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	-
Number of Logic Elements/Cells	2592
Total RAM Bits	43008
Number of I/O	223
Number of Gates	80000
Voltage - Supply	3V ~ 3.6V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 70°C (TA)
Package / Case	256-BGA
Supplier Device Package	256-FPBGA (17x17)
Purchase URL	https://www.e-xfl.com/product-detail/lattice-semiconductor/or3t557ba256-db

Product Line	Ordering Part Number	Product Status	Reference PCN
OR3T55 (Cont'd)	OR3T557BA256-DB	Active / Orderable	
	OR3T556BA256-DB		
	OR3T556BA256I-DB		
	OR3T557BA352-DB	**Discontinued**	PCN#09-10
	OR3T556BA352-DB		
	OR3T556BA352I-DB		
OR3T80	OR3T807S208-DB	**Discontinued**	PCN#09-10
	OR3T806S208-DB		
	OR3T806S208I-DB		
	OR3T807PS240-DB	**Discontinued**	PCN#06-07
	OR3T806PS240-DB		
	OR3T806PS240I-DB		
	OR3T807BA352-DB	**Discontinued**	PCN#09-10
	OR3T806BA352-DB		
	OR3T806BA352I-DB		
	OR3T807BC432-DB		
	OR3T806BC432-DB		
	OR3T806BC432I-DB		
OR3T125	OR3T1257PS208-DB	**Discontinued**	PCN#06-07
	OR3T1256PS208-DB		
	OR3T1256PS208I-DB		
	OR3T1257PS240-DB		
	OR3T1256PS240-DB		
	OR3T1256PS240I-DB		
	OR3T1257BA352-DB		PCN#09-10
	OR3T1256BA352-DB		
	OR3T1256BA352I-DB		
	OR3T1257BC432-DB		
	OR3T1256BC432-DB		
	OR3T1256BC432I-DB		

5555 N.E. Moore Ct. ● Hillsboro, Oregon 97124-6421 ● Phone (503) 268-8000 ● FAX (503) 268-8347
Internet: http://www.latticesemi.com

Programmable Logic Cells (continued)

Table 4. Control Input Functionality

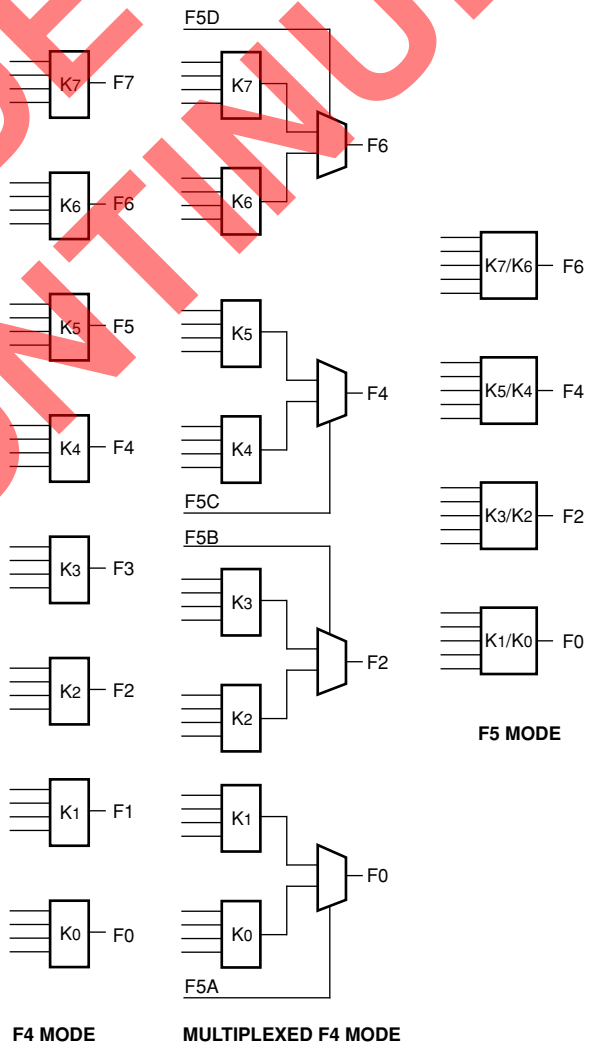
Mode	CLK	LSR	CE	ASWE	SEL
Logic	CLK to all latches/ FFs	LSR to all latches/ FFs, enabled per nibble and for ninth FF	CE to all latches/FFs, selectable per nibble and for ninth FF	CE to all latches/FFs, selectable per nibble and for ninth FF	Select between LUT input and direct input for eight latches/FFs
Half Logic/ Half Ripple	CLK to all latches/ FFs	LSR to all latches/FF, enabled per nibble and for ninth FF	CE to all latches/FFs, selectable per nibble and for ninth FF	Ripple logic control input	Select between LUT input and direct input for eight latches/FFs
Ripple	CLK to all latches/ FFs	LSR to all latches/ FFs, enabled per nibble and for ninth FF	CE to all latches/FFs, selectable per nibble and for ninth FF	Ripple logic control input	Select between LUT input and direct input for eight latches/FFs
Memory (RAM)	CLK to RAM	Port enable 2	Port enable 1	Write enable	Not used
Memory (ROM)	Optional for sync. outputs	Not used	Not used	Not used	Not used

