global buffer, as well as the PLL reference clock, can be driven from one of the following:

- 3 dedicated single-ended I/Os using a hardwired connection
- 2 dedicated differential I/Os using a hardwired connection (not supported for IGLOO nano or ProASIC3 nano devices)
- The FPGA core




*Note: Differential inputs are not supported for IGLOO nano or ProASIC3 nano devices.*

***Figure 3-5* • Global I/O Overview**

## Implementing EXTFB in ProASIC3/E Devices

When the external feedback (EXTFB) signal of the PLL in the ProASIC3/E devices is implemented, the phase detector of the PLL core receives the reference clock (CLKA) and EXTFB as inputs. EXTFB must be sourced as an INBUF macro and located at the global/chip clock location associated with the target PLL by Designer software. EXTFB cannot be sourced from the FPGA fabric.

The following example shows CLKA and EXTFB signals assigned to two global I/Os in the same global area of ProASIC3E device.

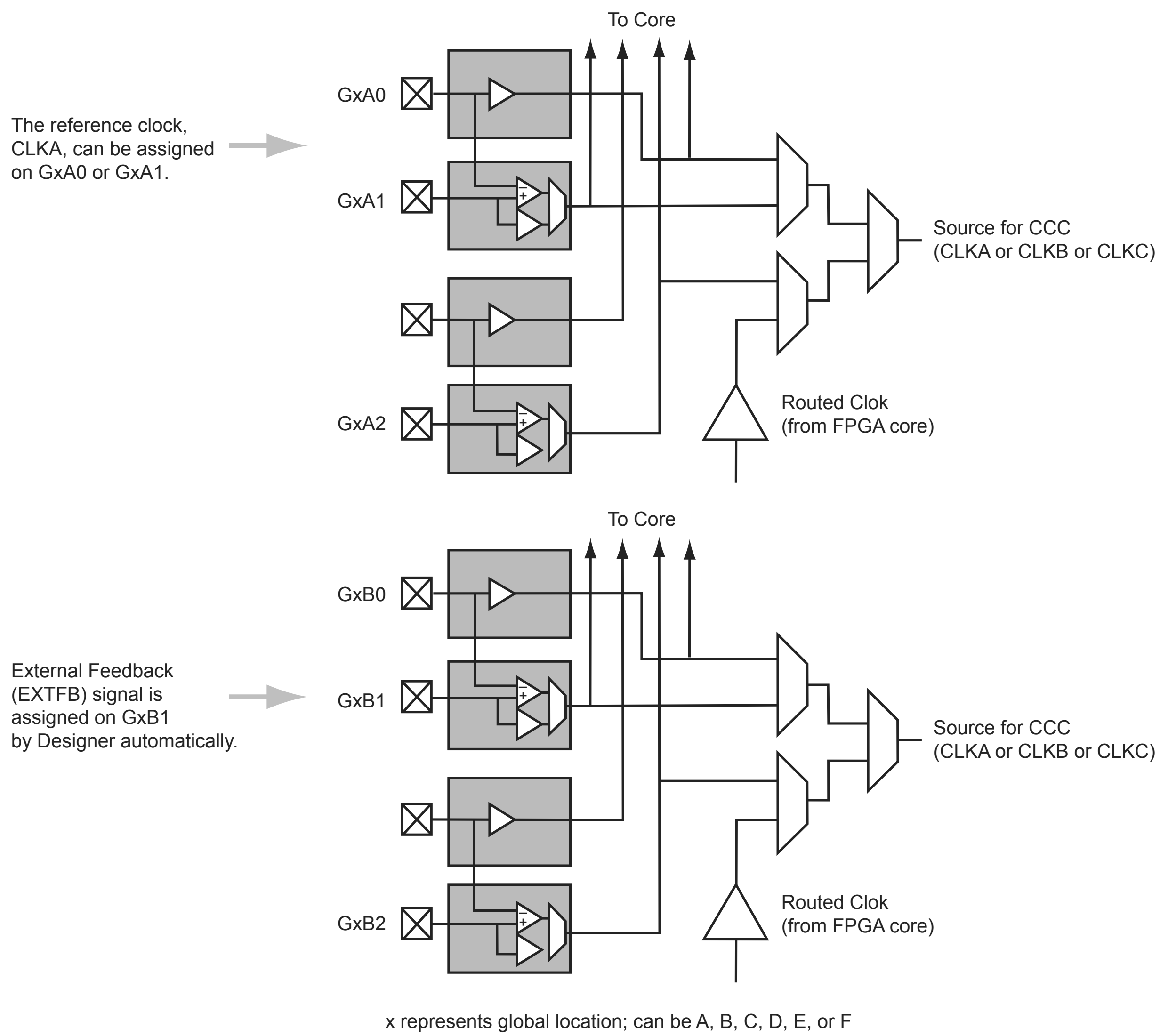



*Figure 4-5* • **CLKA and EXTFB Assigned to Global I/Os**

Each group of control bits is assigned a specific location in the configuration shift register. For a list of the 81 configuration bits (C[80:0]) in the CCC and a description of each, refer to "PLL Configuration Bits Description" on page 90. The configuration register can be serially loaded with the new configuration data and programmed into the CCC using the following ports:

- SDIN: The configuration bits are serially loaded into a shift register through this port. The LSB of the configuration data bits should be loaded first.
- SDOUT: The shift register contents can be shifted out (LSB first) through this port using the shift operation.
- SCLK: This port should be driven by the shift clock.
- SSHIFT: The active-high shift enable signal should drive this port. The configuration data will be shifted into the shift register if this signal is HIGH. Once SSHIFT goes LOW, the data shifting will be halted.
- SUPDATE: The SUPDATE signal is used to configure the CCC with the new configuration bits when shifting is complete.

To access the configuration ports of the shift register (SDIN, SDOUT, SSHIFT, etc.), the user should instantiate the CCC macro in his design with appropriate ports. Microsemi recommends that users choose SmartGen to generate the CCC macros with the required ports for dynamic reconfiguration.

Users must familiarize themselves with the architecture of the CCC core and its input, output, and configuration ports to implement the desired delay and output frequency in the CCC structure. Figure 4-22 shows a model of the CCC with configurable blocks and switches.

