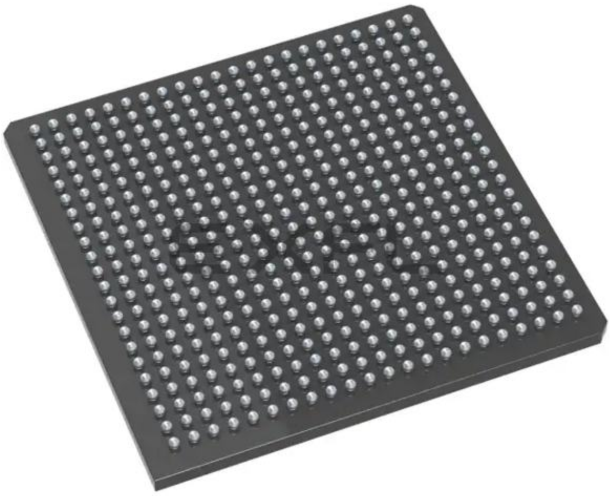
Microchip Technology - A3PE3000L-FG484I Datasheet



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	Active
Number of LABs/CLBs	-
Number of Logic Elements/Cells	-
Total RAM Bits	516096
Number of I/O	341
Number of Gates	3000000
Voltage - Supply	1.14V ~ 1.575V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	484-BGA
Supplier Device Package	484-FPBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/a3pe3000l-fg484i

IGLOO nano and IGLOO PLUS I/O State in Flash*Freeze Mode

In IGLOO nano and IGLOO PLUS devices, users have multiple options in how to configure I/Os during Flash*Freeze mode:

1. Hold the previous state
2. Set I/O pad to weak pull-up or pull-down
3. Tristate I/O pads

The I/O configuration must be configured by the user in the I/O Attribute Editor or in a PDC constraint file, and can be done on a pin-by-pin basis. The output hold feature will hold the output in the last registered state, using the I/O pad weak pull-up or pull-down resistor when the FF pin is asserted. When inputs are configured with the hold feature enabled, the FPGA core side of the input will hold the last valid state of the input pad before the device entered Flash*Freeze mode. The input pad can be driven to any value, configured as tristate, or configured with the weak pull-up or pull-down I/O pad feature during Flash*Freeze mode without affecting the hold state. If the weak pull-up or pull-down feature is used without the output hold feature, the input and output pads will maintain the configured weak pull-up or pull-down status during Flash*Freeze mode and normal operation. If a fixed weak pull-up or pull-down is defined on an output buffer or as bidirectional in output mode, and a hold state is also defined for the same pin, the pin will be configured in hold state mode during Flash*Freeze mode. During normal operation, the pin will be configured with the predefined weak pull-up or pull-down. Any I/Os that do not use the hold state or I/O pad weak pull-up or pull-down features will be tristated during Flash*Freeze mode and the FPGA core will be driven High by inputs. Inputs that are tristated during Flash*Freeze mode may be left floating without any reliability concern or impact to power consumption.

Table 2-6 shows the I/O pad state based on the configuration and buffer type.

Note that configuring weak pull-up or pull-down for the FF pin is not allowed.

Table 2-6 • **IGLOO nano and IGLOO PLUS Flash*Freeze Mode (type 1 and type 2)—I/O Pad State**

Buffer Type		Hold State	I/O Pad Weak Pull-Up/-Down	I/O Pad State in Flash*Freeze Mode
Input		Enabled	Enabled	Weak pull-up/pull-down [1]
		Disabled	Enabled	Weak pull-up/pull-down [2]
		Enabled	Disabled	Tristate [1]
		Disabled	Disabled	Tristate [2]
Output		Enabled	"Don't care"	Weak pull to hold state
		Disabled	Enabled	Weak pull-up/pull-down
		Disabled	Disabled	Tristate
Bidirectional / Tristate Buffer	E = 0 (input/tristate)	Enabled	Enabled	Weak pull-up/pull-down [1]
		Disabled	Enabled	Weak pull-up/pull-down [2]
		Enabled	Disabled	Tristate [1]
		Disabled	Disabled	Tristate [2]
	E = 1 (output)	Enabled	"Don't care"	Weak pull to hold state [3]
		Disabled	Enabled	Weak pull-up/pull-down
		Disabled	Disabled	Tristate

Notes:

1. *Internal core logic driven by this input buffer will be set to the value this I/O had when entering Flash*Freeze mode.*
2. *Internal core logic driven by this input buffer will be tied High as long as the device is in Flash*Freeze mode.*
3. *For bidirectional buffers: Internal core logic driven by the input portion of the bidirectional buffer will be set to the hold state.*

