



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



# Fusion Family of Mixed Signal FPGAs

## Features and Benefits

### High-Performance Reprogrammable Flash Technology

- Advanced 130-nm, 7-Layer Metal, Flash-Based CMOS Process
- Nonvolatile, Retains Program when Powered Off
- Instant On Single-Chip Solution
- 350 MHz System Performance

### Embedded Flash Memory

- User Flash Memory – 2 Mbits to 8 Mbits
  - Configurable 8-, 16-, or 32-Bit Datapath
  - 10 ns Access in Read-Ahead Mode
- 1 Kbit of Additional FlashROM

### Integrated A/D Converter (ADC) and Analog I/O

- Up to 12-Bit Resolution and up to 600 Ksps
- Internal 2.56 V or External Reference Voltage
- ADC: Up to 30 Scalable Analog Input Channels
- High-Voltage Input Tolerance: –10.5 V to +12 V
- Current Monitor and Temperature Monitor Blocks
- Up to 10 MOSFET Gate Driver Outputs
  - P- and N-Channel Power MOSFET Support
  - Programmable 1, 3, 10, 30  $\mu$ A, and 20 mA Drive Strengths
- ADC Accuracy is Better than 1%

### On-Chip Clocking Support

- Internal 100 MHz RC Oscillator (accurate to 1%)
- Crystal Oscillator Support (32 KHz to 20 MHz)
- Programmable Real-Time Counter (RTC)
- 6 Clock Conditioning Circuits (CCCs) with 1 or 2 Integrated PLLs
  - Phase Shift, Multiply/Divide, and Delay Capabilities
  - Frequency: Input 1.5–350 MHz, Output 0.75–350 MHz

### Low Power Consumption

- Single 3.3 V Power Supply with On-Chip 1.5 V Regulator
- Sleep and Standby Low-Power Modes

### In-System Programming (ISP) and Security

- ISP with 128-Bit AES via JTAG
- FlashLock<sup>®</sup> Designed to Protect FPGA Contents

### Advanced Digital I/O

- 1.5 V, 1.8 V, 2.5 V, and 3.3 V Mixed-Voltage Operation
- Bank-Selectable I/O Voltages – Up to 5 Banks per Chip
- Single-Ended I/O Standards: LVTTTL, LVCMOS 3.3 V / 2.5 V / 1.8 V / 1.5 V, 3.3 V PCI / 3.3 V PCI-X, and LVCMOS 2.5 V / 5.0 V Input
- Differential I/O Standards: LVPECL, LVDS, B-LVDS, M-LVDS
  - Built-In I/O Registers
  - 700 Mbps DDR Operation
- Hot-Swappable I/Os
- Programmable Output Slew Rate, Drive Strength, and Weak Pull-Up/Down Resistor
- Pin-Compatible Packages across the Fusion<sup>®</sup> Family

### SRAMs and FIFOs

- Variable-Aspect-Ratio 4,608-Bit SRAM Blocks ( $\times 1$ ,  $\times 2$ ,  $\times 4$ ,  $\times 9$ , and  $\times 18$  organizations available)
- True Dual-Port SRAM (except  $\times 18$ )
- Programmable Embedded FIFO Control Logic

### Soft ARM Cortex-M1 Fusion Devices (M1)

- ARM<sup>®</sup> Cortex-<sup>™</sup>M1-Enabled

### Pigeon Point ATCA IP Support (P1)

- Targeted to Pigeon Point<sup>®</sup> Board Management Reference (BMR) Starter Kits
- Designed in Partnership with Pigeon Point Systems
- ARM Cortex-M1 Enabled

### MicroBlade Advanced Mezzanine Card Support (U1)

- Targeted to Advanced Mezzanine Card (AdvancedMC<sup>™</sup> Designs)
- Designed in Partnership with MicroBlade
- 8051-Based Module Management Controller (MMC)

**Table 1 • Fusion Family**

Fusion Devices		AFS090	AFS250	AFS600	AFS1500
ARM Cortex-M1 <sup>*</sup> Devices			M1AFS250	M1AFS600	M1AFS1500
Pigeon Point Devices				P1AFS600	P1AFS1500
MicroBlade Devices			U1AFS250	U1AFS600	U1AFS1500
General Information	System Gates	90,000	250,000	600,000	1,500,000
	Tiles (D-flip-flops)	2,304	6,144	13,824	38,400
	Secure (AES) ISP	Yes	Yes	Yes	Yes
	PLLs	1	1	2	2
	Globals	18	18	18	18
Memory	Flash Memory Blocks (2 Mbits)	1	1	2	4
	Total Flash Memory Bits	2M	2M	4M	8M
	FlashROM Bits	1,024	1,024	1,024	1,024
	RAM Blocks (4,608 bits)	6	8	24	60
	RAM kbits	27	36	108	270
Analog and I/Os	Analog Quads	5	6	10	10
	Analog Input Channels	15	18	30	30
	Gate Driver Outputs	5	6	10	10
	I/O Banks (+ JTAG)	4	4	5	5
	Maximum Digital I/Os	75	114	172	252
	Analog I/Os	20	24	40	40

*Note:* \*Refer to the [Cortex-M1 product brief](#) for more information.

## Fusion Device Architecture Overview

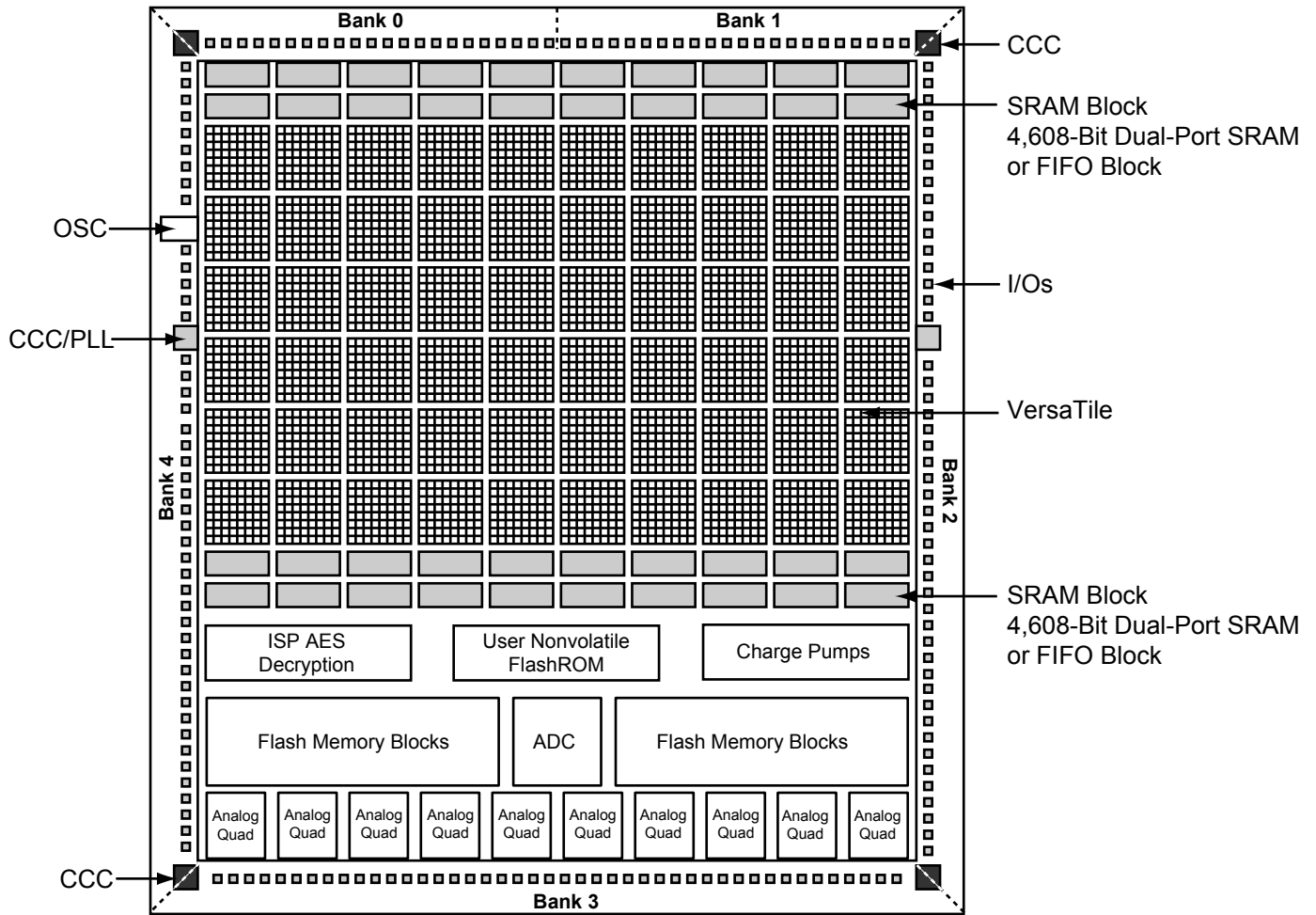


Figure 1 • Fusion Device Architecture Overview (AFS600)

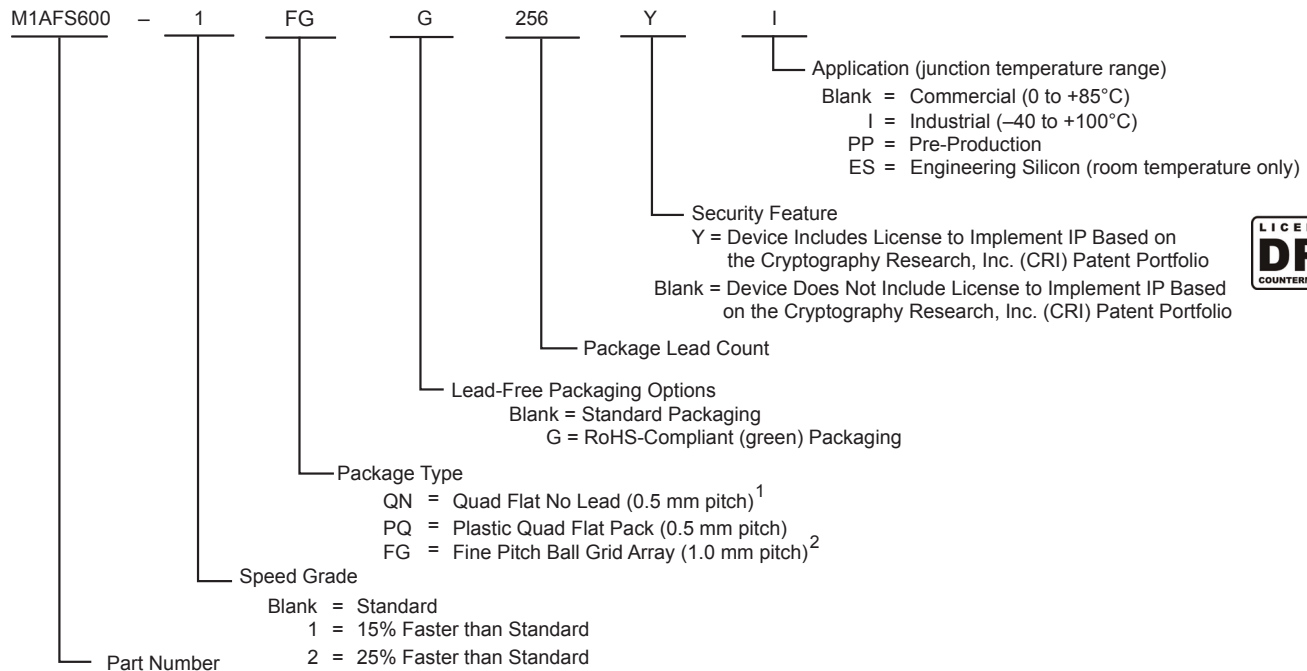
### Package I/Os: Single-/Double-Ended (Analog)