The following is an example of a PLL configuration utilizing the clock frequency synthesis and clock delay adjustment features. The steps include generating the PLL core with SmartGen, performing simulation for verification with Model*Sim*, and performing static timing analysis with SmartTime in Designer.

Parameters of the example PLL configuration:

Input Frequency – 20 MHz

Primary Output Requirement – 20 MHz with clock advancement of 3.02 ns

Secondary 1 Output Requirement – 40 MHz with clock delay of 2.515 ns

Figure 4-29 shows the SmartGen settings. Notice that the overall delays are calculated automatically, allowing the user to adjust the delay elements appropriately to obtain the desired delays.

***Figure 4-29*** **• SmartGen Settings**

After confirming the correct settings, generate a structural netlist of the PLL and verify PLL core settings by checking the log file:

```
Name                            : test_pll_delays
Family                          : ProASIC3E
Output Format                   : VHDL
Type                            : Static PLL
Input Freq(MHz)                 : 20.000
CLKA Source                     : Hardwired I/O
Feedback Delay Value Index      : 21
Feedback Mux Select             : 2
XDLY Mux Select                 : No
Primary Freq(MHz)               : 20.000
Primary PhaseShift              : 0
Primary Delay Value Index       : 1
Primary Mux Select              : 4
Secondary1 Freq(MHz)            : 40.000
Use GLB                         : YES
Use YB                          : NO
…
…
…
Primary Clock frequency 20.000
Primary Clock Phase Shift 0.000
```

DEVICE_INFO displays the FlashROM content, serial number, Design Name, and checksum, as shown below:

```
EXPORT IDCODE[32] = 123261CF
EXPORT SILSIG[32] = 00000000
User information :
CHECKSUM: 61A0
Design Name:        TOP
Programming Method: STAPL
Algorithm Version: 1
Programmer: UNKNOWN
==========================================
FlashROM Information :
EXPORT Region_7_0[128] = FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
==========================================
Security Setting :
Encrypted FlashROM Programming Enabled.
Encrypted FPGA Array Programming Enabled.
==========================================
```

The Libero SoC file manager recognizes the UFC and MEM files and displays them in the appropriate view. Libero SoC also recognizes the multiple programming files if you choose the option to generate multiple files for multiple FlashROM contents in Designer. These features enable a user-friendly flow for the FlashROM generation and programming in Libero SoC.