- 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].
seconday_filter_instance/Latch_For_Clock_Gating:Q
Top/ff1/ff_1_wrapper_inst/user_ff_1_wrapper/genblk1.genblk2.secondary_filter[1].
seconday_filter_instance/Latch_For_Clock_Gating:Q
```

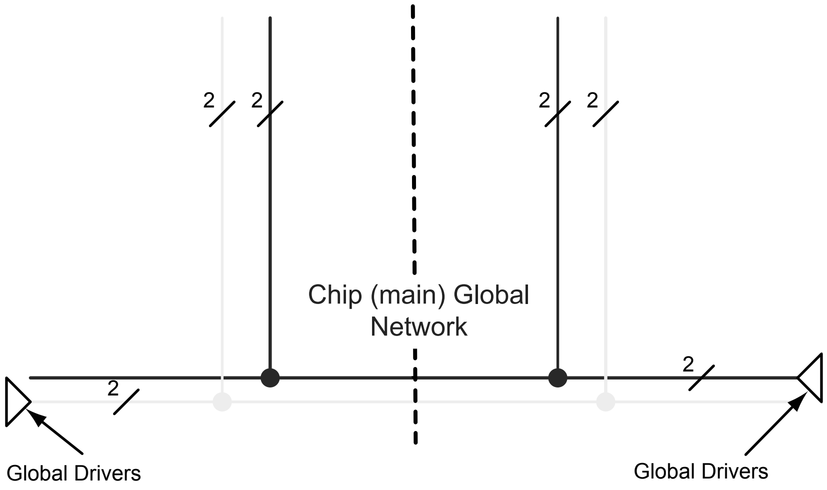


Figure 3-2 • **Simplified VersaNet Global Network (30 k gates and below)**

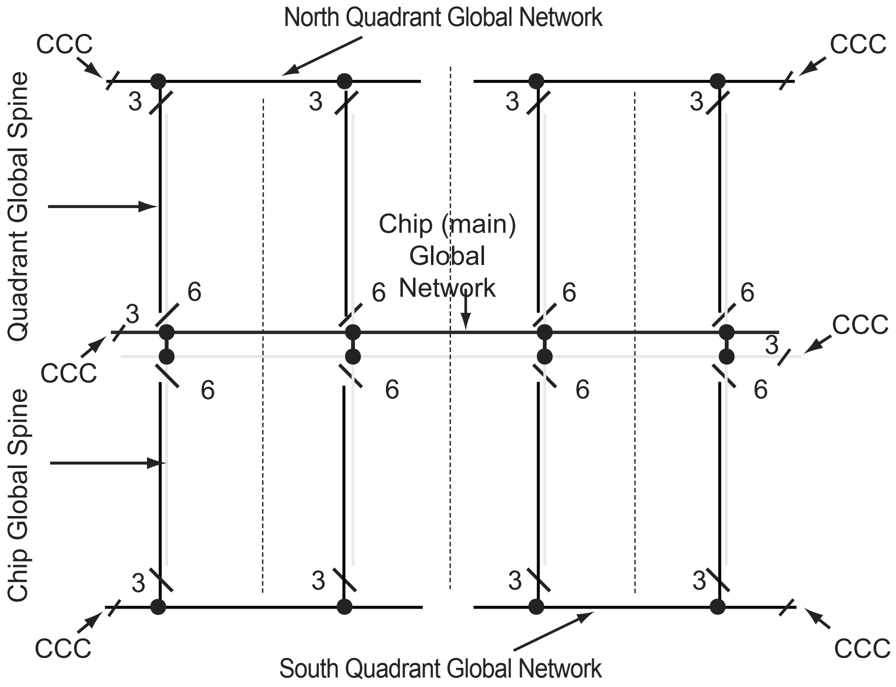


Figure 3-3 • **Simplified VersaNet Global Network (60 k gates and above)**

Figure 3-6 shows all nine global inputs for the location A connected to the top left quadrant global network via CCC.

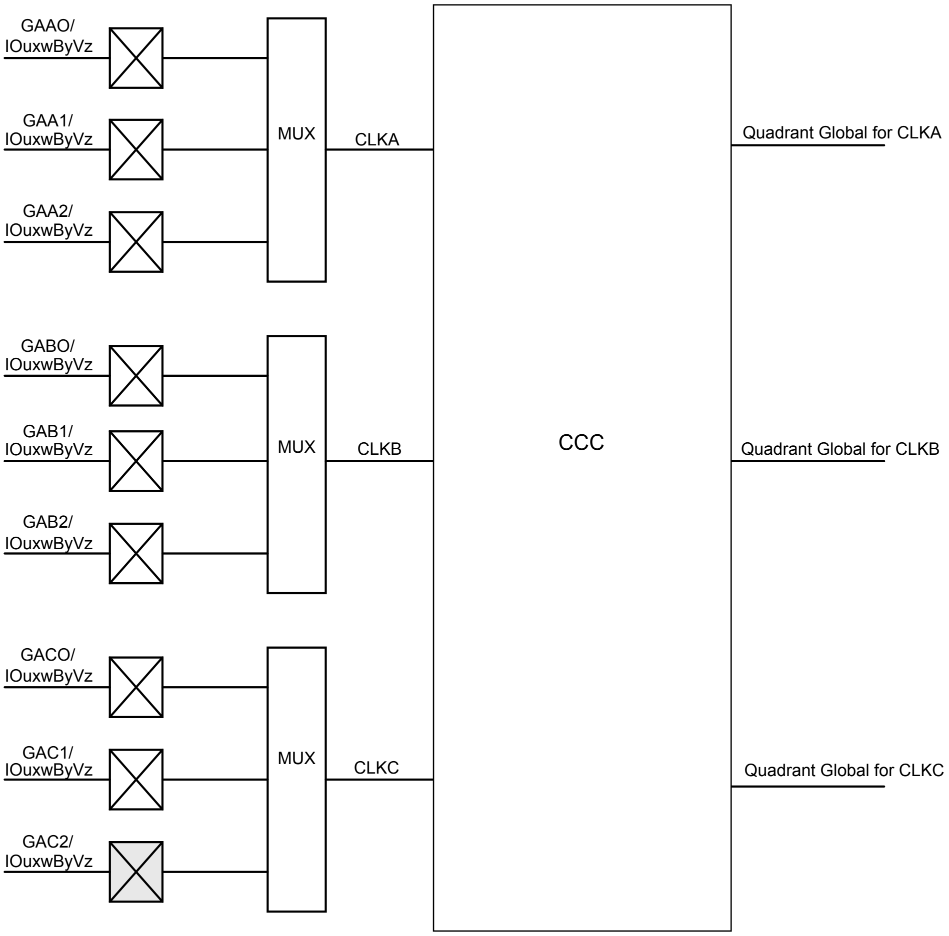


***Figure 3-6* • Global Inputs**

Since each bank can have a different I/O standard, the user should be careful to choose the correct global I/O for the design. There are 54 global pins available to access 18 global networks. For the single-ended and voltage-referenced I/O standards, you can use any of these three available I/Os to access the global network. For differential I/O standards such as LVDS and LVPECL, the I/O macro needs to be placed on (A0, A1), (B0, B1), (C0, C1), or a similar location. The unassigned global I/Os can be used as regular I/Os. Note that pin names starting with GF and GC are associated with the chip global networks, and GA, GB, GD, and GE are used for quadrant global networks. Table 3-2 on page 54 and Table 3-3 on page 55 show the general chip and quadrant global pin names.

External I/O or Local signal as Clock Source

External I/O refers to regular I/O pins are labeled with the I/O convention IOuxwByVz. You can allow the external I/O or internal signal to access the global. To allow the external I/O or internal signal to access the global network, you need to instantiate the CLKINT macro. Refer to Figure 3-4 on page 51 for an example illustration of the connections. Instead of using CLKINT, you can also use PDC to promote signals from external I/O or internal signal to the global network. However, it may cause layout issues because of synthesis logic replication. Refer to the "Global Promotion and Demotion Using PDC" section on page 67 for details.

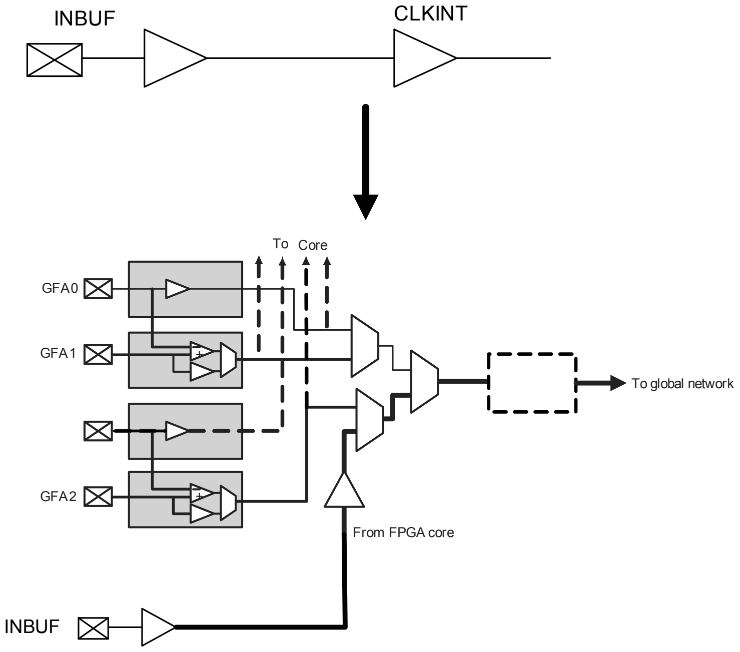


Figure 3-14 • **CLKINT Macro**

Using Global Macros in Synplicity

The Synplify® synthesis tool automatically inserts global buffers for nets with high fanout during synthesis. By default, Synplicity® puts six global macros (CLKBUF or CLKINT) in the netlist, including any global instantiation or PLL macro. Synplify always honors your global macro instantiation. If you have a PLL (only primary output is used) in the design, Synplify adds five more global buffers in the netlist. Synplify uses the following global counting rule to add global macros in the netlist:

1. CLKBUF: 1 global buffer
2. CLKINT: 1 global buffer
3. CLKDLY: 1 global buffer
4. PLL: 1 to 3 global buffers
   - GLA, GLB, GLC, YB, and YC are counted as 1 buffer.
   - GLB or YB is used or both are counted as 1 buffer.
   - GLC or YC is used or both are counted as 1 buffer.

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

Config. Bits	**Signal**	**Name**	**Description**
<31:29>	OAMUX[2:0]	GLA Output Select	Selects from the VCO's four phase outputs for GLA.
<28:24>	OCDIV[4:0]	Secondary 2 Output Divider	Sets the divider value for the GLC/YC outputs. Also known as divider *w* in Figure 4-20 on page 101. The divider value will be OCDIV[4:0] + 1.
<23:19>	OBDIV[4:0]	Secondary 1 Output Divider	Sets the divider value for the GLB/YB outputs. Also known as divider *v* in Figure 4-20 on page 101. The divider value will be OBDIV[4:0] + 1.
<18:14>	OADIV[4:0]	Primary Output Divider	Sets the divider value for the GLA output. Also known as divider *u* in Figure 4-20 on page 101. The divider value will be OADIV[4:0] + 1.
<13:7>	FBDIV[6:0]	Feedback Divider	Sets the divider value for the PLL core feedback. Also known as divider *m* in Figure 4-20 on page 101. The divider value will be FBDIV[6:0] + 1.
