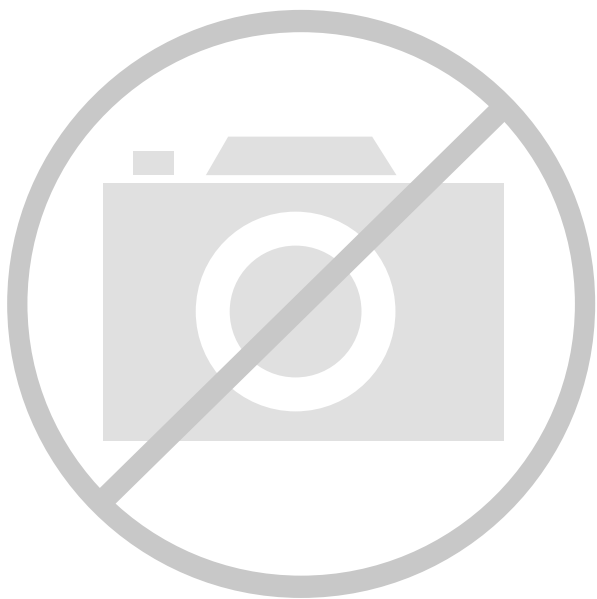




Welcome to [E-XFL.COM](#)

Understanding [Embedded - FPGAs \(Field Programmable Gate Array\)](#)

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

Applications of Embedded - FPGAs

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

Details

Product Status	Obsolete
Number of LABs/CLBs	-
Number of Logic Elements/Cells	-
Total RAM Bits	110592
Number of I/O	154
Number of Gates	600000
Voltage - Supply	1.14V ~ 1.575V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	208-BFQFP
Supplier Device Package	208-PQFP (28x28)
Purchase URL	https://www.e-xfl.com/product-detail/microsemi/a3p600l-1pqg208

FlashROM Security Use Models	311
Generating Programming Files	313
Conclusion	324
Glossary	324
References	324
Related Documents	325
List of Changes	325
13 In-System Programming (ISP) of Microsemi's Low Power Flash Devices Using FlashPro4/3/3X...	327
Introduction	327
ISP Architecture	327
ISP Support in Flash-Based Devices	328
Programming Voltage (VPUMP) and VJTAG	329
Nonvolatile Memory (NVM) Programming Voltage	329
IEEE 1532 (JTAG) Interface	330
Security	330
Security in ARM-Enabled Low Power Flash Devices	331
FlashROM and Programming Files	333
Programming Solution	334
ISP Programming Header Information	335
Board-Level Considerations	337
Conclusion	338
Related Documents	338
List of Changes	339
14 Core Voltage Switching Circuit for IGLOO and ProASIC3L In-System Programming	341
Introduction	341
Microsemi's Flash Families Support Voltage Switching Circuit	342
Circuit Description	343
Circuit Verification	344
DirectC	346
Conclusion	346
List of Changes	347
15 Microprocessor Programming of Microsemi's Low Power Flash Devices	349
Introduction	349
Microprocessor Programming Support in Flash Devices	350
Programming Algorithm	351
Implementation Overview	351
Hardware Requirement	354
Security	354
Conclusion	355
List of Changes	356
16 Boundary Scan in Low Power Flash Devices	357
Boundary Scan	357
TAP Controller State Machine	357
Microsemi's Flash Devices Support the JTAG Feature	358
Boundary Scan Support in Low Power Devices	359
Boundary Scan Opcodes	359

- Use quadrant global region assignments by finding the clock net associated with the CCC macro under the Nets tab and creating a quadrant global region for the net, as shown in Figure 4-33.
-

Figure 4-33 • Quadrant Clock Assignment for a Global Net

External I/O–Driven CCCs

The above-mentioned recommendation for proper layout techniques will ensure the correct assignment. It is possible that, especially with External I/O–Driven CCC macros, placement of the CCC macro in a desired location may not be achieved. For example, assigning an input port of an External I/O–Driven CCC near a particular CCC location does not guarantee global assignments to the desired location. This is because the clock inputs of External I/O–Driven CCCs can be assigned to any I/O location; therefore, it is possible that the CCC connected to the clock input will be routed to a location other than the one closest to the I/O location, depending on resource availability and placement constraints.

Clock Placer

The clock placer is a placement engine for low power flash devices that places global signals on the chip global and quadrant global networks. Based on the clock assignment constraints for the chip global and quadrant global clocks, it will try to satisfy all constraints, as well as creating quadrant clock regions when necessary. If the clock placer fails to create the quadrant clock regions for the global signals, it will report an error and stop Layout.

The user must ensure that the constraints set to promote clock signals to quadrant global networks are valid.

Cascading CCCs

The CCCs in low power flash devices can be cascaded. Cascading CCCs can help achieve more accurate PLL output frequency results than those achievable with a single CCC. In addition, this technique is useful when the user application requires the output clock of the PLL to be a multiple of the reference clock by an integer greater than the maximum feedback divider value of the PLL (divide by 128) to achieve the desired frequency.

For example, the user application may require a 280 MHz output clock using a 2 MHz input reference clock, as shown in Figure 4-34 on page 126.