# Custom Serialization Using FlashROM

You can use FlashROM for device serialization or inventory control by using the Auto Inc region or Read From File region. FlashPoint will automatically generate the serial number sequence for the Auto Inc region with the **Start Value**, **Max Value**, and **Step Value** provided. If you have a unique serial number generation scheme that you prefer, the Read From File region allows you to import the file with your serial number scheme programmed into the region. See the *FlashPro User's Guide* for custom serialization file format information.

The following steps describe how to perform device serialization or inventory control using FlashROM:

1. Generate FlashROM using SmartGen. From the Properties section in the FlashROM Settings dialog box, select the **Auto Inc** or **Read From File** region. For the Auto Inc region, specify the desired step value. You will not be able to modify this value in the FlashPoint software.
2. Go through the regular design flow and finish place-and-route.
3. Select **Programming File in Designer** and open **Generate Programming File** (Figure 5-12 on page 128).
4. Click **Program FlashROM**, browse to the UFC file, and click **Next**. The FlashROM Settings window appears, as shown in Figure 5-13 on page 128.
5. Select the FlashROM page you want to program and the data value for the configured regions. The STAPL file generated will contain only the data that targets the selected FlashROM page.
6. Modify properties for the serialization.
   – For the Auto Inc region, specify the **Start** and **Max** values.
   – For the Read From File region, select the file name of the custom serialization file.
7. Select the FlashROM programming file type you want to generate from the two options below:
   – Single programming file for all devices: generates one programming file with all FlashROM values.
   – One programming file per device: generates a separate programming file for each FlashROM value.
8. Enter the number of devices you want to program and generate the required programming file.
9. Open the programming software and load the programming file. The programming software, FlashPro3 and Silicon Sculptor II, supports the device serialization feature. If, for some reason, the device fails to program a part during serialization, the software allows you to reuse or skip the serial data. Refer to the *FlashPro User's Guide* for details.

Note: **When using the SRAM in single-port mode for Automotive ProASIC3 devices, ADDRB should be tied to ground.**

***Table 6-3* • Address Pins Unused/Used for Various Supported Bus Widths**

| D×W | ADDRx | |
|---|---|---|
| | **Unused** | **Used** |
| 4k×1 | None | [11:0] |
| 2k×2 | [11] | [10:0] |
| 1k×4 | [11:10] | [9:0] |
| 512×9 | [11:9] | [8:0] |

*Note: The "x" in ADDRx implies A or B.*

### DINA and DINB

These are the input data signals, and they are nine bits wide. Not all nine bits are valid in all configurations. When a data width less than nine is specified, unused high-order signals must be grounded (Table 6-4).

Note: **When using the SRAM in single-port mode for Automotive ProASIC3 devices, DINB should be tied to ground.**

### DOUTA and DOUTB

These are the nine-bit output data signals. Not all nine bits are valid in all configurations. As with DINA and DINB, high-order bits may not be used (Table 6-4). The output data on unused pins is undefined.