<6:0>	FINDIV[6:0]	Input Divider	Input Clock Divider (/n). Sets the divider value for the input delay on CLKA. The divider value will be FINDIV[6:0] + 1.

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.*

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

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

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

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

DINA and DINB

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

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

DOUTA and DOUTB

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

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

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

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

RAM512X18 Macro

RAM512X18 is the two-port configuration of the same RAM block (Figure 6-5 on page 156). Like the RAM4K9 nomenclature, the RAM512X18 nomenclature refers to both the deepest possible configuration and the widest possible configuration the two-port RAM block can assume. In two-port mode, the RAM block can be configured to either the 512×9 aspect ratio or the 256×18 aspect ratio. RAM512X18 is also fully synchronous and has the following features:

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

Table 6-8 and Table 6-9 show the maximum potential width and depth configuration for each device. Note that 15 k and 30 k gate devices do not support RAM or FIFO.

Table 6-8 **• Memory Availability per IGLOO and ProASIC3 Device**

Device		**RAM Blocks**	**Maximum Potential Width**[1]		**Maximum Potential Depth**[2]	
IGLOO IGLOO nano IGLOO PLUS	**ProASIC3 ProASIC3 nano ProASIC3L**		**Depth**	**Width**	**Depth**	**Width**
AGL060 AGLN060 AGLP060	A3P060 A3PN060	4	256	72 (4×18)	16,384 (4,096×4)	1
AGL125 AGLN125 AGLP125	A3P125 A3PN125	8	256	144 (8×18)	32,768 (4,094×8)	1
AGL250 AGLN250	A3P250/L A3PN250	8	256	144 (8×18)	32,768 (4,096×8)	1
AGL400	A3P400	12	256	216 (12×18)	49,152 (4,096×12)	1
AGL600	A3P600/L	24	256	432 (24×18)	98,304 (4,096×24)	1
AGL1000	A3P1000/L	32	256	576 (32×18)	131,072 (4,096×32)	1
AGLE600	A3PE600	24	256	432 (24×18)	98,304 (4,096×24)	1
	A3PE1500	60	256	1,080 (60×18)	245,760 (4,096×60)	1
AGLE3000	A3PE3000/L	112	256	2,016 (112×18)	458,752 (4,096×112)	1

Notes:

1. *Maximum potential width uses the two-port configuration.*
2. *Maximum potential depth uses the dual-port configuration.*

Table 6-9 **• Memory Availability per Fusion Device**

Device	**RAM Blocks**	**Maximum Potential Width**[1]		**Maximum Potential Depth**[2]	
		Depth	**Width**	**Depth**	**Width**
AFS090	6	256	108 (6×18)	24,576 (4,094×6)	1
AFS250	8	256	144 (8×18)	32,768 (4,094×8)	1
AFS600	24	256	432 (24×18)	98,304 (4,096×24)	1
AFS1500	60	256	1,080 (60×18)	245,760 (4,096×60)	1

Notes:

1. *Maximum potential width uses the two-port configuration.*
2. *Maximum potential depth uses the dual-port configuration.*

The ROM emulation application is based on RAM block initialization. If the user's main design has access only to the read ports of the RAM block (RADDR, RD, RCLK, and REN), and the contents of the RAM are already initialized through the TAP, then the memory blocks will emulate ROM functionality for the core design. In this case, the write ports of the RAM blocks are accessed only by the user interface block, and the interface is activated only by the TAP Instruction Register contents.

Users should note that the contents of the RAM blocks are lost in the absence of applied power. However, the 1 kbit of flash memory, FlashROM, in low power flash devices can be used to retain data after power is removed from the device. Refer to the "SRAM and FIFO Memories in Microsemi's Low Power Flash Devices" section on page 147 for more information.

Sample Verilog Code

Interface Block