Fusion Devices	AFS090	AFS250	AFS600	AFS1500
ARM Cortex-M1 Devices		M1AFS250	M1AFS600	M1AFS1500
Pigeon Point Devices			P1AFS600 <sup>1</sup>	P1AFS1500 <sup>1</sup>
MicroBlade Devices		U1AFS250 <sup>2</sup>	U1AFS600 <sup>2</sup>	U1AFS1500 <sup>2</sup>
QN108 <sup>3</sup>	37/9 (16)			
QN180 <sup>3</sup>	60/16 (20)	65/15 (24)		
PQ208 <sup>4</sup>		93/26 (24)	95/46 (40)	
FG256	75/22 (20)	114/37 (24)	119/58 (40)	119/58 (40)
FG484			172/86 (40)	223/109 (40)
FG676				252/126 (40)

**Notes:**

1. Pigeon Point devices are only offered in FG484 and FG256.
2. MicroBlade devices are only offered in FG256.
3. Package not available.
4. Fusion devices in the same package are pin compatible with the exception of the PQ208 package (AFS250 and AFS600).

## Product Ordering Codes



### Fusion Devices

AFS090 = 90,000 System Gates  
 AFS250 = 250,000 System Gates  
 AFS600 = 600,000 System Gates  
 AFS1500 = 1,500,000 System Gates

### ARM-Enabled Fusion Devices

M1AFS250 = 250,000 System Gates  
 M1AFS600 = 600,000 System Gates  
 M1AFS1500 = 1,500,000 System Gates

### Pigeon Point Devices

P1AFS600 = 600,000 System Gates  
 P1AFS1500 = 1,500,000 System Gates

### MicroBlade Devices

U1AFS250 = 250,000 System Gates  
 U1AFS600 = 600,000 System Gates  
 U1AFS1500 = 1,500,000 System Gates

### Notes:

- For Fusion devices, Quad Flat No Lead packages are only offered as RoHS compliant, QNG packages.
- MicroBlade and Pigeon Point devices only support FG packages.

## Fusion Device Status

Fusion	Status	Cortex-M1	Status	Pigeon Point	Status	MicroBlade	Status
AFS090	Production						
AFS250	Production	M1AFS250	Production			U1AFS250	Production
AFS600	Production	M1AFS600	Production	P1AFS600	Production	U1AFS600	Production
AFS1500	Production	M1AFS1500	Production	P1AFS1500	Production	U1AFS1500	Production

## Temperature Grade Offerings

Fusion Devices	AFS090	AFS250	AFS600	AFS1500
ARM Cortex-M1 Devices		M1AFS250	M1AFS600	M1AFS1500
Pigeon Point Devices			P1AFS600 <sup>3</sup>	P1AFS1500 <sup>3</sup>
MicroBlade Devices		U1AFS250 <sup>4</sup>	U1AFS600 <sup>4</sup>	U1AFS1500 <sup>4</sup>
QN108 <sup>5</sup>	C, I	–	–	–
QN180 <sup>5</sup>	C, I	C, I	–	–
PQ208	–	C, I	C, I	–
FG256	C, I	C, I	C, I	C, I
FG484	–	–	C, I	C, I
FG676	–	–	–	C, I

**Notes:**

1. C = Commercial Temperature Range: 0°C to 85°C Junction
2. I = Industrial Temperature Range: –40°C to 100°C Junction
3. Pigeon Point devices are only offered in FG484 and FG256.
4. MicroBlade devices are only offered in FG256.
5. Package not available.

## Speed Grade and Temperature Grade Matrix

	Std. <sup>1</sup>	–1	–2 <sup>2</sup>
C <sup>3</sup>	✓	✓	✓
I <sup>4</sup>	✓	✓	✓

**Notes:**

1. MicroBlade devices are only offered in standard speed grade.
2. Pigeon Point devices are only offered in –2 speed grade.
3. C = Commercial Temperature Range: 0°C to 85°C Junction
4. I = Industrial Temperature Range: –40°C to 100°C Junction

Contact your local Microsemi SoC Products Group representative for device availability:

[http://www.microsemi.com/index.php?option=com\\_content&id=137&lang=en&view=article](http://www.microsemi.com/index.php?option=com_content&id=137&lang=en&view=article).

## Cortex-M1, Pigeon Point, and MicroBlade Fusion Device Information

This datasheet provides information for all Fusion (AFS), Cortex-M1 (M1), Pigeon Point (P1), and MicroBlade (U1) devices. The remainder of the document will only list the Fusion (AFS) devices. Please apply relevant information to M1, P1, and U1 devices when appropriate. Please note the following:

- Cortex-M1 devices are offered in the same speed grades and packages as basic Fusion devices.
- Pigeon Point devices are only offered in –2 speed grade and FG484 and FG256 packages.
- MicroBlade devices are only offered in standard speed grade and the FG256 package.

---

# Table of Contents

---

## Fusion Device Family Overview

Introduction .....	1-1
General Description .....	1-1
Unprecedented Integration .....	1-4
Related Documents .....	1-10

## Device Architecture

Fusion Stack Architecture .....	2-1
Core Architecture .....	2-2
Clocking Resources .....	2-18
Real-Time Counter System .....	2-31
Embedded Memories .....	2-39
Analog Block .....	2-76
Analog Configuration MUX .....	2-126
User I/Os .....	2-132
Pin Descriptions .....	2-223
Security .....	2-228

## DC and Power Characteristics

General Specifications .....	3-1
Calculating Power Dissipation .....	3-10
Power Consumption .....	3-32

## Package Pin Assignments

QN108 .....	4-1
QN180 .....	4-3
PQ208 .....	4-7
FG256 .....	4-11
FG484 .....	4-19
FG676 .....	4-27

## Datasheet Information

List of Changes .....	5-1
Datasheet Categories .....	5-17
Safety Critical, Life Support, and High-Reliability Applications Policy .....	5-17

---

# 1 – Fusion Device Family Overview

---

## Introduction

The Fusion mixed signal FPGA satisfies the demand from system architects for a device that simplifies design and unleashes their creativity. As the world's first mixed signal programmable logic family, Fusion integrates mixed signal analog, flash memory, and FPGA fabric in a monolithic device. Fusion devices enable designers to quickly move from concept to completed design and then deliver feature-rich systems to market. This new technology takes advantage of the unique properties of Microsemi flash-based FPGAs, including a high-isolation, triple-well process and the ability to support high-voltage transistors to meet the demanding requirements of mixed signal system design.

Fusion mixed signal FPGAs bring the benefits of programmable logic to many application areas, including power management, smart battery charging, clock generation and management, and motor control. Until now, these applications have only been implemented with costly and space-consuming discrete analog components or mixed signal ASIC solutions. Fusion mixed signal FPGAs present new capabilities for system development by allowing designers to integrate a wide range of functionality into a single device, while at the same time offering the flexibility of upgrades late in the manufacturing process or after the device is in the field. Fusion devices provide an excellent alternative to costly and time-consuming mixed signal ASIC designs. In addition, when used in conjunction with the ARM Cortex-M1 processor, Fusion technology represents the definitive mixed signal FPGA platform.

Flash-based Fusion devices are Instant On. As soon as system power is applied and within normal operating specifications, Fusion devices are working. Fusion devices have a 128-bit flash-based lock and industry-leading AES decryption, used to secure programmed intellectual property (IP) and configuration data. Fusion devices are the most comprehensive single-chip analog and digital programmable logic solution available today.

To support this new ground-breaking technology, Microsemi has developed a series of major tool innovations to help maximize designer productivity. Implemented as extensions to the popular Microsemi Libero<sup>®</sup> System-on-Chip (SoC) software, these new tools allow designers to easily instantiate and configure peripherals within a design, establish links between peripherals, create or import building blocks or reference designs, and perform hardware verification. This tool suite will also add comprehensive hardware/software debug capability as well as a suite of utilities to simplify development of embedded soft-processor-based solutions.

## General Description

The Fusion family, based on the highly successful ProASIC<sup>®</sup>3 and ProASIC3E flash FPGA architecture, has been designed as a high-performance, programmable, mixed signal platform. By combining an advanced flash FPGA core with flash memory blocks and analog peripherals, Fusion devices dramatically simplify system design and, as a result, dramatically reduce overall system cost and board space.

The state-of-the-art flash memory technology offers high-density integrated flash memory blocks, enabling savings in cost, power, and board area relative to external flash solutions, while providing increased flexibility and performance. The flash memory blocks and integrated analog peripherals enable true mixed-mode programmable logic designs. Two examples are using an on-chip soft processor to implement a fully functional flash MCU and using high-speed FPGA logic to offer system and power supervisory capabilities. Instant On, and capable of operating from a single 3.3 V supply, the Fusion family is ideally suited for system management and control applications.

The devices in the Fusion family are categorized by FPGA core density. Each family member contains many peripherals, including flash memory blocks, an analog-to-digital-converter (ADC), high-drive outputs, both RC and crystal oscillators, and a real-time counter (RTC). This provides the user with a high level of flexibility and integration to support a wide variety of mixed signal applications. The flash memory block capacity ranges from 2 Mbits to 8 Mbits. The integrated 12-bit ADC supports up to 30 independently configurable input channels.

The on-chip crystal and RC oscillators work in conjunction with the integrated phase-locked loops (PLLs) to provide clocking support to the FPGA array and on-chip resources. In addition to supporting typical RTC uses such as watchdog timer, the Fusion RTC can control the on-chip voltage regulator to power down the device (FPGA fabric, flash memory block, and ADC), enabling a low power standby mode.

The Fusion family offers revolutionary features, never before available in an FPGA. The nonvolatile flash technology gives the Fusion solution the advantage of being a highly secure, low power, single-chip solution that is Instant On. Fusion is reprogrammable and offers time-to-market benefits at an ASIC-level unit cost. These features enable designers to create high-density systems using existing ASIC or FPGA design flows and tools.

