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/ep1m350f780i6

Tables 2 and 3 show the Mercury™ FineLine BGA™ device package sizes, options, and I/O pin counts.

Table 2. Mercury Package Sizes

Feature	484-Pin FineLine BGA	780-Pin FineLine BGA
Pitch (mm)	1.00	1.00
Area (mm^2)	529	841
Length × width (mm × mm)	23 × 23	29 × 29

Table 3. Mercury Package Options & I/O Count

Device	484-Pin FineLine BGA	780-Pin FineLine BGA
EP1M120	303	
EP1M350		486

General Description

Mercury devices integrate high-speed differential transceivers and support for CDR with a speed-optimized PLD architecture. These transceivers are implemented through the dedicated serializer, deserializer, and clock recovery circuitry in the HSDI and incorporate support for the LVDS, LVPECL, and 3.3-V PCML I/O standards. This circuitry, together with enhanced I/O elements (IOEs) and support for numerous I/O standards, allows Mercury devices to meet high-speed interface requirements.

Mercury devices are the first PLDs optimized for core performance. These LUT-based, enhanced memory devices use a network of fast routing resources to achieve optimal performance. These resources are ideal for data-path, register-intensive, mathematical, digital signal processing (DSP), or communications designs.

Mercury devices include other features for performance such as quad-port RAM, CAM, general purpose PLLs, and dedicated circuitry for implementing multiplier circuits. Table 4 shows Mercury performance.

Table 4. Mercury Performance

Application	Resources Used		Performance			
	LEs	ESBs	-5 Speed Grade	-6 Speed Grade	-7 Speed Grade	Units
16-bit loadable counter *(1)*	16	0	400	400	400	MHz
32-bit loadable counter *(1)*	32	0	400	400	400	MHz
32-bit accumulator *(1)*	32	0	400	400	400	MHz
32-to-1 multiplexer	27	0	1.864	2.466	2.723	ns
32 × 64 asynchronous FIFO	103	2	290	258	242	MHz
8-bit, 37-tap FIR filter	251	1	290	240	205	MSPS

Note to Table 4:

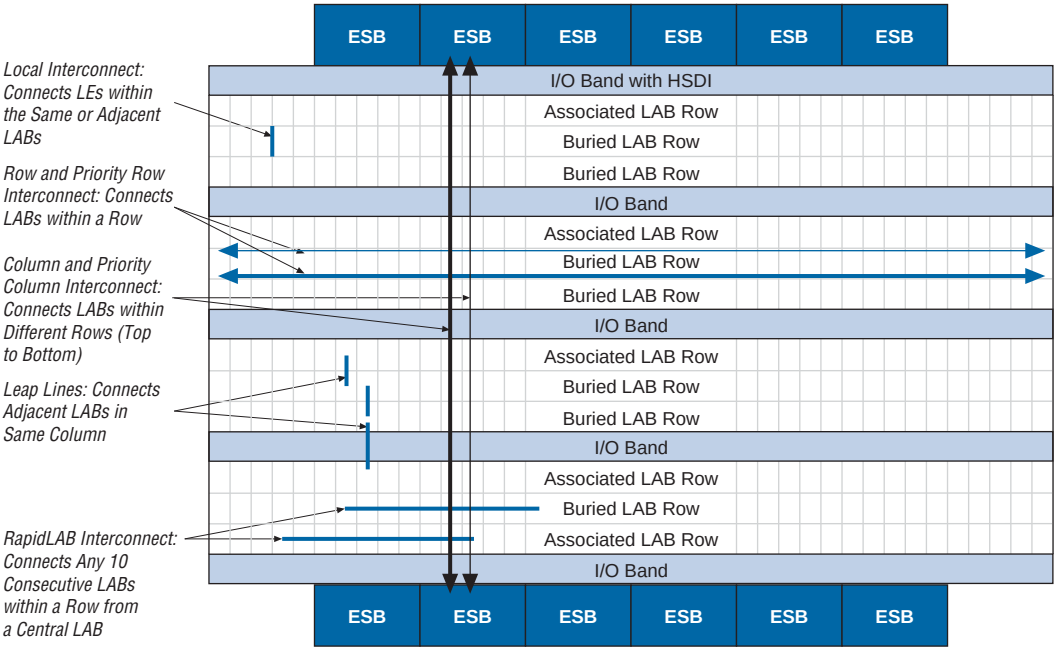
(1) The clock tree supports up to 400 MHz. Although the registered performance for these designs exceed 400 MHz, they are limited by the clock tree limit.