```
`define Initialize_start 8'h22 //INITIALIZATION START COMMAND VALUE
`define Initialize_stop 8'h23 //INITIALIZATION START COMMAND VALUE

module interface(IR, rst_n, data_shift, clk_in, data_update, din_ser, dout_ser, test,
  test_out,test_clk,clk_out,wr_en,rd_en,write_word,read_word,rd_addr, wr_addr);

input [7:0] IR;
input [3:0] read_word; //RAM DATA READ BACK
input rst_n, data_shift, clk_in, data_update, din_ser; //INITIALIZATION SIGNALS
input test, test_clk; //TEST PROCEDURE CLOCK AND COMMAND INPUT
output [3:0] test_out; //READ DATA
output [3:0] write_word; //WRITE DATA
output [1:0] rd_addr; //READ ADDRESS
output [1:0] wr_addr; //WRITE ADDRESS
output dout_ser; //TDO DRIVER
output clk_out, wr_en, rd_en;

wire [3:0] write_word;
wire [1:0] rd_addr;
wire [1:0] wr_addr;
wire [3:0] Q_out;
wire enable, test_active;

reg clk_out;

//SELECT CLOCK FOR INITIALIZATION OR READBACK TEST
always @(enable or test_clk or data_update)
begin
  case ({test_active})
    1 : clk_out = test_clk ;
    0 : clk_out = !data_update;
    default : clk_out = 1'b1;
  endcase
end

assign test_active = test && (IR == 8'h23);
assign enable = (IR == 8'h22);
assign wr_en = !enable;
assign rd_en = !test_active;
assign test_out = read_word;
assign dout_ser = Q_out[3];

//4-bit SIN/POUT SHIFT REGISTER
shift_reg data_shift_reg (.Shiften(data_shift), .Shiftin(din_ser), .Clock(clk_in),
  .Q(Q_out));

