Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details	
Product Status	Active
Number of LABs/CLBs	480
Number of Logic Elements/Cells	-
Total RAM Bits	-
Number of I/O	303
Number of Gates	-
Voltage - Supply	1.71V ~ 1.89V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	484-BBGA, FCBGA
Supplier Device Package	484-FBGA (23x23)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=ep1m120f484c5n

…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.

Notes to Figure 4:

(1) EP1M350 devices have 18 individual receiver and transmitter channels. EP1M120 devices have 8 individual receiver and transmitter channels. Receiver and transmitter channel numbers in parenthesis are for EP1M350 devices.
(2) *W* = 1 to 12, 14, 16, 18, or 20
J = 3 to 12, 14, 16, 18, or 20
W does not have to equal *J*.
(3) For every receiver channel in EP1M350 and EP1M120 devices, the ÷*J* recovered clock can drive the priority column interconnect for use as a clock.
(4) The two center channels adjacent to the HSDI PLLs (channels 4 and 5 for EP1M120 devices, channels 9 and 10 for EP1M350 devices) can drive the Mercury device's global clocks.
(5) `HSDI_CLK1` and `HSDI_CLK2` pins must be differential. These clock pins drive HSDI PLLs only. They do not drive to the logic array.

The multiplied reference clock is also used to synchronize and serialize at the transmitter side.

Up to two different serial data rates are supported for input channels or output channels. Received data must be non-return-to-zero (NRZ).

Table 7 defines the support for CDR-mode applications. Table 8 shows the supported data rates for each speed grade.

Table 7. CDR-Mode Applications

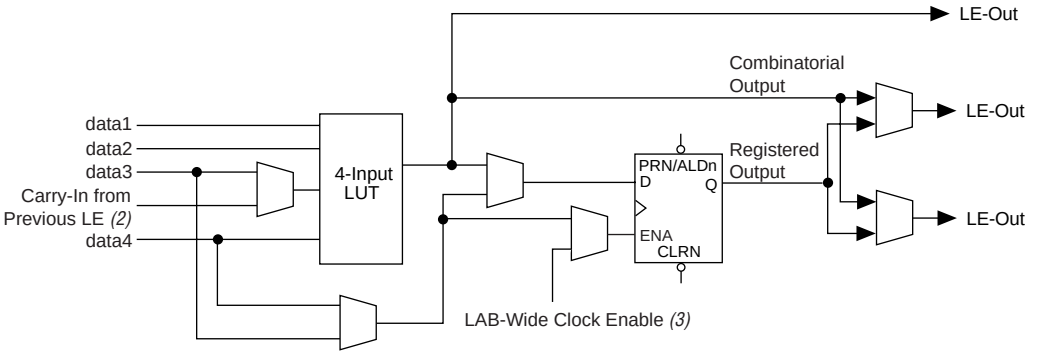
Data Rate	CDR Mode					
	DC-Coupled LVDS	DC-Coupled LVPECL	DC-Coupled 3.3-V PCML	AC-Coupled LVDS *(1)*	AC-Coupled LVPECL *(1)*	AC-Coupled 3.3-V PCML *(1)*
1.0 to 1.25 Gbps	*(2)*	✓	✓	✓	✓	✓
≤ 1.0 Gbps	✓	✓	✓	✓	✓	✓

Notes to Table 7:

(1) The V_{CM} operating range for AC-coupled applications is from 0 to 0.7 V and from 1.8 to 2.4 V.
(2) Use AC-coupled LVDS or another I/O standard. The DC-coupled LVDS I/O standard provides performance up to 1.0 Gbps.

For more information on CDR, see *AN 130: CDR in Mercury Devices*.

