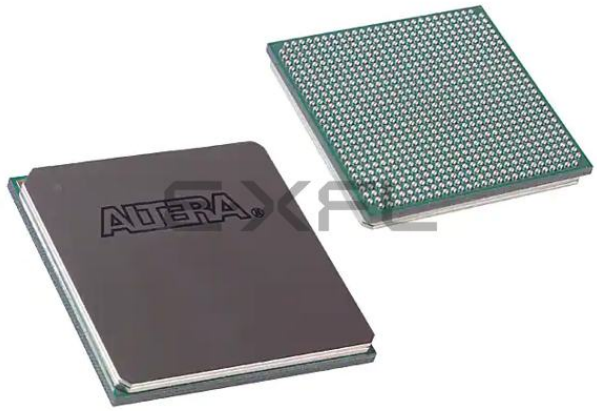


Welcome to E-XFL.COM

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	1440
Number of Logic Elements/Cells	14400
Total RAM Bits	114688
Number of I/O	486
Number of Gates	350000
Voltage - Supply	1.71V ~ 1.89V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	780-BBGA, FCBGA
Supplier Device Package	780-FBGA (29x29)
Purchase URL	https://www.e-xfl.com/product-detail/intel/ep1m350f780i6aa

…and More Features

- Advanced high-speed I/O features
  - Robust I/O standard support, including LVTTL, PCI up to 66 MHz, 3.3-V AGP in 1× and 2× modes, 3.3-V SSTL-3 and 2.5-V SSTL-2, GTL+, HSTL, CTT, LVDS, LVPECL, and 3.3-V PCML
  - High-speed differential interface (HSDI) with dedicated circuitry for CDR at up to 1.25 Gbps for LVDS, LVPECL, and 3.3-V PCML
  - Support for source-synchronous True-LVDS™ circuitry up to 840 megabits per second (Mbps) for LVDS, LVPECL, and 3.3-V PCML
  - Up to 18 input and 18 output dedicated differential channels of high-speed LVDS, LVPECL, or 3.3-V PCML
  - Built-in 100-Ω termination resistor on HSDI data and clock differential pairs
  - Flexible-LVDS™ circuitry provides 624-Mbps support on up to 100 channels with the EP1M350 device
  - Versatile three-register I/O element (IOE) supporting double data rate I/O (DDRIO), double data-rate (DDR) SDRAM, zero bus turnaround (ZBT) SRAM, and quad data rate (QDR) SRAM
- Designed for low-power operation
  - 1.8-V internal supply voltage (V_{CCINT})
  - MultiVolt™ I/O interface voltage levels (V_{CCIO}) compatible with 1.5-V, 1.8-V, 2.5-V, and 3.3-V devices
  - 5.0-V tolerant with external resistor
- Advanced interconnect structure
  - Multi-level FastTrack® Interconnect structure providing fast, predictable interconnect delays
  - Optimized high-speed Priority FastTrack Interconnect for routing critical paths in a design
  - Dedicated carry chain that implements arithmetic functions such as fast adders, counters, and comparators (automatically used by software tools and megafunctions)
  - FastLUT™ connection allowing high speed direct connection between LEs in the same logic array block (LAB)
  - Leap lines allowing a single LAB to directly drive LEs in adjacent rows
  - The RapidLAB interconnect providing a high-speed connection to a 10-LAB-wide region
  - Dedicated clock and control signal resources, including four dedicated clocks, six dedicated fast global signals, and additional row-global signals

After a Mercury device has been configured, it can be reconfigured in-circuit by resetting the device and loading new data. Real-time changes can be made during system operation, enabling innovative reconfigurable computing applications.

Software

Mercury devices are supported by the Altera Quartus™ II development system, a single, integrated package that offers HDL and schematic design entry, compilation and logic synthesis, full simulation and worst-case timing analysis, SignalTap™ logic analysis, and device configuration. The Quartus II software also ships with Altera-specific HDL synthesis tools from Exemplar Logic and Synopsys, and Altera-specific Register Transfer Level (RTL) and timing simulation tools from Model Technology. The Quartus II software supports PCs running Windows 98, Windows NT 4.0, and Windows 2000; UNIX workstations running Solaris 2.6, 7, or 8, or HP-UX 10.2 or 11.0; and PCs running Red Hat Linux 7.1.