## Flash Advantages

### ***Reduced Cost of Ownership***

Advantages to the designer extend beyond low unit cost, high performance, and ease of use. Flash-based Fusion devices are Instant On and do not need to be loaded from an external boot PROM. On-board security mechanisms prevent access to the programming information and enable remote updates of the FPGA logic that are protected with high level security. Designers can perform remote in-system reprogramming to support future design iterations and field upgrades, with confidence that valuable IP is highly unlikely to be compromised or copied. ISP can be performed using the industry-standard AES algorithm with MAC data authentication on the device. The Fusion family device architecture mitigates the need for ASIC migration at higher user volumes. This makes the Fusion family a cost-effective ASIC replacement solution for applications in the consumer, networking and communications, computing, and avionics markets.

### ***Security***

As the nonvolatile, flash-based Fusion family requires no boot PROM, there is no vulnerable external bitstream. Fusion devices incorporate FlashLock, which provides a unique combination of reprogrammability and design security without external overhead, advantages that only an FPGA with nonvolatile flash programming can offer.

Fusion devices utilize a 128-bit flash-based key lock and a separate AES key to provide the highest level of protection in the FPGA industry for programmed IP and configuration data. The FlashROM data in Fusion devices can also be encrypted prior to loading. Additionally, the flash memory blocks can be programmed during runtime using the industry-leading AES-128 block cipher encryption standard (FIPS Publication 192). The AES standard was adopted by the National Institute of Standards and Technology (NIST) in 2000 and replaces the DES standard, which was adopted in 1977. Fusion devices have a built-in AES decryption engine and a flash-based AES key that make Fusion devices the most comprehensive programmable logic device security solution available today. Fusion devices with AES-based security provide a high level of protection for remote field updates over public networks, such as the Internet, and are designed to ensure that valuable IP remains out of the hands of system overbuilders, system cloners, and IP thieves. As an additional security measure, the FPGA configuration data of a programmed Fusion device cannot be read back, although secure design verification is possible. During design, the user controls and defines both internal and external access to the flash memory blocks.

Security, built into the FPGA fabric, is an inherent component of the Fusion family. The flash cells are located beneath seven metal layers, and many device design and layout techniques have been used to make invasive attacks extremely difficult. Fusion with FlashLock and AES security is unique in being highly resistant to both invasive and noninvasive attacks. Your valuable IP is protected with industry-standard security, making remote ISP possible. A Fusion device provides the best available security for programmable logic designs.

### ***Single Chip***

Flash-based FPGAs store their configuration information in on-chip flash cells. Once programmed, the configuration data is an inherent part of the FPGA structure, and no external configuration data needs to be loaded at system power-up (unlike SRAM-based FPGAs). Therefore, flash-based Fusion FPGAs do not require system configuration components such as EEPROMs or microcontrollers to load device configuration data. This reduces bill-of-materials costs and PCB area, and increases security and system reliability.



### **Instant On**

Flash-based Fusion devices are Level 0 Instant On. Instant On Fusion devices greatly simplify total system design and reduce total system cost by eliminating the need for CPLDs. The Fusion Instant On clocking (PLLs) replaces off-chip clocking resources. The Fusion mix of Instant On clocking and analog resources makes these devices an excellent choice for both system supervisor and system management functions. Instant On from a single 3.3 V source enables Fusion devices to initiate, control, and monitor multiple voltage supplies while also providing system clocks. In addition, glitches and brownouts in system power will not corrupt the Fusion device flash configuration. Unlike SRAM-based FPGAs, the device will not have to be reloaded when system power is restored. This enables reduction or complete removal of expensive voltage monitor and brownout detection devices from the PCB design.

Flash-based Fusion devices simplify total system design and reduce cost and design risk, while increasing system reliability.

### **Firm Errors**

Firm errors occur most commonly when high-energy neutrons, generated in the upper atmosphere, strike a configuration cell of an SRAM FPGA. The energy of the collision can change the state of the configuration cell and thus change the logic, routing, or I/O behavior in an unpredictable way. Another source of radiation-induced firm errors is alpha particles. For an alpha to cause a soft or firm error, its source must be in very close proximity to the affected circuit. The alpha source must be in the package molding compound or in the die itself. While low-alpha molding compounds are being used increasingly, this helps reduce but does not entirely eliminate alpha-induced firm errors.

Firm errors are impossible to prevent in SRAM FPGAs. The consequence of this type of error can be a complete system failure. Firm errors do not occur in Fusion flash-based FPGAs. Once it is programmed, the flash cell configuration element of Fusion FPGAs cannot be altered by high-energy neutrons and is therefore immune to errors from them.

Recoverable (or soft) errors occur in the user data SRAMs of all FPGA devices. These can easily be mitigated by using error detection and correction (EDAC) circuitry built into the FPGA fabric.

### **Low Power**

Flash-based Fusion devices exhibit power characteristics similar to those of an ASIC, making them an ideal choice for power-sensitive applications. With Fusion devices, there is no power-on current surge and no high current transition, both of which occur on many FPGAs.

Fusion devices also have low dynamic power consumption and support both low power standby mode and very low power sleep mode, offering further power savings.

## **Advanced Flash Technology**

The Fusion family offers many benefits, including nonvolatility and reprogrammability through an advanced flash-based, 130-nm LVCMOS process with seven layers of metal. Standard CMOS design techniques are used to implement logic and control functions. The combination of fine granularity, enhanced flexible routing resources, and abundant flash switches allows very high logic utilization (much higher than competing SRAM technologies) without compromising device routability or performance. Logic functions within the device are interconnected through a four-level routing hierarchy.

## Advanced Architecture

The proprietary Fusion architecture provides granularity comparable to standard-cell ASICs. The Fusion device consists of several distinct and programmable architectural features, including the following (Figure 1-1 on page 1-5):

- Embedded memories
  - Flash memory blocks
  - FlashROM
  - SRAM and FIFO
- Clocking resources
  - PLL and CCC
  - RC oscillator
  - Crystal oscillator
  - No-Glitch MUX (NGMUX)
- Digital I/Os with advanced I/O standards
- FPGA VersaTiles
- Analog components
  - ADC
  - Analog I/Os supporting voltage, current, and temperature monitoring
  - 1.5 V on-board voltage regulator
  - Real-time counter

The FPGA core consists of a sea of VersaTiles. Each VersaTile can be configured as a three-input logic lookup table (LUT) equivalent or a D-flip-flop or latch (with or without enable) by programming the appropriate flash switch interconnections. This versatility allows efficient use of the FPGA fabric. The VersaTile capability is unique to the Microsemi families of flash-based FPGAs. VersaTiles and larger functions are connected with any of the four levels of routing hierarchy. Flash switches are distributed throughout the device to provide nonvolatile, reconfigurable interconnect programming. Maximum core utilization is possible for virtually any design.

In addition, extensive on-chip programming circuitry allows for rapid (3.3 V) single-voltage programming of Fusion devices via an IEEE 1532 JTAG interface.

## Unprecedented Integration

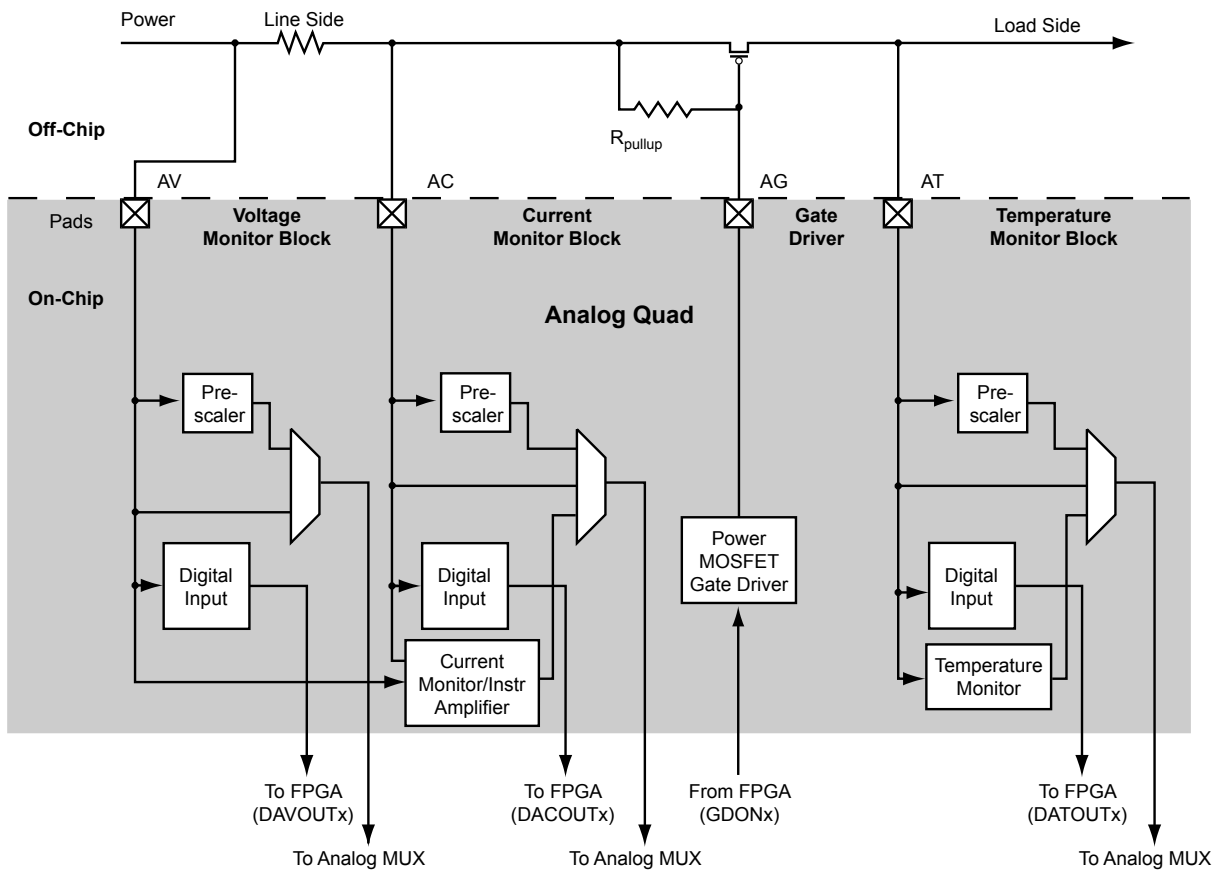
### Integrated Analog Blocks and Analog I/Os

Fusion devices offer robust and flexible analog mixed signal capability in addition to the high-performance flash FPGA fabric and flash memory block. The many built-in analog peripherals include a configurable 32:1 input analog MUX, up to 10 independent MOSFET gate driver outputs, and a configurable ADC. The ADC supports 8-, 10-, and 12-bit modes of operation with a cumulative sample rate up to 600 k samples per second (Ksps), differential nonlinearity (DNL) < 1.0 LSB, and Total Unadjusted Error (TUE) of 0.72 LSB in 10-bit mode. The TUE is used for characterization of the conversion error and includes errors from all sources, such as offset and linearity. Internal bandgap circuitry offers 1% voltage reference accuracy with the flexibility of utilizing an external reference voltage. The ADC channel sampling sequence and sampling rate are programmable and implemented in the FPGA logic using Designer and Libero SoC software tool support.

