



Welcome to [E-XFL.COM](https://www.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	Active
Number of LABs/CLBs	-
Number of Logic Elements/Cells	-
Total RAM Bits	147456
Number of I/O	97
Number of Gates	1000000
Voltage - Supply	1.14V ~ 1.575V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	144-LBGA
Supplier Device Package	144-FPBGA (13x13)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/m1a3p1000l-fg144i

Core Architecture

VersaTile

The proprietary IGLOO and ProASIC3 device architectures provide granularity comparable to gate arrays. The device core consists of a sea-of-VersaTiles architecture.

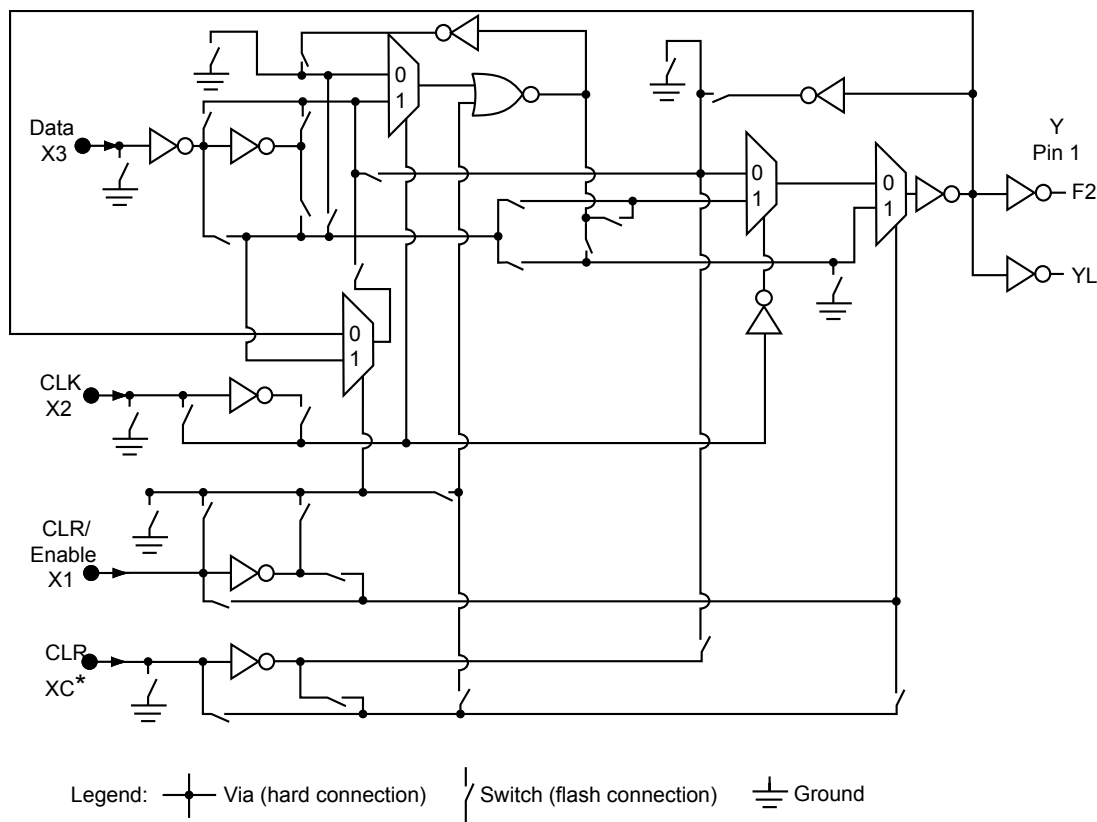
As illustrated in Figure 1-8, there are four inputs in a logic VersaTile cell, and each VersaTile can be configured using the appropriate flash switch connections:

- Any 3-input logic function
- Latch with clear or set
- D-flip-flop with clear or set
- Enable D-flip-flop with clear or set (on a 4th input)

VersaTiles can flexibly map the logic and sequential gates of a design. The inputs of the VersaTile can be inverted (allowing bubble pushing), and the output of the tile can connect to high-speed, very-long-line routing resources. VersaTiles and larger functions can be connected with any of the four levels of routing hierarchy.

When the VersaTile is used as an enable D-flip-flop, SET/CLR is supported by a fourth input. The SET/CLR signal can only be routed to this fourth input over the VersaNet (global) network. However, if, in the user's design, the SET/CLR signal is not routed over the VersaNet network, a compile warning message will be given, and the intended logic function will be implemented by two VersaTiles instead of one.

The output of the VersaTile is F2 when the connection is to the ultra-fast local lines, or YL when the connection is to the efficient long-line or very-long-line resources.



* This input can only be connected to the global clock distribution network.

Figure 1-8 • Low Power Flash Device Core VersaTile

Set/Reset

Since all I/Os and globals are tied High in Flash*Freeze mode (unless hold state is used on IGLOO nano or IGLOO PLUS), Microsemi recommends using active low set/reset at the top-level port. If needed, the signal can be inverted internally.

- If the intention is to always set/reset in Flash*Freeze mode, a self set/reset circuit may be implemented to accomplish this, as shown in Figure 2-9. Configure an active High set/reset input pin so it uses the internal pull-up during Flash*Freeze mode, and drives Low during active mode. When the device exits Flash*Freeze mode, the input will transition from High to Low, releasing the set/reset. Note that this circuit may release set/reset before all outputs become active, since outputs are enabled up to 200 ns after inputs when exiting Flash*Freeze mode.

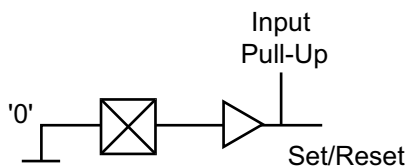


Figure 2-9 • Flash*Freeze Self-Reset Circuit

I/Os

- Floating inputs can cause totem pole currents on the input I/O circuitry when the device is in active mode. If inputs will be released (undriven) during Flash*Freeze mode, Microsemi recommends that they are only released after the device enters Flash*Freeze mode.
- As mentioned earlier, asynchronous input to output paths are subject to possible glitching when entering Flash*Freeze mode. For example, on a direct in-to-out path, if the current state is '0' and the input bank deactivates first, the input and then the output will transition to '1' before the output enters its Flash*Freeze state. This can be prevented by using latches along with Flash*Freeze management IP to gate asynchronous in-to-out paths prior to entering Flash*Freeze mode.