***Table 6-4* • Unused/Used Input and Output Data Pins for Various Supported Bus Widths**

| D×W | DINx/DOUTx | |
|---|---|---|
| | **Unused** | **Used** |
| 4k×1 | [8:1] | [0] |
| 2k×2 | [8:2] | [1:0] |
| 1k×4 | [8:4] | [3:0] |
| 512×9 | None | [8:0] |

*Note: The "x" in DINx or DOUTx implies A or B.*

## *RAM512X18 Macro*

RAM512X18 is the two-port configuration of the same RAM block (Figure 6-5 on page 140). Like the RAM4K9 nomenclature, the RAM512X18 nomenclature refers to both the deepest possible configuration and the widest possible configuration the two-port RAM block can assume. In two-port mode, the RAM block can be configured to either the 512×9 aspect ratio or the 256×18 aspect ratio. RAM512X18 is also fully synchronous and has the following features:

- Dedicated read and write ports
- Active-low read and write enables
- Selectable pipelined or nonpipelined read
- Active-low asynchronous reset
- Designer software will automatically facilitate falling-edge clocks by bubble-pushing the inversion to previous stages.

# Instantiating DDR Registers

Using SmartGen is the simplest way to generate the appropriate RTL files for use in the design. Figure 9-4 shows an example of using SmartGen to generate a DDR SSTL2 Class I input register. SmartGen provides the capability to generate all of the DDR I/O cells as described. The user, through the graphical user interface, can select from among the many supported I/O standards. The output formats supported are Verilog, VHDL, and EDIF.

Figure 9-5 on page 211 through Figure 9-8 on page 214 show the I/O cell configured for DDR using SSTL2 Class I technology. For each I/O standard, the I/O pad is buffered by a special primitive that indicates the I/O standard type.

***Figure 9-4*** **• Example of Using SmartGen to Generate a DDR SSTL2 Class I Input Register**

## Design Example

Figure 9-9 shows a simple example of a design using both DDR input and DDR output registers. The user can copy the HDL code in Libero SoC software and go through the design flow. Figure 9-10 and Figure 9-11 on page 217 show the netlist and ChipPlanner views of the ddr_test design. Diagrams may vary slightly for different families.

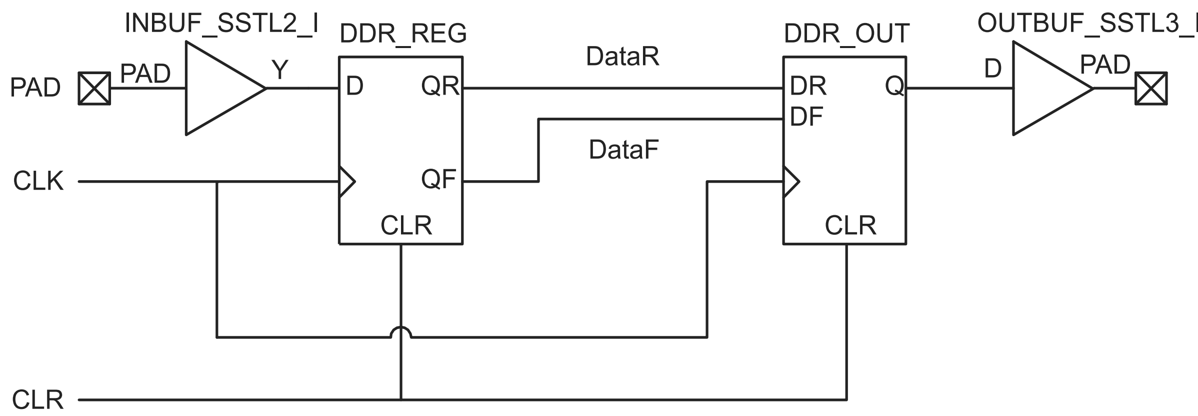



*Figure 9-9* **• Design Example**

*Figure 9-10* **• DDR Test Design as Seen by NetlistViewer for IGLOO/e Devices**

## *Device Programmers*

### Single Device Programmer

Single device programmers are used to program a device before it is mounted on the system board.

The advantage of using device programmers is that no programming hardware is required on the system board. Therefore, no additional components or board space are required.

Adapter modules are purchased with single device programmers to support the FPGA packages used. The FPGA is placed in the adapter module and the programming software is run from a PC. Microsemi supplies the programming software for all of the Microsemi programmers. The software allows for the selection of the correct die/package and programming files. It will then program and verify the device.

