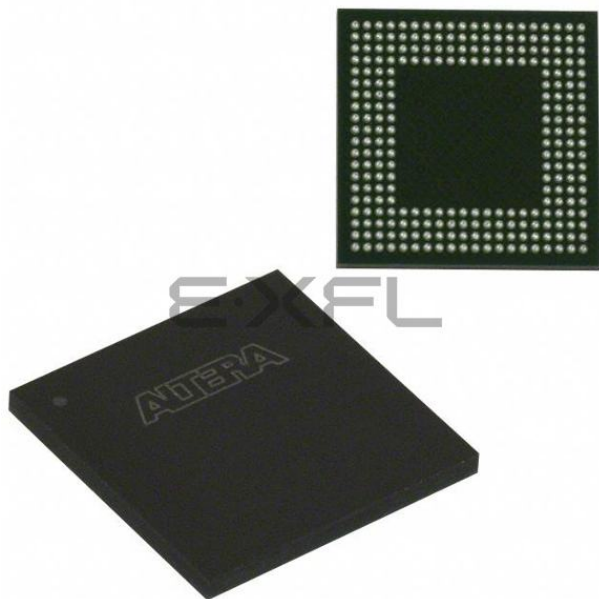




Welcome to [E-XFL.COM](https://www.e-xfl.com)

Understanding Embedded - CPLDs (Complex Programmable Logic Devices)

Embedded - CPLDs, or Complex Programmable Logic Devices, are highly versatile digital logic devices used in electronic systems. These programmable components are designed to perform complex logical operations and can be customized for specific applications. Unlike fixed-function ICs, CPLDs offer the flexibility to reprogram their configuration, making them an ideal choice for various embedded systems. They consist of a set of logic gates and programmable interconnects, allowing designers to implement complex logic circuits without needing custom hardware.

Applications of Embedded - CPLDs

Details

Product Status	Active
Programmable Type	In System Programmable
Delay Time tpd(1) Max	5.4 ns
Voltage Supply - Internal	1.71V ~ 1.89V
Number of Logic Elements/Blocks	570
Number of Macrocells	440
Number of Gates	-
Number of I/O	160
Operating Temperature	-40°C ~ 100°C (TJ)
Mounting Type	Surface Mount
Package / Case	256-TFBGA
Supplier Device Package	256-MBGA (11x11)
Purchase URL	https://www.e-xfl.com/product-detail/intel/epm570gm256i5n

Introduction

The MAX[®] II family of instant-on, non-volatile CPLDs is based on a 0.18- μ m, 6-layer-metal-flash process, with densities from 240 to 2,210 logic elements (LEs) (128 to 2,210 equivalent macrocells) and non-volatile storage of 8 Kbits. MAX II devices offer high I/O counts, fast performance, and reliable fitting versus other CPLD architectures. Featuring MultiVolt core, a user flash memory (UFM) block, and enhanced in-system programmability (ISP), MAX II devices are designed to reduce cost and power while providing programmable solutions for applications such as bus bridging, I/O expansion, power-on reset (POR) and sequencing control, and device configuration control.

Features

The MAX II CPLD has the following features:

- Low-cost, low-power CPLD
- Instant-on, non-volatile architecture
- Standby current as low as 25 μ A
- Provides fast propagation delay and clock-to-output times
- Provides four global clocks with two clocks available per logic array block (LAB)
- UFM block up to 8 Kbits for non-volatile storage
- MultiVolt core enabling external supply voltages to the device of either 3.3 V/2.5 V or 1.8 V
- MultiVolt I/O interface supporting 3.3-V, 2.5-V, 1.8-V, and 1.5-V logic levels
- Bus-friendly architecture including programmable slew rate, drive strength, bus-hold, and programmable pull-up resistors
- Schmitt triggers enabling noise tolerant inputs (programmable per pin)
- I/Os are fully compliant with the Peripheral Component Interconnect Special Interest Group (PCI SIG) PCI Local Bus Specification, Revision 2.2 for 3.3-V operation at 66 MHz
- Supports hot-socketing
- Built-in Joint Test Action Group (JTAG) boundary-scan test (BST) circuitry compliant with IEEE Std. 1149.1-1990
- ISP circuitry compliant with IEEE Std. 1532

MAX II devices are available in space-saving FineLine BGA, Micro FineLine BGA, and thin quad flat pack (TQFP) packages (refer to [Table 1-3](#) and [Table 1-4](#)). MAX II devices support vertical migration within the same package (for example, you can migrate between the EPM570, EPM1270, and EPM2210 devices in the 256-pin FineLine BGA package). Vertical migration means that you can migrate to devices whose dedicated pins and JTAG pins are the same and power pins are subsets or supersets for a given package across device densities. The largest density in any package has the highest number of power pins; you must lay out for the largest planned density in a package to provide the necessary power pins for migration. For I/O pin migration across densities, cross reference the available I/O pins using the device pin-outs for all planned densities of a given package type to identify which I/O pins can be migrated. The Quartus® II software can automatically cross-reference and place all pins for you when given a device migration list.

Table 1-3. MAX II Packages and User I/O Pins

Device	68-Pin Micro FineLine BGA (1)	100-Pin Micro FineLine BGA (1)	100-Pin FineLine BGA	100-Pin TQFP	144-Pin TQFP	144-Pin Micro FineLine BGA (1)	256-Pin Micro FineLine BGA (1)	256-Pin FineLine BGA	324-Pin FineLine BGA
EPM240 EPM240G	—	80	80	80	—	—	—	—	—
EPM570 EPM570G	—	76	76	76	116	—	160	160	—
EPM1270 EPM1270G	—	—	—	—	116	—	212	212	—
EPM2210 EPM2210G	—	—	—	—	—	—	—	204	272
EPM240Z	54	80	—	—	—	—	—	—	—
EPM570Z	—	76	—	—	—	116	160	—	—

Note to Table 1-3:

(1) Packages available in lead-free versions only.

Table 1-4. MAX II TQFP, FineLine BGA, and Micro FineLine BGA Package Sizes

Package	68-Pin Micro FineLine BGA	100-Pin Micro FineLine BGA	100-Pin FineLine BGA	100-Pin TQFP	144-Pin TQFP	144-Pin Micro FineLine BGA	256-Pin Micro FineLine BGA	256-Pin FineLine BGA	324-Pin FineLine BGA
Pitch (mm)	0.5	0.5	1	0.5	0.5	0.5	0.5	1	1
Area (mm ²)	25	36	121	256	484	49	121	289	361
Length × width (mm × mm)	5 × 5	6 × 6	11 × 11	16 × 16	22 × 22	7 × 7	11 × 11	17 × 17	19 × 19

MAX II devices have an internal linear voltage regulator which supports external supply voltages of 3.3 V or 2.5 V, regulating the supply down to the internal operating voltage of 1.8 V. MAX IIG and MAX IIZ devices only accept 1.8 V as the external supply voltage. MAX IIZ devices are pin-compatible with MAX IIG devices in the 100-pin Micro FineLine BGA and 256-pin Micro FineLine BGA packages. Except for external supply voltage requirements, MAX II and MAX II G devices have identical pin-outs and timing specifications. Table 1-5 shows the external supply voltages supported by the MAX II family.