Date	Changes	Page
v1.4 (December 2008)	The "CCC Support in Microsemi's Flash Devices" section was updated to include IGLOO nano and ProASIC3 nano devices.	79
	Figure 4-2 • CCC Options: Global Buffers with No Programmable Delay was revised to add the CLKBIBUF macro.	80
	The description of the reference clock was revised in Table 4-2 • Input and Output Description of the CLKDLY Macro.	81
	Figure 4-7 • Clock Input Sources (30 k gates devices and below) is new. Figure 4-8 • Clock Input Sources Including CLKBUF, CLKBUF_LVDS/LVPECL, and CLKINT (60 k gates devices and above) applies to 60 k gate devices and above.	88
	The "IGLOO and ProASIC3" section was updated to include information for IGLOO nano devices.	89
	A note regarding Fusion CCCs was added to Figure 4-9 • Illustration of Hardwired I/O (global input pins) Usage for IGLOO and ProASIC3 devices 60 k Gates and Larger and the name of the figure was changed from Figure 4-8 • Illustration of Hardwired I/O (global input pins) Usage. Figure 4-10 • Illustration of Hardwired I/O (global input pins) Usage for IGLOO and ProASIC3 devices 30 k Gates and Smaller is new.	90
	Table 4-5 • Number of CCCs by Device Size and Package was updated to include IGLOO nano and ProASIC3 nano devices. Entries were added to note differences for the CS81, CS121, and CS201 packages.	94
	The "Clock Conditioning Circuits without Integrated PLLs" section was rewritten.	95
	The "IGLOO and ProASIC3 CCC Locations" section was updated for nano devices.	97
	Figure 4-13 • CCC Locations in the 15 k and 30 k Gate Devices was deleted.	4-20
v1.3 (October 2008)	This document was updated to include Fusion and RT ProASIC3 device information. Please review the document very carefully.	N/A
	The "CCC Support in Microsemi's Flash Devices" section was updated.	79
	In the "Global Buffer with Programmable Delay" section, the following sentence was changed from: "In this case, the I/O must be placed in one of the dedicated global I/O locations." To "In this case, the software will automatically place the dedicated global I/O in the appropriate locations."	80
	Figure 4-4 • CCC Options: Global Buffers with PLL was updated to include OADIVRST and OADIVHALF.	83
	In Figure 4-6 • CCC with PLL Block "fixed delay" was changed to "programmable delay".	83
	Table 4-3 • Input and Output Signals of the PLL Block was updated to include OADIVRST and OADIVHALF descriptions.	84
	Table 4-8 • Configuration Bit Descriptions for the CCC Blocks was updated to include configuration bits 88 to 81. Note 2 is new. In addition, the description for bit <76:74> was updated.	106
	Table 4-16 • Fusion Dynamic CCC Clock Source Selection and Table 4-17 • Fusion Dynamic CCC NGMUX Configuration are new.	110
	Table 4-18 • Fusion Dynamic CCC Division by Half Configuration and Table 4-19 • Configuration Bit <76:75> / VCOSEL<2:1> Selection for All Families are new.	111

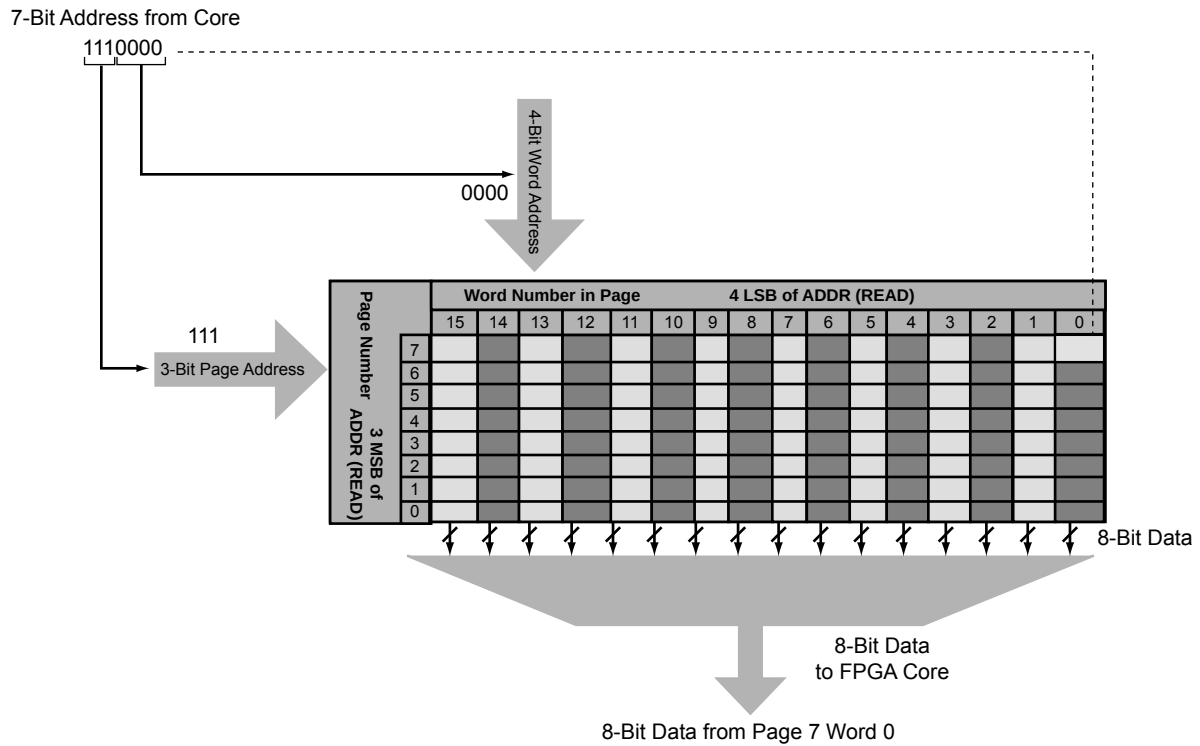


Figure 5-7 • Accessing FlashROM Using FPGA Core

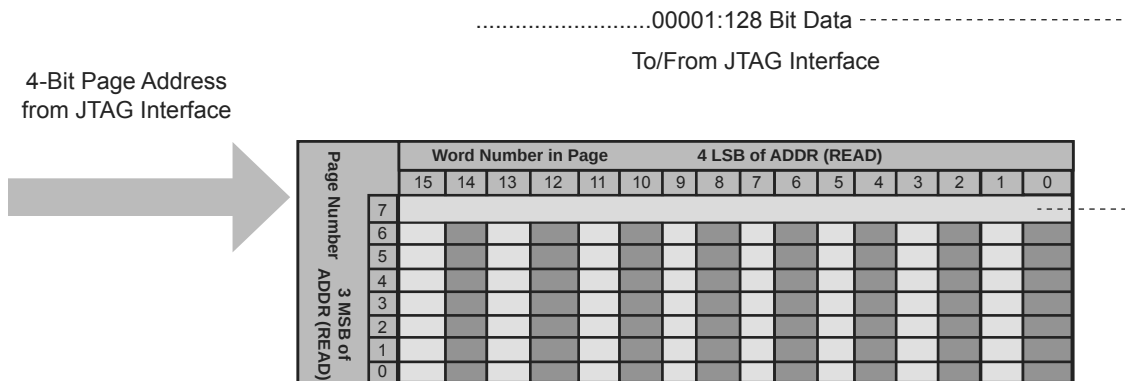


Figure 5-8 • Accessing FlashROM Using JTAG Port

SRAM and FIFO Architecture

To meet the needs of high-performance designs, the memory blocks operate strictly in synchronous mode for both read and write operations. The read and write clocks are completely independent, and each can operate at any desired frequency up to 250 MHz.