//4-bit PIPELINE REGISTER
D_pipeline pipeline_reg (.Data(Q_out), .Clock(data_update), .Q(write_word));
```

I/O Architecture

I/O Tile

The I/O tile provides a flexible, programmable structure for implementing a large number of I/O standards. In addition, the registers available in the I/O tile can be used to support high-performance register inputs and outputs, with register enable if desired (Figure 7-2). The registers can also be used to support the JESD-79C Double Data Rate (DDR) standard within the I/O structure (see the "DDR for Microsemi's Low Power Flash Devices" section on page 271 for more information). In addition, the registers available in the I/O tile can be used to support high-performance register inputs and outputs, with register enable if desired (Figure 7-2).

As depicted in Figure 7-2, all I/O registers share one CLR port. The output register and output enable register share one CLK port.

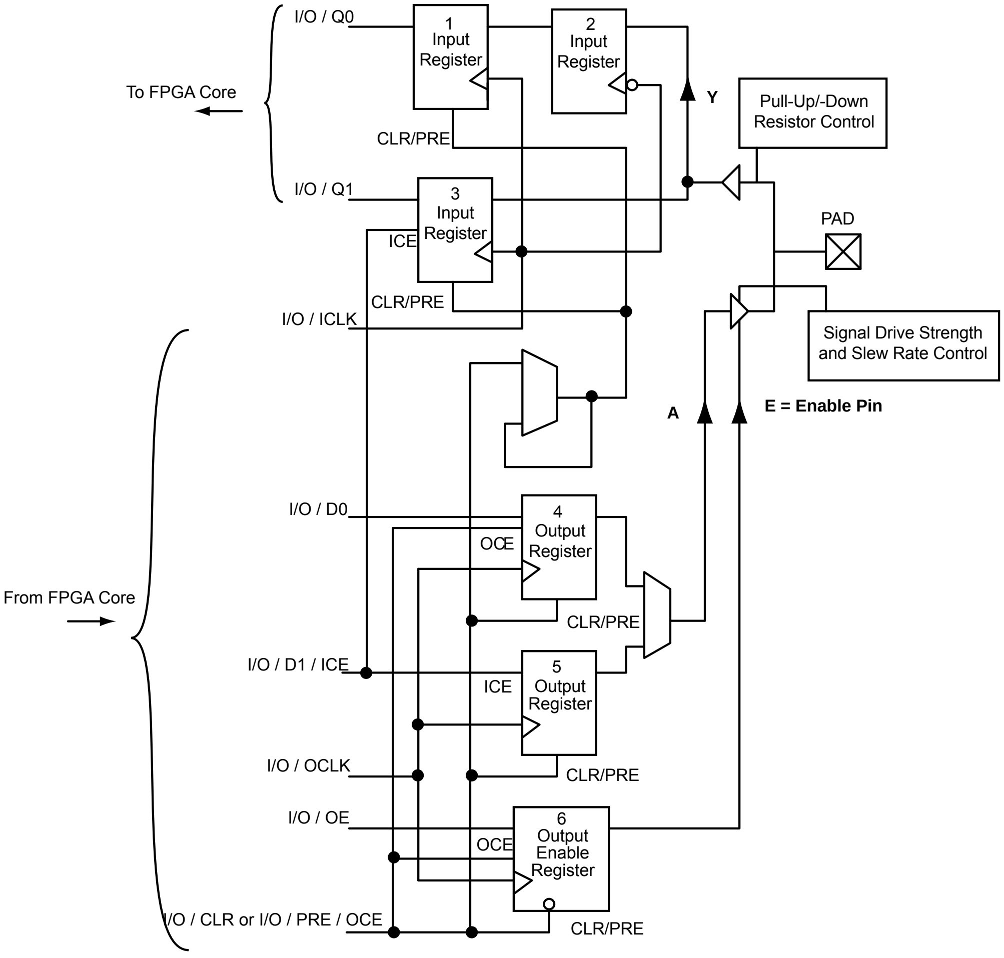


Figure 7-2 • **DDR Configured I/O Block Logical Representation**

9 – I/O Software Control in Low Power Flash Devices

Fusion, IGLOO, and ProASIC3 I/Os provide more design flexibility, allowing the user to control specific features by enabling certain I/O standards. Some features are selectable only for certain I/O standards, whereas others are available for all I/O standards. For example, slew control is not supported by differential I/O standards. Conversely, I/O register combining is supported by all I/O standards. For detailed information about which I/O standards and features are available on each device and each I/O type, refer to the I/O Structures section of the handbook for the device you are using.

Figure 9-1 shows the various points in the software design flow where a user can provide input or control of the I/O selection and parameters. A detailed description is provided throughout this document.

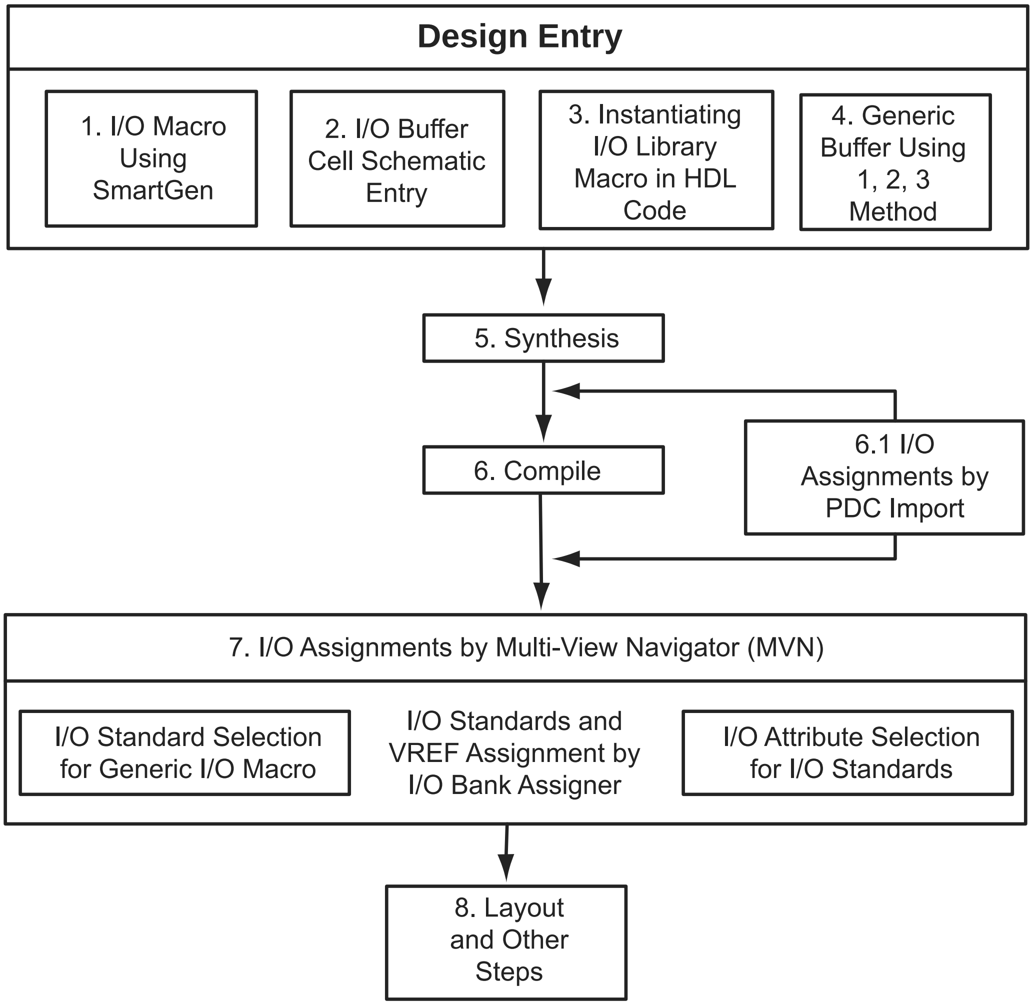


Figure 9-1 **• User I/O Assignment Flow Chart**

Output Buffers

There are two variations: Regular and Special.

If the **Regular** variation is selected, only the Width (1 to 128) needs to be entered. The default value for Width is 1.

The **Special** variation has Width, Technology, Output Drive, and Slew Rate options.

Bidirectional Buffers

There are two variations: Regular and Special.

The **Regular** variation has Enable Polarity (Active High, Active Low) in addition to the Width option.