- Single-site programmers

  A single-site programmer programs one device at a time. Microsemi offers Silicon Sculptor 3, built by BP Microsystems, as a single-site programmer. Silicon Sculptor 3 and associated software are available only from Microsemi.

  – Advantages: Lower cost than multi-site programmers. No additional overhead for programming on the system board. Allows local control of programming and data files for maximum security. Allows on-demand programming on-site.

  – Limitations: Only programs one device at a time.

- Multi-site programmers

Often referred to as batch or gang programmers, multi-site programmers can program multiple devices at the same time using the same programming file. This is often used for large volume programming and by programming houses. The sites often have independent processors and memory enabling the sites to operate concurrently, meaning each site may start programming the same file independently. This enables the operator to change one device while the other sites continue programming, which increases throughput. Multiple adapter modules for the same package are required when using a multi-site programmer. Silicon Sculptor I, II, and 3 programmers can be cascaded to program multiple devices in a chain. Multi-site programmers, such as the BP2610 and BP2710, can also be purchased from BP Microsystems. When using BP Microsystems multi-site programmers, users must use programming adapter modules available only from Microsemi. Visit the Microsemi SoC Products Group website to view the part numbers of the desired adapter module:

http://www.microsemi.com/soc/products/hardware/program_debug/ss/modules.aspx.

Also when using BP Microsystems programmers, customers must use Microsemi programming software to ensure the best programming result will occur.

  – Advantages: Provides the capability of programming multiple devices at the same time. No additional overhead for programming on the system board. Allows local control of programming and data files for maximum security.

  – Limitations: More expensive than a single-site programmer

- Automated production (robotic) programmers

Automated production programmers are based on multi-site programmers. They consist of a large input tray holding multiple parts and a robotic arm to select and place parts into appropriate programming sockets automatically. When the programming of the parts is complete, the parts are removed and placed in a finished tray. The automated programmers are often used in volume programming houses to program parts for which the programming time is small. BP Microsystems part number BP4710, BP4610, BP3710 MK2, and BP3610 are available for this purpose. Auto programmers cannot be used to program RTAX-S devices.

Where an auto-programmer is used, the appropriate open-top adapter module from BP Microsystems must be used.

# Security in ARM-Enabled Low Power Flash Devices

There are slight differences between the regular flash device and the ARM-enabled flash devices, which have the M1 prefix.

The AES key is used by Microsemi and preprogrammed into the device to protect the ARM IP. As a result, the design will be encrypted along with the ARM IP, according to the details below.

## Cortex-M1 and Cortex-M3 Device Security

Cortex-M1–enabled and Cortex-M3 devices are shipped with the following security features:

- FPGA array enabled for AES-encrypted programming and verification
- FlashROM enabled for AES-encrypted write and verify
- Embedded Flash Memory enabled for AES encrypted write