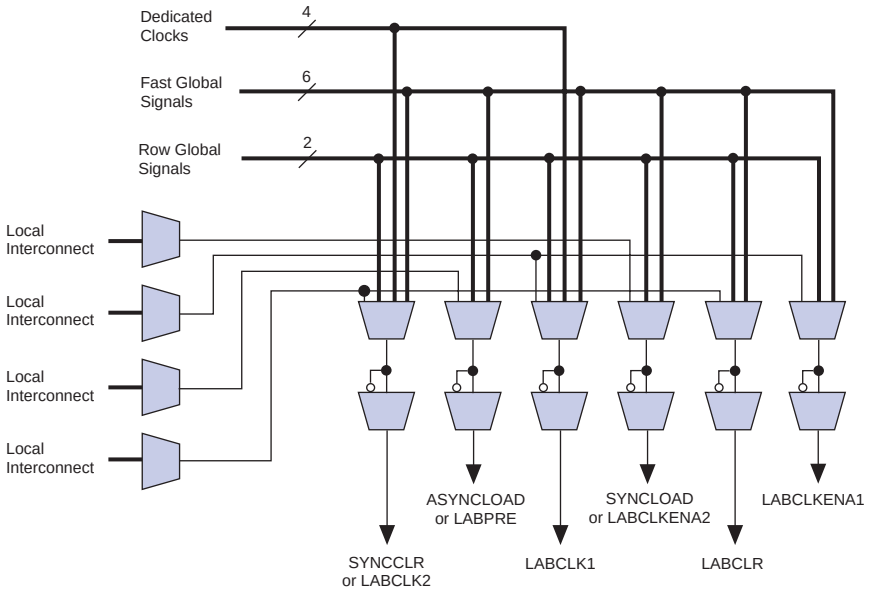
The Quartus II software provides NativeLink™ interfaces to other industry-standard PC- and UNIX-workstation-based EDA tools. For example, designers can invoke the Quartus II software from within the Mentor Graphics LeonardoSpectrum software, Synplicity's Synplify software, and the Synopsys FPGA *Express* software. The Quartus II software also contains built-in optimized synthesis libraries; synthesis tools can use these libraries to optimize designs for Mercury devices. For example, the Synopsys Design Compiler library, supplied with the Quartus II development system, includes DesignWare functions optimized for the Mercury architecture.

For more information on the Quartus II development system, see the *Quartus II Programmable Logic Development System & Software Data Sheet*.

Functional Description

The Mercury architecture contains a row-based logic array to implement general logic and a row-based embedded system array to implement memory and specialized logic functions. Signal interconnections within Mercury devices are provided by a series of row and column interconnects with varying lengths and speeds. The priority FastTrack Interconnect structure is faster than other interconnects; the Quartus II Compiler places design-critical paths on these faster lines to improve design performance.

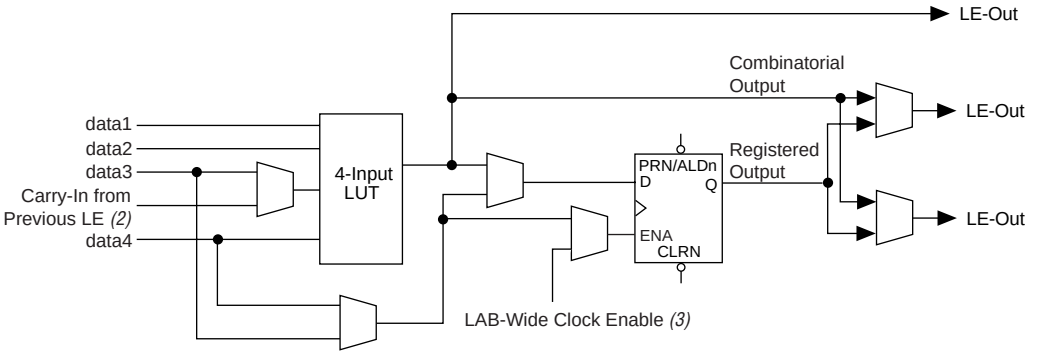
Figure 6. LAB-Wide Control Signals



Logic Element

The LE, the smallest unit of logic in the Mercury architecture, is compact and provides efficient logic usage. Each LE contains a four-input LUT, which is a function generator that can quickly implement any function of four variables. In addition, each LE contains a programmable register and carry chain with carry select look ahead capability. Each LE drives all interconnect types: local interconnect, row and priority row interconnect, column and priority column interconnect, leap lines, and RapidLAB interconnect. Each LE also has the ability to drive its combinatorial output directly to the next LE in the LAB using FastLUT connections. See Figure 7.

Figure 8. Normal-Mode LE *Note (1)*



Notes to Figure 8:

(1) LEs in normal mode support register packing.
(2) When using the carry-in in normal mode, the packed register feature is unavailable.
(3) There are two LAB-wide clock enables per LAB in addition to LE-specific clock enables.

Arithmetic Mode

The arithmetic mode is ideal for implementing adders, accumulators, and comparators. A LE in arithmetic mode contains four 2-input LUTs. The first two 2-input LUTs compute two summations based on a possible carry of `1` or `0`; the other two LUTs generate carry outputs for the two possible chains of the carry-select look-ahead (CSLA) circuitry. As shown in Figure 9, the LAB carry-in signal selects the appropriate carry-in chain (either `carry-in0` or `carry-in1`). The logic level of the chain selected in turn selects which parallel sum is generated as a combinatorial or registered output. For example, when implementing an adder, this output is the signal comprised of the sum `data1` + `data2` + *carry*, where *carry* is `0` or `1`. 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; `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. Figure 9 shows a Mercury LE in arithmetic mode.