Table 1-5. MAX II External Supply Voltages

Devices	EPM240 EPM570 EPM1270 EPM2210	EPM240G EPM570G EPM1270G EPM2210G EPM240Z EPM570Z (1)
MultiVolt core external supply voltage (V_{CCINT}) (2)	3.3 V, 2.5 V	1.8 V
MultiVolt I/O interface voltage levels (V_{CCIO})	1.5 V, 1.8 V, 2.5 V, 3.3 V	1.5 V, 1.8 V, 2.5 V, 3.3 V

Notes to Table 1-5:

- (1) MAX IIG and MAX IIZ devices only accept 1.8 V on their VCCINT pins. The 1.8-V V_{CCINT} external supply powers the device core directly.
- (2) MAX II devices operate internally at 1.8 V.

Referenced Documents

This chapter references the following documents:

- *DC and Switching Characteristics* chapter in the *MAX II Device Handbook*
- *MAX II Logic Element to Macrocell Conversion Methodology* white paper

Document Revision History

Table 1-6 shows the revision history for this chapter.

Table 1-6. Document Revision History

Date and Revision	Changes Made	Summary of Changes
August 2009, version 1.9	■ Updated Table 1-2.	Added information for speed grade -8
October 2008, version 1.8	■ Updated “Introduction” section. ■ Updated new Document Format.	—
December 2007, version 1.7	■ Updated Table 1-1 through Table 1-5. ■ Added “Referenced Documents” section.	Updated document with MAX IIZ information.
December 2006, version 1.6	■ Added document revision history.	—
August 2006, version 1.5	■ Minor update to features list.	—
July 2006, version 1.4	■ Minor updates to tables.	—

Introduction

This chapter describes the architecture of the MAX II device and contains the following sections:

- “Functional Description” on page 2–1
- “Logic Array Blocks” on page 2–4
- “Logic Elements” on page 2–6
- “MultiTrack Interconnect” on page 2–12
- “Global Signals” on page 2–16
- “User Flash Memory Block” on page 2–18
- “MultiVolt Core” on page 2–22
- “I/O Structure” on page 2–23

Functional Description

MAX® II devices contain a two-dimensional row- and column-based architecture to implement custom logic. Row and column interconnects provide signal interconnects between the logic array blocks (LABs).

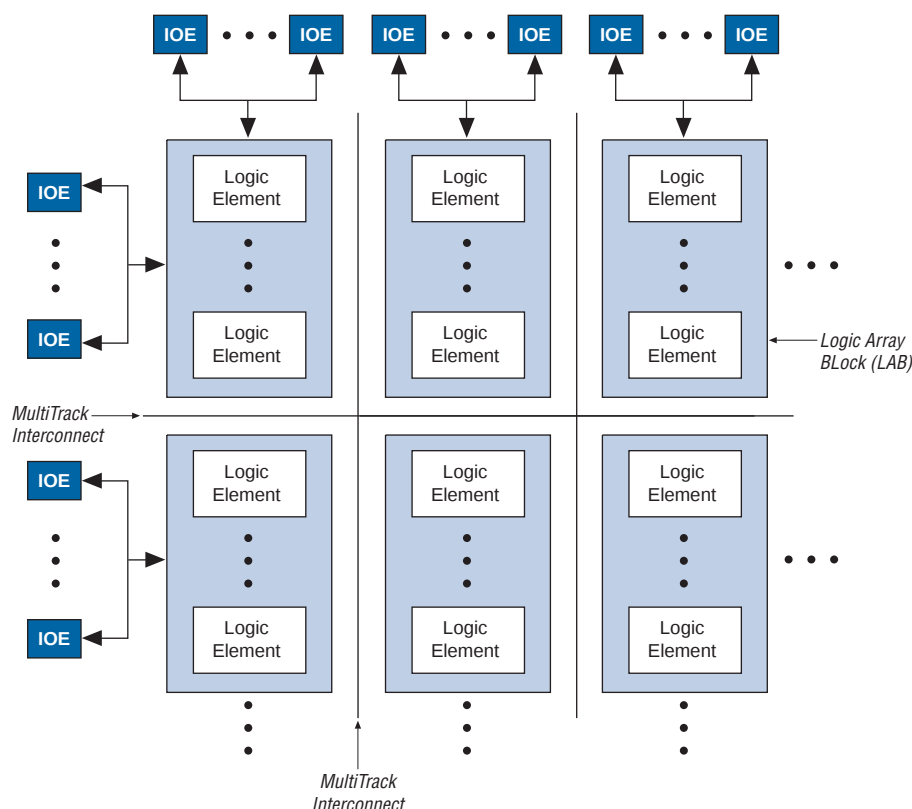
The logic array consists of LABs, with 10 logic elements (LEs) in each LAB. An LE is a small unit of logic providing efficient implementation of user logic functions. LABs are grouped into rows and columns across the device. The MultiTrack interconnect provides fast granular timing delays between LABs. The fast routing between LEs provides minimum timing delay for added levels of logic versus globally routed interconnect structures.

The MAX II device I/O pins are fed by I/O elements (IOE) located at the ends of LAB rows and columns around the periphery of the device. Each IOE contains a bidirectional I/O buffer with several advanced features. I/O pins support Schmitt trigger inputs and various single-ended standards, such as 66-MHz, 32-bit PCI, and LVTTTL.

MAX II devices provide a global clock network. The global clock network consists of four global clock lines that drive throughout the entire device, providing clocks for all resources within the device. The global clock lines can also be used for control signals such as clear, preset, or output enable.

Figure 2–1 shows a functional block diagram of the MAX II device.

Figure 2–1. MAX II Device Block Diagram



Each MAX II device contains a flash memory block within its floorplan. On the EPM240 device, this block is located on the left side of the device. On the EPM570, EPM1270, and EPM2210 devices, the flash memory block is located on the bottom-left area of the device. The majority of this flash memory storage is partitioned as the dedicated configuration flash memory (CFM) block. The CFM block provides the non-volatile storage for all of the SRAM configuration information. The CFM automatically downloads and configures the logic and I/O at power-up, providing instant-on operation.