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



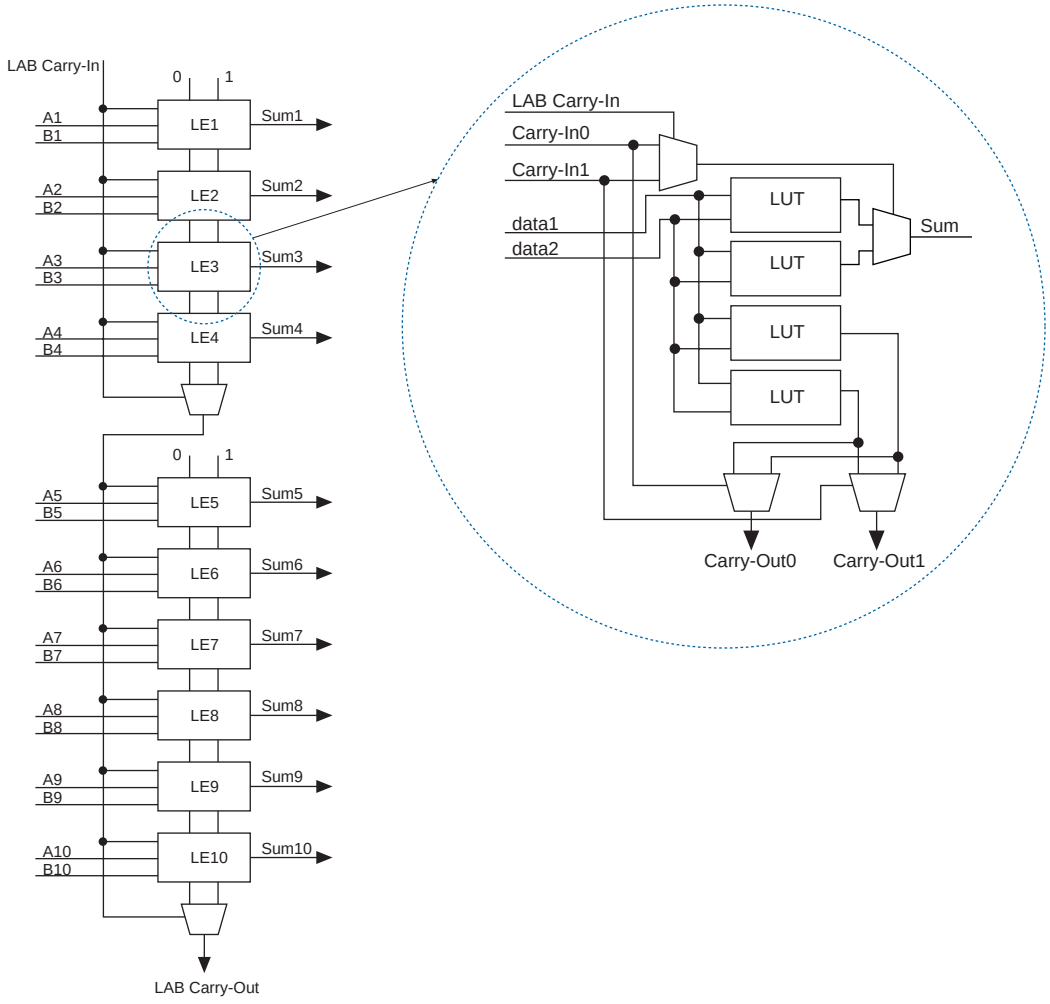
Notes to Figure 8:

(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.

Figure 10. CSLA Details



The Quartus II Compiler can create CSLA logic automatically during design processing. Alternatively, the designer can create CSLA logic manually during design entry. Parameterized functions such as library of parameterized modules (LPM) and DesignWare functions automatically take advantage of carry chains for the appropriate functions.

For a typical 16 × 16-bit binary tree multiplier, five stages are needed to determine the final product. The Mercury LE multiplier mode allows the partial product formation stage (Stage 1) and the first sum of stages (Stage 2) to be combined in a single stage, shown in Figure 13. This feature, combined with the direct connection between RapidLAB lines and LEs in multiplier mode, allows the fast dedicated implementation of multipliers.

