# Lattice Semiconductor Corporation - LAE3-35EA-6LFTN256E Datasheet

**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 | 4125 |
| Number of Logic Elements/Cells | 33000 |
| Total RAM Bits | 1358848 |
| Number of I/O | 133 |
| Number of Gates | - |
| Voltage - Supply | 1.14V ~ 1.26V |
| Mounting Type | Surface Mount |
| Operating Temperature | -40°C ~ 125°C (TJ) |
| Package / Case | 256-BGA |
| Supplier Device Package | 256-FTBGA (17x17) |
| Purchase URL | https://www.e-xfl.com/product-detail/lattice-semiconductor/lae3-35ea-6lftn256e |

*Table 2-5. DLL Signals*

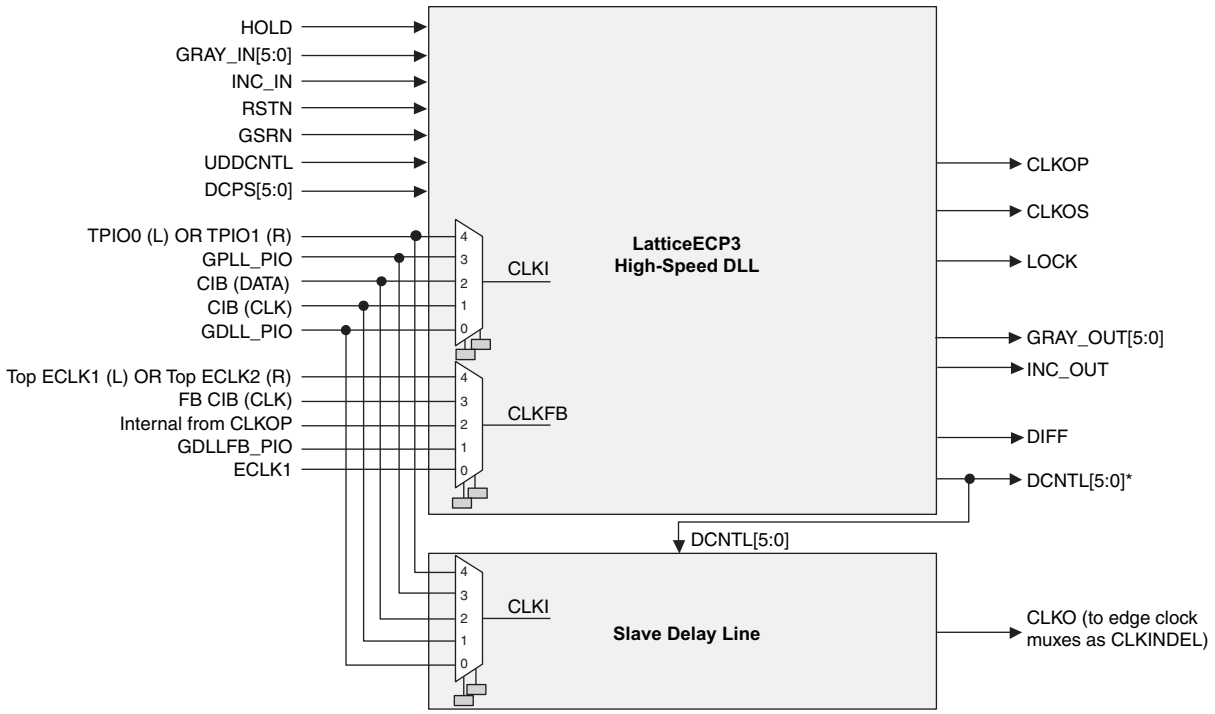
| Signal | I/O | Description |
|---|---|---|
| CLKI | I | Clock input from external pin or routing |
| CLKFB | I | DLL feed input from DLL output, clock net, routing or external pin |
| RSTN | I | Active low synchronous reset |
| ALUHOLD | I | Active high freezes the ALU |
| UDDCNTL | I | Synchronous enable signal (hold high for two cycles) from routing |
| CLKOP | O | The primary clock output |
| CLKOS | O | The secondary clock output with fine delay shift and/or division by 2 or by 4 |
| LOCK | O | Active high phase lock indicator |
| INCI | I | Incremental indicator from another DLL via CIB. |
| GRAYI[5:0] | I | Gray-coded digital control bus from another DLL in time reference mode. |
| DIFF | O | Difference indicator when DCNTL is difference than the internal setting and update is needed. |
| INCO | O | Incremental indicator to other DLLs via CIB. |
| GRAYO[5:0] | O | Gray-coded digital control bus to other DLLs via CIB |

LA-LatticeECP3 devices have two general DLLs and four Slave Delay lines, two per DLL. The DLLs are in the lowest EBR row and located adjacent to the EBR. Each DLL replaces one EBR block. One Slave Delay line is placed adjacent to the DLL and the duplicate Slave Delay line (in Figure 2-6) for the DLL is placed in the I/O ring between Banks 6 and 7 and Banks 2 and 3.

The outputs from the DLL and Slave Delay lines are fed to the clock distribution network.

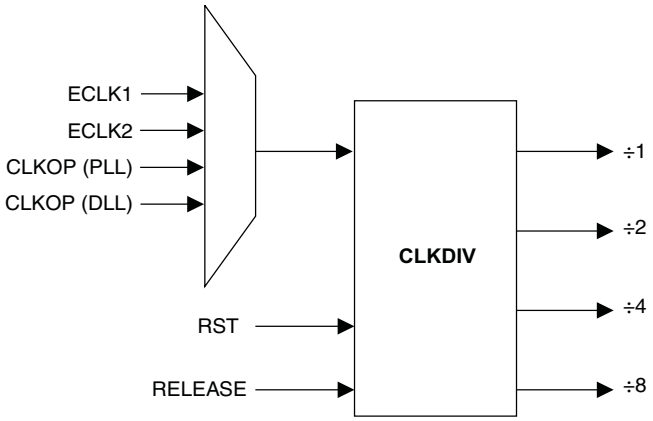
For more information, please see TN1178, LatticeECP3 sysCLOCK PLL/DLL Design and Usage Guide.

*Figure 2-6. Top-Level Block Diagram, High-Speed DLL and Slave Delay Line*




* This signal is not user accessible. It can only be used to feed the slave delay line.

*Figure 2-8. Clock Divider Connections*



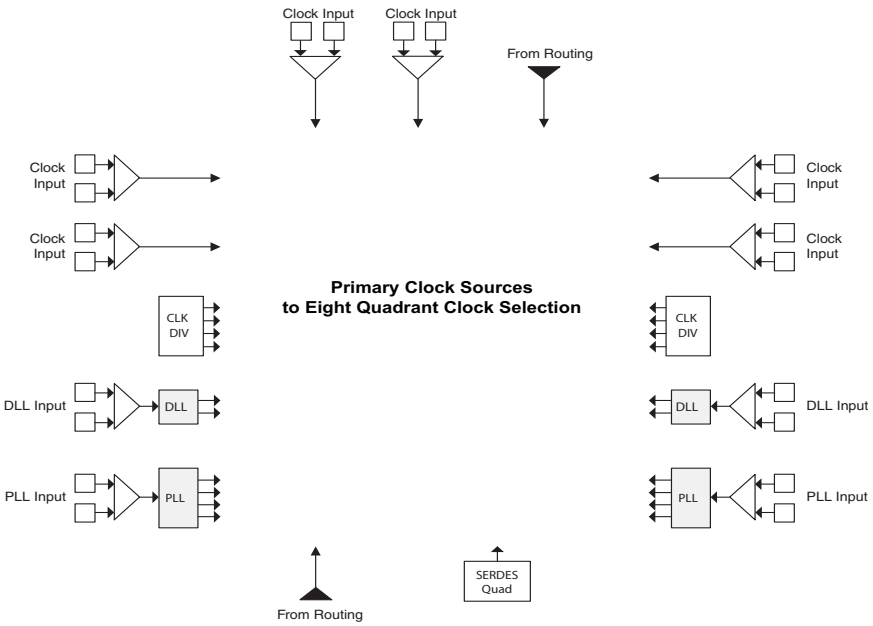

## Clock Distribution Network

LA-LatticeECP3 devices have eight quadrant-based primary clocks and eight secondary clock/control sources. Two high performance edge clocks are available on the top, left, and right edges of the device to support high speed interfaces. These clock sources are selected from external I/Os, the sysCLOCK PLLs, DLLs or routing. These clock sources are fed throughout the chip via a clock distribution system.