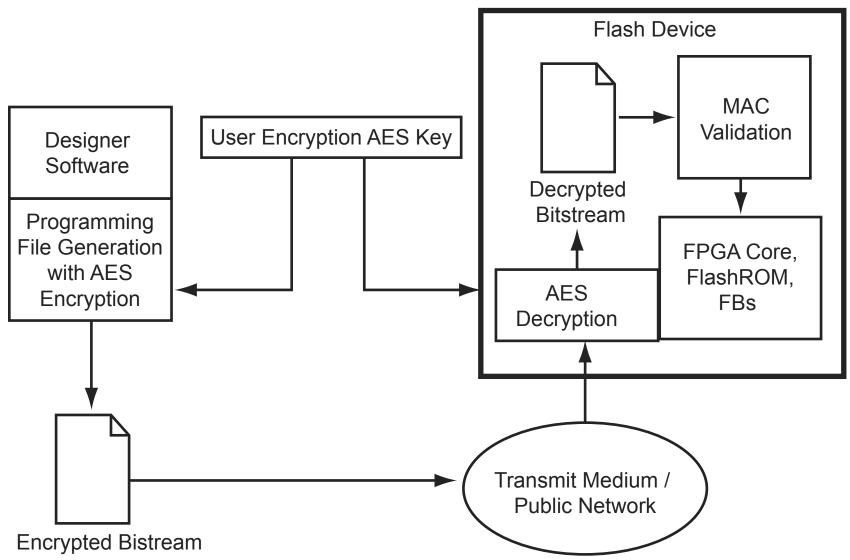
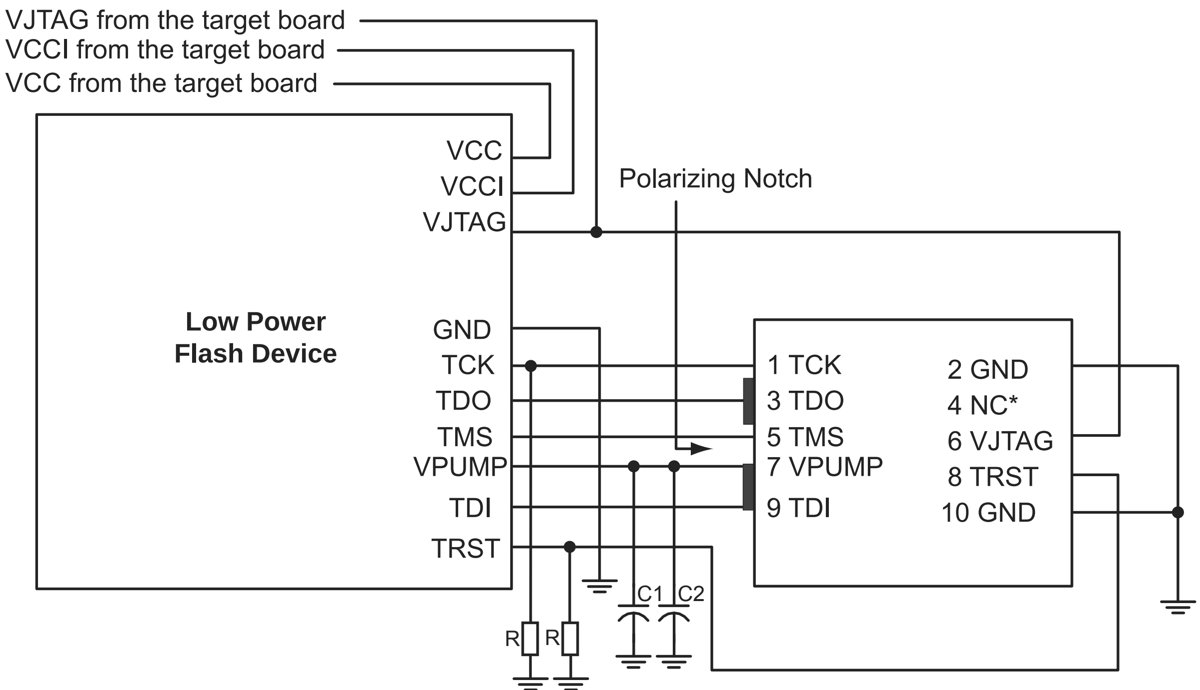



***Figure 12-1*** **• AES-128 Security Features**

# Board-Level Considerations

A bypass capacitor is required from VPUMP to GND for all low power flash devices during programming. This bypass capacitor protects the devices from voltage spikes that may occur on the VPUMP supplies during the erase and programming cycles. Refer to the "Pin Descriptions and Packaging" chapter of the appropriate device datasheet for specific recommendations. For proper programming, 0.01 µF and 0.33 µF capacitors (both rated at 16 V) are to be connected in parallel across VPUMP and GND, and positioned as close to the FPGA pins as possible. The bypass capacitor must be placed within 2.5 cm of the device pins.