Two channels of the 32-channel ADCMUX are dedicated. Channel 0 is connected internally to VCC and can be used to monitor core power supply. Channel 31 is connected to an internal temperature diode which can be used to monitor device temperature. The 30 remaining channels can be connected to external analog signals. The exact number of I/Os available for external connection signals is device-dependent (refer to the "Fusion Family" table on page 1 for details).

With Fusion, Microsemi also introduces the Analog Quad I/O structure (Figure 1-1). Each quad consists of three analog inputs and one gate driver. Each quad can be configured in various built-in circuit combinations, such as three prescaler circuits, three digital input circuits, a current monitor circuit, or a temperature monitor circuit. Each prescaler has multiple scaling factors programmed by FPGA signals to support a large range of analog inputs with positive or negative polarity. When the current monitor circuit is selected, two adjacent analog inputs measure the voltage drop across a small external sense resistor. For more information, refer to the "Analog System Characteristics" section on page 2-117. Built-in operational amplifiers amplify small voltage signals for accurate current measurement. One analog input in each quad can be connected to an external temperature monitor diode. In addition to the external temperature monitor diode(s), a Fusion device can monitor an internal temperature diode using dedicated channel 31 of the ADCMUX.

Figure 1-1 on page 1-5 illustrates a typical use of the Analog Quad I/O structure. The Analog Quad shown is configured to monitor and control an external power supply. The AV pad measures the source of the power supply. The AC pad measures the voltage drop across an external sense resistor to calculate current. The AG MOSFET gate driver pad turns the external MOSFET on and off. The AT pad measures the load-side voltage level.



**Figure 1-1 • Analog Quad**

## Embedded Memories

### Flash Memory Blocks

The flash memory available in each Fusion device is composed of one to four flash blocks, each 2 Mbits in density. Each block operates independently with a dedicated flash controller and interface. Fusion flash memory blocks combine fast access times (60 ns random access and 10 ns access in Read-Ahead mode) with a configurable 8-, 16-, or 32-bit datapath, enabling high-speed flash operation without wait states. The memory block is organized in pages and sectors. Each page has 128 bytes, with 33 pages comprising one sector and 64 sectors per block. The flash block can support multiple partitions. The only constraint on size is that partition boundaries must coincide with page boundaries. The flexibility and granularity enable many use models and allow added granularity in programming updates.

Fusion devices support two methods of external access to the flash memory blocks. The first method is a serial interface that features a built-in JTAG-compliant port, which allows in-system programmability during user or monitor/test modes. This serial interface supports programming of an AES-encrypted stream. Data protected with security measures can be passed through the JTAG interface, decrypted, and then programmed in the flash block. The second method is a soft parallel interface.

FPGA logic or an on-chip soft microprocessor can access flash memory through the parallel interface. Since the flash parallel interface is implemented in the FPGA fabric, it can potentially be customized to meet special user requirements. For more information, refer to the [CoreCFI Handbook](#). The flash memory parallel interface provides configurable byte-wide ( $\times 8$ ), word-wide ( $\times 16$ ), or dual-word-wide ( $\times 32$ ) data-port options. Through the programmable flash parallel interface, the on-chip and off-chip memories can be cascaded for wider or deeper configurations.

The flash memory has built-in security. The user can configure either the entire flash block or the small blocks to protect against unintentional or intrusive attempts to change or destroy the storage contents. Each on-chip flash memory block has a dedicated controller, enabling each block to operate independently.

The flash block logic consists of the following sub-blocks:

- Flash block – Contains all stored data. The flash block contains 64 sectors and each sector contains 33 pages of data.
- Page Buffer – Contains the contents of the current page being modified. A page contains 8 blocks of data.
- Block Buffer – Contains the contents of the last block accessed. A block contains 128 data bits.
- ECC Logic – The flash memory stores error correction information with each block to perform single-bit error correction and double-bit error detection on all data blocks.

### User Nonvolatile FlashROM

In addition to the flash blocks, Fusion devices have 1 Kbit of user-accessible, nonvolatile FlashROM on-chip. The FlashROM is organized as  $8 \times 128$ -bit pages. The FlashROM can be used in diverse system applications:

- Internet protocol addressing (wireless or fixed)
- System calibration settings
- Device serialization and/or inventory control
- Subscription-based business models (for example, set-top boxes)
- Secure key storage for communications algorithms protected by security
- Asset management/tracking
- Date stamping
- Version management

The FlashROM is written using the standard IEEE 1532 JTAG programming interface. Pages can be individually programmed (erased and written). On-chip AES decryption can be used selectively over public networks to load data such as security keys stored in the FlashROM for a user design.

The FlashROM can be programmed (erased and written) via the JTAG programming interface, and its contents can be read back either through the JTAG programming interface or via direct FPGA core addressing.

The FlashPoint tool in the Fusion development software solutions, Libero SoC and Designer, has extensive support for flash memory blocks and FlashROM. One such feature is auto-generation of sequential programming files for applications requiring a unique serial number in each part. Another feature allows the inclusion of static data for system version control. Data for the FlashROM can be generated quickly and easily using the Libero SoC and Designer software tools. Comprehensive programming file support is also included to allow for easy programming of large numbers of parts with differing FlashROM contents.

### **SRAM and FIFO**

Fusion devices have embedded SRAM blocks along the north and south sides of the device. Each variable-aspect-ratio SRAM block is 4,608 bits in size. Available memory configurations are 256×18, 512×9, 1k×4, 2k×2, and 4k×1 bits. The individual blocks have independent read and write ports that can be configured with different bit widths on each port. For example, data can be written through a 4-bit port and read as a single bitstream. The SRAM blocks can be initialized from the flash memory blocks or via the device JTAG port (ROM emulation mode), using the UJTAG macro.

In addition, every SRAM block has an embedded FIFO control unit. The control unit allows the SRAM block to be configured as a synchronous FIFO without using additional core VersaTiles. The FIFO width and depth are programmable. The FIFO also features programmable Almost Empty (AEMPTY) and Almost Full (AFULL) flags in addition to the normal EMPTY and FULL flags. The embedded FIFO control unit contains the counters necessary for the generation of the read and write address pointers. The SRAM/FIFO blocks can be cascaded to create larger configurations.

## **Clock Resources**

### **PLLs and Clock Conditioning Circuits (CCCs)**

Fusion devices provide designers with very flexible clock conditioning capabilities. Each member of the Fusion family contains six CCCs. In the two larger family members, two of these CCCs also include a PLL; the smaller devices support one PLL.

The inputs of the CCC blocks are accessible from the FPGA core or from one of several inputs with dedicated CCC block connections.

The CCC block has the following key features:

- Wide input frequency range ( $f_{IN\_CCC}$ ) = 1.5 MHz to 350 MHz
- Output frequency range ( $f_{OUT\_CCC}$ ) = 0.75 MHz to 350 MHz
- Clock phase adjustment via programmable and fixed delays from -6.275 ns to +8.75 ns
- Clock skew minimization (PLL)
- Clock frequency synthesis (PLL)
- On-chip analog clocking resources usable as inputs:
  - 100 MHz on-chip RC oscillator
  - Crystal oscillator

Additional CCC specifications:

- Internal phase shift = 0°, 90°, 180°, and 270°
- Output duty cycle = 50% ± 1.5%
- Low output jitter. Samples of peak-to-peak period jitter when a single global network is used:
  - 70 ps at 350 MHz
  - 90 ps at 100 MHz
  - 180 ps at 24 MHz
  - Worst case < 2.5% × clock period
- Maximum acquisition time = 150 μs
- Low power consumption of 5 mW

## Global Clocking

Fusion devices have extensive support for multiple clocking domains. In addition to the CCC and PLL support described above, there are on-chip oscillators as well as a comprehensive global clock distribution network.

The integrated RC oscillator generates a 100 MHz clock. It is used internally to provide a known clock source to the flash memory read and write control. It can also be used as a source for the PLLs.

The crystal oscillator supports the following operating modes:

- Crystal (32.768 KHz to 20 MHz)
- Ceramic (500 KHz to 8 MHz)
- RC (32.768 KHz to 4 MHz)

Each VersaTile input and output port has access to nine VersaNets: six main and three quadrant global networks. The VersaNets can be driven by the CCC or directly accessed from the core via MUXes. The VersaNets can be used to distribute low-skew clock signals or for rapid distribution of high-fanout nets.

## Digital I/Os with Advanced I/O Standards

The Fusion family of FPGAs features a flexible digital I/O structure, supporting a range of voltages (1.5 V, 1.8 V, 2.5 V, and 3.3 V). Fusion FPGAs support many different digital I/O standards, both single-ended and differential.

The I/Os are organized into banks, with four or five banks per device. The configuration of these banks determines the I/O standards supported. The banks along the east and west sides of the device support the full range of I/O standards (single-ended and differential). The south bank supports the Analog Quads (analog I/O). In the family's two smaller devices, the north bank supports multiple single-ended digital I/O standards. In the family's larger devices, the north bank is divided into two banks of digital Pro I/Os, supporting a wide variety of single-ended, differential, and voltage-referenced I/O standards.

Each I/O module contains several input, output, and enable registers. These registers allow the implementation of the following applications:

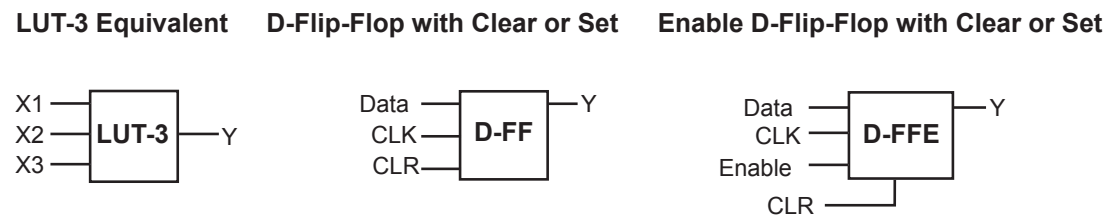
- Single-Data-Rate (SDR) applications
- Double-Data-Rate (DDR) applications—DDR LVDS I/O for chip-to-chip communications
- Fusion banks support LVPECL, LVDS, BLVDS, and M-LVDS with 20 multi-drop points.

