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	95
Number of Gates	600000
Voltage - Supply	1.425V ~ 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/afs600-1pqg208

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

2 – Device Architecture

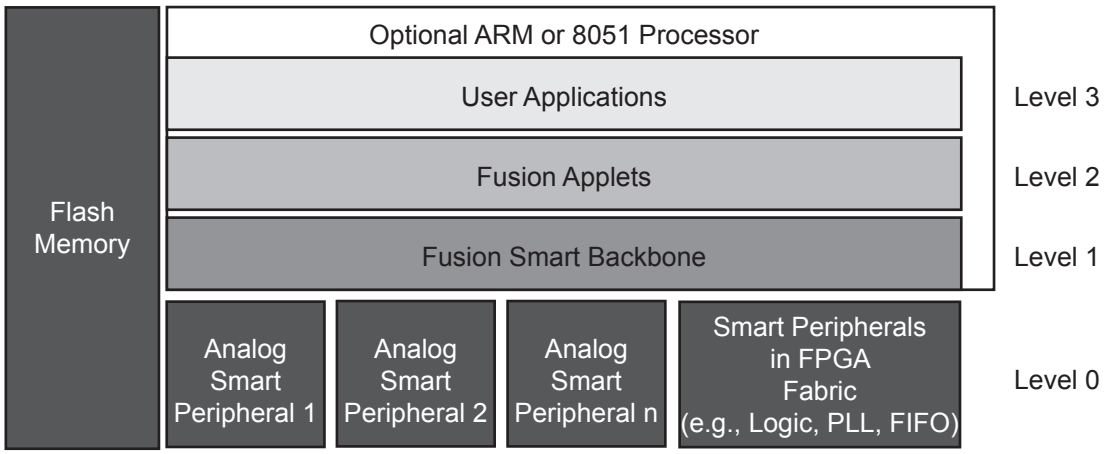
Fusion Stack Architecture

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**

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.

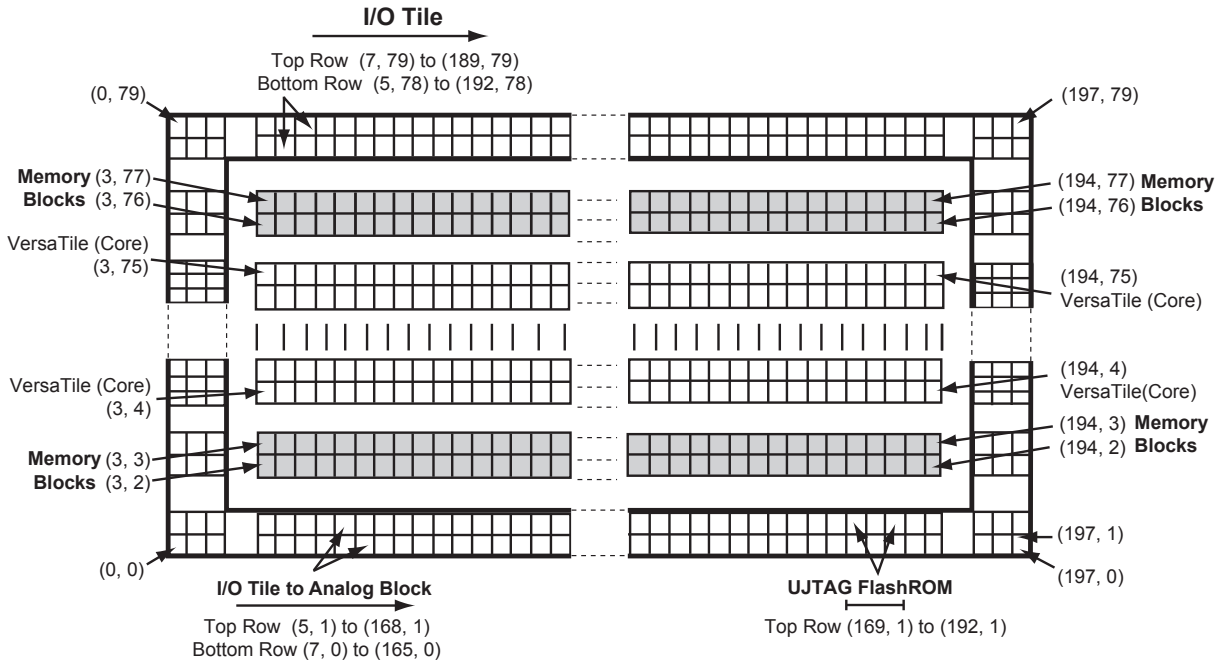
Table 2-3 provides array coordinates of core cells and memory blocks.

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**

RC Oscillator

The RC oscillator is an on-chip free-running clock source generating a 100 MHz clock. It can be used as a source clock for both on-chip and off-chip resources. When used in conjunction with the Fusion PLL and CCC circuits, the RC oscillator clock source can be used to generate clocks of varying frequency and phase.

The Fusion RC oscillator is very accurate at ±1% over commercial temperature ranges and and ±3% over industrial temperature ranges. It is an automated clock, requiring no setup or configuration by the user. It requires only that the power and GNDOSC pins be connected; no external components are required. The RC oscillator can be used to drive either a PLL or another internal signal.

RC Oscillator Characteristics

Table 2-9 • **Electrical Characteristics of RC Oscillator**

Parameter	Description	Conditions	Min.	Typ.	Max.	Units
F_{RC}	Operating Frequency			100		MHz
	Accuracy	Temperature: 0°C to 85°C Voltage: 3.3 V ± 5%		1		%
		Temperature: –40°C to 125°C Voltage: 3.3 V ± 5%		3		%
	Output Jitter	Period Jitter (at 5 k cycles)		100		ps
		Cycle–Cycle Jitter (at 5 k cycles)		100		ps
		Period Jitter (at 5 k cycles) with 1 KHz / 300 mV peak-to-peak noise on power supply		150		ps
		Cycle–Cycle Jitter (at 5 k cycles) with 1 KHz / 300 mV peak-to-peak noise on power supply		150		ps
	Output Duty Cycle			50		%
I_{DYNRC}	Operating Current			1		mA