JTAG

- The JTAG state machine is powered but not active during Flash*Freeze mode.
- TCK should be held in a static state to prevent dynamic power consumption of the JTAG circuit during Flash*Freeze.
- Specific JTAG pin tie-off recommendations suitable for Flash*Freeze mode can be found in the "Pin Descriptions and Packaging" chapter of the device datasheet.

ULSICC

- The User Low Static ICC (ULSICC) macro acts as an access point to the hard Flash*Freeze technology block in the device. The ULSICC macro represents a hard, fixed location block in the device. When the LSICC input of the ULSICC macro is driven Low, the Flash*Freeze pin is blocked, and when LSICC is driven High, the Flash*Freeze pin is enabled.
- If the user decides to build his/her own Flash*Freeze type 2 clock and data management logic, note that the LSICC signal on the ULSICC macro is ANDed internally with the Flash*Freeze signal. In order to reliably enter Flash*Freeze, the LSICC signal must remain asserted High while entering and during Flash*Freeze mode.

Flash*Freeze Management IP

One of the key benefits of Microsemi's Flash*Freeze mode is the ability to preserve the state of all internal registers, SRAM content, and I/Os (IGLOO nano and IGLOO PLUS only). This feature enables seamless continuation of data processing before and after Flash*Freeze, without the need to reload or reinitialize the FPGA system. Microsemi's Flash*Freeze management IP, available for type 2 implementation, offers a robust RTL block that ensures clean clock gating of all system clocks before entering and upon exiting Flash*Freeze mode. This IP also gives users the option to perform housekeeping prior to entering Flash*Freeze mode. This section will provide an overview of the

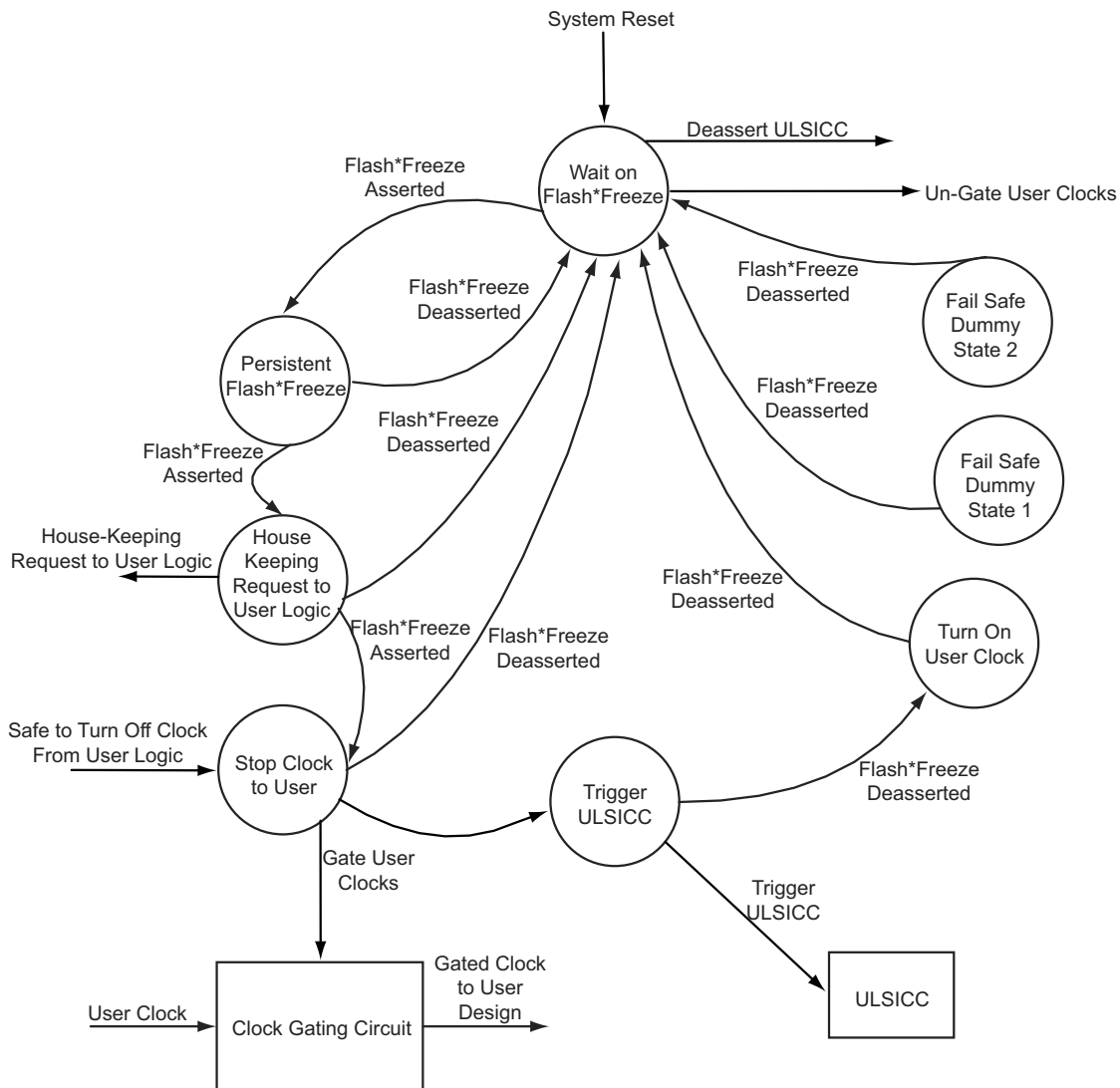


Figure 2-11 • FSM State Diagram

Table 3-5 • Globals/Spines/Rows for IGLOO PLUS Devices

IGLOO PLUS Devices	Chip Globals	Quadrant Globals (4x3)	Clock Trees	Globals/ Spines per Tree	Total Spines per Device	VersaTiles in Each Tree	Total VersaTiles	Rows in Each Spine
AGLP030	6	0	2	9	18	384*	792	12
AGLP060	6	12	4	9	36	384*	1,584	12
AGLP125	6	12	8	9	72	384*	3,120	12

Note: *Clock trees that are located at far left and far right will support more VersaTiles.