### Primary Clock Sources

LA-LatticeECP3 devices derive clocks from six primary source types: PLL outputs, DLL outputs, CLKDIV outputs, dedicated clock inputs, routing and SERDES Quad. LA-LatticeECP3 devices have two to four sysCLOCK PLLs and two DLLs, located on the left and right sides of the device. There are six dedicated clock inputs: two on the top side, two on the left side and two on the right side of the device. Figures 2-9 and 2-10 and show the primary clock sources for LA-LatticeECP3 devices.

*Figure 2-9. Primary Clock Sources for LA-LatticeECP3-17*

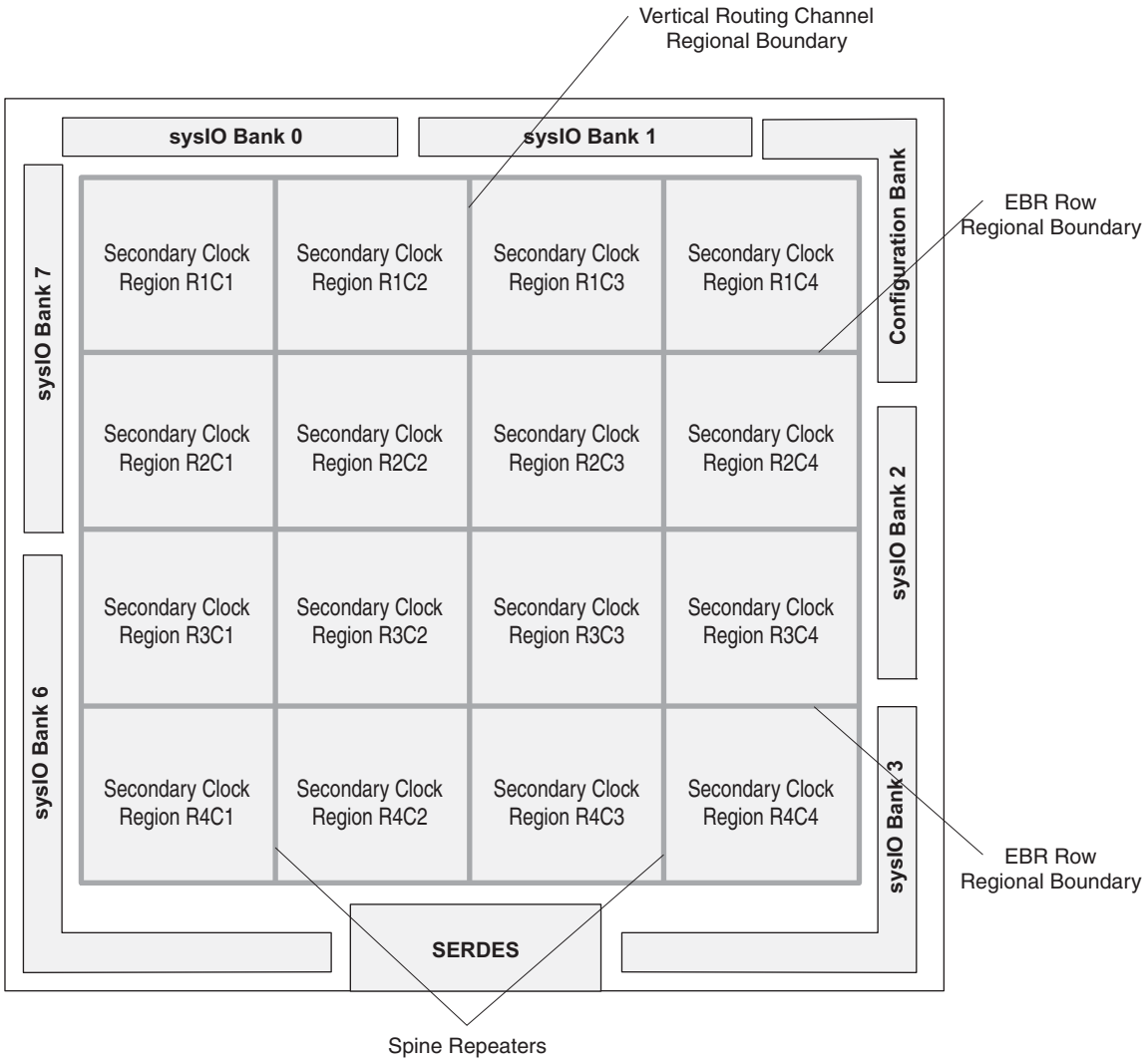



Note: Clock inputs can be configured in differential or single-ended mode.

*Table 2-6. Secondary Clock Regions*

| Device | Number of Secondary Clock Regions |
|---|---|
| LAE3-17 | 16 |
| LAE3-35 | 16 |

*Figure 2-14. LA-LatticeECP3-17 and LA-LatticeECP3-35 Secondary Clock Regions*



The edge clocks on the top, left, and right sides of the device can drive the secondary clocks or general routing resources of the device. The left and right side edge clocks also can drive the primary clock network through the clock dividers (CLKDIV).

## sysMEM Memory

LA-LatticeECP3 devices contain a number of sysMEM Embedded Block RAM (EBR). The EBR consists of an 18-Kbit RAM with memory core, dedicated input registers and output registers with separate clock and clock enable. Each EBR includes functionality to support true dual-port, pseudo dual-port, single-port RAM, ROM and FIFO buffers (via external PFUs).

### sysMEM Memory Block

The sysMEM block can implement single port, dual port or pseudo dual port memories. Each block can be used in a variety of depths and widths as shown in Table 2-7. FIFOs can be implemented in sysMEM EBR blocks by implementing support logic with PFUs. The EBR block facilitates parity checking by supporting an optional parity bit for each data byte. EBR blocks provide byte-enable support for configurations with18-bit and 36-bit data widths. For more information, please see TN1179, LatticeECP3 Memory Usage Guide.

***Table 2-7. sysMEM Block Configurations***

| Memory Mode | Configurations |
|---|---|
| Single Port | 16,384 x 1<br>8,192 x 2<br>4,096 x 4<br>2,048 x 9<br>1,024 x 18<br>512 x 36 |
| True Dual Port | 16,384 x 1<br>8,192 x 2<br>4,096 x 4<br>2,048 x 9<br>1,024 x 18 |
| Pseudo Dual Port | 16,384 x 1<br>8,192 x 2<br>4,096 x 4<br>2,048 x 9<br>1,024 x 18<br>512 x 36 |

### Bus Size Matching

All of the multi-port memory modes support different widths on each of the ports. The RAM bits are mapped LSB word 0 to MSB word 0, LSB word 1 to MSB word 1, and so on. Although the word size and number of words for each port varies, this mapping scheme applies to each port.

### RAM Initialization and ROM Operation

If desired, the contents of the RAM can be pre-loaded during device configuration. By preloading the RAM block during the chip configuration cycle and disabling the write controls, the sysMEM block can also be utilized as a ROM.

### Memory Cascading

Larger and deeper blocks of RAM can be created using EBR sysMEM Blocks. Typically, the Lattice design tools cascade memory transparently, based on specific design inputs.

### Single, Dual and Pseudo-Dual Port Modes

In all the sysMEM RAM modes the input data and address for the ports are registered at the input of the memory array. The output data of the memory is optionally registered at the output.

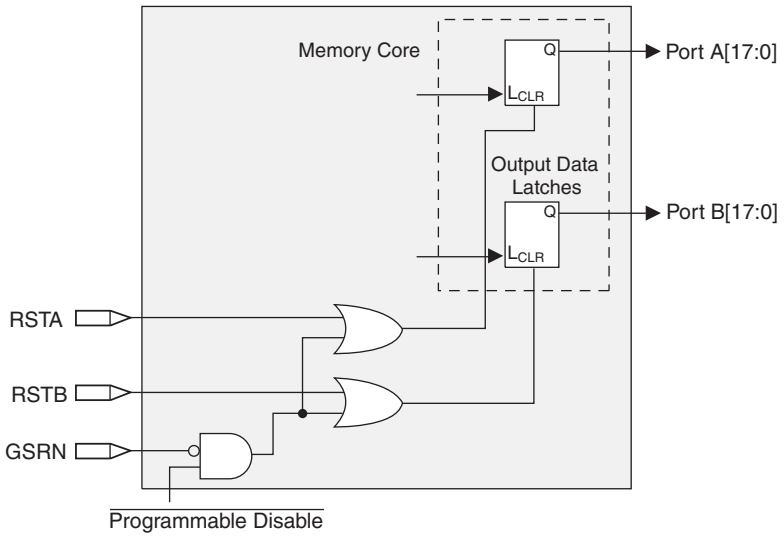
EBR memory supports the following forms of write behavior for single port or dual port operation:

1. **Normal** – Data on the output appears only during a read cycle. During a write cycle, the data (at the current address) does not appear on the output. This mode is supported for all data widths.
2. **Write Through** – A copy of the input data appears at the output of the same port during a write cycle. This mode is supported for all data widths.
3. **Read-Before-Write (EA devices only)** – When new data is written, the old content of the address appears at the output. This mode is supported for x9, x18, and x36 data widths.