Clear & Preset Logic Control

LAB-wide signals control logic for the register's clear and preset signals. The LE directly supports an asynchronous clear and preset function. The direct asynchronous preset does not require a NOT-gate push-back technique. Mercury devices support simultaneous preset, or asynchronous load, and clear. Asynchronous clear takes precedence if both signals are asserted simultaneously. Each LAB supports one clear and one preset signal. Two clears are possible in a single LAB by using a NOT-gate push-back technique on the preset port. The Quartus II Compiler automatically performs this second clear emulation.

In addition to the clear and preset ports, Mercury devices provide a chip-wide reset pin (DEV_CLRn) that resets all registers in the device. Use of this pin is controlled through an option in the Quartus II software that is set before compilation. The chip-wide reset overrides all other control signals.

Multi-Level FastTrack Interconnect

The Mercury architecture provides connections between LEs, ESBs, and device I/O pins via an innovative Multi-Level FastTrack Interconnect structure. The Multi-Level FastTrack Interconnect structure is a series of routing channels that traverse the device, providing a hierarchy of interconnect lines. Regular resources provide efficient and capable connections while priority resources and specialized RapidLAB, leap line, and FastLUT resources enhance performance by accelerating timing on critical paths. The Quartus II Compiler automatically places critical design paths on those faster lines to improve design performance.

This network of routing structures provides predictable performance, even for complex designs. In contrast, the segmented routing in FPGAs requires switch matrices to connect a variable number of routing paths, increasing the delays between logic resources and reducing performance.

The Multi-Level FastTrack Interconnect consists of regular and priority lines that traverse column and row interconnect channels to span sections and the entire device length. Each row of LABs, ESBs, and I/O bands is served by a dedicated row interconnect, which routes signals to and from LABs, ESBs, and I/O row bands in the same row. These row resources include:

- Row interconnect traversing the entire device from left to right
- Priority row interconnect for high speed access across the length of the device
- RapidLAB interconnect for horizontal routing that traverses a 10-LAB-wide region from a central LAB

The RapidLAB interconnect provides a specialized high-speed structure to allow a central LAB to drive other LABs within a 10-LAB-wide region. The RapidLAB lines drive alternating local LAB interconnect regions, allowing communication to all LABs in the 10-LAB-wide region. Even numbered LEs in a LAB directly drive a RapidLAB line that drives one set of alternating local interconnect regions, while odd-numbered LEs drive a RapidLAB line that drives the opposite set of alternating local interconnect regions. Figure 14 shows RapidLAB interconnect connections. This 10-LAB wide region of the RapidLAB interconnect is repeated for every LAB in the row. The region covered by the RapidLAB interconnect is smaller than 10 for source LABs that are four or five LABs in from either edge of the LAB row. The RapidLAB row interconnect is used for LAB-to-LAB routing; it is only used by I/O bands or ESBs indirectly through other interconnects. The RapidLAB interconnect drives an LE directly when that LE is in multiplier mode.

Figure 23. Encoded CAM Address Outputs

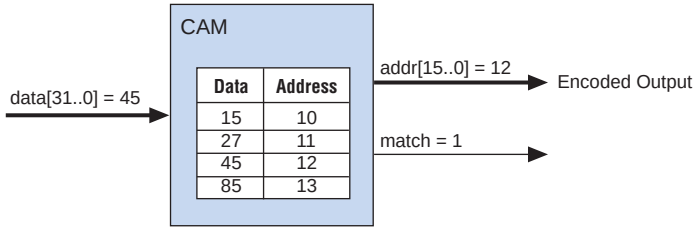
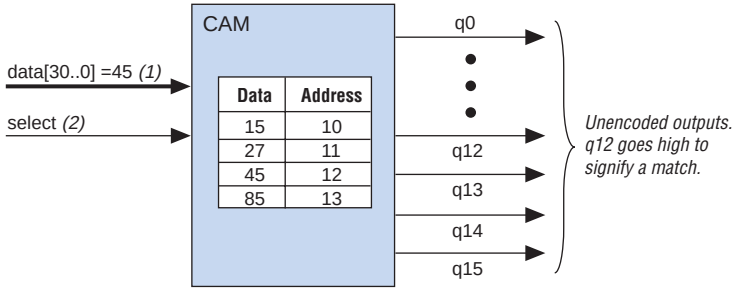


Figure 24. Unencoded CAM Address Outputs



Notes to Figure 24:

(1) For an unencoded output, the ESB only supports 31 input data bits. One input bit is used by the `select` line to choose one of the two banks of 16 outputs.

(2) If the `select` input is a 1, then CAM outputs odd words between 1 through 15. If the `select` input is a 0, CAM outputs words even words between 0 through 14.

In single-match mode, it takes two clock cycles to write into CAM, but only one clock cycle to read from CAM. In this mode, both encoded and unencoded outputs are available without external logic. Single-match mode is better suited for designs without duplicate data in the memory.