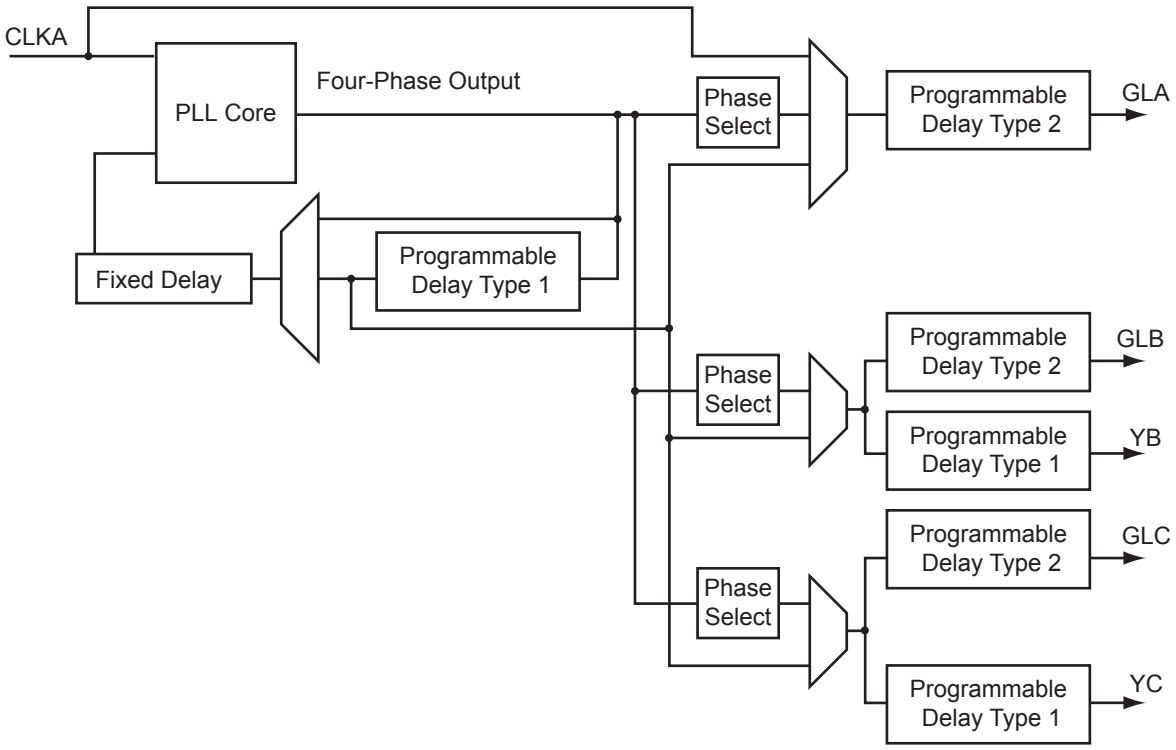
CCC Physical Implementation

The CCC circuit is composed of the following (Figure 2-23):

- PLL core
- 3 phase selectors
- 6 programmable delays and 1 fixed delay
- 5 programmable frequency dividers that provide frequency multiplication/division (not shown in Figure 2-23 because they are automatically configured based on the user's required frequencies)
- 1 dynamic shift register that provides CCC dynamic reconfiguration capability (not shown)

CCC Programming

The CCC block is fully configurable. It is configured via static flash configuration bits in the array, set by the user in the programming bitstream, or configured through an asynchronous dedicated shift register, dynamically accessible from inside the Fusion device. The dedicated shift register permits changes of parameters such as PLL divide ratios and delays during device operation. This latter mode allows the user to dynamically reconfigure the PLL without the need for core programming. The register file is accessed through a simple serial interface.



Note: Clock divider and multiplier blocks are not shown in this figure or in SmartGen. They are automatically configured based on the user's required frequencies.

Figure 2-23 • **PLL Block**

Real-Time Counter (part of AB macro)

The RTC is a 40-bit loadable counter and used as the primary timekeeping element (Figure 2-29). The clock source, RTCCLK, must come from the CLKOUT signal of the crystal oscillator. The RTC can be configured to reset itself when a count value reaches the match value set in the Match Register.

The RTC is part of the Analog Block (AB) macro. The RTC is configured by the analog configuration MUX (ACM). Each address contains one byte of data. The circuitry in the RTC is powered by VCC33A, so the RTC can be used in standby mode when the 1.5 V supply is not present.

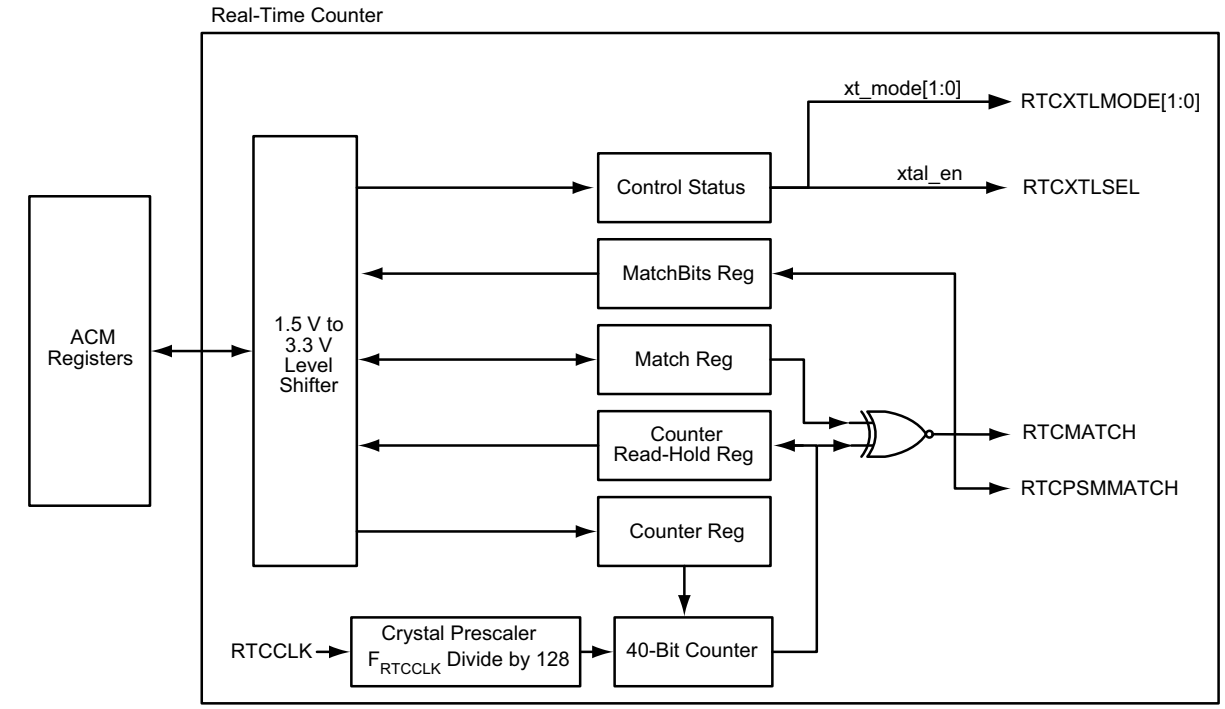


Figure 2-29 • **RTC Block Diagram**

Table 2-14 • **RTC Signal Description**

Signal Name	Width	Direction	Function
RTCCLK	1	In	Must come from CLKOUT of XTLOSC.
RTCXTLMODE[1:0]	2	Out	Controlled by xt_mode in CTRL_STAT. Signal must connect to the RTC_MODE signal in XTLOSC, as shown in Figure 2-27.
RTCXTLSEL	1	Out	Controlled by xtal_en from CTRL_STAT register. Signal must connect to RTC_MODE signal in XTLOSC in Figure 2-27.
RTCMATCH	1	Out	Match signal for FPGA 0 – Counter value does not equal the Match Register value. 1 – Counter value equals the Match Register value.
RTCPSMMATCH	1	Out	Same signal as RTCMATCH. Signal must connect to RTCPSMMATCH in VRPSM, as shown in Figure 2-27.