## VersaTiles

The Fusion core consists of VersaTiles, which are also used in the successful ProASIC3 family. The Fusion VersaTile supports the following:

- All 3-input logic functions—LUT-3 equivalent
- Latch with clear or set
- D-flip-flop with clear or set and optional enable

Refer to [Figure 1-2](#) for the VersaTile configuration arrangement.



**Figure 1-2 • VersaTile Configurations**

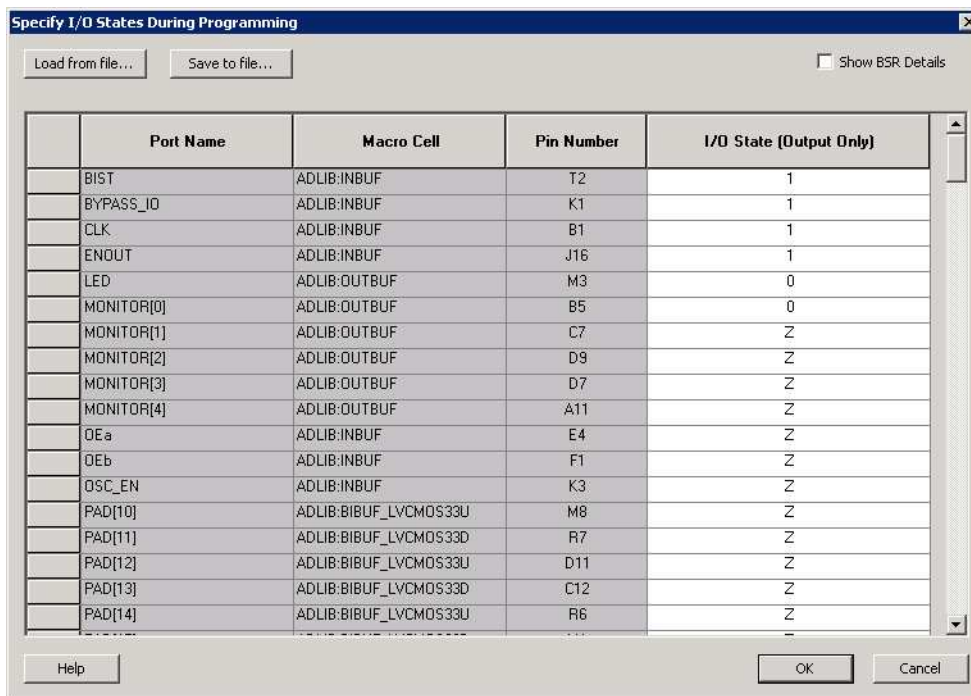
## Specifying I/O States During Programming

You can modify the I/O states during programming in FlashPro. In FlashPro, this feature is supported for PDB files generated from Designer v8.5 or greater. See the *FlashPro User Guide* for more information.

**Note:** PDB files generated from Designer v8.1 to Designer v8.4 (including all service packs) have limited display of Pin Numbers only.

The I/Os are controlled by the JTAG Boundary Scan register during programming, except for the analog pins (AC, AT and AV). The Boundary Scan register of the AG pin can be used to enable/disable the gate driver in software v9.0.

1. Load a PDB from the FlashPro GUI. You must have a PDB loaded to modify the I/O states during programming.
2. From the FlashPro GUI, click **PDB Configuration**. A FlashPoint – Programming File Generator window appears.
3. Click the **Specify I/O States During Programming** button to display the Specify I/O States During Programming dialog box.
4. Sort the pins as desired by clicking any of the column headers to sort the entries by that header. Select the I/Os you wish to modify (Figure 1-3).
5. Set the I/O Output State. You can set Basic I/O settings if you want to use the default I/O settings for your pins, or use Custom I/O settings to customize the settings for each pin. Basic I/O state settings:
  - 1 – I/O is set to drive out logic High
  - 0 – I/O is set to drive out logic Low
  - Last Known State – I/O is set to the last value that was driven out prior to entering the programming mode, and then held at that value during programming
  - Z -Tri-State: I/O is tristated



**Figure 1-3 • I/O States During Programming Window**

6. Click **OK** to return to the FlashPoint – Programming File Generator window.

I/O States During programming are saved to the ADB and resulting programming files after completing programming file generation.

## Related Documents

### Datasheet

Core8051

[www.microsemi.com/soc/ipdocs/Core8051\\_DS.pdf](http://www.microsemi.com/soc/ipdocs/Core8051_DS.pdf)

### Application Notes

*Fusion FlashROM*

[http://www.microsemi.com/soc/documents/Fusion\\_FROM\\_AN.pdf](http://www.microsemi.com/soc/documents/Fusion_FROM_AN.pdf)

*Fusion SRAM/FIFO Blocks*

[http://www.microsemi.com/soc/documents/Fusion\\_RAM\\_FIFO\\_AN.pdf](http://www.microsemi.com/soc/documents/Fusion_RAM_FIFO_AN.pdf)

*Using DDR in Fusion Devices*

[http://www.microsemi.com/index.php?option=com\\_docman&task=doc\\_download&gid=129938](http://www.microsemi.com/index.php?option=com_docman&task=doc_download&gid=129938)

*Fusion Security*

[http://www.microsemi.com/soc/documents/Fusion\\_Security\\_AN.pdf](http://www.microsemi.com/soc/documents/Fusion_Security_AN.pdf)

*Using Fusion RAM as Multipliers*

[http://www.microsemi.com/index.php?option=com\\_docman&task=doc\\_download&gid=129940](http://www.microsemi.com/index.php?option=com_docman&task=doc_download&gid=129940)

### Handbook

*Cortex-M1 Handbook*

[www.microsemi.com/soc/documents/CortexM1\\_HB.pdf](http://www.microsemi.com/soc/documents/CortexM1_HB.pdf)

### User Guides

*Designer User Guide*

[http://www.microsemi.com/soc/documents/designer\\_UG.pdf](http://www.microsemi.com/soc/documents/designer_UG.pdf)

*Fusion FPGA Fabric User Guide*

[http://www.microsemi.com/index.php?option=com\\_docman&task=doc\\_download&gid=130817](http://www.microsemi.com/index.php?option=com_docman&task=doc_download&gid=130817)

*IGLOO, ProASIC3, SmartFusion and Fusion Macro Library Guide*

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

*SmartGen, FlashROM, Flash Memory System Builder, and Analog System Builder User Guide*

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

### White Papers

*Fusion Technology*

[http://www.microsemi.com/soc/documents/Fusion\\_Tech\\_WP.pdf](http://www.microsemi.com/soc/documents/Fusion_Tech_WP.pdf)



## 2 – Device Architecture

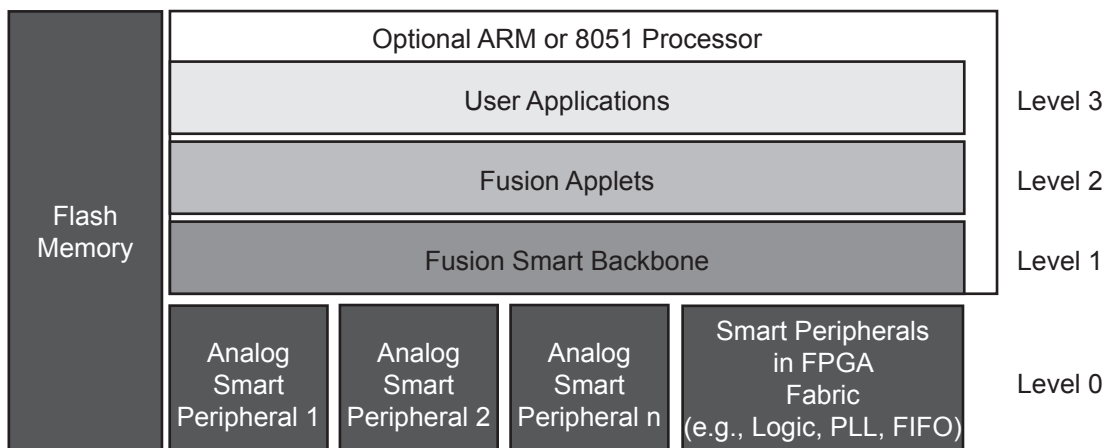
### Fusion Stack Architecture

To manage the unprecedented level of integration in Fusion devices, Microsemi developed the Fusion technology stack (Figure 2-1). This layered model offers a flexible design environment, enabling design at very high and very low levels of abstraction. Fusion peripherals include hard analog IP and hard and soft digital IP. Peripherals communicate across the FPGA fabric via a layer of soft gates—the Fusion backbone. Much more than a common bus interface, this Fusion backbone integrates a micro-sequencer within the FPGA fabric and configures the individual peripherals and supports low-level processing of peripheral data. Fusion applets are application building blocks that can control and respond to peripherals and other system signals. Applets can be rapidly combined to create large applications. The technology is scalable across devices, families, design types, and user expertise, and supports a well-defined interface for external IP and tool integration.

At the lowest level, Level 0, are Fusion peripherals. These are configurable functional blocks that can be hardwired structures such as a PLL or analog input channel, or soft (FPGA gate) blocks such as a UART or two-wire serial interface. The Fusion peripherals are configurable and support a standard interface to facilitate communication and implementation.

Connecting and controlling access to the peripherals is the Fusion backbone, Level 1. The backbone is a soft-gate structure, scalable to any number of peripherals. The backbone is a bus and much more; it manages peripheral configuration to ensure proper operation. Leveraging the common peripheral interface and a low-level state machine, the backbone efficiently offloads peripheral management from the system design. The backbone can set and clear flags based upon peripheral behavior and can define performance criteria. The flexibility of the stack enables a designer to configure the silicon, directly bypassing the backbone if that level of control is desired.

One step up from the backbone is the Fusion applet, Level 2. The applet is an application building block that implements a specific function in FPGA gates. It can react to stimuli and board-level events coming through the backbone or from other sources, and responds to these stimuli by accessing and manipulating peripherals via the backbone or initiating some other action. An applet controls or responds to the peripheral(s). Applets can be easily imported or exported from the design environment. The applet structure is open and well-defined, enabling users to import applets from Microsemi, system developers, third parties, and user groups.



*Note:* Levels 1, 2, and 3 are implemented in FPGA logic gates.

**Figure 2-1 • Fusion Architecture Stack**

The system application, Level 3, is the larger user application that utilizes one or more applets. Designing at the highest level of abstraction supported by the Fusion technology stack, the application can be easily created in FPGA gates by importing and configuring multiple applets.

In fact, in some cases an entire FPGA system design can be created without any HDL coding.

An optional MCU enables a combination of software and HDL-based design methodologies. The MCU can be on-chip or off-chip as system requirements dictate. System portioning is very flexible, allowing the MCU to reside above the applets or to absorb applets, or applets and backbone, if desired.

The Fusion technology stack enables a very flexible design environment. Users can engage in design across a continuum of abstraction from very low to very high.