- 4k×1, 2k×2, 1k×4, 512×9 (dual-port RAM—2 read / 2 write or 1 read / 1 write)
- 512×9, 256×18 (2-port RAM—1 read / 1 write)
- Sync write, sync pipelined / nonpipelined read

Automotive ProASIC3 devices support single-port SRAM capabilities or dual-port SRAM only under specific conditions. Dual-port mode is supported if the clocks to the two SRAM ports are the same and 180° out of phase (i.e., the port A clock is the inverse of the port B clock). The Libero SoC software macro libraries support a dual-port macro only. For use of this macro as a single-port SRAM, the inputs and clock of one port should be tied off (grounded) to prevent errors during design compile. For use in dual-port mode, the same clock with an inversion between the two clock pins of the macro should be used in the design to prevent errors during compile.

The memory block includes dedicated FIFO control logic to generate internal addresses and external flag logic (FULL, EMPTY, AFULL, AEMPTY).

Simultaneous dual-port read/write and write/write operations at the same address are allowed when certain timing requirements are met.

During RAM operation, addresses are sourced by the user logic, and the FIFO controller is ignored. In FIFO mode, the internal addresses are generated by the FIFO controller and routed to the RAM array by internal MUXes.

The low power flash device architecture enables the read and write sizes of RAMs to be organized independently, allowing for bus conversion. For example, the write size can be set to 256×18 and the read size to 512×9.

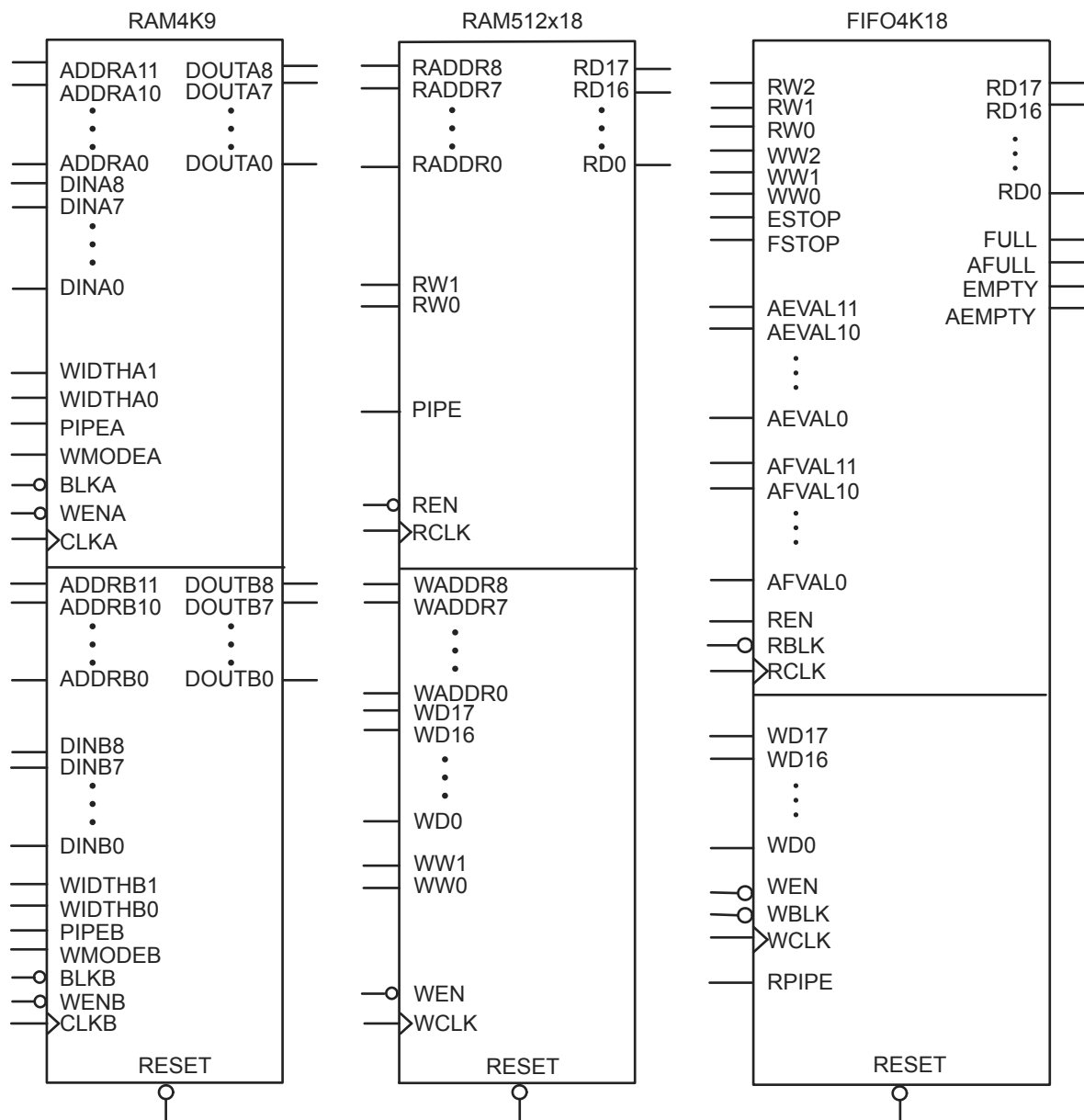
Both the write width and read width for the RAM blocks can be specified independently with the WW (write width) and RW (read width) pins. The different D×W configurations are 256×18, 512×9, 1k×4, 2k×2, and 4k×1. When widths of one, two, or four are selected, the ninth bit is unused. For example, when writing nine-bit values and reading four-bit values, only the first four bits and the second four bits of each nine-bit value are addressable for read operations. The ninth bit is not accessible.