If the same data is written into multiple locations in the memory, a CAM block can be used in multiple-match or fast multiple-match modes. The ESB outputs the matched data's locations as an encoded or unencoded address. In multiple-match mode, it takes two clock cycles to write into a CAM block. For reading, there are 16 outputs from each ESB at each clock cycle. Therefore, it takes two clock cycles to represent the 32 words from a single ESB port. In this mode, encoded and unencoded outputs are available. To implement the encoded version, the Quartus II software adds a priority encoder with LEs. Fast multiple-match is identical to the multiple-match mode, however, it only takes one clock cycle to read from a CAM block and generate valid outputs. To do this, the entire ESB is used to represent 16 outputs. In fast multiple-match mode, the ESB can implement a maximum CAM block size of 16 words.

A CAM block can be pre-loaded with data during configuration, or it can be written during system operation. In most cases, two clock cycles are required to write each word into CAM. When don't-care bits are used, a third clock cycle is required.

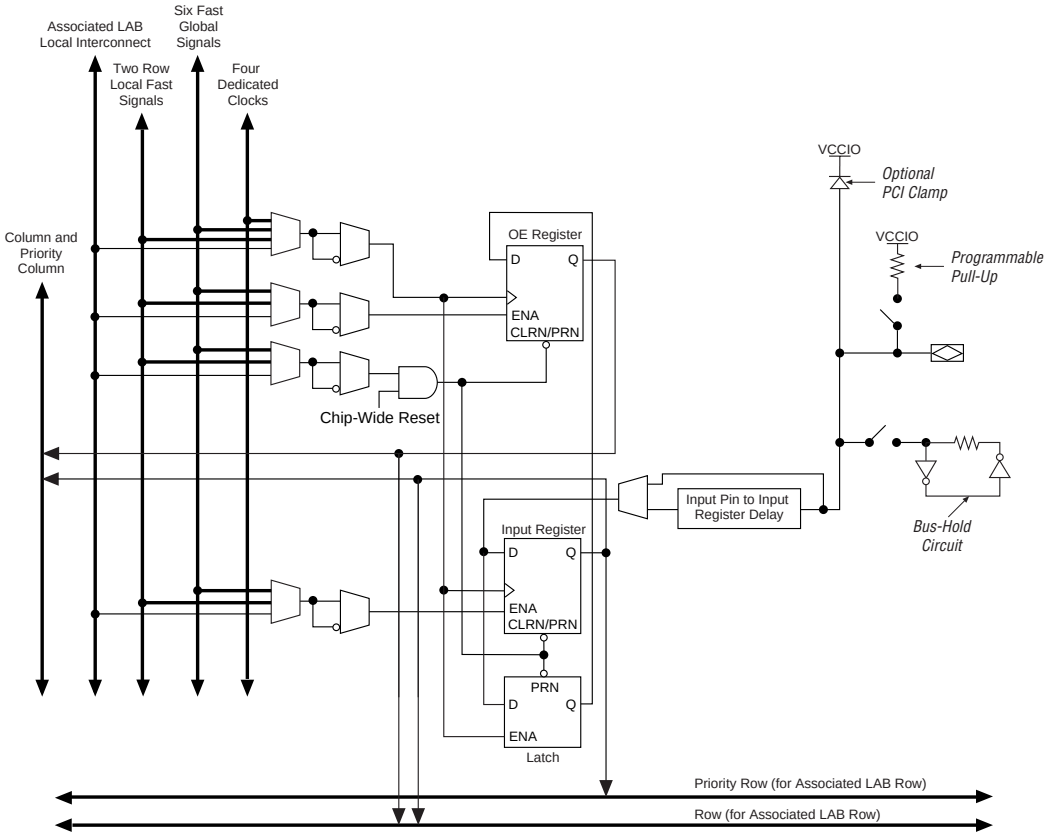
For more information on CAM, see *Application Note 119 (Implementing High-Speed Search Applications with APEX CAM)*.

Driving into ESBs

ESBs provide flexible options for driving control signals. Different clocks can be used for the ESB inputs and outputs. Registers can be inserted independently on the data input, data output, read address, write address, WREN, and RDEN signals on each port of the ESB. The fast global signals and ESB local interconnect can drive the WREN and RDEN signals. The fast global signals, dedicated clock pins, and ESB local interconnect can drive the ESB clock signals. The ESB local interconnect is driven by the ESB row interconnects which, in turn, are driven by all types of column interconnects, including high-speed leap lines. Because the LEs drive the column interconnect to the ESB local interconnect, the LEs can control the WREN and RDEN signals and the ESB clock, clock enable, and asynchronous clear signals. Figure 25 shows the ESB control signal generation logic.

In Mercury device IOEs, the OE register is a multi-purpose register available as a second input or output register. When using the IOE for double data rate inputs, the input register and OE register are automatically configured as input registers to clock input double rate data on alternating edges. An input latch is also used within the IOE for DDR input acquisition. The latch holds the data that is present during the clock high times, driving it to the OE register. This allows the OE register and input register to clock both bits of data into LEs, synchronous to the same clock edge (either rising or falling). Figure 27 shows an IOE configured for DDR input.