Table 3-6 • Globals/Spines/Rows for Fusion Devices

Fusion Device	Chip Globals	Quadrant Globals (4x3)	Clock Trees	Globals/ Spines per Tree	Total Spines per Device	VersaTiles in Each Tree	Total VersaTiles	Rows in Each Spine
AFS090	6	12	6	9	54	384	2,304	12
AFS250	6	12	8	9	72	768	6,144	24
AFS600	6	12	12	9	108	1,152	13,824	36
AFS1500	6	12	20	9	180	1,920	38,400	60

The following will happen during demotion of a global signal to regular nets:

- CLKBUF_x becomes INBUF_x; CLKINT is removed from the netlist.
- The essential global macro, such as the output of the Clock Conditioning Circuit, cannot be demoted.
- No automatic buffering will happen.

Since no automatic buffering happens when a signal is demoted, this net may have a high delay due to large fanout. This may have a negative effect on the quality of the results. Microsemi recommends that the automatic global demotion only be used on small-fanout nets. Use clock networks for high-fanout nets to improve timing and routability.

Spine Assignment

The low power flash device architecture allows the global networks to be segmented and used as clock spines. These spines, also called local clock networks, enable the use of PDC or MVN to assign a signal to a spine.

PDC syntax to promote a net to a spine/local clock:

```
assign_local_clock -net netname -type [quadrant|chip] Tn|Bn|Tn:Bm
```

If the net is driven by a clock macro, Designer automatically demotes the clock net to a regular net before it is assigned to a spine. Nets driven by a PLL or CLKDLY macro cannot be assigned to a local clock.

When assigning a signal to a spine or quadrant global network using PDC (pre-compile), the Designer software will legalize the shared instances. The number of shared instances to be legalized can be controlled by compile options. If these networks are created in MVN (only quadrant globals can be created), no legalization is done (as it is post-compile). Designer does not do legalization between non-clock nets.

As an example, consider two nets, net_clk and net_reset, driving the same flip-flop. The following PDC constraints are used:

```
assign_local_clock -net net_clk -type chip T3
assign_local_clock -net net_reset -type chip T1:T2
```

During Compile, Designer adds a buffer in the reset net and places it in the T1 or T2 region, and places the flip-flop in the T3 spine region (Figure 3-16).

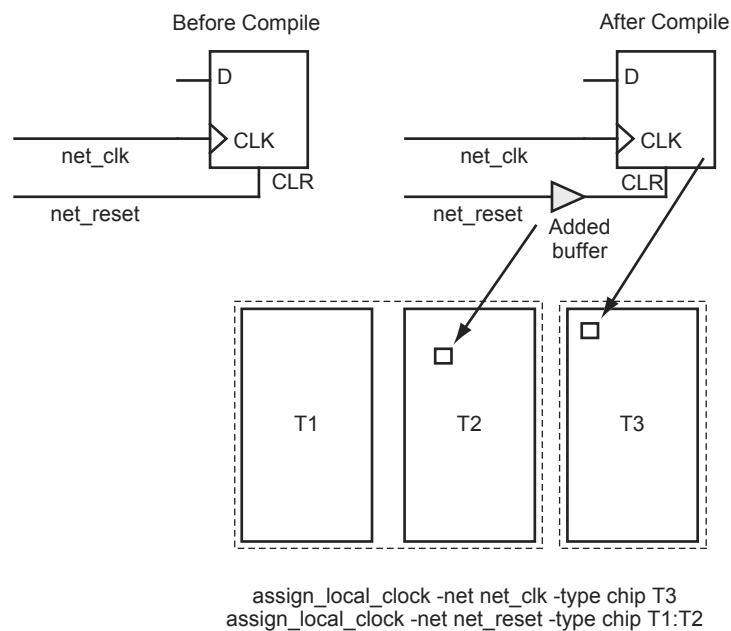


Figure 3-16 • Adding a Buffer for Shared Instances

Table 4-8 • Configuration Bit Descriptions for the CCC Blocks (continued)

Config. Bits	Signal	Name	Description
83	RXCSEL ¹	CLKC input selection	Select the CLKC input clock source between RC oscillator and crystal oscillator (refer to Table 4-16 on page 110). ²
82	RXBSEL ¹	CLKB input selection	Select the CLKB input clock source between RC oscillator and crystal oscillator (refer to Table 4-16 on page 110). ²
81	RXASEL ¹	CLKA input selection	Select the CLKA input clock source between RC oscillator and crystal oscillator (refer to Table 4-16 on page 110). ²
80	RESETEN	Reset Enable	Enables (active high) the synchronization of PLL output dividers after dynamic reconfiguration (SUPDATE). The Reset Enable signal is READ-ONLY.
79	DYNCSEL	Clock Input C Dynamic Select	Configures clock input C to be sent to GLC for dynamic control. ²
78	DYNBSEL	Clock Input B Dynamic Select	Configures clock input B to be sent to GLB for dynamic control. ²
77	DYNASEL	Clock Input A Dynamic Select	Configures clock input A for dynamic PLL configuration. ²
<76:74>	VCOSSEL[2:0]	VCO Gear Control	Three-bit VCO Gear Control for four frequency ranges (refer to Table 4-19 on page 111 and Table 4-20 on page 111).
73	STATCSEL	MUX Select on Input C	MUX selection for clock input C ²
72	STATBSEL	MUX Select on Input B	MUX selection for clock input B ²
71	STATASEL	MUX Select on Input A	MUX selection for clock input A ²
<70:66>	DLYC[4:0]	YC Output Delay	Sets the output delay value for YC.
<65:61>	DLYB[4:0]	YB Output Delay	Sets the output delay value for YB.
<60:56>	DLYGLC[4:0]	GLC Output Delay	Sets the output delay value for GLC.
<55:51>	DLYGLB[4:0]	GLB Output Delay	Sets the output delay value for GLB.
<50:46>	DLYGLA[4:0]	Primary Output Delay	Primary GLA output delay
45	XDLYSEL	System Delay Select	When selected, inserts System Delay in the feedback path in Figure 4-20 on page 101.
<44:40>	FBDLY[4:0]	Feedback Delay	Sets the feedback delay value for the feedback element in Figure 4-20 on page 101.
<39:38>	FBSEL[1:0]	Primary Feedback Delay Select	Controls the feedback MUX: no delay, include programmable delay element, or use external feedback.
<37:35>	OCMUX[2:0]	Secondary 2 Output Select	Selects from the VCO's four phase outputs for GLC/YC.
<34:32>	OBMUX[2:0]	Secondary 1 Output Select	Selects from the VCO's four phase outputs for GLB/YB.