Figure 13. Mercury Binary Tree Implementation

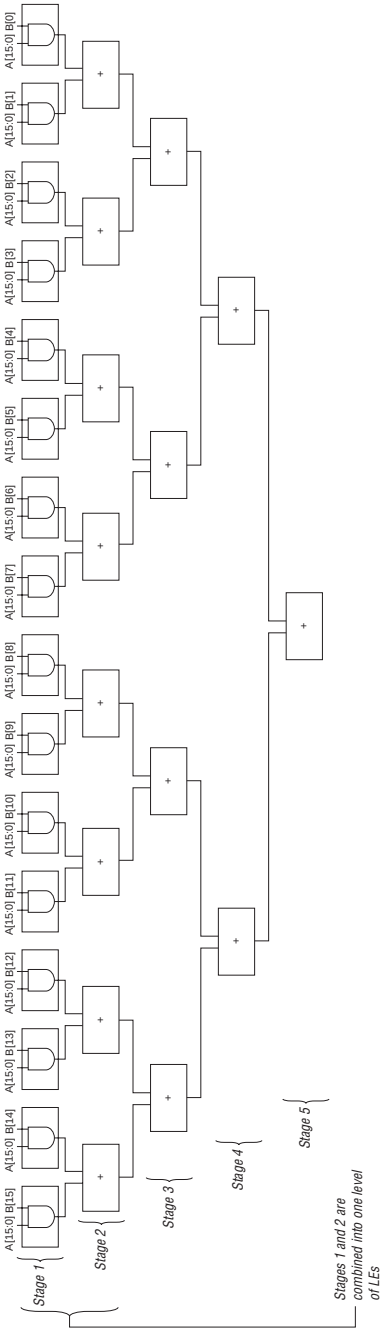


Table 9 summarizes how various elements of the Mercury architecture drive each other.

Table 9. Mercury Routing Scheme

Source	Destination										
	LE	Local Interconnect	IOE	ESB Row Interconnect	ESB	Row	Priority Row	RapidLAB Interconnect	Column	Priority Column	Leap Lines
LE	✓ (1)	✓				✓	✓	✓	✓	✓	✓
Local Interconnect	✓		✓								
IOE		✓ (2)				✓ (3)	✓ (3)		✓	✓	
ESB Row Interconnect					✓						
ESB				✓					✓	✓	✓
Row		✓									
Priority Row		✓									
RapidLAB Interconnect	✓ (4)	✓									
Column				✓		✓	✓		✓		
Priority Column				✓			✓	✓	✓	✓	
Leap Lines				✓		✓	✓	✓	✓		

Notes to Table 9:

(1) This direct connection is possible through the FastLUT connection.
(2) IOEs can connect to the adjacent LAB's local interconnects in the associated LAB row.
(3) IOEs can connect to row and priority row interconnects in the associated LAB row.
(4) This connection is used for multiplier mode.

Embedded System Block

The ESB can implement various types of memory blocks, including quad-port, true dual-port, dual- and single-port RAM, ROM, FIFO, and CAM blocks.

The ESB includes input and output registers; the input registers synchronize reads and/or writes, and the output registers can pipeline designs to further increase system performance. The ESB offers a quad port mode, which supports up to four port operations, two reads and two writes simultaneously, with the ability for a different clock on each of the four ports. Figure 17 shows the ESB quad-port block diagram.

ESBs can implement synchronous RAM, which is easier to use than asynchronous RAM. A circuit using asynchronous RAM must generate the RAM write enable (WE) signal while ensuring that its data and address signals meet setup and hold time specifications relative to the WE signal. In contrast, the ESB's synchronous RAM generates its own WE signal and is self-timed with respect to the global clock. Circuits using the ESB's self-timed RAM must only meet the setup and hold time specifications relative to the global clock.

ESBs are grouped together in rows at the top and bottom of the device for fast horizontal communication. The ESB row interconnect can be driven by any ESB in the row. The row interconnect drives the ESB local interconnect, which in turn drives the ESB ports. ESB outputs drive the ESB local interconnect, which can drive row interconnect as well as all types of column interconnect, including leap lines. The leap lines allow fast access between ESBs and the adjacent LAB row.