### Memory Core Reset

The memory array in the EBR utilizes latches at the A and B output ports. These latches can be reset asynchronously or synchronously. RSTA and RSTB are local signals, which reset the output latches associated with Port A and Port B, respectively. The Global Reset (GSRN) signal can reset both ports. The output data latches and associated resets for both ports are as shown in Figure 2-21.

***Figure 2-21. Memory Core Reset***




For further information on the sysMEM EBR block, please see the list of technical documentation at the end of this data sheet.

## sysDSP™ Slice

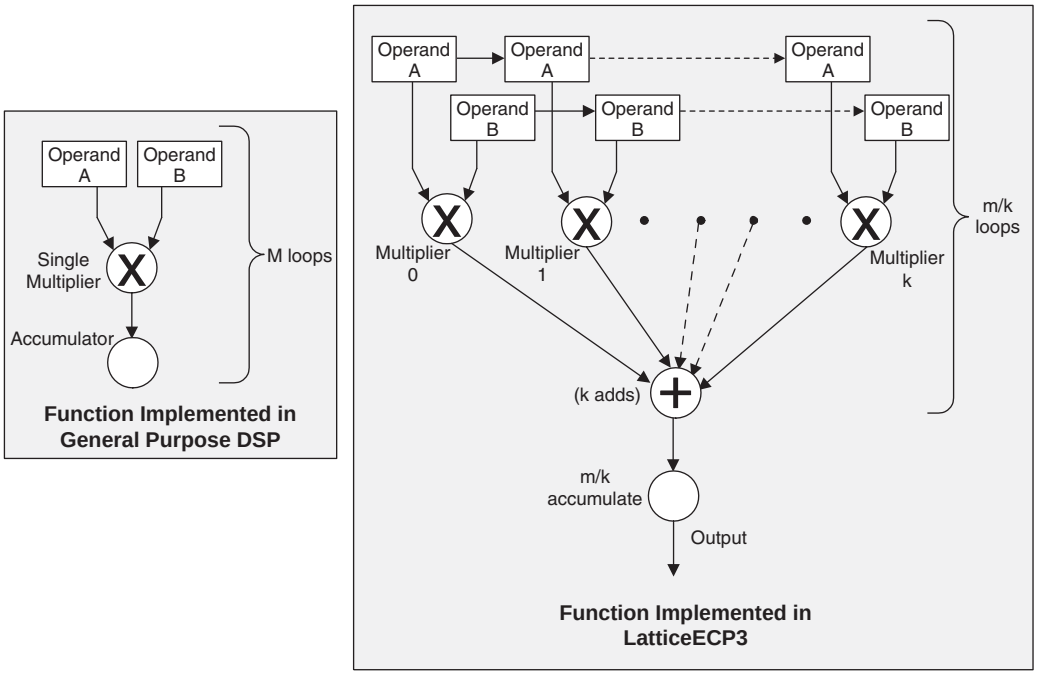
The LA-LatticeECP3 family provides an enhanced sysDSP architecture, making it ideally suited for low-cost, high-performance Digital Signal Processing (DSP) applications. Typical functions used in these applications are Finite Impulse Response (FIR) filters, Fast Fourier Transforms (FFT) functions, Correlators, Reed-Solomon/Turbo/Convolution encoders and decoders. These complex signal processing functions use similar building blocks such as multiply-adders and multiply-accumulators.

### sysDSP Slice Approach Compared to General DSP

Conventional general-purpose DSP chips typically contain one to four (Multiply and Accumulate) MAC units with fixed data-width multipliers; this leads to limited parallelism and limited throughput. Their throughput is increased by higher clock speeds. The LA-LatticeECP3, on the other hand, has many DSP slices that support different data

widths. This allows designers to use highly parallel implementations of DSP functions. Designers can optimize DSP performance vs. area by choosing appropriate levels of parallelism. Figure 2-22 compares the fully serial implementation to the mixed parallel and serial implementation.

***Figure 2-22. Comparison of General DSP and LA-LatticeECP3 Approaches***



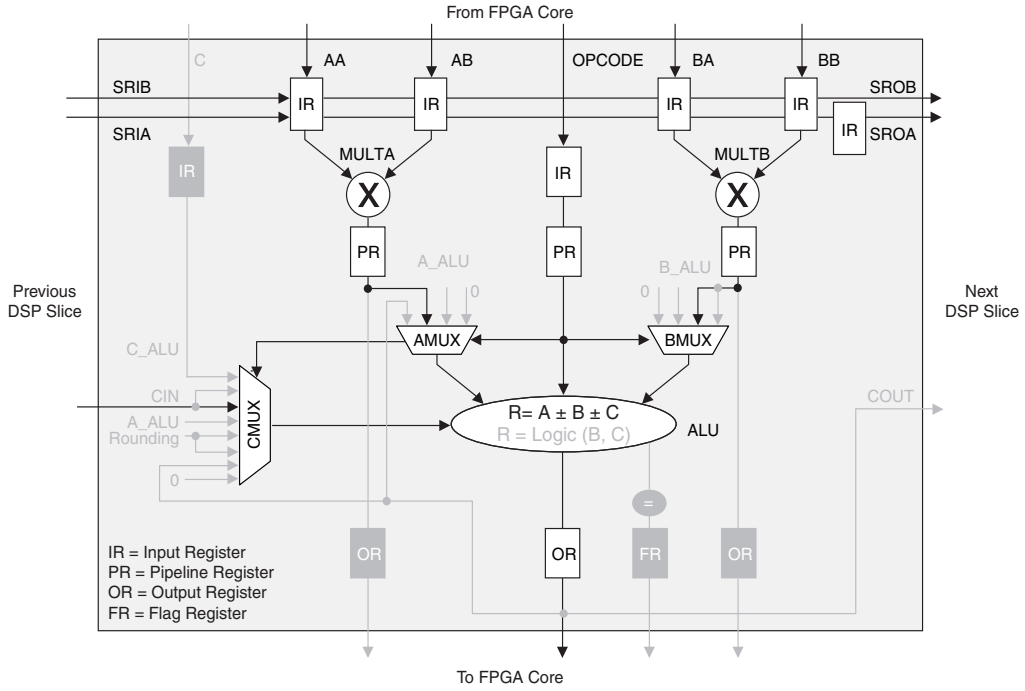

## sysDSP Slice Architecture Features

The LA-LatticeECP3 sysDSP Slice has been significantly enhanced to provide functions needed for advanced processing applications. These enhancements provide improved flexibility and resource utilization.

The LA-LatticeECP3 sysDSP Slice supports many functions that include the following:

- Multiply (one 18x36, two 18x18 or four 9x9 Multiplies per Slice)
- Multiply (36x36 by cascading across two sysDSP slices)
- Multiply Accumulate (up to 18x36 Multipliers feeding an Accumulator that can have up to 54-bit resolution)
- Two Multiplies feeding one Accumulate per cycle for increased processing with lower latency (two 18x18 Multiplies feed into an accumulator that can accumulate up to 52 bits)
- Flexible saturation and rounding options to satisfy a diverse set of applications situations
- Flexible cascading across DSP slices
  - Minimizes fabric use for common DSP and ALU functions
  - Enables implementation of FIR Filter or similar structures using dedicated sysDSP slice resources only
  - Provides matching pipeline registers
  - Can be configured to continue cascading from one row of sysDSP slices to another for longer cascade chains
- Flexible and Powerful Arithmetic Logic Unit (ALU) Supports:
  - Dynamically selectable ALU OPCODE
  - Ternary arithmetic (addition/subtraction of three inputs)
  - Bit-wise two-input logic operations (AND, OR, NAND, NOR, XOR and XNOR)
  - Eight flexible and programmable ALU flags that can be used for multiple pattern detection scenarios, such

*Figure 2-30. MULTADDSUBSUM Slice 1*




# Advanced sysDSP Slice Features

## Cascading

The LA-LatticeECP3 sysDSP slice has been enhanced to allow cascading. Adder trees are implemented fully in sysDSP slices, improving the performance. Cascading of slices uses the signals CIN, COUT and C Mux of the slice.