The **Special** variation has Width, Technology, Output Drive, Slew Rate, and Resistor Pull-Up/-Down options.

Tristate Buffers

Same as Bidirectional Buffers.

DDR

There are eight variations: DDR with Regular Input Buffers, Special Input Buffers, Regular Output Buffers, Special Output Buffers, Regular Tristate Buffers, Special Tristate Buffers, Regular Bidirectional Buffers, and Special Bidirectional Buffers.

These variations resemble the options of the previous I/O macro. For example, the Special Input Buffers variation has Width, Technology, Voltage Level, and Resistor Pull-Up/-Down options. DDR is not available on IGLOO PLUS devices.

4. Once the desired configuration is selected, click the **Generate** button. The Generate Core window opens (Figure 9-4).
5. Enter a name for the macro. Click **OK**. The core will be generated and saved to the appropriate location within the project files (Figure 9-5 on page 257).

Figure 9-4 • **Generate Core Window**

6. Instantiate the I/O macro in the top-level code.

   The user must instantiate the DDR_REG or DDR_OUT macro in the design. Use SmartGen to generate both these macros and then instantiate them in your top level. To combine the DDR macros with the I/O, the following rules must be met:

If the assignment is not successful, an error message appears in the Output window.

To undo the I/O bank assignments, choose **Undo** from the **Edit** menu. Undo removes the I/O technologies assigned by the IOBA. It does not remove the I/O technologies previously assigned.

To redo the changes undone by the Undo command, choose **Redo** from the **Edit** menu.

To clear I/O bank assignments made before using the Undo command, manually unassign or reassign I/O technologies to banks. To do so, choose **I/O Bank Settings** from the **Edit** menu to display the I/O Bank Settings dialog box.

Conclusion

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

Related Documents

User's Guides

Libero SoC User's Guide

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

IGLOO, ProASIC3, SmartFusion, and Fusion Macro Library Guide

http://www.microsemi.com/soc/documents/pa3_libguide_ug.pdf

SmartGen Core Reference Guide

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

DDR Output Register

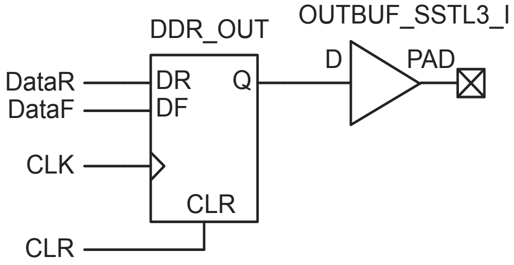


Figure 10-6 **• DDR Output Register (SSTL3 Class I)**

Verilog