Configuration

The logic, circuitry, and interconnects in the Mercury architecture are configured with CMOS SRAM elements. Mercury devices are reconfigurable and are 100% tested prior to shipment. As a result, test vectors do not have to be generated for fault coverage purposes. Instead, the designer can focus on simulation and design verification. In addition, the designer does not need to manage inventories of different ASIC designs; Mercury devices can be configured on the board for the specific functionality required.

Mercury devices are configured at system power-up with data stored in an Altera® serial configuration device or provided by a system controller. Altera offers in-system programmability (ISP)-capable configuration devices, which configure Mercury devices via a serial data stream. Mercury devices can be configured in under 70 ms. Moreover, Mercury devices contain an optimized interface that permits microprocessors to configure Mercury devices serially or in parallel, synchronously or asynchronously. This interface also enables microprocessors to treat Mercury devices as memory and to configure the device by writing to a virtual memory location, simplifying reconfiguration.

Figure 1. Mercury Architecture Block Diagram *Note (1)*



Note to Figure 1:

(1) Figure 1 shows an EP1M120 device. Mercury devices have a varying number of rows, columns, and ESBs, as shown in Table 5.

Table 5 lists the resources available in Mercury devices.

Table 5. Mercury Device Resources

Device	LAB Rows	LAB Columns	I/O Row Bands	ESBs
EP1M120	12	40	5	12
EP1M350	18	80	4	28

Mercury device HSDI performance is finalized for certain speed grades. Also, the industrial-grade CDR specification is the same as the -6 speed grade for commercial-grade CDR specification. See Table 8.

Table 8. CDR & Source-Synchronous Data Rates

Device	Speed Grade	Number of Channels	Maximum CDR Data Rate (Gbps)	Maximum Source-Synchronous Data Rate (Mbps)
EP1M120	-5	8	1.25	840
	-6 *(1)*	8	1.25	840
	-7	8	1.0	840
EP1M350	-5	18	1.25	840
	-6 *(1)*	8 *(2)*	1.25	840
		10 *(2)*	1.0	840
	-7	18	1.0	840

Notes to Table 8:

(1) The -6 speed grade specifications apply for both commercial and industrial devices.

(2) EP1M350 devices can support any 8 channels at 1.25 Gbps. The other 10 channels must run at 1.0 Gbps or less.