For more information about configuration upon power-up, refer to the *Hot Socketing and Power-On Reset in MAX II Devices* chapter in the *MAX II Device Handbook*.

A portion of the flash memory within the MAX II device is partitioned into a small block for user data. This user flash memory (UFM) block provides 8,192 bits of general-purpose user storage. The UFM provides programmable port connections to the logic array for reading and writing. There are three LAB rows adjacent to this block, with column numbers varying by device.

Table 2–1 shows the number of LAB rows and columns in each device, as well as the number of LAB rows and columns adjacent to the flash memory area in the EPM570, EPM1270, and EPM2210 devices. The long LAB rows are full LAB rows that extend from one side of row I/O blocks to the other. The short LAB rows are adjacent to the UFM block; their length is shown as width in LAB columns.

Table 2-1. MAX II Device Resources

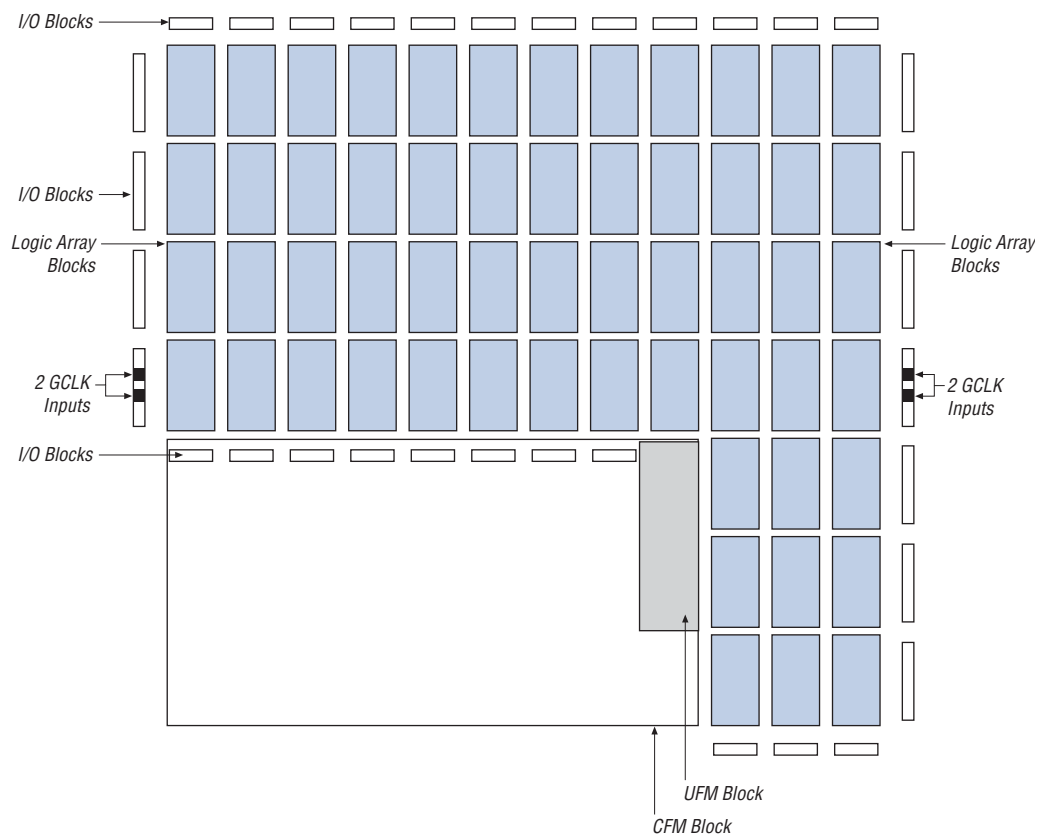
Devices	UFM Blocks	LAB Columns	LAB Rows		Total LABs
			Long LAB Rows	Short LAB Rows (Width) (1)	
EPM240	1	6	4	—	24
EPM570	1	12	4	3 (3)	57
EPM1270	1	16	7	3 (5)	127
EPM2210	1	20	10	3 (7)	221

Note to Table 2-1:

(1) The width is the number of LAB columns in length.

Figure 2-2 shows a floorplan of a MAX II device.

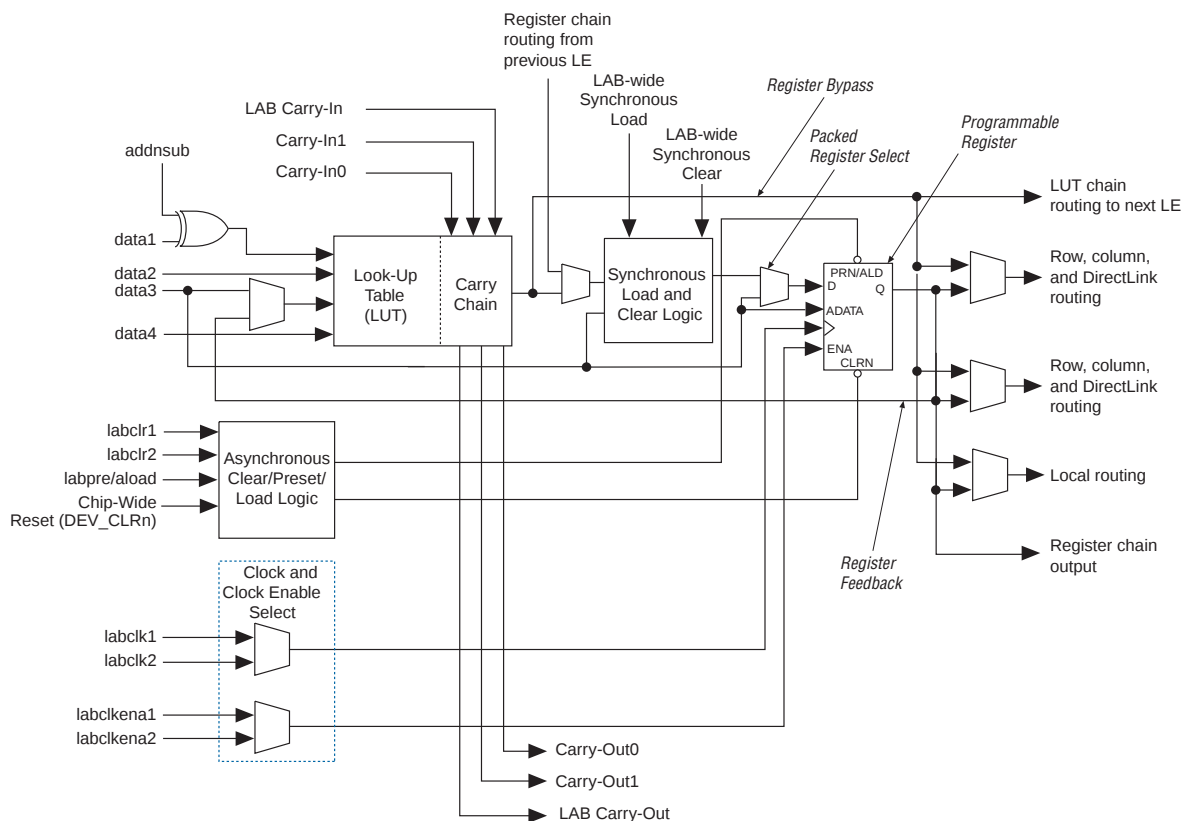
Figure 2-2. MAX II Device Floorplan (Note 1)