Logic Mode

The PFU diagram of Figure 3 represents the logic mode of operation. In logic mode, the eight LUTs are used individually or in flexible groups to implement user logic functions. The latches/FFs may be used in conjunction with the LUTs or separately with the direct PFU data inputs. There are three basic submodes of LUT operation in PFU logic mode: F4 mode, F5 mode, and softwired LUT (SWL) mode. Combinations of these submodes are possible in each PFU.

F4 mode, shown simplified in Figure 4, illustrates the uses of the basic 4-input LUTs in the PFU. The output of an F4 LUT can be passed out of the PFU, captured at the LUTs associated latch/FF, or multiplexed with the adjacent F4 LUT output using one of the F5[A:D] inputs to the PFU. Only adjacent LUT pairs (K_0 and K_1, K_2 and K_3, K_4 and K_5, K_6 and K_7) can be multiplexed, and the output always goes to the even-numbered output of the pair.

The F5 submode of the LUT operation, shown simplified in Figure 4, indicates the use of 5-input LUTs to implement logic. 5-input LUTs are created from two 4-input LUTs and a multiplexer. The F5 LUT is the same as the multiplexing of two F4 LUTs described previously with the constraint that the inputs to the F4 LUTs be the same. The F5[A:D] input is then used as the fifth LUT input. The equations for the two F4 LUTs will differ by the assumed value for the F5[A:D] input, one F4 LUT assuming that the F5[A:D] input is zero, and the other assuming it is a one. The selection of the appropriate F4 LUT output in the F5 MUX by the F5[A:D] signal creates a 5-input LUT. Any combination of F4 and F5 LUTs is allowed per PFU using the eight 16-bit LUTs. Examples are eight F4 LUTs, four F5 LUTs, and a combination of four F4 plus two F5 LUTs.



5-5970(F)

Figure 4. Simplified F4 and F5 Logic Modes

Programmable Logic Cells (continued)

Softwired LUT submode uses F4 and F5 LUTs and internal PFU feedback routing to generate complex logic functions up to three LUT-levels deep. Figure 3 shows multiplexers between the Kz[3:0] inputs to the PFU and the LUTs. These multiplexers can be independently configured to route certain LUT outputs to the input of other LUTs. In this manner, very complex logic functions, some of up to 21 inputs, can be implemented in a single PFU at greatly enhanced speeds.

Figure 5 shows several softwired LUT topologies. In this figure, each circle represents either an F4 or F5 LUT. It is important to note that an LUT output that is fed back for softwired use is still available to be registered or output from the PFU. This means, for instance, that a logic equation that is needed by itself and as a term in a larger equation need only be generated once and PLC routing resources will not be required to use it in the larger equation.