*Note: *NC (FlashPro3/3X); Prog_Mode (FlashPro4). Prog_Mode on FlashPro4 is an output signal that goes High during device programming and returns to Low when programming is complete. This signal can be used to drive a system to provide a 1.5 V programming signal to IGLOO nano, ProASIC3L, and RT ProASIC3 devices that can run with 1.2 V core voltage but require 1.5 V for programming. IGLOO nano V2 devices can be programmed at 1.2 V core voltage (when using FlashPro4 only), but IGLOO nano V5 devices are programmed with a VCC core voltage of 1.5 V.*

***Figure 12-6* • Board Layout and Programming Header Top View**

## Troubleshooting Signal Integrity

### *Symptoms of a Signal Integrity Problem*

A signal integrity problem can manifest itself in many ways. The problem may show up as extra or dropped bits during serial communication, changing the meaning of the communication. There is a normal variation of threshold voltage and frequency response between parts even from the same lot. Because of this, the effects of signal integrity may not always affect different devices on the same board in the same way. Sometimes, replacing a device appears to make signal integrity problems go away, but this is just masking the problem. Different parts on identical boards will exhibit the same problem sooner or later. It is important to fix signal integrity problems early. Unless the signal integrity problems are severe enough to completely block all communication between the device and the programmer, they may show up as subtle problems. Some of the FlashPro4/3/3X exit codes that are caused by signal integrity problems are listed below. Signal integrity problems are not the only possible cause of these

# Circuit Verification

The power switching circuit recommended above is implemented on Microsemi's Icicle board (Figure 13-2). On the Icicle board, VJTAGENB is used to control the N-Channel Digital FET; however, this circuit was modified to use TRST instead of VJTAGENB in this application. There are three important aspects of this circuit that were verified:

1. The rise on VCC from 1.2 V to 1.5 V when TRST is HIGH
2. VCC rises to 1.5 V before programming begins.
3. VCC switches from 1.5 V to 1.2 V when TRST is LOW.

## Verification Steps

1. The rise on VCC from 1.2 V to 1.5 V when TRST is HIGH.

***Figure 13-2*** **• Core Voltage on the IGLOO AGL125-QNG132 Device**

In the oscilloscope plots (Figure 13-2), the TRST from FlashPro3 and the VCC core voltage of the IGLOO device are labeled. This plot shows the rise characteristic of the TRST signal from FlashPro3. Once the TRST signal is asserted HIGH, the LTC3025 shown in Figure 13-1 on page 277 senses the increase in voltage and changes the output from 1.2 V to 1.5 V. It takes the circuit approximately 100 μs to respond to TRST and change the voltage to 1.5 V on the VCC core.

3. VCC switches from 1.5 V to 1.2 V when TRST is LOW.

***Figure 13-4* • TRST Toggled LOW**

In Figure 13-4, the TRST signal and the VCC core voltage signal are labeled. As TRST is pulled to ground, the core voltage is observed to switch from 1.5 V to 1.2 V. The observed fall time is approximately 2 ms.

# DirectC

The above analysis is based on FlashPro3, but there are other solutions to ISP, such as DirectC. DirectC is a microprocessor program that can be run in-system to program Microsemi flash devices. For FlashPro3, TRST is the most convenient control signal to use for the recommended circuit. However, for DirectC, users may use any signal to control the FET. For example, the DirectC code can be edited so that a separate non-JTAG signal can be asserted from the microcontroller that signals the board that it is about to start programming the device. After asserting the N-Channel Digital FET control signal, the programming algorithm must allow sufficient time for the supply to rise to 1.5 V before initiating DirectC programming. As seen in Figure 13-3 on page 279, 50 ms is adequate time. Depending on the size of the PCB and the capacitance on the VCC supply, results may vary from system to system. Microsemi recommends using a conservative value for the wait time to make sure that the VCC core voltage is at the right level.

# Conclusion

For applications using IGLOO and ProASIC3L low power FPGAs and taking advantage of the low core voltage power supplies with less than 1.5 V operation, there must be a way for the core voltage to switch from 1.2 V (or other voltage) to 1.5 V, which is required during in-system programming. The circuit explained in this document illustrates one simple, cost-effective way of handling this requirement. A JTAG signal from the FlashPro3 programmer allows the circuit to sense when programming is in progress, enabling it to switch to the correct core voltage.

