



Welcome to [E-XFL.COM](https://www.e-xfl.com)

### Understanding Embedded - FPGAs (Field Programmable Gate Array)

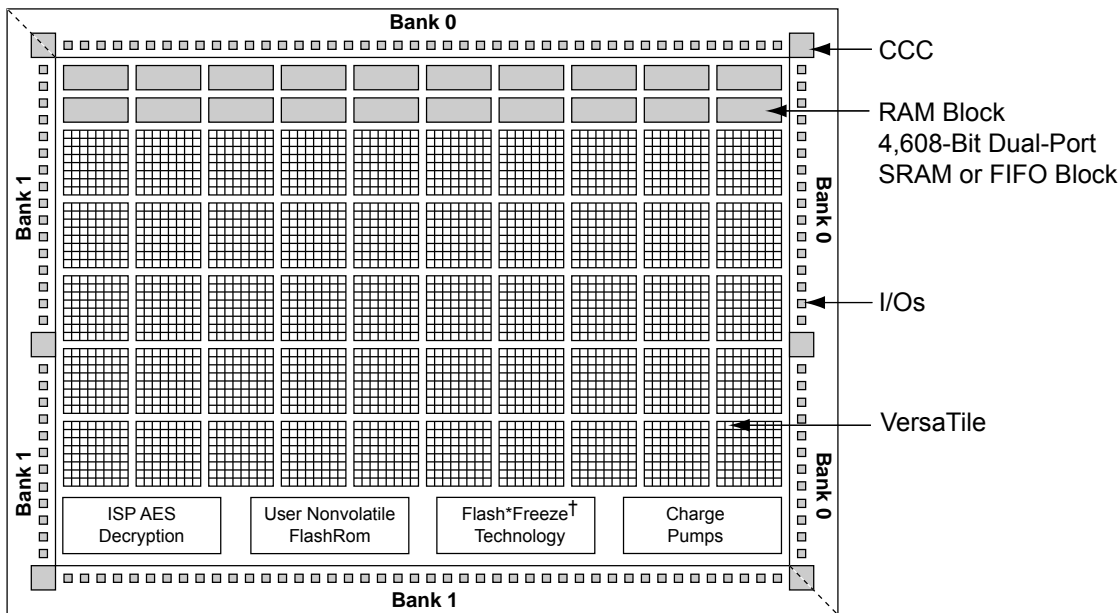
Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling 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.

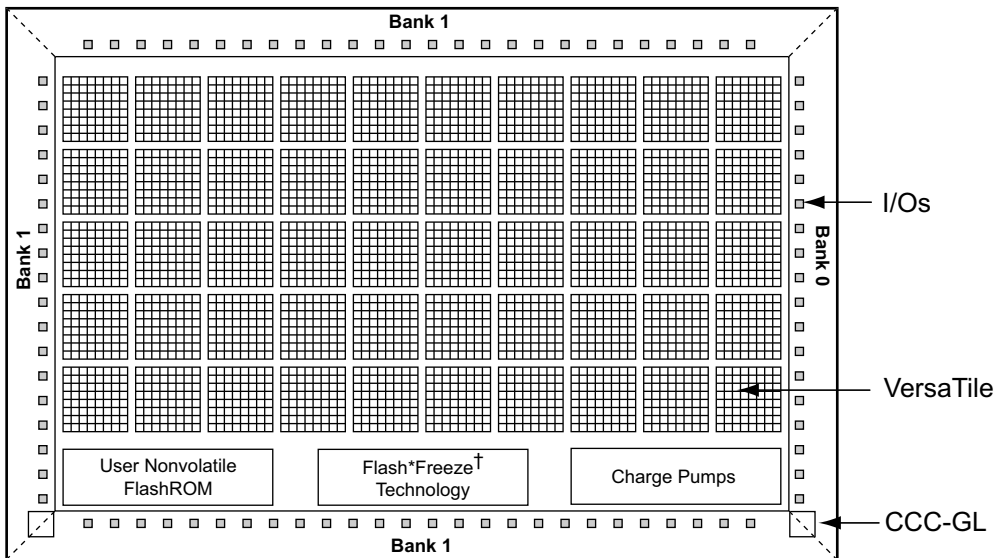
#### Details

Product Status	Active
Number of LABs/CLBs	-
Number of Logic Elements/Cells	-
Total RAM Bits	516096
Number of I/O	147
Number of Gates	3000000
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	<a href="https://www.e-xfl.com/product-detail/microchip-technology/a3pe3000l-pqg208">https://www.e-xfl.com/product-detail/microchip-technology/a3pe3000l-pqg208</a>



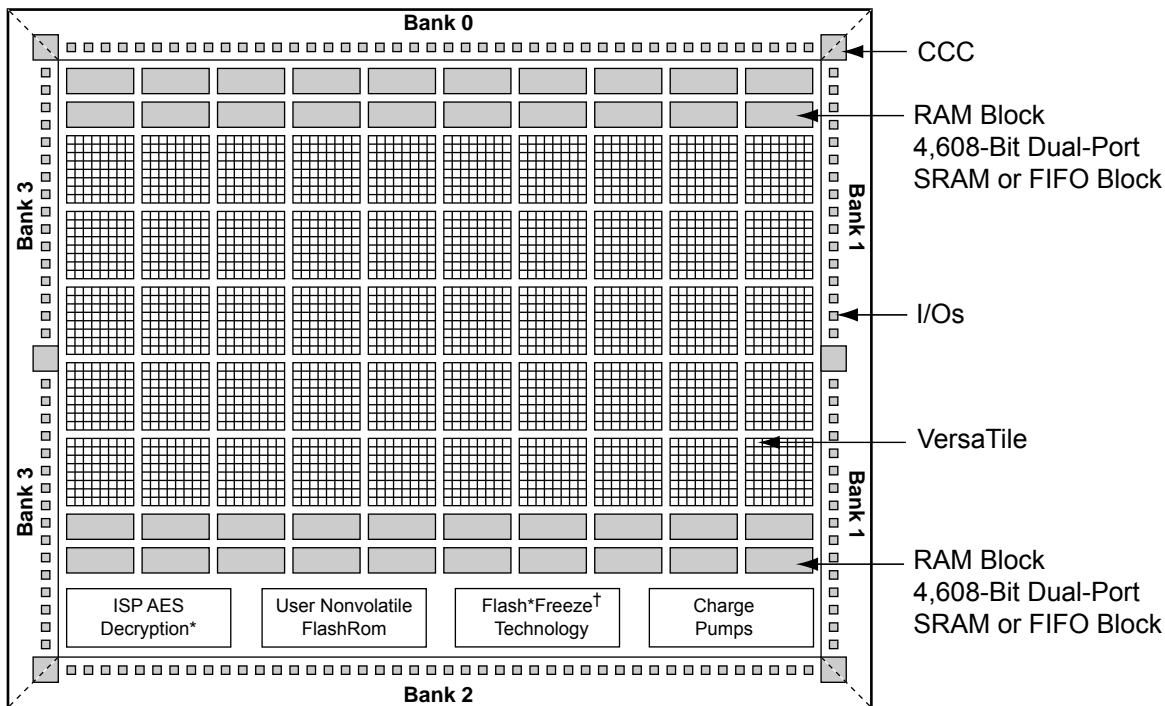
Note: † Flash\*Freeze mode is supported on IGL00 devices.