Note to Figure 2-2:

(1) The device shown is an EPM570 device. EPM1270 and EPM2210 devices have a similar floorplan with more LABs. For EPM240 devices, the CFM and UFM blocks are located on the left side of the device.

Figure 2-6. MAX II LE



Each LE's programmable register can be configured for D, T, JK, or SR operation. Each register has data, true asynchronous load data, clock, clock enable, clear, and asynchronous load/preset inputs. Global signals, general-purpose I/O pins, or any LE can drive the register's clock and clear control signals. Either general-purpose I/O pins or LEs can drive the clock enable, preset, asynchronous load, and asynchronous data. The asynchronous load data input comes from the data3 input of the LE. For combinational functions, the LUT output bypasses the register and drives directly to the LE outputs.

Each LE has three outputs that drive the local, row, and column routing resources. The LUT or register output can drive these three outputs independently. Two LE outputs drive column or row and DirectLink routing connections and one drives local interconnect resources. This allows the LUT to drive one output while the register drives another output. This register packing feature improves device utilization because the device can use the register and the LUT for unrelated functions. Another special packing mode allows the register output to feed back into the LUT of the same LE so that the register is packed with its own fan-out LUT. This provides another mechanism for improved fitting. The LE can also drive out registered and unregistered versions of the LUT output.

LUT Chain and Register Chain

In addition to the three general routing outputs, the LEs within an LAB have LUT chain and register chain outputs. LUT chain connections allow LUTs within the same LAB to cascade together for wide input functions. Register chain outputs allow registers within the same LAB to cascade together. The register chain output allows an LAB to use LUTs for a single combinational function and the registers to be used for an unrelated shift register implementation. These resources speed up connections between LABs while saving local interconnect resources. Refer to “[MultiTrack Interconnect](#)” on page 2-12 for more information about LUT chain and register chain connections.

addnsub Signal

The LE’s dynamic adder/subtractor feature saves logic resources by using one set of LEs to implement both an adder and a subtractor. This feature is controlled by the LAB-wide control signal `addnsub`. The `addnsub` signal sets the LAB to perform either $A + B$ or $A - B$. The LUT computes addition; subtraction is computed by adding the two’s complement of the intended subtractor. The LAB-wide signal converts to two’s complement by inverting the B bits within the LAB and setting carry-in to 1, which adds one to the least significant bit (LSB). The LSB of an adder/subtractor must be placed in the first LE of the LAB, where the LAB-wide `addnsub` signal automatically sets the carry-in to 1. The Quartus II Compiler automatically places and uses the adder/subtractor feature when using adder/subtractor parameterized functions.

LE Operating Modes

The MAX II LE can operate in one of the following modes:

- “Normal Mode”
- “Dynamic Arithmetic Mode”

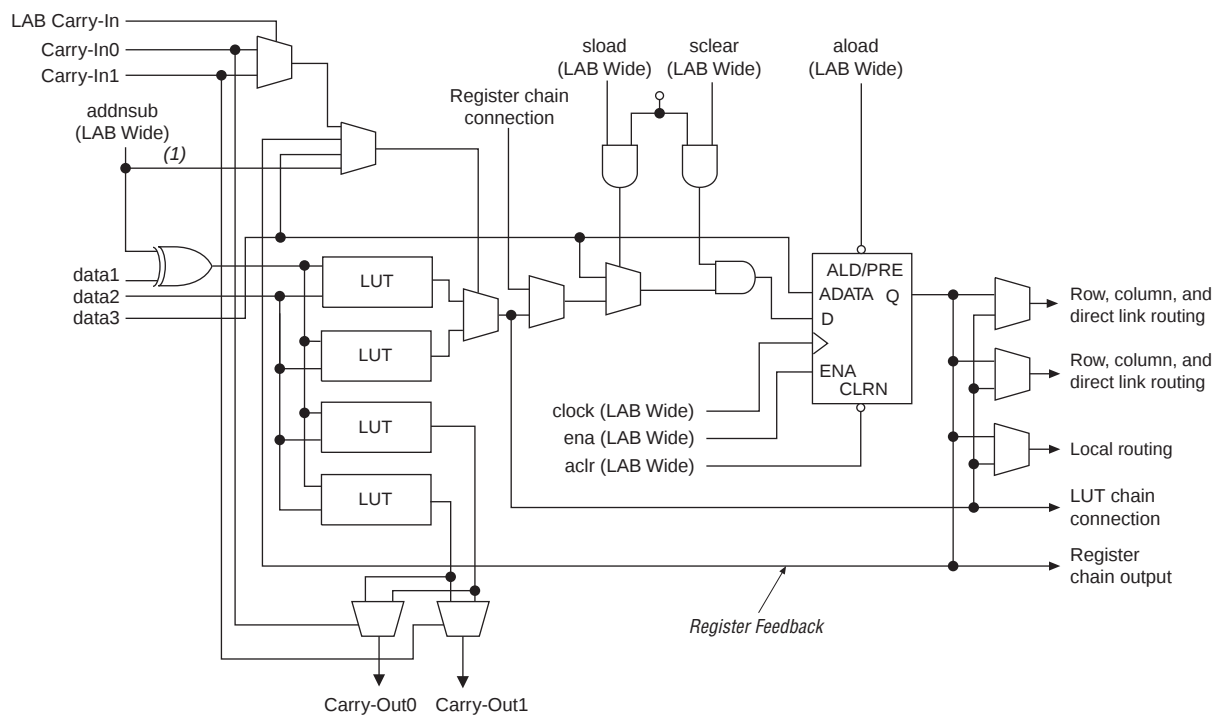
Each mode uses LE resources differently. In each mode, eight available inputs to the LE, the four data inputs from the LAB local interconnect, `carry-in0` and `carry-in1` from the previous LE, the LAB carry-in from the previous carry-chain LAB, and the register chain connection are directed to different destinations to implement the desired logic function. LAB-wide signals provide clock, asynchronous clear, asynchronous preset/load, synchronous clear, synchronous load, and clock enable control for the register. These LAB-wide signals are available in all LE modes. The `addnsub` control signal is allowed in arithmetic mode.