Notes:

1. The <88:81> configuration bits are only for the Fusion dynamic CCC.
2. This value depends on the input clock source, so Layout must complete before these bits can be set. After completing Layout in Designer, generate the "CCC_Configuration" report by choosing **Tools > Report > CCC_Configuration**. The report contains the appropriate settings for these bits.

```
DYNCCC Core(.CLKA(CLKA), .EXTFB(GND), .POWERDOWN(POWERDOWN), .GLA(GLA), .LOCK(LOCK),
.CLKB(CLKB), .GLB(GLB), .YB(), .CLKC(CLKC), .GLC(GLC), .YC(), .SDIN(SDIN),
.SCLK(SCLK), .SShift(SShift), .SUPDATE(SUPDATE), .MODE(MODE), .SDOUT(SDOUT),
.OADIV0(GND), .OADIV1(GND), .OADIV2(VCC), .OADIV3(GND), .OADIV4(GND), .OAMUX0(GND),
.OAMUX1(GND), .OAMUX2(VCC), .DLYGLA0(GND), .DLYGLA1(GND), .DLYGLA2(GND),
.DLYGLA3(GND), .DLYGLA4(GND), .OBDIV0(GND), .OBDIV1(GND), .OBDIV2(GND),
.OBDIV3(GND), .OBDIV4(GND), .OBMUX0(GND), .OBMUX1(GND), .OBMUX2(GND), .DLYYB0(GND),
.DLYYB1(GND), .DLYYB2(GND), .DLYYB3(GND), .DLYYB4(GND), .DLYGLB0(GND),
.DLYGLB1(GND), .DLYGLB2(GND), .DLYGLB3(GND), .DLYGLB4(GND), .OCDIV0(GND),
.OCDIV1(GND), .OCDIV2(GND), .OCDIV3(GND), .OCDIV4(GND), .OCMUX0(GND), .OCMUX1(GND),
.OCMUX2(GND), .DLYYC0(GND), .DLYYC1(GND), .DLYYC2(GND), .DLYYC3(GND), .DLYYC4(GND),
.DLYGLC0(GND), .DLYGLC1(GND), .DLYGLC2(GND), .DLYGLC3(GND), .DLYGLC4(GND),
.FINDIV0(VCC), .FINDIV1(GND), .FINDIV2(VCC), .FINDIV3(GND), .FINDIV4(GND),
.FINDIV5(GND), .FINDIV6(GND), .FBDIV0(GND), .FBDIV1(GND), .FBDIV2(GND),
.FBDIV3(GND), .FBDIV4(GND), .FBDIV5(VCC), .FBDIV6(GND), .FBDLY0(GND), .FBDLY1(GND),
.FBDLY2(GND), .FBDLY3(GND), .FBDLY4(GND), .FBSEL0(VCC), .FBSEL1(GND),
.XDLYSEL(GND), .VCOSEL0(GND), .VCOSEL1(GND), .VCOSEL2(VCC));
defparam Core.VCOFREQUENCY = 165.000;

endmodule
```

Delayed Clock Configuration

The CLKDLY macro can be generated with the desired delay and input clock source (Hardwired I/O, External I/O, or Core Logic), as in Figure 4-28.

Figure 4-28 • Delayed Clock Configuration Dialog Box

After setting all the required parameters, users can generate one or more PLL configurations with HDL or EDIF descriptions by clicking the **Generate** button. SmartGen gives the option of saving session results and messages in a log file:

```
*****
Macro Parameters
*****
```

Name	: delay_macro
Family	: ProASIC3
Output Format	: Verilog
Type	: Delayed Clock
Delay Index	: 2
CLKA Source	: Hardwired I/O

Total Clock Delay = 0.935 ns.

The resultant CLKDLY macro Verilog netlist is as follows:

```
module delay_macro(GL,CLK);

output GL;
input CLK;
```