## Addition

The LA-LatticeECP3 sysDSP slice allows for the bypassing of multipliers and cascading of adder logic. High performance adder functions are implemented without the use of LUTs. The maximum width adders that can be implemented are 54-bit.

## Rounding

The rounding operation is implemented in the ALU and is done by adding a constant followed by a truncation operation. The rounding methods supported are:

- Rounding to zero (RTZ)
- Rounding to infinity (RTI)
- Dynamic rounding
- Random rounding
- Convergent rounding

Two adjacent PIOs can be joined to provide a differential I/O pair (labeled as “T” and “C”) as shown in Figure 2-31. The PAD Labels “T” and “C” distinguish the two PIOs. Approximately 50% of the PIO pairs on the left and right edges of the device can be configured as true LVDS outputs. All I/O pairs can operate as LVDS inputs.

***Table 2-11. PIO Signal List***

| Name | Type | Description |
|---|---|---|
| INDD | Input Data | Register bypassed input. This is not the same port as INCK. |
| IPA, INA, IPB, INB | Input Data | Ports to core for input data |
| OPOSA, ONEGA[1], OPOSB, ONEGB[1] | Output Data | Output signals from core. An exception is the ONEGB port, used for tristate logic at the DQS pad. |
| CE | PIO Control | Clock enables for input and output block flip-flops. |
| SCLK | PIO Control | System Clock (PCLK) for input and output/TS blocks. Connected from clock ISB. |
| LSR | PIO Control | Local Set/Reset |
| ECLK1, ECLK2 | PIO Control | Edge clock sources. Entire PIO selects one of two sources using mux. |
| ECLKDQSR[1] | Read Control | From DQS_STROBE, shifted strobe for memory interfaces only. |
| DDRCLKPOL[1] | Read Control | Ensures transfer from DQS domain to SCLK domain. |
| DDRLAT[1] | Read Control | Used to guarantee INDDRX2 gearing by selectively enabling a D-Flip-Flop in datapath. |
| DEL[3:0] | Read Control | Dynamic input delay control bits. |
| INCK | To Clock Distribution and PLL | PIO treated as clock PIO, path to distribute to primary clocks and PLL. |
| TS | Tristate Data | Tristate signal from core (SDR) |
| DQCLK0[1], DQCLK1[1] | Write Control | Two clocks edges, 90 degrees out of phase, used in output gearing. |
| DQSW[2] | Write Control | Used for output and tristate logic at DQS only. |
| DYNDEL[7:0] | Write Control | Shifting of write clocks for specific DQS group, using 6:0 each step is approximately 25ps, 128 steps. Bit 7 is an invert (timing depends on input frequency). There is also a static control for this 8-bit setting, enabled with a memory cell. |
| DCNTL[6:0] | PIO Control | Original delay code from DDR DLL |
| DATAVALID[1] | Output Data | Status flag from DATAVALID logic, used to indicate when input data is captured in IOLOGIC and valid to core. |
| READ | For DQS_Strobe | Read signal for DDR memory interface |
| DQSI | For DQS_Strobe | Unshifted DQS strobe from input pad |
| PRMBDET | For DQS_Strobe | DQSI biased to go high when DQSI is tristate, goes to input logic block as well as core logic. |
| GSRN | Control from routing | Global Set/Reset |

1. Signals available on left/right/top edges only.
2. Selected PIO.

# PIO

The PIO contains four blocks: an input register block, output register block, tristate register block and a control logic block. These blocks contain registers for operating in a variety of modes along with the necessary clock and selection logic.

## Input Register Block

The input register blocks for the PIOs, in the left, right and top edges, contain delay elements and registers that can be used to condition high-speed interface signals, such as DDR memory interfaces and source synchronous interfaces, before they are passed to the device core. Figure 2-32 shows the input register block for the left, right and top edges. The input register block for the bottom edge contains one element to register the input signal and no DDR registers. The following description applies to the input register block for PIOs in the left, right and top edges only.

### Control Logic Block

The control logic block allows the selection and modification of control signals for use in the PIO block.

## DDR Memory Support

Certain PICs have additional circuitry to allow the implementation of high-speed source synchronous and DDR, DDR2 and DDR3 memory interfaces. The support varies by the edge of the device as detailed below.

### Left and Right Edges

The left and right sides of the PIC have fully functional elements supporting DDR, DDR2, and DDR3 memory interfaces. One of every 12 PIOs supports the dedicated DQS pins with the DQS control logic block. Figure 2-34 shows the DQS bus spanning 11 I/O pins. Two of every 12 PIOs support the dedicated DQS and DQS# pins with the DQS control logic block.

### Bottom Edge

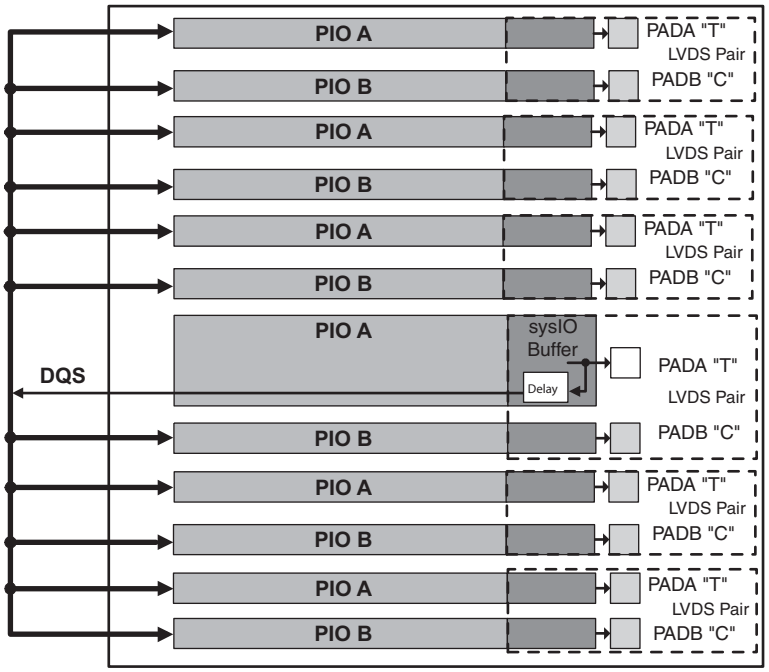
PICs on the bottom edge of the device do not support DDR memory and Generic DDR interfaces.

### Top Edge

PICs on the top side are similar to the PIO elements on the left and right sides but do not support gearing on the output registers. Hence, the modes to support output/tristate DDR3 memory are removed on the top side.

The exact DQS pins are shown in a dual function in the Logic Signal Connections table in this data sheet. Additional detail is provided in the Signal Descriptions table. The DQS signal from the bus is used to strobe the DDR data from the memory into input register blocks. Interfaces on the left, right and top edges are designed for DDR memories that support 10 bits of data.

***Figure 2-34. DQS Grouping on the Left, Right and Top Edges***




### DLL Calibrated DQS Delay Block

Source synchronous interfaces generally require the input clock to be adjusted in order to correctly capture data at the input register. For most interfaces, a PLL is used for this adjustment. However, in DDR memories the clock

(referred to as DQS) is not free-running so this approach cannot be used. The DQS Delay block provides the required clock alignment for DDR memory interfaces.

The delay required for the DQS signal is generated by two dedicated DLLs (DDR DLL) on opposite side of the device. Each DLL creates DQS delays in its half of the device as shown in Figure 2-36. The DDR DLL on the left side will generate delays for all the DQS Strobe pins on Banks 0, 7 and 6 and DDR DLL on the right will generate delays for all the DQS pins on Banks 1, 2 and 3. The DDR DLL loop compensates for temperature, voltage and process variations by using the system clock and DLL feedback loop. DDR DLL communicates the required delay to the DQS delay block using a 7-bit calibration bus (DCNTL[6:0])

The DQS signal (selected PIOs only, as shown in Figure 2-34) feeds from the PAD through a DQS control logic block to a dedicated DQS routing resource. The DQS control logic block consists of DQS Read Control logic block that generates control signals for the read side and DQS Write Control logic that generates the control signals required for the write side. A more detailed DQS control diagram is shown in Figure 2-36, which shows how the DQS control blocks interact with the data paths.