**Figure 1-3 • IGLOO Device Architecture Overview with Two I/O Banks with RAM and PLL (60 k and 125 k gate densities)**



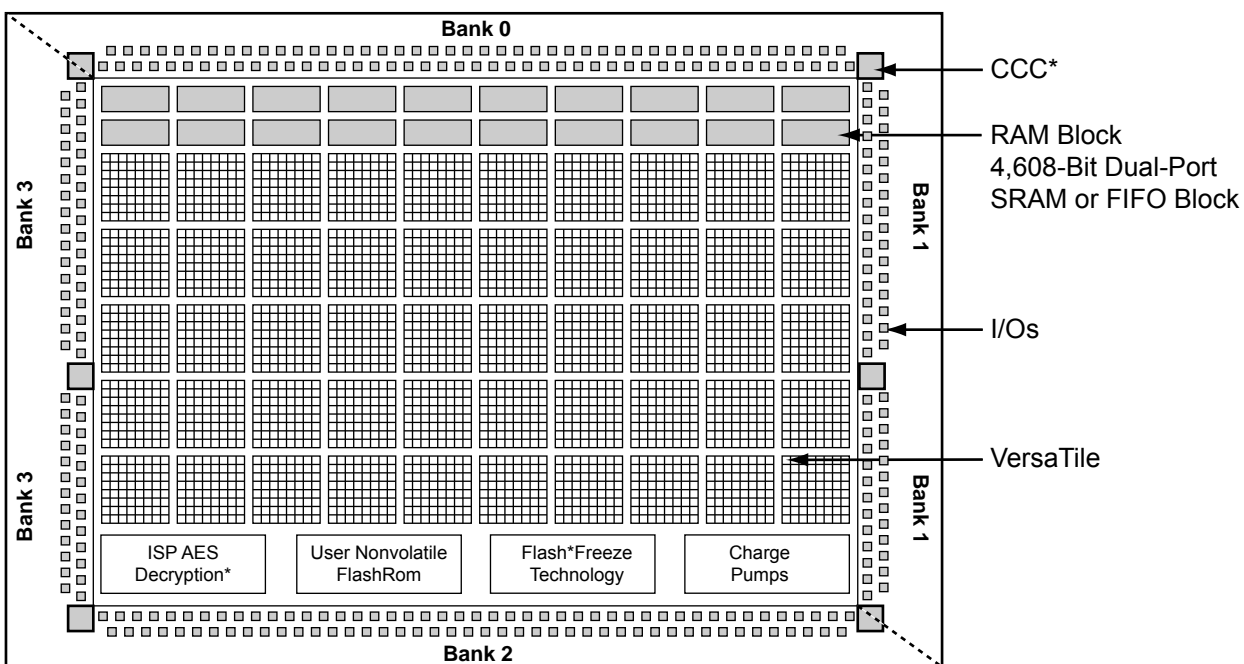
Note: † Flash\*Freeze mode is supported on IGL00 devices.

**Figure 1-4 • IGLOO Device Architecture Overview with Three I/O Banks (AGLN015, AGLN020, A3PN015, and A3PN020)**



*Note: Flash\*Freeze technology only applies to IGL00 and ProASIC3L families.*

**Figure 1-5 • IGLOO, IGLOO nano, ProASIC3 nano, and ProASIC3/L Device Architecture Overview with Four I/O Banks (AGL600 device is shown)**

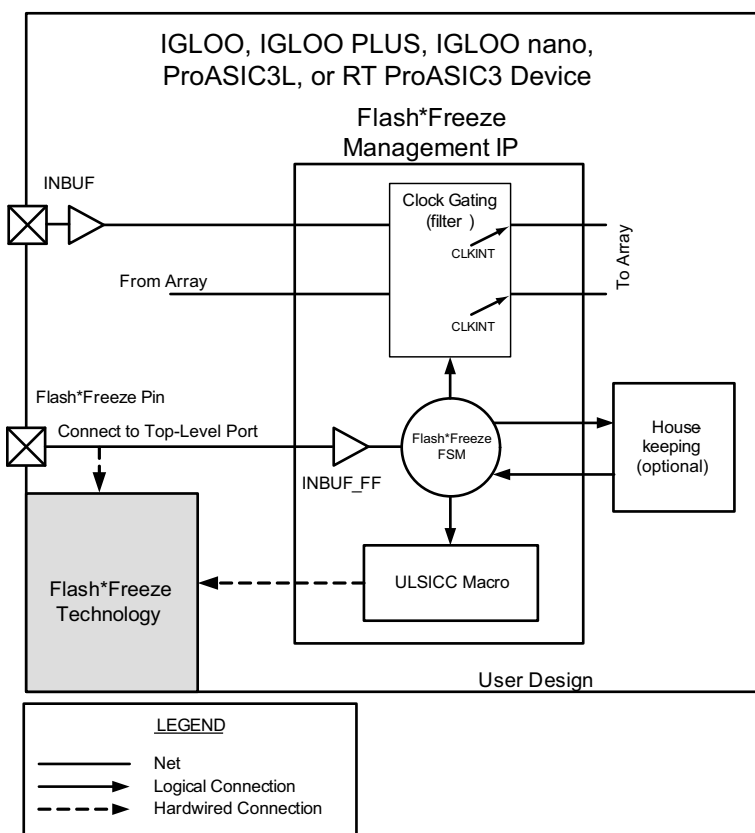


Note: \* AGLP030 does not contain a PLL or support AES security.