The 40-bit counter can be preloaded with an initial value as a starting point by the Counter Register. The count from the 40-bit counter can be read through the same set of address space. The count comes from a Read-Hold Register to avoid data changing during read. When the counter value equals the Match Register value, all Match Bits Register values will be 0xFFFFFFFFFF. The RTCMATCH and RTCPSMMATCH signals will assert. The 40-bit counter can be configured to automatically reset to 0x0000000000 when the counter value equals the Match Register value. The automatic reset does not apply if the Match Register value is 0x0000000000. The RTCCLK has a prescaler to divide the clock by 128 before it is used for the 40-bit counter. Below is an example of how to calculate the OFF time.

***Table 2-16* • RTC Control/Status Register**

Bit	Name	Description	Default Value
7	rtc_rst	RTC Reset 1 – Resets the RTC 0 – Deassert reset on after two ACM_CLK cycle.	
6	cntr_en	Counter Enable 1 – Enables the counter; rtc_rst must be deasserted as well. First counter increments after 64 RTCCLK positive edges. 0 – Disables the crystal prescaler but does not reset the counter value. Counter value can only be updated when the counter is disabled.	0
5	vr_en_mat	Voltage Regulator Enable on Match 1 – Enables RTCMATCH and RTCPSMMATCH to output 1 when the counter value equals the Match Register value. This enables the 1.5 V voltage regulator when RTCPSMMATCH connects to the RTCPSMMATCH signal in VRPSM. 0 – RTCMATCH and RTCPSMMATCH output 0 at all times.	0
4:3	xt_mode[1:0]	Crystal Mode Controls RTCXTLMODE[1:0]. Connects to RTC_MODE signal in XTLOSC. XTL_MODE uses this value when xtal_en is 1. See the "Crystal Oscillator" section on page 2-20 for mode configuration.	00
2	rst_cnt_omat	Reset Counter on Match 1 – Enables the sync clear of the counter when the counter value equals the Match Register value. The counter clears on the rising edge of the clock. If all the Match Registers are set to 0, the clear is disabled. 0 – Counter increments indefinitely	0
1	rstb_cnt	Counter Reset, active Low 0 - Resets the 40-bit counter value	0
0	xtal_en	Crystal Enable Controls RTCXTLSEL. Connects to SELMODE signal in XTLOSC. 0 – XTLOSC enables control by FPGA_EN; xt_mode is not used. Sleep mode requires this bit to equal 0. 1 – Enables XTLOSC, XTL_MODE control by xt_mode Standby mode requires this bit to be set to 1. See the "Crystal Oscillator" section on page 2-20 for further details on SELMODE configuration.	0

Table 2-25 • **Flash Memory Block Timing (continued)**
Commercial Temperature Range Conditions: T_J = 70°C, Worst-Case VCC = 1.425 V

Parameter	Description	–2	–1	Std.	Units
$t_{SUPGLOSSPRO}$	Page Loss Protect Setup Time for the Control Logic	1.69	1.93	2.27	ns
$t_{HDPGLOSSPRO}$	Page Loss Protect Hold Time for the Control Logic	0.00	0.00	0.00	ns
$t_{SUPGSTAT}$	Page Status Setup Time for the Control Logic	2.49	2.83	3.33	ns
$t_{HDPGSTAT}$	Page Status Hold Time for the Control Logic	0.00	0.00	0.00	ns
$t_{SUOVERWRPG}$	Over Write Page Setup Time for the Control Logic	1.88	2.14	2.52	ns
$t_{HDOVERWRPG}$	Over Write Page Hold Time for the Control Logic	0.00	0.00	0.00	ns
$t_{SULOCKREQUEST}$	Lock Request Setup Time for the Control Logic	0.87	0.99	1.16	ns
$t_{HDLOCKREQUEST}$	Lock Request Hold Time for the Control Logic	0.00	0.00	0.00	ns
$t_{RECARNVM}$	Reset Recovery Time	0.94	1.07	1.25	ns
$t_{REMARNVM}$	Reset Removal Time	0.00	0.00	0.00	ns
$t_{MPWARNVM}$	Asynchronous Reset Minimum Pulse Width for the Control Logic	10.00	12.50	12.50	ns
$t_{MPWCLKNVM}$	Clock Minimum Pulse Width for the Control Logic	4.00	5.00	5.00	ns
$t_{FMAXCLKNVM}$	Maximum Frequency for Clock for the Control Logic – for AFS1500/AFS600	80.00	80.00	80.00	MHz
	Maximum Frequency for Clock for the Control Logic – for AFS250/AFS090	100.00	80.00	80.00	MHz

FlashROM

Fusion devices have 1 kbit of on-chip nonvolatile flash memory that can be read from the FPGA core fabric. The FlashROM is arranged in eight banks of 128 bits during programming. The 128 bits in each bank are addressable as 16 bytes during the read-back of the FlashROM from the FPGA core (Figure 2-45).

The FlashROM can only be programmed via the IEEE 1532 JTAG port. It cannot be programmed directly from the FPGA core. When programming, each of the eight 128-bit banks can be selectively reprogrammed. The FlashROM can only be reprogrammed on a bank boundary. Programming involves an automatic, on-chip bank erase prior to reprogramming the bank. The FlashROM supports a synchronous read and can be read on byte boundaries. The upper three bits of the FlashROM address from the FPGA core define the bank that is being accessed. The lower four bits of the FlashROM address from the FPGA core define which of the 16 bytes in the bank is being accessed.

The maximum FlashROM access clock is given in Table 2-26 on page 2-54. Figure 2-46 shows the timing behavior of the FlashROM access cycle—the address has to be set up on the rising edge of the clock for DOUT to be valid on the next falling edge of the clock.

If the address is unchanged for two cycles:

- D0 becomes invalid t_{CK2Q} ns after the second rising edge of the clock.
- D0 becomes valid again t_{CK2Q} ns after the second falling edge.

If the address unchanged for three cycles:

- D0 becomes invalid t_{CK2Q} ns after the second rising edge of the clock.
- D0 becomes valid again t_{CK2Q} ns after the second falling edge.
- D0 becomes invalid t_{CK2Q} ns after the third rising edge of the clock.
- D0 becomes valid again t_{CK2Q} ns after the third falling edge.

The following signals are used to configure the RAM4K9 memory element.

WIDTHA and WIDTHB

These signals enable the RAM to be configured in one of four allowable aspect ratios (Table 2-27).

Table 2-27 • Allowable Aspect Ratio Settings for WIDTHA[1:0]

WIDTHA1, WIDTHA0	WIDTHB1, WIDTHB0	D×W
00	00	4k×1
01	01	2k×2
10	10	1k×4
11	11	512×9
Note: The aspect ratio settings are constant and cannot be changed on the fly.		

BLKA and BLKB

These signals are active low and will enable the respective ports when asserted. When a BLKx signal is deasserted, the corresponding port's outputs hold the previous value.

WENA and WENB

These signals switch the RAM between read and write mode for the respective ports. A Low on these signals indicates a write operation, and a High indicates a read.

CLKA and CLKB

These are the clock signals for the synchronous read and write operations. These can be driven independently or with the same driver.

PIPEA and PIPEB

These signals are used to specify pipelined read on the output. A Low on PIPEA or PIPEB indicates a nonpipelined read, and the data appears on the corresponding output in the same clock cycle. A High indicates a pipelined, read and data appears on the corresponding output in the next clock cycle.