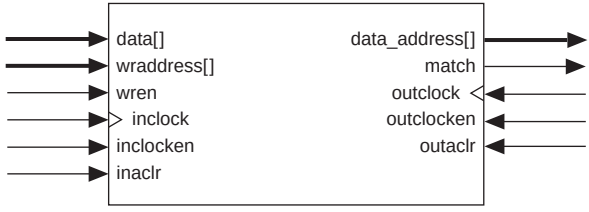
When implementing memory, each ESB can be configured in any of the following sizes for quad port and true dual-port memory modes: 256 × 16; 512 × 8; 1,024 × 4; 2,048 × 2; or 4,096 × 1. For dual-port and single-port modes, the ESB can be configured for 128 × 32 in addition to the list above. For variable port width RAMs, any port width ratio combination must be 1, 2, 4, 8, or 16. For example, a RAM with data ports of width 1 and 16 or 2 and 32 will work, but not 1 and 32.

The ESB can also be split in half and used for two independent 2,048-bit single-port or dual-port RAM blocks. For example, one half of the ESB can be used as a 128 × 16 memory single-port memory while the other half can be used for a 1,024 × 2 dual-port memory. This effectively doubles the number of RAMs a Mercury device can implement for its given number of ESBs. The Quartus II software automatically merges two logical memory functions in a design into an ESB; the designer does not need to merge the functions manually.

By combining multiple ESBs, the Quartus II software implements larger memory blocks automatically. For example, two 256 × 16 RAM blocks can be combined to form a 256 × 32 RAM block, and two 512 × 8 RAM blocks can be combined to form a 512 × 16 RAM block. Memory performance does not degrade for memory blocks up to 4,096 words deep. Each ESB can implement a 4,096-word-deep memory; the ESBs are used in parallel, eliminating the need for any external control logic and its associated delays. To create a high-speed memory block more than 4,096 words deep, the Quartus II software will automatically combine ESBs with LE control logic.

CAM is used for high-speed search operations. When searching for data within a RAM block, the search is performed serially. Thus, finding a particular data word can take many cycles. CAM searches all addresses in parallel and outputs the address storing a particular word. When a match is found, a match flag is set high. CAM is ideally suited for applications such as Ethernet address lookup, data compression, pattern recognition, cache tags, fast routing table lookup, and high-bandwidth address filtering. Figure 22 shows the CAM block diagram.

Figure 22. CAM Block Diagram



The Mercury on-chip CAM provides faster system performance than traditional discrete CAM. Integrating CAM and logic into the Mercury device eliminates off-chip and on-chip delays, improving system performance.

When in CAM mode, the ESB implements a 32-word, 32-bit CAM. Wider or deeper CAM, such as a 32-word, 64-bit or 128-word, 32-bit block, can be implemented by combining multiple CAM blocks with some ancillary logic implemented in LEs. The Quartus II software automatically combines ESBs and LEs to create larger CAM blocks.

CAM supports writing “don’t care” bits into words of the memory. The don’t-care bit can be used as a mask for CAM comparisons; any bit set to don’t-care has no effect on matches.

CAM can generate outputs in three different modes: single-match mode, multiple-match mode, and fast multiple-match mode. In each mode, the ESB outputs the matched data’s location as an encoded or unencoded address. When encoded, the ESB outputs an encoded address of the data’s location. For instance, if the data is located in address 12, the ESB output is 12. When unencoded, each ESB port uses its 16 outputs to show the location of the data over two clock cycles. In this case, if the data is located in address 12, the 12th output line goes high. Figures 22 and 23 show the encoded CAM outputs and unencoded CAM outputs, respectively.

Zero Bus Turnaround SRAM Interface Support

In addition to DDR SDRAM support, Mercury device I/O pins also support interfacing with ZBT SRAM blocks at up to 200 MHz. ZBT SRAM blocks are designed to eliminate dead bus cycles when turning a bidirectional bus around between reads and writes, or writes and reads. ZBT allows for 100% bus utilization because ZBT SRAM can read or write on every clock cycle.

To avoid bus contention, the output t_{ZX} delay ensures that the clock-to-low-impedance time ($\mathbf{t_{ZX}}$) is greater than the clock-to-high-impedance time ($\mathbf{t_{XZ}}$). Time delay control of clocks to the OE/output and input register, using a single general purpose PLL, enable the Mercury device to meet ZBT $\mathbf{t_{CO}}$ and $\mathbf{t_{SU}}$ times.