**Figure 1-6 • IGL00 PLUS Device Architecture Overview with Four I/O Banks**

Flash\*Freeze management IP. Additional information on this IP core can be found in the Libero online help.

The Flash\*Freeze management IP is comprised of three blocks: the Flash\*Freeze finite state machine (FSM), the clock gating (filter) block, and the ULSICC macro, as shown in Figure 2-10.



**Figure 2-10 • Flash\*Freeze Management IP Block Diagram**

### **Flash\*Freeze Management FSM**

The Flash\*Freeze FSM block is a simple, robust, fully encoded 3-bit state machine that ensures clean entrance to and exit from Flash\*Freeze mode by controlling activities of the clock gating, ULSICC, and optional housekeeping blocks. The state diagram for the FSM is shown in Figure 2-11 on page 38. In normal operation, the state machine waits for Flash\*Freeze pin assertion, and upon detection of a request, it waits for a short period of time to ensure the assertion persists; then it asserts WAIT\_HOUSEKEEPING (active High) synchronous to the user's designated system clock. This flag can be used by user logic to perform any needed shutdown processes prior to entering Flash\*Freeze mode, such as storing data into SRAM, notifying other system components of the request, or timing/validating the Flash\*Freeze request. 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. The Flash\_Freeze\_Enabled signal remains asserted, even during Flash\*Freeze mode, until the Flash\*Freeze pin is deasserted. Use the Flash\_Freeze\_Enabled signal to drive any logic in the design that needs to be in a particular state during Flash\*Freeze mode. The DONE\_HOUSEKEEPING (active High) signal should be asserted to notify the FSM when all the housekeeping tasks are completed. If the user chooses not to use housekeeping, the Flash\*Freeze management IP core generator in Libero SoC will connect WAIT\_HOUSEKEEPING to DONE\_HOUSEKEEPING.



- The INBUF\_FF must be driven by a top-level input port of the design.
- The INBUF\_FF AND the ULSICC macro must be used to enable type 2 Flash\*Freeze mode.
- For type 2 Flash\*Freeze mode, the INBUF\_FF MUST drive some logic in the design.
- For type 1 Flash\*Freeze mode, the INBUF\_FF may drive some logic in the design, but it may also be left floating.
- Only one INBUF\_FF may be instantiated in a device.
- The FF pin threshold voltages are defined by VCCI and the supported single-ended I/O standard in the corresponding I/O bank.
- The FF pin Schmitt trigger option may be configured in the I/O attribute editor in Microsemi's Designer software. The Schmitt trigger option is only available for IGLOOe, IGLOO nano, IGLOO PLUS, ProASIC3EL, and RT ProASIC3 devices.
- A 2 ns glitch filter resides in the Flash\*Freeze Technology block to filter unwanted glitches on the FF pin.

## ULSICC

The User Low Static ICC (ULSICC) macro allows the FPGA core to access the Flash\*Freeze Technology block so that entering and exiting Flash\*Freeze mode can be controlled by the user's design. The ULSICC macro enables a hard block with an available LSICC input port, as shown in Figure 2-3 on page 27 and Figure 2-10 on page 37. Design rules for the ULSICC macro are as follows:

- The ULSICC macro by itself cannot enable Flash\*Freeze mode. The INBUF\_FF AND the ULSICC macro must both be used to enable type 2 Flash\*Freeze mode.
- The ULSICC controls entering the Flash\*Freeze mode by asserting the LSICC input (logic '1') of the ULSICC macro. The FF pin must also be asserted (logic '0') to enter Flash\*Freeze mode.
- When the LSICC signal is '0', the device cannot enter Flash\*Freeze mode; and if already in Flash\*Freeze mode, it will exit.
- When the ULSICC macro is not instantiated in the user's design, the LSICC port will be tied High.

## Flash\*Freeze Management IP

The Flash\*Freeze management IP can be configured with the Libero (or SmartGen) core generator in a simple, intuitive interface. With the core configuration tool, users can select the number of clocks to be gated, and select whether or not to implement housekeeping. All port names on the Flash\*Freeze management IP block can be renamed by the user.

- The clock gating (filter) blocks include CLKINT buffers for each gated clock output (version 8.3).
- When housekeeping is NOT used, the WAIT\_HOUSEKEEPING signal will be automatically fed back into DONE\_HOUSEKEEPING inside the core, and the ports will not be available at the IP core interface.
- The INBUF\_FF macro is automatically instantiated within the IP core.
- The INBUF\_FF port (default name is "Flash\_Freeze\_N") must be connected to a top-level input port of the design.
- The ULSICC macro is automatically instantiated within the IP core, and the LSICC signal is driven by the FSM.
- Timing analysis can be performed on the clock domain of the source clock (i.e., input to the clock gating filters). For example, if CLKIn becomes CLKIn\_gated, the timing can be performed on the CLKIn domain in SmartTime.
- The gated clocks can be added to the clock list if the user wishes to analyze these clocks specifically. The user can locate the gated clocks by looking for instance names such as those below:

```
Top/ff1/ff_1_wrapper_inst/user_ff_1_wrapper/Primary_Filter_Instance/  
Latch_For_Clock_Gating:Q  
Top/ff1/ff_1_wrapper_inst/user_ff_1_wrapper/genblk1.genblk2.secondary_filter[0].  
secondary_filter_instance/Latch_For_Clock_Gating:Q  
Top/ff1/ff_1_wrapper_inst/user_ff_1_wrapper/genblk1.genblk2.secondary_filter[1].  
secondary_filter_instance/Latch_For_Clock_Gating:Q
```