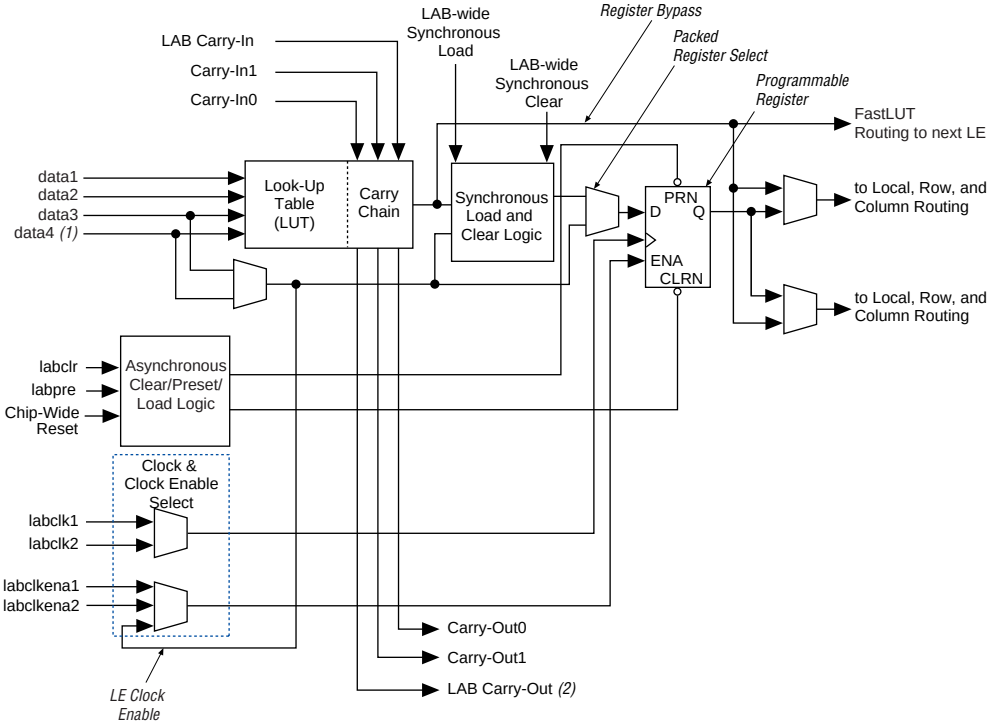
Logic & Interconnect

Mercury device logic is implemented in LEs. LE resources are used differently according to specific operating modes and the type of logic function being implemented. LEs are grouped into LABs in a row-based architecture. The multi-level FastTrack Interconnect structure provides the routing connection between LEs, ESBs, and IOEs.

Logic Array Block

Each LAB consists of 10 LEs, LE carry chains, multiplier circuitry, LAB control signals, local interconnect, and FastLUT connection lines. The local interconnect transfers signals between LEs within the same or adjacent LABs. FastLUT connections transfer the output of one LE to the adjacent LE for ultra-fast sequential LE connections within the same LAB. The Quartus II Compiler places associated logic within a LAB or adjacent LABs, allowing the use of fast local and FastLUT connections for high performance. Figure 5 shows the Mercury LAB structure.

Figure 7. Mercury LE



Notes to Figure 7:

(1) FastLUT interconnect uses the `data4` input.

(2) LAB carry-out can only be generated by LE 4 and / or LE 10.

Each LE's programmable register can be configured for D, T, JK, or SR operation. The register's clock, clock enable, and clear control signals can be driven by global signals, general-purpose I / O pins, or any internal logic. For combinatorial functions, the register is bypassed and the output of the LUT drives directly to the outputs of the LE.

Each LE has four data inputs that can drive the internal LUT. One of these inputs has a shorter delay than the others, improving overall LE performance. This input is chosen automatically by the Quartus II software as appropriate.

Each LE has two outputs that drive the local, row, and column routing resources. Each output can be driven independently by the LUT's or register's output. For example, the LUT can drive one output, while the register drives the other output. This feature, called register packing, improves device utilization because the register and the LUT can be used for unrelated functions. The LE can also drive out registered and unregistered versions of the LUT output.

LE Operating Modes

The Mercury LE can operate in one of the following modes:

- Normal
- Arithmetic
- Multiplier

Each operating mode uses LE resources differently. In each operating mode, eight available inputs to the LE—the four data inputs from the LAB local interconnect; `carry-in0`, `carry-in1` from the previous LE; the LAB carry-in from the previous carry-chain generation; and the FastLUT Connection input from the previous LE—are directed to different destinations to implement the desired logic function. LAB-wide signals provide clock, asynchronous clear, asynchronous preset, asynchronous load, synchronous clear, synchronous load, and clock enable control for the register. These LAB-wide signals are available in all normal and arithmetic LE modes.

The Quartus II software, in conjunction with parameterized functions such as LPM and DesignWare functions, automatically chooses the appropriate mode for common functions, such as counters, adders, and multipliers. If required, the designer can also create special-purpose functions that specify which LE operating mode to use for optimal performance.