Programmable Drive Strength

The output buffer for each Mercury device I/O pin has a programmable drive strength control for certain I/O standards. The LVTTL standard has several levels of drive strength that can be controlled by the user. SSTL-3 class I and II, SSTL-2 class I and II, HSTL class I and II, and 3.3-V GTL+ support a minimum or maximum setting. The minimum setting is the lowest drive strength that guarantees the I_{OH}/I_{OL} of the standard. The maximum setting provides higher drive strength that allows for faster switching and is the default setting. Using settings below the maximum provides signal slew-rate control to reduce system noise and signal overshoot. Table 11 shows the possible settings for the I/O standards with drive strength control.

Bus Hold

Each Mercury device I/O pin provides an optional bus-hold feature. When this feature is enabled for an I/O pin, the bus-hold circuitry weakly holds the signal at its last driven state. By holding the last driven state of the pin until the next input signal is present, the bus-hold feature eliminates the need to add external pull-up or pull-down resistors to hold a signal level when the bus is tri-stated. The bus-hold circuitry also pulls undriven pins away from the input threshold voltage where noise can cause unintended high-frequency switching. This feature can be selected individually for each I/O pin. The bus-hold output will drive no higher than V_{CCIO} to prevent overdriving signals. If the bus-hold feature is enabled, the programmable pull-up option cannot be used. The bus-hold feature should also be disabled if open-drain outputs are used with the GTL+ I/O standard.

The bus-hold circuitry weakly pulls the signal level to the last driven state through a resistor with a nominal resistance (R_{BH}) of approximately 8 kΩ. Table 42 gives specific sustaining current that will be driven through this resistor and overdrive current that will identify the next driven input level. This information is provided for each V_{CCIO} voltage level.

The bus-hold circuitry is active only after configuration. When going into user mode, the bus-hold circuit captures the value on the pin present at the end of configuration.

Programmable Pull-Up Resistor

Each Mercury device I/O pin provides an optional programmable pull-up resistor during user mode. When this feature is enabled for an I/O pin, the pull-up resistor (50 kΩ) weakly holds the output to the V_{CCIO} level of the bank that the output pin resides in.

I/O Row Bands

The I/O row bands are one of the advanced features of the Mercury architecture. All IOEs are grouped in I/O row bands across the device. The number of I/O row bands depends on the Mercury device size. The I/O row bands are designed for flip-chip technology, allowing I/O pins to be distributed across the entire chip, not only in the periphery. This array driver technology allows higher I/O pin density (I/O pins per device area) than peripheral I/O pins.

Dedicated Fast Lines & I/O Pins

Mercury devices incorporate dedicated bidirectional pins for signals with high internal fanout, such as PCI control signals. These pins are called dedicated fast I/O pins (FAST1, FAST2, FAST3, FAST4, FAST5, and FAST6) and can drive the six global fast lines throughout the device, ideal for fast clock, clock enable, clear, preset, or high fanout logic signal distribution. The dedicated fast I/O pins have the same IOE as a regular I/O pin. The dedicated fast lines can also be driven by a LE local interconnect to generate internal global signals.

In addition to the device global fast lines, each LAB row has two dedicated fast lines local to the row. This is ideal for high fanout control signals for a section of a design that may fit into a single LAB row. Each I/O band (with the exception of the top I/O band) has two dedicated row-global fast I/O pins to drive the row-global fast resources for the associated LAB. The dedicated local fast I/O pins have the same IOE as a regular I/O pin. The LE local interconnect can drive dedicated row-global fast lines to generate internal global signals specific to a row. There are no pin connections for buried LAB rows; LE local interconnects drive the row-global signals in those rows.

I/O Standard Support

Mercury device IOEs support the following I/O standards:

- LVTTL
- LVCMOS
- 1.8-V
- 2.5-V
- 3.3-V PCI
- 3.3-V PCI-X
- 3.3-V AGP (1×, 2×)
- LVDS
- LVPECL
- 3.3-V PCML
- GTL+
- HSTL class I and II
- SSTL-3 class I and II
- SSTL-2 class I and II
- CTT

For more information, see the following documents:

- *Application Note 39 (IEEE Std. 1149.1 (JTAG) Boundary-Scan Testing in Altera Devices)*
- Jam Programming & Test Language Specification

Generic Testing

Each Mercury device is functionally tested. Complete testing of each configurable static random access memory (SRAM) bit and all logic functionality ensures 100% yield. AC test measurements for Mercury devices are made under conditions equivalent to those shown in Figure 33. Multiple test patterns can be used to configure devices during all stages of the production flow.