Date	Changes	Page
51900147-2/5.07	In the following sentence, located in the "Flash*Freeze Mode" section, the bold text was changed from active high to active Low. The Flash*Freeze pin ( <b>active low</b> ) is a dedicated pin used to enter or exit Flash*Freeze mode directly, or alternatively the pin can be routed internally to the FPGA core to allow the user's logic to decide if it is safe to transition to this mode.	24
	Figure 2-2 • Flash*Freeze Mode Type 1 – Timing Diagram was updated.	25
	Information about ULSICC was added to the "Prototyping for IGLOO and ProASIC3L Devices Using ProASIC3" section.	2-21
51900147-1/3.07	In the "Flash*Freeze Mode" section, "active high" was changed to "active low."	24
	The "Prototyping for IGLOO and ProASIC3L Devices Using ProASIC3" section was updated with information concerning the Flash*Freeze pin.	2-21

## 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
	The "Global Architecture" section and "VersaNet Global Network Distribution" section were revised for clarity (SARs 20646, 24779).	47, 49
	The "I/O Banks and Global I/Os" section was moved earlier in the document, renamed to "Chip and Quadrant Global I/Os", and revised for clarity. Figure 3-4 • Global Connections Details, Figure 3-6 • Global Inputs, Table 3-2 • Chip Global Pin Name, and Table 3-3 • Quadrant Global Pin Name are new (SARs 20646, 24779).	51
	The "Clock Aggregation Architecture" section was revised (SARs 20646, 24779).	57
	Figure 3-7 • Chip Global Aggregation was revised (SARs 20646, 24779).	59
	The "Global Macro and Placement Selections" section is new (SARs 20646, 24779).	64
v1.4 (December 2008)	The "Global Architecture" section was updated to include 10 k devices, and to include information about VersaNet global support for IGLOO nano devices.	47
	The Table 3-1 • Flash-Based FPGAs was updated to include IGLOO nano and ProASIC3 nano devices.	48
	The "VersaNet Global Network Distribution" section was updated to include 10 k devices and to note an exception in global lines for nano devices.	49
	Figure 3-2 • Simplified VersaNet Global Network (30 k gates and below) is new.	50
	The "Spine Architecture" section was updated to clarify support for 10 k and nano devices.	57
	Table 3-4 • Globals/Spines/Rows for IGLOO and ProASIC3 Devices was updated to include IGLOO nano and ProASIC3 nano devices.	57
	The figure in the CLKBUF_LVDS/LVPECL row of Table 3-8 • Clock Macros was updated to change CLKBIBUF to CLKBUF.	62
v1.3 (October 2008)	A third bullet was added to the beginning of the "Global Architecture" section: In Fusion devices, the west CCC also contains a PLL core. In the two larger devices (AFS600 and AFS1500), the west and east CCCs each contain a PLL.	47
	The "Global Resource Support in Flash-Based Devices" section was revised to include new families and make the information more concise.	48
	Table 3-4 • Globals/Spines/Rows for IGLOO and ProASIC3 Devices was updated to include A3PE600/L in the device column.	57
	Table note 1 was revised in Table 3-9 • I/O Standards within CLKBUF to include AFS600 and AFS1500.	63
v1.2 (June 2008)	The following changes were made to the family descriptions in Table 3-1 • Flash-Based FPGAs: <ul style="list-style-type: none"> <li>ProASIC3L was updated to include 1.5 V.</li> <li>The number of PLLs for ProASIC3E was changed from five to six.</li> </ul>	48

Date	Changes	Page
v1.1 (March 2008)	The "Global Architecture" section was updated to include the IGLOO PLUS family. The bullet was revised to include that the west CCC does not contain a PLL core in 15 k and 30 k devices. Instances of "A3P030 and AGL030 devices" were replaced with "15 k and 30 k gate devices."	47
v1.1 (continued)	Table 3-1 • Flash-Based FPGAs and the accompanying text was updated to include the IGLOO PLUS family. The "IGLOO Terminology" section and "ProASIC3 Terminology" section are new.	48
	The "VersaNet Global Network Distribution" section, "Spine Architecture" section, the note in Figure 3-1 • Overview of VersaNet Global Network and Device Architecture, and the note in Figure 3-3 • Simplified VersaNet Global Network (60 k gates and above) were updated to include mention of 15 k gate devices.	49, 50
	Table 3-4 • Globals/Spines/Rows for IGLOO and ProASIC3 Devices was updated to add the A3P015 device, and to revise the values for clock trees, globals/spines per tree, and globals/spines per device for the A3P030 and AGL030 devices.	57
	Table 3-5 • Globals/Spines/Rows for IGLOO PLUS Devices is new.	58
	CLKBUF_LVCMOS12 was added to Table 3-9 • I/O Standards within CLKBUF.	63
	The "User's Guides" section was updated to include the three different I/O Structures chapters for ProASIC3 and IGLOO device families.	74
v1.0 (January 2008)	Figure 3-3 • Simplified VersaNet Global Network (60 k gates and above) was updated.	50
	The "Naming of Global I/Os" section was updated.	51
	The "Using Global Macros in Synplicity" section was updated.	66
	The "Global Promotion and Demotion Using PDC" section was updated.	67
	The "Designer Flow for Global Assignment" section was updated.	69
	The "Simple Design Example" section was updated.	71
51900087-0/1.05 (January 2005)	Table 3-4 • Globals/Spines/Rows for IGLOO and ProASIC3 Devices was updated.	57

## Global Buffers with No Programmable Delays

Access to the global / quadrant global networks can be configured directly from the global I/O buffer, bypassing the CCC functional block (as indicated by the dotted lines in Figure 4-1 on page 77). Internal signals driven by the FPGA core can use the global / quadrant global networks by connecting via the routed clock input of the multiplexer tree.

There are many specific CLKBUF macros supporting the wide variety of single-ended I/O inputs (CLKBUF) and differential I/O standards (CLKBUF\_LVDS/LVPECL) in the low power flash families. They are used when connecting global I/Os directly to the global/quadrant networks.

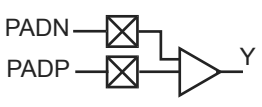
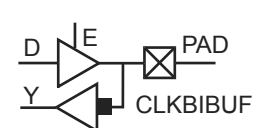
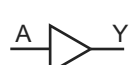

Note: IGLOO nano and ProASIC nano devices do not support differential inputs.