WMODEA and WMODEB

These signals are used to configure the behavior of the output when the RAM is in write mode. A Low on these signals makes the output retain data from the previous read. A High indicates pass-through behavior, wherein the data being written will appear immediately on the output. This signal is overridden when the RAM is being read.

RESET

This active low signal resets the output to zero, disables reads and writes from the SRAM block, and clears the data hold registers when asserted. It does not reset the contents of the memory.

ADDRA and ADDRB

These are used as read or write addresses, and they are 12 bits wide. When a depth of less than 4 k is specified, the unused high-order bits must be grounded (Table 2-28).

Table 2-28 • Address Pins Unused/Used for Various Supported Bus Widths

D×W	ADDRx	
	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.

SRAM Characteristics

Timing Waveforms

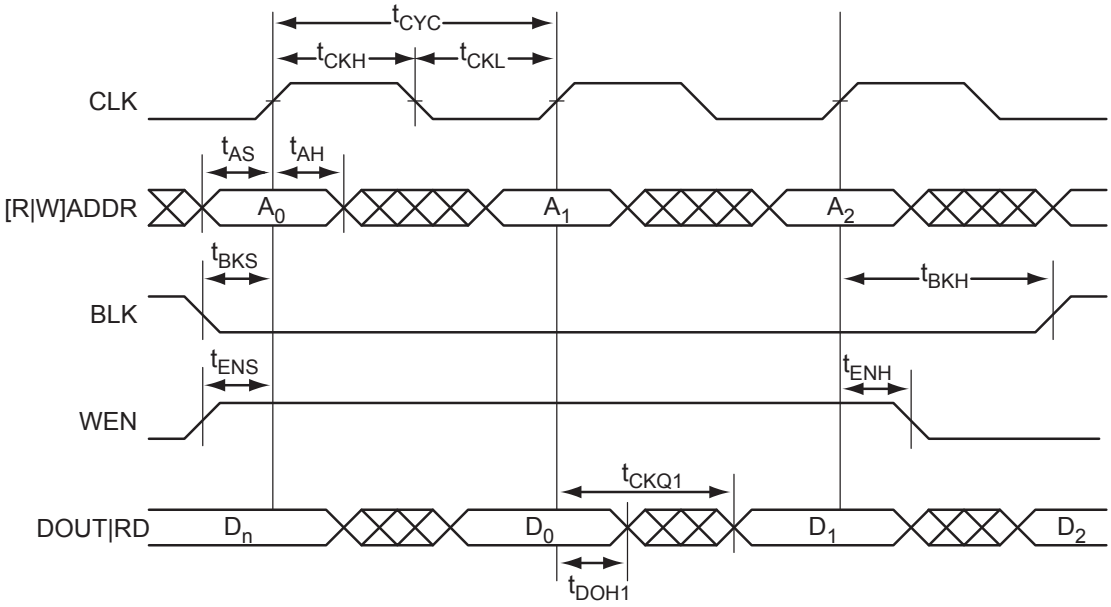


Figure 2-50 • **RAM Read for Flow-Through Output. Applicable to both RAM4K9 and RAM512x18.**

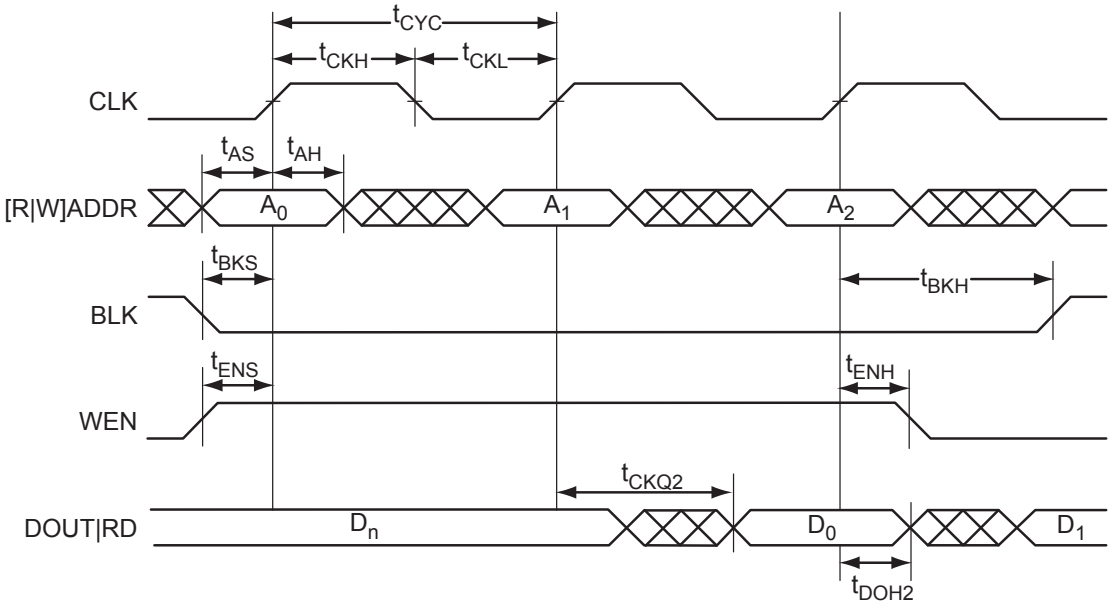


Figure 2-51 • **RAM Read for Pipelined Output. Applicable to both RAM4K9 and RAM512x18.**

EQ 16 through EQ 18 can be used to calculate the acquisition time required for a given input. The STC signal gives the number of sample periods in ADCCLK for the acquisition time of the desired signal. If the actual acquisition time is higher than the STC value, the settling time error can affect the accuracy of the ADC, because the sampling capacitor is only partially charged within the given sampling cycle. Example acquisition times are given in Table 2-44 and Table 2-45. When controlling the sample time for the ADC along with the use of the active bipolar prescaler, current monitor, or temperature monitor, the minimum sample time(s) for each must be obeyed. EQ 19 can be used to determine the appropriate value of STC.

You can calculate the minimum actual acquisition time by using EQ 16:

$$VOUT = VIN(1 - e^{-t/RC})$$

EQ 16

For 0.5 LSB gain error, VOUT should be replaced with (VIN –(0.5 × LSB Value)):

$$(VIN - 0.5 \times LSB\ Value) = VIN(1 - e^{-t/RC})$$

EQ 17

where VIN is the ADC reference voltage (VREF)

Solving EQ 17:

$$t = RC \times \ln (VIN / (0.5 \times LSB\ Value))$$

EQ 18

where $R = Z_{INAD} + R_{SOURCE}$ and $C = C_{INAD}$.

Calculate the value of STC by using EQ 19.

$$t_{SAMPLE} = (2 + STC) \times (1 / ADCCLK) \text{ or } t_{SAMPLE} = (2 + STC) \times (ADC\ Clock\ Period)$$

EQ 19

where ADCCLK = ADC clock frequency in MHz.

t_{SAMPLE} = 0.449 µs from bit resolution in Table 2-44.

ADC Clock frequency = 10 MHz or a 100 ns period.