Table 26. 1.8-V I/O Specifications *Note (10)*					
Symbol	**Parameter**	**Conditions**	**Minimum**	**Maximum**	**Units**
V_{CCIO}	Output supply voltage		1.71	1.89	V
V_{IH}	High-level input voltage		$0.65 \times V_{CCIO}$	4.1	V
V_{IL}	Low-level input voltage		–0.5	$0.35 \times V_{CCIO}$	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} = –2 mA	V_{CCIO} – 0.45		V
V_{OL}	Low-level output voltage	I_{OL} = 2 mA		0.45	V

Figures 34 and 35 show receiver input and transmitter output waveforms, respectively, for all differential I/O standards (LVPECL, 3.3-V PCML, LVDS, and HyperTransport technology).

Figure 34. Receiver Input Waveforms for Differential I/O Standards

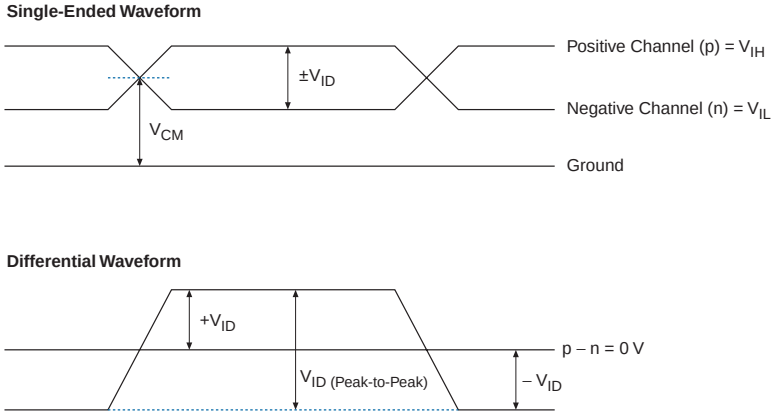


Table 39. 1.5-V HSTL Class I Specifications *Note (10)*

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Units
V_{CCIO}	I/O supply voltage		1.4	1.5	1.6	V
V_{REF}	Input reference voltage		0.68	0.75	0.9	V
V_{TT}	Termination voltage		0.7	0.75	0.8	V
V_{IH} (DC)	DC high-level input voltage		V_{REF} + 0.1			V
V_{IL} (DC)	DC low-level input voltage		–0.3		V_{REF} – 0.1	V
V_{IH} (AC)	AC high-level input voltage		V_{REF} + 0.2			V
V_{IL} (AC)	AC low-level input voltage				V_{REF} – 0.2	V
V_{OH}	High-level output voltage	I_{OH} = 8 mA	V_{CCIO} – 0.4			V
V_{OL}	Low-level output voltage	I_{OH} = –8 mA			0.4	V

Table 40. 1.5-V HSTL Class II Specifications *Note (10)*

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Units
V_{CCIO}	I/O supply voltage		1.4	1.5	1.6	V
V_{REF}	Input reference voltage		0.68	0.75	0.9	V
V_{TT}	Termination voltage		0.7	0.75	0.8	V
V_{IH} (DC)	DC high-level input voltage		V_{REF} + 0.1			V
V_{IL} (DC)	DC low-level input voltage		–0.3		V_{REF} – 0.1	V
V_{IH} (AC)	AC high-level input voltage		V_{REF} + 0.2			V
V_{IL} (AC)	AC low-level input voltage				V_{REF} – 0.2	V
V_{OH}	High-level output voltage	I_{OH} = 16 mA	V_{CCIO} – 0.4			V
V_{OL}	Low-level output voltage	I_{OH} = –16 mA			0.4	V