```
module DDR_OutBuf_SSTL3_I(DataR,DataF,CLR,CLK,PAD);

input   DataR, DataF, CLR, CLK;
output  PAD;

wire Q, VCC;

  VCC VCC_1_net(.Y(VCC));
  DDR_OUT DDR_OUT_0_inst(.DR(DataR),.DF(DataF),.CLK(CLK),.CLR(CLR),.Q(Q));
  OUTBUF_SSTL3_I OUTBUF_SSTL3_I_0_inst(.D(Q),.PAD(PAD));

endmodule
```

VHDL

```
library ieee;
use ieee.std_logic_1164.all;
library proasic3; use proasic3.all;

entity DDR_OutBuf_SSTL3_I is
  port(DataR, DataF, CLR, CLK : in std_logic;  PAD : out std_logic) ;
end DDR_OutBuf_SSTL3_I;

architecture DEF_ARCH of  DDR_OutBuf_SSTL3_I is

  component DDR_OUT
    port(DR, DF, CLK, CLR : in std_logic := 'U'; Q : out std_logic) ;
  end component;

  component OUTBUF_SSTL3_I
    port(D : in std_logic := 'U'; PAD : out std_logic) ;
  end component;

  component VCC
    port( Y : out std_logic);
  end component;

signal Q, VCC_1_net : std_logic ;

begin

  VCC_2_net : VCC port map(Y => VCC_1_net);
  DDR_OUT_0_inst : DDR_OUT
  port map(DR => DataR, DF => DataF, CLK => CLK, CLR => CLR, Q => Q);
  OUTBUF_SSTL3_I_0_inst : OUTBUF_SSTL3_I
  port map(D => Q, PAD => PAD);