## Core Architecture

### VersaTile

Based upon successful ProASIC3/E logic architecture, Fusion devices provide granularity comparable to gate arrays. The Fusion device core consists of a sea-of-VersaTiles architecture.

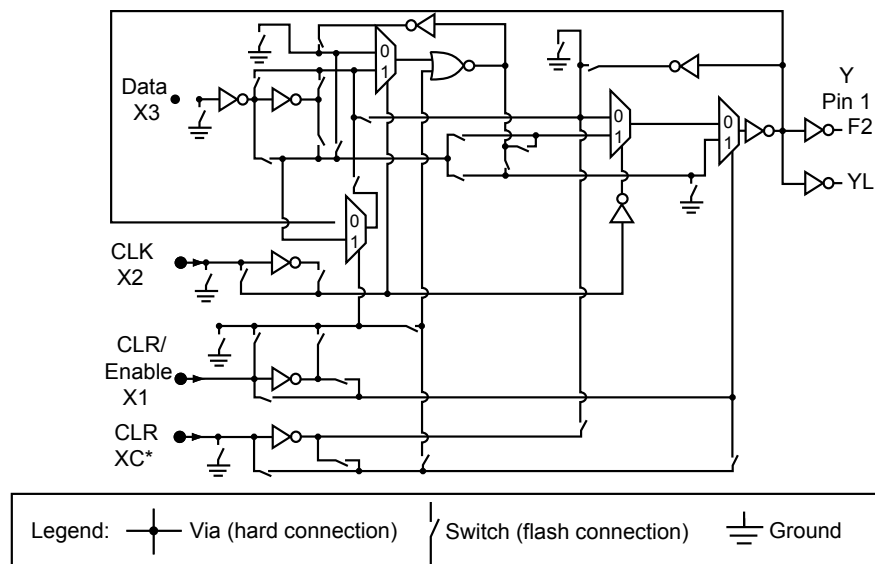
As illustrated in [Figure 2-2](#), there are four inputs in a logic VersaTile cell, and each VersaTile can be configured using the appropriate flash switch connections:

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

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

When the VersaTile is used as an enable D-flip-flop, the SET/CLR signal is supported by a fourth input, which can only be routed to the core cell over the VersaNet (global) network.

The output of the VersaTile is F2 when the connection is to the ultra-fast local lines, or YL when the connection is to the efficient long-line or very-long-line resources ([Figure 2-2](#)).



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

**Figure 2-2 • Fusion Core VersaTile**

## VersaTile Characteristics

### Sample VersaTile Specifications—Combinatorial Module

The Fusion library offers all combinations of LUT-3 combinatorial functions. In this section, timing characteristics are presented for a sample of the library (Figure 2-3). For more details, refer to the *IGLOO, ProASIC3, SmartFusion, and Fusion Macro Library Guide*.

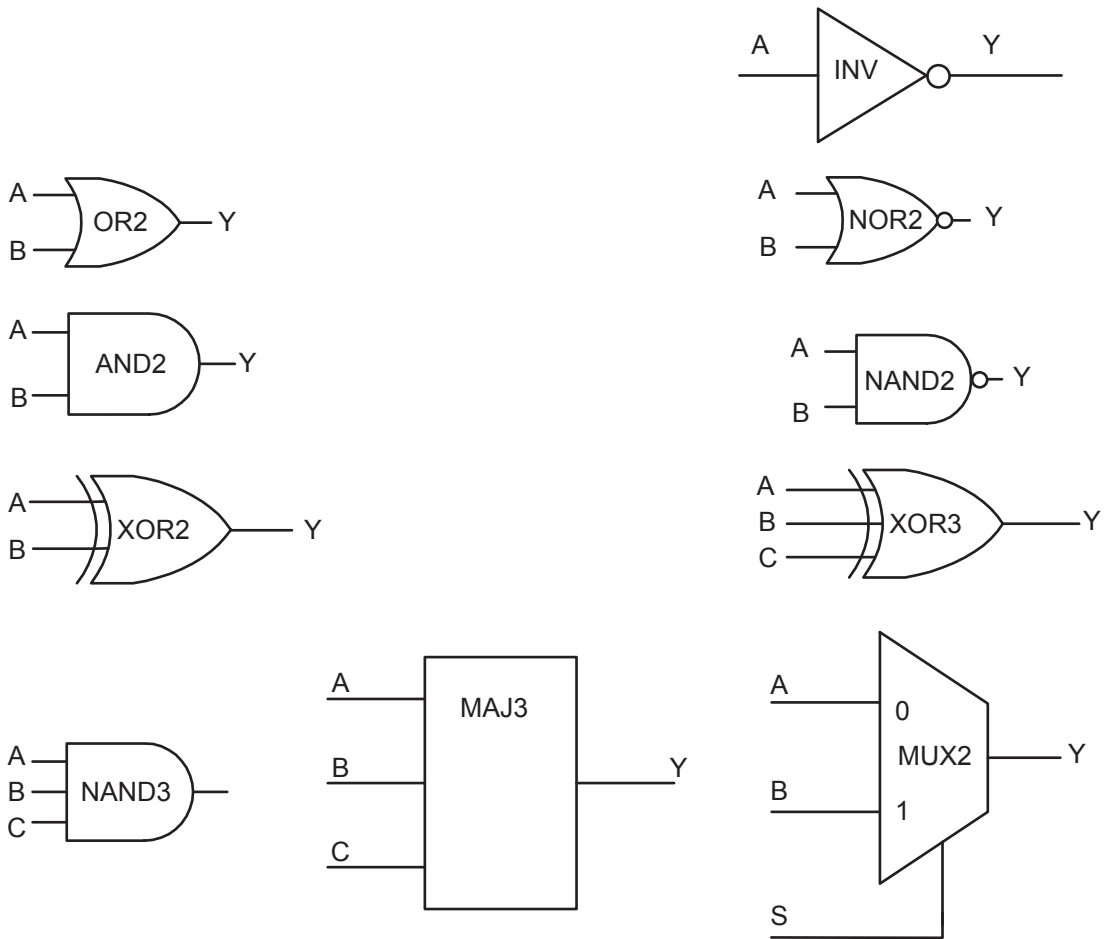
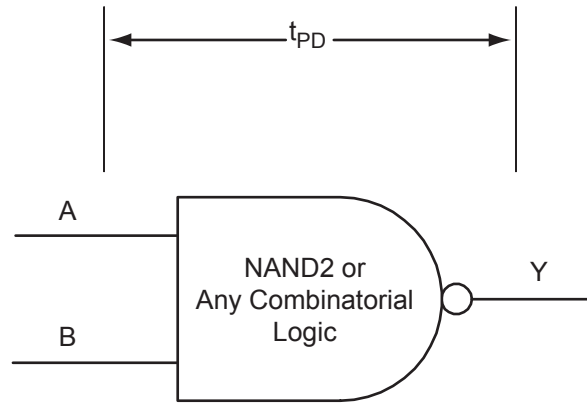
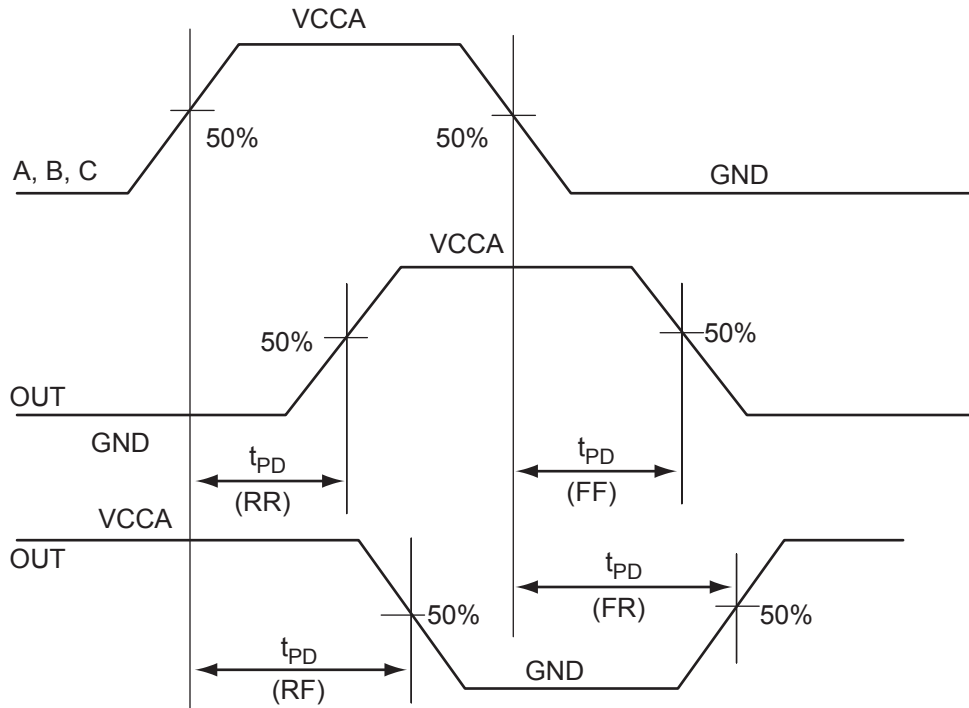


Figure 2-3 • Sample of Combinatorial Cells



$t_{PD} = \text{MAX}(t_{PD(RR)}, t_{PD(RF)}, t_{PD(FF)}, t_{PD(FR)})$   
 where edges are applicable for the particular combinatorial cell



**Figure 2-4 • Combinatorial Timing Model and Waveforms**

**Timing Characteristics**

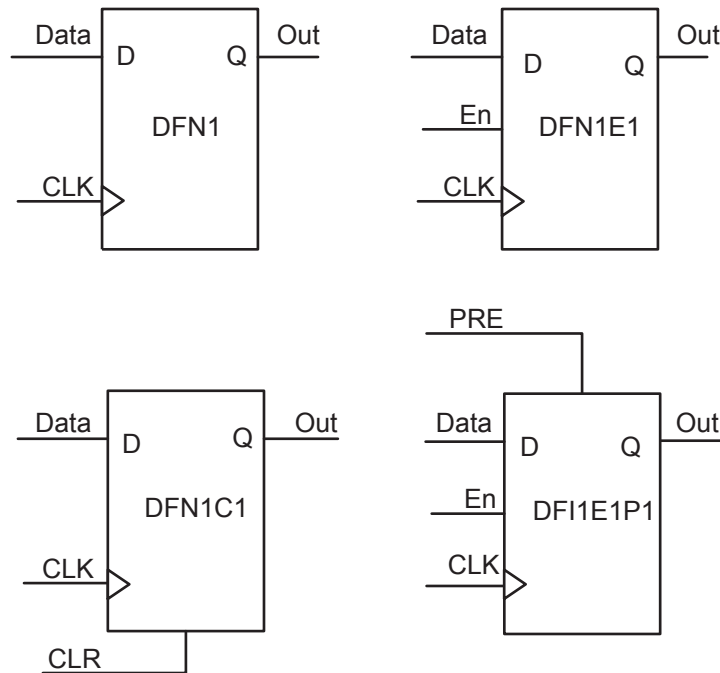
**Table 2-1 • Combinatorial Cell Propagation Delays**  
Commercial Temperature Range Conditions:  $T_J = 70^\circ\text{C}$ , Worst-Case  $V_{CC} = 1.425\text{ V}$