Table 41. CTT I/O Specifications

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Units
V_{CCIO}	I/O supply voltage		3.0	3.3	3.6	V
V_{TT}/V_{REF}	Termination and input reference voltage		1.35	1.5	1.65	V
V_{IH}	High-level input voltage		V_{REF} + 0.2			V
V_{IL}	Low-level input voltage				V_{REF} – 0.2	V
I_I	Input pin leakage current	0 < V_{IN} < V_{CCIO}			±10	µA
V_{OH}	High-level output voltage	I_{OH} = –8 mA	V_{REF} + 0.4			V
V_{OL}	Low-level output voltage	I_{OL} = 8 mA			V_{REF} – 0.4	V
I_O	Output leakage current (when output is high *Z*)	GND ð V_{OUT} ð V_{CCIO}			±10	µA

Notes to Tables 20 – 43.:

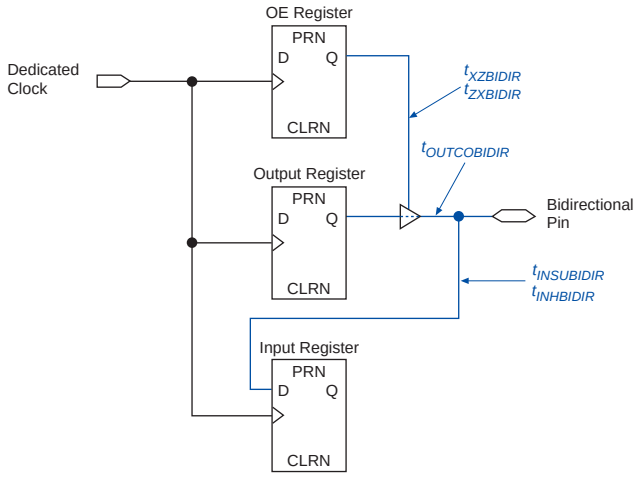
(1) See the *Operating Requirements for Altera Devices Data Sheet*.
(2) Minimum DC input is –0.5 V. During transitions, the inputs may undershoot to –0.5 V or overshoot to 4.1 V for input currents less than 100 mA and periods shorter than 20 ns.
(3) Maximum V_{CC} rise time is 100 ms, and V_{CC} must rise monotonically.
(4) V_{CCIO} maximum and minimum conditions for LVPECL, LVDS, RapidIO, and 3.3-V PCML are shown in parentheses.
(5) All pins, including dedicated inputs, clock, I/O, and JTAG pins, may be driven before V_{CCINT} and V_{CCIO} are powered.
(6) Typical values are for T_A = 25° C, V_{CCINT} = 1.8 V, and V_{CCIO} = 1.8 V, 2.5 V, and 3.3 V.
(7) These values are specified under the Mercury Device Recommended Operating Conditions shown in Table 3 on page 3.
(8) Input pins are grounded. In the test design, internal logic does not toggle. The test design does not use PLL or HSDI circuitry. All ESBs are in power-down mode.
(9) Pin pull-up resistance values will lower if an external source drives the pin higher than V_{CCIO}.
(10) Drive strength is programmable according to values in Table 11 on page 53.
(11) For more information on termination, see *AN 134: Using Programmable I/O Standards in Mercury Devices* or *AN 159: Using HSDI in Source-Synchronous Mode in Mercury Devices*.
(12) V_{REF} specifies the center point of the switching range.
(13) Capacitance is sample-tested only. Capacitance is measured using time-domain reflections (TDR). Measurement accuracy is within ±5%.

Timing Model

The high-performance multi-level FastTrack Interconnect routing resources ensure predictable performance, accurate simulation, and accurate timing analysis. The predictable performance of Mercury devices offer an advantage over FPGAs, which use a segmented connection scheme and therefore have unpredictable performance.

Figure 36 shows the timing model for bidirectional IOE pin timing. All registers are within the IOE.

Figure 36. Synchronous Bidirectional Pin External Timing Model



Tables 44 and 45 describe the Mercury device's external timing parameters.