end DEF_ARCH;
```

Programming Voltage (VPUMP) and VJTAG

Low-power flash devices support on-chip charge pumps, and therefore require only a single 3.3 V programming voltage for the VPUMP pin during programming. When the device is not being programmed, the VPUMP pin can be left floating or can be tied (pulled up) to any voltage between 0 V and 3.6 V[2]. During programming, the target board or the FlashPro4/3/3X programmer can provide VPUMP. FlashPro4/3/3X is capable of supplying VPUMP to a single device. If more than one device is to be programmed using FlashPro4/3/3X on a given board, FlashPro4/3/3X should not be relied on to supply the VPUMP voltage. A FlashPro4/3/3X programmer is not capable of providing reliable VJTAG voltage. The board must supply VJTAG voltage to the device and the VJTAG pin of the programmer header must be connected to the device VJTAG pin. Microsemi recommends that VPUMP[3] and VJTAG power supplies be kept separate with independent filtering capacitors rather than supplying them from a common rail. Refer to the "Board-Level Considerations" section on page 337 for capacitor requirements.

Low power flash device I/Os support a bank-based, voltage-supply architecture that simultaneously supports multiple I/O voltage standards (Table 13-2). By isolating the JTAG power supply in a separate bank from the user I/Os, low power flash devices provide greater flexibility with supply selection and simplify power supply and printed circuit board (PCB) design. The JTAG pins can be run at any voltage from 1.5 V to 3.3 V (nominal). Microsemi recommends that TCK be tied to GND through a 200 ohm to 1 Kohm resistor. This prevents a possible totempole current on the input buffer stage. For TDI, TMS, and TRST pins, the devices provide an internal nominal 10 Kohm pull-up resistor. During programming, all I/O pins, except for JTAG interface pins, are tristated and weakly pulled up to VCCI. This isolates the part and prevents the signals from floating. The JTAG interface pins are driven by the FlashPro4/3/3X during programming, including the TRST pin, which is driven HIGH.

***Table 13-2* • Power Supplies**

Power Supply	Programming Mode	Current during Programming
VCC	1.2 V / 1.5 V	< 70 mA
VCCI	1.2 V / 1.5 V / 1.8 V / 2.5 V / 3.3 V (bank-selectable)	I/Os are weakly pulled up.
VJTAG	1.2 V / 1.5 V / 1.8 V / 2.5 V / 3.3 V	< 20 mA
VPUMP	3.15 V to 3.45 V	< 80 mA

Note: *All supply voltages should be at 1.5 V or higher, regardless of the setting during normal operation, except for IGLOO nano, where 1.2 V VCC and VJTAG programming is allowed.*

Nonvolatile Memory (NVM) Programming Voltage

SmartFusion and Fusion devices need stable VCCNVM/VCCENVM[3] (1.5 V power supply to the embedded nonvolatile memory blocks) and VCCOSC/VCCROSC[4] (3.3 V power supply to the integrated RC oscillator). The tolerance of VCCNVM/VCCENVM is ± 5% and VCCOSC/VCCROSC is ± 5%.

Unstable supply voltage on these pins can cause an NVM programming failure due to NVM page corruption. The NVM page can also be corrupted if the NVM reset pin has noise. This signal must be tied off properly.

Microsemi recommends installing the following capacitors[5] on the VCCNVM/VCCENVM and VCCOSC/VCCROSC pins:

- Add one bypass capacitor of 10 μF for each power supply plane followed by an array of decoupling capacitors of 0.1 μF.
- Add one 0.1 μF capacitor near each pin.

---

[2] *During sleep mode in IGLOO devices connect VPUMP to GND.*
[3] *VPUMP has to be quiet for successful programming. Therefore VPUMP must be separate and required capacitors must be installed close to the FPGA VPUMP pin.*
[4] *VCCROSC is for SmartFusion.*
[5] *The capacitors cannot guarantee reliable operation of the device if the board layout is not done properly.*

Circuit Description

All IGLOO devices as well as the ProASIC3L product family are available in two versions: V5 devices, which are powered by a 1.5 V supply and V2 devices, which are powered by a supply anywhere in the range of 1.2 V to 1.5 V in 50 mV increments. Applications that use IGLOO or ProASIC3L devices powered by a 1.2 V core supply must have a mechanism that switches the core voltage from 1.2 V (or other voltage below 1.5 V) to 1.5 V during in-system programming (ISP). There are several possible techniques to meet this requirement. Microsemi recommends utilizing a linear voltage regulator, a resistor voltage divider, and an N-Channel Digital FET to set the appropriate VCC voltage, as shown in Figure 14-1.

Where 1.2 V is mentioned in the following text, the meaning applies to any voltage below the 1.5 V range. Resistor values in the figures have been calculated for 1.2 V, so refer to power regulator datasheets if a different core voltage is required.

The main component of Microsemi's recommended circuit is the LTC3025 linear voltage regulator from LinearTech. The output voltage of the LTC3025 on the OUT pin is set by the ratio of two external resistors, R37 and R38, in a voltage divider. The linear voltage regulator adjusts the voltage on the OUT pin to maintain the ADJ pin voltage at 0.4 V (referenced to ground). By using an R38 value of 40.2 kΩ and an R37 value of 80.6 kΩ, the output voltage on the OUT pin is 1.2 V. To achieve 1.5 V on the OUT pin, R44 can be used in parallel with R38. The OUT pin can now be used as a switchable source for the VCC supply. Refer to the *LTC3025 Linear Voltage Regulator datasheet* for more information.

In Figure 14-1, the N-Channel Digital FET is used to enable and disable R44. This FET is controlled by the JTAG TRST signal driven by the FlashPro3 programmer. During programming of the device, the TRST signal is driven HIGH by the FlashPro3, and turns the N-Channel Digital FET ON. When the FET is ON, R44 becomes enabled as a parallel resistance to R38, which forces the regulator to set OUT to 1.5 V.

When the FlashPro3 is connected and not in programming mode or when it is not connected, the pull-down resistor, R10, will pull the TRST signal LOW. When this signal is LOW, the N-Channel Digital FET is "open" and R44 is not part of the resistance seen by the LTC3025. The new resistance momentarily changes the voltage value on the ADJ pin, which in turn causes the output of the LTC3025 to compensate by setting OUT to 1.2 V. Now the device will run in regular active mode at the regular 1.2 V core voltage.

***Figure 14-1* • Circuit Diagram**

Circuit Verification

The power switching circuit recommended above is implemented on Microsemi's Icicle board (Figure 14-2). On the Icicle board, VJTAGENB is used to control the N-Channel Digital FET; however, this circuit was modified to use TRST instead of VJTAGENB in this application. There are three important aspects of this circuit that were verified:

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

Verification Steps

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

***Figure 14-2* • Core Voltage on the IGLOO AGL125-QNG132 Device**

In the oscilloscope plots (Figure 14-2), the TRST from FlashPro3 and the VCC core voltage of the IGLOO device are labeled. This plot shows the rise characteristic of the TRST signal from FlashPro3. Once the TRST signal is asserted HIGH, the LTC3025 shown in Figure 14-1 on page 343 senses the increase in voltage and changes the output from 1.2 V to 1.5 V. It takes the circuit approximately 100 μs to respond to TRST and change the voltage to 1.5 V on the VCC core.

Internal Pull-Up and Pull-Down

Low power flash device I/Os are equipped with internal weak pull-up/-down resistors that can be used by designers. If used, these internal pull-up/-down resistors will be activated during power-up, once both VCC and VCCI are above their functional activation level. Similarly, during power-down, these internal pull-up/-down resistors will turn off once the first supply voltage falls below its brownout deactivation level.

Cold-Sparing

In cold-sparing applications, voltage can be applied to device I/Os before and during power-up. Cold-sparing applications rely on three important characteristics of the device:

1. I/Os must be tristated before and during power-up.
2. Voltage applied to the I/Os must not power up any part of the device.
3. VCCI should not exceed 3.6 V, per datasheet specifications.

As described in the "Power-Up to Functional Time" section on page 378, Microsemi's low power flash I/Os are tristated before and during power-up until the last voltage supply (VCC or VCCI) is powered up past its functional level. Furthermore, applying voltage to the FPGA I/Os does not pull up VCC or VCCI and, therefore, does not partially power up the device. Table 18-4 includes the cold-sparing test results on A3PE600-PQ208 devices. In this test, leakage current on the device I/O and residual voltage on the power supply rails were measured while voltage was applied to the I/O before power-up.

***Table 18-4* • Cold-Sparing Test Results for A3PE600 Devices**

Device I/O	Residual Voltage (V)		Leakage Current
	VCC	VCCI	
Input	0	0.003	<1 µA
Output	0	0.003	<1 µA

VCCI must not exceed 3.6 V, as stated in the datasheet specification. Therefore, ProASIC3E devices meet all three requirements stated earlier in this section and are suitable for cold-sparing applications.

The following devices and families support cold-sparing:

- IGLOO: AGL015 and AGL030
- All IGLOO nano
- All IGLOO PLUS
- All IGLOOe
- ProASIC3L: A3PE3000L
- ProASIC3: A3P015 and A3P030
- All ProASIC3 nano
- All ProASIC3E
- Military ProASIC3EL: A3PE600L and A3PE3000L
- RT ProASIC3: RT3PE600L and RT3PE3000L

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

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