Normal Mode

The normal mode is suitable for general logic applications and combinatorial functions. In normal mode, four data inputs from the LAB local interconnect and a single carry-in are inputs to a four-input LUT. The Quartus II Compiler automatically selects the carry-in or the `data3` signal as one of the inputs to the LUT. The LUT (combinatorial) output can be driven to the FastLUT connection to the next LE in the LAB. LEs in normal mode support packed registers. Figure 8 shows an LE in normal mode.

The Quartus II Compiler creates carry chains longer than ten LEs by linking LABs together automatically. For enhanced fitting, a long carry chain skips intermediate LABs in a row structure. A carry chain longer than one LAB skips either from an even-numbered LAB to the next even-numbered LAB, or from an odd-numbered LAB to the next odd-numbered LAB. For example, the last LE of the first LAB in a LAB row carries to the first LE of the third LAB in the same LAB row.

Multiplier Mode

Multiplier mode is used for implementing high-speed multipliers up to 16×16 in size. The LUT implements the partial product formation and summation in a single stage for a $N \times M$-bit multiply operation. A single LE can implement the summation of $A_N B_{M+1} + A_{N+1} B_M$ for the multiplier and multiplicand inputs. To increase the speed of the multiplication, LAB wide signals are used to control the partial product sum generation. These multiplier LAB-wide signals use the LABCLKENA1 and PRESET/ASYNCLOAD resources. The multiplier mode takes advantage of the CSLA circuitry for optimized sum and carry generation in the partial product sum. There is a special CSLA circuitry mode used for the multiplier where the carry chain runs vertically between LABs in the same column. The Quartus II Compiler automatically uses this special mode for dedicated multiplier implementation only. The summation of the multiplier and multiplicand bits is driven out along with the carry-out0 and carry-out1 bits. The combinatorial or registered versions of the sum can be driven out, allowing the multiplier to be pipelined.

The RapidLAB interconnect has dedicated fast connections to the LE inputs in multiplier mode, further increasing the speed of the multiplier. These dedicated connections allow RapidLAB lines to avoid delay incurred by driving onto local interconnects and then into the LE.

The Quartus II software implements parameterized functions that use the multiplier mode automatically when multiply operators are used.

Figure 11 shows a Mercury device LE in multiplier mode.

Figure 12. Partial Product Formation

The Mercury IOE includes programmable delays that can be activated to ensure zero hold times, minimum clock-to-output times, input IOE register-to-core register transfers, or core-to-output IOE register transfers. A path in which a pin directly drives a register may require the delay to ensure zero hold time, whereas a path in which a pin drives a register through combinatorial logic may not require the delay. Programmable delays exist for decreasing input pin to core and IOE input register delays. The Quartus II Compiler can program these delays automatically to minimize setup time while providing a zero hold time. Delays are also programmable for increasing the register to pin delays for output and / or output enable registers. A programmable delay exists for increasing the t_{ZX} delay to the output pin, which is required for ZBT interfaces. Table 10 shows the programmable delays for Mercury devices.