Figure 27. IOE Configured for DDR Input



The Mercury VCCINT pins must always be connected to a 1.8-V power supply. With a 1.8-V V_{CCINT} level, input pins are 1.8-V, 2.5-V and 3.3-V tolerant. The VCCIO pins can be connected to either a 1.5-V, 1.8-V, 2.5-V or 3.3-V power supply, depending on the output requirements. When VCCIO pins are connected to a 1.5-V power supply, the output levels are compatible with HSTL systems. When VCCIO pins are connected to a 1.8-V power supply, the output levels are compatible with 1.8-V systems. When VCCIO pins are connected to a 2.5-V power supply, the output levels are compatible with 2.5-V systems. When the VCCIO pins are connected to a 3.3-V power supply, the output high is 3.3 V and is compatible with 3.3-V or 5.0-V systems.

Table 14 summarizes Mercury MultiVolt I/O support.

Table 14. Mercury MultiVolt I/O Support *Note (1)*

V_{CCIO} (V)	Input Signal					Output Signal				
	1.5 V	1.8 V	2.5 V	3.3 V	5.0 V	1.5 V	1.8 V	2.5 V	3.3 V	5.0 V
1.5	✓	✓	✓	✓		✓				
1.8	✓ *(2)*	✓	✓	✓			✓ *(3)*			
2.5	✓ *(2)*	✓ *(2)*	✓	✓			✓ *(4)*	✓		
3.3	✓ *(2)*	✓ *(2)*	✓ *(2)*	✓	✓ *(6)*			✓ *(5)*	✓	✓

Notes to Table 14:

(1) The PCI clamping diode must be disabled to drive an input with voltages higher than V_{CCIO} unless an external resistor is used.

(2) These input levels are only available if the input standard is set to any V_{REF}-based input standard (SSTL-2, SSTL-3, HSTL, GTL+, AGP 2×). The input buffers are powered from V_{CCINT} when using V_{REF}-based input standards. LVTTL, PCI, PCI-X, AGP 1× input buffers are powered by V_{CCIO}. Therefore, these standards cannot be driven with input levels below the V_{CCIO} setting except for when V_{CCIO} = 3.3 V and the input voltage (V_I) = 2.5 V.

(3) When V_{CCIO} = 1.8 V, the Mercury device can drive a 1.5-V device with 1.8-V tolerant inputs.

(4) When V_{CCIO} = 2.5 V, the Mercury device can drive a 1.8-V device with 2.5-V tolerant inputs.

(5) When V_{CCIO} = 3.3 V, the Mercury device can drive a 2.5-V device with 3.3-V tolerant inputs.

(6) Designers can set Mercury devices to be 5.0-V tolerant by adding an external resistor and enabling the PCI clamping diode.

Power Sequencing & Hot-Socketing

Because Mercury devices can be used in a mixed-voltage environment, the devices are designed specifically to tolerate any possible power-up sequence. Therefore, the VCCIO and VCCINT power supplies may be powered in any order.

Signals can be driven into Mercury devices before and during power-up without damaging the device. In addition, Mercury devices do not drive out during power-up. Once operating conditions are reached and the device is configured, Mercury devices operate as specified by the user.

General Purpose PLL

Mercury devices have ClockLock™, ClockBoost™, and advanced ClockShift™ features, which use up to four general-purpose PLLs (separate from the two HSDI PLLs) to province clock management and clock-frequency synthesis. EP1M120 devices contain two general purpose PLLs; EP1M350 devices contain four general purpose PLLs. These PLLs allow designers to increase performance and provide clock-frequency synthesis. The PLL reduces the clock delay within a device. This reduction minimizes clock-to-output and setup times while maintaining zero hold times. The PLLs, which provide programmable multiplication, allow the designer to distribute a low-speed clock and multiply that clock on-device. Mercury devices include a high-speed clock tree: unlike ASICs, the user does not have to design and optimize the clock tree. The PLLs work in conjunction with the Mercury device's high-speed clock to provide significant improvements in system performance and bandwidth. Table 15 shows the general purpose PLL features for Mercury devices. Figure 31 shows a Mercury PLL.

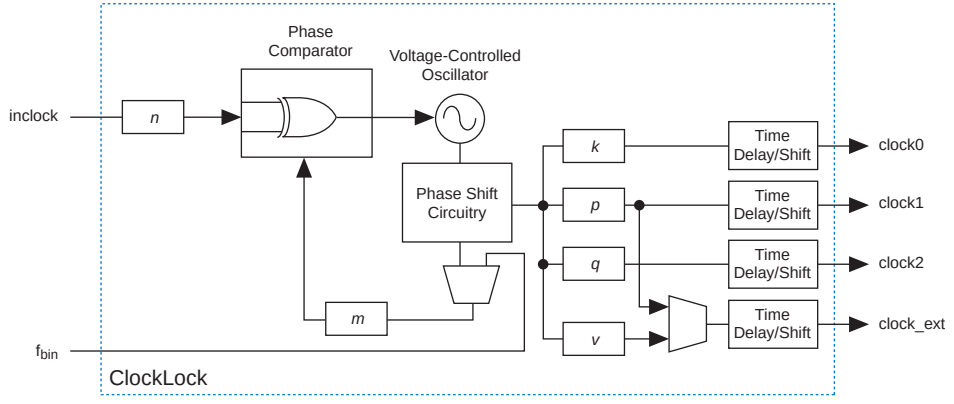
Table 15. Mercury General Purpose PLL Features