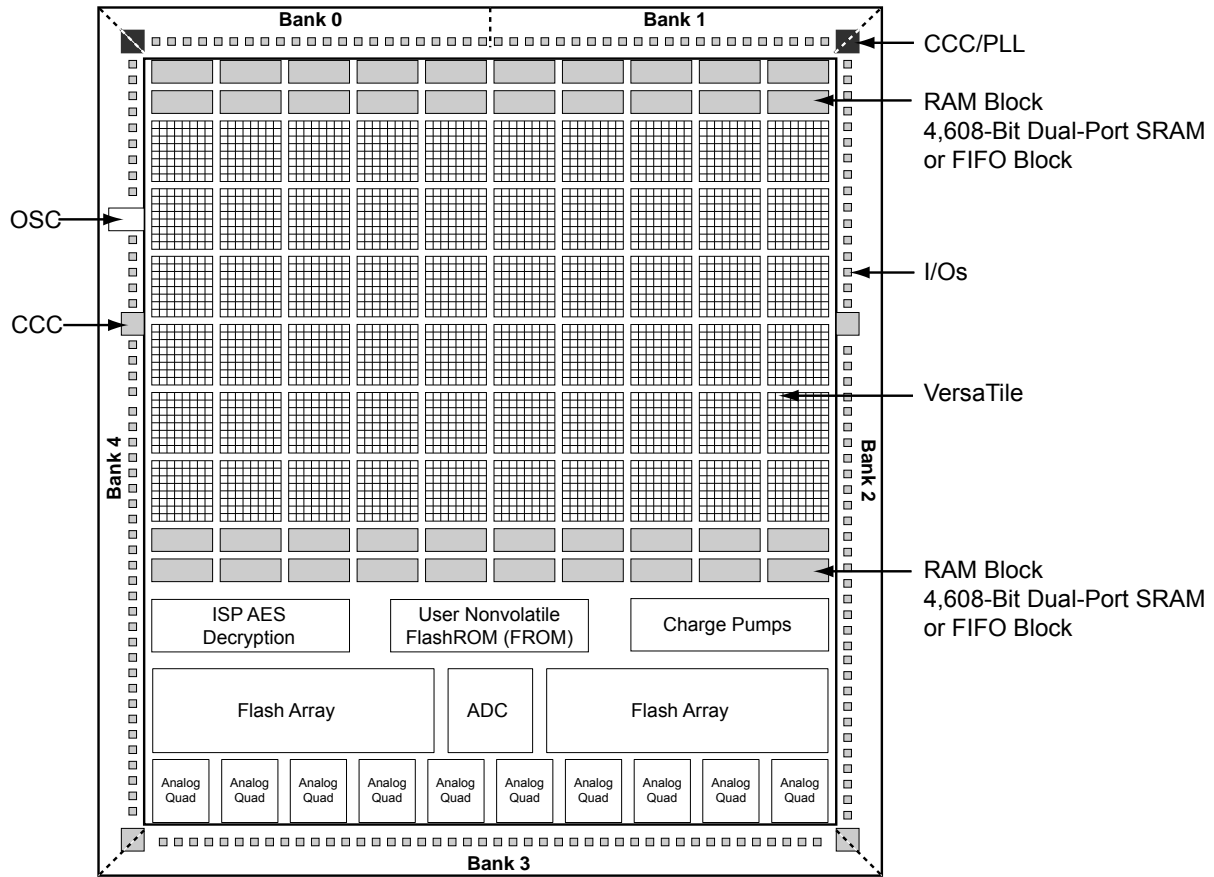


Figure 6-2 • Fusion Device Architecture Overview (AFS600)

Conclusion

Fusion, IGLOO, and ProASIC3 devices provide users with extremely flexible SRAM blocks for most design needs, with the ability to choose between an easy-to-use dual-port memory or a wide-word two-port memory. Used with the built-in FIFO controllers, these memory blocks also serve as highly efficient FIFOs that do not consume user gates when implemented. The SmartGen core generator provides a fast and easy way to configure these memory elements for use in designs.

List of Changes

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

Date	Changes	Page
August 2012	The note connected with Figure 6-3 • Supported Basic RAM Macros, regarding RAM4K9, was revised to explain that it applies only to part numbers of certain revisions and earlier (SAR 29574).	152
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
v1.5 (December 2008)	IGLOO nano and ProASIC3 nano devices were added to Table 6-1 • Flash-Based FPGAs.	150
	IGLOO nano and ProASIC3 nano devices were added to Figure 6-8 • Interfacing TAP Ports and SRAM Blocks.	164
v1.4 (October 2008)	The "SRAM/FIFO Support in Flash-Based Devices" section was revised to include new families and make the information more concise.	150
	The "SRAM and FIFO Architecture" section was modified to remove "IGLOO and ProASIC3E" from the description of what the memory block includes, as this statement applies to all memory blocks.	151
	Wording in the "Clocking" section was revised to change "IGLOO and ProASIC3 devices support inversion" to "Low power flash devices support inversion." The reference to IGLOO and ProASIC3 development tools in the last paragraph of the section was changed to refer to development tools in general.	157
	The "ESTOP and FSTOP Usage" section was updated to refer to FIFO counters in devices in general rather than only IGLOO and ProASIC3E devices.	160
v1.3 (August 2008)	The note was removed from Figure 6-7 • RAM Block with Embedded FIFO Controller and placed in the WCLK and RCLK description.	158
	The "WCLK and RCLK" description was revised.	159
v1.2 (June 2008)	The following changes were made to the family descriptions in Table 6-1 • Flash-Based FPGAs: <ul style="list-style-type: none"> ProASIC3L was updated to include 1.5 V. The number of PLLs for ProASIC3E was changed from five to six. 	150
v1.1 (March 2008)	The "Introduction" section was updated to include the IGLOO PLUS family.	147
	The "Device Architecture" section was updated to state that 15 k gate devices do not support SRAM and FIFO.	147
	The first note in Figure 6-1 • IGLOO and ProASIC3 Device Architecture Overview was updated to include mention of 15 k gate devices, and IGLOO PLUS was added to the second note.	149

I/O Banks and I/O Standards Compatibility

I/Os are grouped into I/O voltage banks.

Each I/O voltage bank has dedicated I/O supply and ground voltages (VMV/GNDQ for input buffers and V_{CCI} /GND for output buffers). Because of these dedicated supplies, only I/Os with compatible standards can be assigned to the same I/O voltage bank. Table 8-3 on page 217 shows the required voltage compatibility values for each of these voltages.

There are eight I/O banks (two per side).

Every I/O bank is divided into minibanks. Any user I/O in a VREF minibank (a minibank is the region of scope of a VREF pin) can be configured as a VREF pin (Figure 8-2). Only one V_{REF} pin is needed to control the entire V_{REF} minibank. The location and scope of the V_{REF} minibanks can be determined by the I/O name. For details, see the user I/O naming conventions for "IGLOOe and ProASIC3E" on page 245. Table 8-5 on page 217 shows the I/O standards supported by IGLOOe and ProASIC3E devices, and the corresponding voltage levels.

I/O standards are compatible if they comply with the following:

- Their V_{CCI} and VMV values are identical.
- Both of the standards need a VREF, and their VREF values are identical.
- All inputs and disabled outputs are voltage tolerant up to 3.3 V.

For more information about I/O and global assignments to I/O banks in a device, refer to the specific pin table for the device in the packaging section of the datasheet, and see the user I/O naming conventions for "IGLOOe and ProASIC3E" on page 245.