The Quartus II software, in conjunction with parameterized functions such as library of parameterized modules (LPM) functions, automatically chooses the appropriate mode for common functions such as counters, adders, subtractors, and arithmetic functions.

The other two LUTs use the data1 and data2 signals to generate two possible carry-out signals: one for a carry of 1 and the other for a carry of 0. The carry-in0 signal acts as the carry-select for the carry-out0 output and carry-in1 acts as the carry-select for the carry-out1 output. LEs in arithmetic mode can drive out registered and unregistered versions of the LUT output.

The dynamic arithmetic mode also offers clock enable, counter enable, synchronous up/down control, synchronous clear, synchronous load, and dynamic adder/subtractor options. The LAB local interconnect data inputs generate the counter enable and synchronous up/down control signals. The synchronous clear and synchronous load options are LAB-wide signals that affect all registers in the LAB. The Quartus II software automatically places any registers that are not used by the counter into other LABs. The addnsub LAB-wide signal controls whether the LE acts as an adder or subtractor.

Figure 2-8. LE in Dynamic Arithmetic Mode



Note to Figure 2-8:

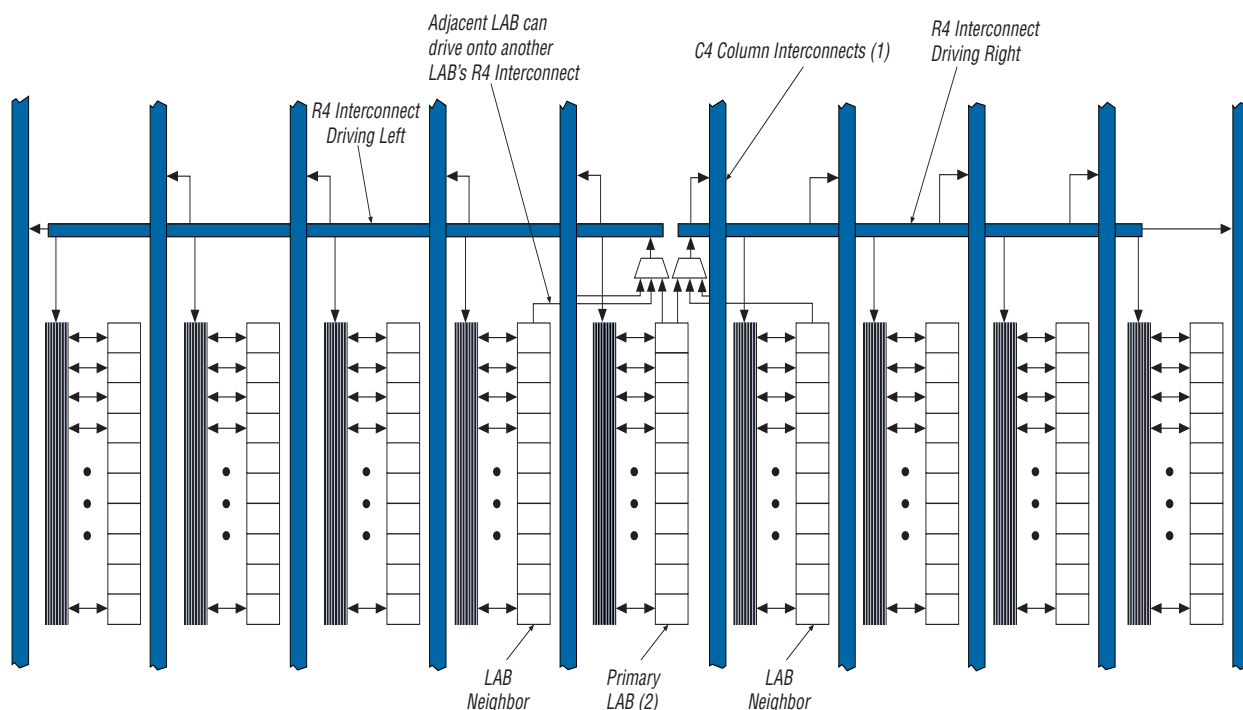
(1) The addnsub signal is tied to the carry input for the first LE of a carry chain only.

Carry-Select Chain

The carry-select chain provides a very fast carry-select function between LEs in dynamic arithmetic mode. The carry-select chain uses the redundant carry calculation to increase the speed of carry functions. The LE is configured to calculate outputs for a possible carry-in of 0 and carry-in of 1 in parallel. The carry-in0 and carry-in1 signals from a lower-order bit feed forward into the higher-order bit via the parallel carry chain and feed into both the LUT and the next portion of the carry chain. Carry-select chains can begin in any LE within an LAB.

The R4 interconnects span four LABs and are used for fast row connections in a four-LAB region. Every LAB has its own set of R4 interconnects to drive either left or right. Figure 2-10 shows R4 interconnect connections from an LAB. R4 interconnects can drive and be driven by row IOEs. For LAB interfacing, a primary LAB or horizontal LAB neighbor can drive a given R4 interconnect. For R4 interconnects that drive to the right, the primary LAB and right neighbor can drive on to the interconnect. For R4 interconnects that drive to the left, the primary LAB and its left neighbor can drive on to the interconnect. R4 interconnects can drive other R4 interconnects to extend the range of LABs they can drive. R4 interconnects can also drive C4 interconnects for connections from one row to another.