Table 10. Mercury Programmable Delay Chain

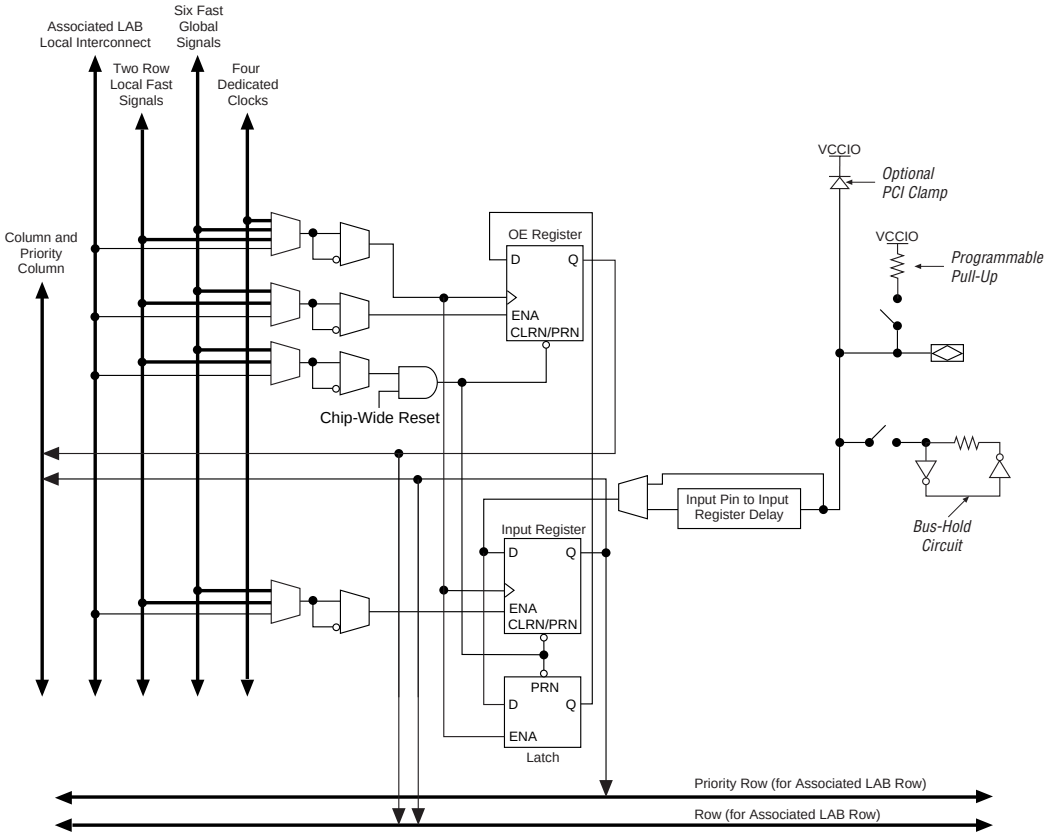
Programmable Delays	Quartus II Logic Option
Input pin to core delay *(1)*	Decrease input delay to internal cells
Input pin to input register delay	Decrease input delay to input register
Output propagation delay	Increase delay to output pin
Output enable register t_{CO} delay	Increase delay to OE pin
Output t_{ZX} delay	Increase t_{ZX} delay to output pin

Note to Table 10:
(1) This delay has four settings: off and three levels of delay.

The IOE registers in Mercury devices share the same source for clear or preset. Use of the preset / clear is programmable for each individual IOE. The register(s) can be programmed to power up high or low after configuration is complete. If programmed to power up low, an asynchronous clear can control the register(s). If programmed to power up high, an asynchronous preset can control the register(s). This feature prevents the inadvertent activation of another device's active-low input upon power-up. Figure 26 shows the IOE for Mercury devices.