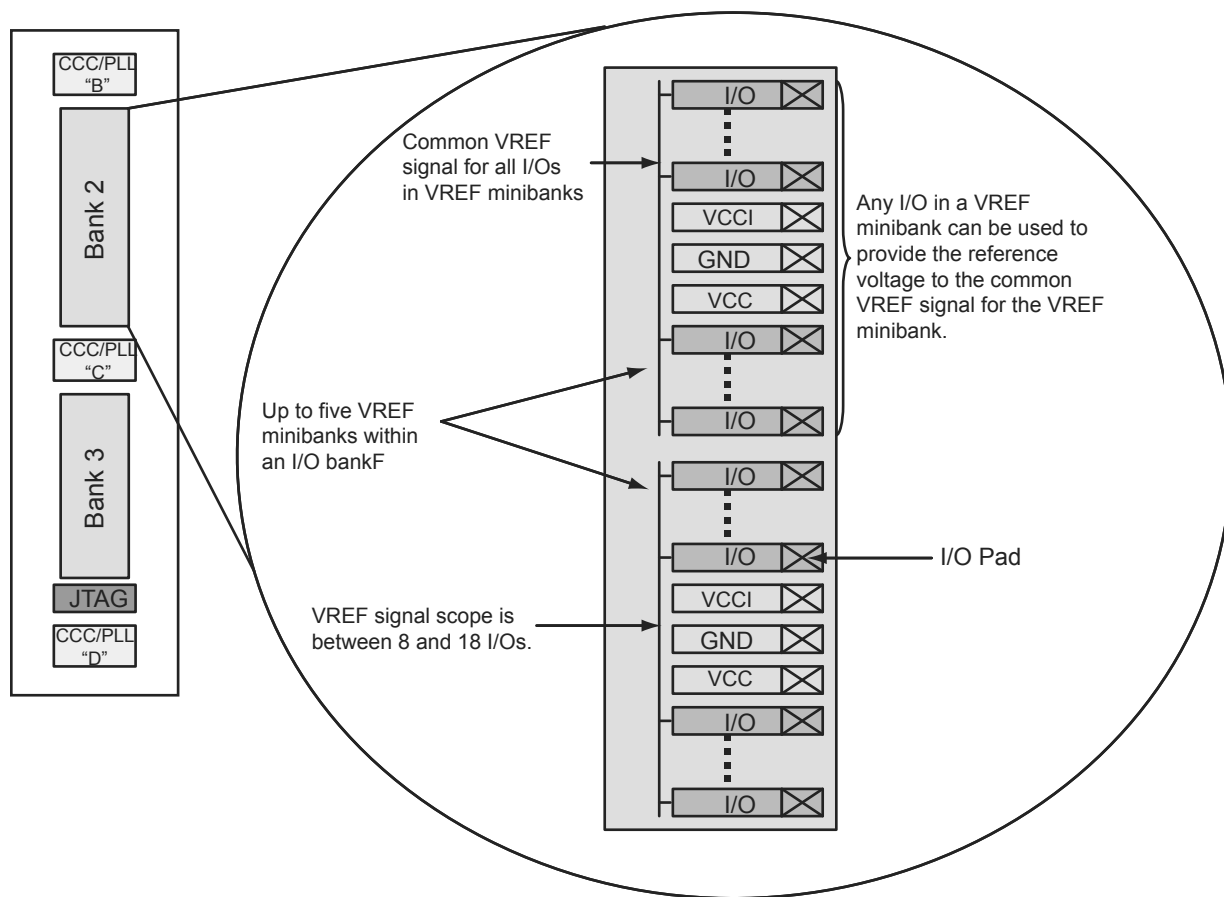


Figure 8-2 • Typical IGLOOe and ProASIC3E I/O Bank Detail Showing V_{REF} Minibanks

Power-Up Behavior

Low power flash devices are power-up/-down friendly; i.e., no particular sequencing is required for power-up and power-down. This eliminates extra board components for power-up sequencing, such as a power-up sequencer.

During power-up, all I/Os are tristated, irrespective of I/O macro type (input buffers, output buffers, I/O buffers with weak pull-ups or weak pull-downs, etc.). Once I/Os become activated, they are set to the user-selected I/O macros. Refer to the "Power-Up/-Down Behavior of Low Power Flash Devices" section on page 373 for details.

Drive Strength

Low power flash devices have up to seven programmable output drive strengths. The user can select the drive strength of a particular output in the I/O Attribute Editor or can instantiate a specialized I/O macro, such as OUTBUF_S_12 (slew = low, out_drive = 12 mA).

The maximum available drive strength is 24 mA per I/O. Though no I/O should be forced to source or sink more than 24 mA indefinitely, I/Os may handle a higher amount of current (refer to the device IBIS model for maximum source/sink current) during signal transition (AC current). Every device package has its own power dissipation limit; hence, power calculation must be performed accurately to determine how much current can be tolerated per I/O within that limit.

I/O Interfacing

Low power flash devices are 5 V–input– and 5 V–output–tolerant if certain I/O standards are selected (refer to the "5 V Input and Output Tolerance" section on page 232). Along with other low-voltage I/O macros, this 5 V tolerance makes these devices suitable for many types of board component interfacing.

Table 8-19 shows some high-level interfacing examples using low power flash devices.

Table 8-19 • 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.

Flash FPGAs I/O Support

The flash FPGAs listed in Table 9-1 support I/Os and the functions described in this document.

Table 9-1 • 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 ProASIC3 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 9-1. 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 9-1. 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*.

The procedure is as follows:

1. Select the bank to which you want VCCI to be assigned from the **Choose Bank** list.
2. Select the I/O standards for that bank. If you select any standard, the tool will automatically show all compatible standards that have a common VCCI voltage requirement.
3. Click **Apply**.
4. Repeat steps 1–3 to assign VCCI voltages to other banks. Refer to Figure 9-11 on page 263 to find out how many I/O banks are needed for VCCI bank assignment.

Manually Assigning VREF Pins

Voltage-referenced inputs require an input reference voltage (VREF). The user must assign VREF pins before running Layout. Before assigning a VREF pin, the user must set a VREF technology for the bank to which the pin belongs.