Device	Number of PLLs	ClockBoost Feature *(1)*	Number of External Clock Outputs	Number of Feedback Inputs	Advanced ClockShift
EP1M120	2	*m*/(*n* × *k*, *p*, *q*, *v*)	2	2	✓
EP1M350	4	*m*/(*n* × *k*, *p*, *q*, *v*)	4	4	✓

Note to Table 15:

(1) *n* represents the prescale divider for the PLL input. *k*, *p*, *q*, and *v* represent the different post scale dividers for the four possible PLL outputs. *m*, *k*, *p*, and *q* are integers that range from 1 to 160. *n* and *v* are integers that can range from 1 to 16.

Figure 31. Mercury General-Purpose PLL



- Normal Mode: The external clock output pin will have phase delay relative to the clock input pin. If an internal clock is used in this mode, the IOE register clock will be phase aligned to the input clock pin. Multiplication is allowed with the normal mode.

Advanced ClockShift Circuitry

General purpose PLLs in Mercury devices have advanced ClockShift™ circuitry that provides programmable phase shift and fine tune time delay shift. For phase shifting, users can enter a phase shift (in degrees or time units) that affects all PLL outputs. Phase shifts of 90, 180, and 270 can be implemented exactly. Other values of phase shifting, or delay shifting in time units, are allowed with a resolution range of 0.3 ns to 1.0 ns. This resolution varies with frequency input and the user-entered multiplication and division factors. The phase shift ability is only possible on a multiplied or divided clock if the input and output frequency have an integer multiple relationship (i.e., f_{IN} / f_{OUT} or f_{OUT} / f_{IN} must be an integer).

In addition to the phase shift feature that affects all outputs, there is an advanced fine time delay shift control on each of the four PLL outputs. Each PLL output can be shifted in 250-ps increments for a range of –2.0 ns to +2.0 ns. This ability can be used in conjunction with the phase shifting ability that affects all outputs. f_{IN} / f_{OUT} does not need to have an integer relationship for the advanced fine time delay shift control.

Clock Enable Signal

Mercury PLLs have a CLKLK_ENA pin for enabling / disabling all of the device PLLs. When the CLKLK_ENA pin is high, the PLL drives a clock to all its output ports. When the CLKLK_ENA pin is low, the clock0, clock1, clock2 and extclock ports are driven by GND and all of the PLLs go out of lock. When the CLKLK_ENA pin goes high again, the PLL must relock.

The individual enable port for each general purpose PLL is programmable. If more than one general-purpose PLL is instantiated, each one does not have to use the clock enable. To enable / disable the device PLLs with the CLKLK_ENA pin, the inclocken port on the altclklock instance must be connected to the CLKLK_ENA input pin.

Figure 32 shows the timing requirements for the JTAG signals.

Figure 32. Mercury JTAG Waveforms

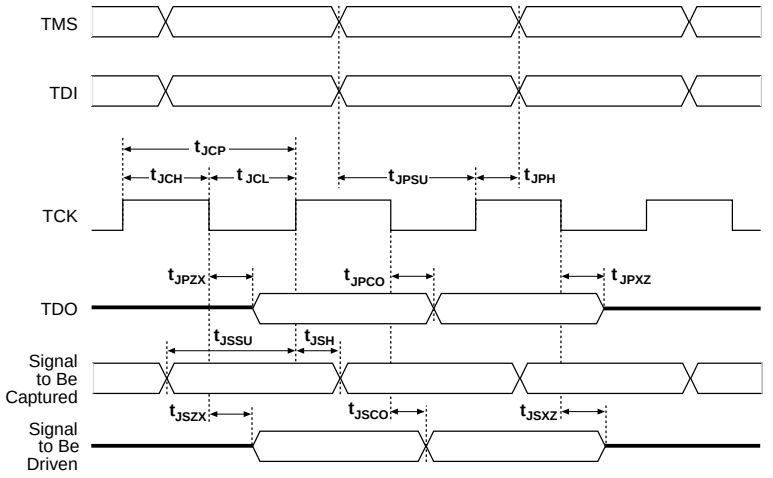


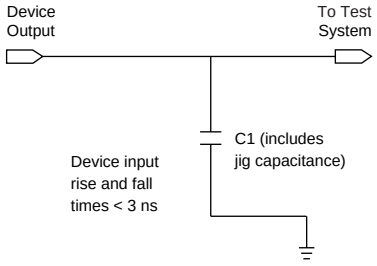
Table 19 shows the JTAG timing parameters and values for Mercury devices.