Conversely, when writing four-bit values and reading nine-bit values, the ninth bit of a read operation will be undefined. The RAM blocks employ little-endian byte order for read and write operations.

Memory Blocks and Macros

Memory blocks can be configured with many different aspect ratios, but are generically supported in the macro libraries as one of two memory elements: RAM4K9 or RAM512X18. The RAM4K9 is configured as a true dual-port memory block, and the RAM512X18 is configured as a two-port memory block. Dual-port memory allows the RAM to both read from and write to either port independently. Two-port memory allows the RAM to read from one port and write to the other using a common clock or independent read and write clocks. If needed, the RAM4K9 blocks can be configured as two-port memory blocks. The memory block can be configured as a FIFO by combining the basic memory block with dedicated FIFO controller logic. The FIFO macro is named FIFO4KX18 (Figure 6-3 on page 152).

Clocks for the RAM blocks can be driven by the VersaNet (global resources) or by regular nets. When using local clock segments, the clock segment region that encompasses the RAM blocks can drive the RAMs. In the dual-port configuration (RAM4K9), each memory block port can be driven by either rising-edge or falling-edge clocks. Each port can be driven by clocks with different edges. Though only a rising-edge clock can drive the physical block itself, the Microsemi Designer software will automatically bubble-push the inversion to properly implement the falling-edge trigger for the RAM block.



Notes:

1. Automotive ProASIC3 devices restrict RAM4K9 to a single port or to dual ports with the same clock 180° out of phase (inverted) between clock pins. In single-port mode, inputs to port B should be tied to ground to prevent errors during compile. This warning applies only to automotive ProASIC3 parts of certain revisions and earlier. Contact Technical Support at soc_tech@microsemi.com for information on the revision number for a particular lot and date code.
2. For FIFO4K18, the same clock 180° out of phase (inverted) between clock pins should be used.

Figure 6-3 • Supported Basic RAM Macros

Software Support

The SmartGen core generator is the easiest way to select and configure the memory blocks (Figure 6-12). SmartGen automatically selects the proper memory block type and aspect ratio, and cascades the memory blocks based on the user's selection. SmartGen also configures any additional signals that may require tie-off.

SmartGen will attempt to use the minimum number of blocks required to implement the desired memory. When cascading, SmartGen will configure the memory for width before configuring for depth. For example, if the user requests a 256×8 FIFO, SmartGen will use a 512×9 FIFO configuration, not 256×18.

Figure 6-12 • SmartGen Core Generator Interface

SmartGen enables the user to configure the desired RAM element to use either a single clock for read and write, or two independent clocks for read and write. The user can select the type of RAM as well as the width/depth and several other parameters (Figure 6-13).

Figure 6-13 • SmartGen Memory Configuration Interface

SmartGen also has a Port Mapping option that allows the user to specify the names of the ports generated in the memory block (Figure 6-14).

Figure 6-14 • Port Mapping Interface for SmartGen-Generated Memory

SmartGen also configures the FIFO according to user specifications. Users can select no flags, static flags, or dynamic flags. Static flag settings are configured using configuration flash and cannot be altered

I/O Standards

Single-Ended Standards

These I/O standards use a push-pull CMOS output stage with a voltage referenced to system ground to designate logical states. The input buffer configuration, output drive, and I/O supply voltage (VCCI) vary among the I/O standards (Figure 7-5).

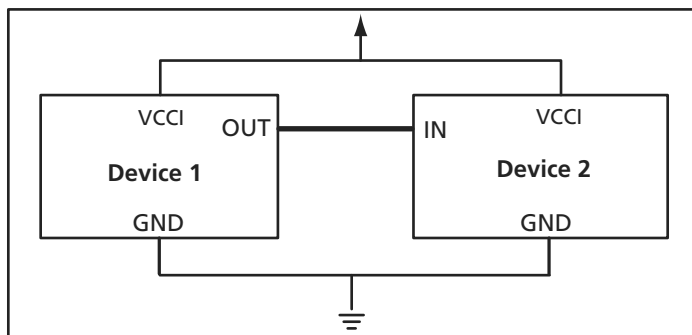


Figure 7-5 • Single-Ended I/O Standard Topology

The advantage of these standards is that a common ground can be used for multiple I/Os. This simplifies board layout and reduces system cost. Their low-edge-rate (dv/dt) data transmission causes less electromagnetic interference (EMI) on the board. However, they are not suitable for high-frequency (>200 MHz) switching due to noise impact and higher power consumption.

LVTTL (Low-Voltage TTL)

This is a general-purpose standard (EIA/JESD8-B) for 3.3 V applications. It uses an LVTTL input buffer and a push-pull output buffer. The LVTTL output buffer can have up to six different programmable drive strengths. The default drive strength is 12 mA. VCCI is 3.3 V. Refer to "I/O Programmable Features" on page 188 for details.

LVC MOS (Low-Voltage CMOS)

The low power flash devices provide four different kinds of LVC MOS: LVC MOS 3.3 V, LVC MOS 2.5 V, LVC MOS 1.8 V, and LVC MOS 1.5 V. LVC MOS 3.3 V is an extension of the LVC MOS standard (JESD8-B-compliant) used for general-purpose 3.3 V applications.

LVC MOS 2.5 V is an extension of the LVC MOS standard (JESD8-5-compliant) used for general-purpose 2.5 V applications.

There is yet another standard supported by IGLOO and ProASIC3 devices (except A3P030): LVC MOS 2.5/5.0 V. This standard is similar to LVC MOS 2.5 V, with the exception that it can support up to 3.3 V on the input side (2.5 V output drive).

LVC MOS 1.8 V is an extension of the LVC MOS standard (JESD8-7-compliant) used for general-purpose 1.8 V applications. LVC MOS 1.5 V is an extension of the LVC MOS standard (JESD8-11-compliant) used for general-purpose 1.5 V applications.