When an internal signal needs to be connected to the global/quadrant network, the CLKINT macro is used to connect the signal to the routed clock input of the network's MUX tree.

To utilize direct connection from global I/Os or from internal signals to the global/quadrant networks, CLKBUF, CLKBUF\_LVPECL/LVDS, and CLKINT macros are used (Figure 4-2).

- The CLKBUF and CLKBUF\_LVPECL/LVDS<sup>1</sup> macros are composite macros that include an I/O macro driving a global buffer, which uses a hardwired connection.
- The CLKBUF, CLKBUF\_LVPECL/LVDS<sup>1</sup> and CLKINT macros are pass-through clock sources and do not use the PLL or provide any programmable delay functionality.
- The CLKINT macro provides a global buffer function driven internally by the FPGA core.

The available CLKBUF macros are described in the *IGLOO, ProASIC3, SmartFusion, and Fusion Macro Library Guide*.

Clock Source		Clock Conditioning	Output
<p>CLKBUF_LVDS/LVPECL Macro</p> 	<p>CLKBIBUF Macro</p> 	None	GLA, GLB, or GLC
<p>CLKINT Macro</p> 	<p>CLKBUF Macro</p> 		

Note: IGLOO nano and ProASIC nano devices do not support differential inputs.

**Figure 4-2 • CCC Options: Global Buffers with No Programmable Delay**

## Global Buffer with Programmable Delay

Clocks requiring clock adjustments can utilize the programmable delay cores before connecting to the global / quadrant global networks. A maximum of 18 CCC global buffers can be instantiated in a device—three per CCC and up to six CCCs per device.

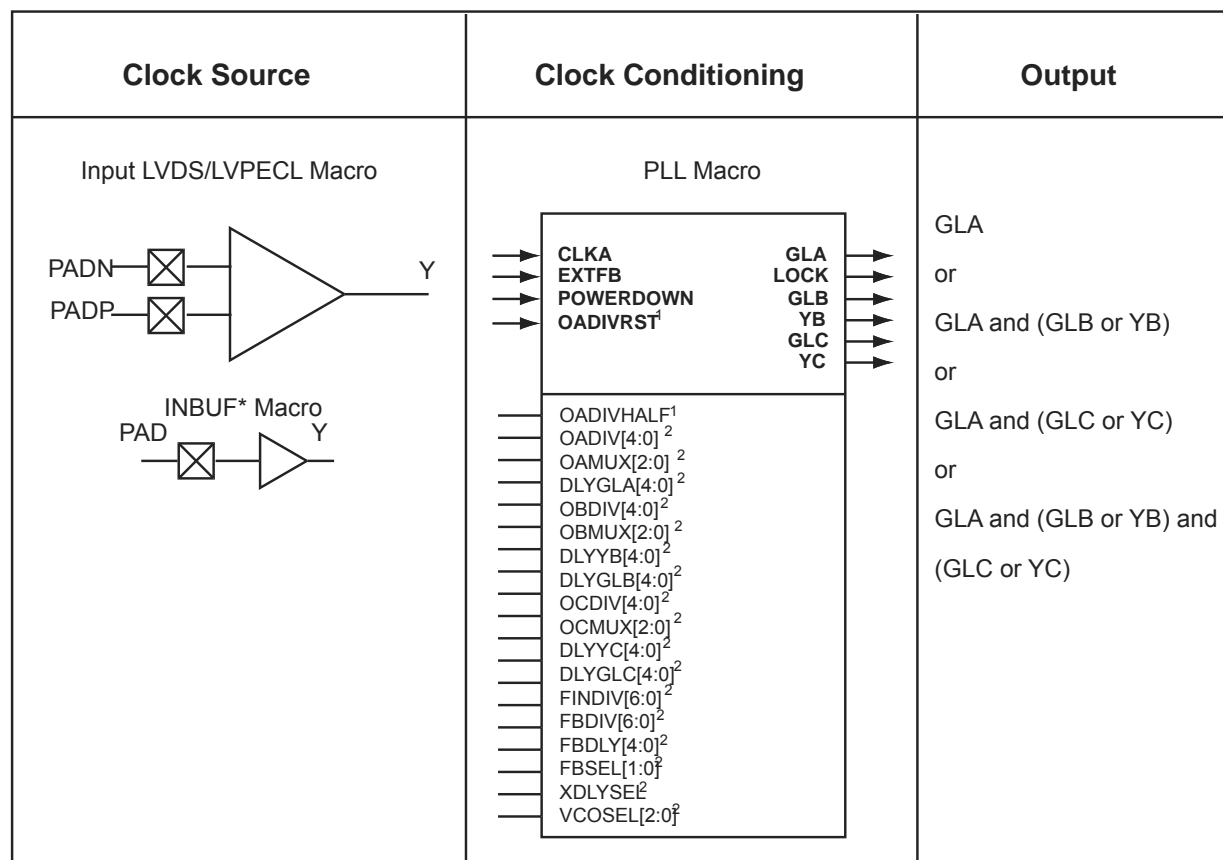
Each CCC functional block contains a programmable delay element for each of the global networks (up to three), and users can utilize these features by using the corresponding macro (Figure 4-3 on page 81).

1. B-LVDS and M-LVDS are supported with the LVDS macro.

## Global Buffers with PLL Function

Clocks requiring frequency synthesis or clock adjustments can utilize the PLL core before connecting to the global / quadrant global networks. A maximum of 18 CCC global buffers can be instantiated in a device—three per CCC and up to six CCCs per device. Each PLL core can generate up to three global/quadrant clocks, while a clock delay element provides one.

The PLL functionality of the clock conditioning block is supported by the PLL macro.



Notes:

1. For Fusion only.
2. Refer to the IGLOO, ProASIC3, SmartFusion, and Fusion Macro Library Guide for more information.
3. For INBUF\* driving a PLL macro or CLKDLY macro, the I/O will be hard-routed to the CCC; i.e., will be placed by software to a dedicated Global I/O.
4. IGLOO nano and ProASIC3 nano devices do not support differential inputs.