VREF Rules for the Implementation of Voltage-Referenced I/O Standards

The VREF rules are as follows:

1. Any I/O (except JTAG I/Os) can be used as a V_{REF} pin.
2. One V_{REF} pin can support up to 15 I/Os. It is recommended, but not required, that eight of them be on one side and seven on the other side (in other words, all 15 can still be on one side of VREF).
3. SSTL3 (I) and (II): Up to 40 I/Os per north or south bank in any position
4. LVPECL / GTL+ 3.3 V / GTL 3.3 V: Up to 48 I/Os per north or south bank in any position (not applicable for IGLOO nano and ProASIC3 nano devices)
5. SSTL2 (I) and (II) / GTL+ 2.5 V / GTL 2.5 V: Up to 72 I/Os per north or south bank in any position
6. VREF minibanks partition rule: Each I/O bank is physically partitioned into VREF minibanks. The VREF pins within a VREF minibank are interconnected internally, and consequently, only one VREF voltage can be used within each VREF minibank. If a bank does not require a VREF signal, the VREF pins of that bank are available as user I/Os.
7. The first VREF minibank includes all I/Os starting from one end of the bank to the first power triple and eight more I/Os after the power triple. Therefore, the first VREF minibank may contain (0 + 8), (2 + 8), (4 + 8), (6 + 8), or (8 + 8) I/Os.

The second VREF minibank is adjacent to the first VREF minibank and contains eight I/Os, a power triple, and eight more I/Os after the triple. An analogous rule applies to all other VREF minibanks but the last.

The last VREF minibank is adjacent to the previous one but contains eight I/Os, a power triple, and all I/Os left at the end of the bank. This bank may also contain (8 + 0), (8 + 2), (8 + 4), (8 + 6), or (8 + 8) available I/Os.

Example:

4 I/Os → Triple → 8 I/Os, 8 I/Os → Triple → 8 I/Os, 8 I/Os → Triple → 2 I/Os

That is, minibank A = (4 + 8) I/Os, minibank B = (8 + 8) I/Os, minibank C = (8 + 2) I/Os.

8. Only minibanks that contain input or bidirectional I/Os require a VREF. A VREF is not needed for minibanks composed of output or tristated I/Os.

Assigning the VREF Voltage to a Bank

When importing the PDC file, the VREF voltage can be assigned to the I/O bank. The PDC command is as follows:

```
set_iobank -vref [value]
```

Another method for assigning VREF is by using **MVN > Edit > I/O Bank Settings** (Figure 9-13 on page 266).

4. Right-click and then choose **Highlight VREF range**. All the pins covered by that VREF pin will be highlighted (Figure 9-14).
-

Figure 9-14 • VREF Range

Using PinEditor or ChipPlanner, VREF pins can also be assigned (Figure 9-15).

Figure 9-15 • Assigning VREF from PinEditor

To unassign a VREF pin:

1. Select the pin to unassign.
2. Right-click and choose **Use Pin for VREF**. The check mark next to the command disappears. The VREF pin is now a regular pin.

Resetting the pin may result in unassigning I/O cores, even if they are locked. In this case, a warning message appears so you can cancel the operation.

After you assign the VREF pins, right-click a VREF pin and choose **Highlight VREF Range** to see how many I/Os are covered by that pin. To unhighlight the range, choose **Unhighlight All** from the **Edit** menu.

Signal Integrity While Using ISP

For ISP of flash devices, customers are expected to follow the board-level guidelines provided on the Microsemi SoC Products Group website. These guidelines are discussed in the datasheets and application notes (refer to the “Related Documents” section of the datasheet for application note links). Customers are also expected to troubleshoot board-level signal integrity issues by measuring voltages and taking oscilloscope plots.

Programming Failure Allowances

Microsemi has strict policies regarding programming failure allowances. Please refer to *Programming and Functional Failure Guidelines* on the Microsemi SoC Products Group website for details.

Contacting the Customer Support Group

Highly skilled engineers staff the Customer Applications Center from 7:00 A.M. to 6:00 P.M., Pacific time, Monday through Friday. You can contact the center by one of the following methods:

Electronic Mail

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. Microsemi monitors the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and contact information for efficient processing of your request. The technical support email address is soc_tech@microsemi.com.

Telephone

Our Technical Support Hotline answers all calls. The center retrieves information, such as your name, company name, telephone number, and question. Once this is done, a case number is assigned. Then the center forwards the information to a queue where the first available applications engineer receives the data and returns your call. The phone hours are from 7:00 A.M. to 6:00 P.M., Pacific time, Monday through Friday.

The Customer Applications Center number is (800) 262-1060.

European customers can call +44 (0) 1256 305 600.

ISP Programming Header Information

The FlashPro4/3/3X programming cable connector can be connected with a 10-pin, 0.1"-pitch programming header. The recommended programming headers are manufactured by AMP (103310-1) and 3M (2510-6002UB). If you have limited board space, you can use a compact programming header manufactured by Samtec (FTSH-105-01-L-D-K). Using this compact programming header, you are required to order an additional header adapter manufactured by Microsemi SoC Products Group (FP3-10PIN-ADAPTER-KIT).

Existing ProASIC^{PLUS} family customers who are using the Samtec Small Programming Header (FTSH-113-01-L-D-K) and are planning to migrate to IGLOO or ProASIC3 devices can also use FP3-10PIN-ADAPTER-KIT.

Table 13-3 • Programming Header Ordering Codes

Manufacturer	Part Number	Description
AMP	103310-1	10-pin, 0.1"-pitch cable header (right-angle PCB mount angle)
3M	2510-6002UB	10-pin, 0.1"-pitch cable header (straight PCB mount angle)
Samtec	FTSH-113-01-L-D-K	Small programming header supported by FlashPro and Silicon Sculptor
Samtec	FTSH-105-01-L-D-K	Compact programming header
Samtec	FFSD-05-D-06.00-01-N	10-pin cable with 50 mil pitch sockets; included in FP3-10PIN-ADAPTER-KIT.
Microsemi	FP3-10PIN-ADAPTER-KIT	Transition adapter kit to allow FP3 to be connected to a micro 10-pin header (50 mil pitch). Includes a 6 inch Samtec FFSD-05-D-06.00-01-N cable in the kit. The transition adapter board was previously offered as FP3-26PIN-ADAPTER and includes a 26-pin adapter for design transitions from ProASIC ^{PLUS} based boards to ProASIC3 based boards.

TCK	1	2	GND
TDO	3	4	NC (FlashPro3/3X); Prog_Mode* (FlashPro4)
TMS	5	6	VJTAG
VPUMP	7	8	TRST
TDI	9	10	GND

*Note: *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 13-5 • Programming Header (top view)

Microsemi's Flash Families Support Voltage Switching Circuit

The flash FPGAs listed in Table 14-1 support the voltage switching circuit feature and the functions described in this document.