# List of Changes

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

| Date | Changes | Page |
|---|---|---|
| September 2012 | The "Security" section was modified to clarify that Microsemi does not support read-back of FPGA core-programmed data (SAR 41235). | 288 |
| 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<br>(December 2008) | IGLOO nano and ProASIC3 nano devices were added to Table 14-1 • Flash-Based FPGAs. | 284 |
| v1.3<br>(October 2008) | The "Microprocessor Programming Support in Flash Devices" section was revised to include new families and make the information more concise. | 284 |
| v1.2<br>(June 2008) | The following changes were made to the family descriptions in Table 14-1 • Flash-Based FPGAs:<br>• ProASIC3L was updated to include 1.5 V.<br>• The number of PLLs for ProASIC3E was changed from five to six. | 284 |
| v1.1<br>(March 2008) | The "Microprocessor Programming Support in Flash Devices" section was updated to include information on the IGLOO PLUS family. The "IGLOO Terminology" section and "ProASIC3 Terminology" section are new. | 284 |

# Related Documents

## Datasheets

*ProASIC3 Flash Family FPGAs*

http://www.microsemi.com/soc/documents/PA3_DS.pdf

*ProASIC3E Flash Family FPGAs*

http://www.microsemi.com/soc/documents/PA3E_DS.pdf

# List of Changes

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

| Date | Changes | Page |
|---|---|---|
| v1.2 (December 2008) | IGLOO nano and ProASIC3 nano devices were added to the document as supported device types. | |
| v1.1 (October 2008) | The "Introduction" section was updated to add Military ProASIC3EL and RT ProASIC3 devices to the list of devices that can have inputs driven in while the device is not powered. | 307 |
| | The "Flash Devices Support Power-Up Behavior" section was revised to include new families and make the information more concise. | 308 |
| | The "Cold-Sparing" section was revised to add Military ProASIC3/EL and RT ProASIC3 devices to the lists of devices with and without cold-sparing support. | 316 |
| | The "Hot-Swapping" section was revised to add Military ProASIC3/EL and RT ProASIC3 devices to the lists of devices with and without hot-swap support. AGL400 was added to the list of devices that do not support hot-swapping. | 317 |
| v1.0 (August 2008) | This document was revised, renamed, and assigned a new part number. It now includes data for the IGLOO and ProASIC3L families. | N/A |
| v1.3 (March 2008) | The "List of Changes" section was updated to include the three different I/O Structure handbook chapters. | 318 |
| v1.2 (February 2008) | The first sentence of the "PLL Behavior at Brownout Condition" section was updated to read, "When PLL power supply voltage and/or $V_{CC}$ levels drop below the VCC brownout levels (0.75 V ± 0.25 V), the PLL output lock signal goes low and/or the output clock is lost." | 315 |
| v1.1 (January 2008) | The "PLL Behavior at Brownout Condition" section was added. | 315 |

| Revision (month/year) | Chapter Affected | List of Changes (page number) |
|---|---|---|
| Revision 1 (continued) | "In-System Programming (ISP) of Microsemi's Low Power Flash Devices Using FlashPro4/3/3X" was revised. | 273 |
| | "Core Voltage Switching Circuit for IGLOO and ProASIC3L In-System Programming" was revised. | 281 |
| | "Boundary Scan in Low Power Flash Devices" was revised. | 296 |
| Revision 0 (April 2010) | The ProASIC3 nano Low Power Flash FPGAs Handbook was divided into two parts to create the ProASIC3 nano Datasheet ProASIC3 nano Device Family User's Guide. | N/A |

# Index

**Microsemi Corporate Headquarters**
One Enterprise, Aliso Viejo CA 92656 USA
Within the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996

Microsemi Corporation (NASDAQ: MSCC) offers a comprehensive portfolio of semiconductor solutions for: aerospace, defense and security; enterprise and communications; and industrial and alternative energy markets. Products include high-performance, high-reliability analog and RF devices, mixed signal and RF integrated circuits, customizable SoCs, FPGAs, and complete subsystems. Microsemi is headquartered in Aliso Viejo, Calif. Learn more at **www.microsemi.com**.



50200255-5/9.12