**Figure 4-4 • CCC Options: Global Buffers with PLL**

The PLL macro provides five derived clocks (three independent) from a single reference clock. The PLL macro also provides power-down input and lock output signals. The additional inputs shown on the macro are configuration settings, which are configured through the use of SmartGen. For manual setting of these bits refer to the *IGLOO, ProASIC3, SmartFusion, and Fusion Macro Library Guide* for details.

Figure 4-6 on page 87 illustrates the various clock output options and delay elements.

## Dynamic PLL Configuration

To generate a dynamically reconfigurable CCC, the user should select **Dynamic CCC** in the configuration section of the SmartGen GUI (Figure 4-26). This will generate both the CCC core and the configuration shift register / control bit MUX.

---

---

**Figure 4-26 • SmartGen GUI**

Even if dynamic configuration is selected in SmartGen, the user must still specify the static configuration data for the CCC (Figure 4-27). The specified static configuration is used whenever the MODE signal is set to LOW and the CCC is required to function in the static mode. The static configuration data can be used as the default behavior of the CCC where required.

---

---

**Figure 4-27 • Dynamic CCC Configuration in SmartGen**

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

### Solution 1

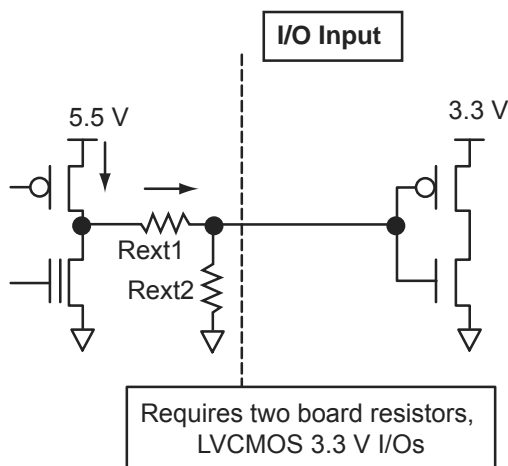


Figure 7-9 • Solution 1

### 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  $3.3\text{ V} + 0.7\text{ V} = 4\text{ V}$ .

### Solution 2

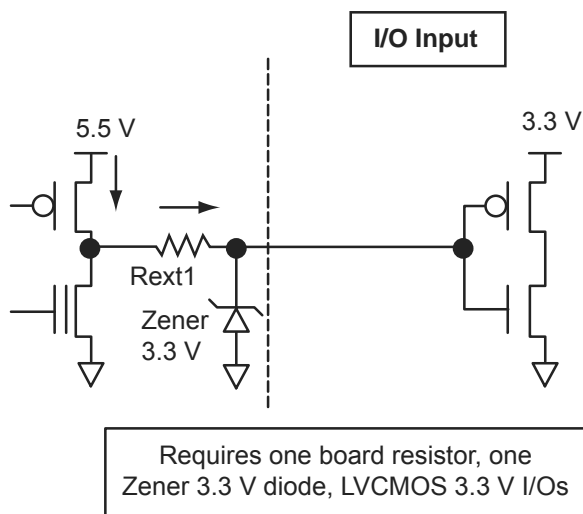


Figure 7-10 • Solution 2



## Related Documents

### Application Notes

*Board-Level Considerations*

[http://www.microsemi.com/soc/documents/ALL\\_AC276\\_AN.pdf](http://www.microsemi.com/soc/documents/ALL_AC276_AN.pdf)

### User's Guides

*Libero SoC User's Guide*

[http://www.microsemi.com/soc/documents/libero\\_ug.pdf](http://www.microsemi.com/soc/documents/libero_ug.pdf)

*IGLOO, Fusion, and ProASIC3 Macro Library Guide*

[http://www.microsemi.com/soc/documents/pa3\\_libguide\\_ug.pdf](http://www.microsemi.com/soc/documents/pa3_libguide_ug.pdf)

*SmartGen Core Reference Guide*

[http://www.microsemi.com/soc/documents/genguide\\_ug.pdf](http://www.microsemi.com/soc/documents/genguide_ug.pdf)

## List of Changes

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

Date	Change	Page
August 2012	Figure 7-1 • DDR Configured I/O Block Logical Representation and Figure 7-2 • DDR Configured I/O Block Logical Representation were revised to indicate that resets on registers 1, 3, 4, and 5 are active high rather than active low. The title of the figures was revised from "I/O Block Logical Representation" (SAR 38215).	175, 181
	AGL015 and A3P015 were added to Table 7-2 • Supported I/O Standards. 1.2 V was added under single-ended I/O standards. LVCMOS 1.2 was added to Table 7-3 • VCCI Voltages and Compatible IGLOO and ProASIC3 Standards (SAR 38096).	177
	Figure 7-4 • Simplified I/O Buffer Circuitry and Table 7-7 • Programmable I/O Features (user control via I/O Attribute Editor) were modified to indicate that programmable input delay control is applicable only to ProASIC3EL and RT ProASIC3 devices (SAR 39666).	183, 188
	The following sentence is incorrect and was removed from the "LVCMOS (Low-Voltage CMOS)" section (SAR 40191): LVCMOS 2.5 V for the 30 k gate devices has a clamp diode to VCCI, but for all other devices there is no clamp diode.	184
	The hyperlink for the <i>Board-Level Considerations</i> application note was corrected (SAR 36663).	208, 210
June 2011	Figure 7-1 • DDR Configured I/O Block Logical Representation and Figure 7-2 • DDR Configured I/O Block Logical Representation were revised so that the I/O_CLR and I/O_OCLK nets are no longer joined in front of Input Register 3 but instead on the branch of the CLR/PRE signal (SAR 26052).	175, 181
	Table 7-1 • Flash-Based FPGAs was revised to remove RT ProASIC3 and add Military ProASIC3/EL in its place (SAR 31824, 31825).	176
	The "Advanced I/Os—IGLOO, ProASIC3L, and ProASIC3" section was revised. Formerly it stated, "3.3 V PCI and 3.3 V PCI-X are 5 V-tolerant." This sentence now reads, "3.3 V PCI and 3.3 V PCI-X can be configured to be 5 V-tolerant" (SAR 20983).	177

## Pro I/Os—IGLOOe, ProASIC3EL, and ProASIC3E

Table 8-2 shows the voltages and compatible I/O standards for Pro I/Os. I/Os provide programmable slew rates, drive strengths, and weak pull-up and pull-down circuits. All I/O standards, except 3.3 V PCI and 3.3 V PCI-X, are capable of hot-insertion. 3.3 V PCI and 3.3 V PCI-X can be configured to be 5 V-tolerant. See the "5 V Input Tolerance" section on page 232 for possible implementations of 5 V tolerance. Single-ended input buffers support both the Schmitt trigger and programmable delay options on a per-I/O basis.