The VCCI values for these standards are 3.3 V, 2.5 V, 1.8 V, and 1.5 V, respectively. Like LVTTL, the output buffer has up to seven different programmable drive strengths (2, 4, 6, 8, 12, 16, and 24 mA). Refer to "I/O Programmable Features" on page 188 for details.

3.3 V PCI (Peripheral Component Interface)

This standard specifies support for both 33 MHz and 66 MHz PCI bus applications. It uses an LVTTL input buffer and a push-pull output buffer. With the aid of an external resistor, this I/O standard can be 5 V-compliant for low power flash devices. It does not have programmable drive strength.

3.3 V PCI-X (Peripheral Component Interface Extended)

An enhanced version of the PCI specification, 3.3 V PCI-X can support higher average bandwidths; it increases the speed that data can move within a computer from 66 MHz to 133 MHz. It is backward-

GTL+ (Gunning Transceiver Logic Plus)

This is an enhanced version of GTL that has defined slew rates and higher voltage levels. It requires a differential amplifier input buffer and an open-drain output buffer. Even though the output is open-drain, VCCI must be connected to either 2.5 V or 3.3 V. The reference voltage (VREF) is 1 V.

Differential Standards

These standards require two I/Os per signal (called a “signal pair”). Logic values are determined by the potential difference between the lines, not with respect to ground. This is why differential drivers and receivers have much better noise immunity than single-ended standards. The differential interface standards offer higher performance and lower power consumption than their single-ended counterparts. Two I/O pins are used for each data transfer channel. Both differential standards require resistor termination.

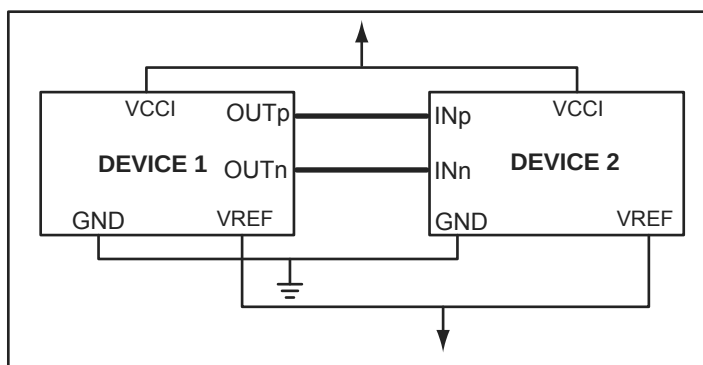


Figure 7-7 • Differential Topology

LVPECL (Low-Voltage Positive Emitter Coupled Logic)

LVPECL requires that one data bit be carried through two signal lines; therefore, two pins are needed per input or output. It also requires external resistor termination. The voltage swing between the two signal lines is approximately 850 mV. When the power supply is +3.3 V, it is commonly referred to as Low-Voltage PECL (LVPECL). Refer to the device datasheet for the full implementation of the LVPECL transmitter and receiver.

LVDS (Low-Voltage Differential Signal)

LVDS is a moderate-speed differential signaling system, in which the transmitter generates two different voltages that are compared at the receiver. LVDS uses a differential driver connected to a terminated receiver through a constant-impedance transmission line. It requires that one data bit be carried through two signal lines; therefore, the user will need two pins per input or output. It also requires external resistor termination. The voltage swing between the two signal lines is approximately 350 mV. VCCI is 2.5 V. Low power flash devices contain dedicated circuitry supporting a high-speed LVDS standard that has its own user specification. Refer to the device datasheet for the full implementation of the LVDS transmitter and receiver.

B-LVDS/M-LVDS

Bus LVDS (B-LVDS) refers to bus interface circuits based on LVDS technology. Multipoint LVDS (M-LVDS) specifications extend the LVDS standard to high-performance multipoint bus applications. Multidrop and multipoint bus configurations may contain any combination of drivers, receivers, and transceivers. Microsemi LVDS drivers provide the higher drive current required by B-LVDS and M-LVDS to accommodate the loading. The driver requires series terminations for better signal quality and to control voltage swing. Termination is also required at both ends of the bus, since the driver can be located anywhere on the bus. These configurations can be implemented using TRIBUF_LVDS and BIBUF_LVDS macros along with appropriate terminations. Multipoint designs using Microsemi LVDS macros can achieve up to 200 MHz with a maximum of 20 loads. A sample application is given in Figure 7-8. The input and output buffer delays are available in the LVDS sections in the datasheet.

I/O Features

Low power flash devices support multiple I/O features that make board design easier. For example, an I/O feature like Schmitt Trigger in the ProASIC3E input buffer saves the board space that would be used by an external Schmitt trigger for a slow or noisy input signal. These features are also programmable for each I/O, which in turn gives flexibility in interfacing with other components. The following is a detailed description of all available features in low power flash devices.

I/O Programmable Features

Low power flash devices offer many flexible I/O features to support a wide variety of board designs. Some of the features are programmable, with a range for selection. Table 7-7 lists programmable I/O features and their ranges.

Table 7-7 • Programmable I/O Features (user control via I/O Attribute Editor)

Feature ¹	Description	Range
Slew Control	Output slew rate	HIGH, LOW
Output Drive (mA)	Output drive strength	2, 4, 6, 8, 12, 16, 24
Skew Control	Output tristate enable delay option	ON, OFF
Resistor Pull	Resistor pull circuit	Up, Down, None
Input Delay ²	Input delay	OFF, 0–7
Schmitt Trigger	Schmitt trigger for input only	ON, OFF

Notes:

1. Limitations of these features with respect to different devices are discussed in later sections.
2. Programmable input delay is applicable only to ProASIC3EL and RT ProASIC3 devices.

Hot-Swap Support

A pull-up clamp diode must not be present in the I/O circuitry if the hot-swap feature is used. The 3.3 V PCI standard requires a pull-up clamp diode on the I/O, so it cannot be selected if hot-swap capability is required. The A3P030 device does not support 3.3 V PCI, so it is the only device in the ProASIC3 family that supports the hot-swap feature. All devices in the ProASIC3E family are hot-swappable. All standards except LVCMOS 2.5/5.0 V and 3.3 V PCI/PCI-X support the hot-swap feature.