STC = (t_{SAMPLE} / (1 / 10 MHz)) – 2 = 4.49 – 2 = 2.49.

You must round up to 3 to accommodate the minimum sample time.

***Table 2-44* • Acquisition Time Example with VAREF = 2.56 V**

VIN = 2.56V, R = 4K (R_{SOURCE} ~ 0), C = 18 pF		
Resolution	**LSB Value (mV)**	**Min. Sample/Hold Time for 0.5 LSB (µs)**
8	10	0.449
10	2.5	0.549
12	0.625	0.649

***Table 2-45* • Acquisition Time Example with VAREF = 3.3 V**

VIN = 3.3V, R = 4K (R_{SOURCE} ~ 0), C = 18 pF		
Resolution	**LSB Value (mV)**	**Min. Sample/Hold time for 0.5 LSB (µs)**
8	12.891	0.449
10	3.223	0.549
12	0.806	0.649

Sample Phase

A conversion is performed in three phases. In the first phase, the analog input voltage is sampled on the input capacitor. This phase is called sample phase. During the sample phase, the output signals BUSY and SAMPLE change from '0' to '1', indicating the ADC is busy and sampling the analog signal. The sample time can be controlled by input signals STC[7:0]. The sample time can be calculated by EQ 20. When controlling the sample time for the ADC along with the use of Prescaler or Current Monitor or Temperature Monitor, the minimum sample time for each must be obeyed.

***Table 2-49* • Analog Channel Specifications (continued)**
Commercial Temperature Range Conditions, T_J = 85°C (unless noted otherwise),
Typical: VCC33A = 3.3 V, VCC = 1.5 V

Parameter	Description	Condition	Min.	Typ.	Max.	Units
Digital Input using Analog Pads AV, AC and AT						
VIND[2,3]	Input Voltage	Refer to Table 3-2 on page 3-3				
VHYSDIN	Hysteresis			0.3		V
VIHDIN	Input High			1.2		V
VILDIN	Input Low			0.9		V
VMPWDIN	Minimum Pulse With		50			ns
F_{DIN}	Maximum Frequency				10	MHz
ISTBDIN	Input Leakage Current			2		µA
IDYNDIN	Dynamic Current			20		µA
t_{INDIN}	Input Delay			10		ns
Gate Driver Output Using Analog Pad AG						
VG	Voltage Range	Refer to Table 3-2 on page 3-3				
IG	Output Current Drive	High Current Mode[6] at 1.0 V			±20	mA
		Low Current Mode: ±1 µA	0.8	1.0	1.3	µA
		Low Current Mode: ±3 µA	2.0	2.7	3.3	µA
		Low Current Mode: ± 10 µA	7.4	9.0	11.5	µA
		Low Current Mode: ± 30 µA	21.0	27.0	32.0	µA
IOFFG	Maximum Off Current				100	nA
F_G	Maximum switching rate	High Current Mode[6] at 1.0 V, 1 kΩ resistive load		1.3		MHz
		Low Current Mode: ±1 µA, 3 MΩ resistive load		3		KHz
		Low Current Mode: ±3 µA, 1 MΩ resistive load		7		KHz
		Low Current Mode: ±10 µA, 300 kΩ resistive load		25		KHz
		Low Current Mode: ±30 µA, 105 kΩ resistive load		78		KHz

Notes:

1. *VRSM is the maximum voltage drop across the current sense resistor.*
2. *Analog inputs used as digital inputs can tolerate the same voltage limits as the corresponding analog pad. There is no reliability concern on digital inputs as long as VIND does not exceed these limits.*
3. *VIND is limited to VCC33A + 0.2 to allow reaching 10 MHz input frequency.*
4. *An averaging of 1,024 samples (LPF setting in Analog System Builder) is required and the maximum capacitance allowed across the AT pins is 500 pF.*
5. *The temperature offset is a fixed positive value.*
6. *The high current mode has a maximum power limit of 20 mW. Appropriate current limit resistors must be used, based on voltage on the pad.*
7. *When using SmartGen Analog System Builder, CalibIP is required to obtain specified offset. For further details on CalibIP, refer to the "Temperature, Voltage, and Current Calibration in Fusion FPGAs" chapter of the* Fusion FPGA Fabric User Guide.

Table 2-121 • 1.8 V LVCMOS High Slew
Commercial Temperature Range Conditions: T_J = 70°C, Worst-Case VCC = 1.425 V, Worst-Case VCCI = 1.7 V
Applicable to Pro I/Os

Drive Strength	Speed Grade	t_{DOUT}	t_{DP}	t_{DIN}	t_{PY}	t_{PYS}	t_{EOUT}	t_{ZL}	t_{ZH}	t_{LZ}	t_{HZ}	t_{ZLS}	t_{ZHS}	Units
2 mA	Std.	0.66	12.10	0.04	1.45	1.91	0.43	9.59	12.10	2.78	1.64	11.83	14.34	ns
	–1	0.56	10.30	0.04	1.23	1.62	0.36	8.16	10.30	2.37	1.39	10.06	12.20	ns
	–2	0.49	9.04	0.03	1.08	1.42	0.32	7.16	9.04	2.08	1.22	8.83	10.71	ns
4 mA	Std.	0.66	7.05	0.04	1.45	1.91	0.43	6.20	7.05	3.25	2.86	8.44	9.29	ns
	–1	0.56	6.00	0.04	1.23	1.62	0.36	5.28	6.00	2.76	2.44	7.18	7.90	ns
	–2	0.49	5.27	0.03	1.08	1.42	0.32	4.63	5.27	2.43	2.14	6.30	6.94	ns
8 mA	Std.	0.66	4.52	0.04	1.45	1.91	0.43	4.47	4.52	3.57	3.47	6.70	6.76	ns
	–1	0.56	3.85	0.04	1.23	1.62	0.36	3.80	3.85	3.04	2.95	5.70	5.75	ns
	–2	0.49	3.38	0.03	1.08	1.42	0.32	3.33	3.38	2.66	2.59	5.00	5.05	ns
12 mA	Std.	0.66	4.12	0.04	1.45	1.91	0.43	4.20	3.99	3.63	3.62	6.43	6.23	ns
	–1	0.56	3.51	0.04	1.23	1.62	0.36	3.57	3.40	3.09	3.08	5.47	5.30	ns
	–2	0.49	3.08	0.03	1.08	1.42	0.32	3.14	2.98	2.71	2.71	4.81	4.65	ns
16 mA	Std.	0.66	3.80	0.04	1.45	1.91	0.43	3.87	3.09	3.73	4.24	6.10	5.32	ns
	–1	0.56	3.23	0.04	1.23	1.62	0.36	3.29	2.63	3.18	3.60	5.19	4.53	ns
	–2	0.49	2.83	0.03	1.08	1.42	0.32	2.89	2.31	2.79	3.16	4.56	3.98	ns