All I/Os are in a known state during power-up, and any power-up sequence is allowed without current impact. Refer to the "I/O Power-Up and Supply Voltage Thresholds for Power-On Reset (Commercial and Industrial)" section in the datasheet for more information. During power-up, before reaching activation levels, the I/O input and output buffers are disabled while the weak pull-up is enabled. Activation levels are described in the datasheet.

**Table 8-2 • Supported I/O Standards**

	A3PE600	AGLE600	A3PE1500	A3PE3000/ A3PE3000L	AGLE3000
<b>Single-Ended</b>					
LVTTL/LVCMOS 3.3 V, LVCMOS 2.5 V / 1.8 V / 1.5 V, LVCMOS 2.5/5.0 V, 3.3 V PCI/PCI-X	✓	✓	✓	✓	✓
LVCMOS 1.2 V	–	✓	–	–	✓
<b>Differential</b>					
LVPECL, LVDS, B-LVDS, M-LVDS	✓	✓	✓	✓	✓
<b>Voltage-Referenced</b>					
GTL+ 2.5 V / 3.3 V, GTL 2.5 V / 3.3 V, HSTL Class I and II, SSTL2 Class I and II, SSTL3 Class I and II	✓	✓	✓	✓	✓

## 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.

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

Command	Action	Example	Comment
<b>I/O Attribute Constraint</b>			
set_io	Sets the attributes of an I/O	<pre>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</pre>	<p>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.</p> <p>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.</p> <p>Example: OUTBUF_S_24 (low slew, output drive 24 mA)</p> <p>Slew and output drive cannot be changed.</p>
<b>I/O Region Placement Constraints</b>			
define_region	Defines either a rectangular region or a rectilinear region	<pre>define_region -name [region_name] -type [region_type] x1 y1 x2 y2  define_region -name test -type inclusive 0 15 2 29</pre>	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	<pre>assign_region [region name] [macro_name...]  assign_region test U12</pre>	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 document.

Date	Changes	Page
August 2012	The notes in Table 9-2 • Designer State (resulting from I/O attribute modification) were revised to clarify which device families support programmable input delay (SAR 39666).	253
June 2011	Figure 9-2 • SmartGen Catalog was updated (SAR 24310). Figure 8-3 • Expanded I/O Section and the step associated with it were deleted to reflect changes in the software.	254
	The following rule was added to the "VREF Rules for the Implementation of Voltage-Referenced I/O Standards" section: 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 (SAR 24310).	265
July 2010	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 9-1 • Flash-Based FPGAs.	252
	The notes for Table 9-2 • Designer State (resulting from I/O attribute modification) were revised to indicate that skew control and input delay do not apply to nano devices.	253
v1.3 (October 2008)	The "Flash FPGAs I/O Support" section was revised to include new families and make the information more concise.	252
v1.2 (June 2008)	The following changes were made to the family descriptions in Table 9-1 • Flash-Based FPGAs: <ul style="list-style-type: none"> <li>ProASIC3L was updated to include 1.5 V.</li> <li>The number of PLLs for ProASIC3E was changed from five to six.</li> </ul>	252
v1.1 (March 2008)	This document was previously part of the <i>I/O Structures in IGLOO and ProASIC3 Devices</i> document. The content was separated and made into a new document.	N/A
	Table 9-2 • Designer State (resulting from I/O attribute modification) was updated to include note 2 for IGLOO PLUS.	253

## DDR Support in Flash-Based Devices

The flash FPGAs listed in Table 10-1 support the DDR feature and the functions described in this document.

**Table 10-1 • 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
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 10-1. 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 10-1. 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*.

### STAPL File with AES Encryption

- Does not contain AES key / FlashLock Key information
- Intended for transmission through web or service to unsecured locations for programming

```
=====
NOTE "CREATOR" "Designer Version: 6.1.1.108";
NOTE "DEVICE" "A3PE600";
NOTE "PACKAGE" "208 PQFP";
NOTE "DATE" "2005/04/08";
NOTE "STAPL_VERSION" "JESD71";
NOTE "IDCODE" "$123261CF";
NOTE "DESIGN" "counter32";
NOTE "CHECKSUM" "$EF57";
NOTE "SAVE_DATA" "FFromStream";
NOTE "SECURITY" "ENCRYPT FROM CORE ";
NOTE "ALG_VERSION" "1";
NOTE "MAX_FREQ" "20000000";
NOTE "SILSIG" "$00000000";
```

## Conclusion

The new and enhanced security features offered in Fusion, IGLOO, and ProASIC3 devices provide state-of-the-art security to designs programmed into these flash-based devices. Microsemi low power flash devices employ the encryption standard used by NIST and the U.S. government—AES using the 128-bit Rijndael algorithm.

The combination of an on-chip AES decryption engine and FlashLock technology provides the highest level of security against invasive attacks and design theft, implementing the most robust and secure ISP solution. These security features protect IP within the FPGA and protect the system from cloning, wholesale “black box” copying of a design, invasive attacks, and explicit IP or data theft.