The hot-swap feature appears as a read-only check box in the I/O Attribute Editor that shows whether an I/O is hot-swappable or not. Refer to the *"Power-Up/Down Behavior of Low Power Flash Devices"* section on page 373 for details on hot-swapping.

Hot-swapping (also called hot-plugging) is the operation of hot insertion or hot removal of a card in a powered-up system. The levels of hot-swap support and examples of related applications are described in Table 7-8 on page 189 to Table 7-11 on page 190. The I/Os also need to be configured in hot-insertion mode if hot-plugging compliance is required. The AGL030 and A3P030 devices have an I/O structure that allows the support of Level 3 and Level 4 hot-swap with only two levels of staging.

Solution 4

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.

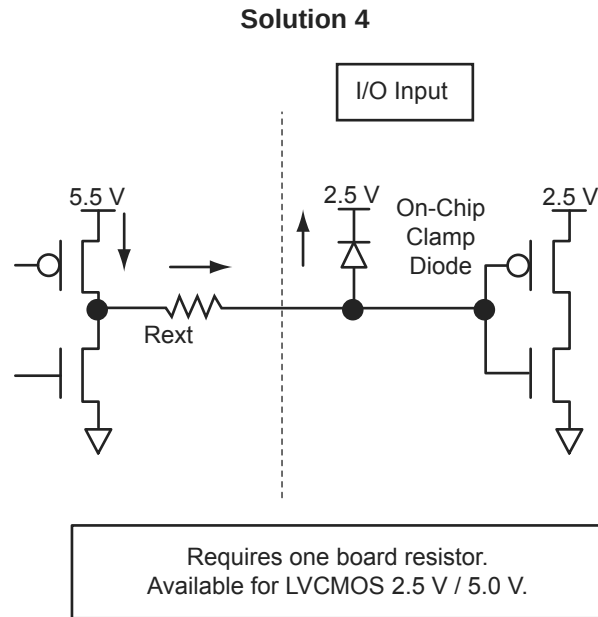


Figure 7-12 • Solution 4

I/O Register Combining

Every I/O has several embedded registers in the I/O tile that are close to the I/O pads. Rather than using the internal register from the core, the user has the option of using these registers for faster clock-to-out timing, and external hold and setup. When combining these registers at the I/O buffer, some architectural rules must be met. Provided these rules are met, the user can enable register combining globally during Compile (as shown in the "Compiling the Design" section on page 261).

This feature is supported by all I/O standards.

Rules for Registered I/O Function

1. The fanout between an I/O pin (D, Y, or E) and a register must be equal to one for combining to be considered on that pin.
2. All registers (Input, Output, and Output Enable) connected to an I/O must share the same clear or preset function:
 - If one of the registers has a CLR pin, all the other registers that are candidates for combining in the I/O must have a CLR pin.
 - If one of the registers has a PRE pin, all the other registers that are candidates for combining in the I/O must have a PRE pin.
 - If one of the registers has neither a CLR nor a PRE pin, all the other registers that are candidates for combining must have neither a CLR nor a PRE pin.
 - If the clear or preset pins are present, they must have the same polarity.
 - If the clear or preset pins are present, they must be driven by the same signal (net).
3. Registers connected to an I/O on the Output and Output Enable pins must have the same clock and enable function:
 - Both the Output and Output Enable registers must have an E pin (clock enable), or none at all.
 - If the E pins are present, they must have the same polarity. The CLK pins must also have the same polarity.

In some cases, the user may want registers to be combined with the input of a buffer while maintaining the output as-is. This can be achieved by using PDC commands as follows:

```
set_io <signal name> -REGISTER yes -----register will combine
set_preserve <signal name> ----register will not combine
```

Weak Pull-Up and Weak Pull-Down Resistors

IGLOO and ProASIC3 devices support optional weak pull-up and pull-down resistors on each I/O pin. When the I/O is pulled up, it is connected to the VCCI of its corresponding I/O bank. When it is pulled down, it is connected to GND. Refer to the datasheet for more information.

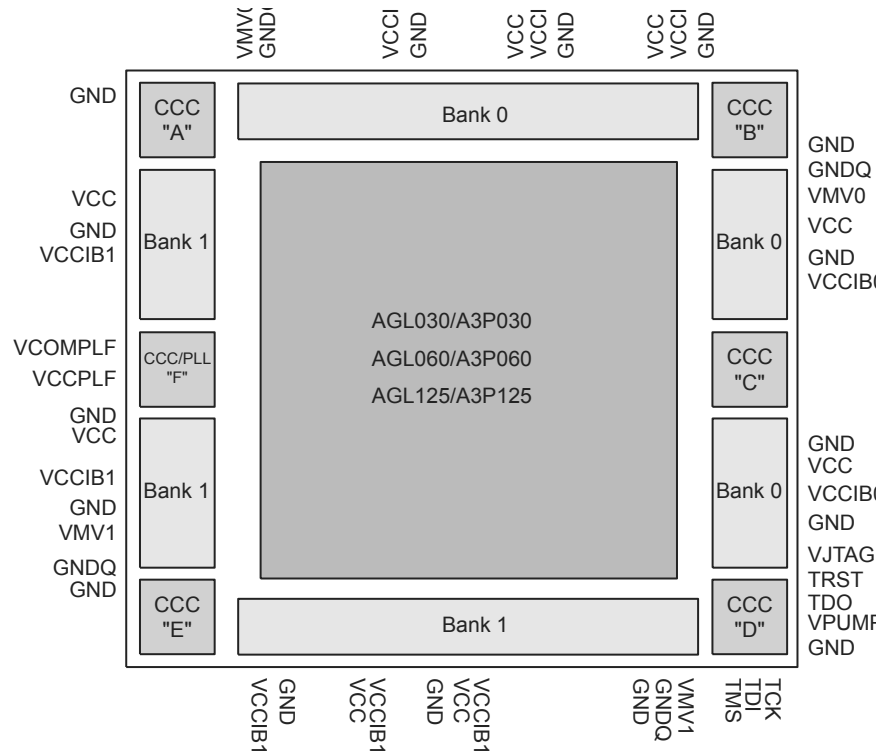
For low power applications, configuration of the pull-up or pull-down of the I/O can be used to set the I/O to a known state while the device is in Flash*Freeze mode. Refer to the "Flash*Freeze Technology and Low Power Modes in IGLOO and ProASIC3L Devices" chapter of the *IGLOO FPGA Fabric User's Guide* or *ProASIC3L FPGA Fabric User's Guide* for more information.