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

Output Register

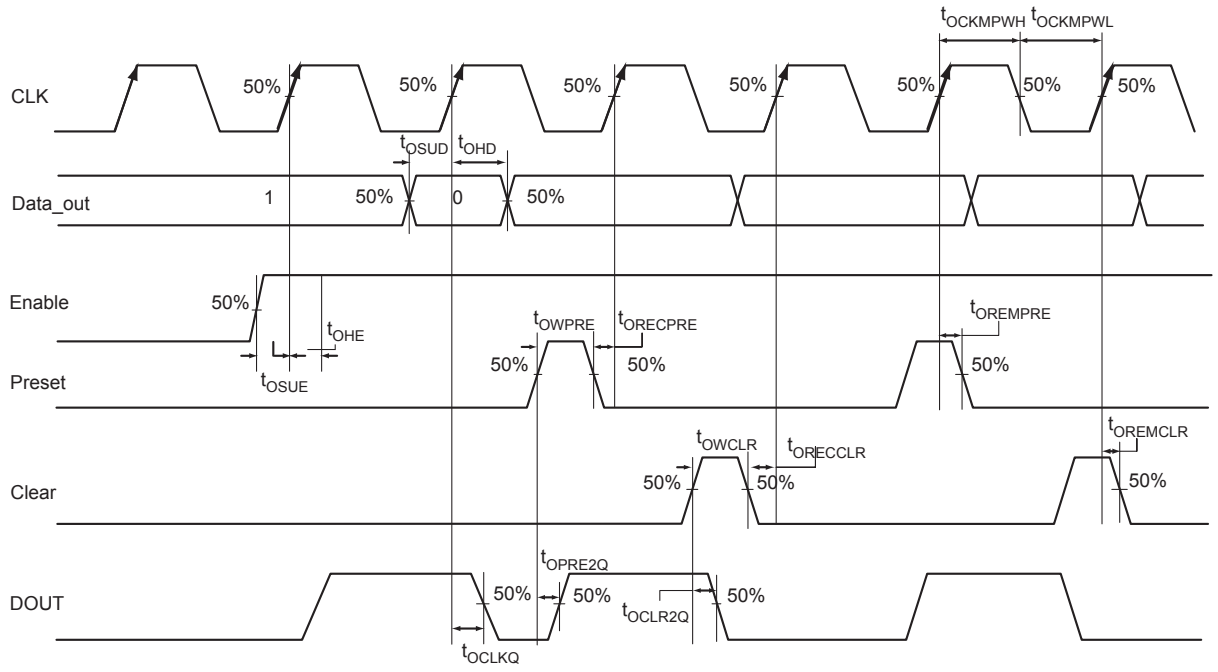


Figure 2-140 • **Output Register Timing Diagram**

Timing Characteristics

Table 2-177 • **Output Data Register Propagation Delays**
Commercial Temperature Range Conditions: T_J = 70°C, Worst-Case VCC = 1.425 V

Parameter	Description	–2	–1	Std.	Units
t_{OCLKQ}	Clock-to-Q of the Output Data Register	0.59	0.67	0.79	ns
t_{OSUD}	Data Setup Time for the Output Data Register	0.31	0.36	0.42	ns
t_{OHD}	Data Hold Time for the Output Data Register	0.00	0.00	0.00	ns
t_{OSUE}	Enable Setup Time for the Output Data Register	0.44	0.50	0.59	ns
t_{OHE}	Enable Hold Time for the Output Data Register	0.00	0.00	0.00	ns
t_{OCLR2Q}	Asynchronous Clear-to-Q of the Output Data Register	0.80	0.91	1.07	ns
t_{OPRE2Q}	Asynchronous Preset-to-Q of the Output Data Register	0.80	0.91	1.07	ns
$t_{OREMCLR}$	Asynchronous Clear Removal Time for the Output Data Register	0.00	0.00	0.00	ns
$t_{ORECCLR}$	Asynchronous Clear Recovery Time for the Output Data Register	0.22	0.25	0.30	ns
$t_{OREMPRE}$	Asynchronous Preset Removal Time for the Output Data Register	0.00	0.00	0.00	ns
$t_{ORECPRE}$	Asynchronous Preset Recovery Time for the Output Data Register	0.22	0.25	0.30	ns
t_{OWCLR}	Asynchronous Clear Minimum Pulse Width for the Output Data Register	0.22	0.25	0.30	ns
t_{OWPRE}	Asynchronous Preset Minimum Pulse Width for the Output Data Register	0.22	0.25	0.30	ns
$t_{OCKMPWH}$	Clock Minimum Pulse Width High for the Output Data Register	0.36	0.41	0.48	ns
$t_{OCKMPWL}$	Clock Minimum Pulse Width Low for the Output Data Register	0.32	0.37	0.43	ns

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

Output DDR

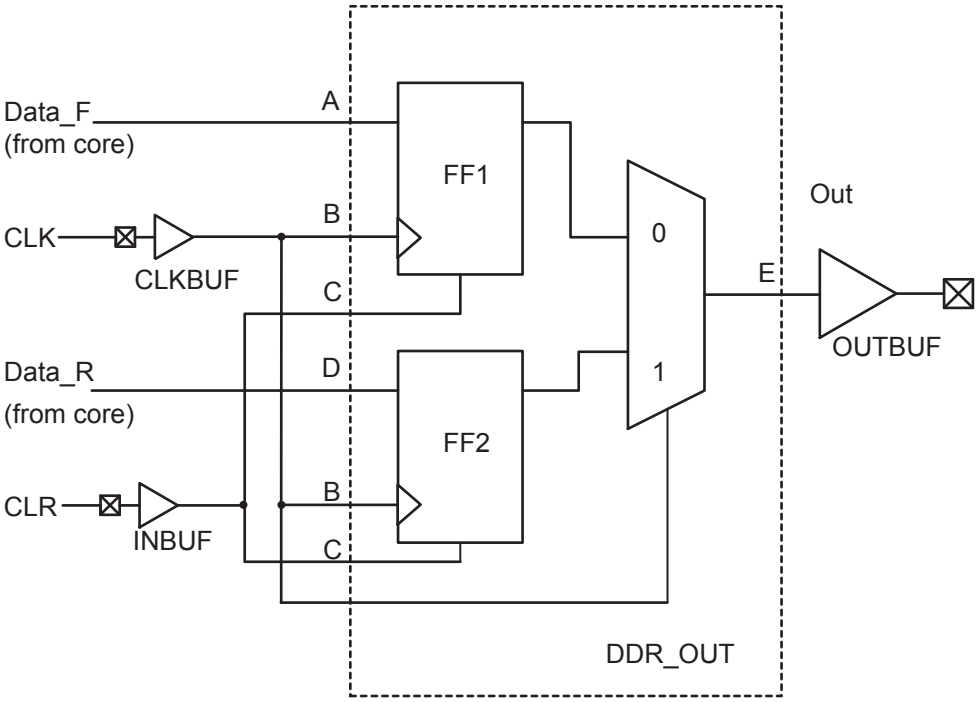


Figure 2-144 • **Output DDR Timing Model**

Table 2-181 • **Parameter Definitions**

Parameter Name	Parameter Definition	Measuring Nodes (From, To)
$t_{DDROCLKQ}$	Clock-to-Out	B, E
$t_{DDROCLR2Q}$	Asynchronous Clear-to-Out	C, E
$t_{DDROREMCLR}$	Clear Removal	C, B
$t_{DDRORECCLR}$	Clear Recovery	C, B
$t_{DDROSUD1}$	Data Setup Data_F	A, B
$t_{DDROSUD2}$	Data Setup Data_R	D, B
$t_{DDROHD1}$	Data Hold Data_F	A, B
$t_{DDROHD2}$	Data Hold Data_R	D, B