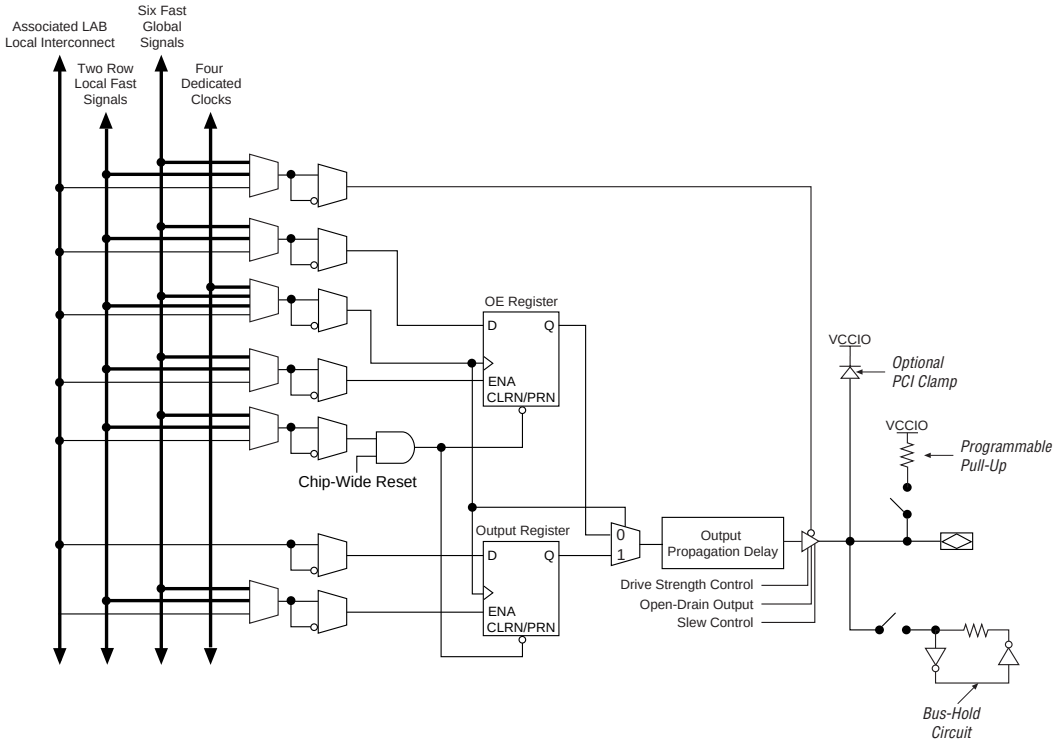
In Mercury device IOEs, the OE register is a multi-purpose register available as a second input or output register. When using the IOE for double data rate inputs, the input register and OE register are automatically configured as input registers to clock input double rate data on alternating edges. 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



When using the IOE for double data rate outputs, the output register and OE register are automatically configured to clock two data paths from LEs on rising clock edges. These register outputs are multiplexed by the clock to drive the output pin at a ×2 rate. The output register clocks the first bit out on the clock high time, while the OE register clocks the second bit out on the clock low time. Figure 28 shows the IOE configured for DDR output.

Figure 28. IOE Configured for DDR Output



Bidirectional DDR on an I/O pin is possible by using the IOE for DDR output and using LEs to acquire the double data rate input. Bidirectional DDR I/O pins support double data rate synchronous DRAM (DDR SDRAM) at 166 MHz (334 Mbps), which transfer data on a double data rate bidirectional bus. QDR SRAMs are also supported with DDR I/O pins on separate read and write ports.

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.

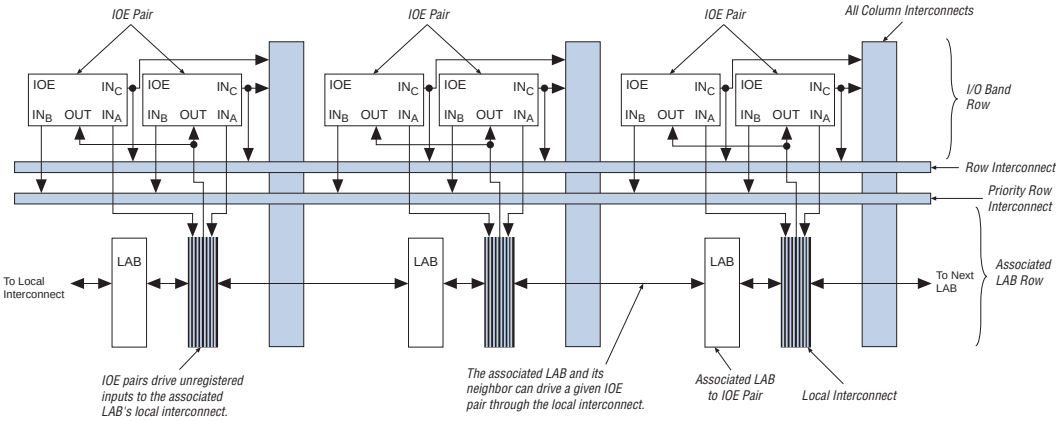
I/O Row Bands

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.

Each row of I/O pins has an associated LAB row for driving to and from the core of the Mercury device. For a given I/O band row, its associated LAB row is located below it with the exception of the bottom I/O band row. The bottom I/O band is located at the bottom periphery of the device, hence its associated LAB row is located above it. Figure 29 shows an example of an I/O band to associated LAB row interconnect in a Mercury device.