## Glossary

Term	Explanation
Security Header programming file	Programming file used to program the FlashLock Pass Key and/or AES key into the device to secure the FPGA, FlashROM, and/or FBs.
AES (encryption) key	128-bit key defined by the user when the AES encryption option is set in the Microsemi Designer software when generating the programming file.
FlashLock Pass Key	128-bit key defined by the user when the FlashLock option is set in the Microsemi Designer software when generating the programming file.  The FlashLock Key protects the security settings programmed to the device. Once a device is programmed with FlashLock, whatever settings were chosen at that time are secure.
FlashLock	The combined security features that protect the device content from attacks. These features are the following: <ul style="list-style-type: none"> <li>• Flash technology that does not require an external bitstream to program the device</li> <li>• FlashLock Pass Key that secures device content by locking the security settings and preventing access to the device as defined by the user</li> <li>• AES key that allows secure, encrypted device reprogrammability</li> </ul>

## References

National Institute of Standards and Technology. “ADVANCED ENCRYPTION STANDARD (AES) Questions and Answers.” 28 January 2002 (10 January 2005).  
See <http://csrc.nist.gov/archive/aes/index1.html> for more information.

## Security in ARM-Enabled Low Power Flash Devices

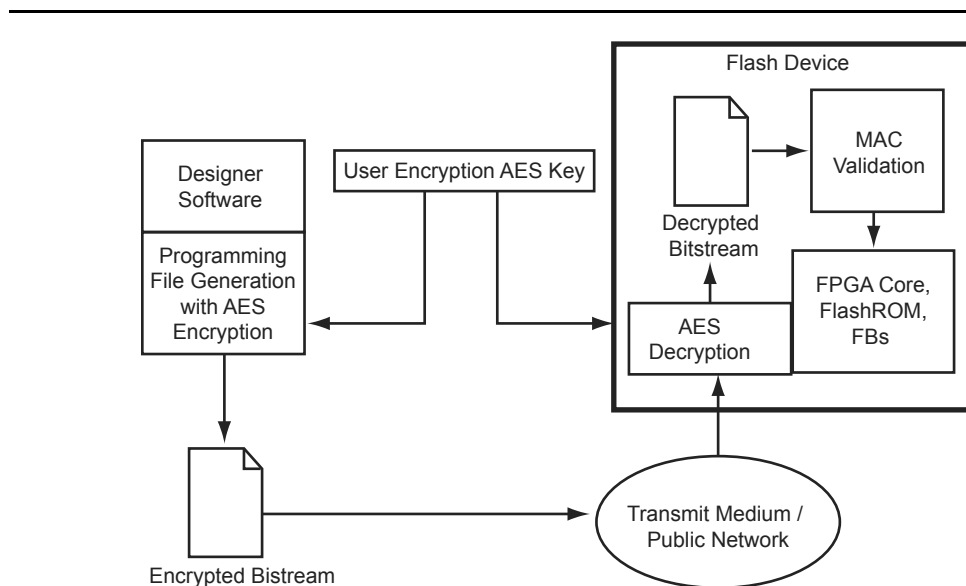
There are slight differences between the regular flash device and the ARM-enabled flash devices, which have the M1 prefix.

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



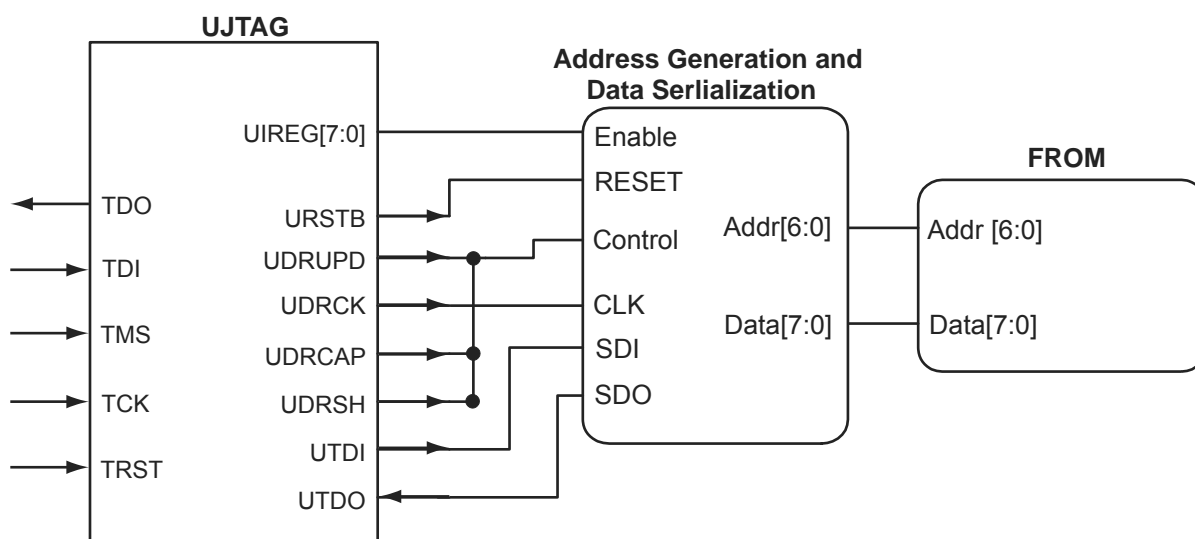
**Figure 13-1 • AES-128 Security Features**



## 17 – UJTAG Applications in Microsemi's Low Power Flash Devices

### Introduction

In Fusion, IGLOO, and ProASIC3 devices, there is bidirectional access from the JTAG port to the core VersaTiles during normal operation of the device (Figure 17-1). User JTAG (UJTAG) is the ability for the design to use the JTAG ports for access to the device for updates, etc. While regular JTAG is used, the UJTAG tiles, located at the southeast area of the die, are directly connected to the JTAG Test Access Port (TAP) Controller in normal operating mode. As a result, all the functional blocks of the device, such as Clock Conditioning Circuits (CCCs) with PLLs, SRAM blocks, embedded FlashROM, flash memory blocks, and I/O tiles, can be reached via the JTAG ports. The UJTAG functionality is available by instantiating the UJTAG macro directly in the source code of a design. Access to the FPGA core VersaTiles from the JTAG ports enables users to implement different applications using the TAP Controller (JTAG port). This document introduces the UJTAG tile functionality and discusses a few application examples. However, the possible applications are not limited to what is presented in this document. UJTAG can serve different purposes in many designs as an elementary or auxiliary part of the design. For detailed usage information, refer to the "Boundary Scan in Low Power Flash Devices" section on page 357.



**Figure 17-1 • Block Diagram of Using UJTAG to Read FlashROM Contents**