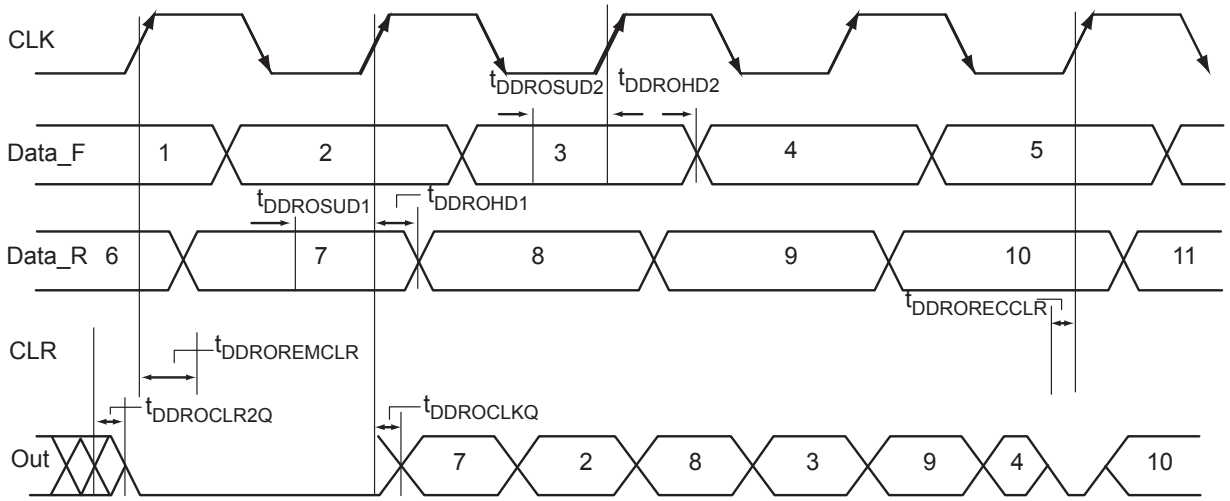


Figure 2-145 • **Output DDR Timing Diagram**

Timing Characteristics

Table 2-182 • **Output DDR Propagation Delays**
Commercial Temperature Range Conditions: T_J = 70°C, Worst-Case VCC = 1.425 V

Parameter	Description	–2	–1	Std.	Units
$t_{DDROCLKQ}$	Clock-to-Out of DDR for Output DDR	0.70	0.80	0.94	ns
$t_{DDROSUD1}$	Data_F Data Setup for Output DDR	0.38	0.43	0.51	ns
$t_{DDROSUD2}$	Data_R Data Setup for Output DDR	0.38	0.43	0.51	ns
$t_{DDROHD1}$	Data_F Data Hold for Output DDR	0.00	0.00	0.00	ns
$t_{DDROHD2}$	Data_R Data Hold for Output DDR	0.00	0.00	0.00	ns
$t_{DDROCLR2Q}$	Asynchronous Clear-to-Out for Output DDR	0.80	0.91	1.07	ns
$t_{DDROREMCLR}$	Asynchronous Clear Removal Time for Output DDR	0.00	0.00	0.00	ns
$t_{DDRORECCLR}$	Asynchronous Clear Recovery Time for Output DDR	0.22	0.25	0.30	ns
$t_{DDROWCLR1}$	Asynchronous Clear Minimum Pulse Width for Output DDR	0.22	0.25	0.30	ns
$t_{DDROCKMPWH}$	Clock Minimum Pulse Width High for the Output DDR	0.36	0.41	0.48	ns
$t_{DDROCKMPWL}$	Clock Minimum Pulse Width Low for the Output DDR	0.32	0.37	0.43	ns
F_{DDOMAX}	Maximum Frequency for the Output DDR	1404	1232	1048	MHz

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

Table 3-13 • **Summary of I/O Output Buffer Power (per pin)—Default I/O Software Settings[1]**

	C_{LOAD} (pF)	VCCI (V)	Static Power PDC8 (mW)[2]	Dynamic Power PAC10 (µW/MHz)[3]
Applicable to Pro I/O Banks				
Single-Ended				
3.3 V LVTTL/LVCMOS	35	3.3	–	474.70
2.5 V LVCMOS	35	2.5	–	270.73
1.8 V LVCMOS	35	1.8	–	151.78
1.5 V LVCMOS (JESD8-11)	35	1.5	–	104.55
3.3 V PCI	10	3.3	–	204.61
3.3 V PCI-X	10	3.3	–	204.61
Voltage-Referenced				
3.3 V GTL	10	3.3	–	24.08
2.5 V GTL	10	2.5	–	13.52
3.3 V GTL+	10	3.3	–	24.10
2.5 V GTL+	10	2.5	–	13.54
HSTL (I)	20	1.5	7.08	26.22
HSTL (II)	20	1.5	13.88	27.22
SSTL2 (I)	30	2.5	16.69	105.56
SSTL2 (II)	30	2.5	25.91	116.60
SSTL3 (I)	30	3.3	26.02	114.87
SSTL3 (II)	30	3.3	42.21	131.76
Differential				
LVDS	–	2.5	7.70	89.62
LVPECL	–	3.3	19.42	168.02
Applicable to Advanced I/O Banks				
Single-Ended				
3.3 V LVTTL / 3.3 V LVCMOS	35	3.3	–	468.67
2.5 V LVCMOS	35	2.5	–	267.48
1.8 V LVCMOS	35	1.8	–	149.46
1.5 V LVCMOS (JESD8-11)	35	1.5	–	103.12
3.3 V PCI	10	3.3	–	201.02
3.3 V PCI-X	10	3.3	–	201.02

Notes:

1. *Dynamic power consumption is given for standard load and software-default drive strength and output slew.*
2. *PDC8 is the static power (where applicable) measured on VCCI.*
3. *PAC10 is the total dynamic power measured on VCC and VCCI.*