Figure 2-10. R4 Interconnect Connections



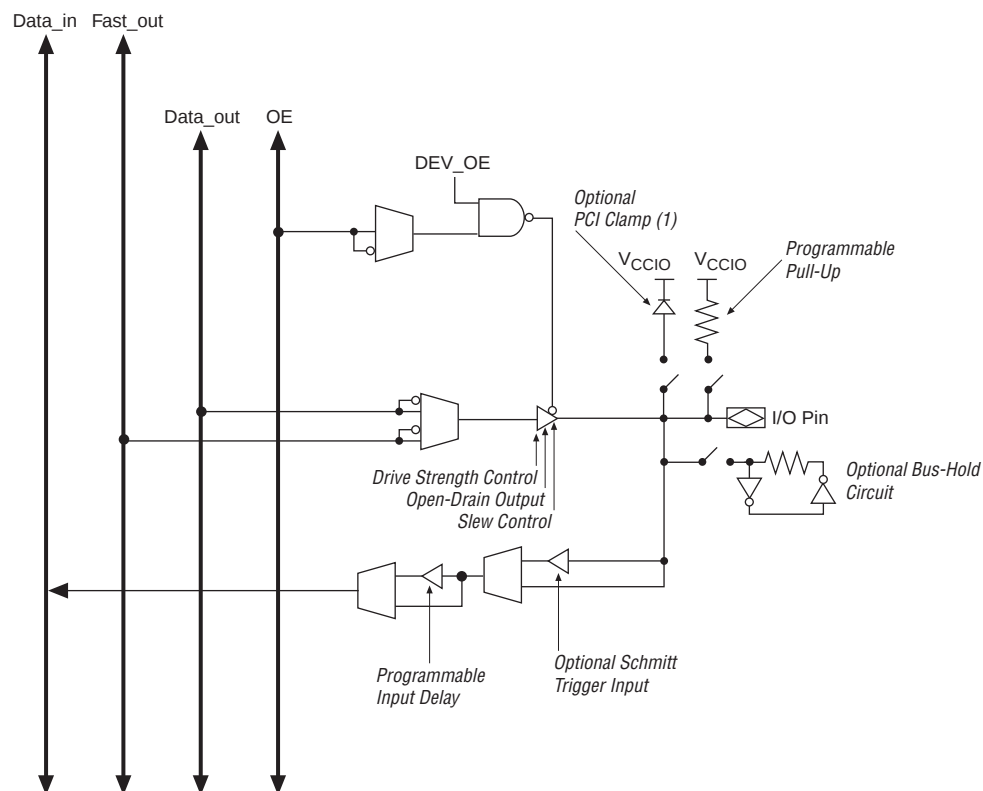
Notes to Figure 2-10:

- (1) C4 interconnects can drive R4 interconnects.
- (2) This pattern is repeated for every LAB in the LAB row.

The column interconnect operates similarly to the row interconnect. Each column of LABs is served by a dedicated column interconnect, which vertically routes signals to and from LABs and row and column IOEs. These column resources include:

- LUT chain interconnects within an LAB
- Register chain interconnects within an LAB
- C4 interconnects traversing a distance of four LABs in an up and down direction

MAX II devices include an enhanced interconnect structure within LABs for routing LE output to LE input connections faster using LUT chain connections and register chain connections. The LUT chain connection allows the combinational output of an LE to directly drive the fast input of the LE right below it, bypassing the local interconnect. These resources can be used as a high-speed connection for wide fan-in

Figure 2-19. MAX II IOE Structure**Note to Figure 2-19:**

(1) Available in EPM1270 and EPM2210 devices only.

I/O Blocks

The IOEs are located in I/O blocks around the periphery of the MAX II device. There are up to seven IOEs per row I/O block (5 maximum in the EPM240 device) and up to four IOEs per column I/O block. Each column or row I/O block interfaces with its adjacent LAB and MultiTrack interconnect to distribute signals throughout the device. The row I/O blocks drive row, column, or DirectLink interconnects. The column I/O blocks drive column interconnects.

Table 3–1. MAX II JTAG Instructions (Part 2 of 2)

JTAG Instruction	Instruction Code	Description
CLAMP (1)	00 0000 1010	Places the 1-bit bypass register between the TDI and TDO pins, which allows the boundary scan test data to pass synchronously through selected devices to adjacent devices during normal device operation, while holding I/O pins to a state defined by the data in the boundary-scan register.
USER0	00 0000 1100	This instruction allows you to define the scan chain between TDI and TDO in the MAX II logic array. This instruction is also used for custom logic and JTAG interfaces.
USER1	00 0000 1110	This instruction allows you to define the scan chain between TDI and TDO in the MAX II logic array. This instruction is also used for custom logic and JTAG interfaces.
IEEE 1532 instructions	(2)	IEEE 1532 ISC instructions used when programming a MAX II device via the JTAG port.

Notes to Table 3–1:

- (1) HIGHZ, CLAMP, and EXTEST instructions do not disable weak pull-up resistors or bus hold features.
- (2) These instructions are shown in the 1532 BSDL files, which will be posted on the Altera® website at www.altera.com when they are available.



Unsupported JTAG instructions should not be issued to the MAX II device as this may put the device into an unknown state, requiring a power cycle to recover device operation.

The MAX II device instruction register length is 10 bits and the USERCODE register length is 32 bits. Table 3–2 and Table 3–3 show the boundary-scan register length and device IDCODE information for MAX II devices.

Table 3–2. MAX II Boundary-Scan Register Length

Device	Boundary-Scan Register Length
EPM240	240
EPM570	480
EPM1270	636
EPM2210	816

Table 3–3. 32-Bit MAX II Device IDCODE (Part 1 of 2)

Device	Binary IDCODE (32 Bits) (1)				HEX IDCODE
	Version (4 Bits)	Part Number	Manufacturer Identity (11 Bits)	LSB (1 Bit) (2)	
EPM240 EPM240G	0000	0010 0000 1010 0001	000 0110 1110	1	0x020A10DD
EPM570 EPM570G	0000	0010 0000 1010 0010	000 0110 1110	1	0x020A20DD
EPM1270 EPM1270G	0000	0010 0000 1010 0011	000 0110 1110	1	0x020A30DD
EPM2210 EPM2210G	0000	0010 0000 1010 0100	000 0110 1110	1	0x020A40DD

Table 3-3. 32-Bit MAX II Device IDCODE (Part 2 of 2)

Device	Binary IDCODE (32 Bits) (1)				HEX IDCODE
	Version (4 Bits)	Part Number	Manufacturer Identity (11 Bits)	LSB (1 Bit) (2)	
EPM240Z	0000	0010 0000 1010 0101	000 0110 1110	1	0x020A50DD
EPM570Z	0000	0010 0000 1010 0110	000 0110 1110	1	0x020A60DD

Notes to Table 3-2:

- (1) The most significant bit (MSB) is on the left.
- (2) The IDCODE's least significant bit (LSB) is always 1.



For JTAG AC characteristics, refer to the *DC and Switching Characteristics* chapter in the *MAX II Device Handbook*.



For more information about JTAG BST, refer to the *IEEE 1149.1 (JTAG) Boundary-Scan Testing for MAX II Devices* chapter in the *MAX II Device Handbook*.

JTAG Block

The MAX II JTAG block feature allows you to access the JTAG TAP and state signals when either the USER0 or USER1 instruction is issued to the JTAG TAP. The USER0 and USER1 instructions bring the JTAG boundary-scan chain (TDI) through the user logic instead of the MAX II device's boundary-scan cells. Each USER instruction allows for one unique user-defined JTAG chain into the logic array.