Combinatorial Cell	Equation	Parameter	-2	-1	Std.	Units
INV	$Y = !A$	$t_{PD}$	0.40	0.46	0.54	ns
AND2	$Y = A \cdot B$	$t_{PD}$	0.47	0.54	0.63	ns
NAND2	$Y = !(A \cdot B)$	$t_{PD}$	0.47	0.54	0.63	ns
OR2	$Y = A + B$	$t_{PD}$	0.49	0.55	0.65	ns
NOR2	$Y = !(A + B)$	$t_{PD}$	0.49	0.55	0.65	ns
XOR2	$Y = A \oplus B$	$t_{PD}$	0.74	0.84	0.99	ns
MAJ3	$Y = \text{MAJ}(A, B, C)$	$t_{PD}$	0.70	0.79	0.93	ns
XOR3	$Y = A \oplus B \oplus C$	$t_{PD}$	0.87	1.00	1.17	ns
MUX2	$Y = A \text{ IS } + B \text{ S}$	$t_{PD}$	0.51	0.58	0.68	ns
AND3	$Y = A \cdot B \cdot C$	$t_{PD}$	0.56	0.64	0.75	ns

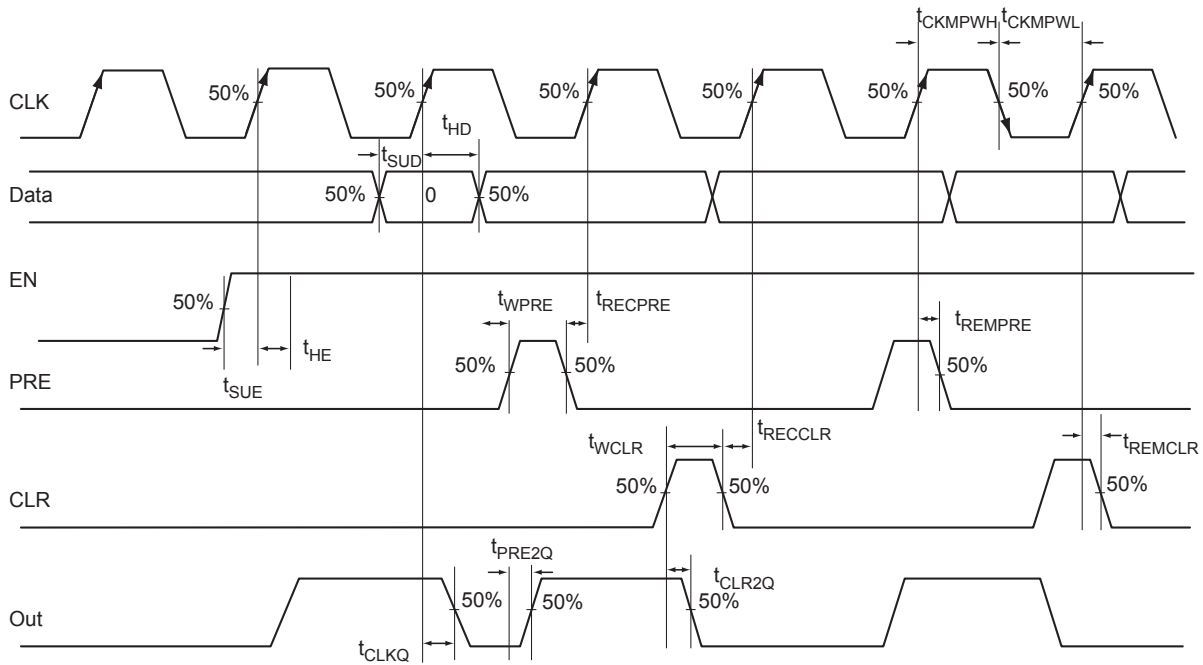
*Note:* For the derating values at specific junction temperature and voltage supply levels, refer to [Table 3-7](#) on page 3-9.

**Sample VersaTile Specifications—Sequential Module**

The Fusion library offers a wide variety of sequential cells, including flip-flops and latches. Each has a data input and optional enable, clear, or preset. In this section, timing characteristics are presented for a representative sample from the library ([Figure 2-5](#)). For more details, refer to the *IGLOO*, *ProASIC3*, *SmartFusion* and *Fusion Macro Library Guide*.



**Figure 2-5 • Sample of Sequential Cells**



**Figure 2-6 • Sequential Timing Model and Waveforms**

**Sequential Timing Characteristics**

**Table 2-2 • Register Delays**  
Commercial Temperature Range Conditions:  $T_J = 70^\circ\text{C}$ , Worst-Case  $V_{CC} = 1.425\text{ V}$

Parameter	Description	-2	-1	Std.	Units
$t_{CLKQ}$	Clock-to-Q of the Core Register	0.55	0.63	0.74	ns
$t_{SUD}$	Data Setup Time for the Core Register	0.43	0.49	0.57	ns
$t_{HD}$	Data Hold Time for the Core Register	0.00	0.00	0.00	ns
$t_{SUE}$	Enable Setup Time for the Core Register	0.45	0.52	0.61	ns
$t_{HE}$	Enable Hold Time for the Core Register	0.00	0.00	0.00	ns
$t_{CLR2Q}$	Asynchronous Clear-to-Q of the Core Register	0.40	0.45	0.53	ns
$t_{PRE2Q}$	Asynchronous Preset-to-Q of the Core Register	0.40	0.45	0.53	ns
$t_{REMCLR}$	Asynchronous Clear Removal Time for the Core Register	0.00	0.00	0.00	ns
$t_{RECLR}$	Asynchronous Clear Recovery Time for the Core Register	0.22	0.25	0.30	ns
$t_{REMPRE}$	Asynchronous Preset Removal Time for the Core Register	0.00	0.00	0.00	ns
$t_{RECPRE}$	Asynchronous Preset Recovery Time for the Core Register	0.22	0.25	0.30	ns
$t_{WCLR}$	Asynchronous Clear Minimum Pulse Width for the Core Register	0.22	0.25	0.30	ns
$t_{WPRE}$	Asynchronous Preset Minimum Pulse Width for the Core Register	0.22	0.25	0.30	ns
$t_{CKMPWH}$	Clock Minimum Pulse Width High for the Core Register	0.32	0.37	0.43	ns
$t_{CKMPWL}$	Clock Minimum Pulse Width Low for the Core Register	0.36	0.41	0.48	ns

*Note:* For the derating values at specific junction temperature and voltage supply levels, refer to [Table 3-7 on page 3-9](#).

## Array Coordinates

During many place-and-route operations in the Microsemi Designer software tool, it is possible to set constraints that require array coordinates. Table 2-3 is provided as a reference. The array coordinates are measured from the lower left (0, 0). They can be used in region constraints for specific logic groups/blocks, designated by a wildcard, and can contain core cells, memories, and I/Os.

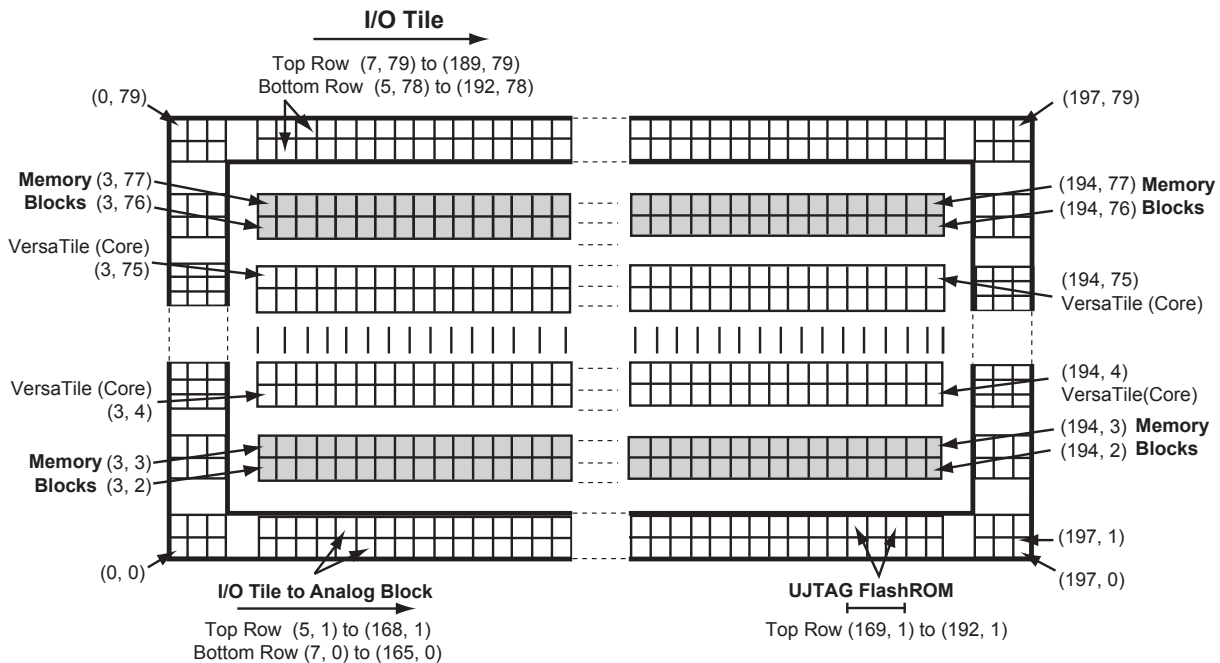
Table 2-3 provides array coordinates of core cells and memory blocks.

I/O and cell coordinates are used for placement constraints. Two coordinate systems are needed because there is not a one-to-one correspondence between I/O cells and edge core cells. In addition, the I/O coordinate system changes depending on the die/package combination. It is not listed in Table 2-3. The Designer ChipPlanner tool provides array coordinates of all I/O locations. I/O and cell coordinates are used for placement constraints. However, I/O placement is easier by package pin assignment.

Figure 2-7 illustrates the array coordinates of an AFS600 device. For more information on how to use array coordinates for region/placement constraints, see the *Designer User's Guide* or online help (available in the software) for Fusion software tools.