Dynamic Power Consumption of Various Internal Resources

***Table 3-14* • Different Components Contributing to the Dynamic Power Consumption in Fusion Devices**

Parameter	Definition	Power Supply		Device-Specific Dynamic Contributions				Units
		Name	Setting	AFS1500	AFS600	AFS250	AFS090	
PAC1	Clock contribution of a Global Rib	VCC	1.5 V	14.5	12.8	11	11	µW/MHz
PAC2	Clock contribution of a Global Spine	VCC	1.5 V	2.5	1.9	1.6	0.8	µW/MHz
PAC3	Clock contribution of a VersaTile row	VCC	1.5 V	0.81				µW/MHz
PAC4	Clock contribution of a VersaTile used as a sequential module	VCC	1.5 V	0.11				µW/MHz
PAC5	First contribution of a VersaTile used as a sequential module	VCC	1.5 V	0.07				µW/MHz
PAC6	Second contribution of a VersaTile used as a sequential module	VCC	1.5 V	0.29				µW/MHz
PAC7	Contribution of a VersaTile used as a combinatorial module	VCC	1.5 V	0.29				µW/MHz
PAC8	Average contribution of a routing net	VCC	1.5 V	0.70				µW/MHz
PAC9	Contribution of an I/O input pin (standard dependent)	VCCI	See Table 3-12 on page 3-18					
PAC10	Contribution of an I/O output pin (standard dependent)	VCCI	See Table 3-13 on page 3-20					
PAC11	Average contribution of a RAM block during a read operation	VCC	1.5 V	25				µW/MHz
PAC12	Average contribution of a RAM block during a write operation	VCC	1.5 V	30				µW/MHz
PAC13	Dynamic Contribution for PLL	VCC	1.5 V	2.6				µW/MHz
PAC15	Contribution of NVM block during a read operation (F < 33MHz)	VCC	1.5 V	358				µW/MHz
PAC16	1st contribution of NVM block during a read operation (F > 33 MHz)	VCC	1.5 V	12.88				mW
PAC17	2nd contribution of NVM block during a read operation (F > 33 MHz)	VCC	1.5 V	4.8				µW/MHz
PAC18	Crystal Oscillator contribution	VCC33A	3.3 V	0.63				mW
PAC19	RC Oscillator contribution	VCC33A	3.3 V	3.3				mW
PAC20	Analog Block dynamic power contribution of ADC	VCC	1.5 V	3				mW

$P_{S\text{-}CELL}$ = 0 W

$P_{C\text{-}CELL}$ = 0 W

P_{NET} = 0 W

P_{LOGIC} = 0 W

I/O Input and Output Buffer Contribution—$P_{I/O}$

This example uses LVTTL 3.3 V I/O cells. The output buffers are 12 mA–capable, configured with high output slew and driving a 35 pF output load.

F_{CLK} = 50 MHz

Number of input pins used: N_{INPUTS} = 30

Number of output pins used: $N_{OUTPUTS}$ = 40

Estimated I/O buffer toggle rate: α_2 = 0.1 (10%)

Estimated IO buffer enable rate: β_1 = 1 (100%)

Operating Mode

$P_{INPUTS} = N_{INPUTS} * (\alpha_2 / 2) * PAC9 * F_{CLK}$

P_{INPUTS} = 30 * (0.1 / 2) * 0.01739 * 50

P_{INPUTS} = 1.30 mW

$P_{OUTPUTS} = N_{OUTPUTS} * (\alpha_2 / 2) * \beta_1 * PAC10 * F_{CLK}$

$P_{OUTPUTS}$ = 40 * (0.1 / 2) * 1 * 0.4747 * 50

$P_{OUTPUTS}$ = 47.47 mW

$P_{I/O} = P_{INPUTS} + P_{OUTPUTS}$

$P_{I/O}$ = 1.30 mW + 47.47 mW

$P_{I/O}$ = 48.77 mW

Standby Mode and Sleep Mode

P_{INPUTS} = 0 W

$P_{OUTPUTS}$ = 0 W

$P_{I/O}$ = 0 W

RAM Contribution—P_{MEMORY}

Frequency of Read Clock: $F_{READ\text{-}CLOCK}$ = 10 MHz

Frequency of Write Clock: $F_{WRITE\text{-}CLOCK}$ = 10 MHz

Number of RAM blocks: N_{BLOCKS} = 20

Estimated RAM Read Enable Rate: β_2 = 0.125 (12.5%)

Estimated RAM Write Enable Rate: β_3 = 0.125 (12.5%)

Operating Mode

$P_{MEMORY} = (N_{BLOCKS} * PAC11 * \beta_2 * F_{READ\text{-}CLOCK}) + (N_{BLOCKS} * PAC12 * \beta_3 * F_{WRITE\text{-}CLOCK})$

P_{MEMORY} = (20 * 0.025 * 0.125 * 10) + (20 * 0.030 * 0.125 * 10)

P_{MEMORY} = 1.38 mW

Standby Mode and Sleep Mode

P_{MEMORY} = 0 W

FG484		
Pin Number	AFS600 Function	AFS1500 Function
E9	NC	IO08PDB0V1
E10	GND	GND
E11	IO15NDB1V0	IO22NDB1V0
E12	IO15PDB1V0	IO22PDB1V0
E13	GND	GND
E14	NC	IO32PPB1V1
E15	NC	IO36NPB1V2
E16	VCCIB1	VCCIB1
E17	GND	GND
E18	NC	IO47NPB2V0
E19	IO33PDB2V0	IO49PDB2V0
E20	VCCIB2	VCCIB2
E21	IO32NDB2V0	IO46NDB2V0
E22	GBC2/IO32PDB2V0	GBC2/IO46PDB2V0
F1	IO80NDB4V0	IO118NDB4V0
F2	IO80PDB4V0	IO118PDB4V0
F3	NC	IO119NSB4V0
F4	IO84NDB4V0	IO124NDB4V0
F5	GND	GND
F6	VCOMPLA	VCOMPLA
F7	VCCPLA	VCCPLA
F8	VCCIB0	VCCIB0
F9	IO08NDB0V1	IO12NDB0V1
F10	IO08PDB0V1	IO12PDB0V1
F11	VCCIB0	VCCIB0
F12	VCCIB1	VCCIB1
F13	IO22NDB1V0	IO30NDB1V1
F14	IO22PDB1V0	IO30PDB1V1
F15	VCCIB1	VCCIB1
F16	NC	IO36PPB1V2
F17	NC	IO38NPB1V2
F18	GND	GND
F19	IO33NDB2V0	IO49NDB2V0
F20	IO34PDB2V0	IO50PDB2V0
F21	IO34NDB2V0	IO50NDB2V0

FG484		
Pin Number	AFS600 Function	AFS1500 Function
F22	IO35PDB2V0	IO51PDB2V0
G1	IO77PDB4V0	IO115PDB4V0
G2	GND	GND
G3	IO78NDB4V0	IO116NDB4V0
G4	IO78PDB4V0	IO116PDB4V0
G5	VCCIB4	VCCIB4
G6	NC	IO117PDB4V0
G7	VCCIB4	VCCIB4
G8	GND	GND
G9	IO04NDB0V0	IO06NDB0V1
G10	IO04PDB0V0	IO06PDB0V1
G11	IO12NDB0V1	IO16NDB0V2
G12	IO12PDB0V1	IO16PDB0V2
G13	NC	IO28NDB1V1
G14	NC	IO28PDB1V1
G15	GND	GND
G16	NC	IO38PPB1V2
G17	NC	IO53PDB2V0
G18	VCCIB2	VCCIB2
G19	IO36PDB2V0	IO52PDB2V0
G20	IO36NDB2V0	IO52NDB2V0
G21	GND	GND
G22	IO35NDB2V0	IO51NDB2V0
H1	IO77NDB4V0	IO115NDB4V0
H2	IO76PDB4V0	IO113PDB4V0
H3	VCCIB4	VCCIB4
H4	IO79NDB4V0	IO114NDB4V0
H5	IO79PDB4V0	IO114PDB4V0
H6	NC	IO117NDB4V0
H7	GND	GND
H8	VCC	VCC
H9	VCCIB0	VCCIB0
H10	GND	GND
H11	VCCIB0	VCCIB0
H12	VCCIB1	VCCIB1