Parallel Flash Loader

The JTAG block ability to interface JTAG to non-JTAG devices is ideal for general-purpose flash memory devices (such as Intel- or Fujitsu-based devices) that require programming during in-circuit test. The flash memory devices can be used for FPGA configuration or be part of system memory. In many cases, the MAX II device is already connected to these devices as the configuration control logic between the FPGA and the flash device. Unlike ISP-capable CPLD devices, bulk flash devices do not have JTAG TAP pins or connections. For small flash devices, it is common to use the serial JTAG scan chain of a connected device to program the non-JTAG flash device. This is slow and inefficient in most cases and impractical for large parallel flash devices. Using the MAX II device's JTAG block as a parallel flash loader, with the Quartus II software, to program and verify flash contents provides a fast and cost-effective means of in-circuit programming during test. Figure 3-1 shows MAX II being used as a parallel flash loader.

IEEE 1532 Support

The JTAG circuitry and ISP instruction set in MAX II devices is compliant to the IEEE 1532-2002 programming specification. This provides industry-standard hardware and software for in-system programming among multiple vendor programmable logic devices (PLDs) in a JTAG chain.

The MAX II 1532 BSDL files will be released on the Altera website when available.

Jam Standard Test and Programming Language (STAPL)

The Jam STAPL JEDEC standard, JESD71, can be used to program MAX II devices with in-circuit testers, PCs, or embedded processors. The Jam byte code is also supported for MAX II devices. These software programming protocols provide a compact embedded solution for programming MAX II devices.



For more information, refer to the *Using Jam STAPL for ISP via an Embedded Processor* chapter in the *MAX II Device Handbook*.

Programming Sequence

During in-system programming, 1532 instructions, addresses, and data are shifted into the MAX II device through the TDI input pin. Data is shifted out through the TDO output pin and compared against the expected data. Programming a pattern into the device requires the following six ISP steps. A stand-alone verification of a programmed pattern involves only stages 1, 2, 5, and 6. These steps are automatically executed by third-party programmers, the Quartus II software, or the Jam STAPL and Jam Byte-Code Players.

1. *Enter ISP*—The enter ISP stage ensures that the I/O pins transition smoothly from user mode to ISP mode.
2. *Check ID*—Before any program or verify process, the silicon ID is checked. The time required to read this silicon ID is relatively small compared to the overall programming time.
3. *Sector Erase*—Erasing the device in-system involves shifting in the instruction to erase the device and applying an erase pulse(s). The erase pulse is automatically generated internally by waiting in the run/test/idle state for the specified erase pulse time of 500 ms for the CFM block and 500 ms for each sector of the UFM block.
4. *Program*—Programming the device in-system involves shifting in the address, data, and program instruction and generating the program pulse to program the flash cells. The program pulse is automatically generated internally by waiting in the run/test/idle state for the specified program pulse time of 75 μ s. This process is repeated for each address in the CFM and UFM blocks.
5. *Verify*—Verifying a MAX II device in-system involves shifting in addresses, applying the verify instruction to generate the read pulse, and shifting out the data for comparison. This process is repeated for each CFM and UFM address.
6. *Exit ISP*—An exit ISP stage ensures that the I/O pins transition smoothly from ISP mode to user mode.

Document Revision History

Table 3-5 shows the revision history for this chapter.

Table 3-5. Document Revision History

Date and Revision	Changes Made	Summary of Changes
October 2008, version 1.6	■ Updated New Document Format.	—
December 2007, version 1.5	■ Added warning note after Table 3-1. ■ Updated Table 3-3 and Table 3-4. ■ Added “Referenced Documents” section.	—
December 2006, version 1.4	■ Added document revision history.	—
June 2005, version 1.3	■ Added text and Table 3-4.	—
June 2005, version 1.3	■ Updated text on pages 3-5 to 3-8.	—
June 2004, version 1.1	■ Corrected Figure 3-1. Added CFM acronym.	—

Referenced Documents

This chapter references the following documents:

- *DC and Switching Characteristics* chapter in the *MAX II Device Handbook*
- *Using MAX II Devices in Multi-Voltage Systems* chapter in the *MAX II Device Handbook*

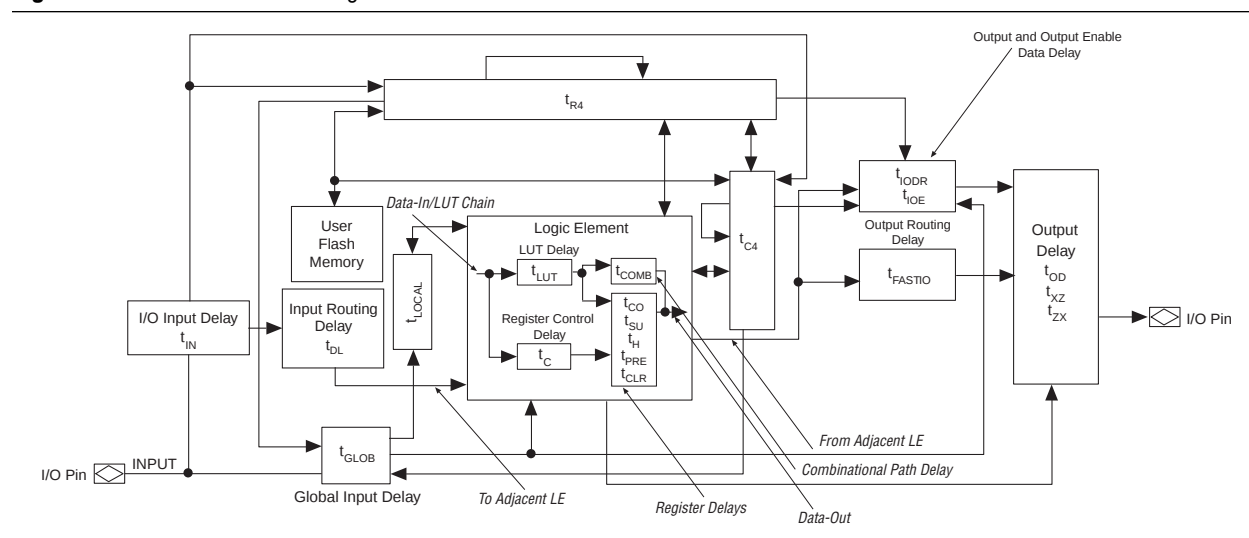
Document Revision History

Table 4–1 shows the revision history for this chapter.