There is a maximum of two IOEs associated with each LAB in the associated LAB row. The local interconnect of the associated LAB drives the IOEs. Since local interconnect is shared with the LAB neighbor, any given LAB can directly drive up to four IOEs. The local interconnect drives the data and OE signals when the IOE is used as an output or bidirectional pin.

Figure 29. IOE Connection to Interconnects and Adjacent LAB *Note (1)*



Note to Figure 29:

(1) IN_A: unregistered input; IN_B: registered / unregistered input; IN_C: registered / unregistered input or OE register output in DDR mode.

The IOEs drive registered or combinatorial versions of input data into the device. The unregistered input data can be driven to the local interconnect (for fast input setup), row and priority row interconnect, and column and priority column interconnects. The registered data can also be driven to the same row and column resources. The OE register output can be fed back through column and row interconnects to implement DDR 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.

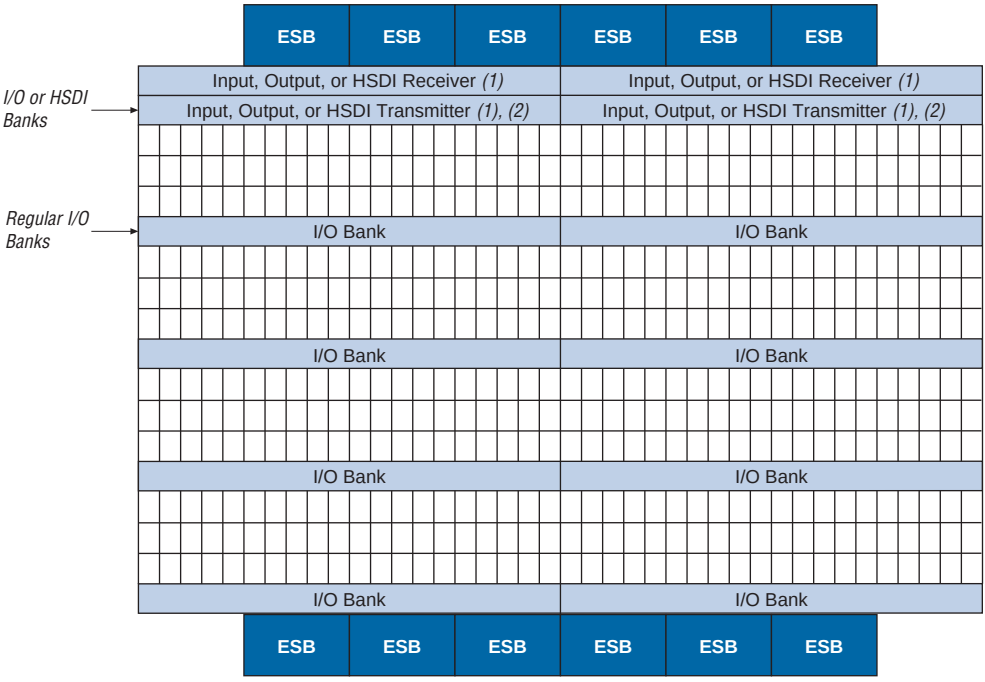
In addition to the device global fast lines, each LAB row has two dedicated fast lines local to the row. This is ideal for high fanout control signals for a section of a design that may fit into a single LAB row. Each I/O band (with the exception of the top I/O band) has two dedicated row-global fast 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

Figure 30. I/O Bank Layout



Notes to Figure 30:

(1) If HSDI I/O channels are not used, the HSDI banks can be used as regular I/O banks.

(2) When used as regular I/O banks, these banks must be set to the same V_{CCIO} level, but can have separate V_{REF} bank settings.

For more information on I/O standards, see *Application Note 117 (Using Selectable I/O Standards in Altera Devices)*.