The DQS Read control logic receives the delay generated by the DDR DLL on its side and delays the incoming DQS signal by 90 degrees. This delayed ECLKDQSR is routed to 10 or 11 DQ pads covered by that DQS signal. This block also contains a polarity control logic that generates a DDRCLKPOL signal, which controls the polarity of the clock to the sync registers in the input register blocks. The DQS Read control logic also generates a DDRLAT signal that is in the input register block to transfer data from the first set of DDR register to the second set of DDR registers when using the DDRX2 gearbox mode for DDR3 memory interface.

The DQS Write control logic block generates the DQCLK0 and DQCLK1 clocks used to control the output gearing in the Output register block which generates the DDR data output and the DQS output. They are also used to control the generation of the DQS output through the DQS output register block. In addition to the DCNTL [6:0] input from the DDR DLL, the DQS Write control block also uses a Dynamic Delay DYN DEL [7:0] attribute which is used to further delay the DQS to accomplish the write leveling found in DDR3 memory. Write leveling is controlled by the DDR memory controller implementation. The DYN DELAY can set 128 possible delay step settings. In addition, the most significant bit will invert the clock for a 180-degree shift of the incoming clock. This will generate the DQSW signal used to generate the DQS output in the DQS output register block.

Figure 2-35 and Figure 2-36 show how the DQS transition signals that are routed to the PIOs.

Please see TN1180, LatticeECP3 High-Speed I/O Interface for more information on this topic.

system FPGA. After the system is running with a basic configuration, a new boot image can be downloaded remotely and stored in a separate location in the configuration storage device. Any time after the update the LA-LatticeECP3 can be re-booted from this new configuration file. If there is a problem, such as corrupt data during download or incorrect version number with this new boot image, the LA-LatticeECP3 device can revert back to the original backup golden configuration and try again. This all can be done without power cycling the system. For more information, please see TN1169, LatticeECP3 sysCONFIG Usage Guide.

## Soft Error Detect (SED) Support

LA-LatticeECP3 devices have dedicated logic to perform Cycle Redundancy Code (CRC) checks. During configuration, the configuration data bitstream can be checked with the CRC logic block. In addition, the LA-LatticeECP3 device can also be programmed to utilize a Soft Error Detect (SED) mode that checks for soft errors in configuration SRAM. The SED operation can be run in the background during user mode. If a soft error occurs, during user mode (normal operation) the device can be programmed to generate an error signal.

For further information on SED support, please see TN1184, LatticeECP3 Soft Error Detection (SED) Usage Guide.

## External Resistor

LA-LatticeECP3 devices require a single external, 10K ohm ±1% value between the XRES pin and ground. Device configuration will not be completed if this resistor is missing. There is no boundary scan register on the external resistor pad.

## On-Chip Oscillator

Every LA-LatticeECP3 device has an internal CMOS oscillator which is used to derive a Master Clock (MCCLK) for configuration. The oscillator and the MCCLK run continuously and are available to user logic after configuration is completed. The software default value of the MCCLK is nominally 2.5MHz. Table 2-16 lists all the available MCCLK frequencies. When a different Master Clock is selected during the design process, the following sequence takes place:

1. Device powers up with a nominal Master Clock frequency of 3.1MHz.
2. During configuration, users select a different master clock frequency.
3. The Master Clock frequency changes to the selected frequency once the clock configuration bits are received.
4. If the user does not select a master clock frequency, then the configuration bitstream defaults to the MCCLK frequency of 2.5MHz.

This internal CMOS oscillator is available to the user by routing it as an input clock to the clock tree. For further information on the use of this oscillator for configuration or user mode, please see TN1169, LatticeECP3 sysCONFIG Usage Guide.

***Table 2-16. Selectable Master Clock (MCCLK) Frequencies During Configuration (Nominal)***

| MCCLK (MHz) | MCCLK (MHz) |
|---|---|
| | 10 |
| 2.5[1] | 13 |
| 4.3 | 15[2] |
| 5.4 | 20 |
| 6.9 | 26 |
| 8.1 | 33[3] |
| 9.2 | |

1. Software default MCCLK frequency. Hardware default is 3.1MHz.
2. Maximum MCCLK with encryption enabled.
3. Maximum MCCLK without encryption.

## RSDS25E

The LA-LatticeECP3 devices support differential RSDS and RSDSE standards. This standard is emulated using complementary LVCMOS outputs in conjunction with a parallel resistor across the driver outputs. The RSDS input standard is supported by the LVDS differential input buffer. The scheme shown in Figure 3-4 is one possible solution for RSDS standard implementation. Resistor values in Figure 3-4 are industry standard values for 1% resistors.

***Figure 3-4. RSDS25E (Reduced Swing Differential Signaling)***

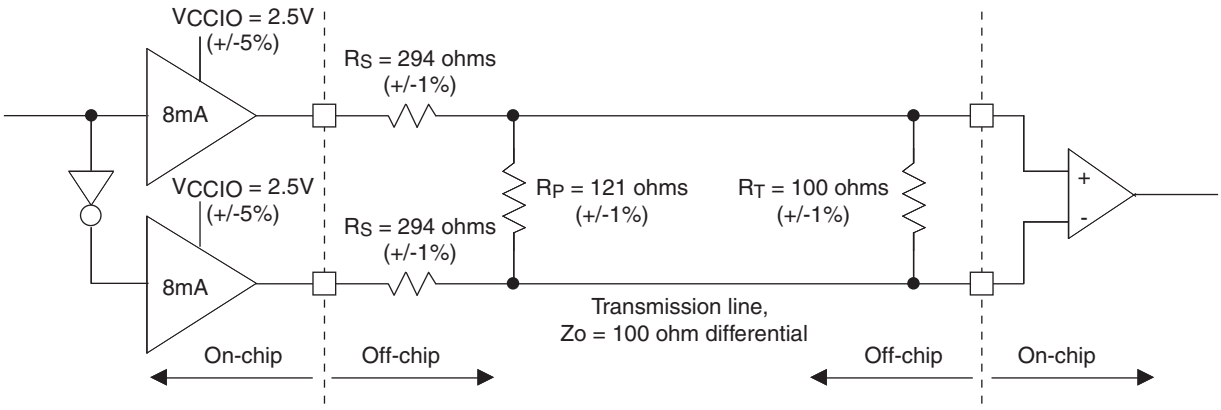



***Table 3-6. RSDS25E DC Conditions[1]***

**Over Recommended Operating Conditions**

| Parameter | Description | Typical | Units |
|---|---|---|---|
| $V_{CCIO}$ | Output Driver Supply (+/-5%) | 2.50 | V |
| $Z_{OUT}$ | Driver Impedance | 20 | Ω |
| $R_S$ | Driver Series Resistor (+/-1%) | 294 | Ω |
| $R_P$ | Driver Parallel Resistor (+/-1%) | 121 | Ω |
| $R_T$ | Receiver Termination (+/-1%) | 100 | Ω |
| $V_{OH}$ | Output High Voltage | 1.35 | V |
| $V_{OL}$ | Output Low Voltage | 1.15 | V |
| $V_{OD}$ | Output Differential Voltage | 0.20 | V |
| $V_{CM}$ | Output Common Mode Voltage | 1.25 | V |
| $Z_{BACK}$ | Back Impedance | 101.5 | Ω |
| $I_{DC}$ | DC Output Current | 3.66 | mA |