Table 44. Mercury External Timing Parameters *Notes (1), (2)*

Symbol	Parameter	Conditions
t_{INSU}	Setup time with global clock at IOE register	
t_{INH}	Hold time with global clock at IOE register	
t_{OUTCO}	Clock-to-output delay with global clock at IOE register	C1 = 35 pF
$t_{INSUPLL}$	Setup time with PLL clock at IOE input register	
t_{INHPLL}	Hold time with PLL clock at IOE input register	
$t_{OUTCOPLL}$	Clock-to-output delay with PLL clock at IOE output register	C1 = 35 pF

Table 45. Mercury External Bidirectional Timing Parameters *Notes (1), (2)*

Symbol	Parameter	Conditions
$t_{INSUBIDIR}$	Setup time for bidirectional pins with gobal clock at IOE input register	
$t_{INHBIDIR}$	Hold time for bidirectional pins with global clock at IOE input register	
$t_{OUTCOBIDIR}$	Clock-to-output delay for bidirectional pins with global clock at IOE output register	C1 = 35 pF
$t_{XZBIDIR}$	Synchronous IOE output enable register to output buffer disable delay	C1 = 35 pF
$t_{ZXBIDIR}$	Synchronous IOE output enable register output buffer enable delay	C1 = 35 pF
$t_{INSUBIDIRPLL}$	Setup time for bidirectional pins with PLL clock at IOE input register	
$t_{INHBIDIRPLL}$	Hold time for bidirectional pins with PLL clock at IOE input register	
$t_{OUTCOBIDIRPLL}$	Clock-to-output delay for bidirectional pins with PLL clock at IOE output register	C1 = 35 pF
$t_{XZBIDIRPLL}$	Synchronous IOE output enable register to output buffer disable delay with PLL	C1 = 35 pF
$t_{ZXBIDIRPLL}$	Synchronous IOE output enable register output buffer enable delay with PLL	C1 = 35 pF

Notes to Tables 44 and 45:

(1) These timing parameters are sample-tested only.

(2) All timing parameters are either to and / or from pins, including global clock pins.

Table 48. EP1M120 External Timing Parameters *Note (1)*

Symbol	-7A Speed Grade		-8A Speed Grade		Unit
	Min	Max	Min	Max	
t_{INSU}	0.74		0.79		ns
t_{INH}	0.00		0.00		ns
t_{OUTCO}	2.00	3.50	2.00	4.10	ns
$t_{INSUPLL}$	0.62		0.75		ns
t_{INHPLL}	0.00		0.00		ns
$t_{OUTCOPLL}$	0.50	2.15	0.50	2.43	ns

Table 49. EP1M120 External Bidirectional Timing Parameters *Note (1)*

Symbol	-7A Speed Grade		-8A Speed Grade		Unit
	Min	Max	Min	Max	
$t_{INSUBIDIR}$	0.74		0.79		ns
$t_{INHBIDIR}$	0.00		0.00		ns
$t_{OUTCOBIDIR}$	2.00	3.50	2.00	4.10	ns
$t_{XZBIDIR}$		3.75		4.30	ns
$t_{ZXBIDIR}$ *(2)*		3.75		4.30	ns
$t_{ZXBIDIR}$ *(3)*		4.00		4.58	ns
$t_{INSUBIDIRPLL}$	0.62		0.75		ns
$t_{INHBIDIRPLL}$	0.00		0.00		ns
$t_{OUTCOBIDIRPLL}$	0.50	2.15	0.50	2.43	ns
$t_{XZBIDIRPLL}$		2.39		2.67	ns
$t_{ZXBIDIRPLL}$ *(2)*		2.39		2.67	ns
$t_{ZXBIDIRPLL}$ *(3)*		2.64		2.95	ns

Table 50. EP1M350 External Timing Parameters *Note (1)*

Symbol	-5 Speed Grade		-6 Speed Grade		-7 Speed Grade		Unit
	Min	Max	Min	Max	Min	Max	
t_{INSU}	0.60		0.57		0.71		ns
t_{INH}	0.00		0.00		0.00		ns
t_{OUTCO}	2.00	3.95	2.00	3.97	2.00	4.75	ns
$t_{INSUPLL}$	0.69		0.70		0.82		ns
t_{INHPLL}	0.00		0.00		0.00		ns
$t_{OUTCOPLL}$	0.50	2.23	0.50	2.23	0.50	2.69	ns

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.