Table 19. Mercury JTAG Timing Parameters & Values

Symbol	Parameter	Min	Max	Unit
t_{JCP}	TCK clock period	100		ns
t_{JCH}	TCK clock high time	50		ns
t_{JCL}	TCK clock low time	50		ns
t_{JPSU}	JTAG port setup time	20		ns
t_{JPH}	JTAG port hold time	45		ns
t_{JPCO}	JTAG port clock to output		25	ns
t_{JPZX}	JTAG port high impedance to valid output		25	ns
t_{JPXZ}	JTAG port valid output to high impedance		25	ns
t_{JSSU}	Capture register setup time	20		ns
t_{JSH}	Capture register hold time	45		ns
t_{JSCO}	Update register clock to output		35	ns
t_{JSZX}	Update register high impedance to valid output		35	ns
t_{JSXZ}	Update register valid output to high impedance		35	ns

Figure 33. Mercury AC Test Conditions

Power supply transients can affect AC measurements. Simultaneous transitions of multiple outputs should be avoided for accurate measurement. Threshold tests must not be performed under AC conditions. Large-amplitude, fast-ground-current transients normally occur as the device outputs discharge the load capacitances. When these transients flow through the parasitic inductance between the device ground pin and the test system ground, significant reductions in observable noise immunity can result.



Operating Conditions

Table 20 through 43 provide information on absolute maximum ratings, recommended operating conditions, DC operating conditions, and capacitance for 1.8-V Mercury devices.

Table 20. Mercury Device Absolute Maximum Ratings *Note (1)*

Symbol	Parameter	Conditions	Minimum	Maximum	Unit
V_{CCINT}	Supply voltage	With respect to ground *(2)*	–0.5	2.5	V
V_{CCIO}			–0.5	4.6	V
V_I	DC input voltage		–0.5	4.6	V
I_{OUT}	DC output current, per pin		–34	34	mA
T_{STG}	Storage temperature	No bias	–65	150	° C
T_{AMB}	Ambient temperature	Under bias	–65	135	° C
T_J	Junction temperature	BGA packages under bias		135	° C

Table 23. LVTTL Specifications *Note (10)*

Symbol	Parameter	Conditions	Minimum	Maximum	Units
V_{CCIO}	Output supply voltage		3.0	3.6	V
V_{IH}	High-level input voltage		1.7	4.1	V
V_{IL}	Low-level input voltage		–0.5	0.7	V
I_I	Input pin leakage current	V_{IN} = 0 V or V_{CCIO}	–10	10	µA
V_{OH}	High-level output voltage	I_{OH} = –4 mA	2.4		V
V_{OL}	Low-level output voltage	I_{OL} = 4 mA		0.45	V

Table 24. LVCMOS Specifications

Symbol	Parameter	Conditions	Minimum	Maximum	Units
V_{CCIO}	Power supply voltage range		3.0	3.6	V
V_{IH}	High-level input voltage		1.7	4.1	V
V_{IL}	Low-level input voltage		–0.5	0.7	V
I_I	Input pin leakage current	V_{IN} = 0 V or V_{CCIO}	–10	10	µA
V_{OH}	High-level output voltage	V_{CCIO} = 3.0, I_{OH} = –0.1 mA	V_{CCIO} – 0.2		V
V_{OL}	Low-level output voltage	V_{CCIO} = 3.0, I_{OL} = 0.1 mA		0.2	V

Table 25. 2.5-V I/O Specifications *Note (10)*

Symbol	Parameter	Conditions	Minimum	Maximum	Units
V_{CCIO}	Output supply voltage		2.375	2.625	V
V_{IH}	High-level input voltage		1.7	4.1	V
V_{IL}	Low-level input voltage		–0.5	0.7	V
I_I	Input pin leakage current	V_{IN} = 0 V or V_{CCIO}	10	10	µA
V_{OH}	High-level output voltage	I_{OH} = –0.1 mA	2.1		V
		I_{OH} = –1 mA	2.0		V
		I_{OH} = –2 mA	1.7		V
V_{OL}	Low-level output voltage	I_{OL} = 0.1 mA		0.2	V
		I_{OH} = 1 mA		0.4	V
		I_{OH} = 2 mA		0.7	V

Table 42. Bus Hold Parameters

Parameter	Conditions	VCCIO Level						Units
		1.8 V		2.5 V		3.3 V		
		Minimum	Maximum	Minimum	Maximum	Minimum	Maximum	
Low sustaining current	$V_{IN} > V_{IL}$ (maximum)	30		50		70		µA
High sustaining current	$V_{IN} < V_{IH}$ (minimum)	–30		–50		–70		µA
Low overdrive current	$0 V < V_{IN} < V_{CCIO}$		200		300		500	µA
High overdrive current	$0 V < V_{IN} < V_{CCIO}$		–200		–300		–500	µA

Table 43. Mercury Device Capacitance *Note (13)*

Symbol	Parameter	Minimum	Typical	Maximum	Unit
C_{IO}	I/O pin capacitance		13.5		pF
C_{CLK}	Input capacitance on CLK[4..1] pins		16.9		pF
C_{RXHSDI}	Input capacitance on HSDI receiver pins		8.0		pF
C_{TXHSDI}	Input capacitance on HSDI transmitter pins		18.0		pF
$C_{CLKHSDI}$	Input capacitance on HSDI clock pins		7.5		pF
$C_{FLEXLVDSRX}$	Input capacitance on flexible LVDS receiver pins		13.4		pF
$C_{FLEXLVDSTX}$	Input capacitance on flexible LVDS transmitter pins		13.4		pF