The Flash*Freeze (FF) pin cannot be configured with a weak pull-down or pull-up I/O attribute, as the signal needs to be driven at all times.

Output Slew Rate Control

The slew rate is the amount of time an input signal takes to get from logic Low to logic High or vice versa. It is commonly defined as the propagation delay between 10% and 90% of the signal's voltage swing. Slew rate control is available for the output buffers of low power flash devices. The output buffer has a programmable slew rate for both HIGH-to-LOW and LOW-to-HIGH transitions. Slew rate control is available for LVTTTL, LVCMOS, and PCI-X I/O standards. The other I/O standards have a preset slew value.

The slew rate can be implemented by using a PDC command (Table 7-5 on page 179), setting it "High" or "Low" in the I/O Attribute Editor in Designer, or instantiating a special I/O macro. The default slew rate value is "High."



Note: The 30 k gate devices do not support a PLL (V_{COMPLF} and V_{CCPLF} pins).

Figure 7-19 • Naming Conventions of IGLOO and ProASIC3 Devices with Two I/O Banks – Top View

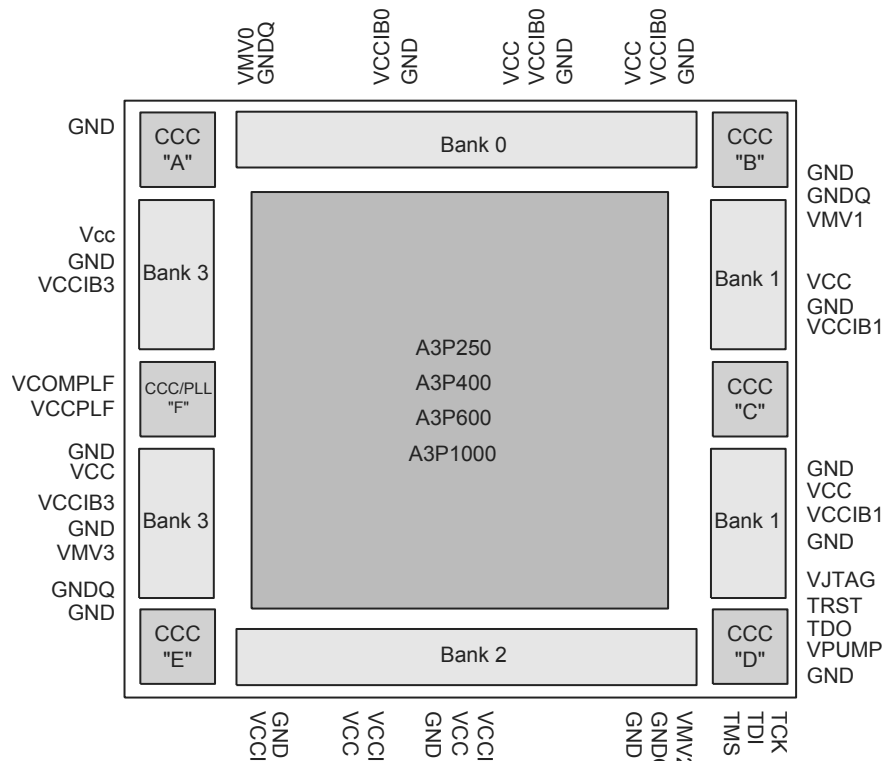


Figure 7-20 • Naming Conventions of IGLOO and ProASIC3 Devices with Four I/O Banks – Top View

Table 8-9 • Hot-Swap Level 1

Description	Cold-swap
Power Applied to Device	No
Bus State	–
Card Ground Connection	–
Device Circuitry Connected to Bus Pins	–
Example Application	System and card with Microsemi FPGA chip are powered down, and the card is plugged into the system. Then the power supplies are turned on for the system but not for the FPGA on the card.
Compliance of IGLOO and ProASIC3 Devices	30 k gate devices: Compliant Other IGLOO/ProASIC3 devices: Compliant if bus switch used to isolate FPGA I/Os from rest of system IGLOOe/ProASIC3E devices: Compliant I/Os can, but do not have to be set to hot-insertion mode.

Table 8-10 • Hot-Swap Level 2

Description	Hot-swap while reset
Power Applied to Device	Yes
Bus State	Held in reset state
Card Ground Connection	Reset must be maintained for 1 ms before, during, and after insertion/removal.
Device Circuitry Connected to Bus Pins	–
Example Application	In the PCI hot-plug specification, reset control circuitry isolates the card busses until the card supplies are at their nominal operating levels and stable.
Compliance of IGLOO and ProASIC3 Devices	30 k gate devices, all IGLOOe/ProASIC3E devices: Compliant I/Os can but do not have to be set to hot-insertion mode. Other IGLOO/ProASIC3 devices: Compliant

Instantiating DDR Registers

Using SmartGen is the simplest way to generate the appropriate RTL files for use in the design. Figure 10-4 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 10-5 on page 277 through Figure 10-8 on page 280 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 10-4 • Example of Using SmartGen to Generate a DDR SSTL2 Class I Input Register

```

DDR_OUT_0_inst : DDR_OUT
port map(DR => DataR, DF => DataF, CLK => CLK, CLR => CLR, Q => Q);
TRIBUFF_F_8U_0_inst : TRIBUFF_F_8U
port map(D => Q, E => TrienAux, PAD => PAD);

end DEF_ARCH;