1. For input buffer, see LVDS table.

| Parameter | Description | Device | -6 / -6L | | Units |
|---|---|---|---|---|---|
| | | | Min. | Max. | |
| **GenericDDRX1 Output with Clock and Data Aligned at Pin (GDDRX1_TX.SCLK.Aligned)[10]** | | | | | |
| $t_{DIBGDDR}$ | Data Invalid Before Clock<br>Data Invalid After Clock | LAE3-35EA | - | 321 | ps |
| $t_{DIAGDDR}$ | | LAE3-35EA | - | 321 | ps |
| $f_{MAX\_GDDR}$ | DDRX1 Clock Frequency | LAE3-35EA | - | 250 | MHz |
| $t_{DIBGDDR}$ | Data Invalid Before Clock | LAE3-17EA | - | 321 | ps |
| $t_{DIAGDDR}$ | Data Invalid After Clock | LAE3-17EA | - | 321 | ps |
| $f_{MAX\_GDDR}$ | DDRX1 Clock Frequency | LAE3-17EA | - | 250 | MHz |
| **GenericDDRX1 Output with Clock and Data (<10 Bits Wide) Centered at Pin (GDDRX1_TX.DQS.Centered)10** | | | | | |
| **Left and Right Sides** | | | | | |
| $t_{DVBGDDR}$ | Data Valid Before CLK | LAE3-35EA | 676 | - | ps |
| $t_{DVAGDDR}$ | Data Valid After CLK | LAE3-35EA | 676 | - | ps |
| $f_{MAX\_GDDR}$ | DDRX1 Clock Frequency | LAE3-35EA | - | 250 | MHz |
| $t_{DVBGDDR}$ | Data Valid Before CLK | LAE3-17EA | 670 | - | ps |
| $t_{DVAGDDR}$ | Data Valid After CLK | LAE3-17EA | 670 | - | ps |
| $f_{MAX\_GDDR}$ | DDRX1 Clock Frequency | LAE3-17EA | - | 250 | MHz |
| **GenericDDRX2 Output with Clock and Data (>10 Bits Wide) Aligned at Pin (GDDRX2_TX.Aligned)** | | | | | |
| **Left and Right Sides** | | | | | |
| $t_{DIBGDDR}$ | Data Invalid Before Clock | All Devices | - | 220 | ps |
| $t_{DIAGDDR}$ | Data Invalid After Clock | All Devices | - | 220 | ps |
| $f_{MAX\_GDDR}$ | DDRX2 Clock Frequency | All Devices | - | 375 | MHz |
| **GenericDDRX2 Output with Clock and Data (>10 Bits Wide) Centered at Pin Using DQSDLL (GDDRX2_TX.DQS-DLL.Centered)[11]** | | | | | |
| **Left and Right Sides** | | | | | |
| $t_{DVBGDDR}$ | Data Valid Before CLK | All Devices | 431 | - | ps |
| $t_{DVAGDDR}$ | Data Valid After CLK | All Devices | 432 | - | ps |
| $f_{MAX\_GDDR}$ | DDRX2 Clock Frequency | All Devices | - | 375 | MHz |
| **GenericDDRX2 Output with Clock and Data (>10 Bits Wide) Centered at Pin Using PLL (GDDRX2_TX.PLL.Centered)10** | | | | | |
| **Left and Right Sides** | | | | | |
| $t_{DVBGDDR}$ | Data Valid Before CLK | All Devices | 431 | - | ps |
| $t_{DVAGDDR}$ | Data Valid After CLK | All Devices | 432 | - | ps |
| $f_{MAX\_GDDR}$ | DDRX2 Clock Frequency | All Devices | - | 375 | MHz |

## LA-LatticeECP3 Family Timing Adders [1, 2, 3, 4, 5]

**Over Recommended Operating Conditions**

| Buffer Type | Description | -6 / -6L | Units |
|---|---|---|---|
| **Input Adjusters** | | | |
| LVDS25E | LVDS, Emulated, VCCIO = 2.5V | -0.04 | ns |
| LVDS25 | LVDS, VCCIO = 2.5V | -0.04 | ns |
| BLVDS25 | BLVDS, Emulated, VCCIO = 2.5V | -0.04 | ns |
| MLVDS25 | MLVDS, Emulated, VCCIO = 2.5V | -0.04 | ns |
| RSDS25 | RSDS, VCCIO = 2.5V | -0.04 | ns |
| PPLVDS | Point-to-Point LVDS | -0.04 | ns |
| TRLVDS | Transition-Reduced LVDS | -0.04 | ns |
| HYPT | HyperTransport | -0.04 | ns |
| Mini MLVDS | Mini LVDS | -0.04 | ns |
| LVPECL33 | LVPECL, Emulated, VCCIO = 3.0V | -0.04 | ns |
| HSTL18_I | HSTL_18 class I, VCCIO = 1.8V | 0.14 | ns |
| HSTL18_II | HSTL_18 class II, VCCIO = 1.8V | 0.14 | ns |
| HSTL18D_I | Differential HSTL 18 class I | 0.14 | ns |
| HSTL18D_II | Differential HSTL 18 class II | 0.14 | ns |
| HSTL15_I | HSTL_15 class I, VCCIO = 1.5V | 0.14 | ns |
| HSTL15D_I | Differential HSTL 15 class I | 0.14 | ns |
| SSTL33_I | SSTL_3 class I, VCCIO = 3.0V | 0.30 | ns |
| SSTL33_II | SSTL_3 class II, VCCIO = 3.0V | 0.30 | ns |
| SSTL33D_I | Differential SSTL_3 class I | 0.30 | ns |
| SSTL33D_II | Differential SSTL_3 class II | 0.30 | ns |
| SSTL25_I | SSTL_2 class I, VCCIO = 2.5V | 0.17 | ns |
| SSTL25_II | SSTL_2 class II, VCCIO = 2.5V | 0.17 | ns |
| SSTL25D_I | Differential SSTL_2 class I | 0.17 | ns |
| SSTL25D_II | Differential SSTL_2 class II | 0.17 | ns |
| SSTL18_I | SSTL_18 class I, VCCIO = 1.8V | 0.04 | ns |
| SSTL18_II | SSTL_18 class II, VCCIO = 1.8V | 0.04 | ns |
| SSTL18D_I | Differential SSTL_18 class I | 0.04 | ns |
| SSTL18D_II | Differential SSTL_18 class II | 0.04 | ns |
| SSTL15 | SSTL_15, VCCIO = 1.5V | 0.03 | ns |
| SSTL15D | Differential SSTL_15 | -0.04 | ns |
| LVTTL33 | LVTTL, VCCIO = 3.0V | 0.05 | ns |
| LVCMOS33 | LVCMOS, VCCIO = 3.0V | 0.05 | ns |
| LVCMOS25 | LVCMOS, VCCIO = 2.5V | 0.00 | ns |
| LVCMOS18 | LVCMOS, VCCIO = 1.8V | 0.11 | ns |
| LVCMOS15 | LVCMOS, VCCIO = 1.5V | 0.26 | ns |
| LVCMOS12 | LVCMOS, VCCIO = 1.2V | 0.09 | ns |
| PCI33 | PCI, VCCIO = 3.0V | 0.05 | ns |
| **Output Adjusters** | | | |
| LVDS25E | LVDS, Emulated, VCCIO = 2.5V | 0.16 | ns |
| LVDS25 | LVDS, VCCIO = 2.5V | 0.01 | ns |
| BLVDS25 | BLVDS, Emulated, VCCIO = 2.5V | -0.04 | ns |

## SERDES High Speed Data Receiver

***Table 3-11. Serial Input Data Specifications***

| Symbol | Description | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $RX\text{-}CID_S$ | Stream of nontransitions[1] (CID = Consecutive Identical Digits) @ $10^{-12}$ BER | 3.125G | — | — | 136 | Bits |
| | | 2.5G | — | — | 144 | |
| | | 1.485G | — | — | 160 | |
| | | 622M | — | — | 204 | |
| | | 270M | — | — | 228 | |
| | | 150M | — | — | 296 | |
| $V_{RX\text{-}DIFF\text{-}S}$ | Differential input sensitivity | | 150 | — | 1760 | mV, p-p |
| $V_{RX\text{-}IN}$ | Input levels | | 0 | — | $V_{CCA}$ +0.5[4] | V |
| $V_{RX\text{-}CM\text{-}DC}$ | Input common mode range (DC coupled) | | 0.6 | — | $V_{CCA}$ | V |
| $V_{RX\text{-}CM\text{-}AC}$ | Input common mode range (AC coupled)[3] | | 0.1 | — | $V_{CCA}$ +0.2 | V |
| $T_{RX\text{-}RELOCK}$ | SCDR re-lock time[2] | | — | 1000 | — | Bits |
| $Z_{RX\text{-}TERM}$ | Input termination 50/75 Ohm/High Z | | -20% | 50/75/HiZ | +20% | Ohms |
| $RL_{RX\text{-}RL}$ | Return loss (without package) | | 10 | — | — | dB |

1. This is the number of bits allowed without a transition on the incoming data stream when using DC coupling.
2. This is the typical number of bit times to re-lock to a new phase or frequency within +/- 300 ppm, assuming 8b10b encoded data.
3. AC coupling is used to interface to LVPECL and LVDS. LVDS interfaces are found in laser drivers and Fibre Channel equipment. LVDS interfaces are generally found in 622 Mbps SERDES devices.
4. Up to 1.76V.

### Input Data Jitter Tolerance

A receiver's ability to tolerate incoming signal jitter is very dependent on jitter type. High speed serial interface standards have recognized the dependency on jitter type and have specifications to indicate tolerance levels for different jitter types as they relate to specific protocols. Sinusoidal jitter is considered to be a worst case jitter type.