Table 4–1. Document Revision History

Date and Revision	Changes Made	Summary of Changes
October 2008, version 2.1	■ Updated “MAX II Hot-Socketing Specifications” and “Power-On Reset Circuitry” sections. ■ Updated New Document Format.	—
December 2007, version 2.0	■ Updated “Hot Socketing Feature Implementation in MAX II Devices” section. ■ Updated “Power-On Reset Circuitry” section. ■ Updated Figure 4–5. ■ Added “Referenced Documents” section.	Updated document with MAX IIZ information.
December 2006, version 1.5	■ Added document revision history.	—
February 2006, version 1.4	■ Updated “MAX II Hot-Socketing Specifications” section. ■ Updated “AC and DC Specifications” section. ■ Updated “Power-On Reset Circuitry” section.	—
June 2005, version 1.3	■ Updated AC and DC specifications on page 4-2.	—
December 2004, version 1.2	■ Added content to Power-Up Characteristics section. ■ Updated Figure 4-5.	—
June 2004, version 1.1	■ Corrected Figure 4-2.	—

Figure 5-2. MAX II Device Timing Model



The timing characteristics of any signal path can be derived from the timing model and parameters of a particular device. External timing parameters, which represent pin-to-pin timing delays, can be calculated as the sum of internal parameters.

Refer to the *Understanding Timing in MAX II Devices* chapter in the *MAX II Device Handbook* for more information.

This section describes and specifies the performance, internal, external, and UFM timing specifications. All specifications are representative of the worst-case supply voltage and junction temperature conditions.

Preliminary and Final Timing

Timing models can have either preliminary or final status. The Quartus II software issues an informational message during the design compilation if the timing models are preliminary. Table 5-13 shows the status of the MAX II device timing models.

Preliminary status means the timing model is subject to change. Initially, timing numbers are created using simulation results, process data, and other known parameters. These tests are used to make the preliminary numbers as close to the actual timing parameters as possible.

Final timing numbers are based on actual device operation and testing. These numbers reflect the actual performance of the device under the worst-case voltage and junction temperature conditions.

Table 5-13. MAX II Device Timing Model Status (Part 1 of 2)

Device	Preliminary	Final
EPM240	—	✓
EPM240Z (1)	—	✓
EPM570	—	✓
EPM570Z (1)	—	✓

Table 5-21. UFM Block Internal Timing Microparameters (Part 3 of 3)

Symbol	Parameter	MAX II / MAX IIG						MAX IIZ						Unit
		-3 Speed Grade		-4 Speed Grade		-5 Speed Grade		-6 Speed Grade		-7 Speed Grade		-8 Speed Grade		
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
t _{OE}	Delay from data register clock to data register output	180	—	180	—	180	—	180	—	180	—	180	—	ns
t _{RA}	Maximum read access time	—	65	—	65	—	65	—	65	—	65	—	65	ns
t _{OSCS}	Maximum delay between the OSC_ENA rising edge to the erase/program signal rising edge	250	—	250	—	250	—	250	—	250	—	250	—	ns
t _{OSCH}	Minimum delay allowed from the erase/program signal going low to OSC_ENA signal going low	250	—	250	—	250	—	250	—	250	—	250	—	ns

Figure 5-3 through Figure 5-5 show the read, program, and erase waveforms for UFM block timing parameters shown in Table 5-21.

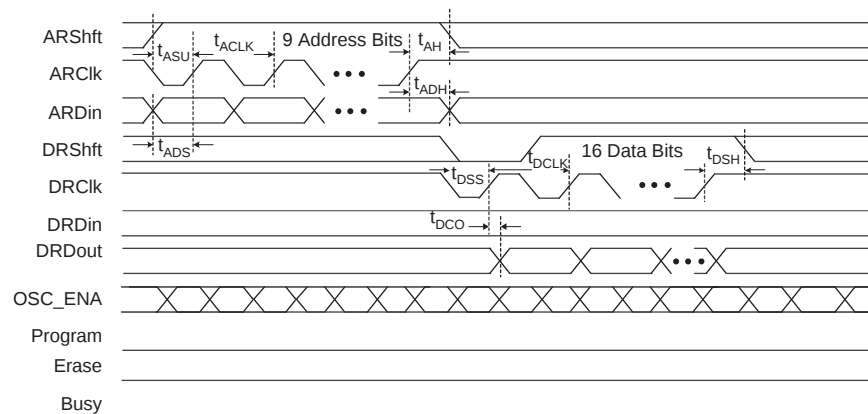
Figure 5-3. UFM Read Waveforms

Table 5-34. MAX II JTAG Timing Parameters (Part 2 of 2)

Symbol	Parameter	Min	Max	Unit
t_{JPSU}	JTAG port setup time (2)	8	—	ns
t_{JPH}	JTAG port hold time	10	—	ns
t_{JPCO}	JTAG port clock to output (2)	—	15	ns
t_{JPZX}	JTAG port high impedance to valid output (2)	—	15	ns
t_{JPXZ}	JTAG port valid output to high impedance (2)	—	15	ns
t_{JSU}	Capture register setup time	8	—	ns
t_{JSH}	Capture register hold time	10	—	ns
t_{JSCO}	Update register clock to output	—	25	ns
t_{JSZX}	Update register high impedance to valid output	—	25	ns
t_{JSXZ}	Update register valid output to high impedance	—	25	ns

Notes to Table 5-34:

- (1) Minimum clock period specified for 10 pF load on the TDO pin. Larger loads on TDO will degrade the maximum TCK frequency.
- (2) This specification is shown for 3.3-V LVTTTL/LVCMOS and 2.5-V LVTTTL/LVCMOS operation of the JTAG pins. For 1.8-V LVTTTL/LVCMOS and 1.5-V LVCMOS, the t_{JPSU} minimum is 6 ns and t_{JPCO} , t_{JPZX} , and t_{JPXZ} are maximum values at 35 ns.

Referenced Documents

This chapter references the following documents:

- *I/O Structure* section in the *MAX II Architecture* chapter in the *MAX II Device Handbook*
- *Hot Socketing and Power-On Reset in MAX II Devices* chapter in the *MAX II Device Handbook*
- *Operating Requirements for Altera Devices Data Sheet*
- *PowerPlay Power Analysis* chapter in volume 3 of the *Quartus II Handbook*
- *Understanding and Evaluating Power in MAX II Devices* chapter in the *MAX II Device Handbook*
- *Understanding Timing in MAX II Devices* chapter in the *MAX II Device Handbook*
- *Using MAX II Devices in Multi-Voltage Systems* chapter in the *MAX II Device Handbook*