Table 14-1 • Flash-Based FPGAs Supporting Voltage Switching Circuit

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	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

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 14-1. 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 14-1. 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*.

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
v1.1 (October 2008)	The "Introduction" was revised to include information about the core supply voltage range of operation in V2 devices.	341
	IGLOO nano device support was added to Table 14-1 • Flash-Based FPGAs Supporting Voltage Switching Circuit.	342
	The "Circuit Description" section was updated to include IGLOO PLUS core operation from 1.2 V to 1.5 V in 50 mV increments.	343
v1.0 (August 2008)	The "Microsemi's Flash Families Support Voltage Switching Circuit" section was revised to include new families and make the information more concise.	342

15 – Microprocessor Programming of Microsemi's Low Power Flash Devices

Introduction

The Fusion, IGLOO, and ProASIC3 families of flash FPGAs support in-system programming (ISP) with the use of a microprocessor. Flash-based FPGAs store their configuration information in the actual cells within the FPGA fabric. SRAM-based devices need an external configuration memory, and hybrid nonvolatile devices store the configuration in a flash memory inside the same package as the SRAM FPGA. Since the programming of a true flash FPGA is simpler, requiring only one stage, it makes sense that programming with a microprocessor in-system should be simpler than with other SRAM FPGAs. This reduces bill-of-materials costs and printed circuit board (PCB) area, and increases system reliability.

Nonvolatile flash technology also gives the low power flash devices the advantage of a secure, low power, live-at-power-up, and single-chip solution. Low power flash devices are reprogrammable and offer time-to-market benefits at an ASIC-level unit cost. These features enable engineers to create high-density systems using existing ASIC or FPGA design flows and tools.

This document is an introduction to microprocessor programming only. To explain the difference between the options available, user's guides for DirectC and STAPL provide more detail on implementing each style.

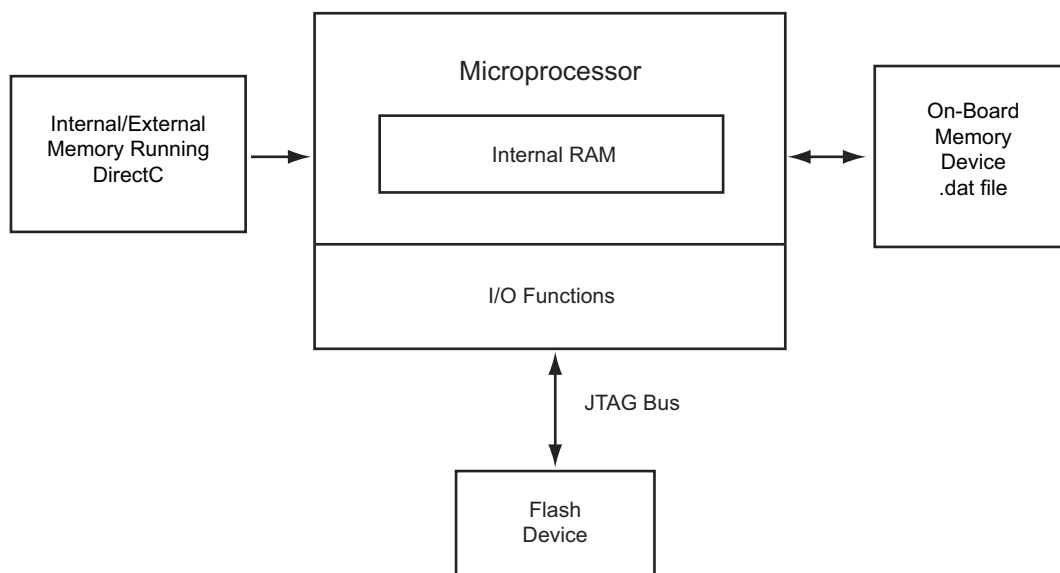


Figure 15-1 • ISP Using Microprocessor

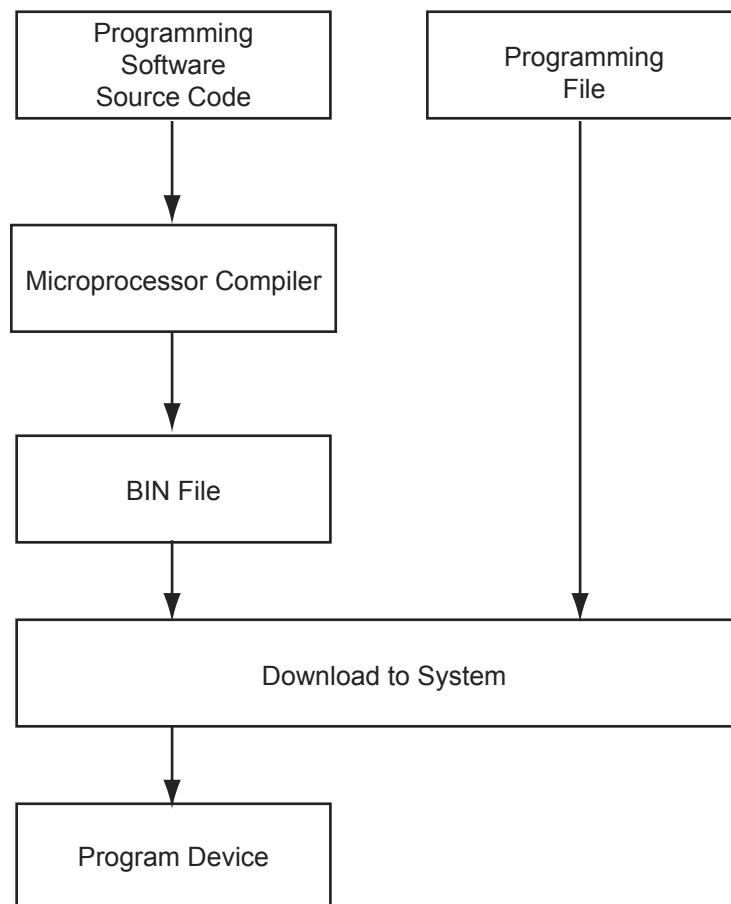


Figure 15-3 • MCU FPGA Programming Model

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.