***Table 3-12. Receiver Total Jitter Tolerance Specification***

| Description | Frequency | Condition | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Deterministic | 3.125 Gbps | 600 mV differential eye | — | — | 0.47 | UI, p-p |
| Random | | 600 mV differential eye | — | — | 0.18 | UI, p-p |
| Total | | 600 mV differential eye | — | — | 0.65 | UI, p-p |
| Deterministic | 2.5 Gbps | 600 mV differential eye | — | — | 0.47 | UI, p-p |
| Random | | 600 mV differential eye | — | — | 0.18 | UI, p-p |
| Total | | 600 mV differential eye | — | — | 0.65 | UI, p-p |
| Deterministic | 1.25 Gbps | 600 mV differential eye | — | — | 0.47 | UI, p-p |
| Random | | 600 mV differential eye | — | — | 0.18 | UI, p-p |
| Total | | 600 mV differential eye | — | — | 0.65 | UI, p-p |
| Deterministic | 622 Mbps | 600 mV differential eye | — | — | 0.47 | UI, p-p |
| Random | | 600 mV differential eye | — | — | 0.18 | UI, p-p |
| Total | | 600 mV differential eye | — | — | 0.65 | UI, p-p |

Note: Values are measured with CJPAT, all channels operating, FPGA Logic active, I/Os around SERDES pins quiet, voltages are nominal, room temperature.

# PCI Express Electrical and Timing Characteristics

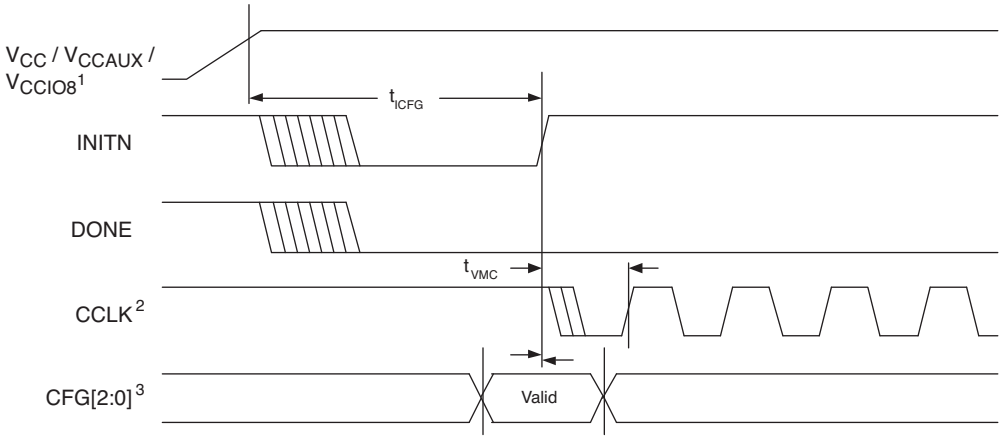
## AC and DC Characteristics

**Over Recommended Operating Conditions**

| Symbol | Description | Test Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| **Transmit**[1] | | | | | | |
| UI | Unit interval | | 399.88 | 400 | 400.12 | ps |
| $V_{TX-DIFF\_P-P}$ | Differential peak-to-peak output voltage | | 0.8 | 1.0 | 1.2 | V |
| $V_{TX-DE-RATIO}$ | De-emphasis differential output voltage ratio | | -3 | -3.5 | -4 | dB |
| $V_{TX-CM-AC\_P}$ | RMS AC peak common-mode output voltage | | — | — | 20 | mV |
| $V_{TX-RCV-DETECT}$ | Amount of voltage change allowed during receiver detection | | — | — | 600 | mV |
| $V_{TX-DC-CM}$ | Tx DC common mode voltage | | 0 | — | $V_{CCOB}$ + 5% | V |
| $I_{TX-SHORT}$ | Output short circuit current | $V_{TX-D+}$=0.0V $V_{TX-D-}$=0.0V | — | — | 90 | mA |
| $Z_{TX-DIFF-DC}$ | Differential output impedance | | 80 | 100 | 120 | Ohms |
| $RL_{TX-DIFF}$ | Differential return loss | | 10 | — | — | dB |
| $RL_{TX-CM}$ | Common mode return loss | | 6.0 | — | — | dB |
| $T_{TX-RISE}$ | Tx output rise time | 20 to 80% | 0.125 | — | — | UI |
| $T_{TX-FALL}$ | Tx output fall time | 20 to 80% | 0.125 | — | — | UI |
| $L_{TX-SKEW}$ | Lane-to-lane static output skew for all lanes in port/link | | — | — | 1.3 | ns |
| $T_{TX-EYE}$ | Transmitter eye width | | 0.75 | — | — | UI |
| $T_{TX-EYE-MEDIAN-TO-MAX-JITTER}$ | Maximum time between jitter median and maximum deviation from median | | — | — | 0.125 | UI |
| **Receive**[1, 2] | | | | | | |
| UI | Unit Interval | | 399.88 | 400 | 400.12 | ps |
| $V_{RX-DIFF\_P-P}$ | Differential peak-to-peak input voltage | | 0.34[3] | — | 1.2 | V |
| $V_{RX-IDLE-DET-DIFF\_P-P}$ | Idle detect threshold voltage | | 65 | — | 340[3] | mV |
| $V_{RX-CM-AC\_P}$ | Receiver common mode voltage for AC coupling | | — | — | 150 | mV |
| $Z_{RX-DIFF-DC}$ | DC differential input impedance | | 80 | 100 | 120 | Ohms |
| $Z_{RX-DC}$ | DC input impedance | | 40 | 50 | 60 | Ohms |
| $Z_{RX-HIGH-IMP-DC}$ | Power-down DC input impedance | | 200K | — | — | Ohms |
| $RL_{RX-DIFF}$ | Differential return loss | | 10 | — | — | dB |
| $RL_{RX-CM}$ | Common mode return loss | | 6.0 | — | — | dB |
| $T_{RX-IDLE-DET-DIFF-ENTERTIME}$ | Maximum time required for receiver to recognize and signal an unexpected idle on link | | — | — | — | ms |

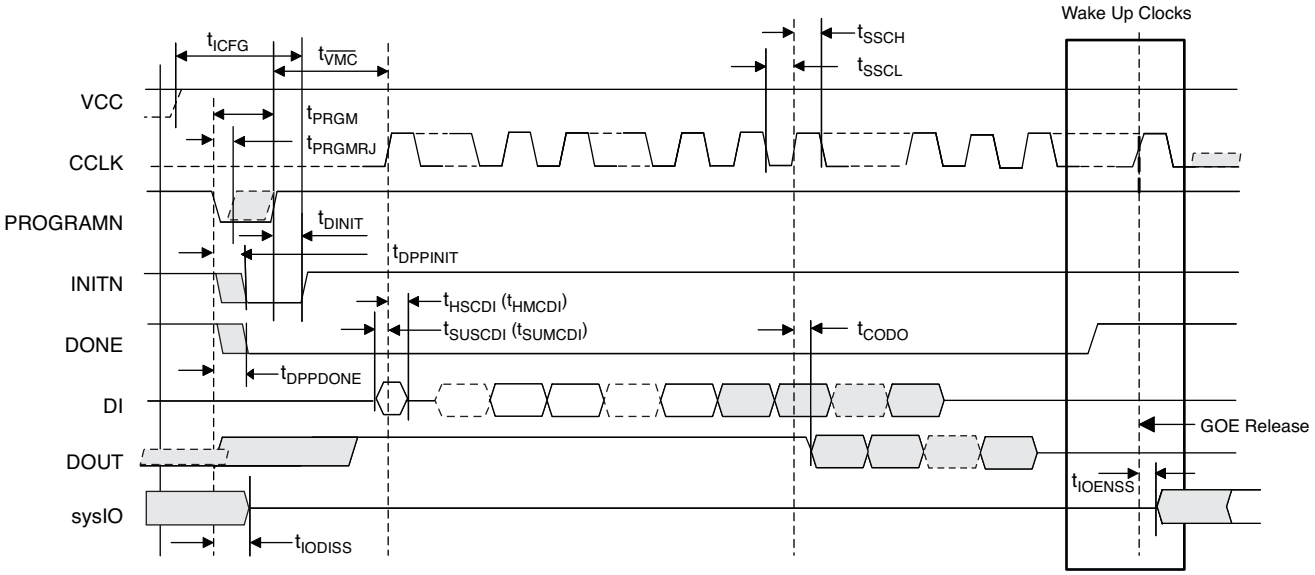
1. Values are measured at 2.5 Gbps.
2. Measured with external AC-coupling on the receiver.
3. Not in compliance with PCI Express 1.1 standard.