**Table 2-3 • Array Coordinates**

Device	VersaTiles				Memory Rows		All	
	Min.		Max.		Bottom	Top	Min.	Max.
	x	y	x	y	(x, y)	(x, y)	(x, y)	(x, y)
AFS090	3	2	98	25	None	(3, 26)	(0, 0)	(101, 29)
AFS250	3	2	130	49	None	(3, 50)	(0, 0)	(133, 53)
AFS600	3	4	194	75	(3, 2)	(3, 76)	(0, 0)	(197, 79)
AFS1500	3	4	322	123	(3, 2)	(3, 124)	(0, 0)	(325, 129)



**Note:** The vertical I/O tile coordinates are not shown. West side coordinates are {(0, 2) to (2, 2)} to {(0, 77) to (2, 77)}; east side coordinates are {(195, 2) to (197, 2)} to {(195, 77) to (197, 77)}.

**Figure 2-7 • Array Coordinates for AFS600**

## Routing Architecture

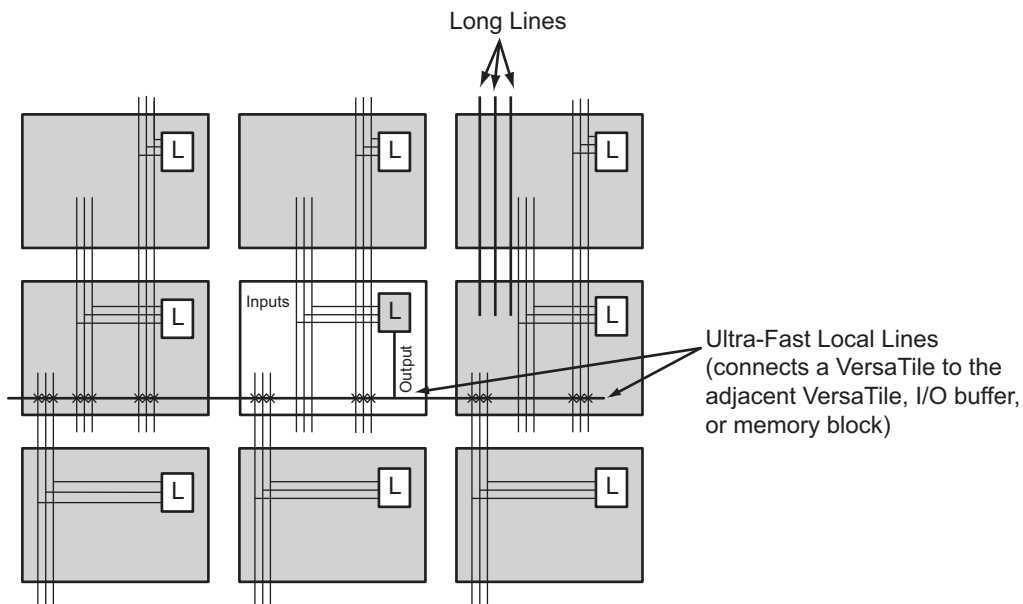
The routing structure of Fusion devices is designed to provide high performance through a flexible four-level hierarchy of routing resources: ultra-fast local resources; efficient long-line resources; high-speed very-long-line resources; and the high-performance VersaNet networks.

The ultra-fast local resources are dedicated lines that allow the output of each VersaTile to connect directly to every input of the eight surrounding VersaTiles (Figure 2-8). The exception to this is that the SET/CLR input of a VersaTile configured as a D-flip-flop is driven only by the VersaNet global network.

The efficient long-line resources provide routing for longer distances and higher-fanout connections. These resources vary in length (spanning one, two, or four VersaTiles), run both vertically and horizontally, and cover the entire Fusion device (Figure 2-9 on page 2-9). Each VersaTile can drive signals onto the efficient long-line resources, which can access every input of every VersaTile. Active buffers are inserted automatically by routing software to limit loading effects.

The high-speed very-long-line resources, which span the entire device with minimal delay, are used to route very long or high-fanout nets: length  $\pm 12$  VersaTiles in the vertical direction and length  $\pm 16$  in the horizontal direction from a given core VersaTile (Figure 2-10 on page 2-10). Very long lines in Fusion devices, like those in ProASIC3 devices, have been enhanced. This provides a significant performance boost for long-reach signals.

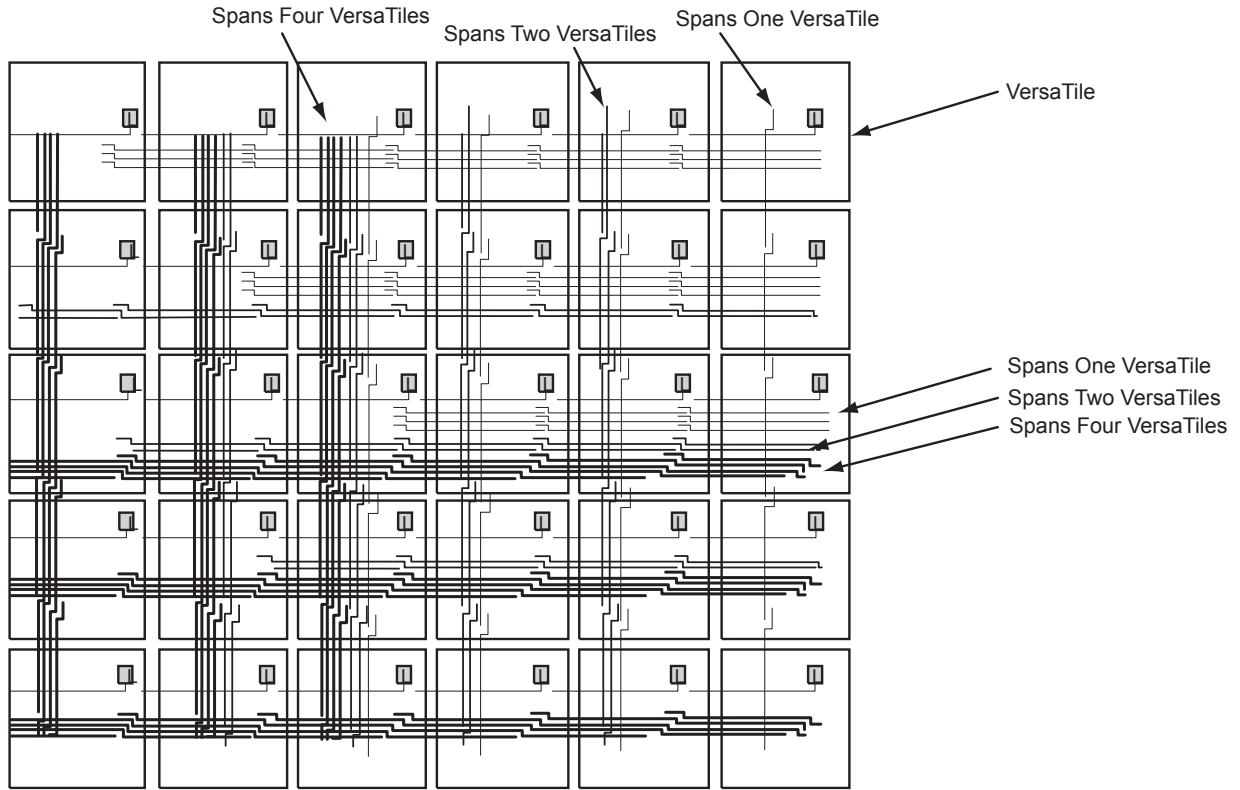
The high-performance VersaNet global networks are low-skew, high-fanout nets that are accessible from external pins or from internal logic (Figure 2-11 on page 2-11). These nets are typically used to distribute clocks, reset signals, and other high-fanout nets requiring minimum skew. The VersaNet networks are implemented as clock trees, and signals can be introduced at any junction. These can be employed hierarchically, with signals accessing every input on all VersaTiles.



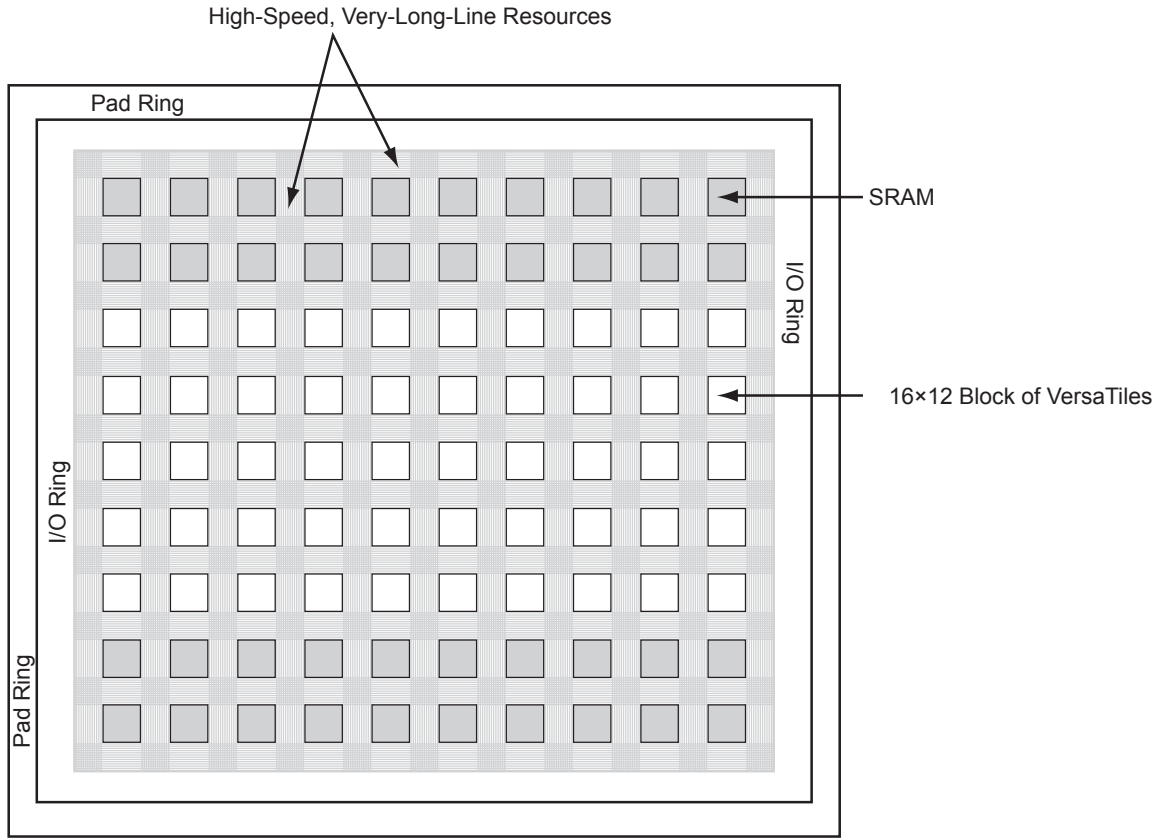
*Note:* Input to the core cell for the D-flip-flop set and reset is only available via the VersaNet global network connection.

**Figure 2-8 • Ultra-Fast Local Lines Connected to the Eight Nearest Neighbors**





**Figure 2-9 • Efficient Long-Line Resources**



**Figure 2-10 • Very-Long-Line Resources**