MultiVolt I/O Interface

The Mercury architecture supports the MultiVolt I/O interface feature, which allows Mercury devices in all packages to interface with devices with different supply voltages. The devices have one set of VCC pins for internal operation and input buffers (VCCINT), and another set for I/O output drivers (VCCIO).

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.

Figure 32 shows the timing requirements for the JTAG signals.

Figure 32. Mercury JTAG Waveforms

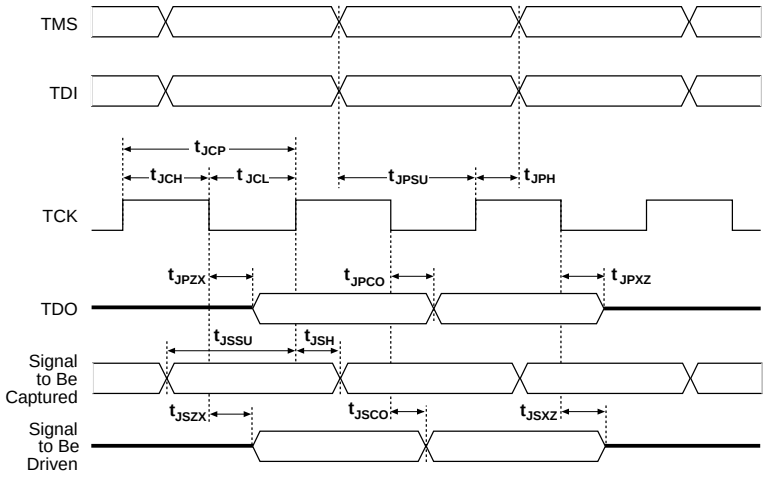


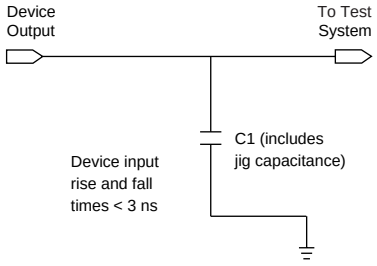
Table 19 shows the JTAG timing parameters and values for Mercury devices.

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 21. Mercury Device Recommended Operating Conditions

Symbol	Parameter	Conditions	Minimum	Maximum	Unit
V_{CCINT}	Supply voltage for internal logic and input buffers	*(3)*	1.71	1.89	V
V_{CCIO}	Supply voltage for output buffers, 3.3-V operation	*(3)*, *(4)*	3.00 (3.135)	3.60 (3.465)	V
	Supply voltage for output buffers, 2.5-V operation	*(3)*	2.375	2.625	V
	Supply voltage for output buffers, 1.8-V operation	*(3)*	1.71	1.89	V
	Supply voltage for output buffers, 1.5-V operation	*(3)*	1.4	1.6	V
V_I	Input voltage	*(2)*, *(5)*	–0.5	4.1	V
V_O	Output voltage		0	V_{CCIO}	V
T_J	Operating temperature	For commercial use	0	85	° C
		For industrial use	–40	100	° C
t_R	Input rise time			40	ns
t_F	Input fall time			40	ns

Table 22. Mercury Device DC Operating Conditions *Note (6)*, *(7)*

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Unit
I_I	Input pin leakage current	$V_I = V_{CCIOmax}$ to 0 V *(5)*	–10		10	µA
I_{OZ}	Tri-stated I/O pin leakage current	$V_O = V_{CCIOmax}$ to 0 V *(5)*	–10		10	µA
I_{CC0}	V_{CC} supply current (standby) for EP1M120 devices	For commercial use *(8)*		30		mA
		For Industrial use *(8)*		40		mA
	V_{CC} supply current (standby) for EP1M350 devices	For commercial use *(8)*		50		mA
		For Industrial use *(8)*		60		mA
R_{CONF}	Value of I/O pin pull-up resistor before and during configuration	V_{CCIO} = 3.0 V *(9)*	20		50	kΩ
		V_{CCIO} = 2.375 V *(9)*	30		80	kΩ
		V_{CCIO} = 1.71 V *(9)*	60		150	kΩ

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