Table 51. EP1M350 External Bidirectional Timing Parameters *Note (1)*

Symbol	-5 Speed Grade		-6 Speed Grade		-7 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
$t_{INSUBIDIR}$	0.60		0.57		0.71		ns
$t_{INHBIDIR}$	0.00		0.00		0.00		ns
$t_{OUTCOBIDIR}$	2.00	3.95	2.00	3.97	2.00	4.75	ns
$t_{XZBIDIR}$		3.90		3.93		4.70	ns
$t_{ZXBIDIR}$ *(2)*		3.90		3.93		4.70	ns
$t_{ZXBIDIR}$ *(3)*		4.10		4.13		4.94	ns
$t_{INSUBIDIRPLL}$	0.69		0.70		0.82		ns
$t_{INHBIDIRPLL}$	0.00		0.00		0.00		ns
$t_{OUTCOBIDIRPLL}$	0.50	2.23	0.50	2.23	0.50	2.69	ns
$t_{XZBIDIRPLL}$		2.19		2.18		2.63	ns
$t_{ZXBIDIRPLL}$ *(2)*		2.19		2.18		2.63	ns
$t_{ZXBIDIRPLL}$ *(3)*		2.39		2.38		2.87	ns

Notes to Tables 46 – 51:

(1) Timing will vary by I/O pin placement. Therefore, use the Quartus II software to determine exact I/O timing for each pin.

(2) This parameter is measured with the **Increase t_{ZX} Delay to Output Pin** option set to **Off**.

(3) This parameter is measured with the **Increase t_{ZX} Delay to Output Pin** option set to **On**.

Power Consumption

Detailed power consumption information for Mercury devices will be released when available.

Configuration & Operation

The Mercury architecture supports several configuration schemes. This section summarizes the device operating modes and available device configuration schemes.

Operating Modes

The Mercury architecture uses SRAM configuration elements that require configuration data to be loaded each time the circuit powers up. The process of physically loading the SRAM data into the device is called configuration. During initialization, which occurs immediately after configuration, the device resets registers, enables I/O pins, and begins to operate as a logic device. The I/O pins are tri-stated during power-up and before and during configuration. Together, the configuration and initialization processes are called command mode; normal device operation is called user mode.

Before and during device configuration, all I/O pins are pulled to V_{CCIO} by a built-in weak pull-up resistor.

SRAM configuration elements allow Mercury devices to be reconfigured in-circuit by loading new configuration data into the device. Real-time reconfiguration is performed by forcing the device into command mode with a device pin, loading different configuration data, reinitializing the device, and resuming user-mode operation. In-field upgrades can be performed by distributing new configuration files.

Configuration Schemes

The configuration data for a Mercury device can be loaded with one of five configuration schemes (see Table 52), chosen on the basis of the target application. A configuration device, intelligent controller, or the JTAG port can be used to control the configuration of a Mercury device. When a configuration device is used, the system can configure automatically at system power-up.

By connecting the configuration enable (nCE) and configuration enable output (nCEO) pins on each device, multiple Mercury devices can be configured in any of five configuration schemes.

Table 52. Data Sources for Configuration

Configuration Scheme	Data Source
Configuration device	Configuration device
Passive serial (PS)	MasterBlaster™ or ByteBlasterMV™ download cable or serial data source
Passive parallel asynchronous (PPA)	Parallel data source
Passive parallel synchronous (PPS)	Parallel data source
JTAG	MasterBlaster or ByteBlasterMV download cable or a microprocessor with a Jam STAPL or JBC file

For more information on configuration, see *Application Note 116 (Configuring APEX 20K, FLEX 10K & FLEX 6000 Devices)*.

Device Pin-Outs

See the Altera web site (**http://www.altera.com**) or the ***Altera Digital Library*** for pin-out information.

Revision History

The information contained in the *Mercury Programmable Logic Device Family Data Sheet* version 2.2 supersedes information published in previous versions.

Version 2.2

The following changes were made to the *Mercury Programmable Logic Device Family Data Sheet* version 2.2:

- Updated the condition values (symbols I_I and I_{OZ}) in Table 22.

Version 2.1

The following changes were made to the *Mercury Programmable Logic Device Family Data Sheet* version 2.1:

- Updated Table 8.
- Updated EP1M350 regular I/O banks in Table 13.
- Updated *Note (6)* in Table 14.

Version 2.0

The following changes were made to the *Mercury Programmable Logic Device Family Data Sheet* version 2.0:

- Changed all references to PCML to 3.3-V PCML.
- Updated Table 4.
- Updated "High-Speed Differential Interface" on page 8.
- Added Tables 6 through 8.
- Added Figures 34 and 35.
- Updated I/O specifications in Tables 28 and 29.
- Updated Mercury device capacitance in Table 43.
- Updated EP1M120 device timing in Tables 46 through 49.
- Added EP1M350 device timing in Tables 50 and 51.

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com
Applications Hotline:
(800) 800-EPLD
Customer Marketing:
(408) 544-7104
Literature Services:
lit_req@altera.com