*Figure 3-24. Power-On-Reset (POR) Timing*




1. Time taken from $V_{CC}$, $V_{CCAUX}$ or $V_{CCIO8}$, whichever is the last to cross the POR trip point.
2. Device is in a Master Mode (SPI, SPIm).
3. The CFG pins are normally static (hard wired).

*Figure 3-25. sysCONFIG Port Timing*



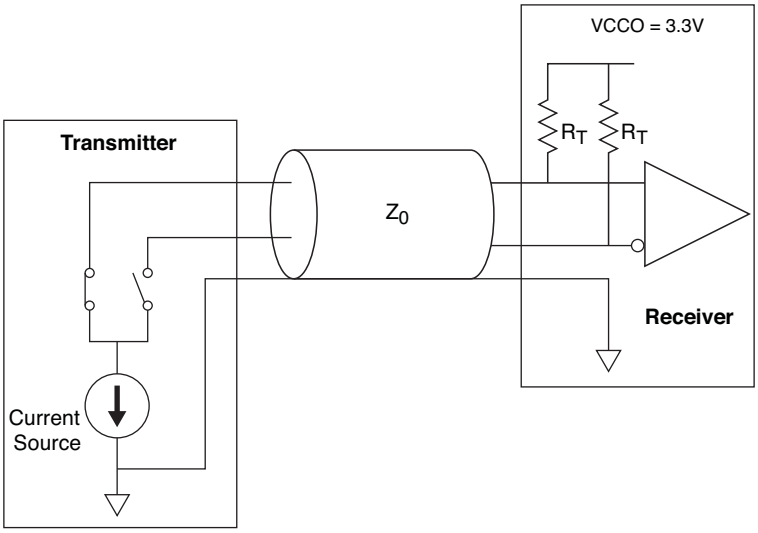
# sysI/O Differential Electrical Characteristics

## Transition Reduced LVDS (TRLVDS DC Specification)

**Over Recommended Operating Conditions**

| Symbol | Description | Min. | Nom. | Max. | Units |
|---|---|---|---|---|---|
| $V_{CCO}$ | Driver supply voltage (+/- 5%) | 3.14 | 3.3 | 3.47 | V |
| $V_{ID}$ | Input differential voltage | 150 | — | 1200 | mV |
| $V_{ICM}$ | Input common mode voltage | 3 | — | 3.265 | V |
| $V_{CCO}$ | Termination supply voltage | 3.14 | 3.3 | 3.47 | V |
| $R_T$ | Termination resistance (off-chip) | 45 | 50 | 55 | Ohms |

Note: LA-LatticeECP3 only supports the TRLVDS receiver.




## Mini LVDS

**Over Recommended Operating Conditions**

| Parameter Symbol | Description | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $Z_O$ | Single-ended PCB trace impedance | 30 | 50 | 75 | ohms |
| $R_T$ | Differential termination resistance | 50 | 100 | 150 | ohms |
| $V_{OD}$ | Output voltage, differential, $\|V_{OP} - V_{OM}\|$ | 300 | — | 600 | mV |
| $V_{OS}$ | Output voltage, common mode, $\|V_{OP} + V_{OM}\|/2$ | 1 | 1.2 | 1.4 | V |
| $\Delta V_{OD}$ | Change in $V_{OD}$, between H and L | — | — | 50 | mV |
| $\Delta V_{ID}$ | Change in $V_{OS}$, between H and L | — | — | 50 | mV |
| $V_{THD}$ | Input voltage, differential, $\|V_{INP} - V_{INM}\|$ | 200 | — | 600 | mV |
| $V_{CM}$ | Input voltage, common mode, $\|V_{INP} + V_{INM}\|/2$ | $0.3+(V_{THD}/2)$ | — | $2.1-(V_{THD}/2)$ | |
| $T_R$, $T_F$ | Output rise and fall times, 20% to 80% | — | — | 550 | ps |
| $T_{ODUTY}$ | Output clock duty cycle | 40 | — | 60 | % |

Note: Data is for 6mA differential current drive. Other differential driver current options are available.

## Signal Descriptions

| Signal Name | I/O | Description |
|---|---|---|
| **General Purpose** | | |
| P[Edge] [Row/Column Number]_[A/B] | I/O | [Edge] indicates the edge of the device on which the pad is located. Valid edge designations are L (Left), B (Bottom), R (Right), T (Top).<br><br>[Row/Column Number] indicates the PFU row or the column of the device on which the PIC exists. When Edge is T (Top) or B (Bottom), only need to specify Column Number. When Edge is L (Left) or R (Right), only need to specify Row Number.<br><br>[A/B] indicates the PIO within the PIC to which the pad is connected. Some of these user-programmable pins are shared with special function pins. These pins, when not used as special purpose pins, can be programmed as I/Os for user logic. During configuration the user-programmable I/Os are tri-stated with an internal pull-up resistor enabled. If any pin is not used (or not bonded to a package pin), it is also tri-stated with an internal pull-up resistor enabled after configuration. |
| P[Edge][Row Number]E_[A/B/C/D] | I | These general purpose signals are input-only pins and are located near the PLLs. |
| GSRN | I | Global RESET signal (active low). Any I/O pin can be GSRN. |
| NC | — | No connect. |
| RESERVED | — | This pin is reserved and should not be connected to anything on the board. |
| GND | — | Ground. Dedicated pins. |
| $V_{CC}$ | — | Power supply pins for core logic. Dedicated pins. |
| $V_{CCAUX}$ | — | Auxiliary power supply pin. This dedicated pin powers all the differential and referenced input buffers. |
| $V_{CCIOx}$ | — | Dedicated power supply pins for I/O bank x. |
| $V_{CCA}$ | — | SERDES, transmit, receive, PLL and reference clock buffer power supply. All $V_{CCA}$ supply pins must always be powered to the recommended operating voltage range. If no SERDES channels are used, connect $V_{CCA}$ to $V_{CC}$. |
| $V_{CCPLL\_[LOC]}$ | — | General purpose PLL supply pins where LOC=L (left) or R (right). |
| $V_{REF1\_x}$, $V_{REF2\_x}$ | — | Reference supply pins for I/O bank x. Pre-determined pins in each bank are assigned as $V_{REF}$ inputs. When not used, they may be used as I/O pins. |
| VTTx | — | Power supply for on-chip termination of I/Os. |
| XRES[1] | — | 10K ohm +/-1% resistor must be connected between this pad and ground. |
| **PLL, DLL and Clock Functions** | | |
| [LOC][num]_GPLL[T, C]_IN_[index] | I | General Purpose PLL (GPLL) input pads: LUM, LLM, RUM, RLM, num = row from center, T = true and C = complement, index A,B,C...at each side. |
| [LOC][num]_GPLL[T, C]_FB_[index] | I | Optional feedback GPLL input pads: LUM, LLM, RUM, RLM, num = row from center, T = true and C = complement, index A,B,C...at each side. |
| [LOC]0_GDLLT_IN_[index][2] | I/O | General Purpose DLL (GDLL) input pads where LOC=RUM or LUM, T is True Complement, index is A or B. |
| [LOC]0_GDLLT_FB_[index][2] | I/O | Optional feedback GDLL input pads where LOC=RUM or LUM, T is True Complement, index is A or B. |
| PCLK[T, C][n:0]_[3:0][2] | I/O | Primary Clock pads, T = true and C = complement, n per side, indexed by bank and 0, 1, 2, 3 within bank. |

## PICs and DDR Data (DQ) Pins Associated with the DDR Strobe (DQS) Pin

| PICs Associated with DQS Strobe | PIO Within PIC | DDR Strobe (DQS) and Data (DQ) Pins |
|---|---|---|
| **For Left and Right Edges of the Device** | | |
| P[Edge] [n-3] | A | DQ |
| | B | DQ |
| P[Edge] [n-2] | A | DQ |
| | B | DQ |
| P[Edge] [n-1] | A | DQ |
| | B | DQ |
| P[Edge] [n] | A | [Edge]DQSn |
| | B | DQ |
| P[Edge] [n+1] | A | DQ |
| | B | DQ |
| P[Edge] [n+2] | A | DQ |
| | B | DQ |
| **For Top Edge of the Device** | | |
| P[Edge] [n-3] | A | DQ |
| | B | DQ |
| P[Edge] [n-2] | A | DQ |
| | B | DQ |
| P[Edge] [n-1] | A | DQ |
| | B | DQ |
| P[Edge] [n] | A | [Edge]DQSn |
| | B | DQ |
| P[Edge] [n+1] | A | DQ |
| | B | DQ |
| P[Edge] [n+2] | A | DQ |
| | B | DQ |

Note: “n” is a row PIC number.