```

DDR Bidirectional Buffer

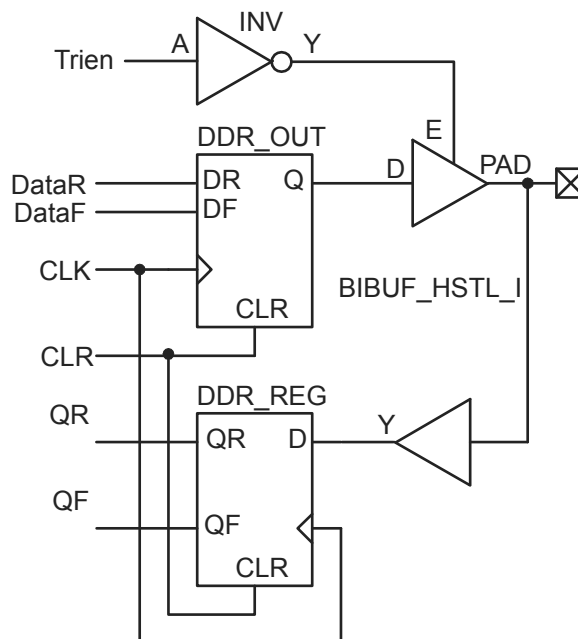


Figure 10-8 • DDR Bidirectional Buffer, LOW Output Enable (HSTL Class II)

Verilog

```

module DDR_BiDir_HSTL_I_LowEnb(DataR,DataF,CLR,CLK,Trien,QR,QF,PAD);

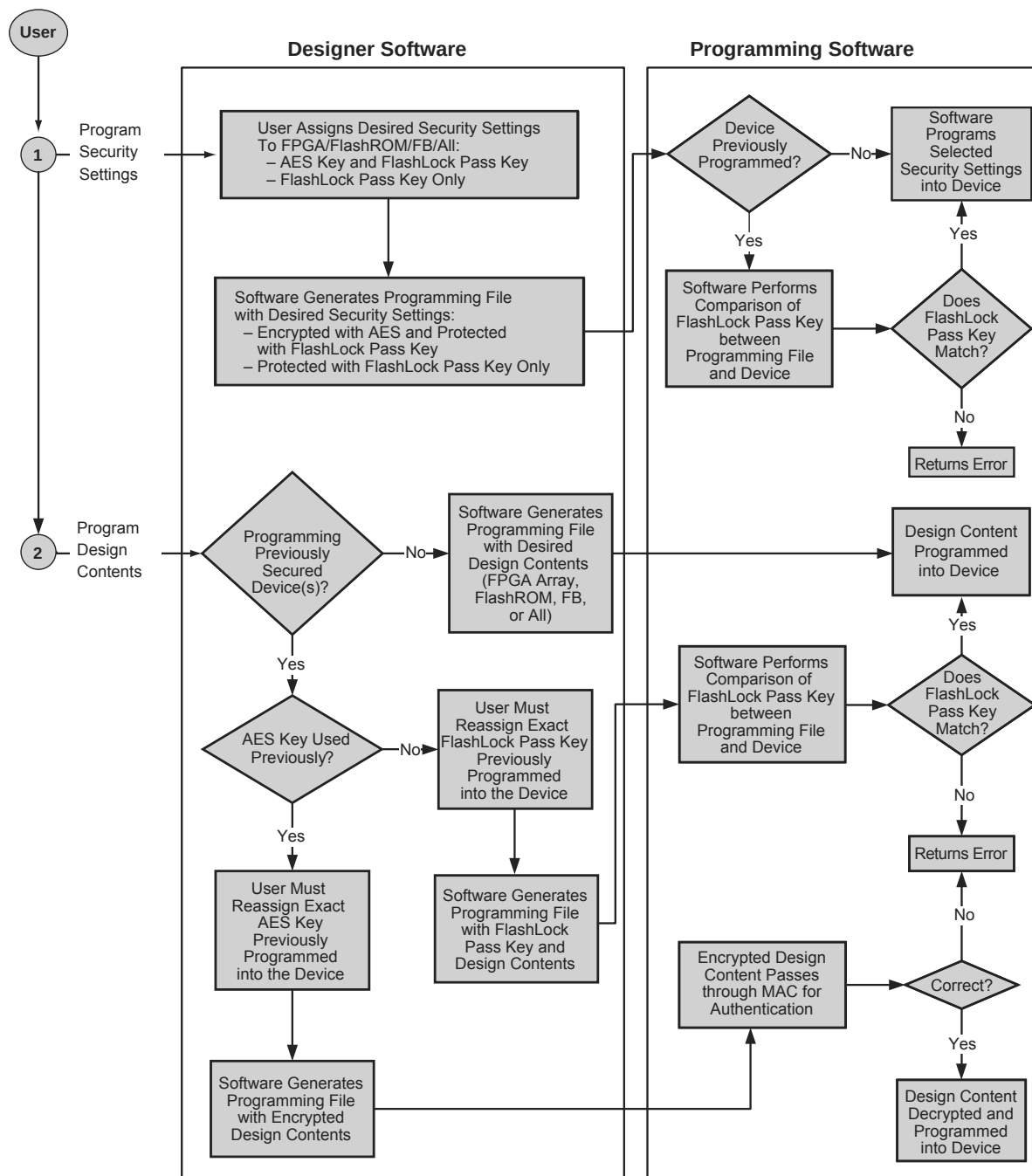
input  DataR, DataF, CLR, CLK, Trien;
output QR, QF;
inout  PAD;

wire TrienAux, D, Q;

    INV Inv_Tri(.A(Trien), .Y(TrienAux));
    DDR_OUT DDR_OUT_0_inst(.DR(DataR), .DF(DataF), .CLK(CLK), .CLR(CLR), .Q(Q));
    DDR_REG DDR_REG_0_inst(.D(D), .CLK(CLK), .CLR(CLR), .QR(QR), .QF(QF));
    BIBUF_HSTL_I BIBUF_HSTL_I_0_inst(.PAD(PAD), .D(Q), .E(TrienAux), .Y(D));

endmodule

```



Note: If programming the Security Header only, just perform sub-flow 1.
If programming design content only, just perform sub-flow 2.

Figure 12-9 • Security Programming Flows

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

Figure 14-4 • TRST Toggled LOW

In Figure 14-4, 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 14-3 on page 345, 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.