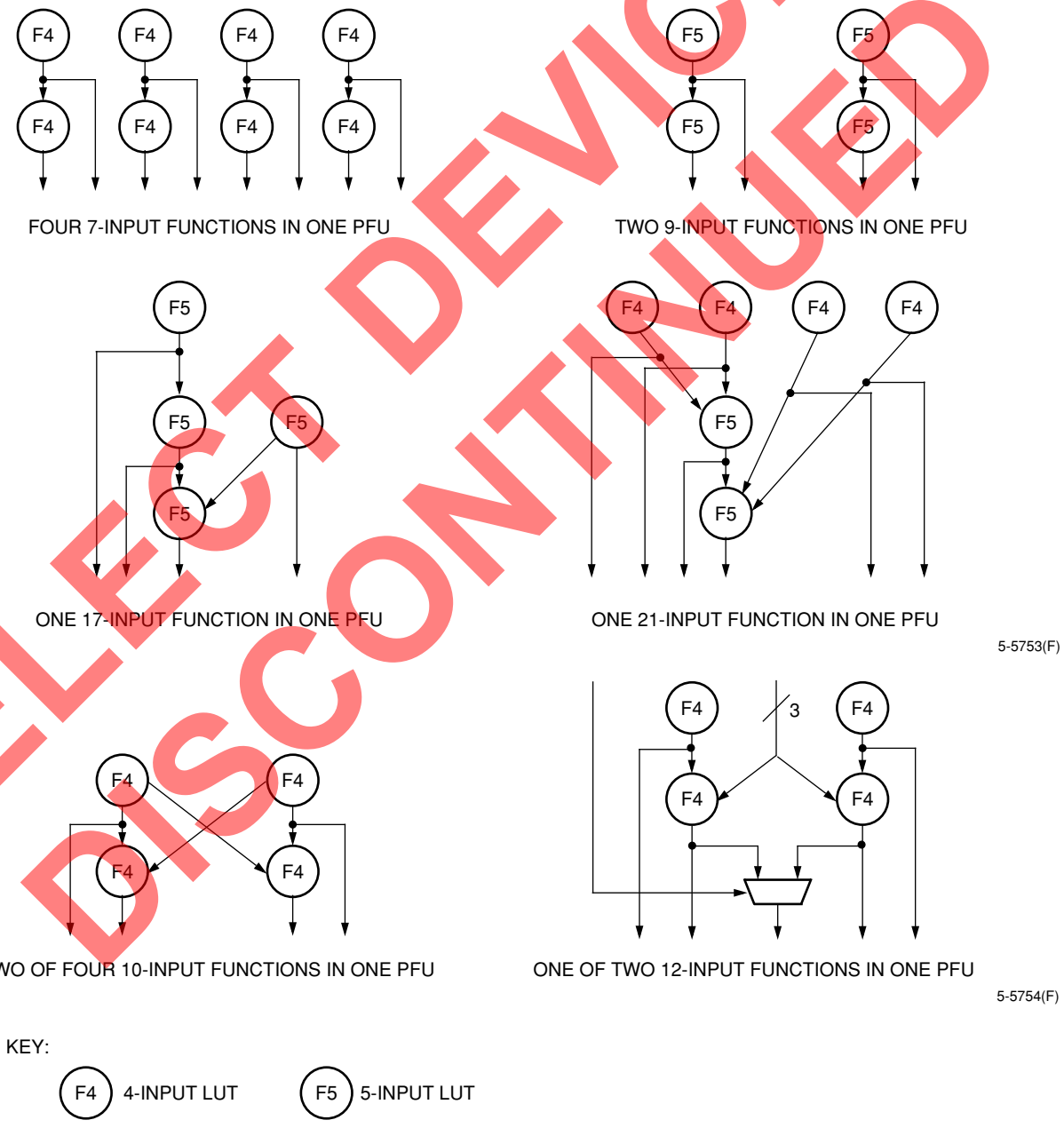


Figure 5. Softwired LUT Topology Examples

Programmable Logic Cells (continued)

Data is written to the write data, write address, and write enable registers on the active edge of the clock, but data is not written into the RAM until the next clock edge one-half cycle later. The read port is actually asynchronous, providing the user with read data very quickly after setting the read address, but timing is also provided so that the read port may be treated as fully synchronous for write then read applications. If the read and write address lines are tied together (maintaining MSB to MSB, etc.), then the dual-port RAM operates as a synchronous single-port RAM. If the write enable is disabled, and an initial memory contents is provided at configuration time, the memory acts as a ROM (the write data and write address ports and write port enables are not used).

Wider memories can be created by operating two or more memory mode PFUs in parallel, all with the same address and control signals, but each with a different nibble of data. To increase memory word depth above 32, two or more PLCs can be used. Figure 10 shows a 128 x 8 dual-port RAM that is implemented in eight PLCs. This figure demonstrates data path width expansion by placing two memories in parallel to achieve an 8-bit data path. Depth expansion is applied to achieve 128 words deep using the 32-word deep PFU memories. In addition to the PFU in each PLC, the SLIC (described in the next section) in each PLC is used for read address decodes and 3-state drivers. The 128 x 8 RAM shown could be made to operate as a single-port RAM by tying (bit-for-bit) the read and write addresses.

To achieve depth expansion, one or two of the write address bits (generally the MSBs) are routed to the write port enables as in Figure 10. For 2 bits, the bits select which 32-word bank of RAM of the four available from a decode of two WPE inputs is to be written. Similarly, 2 bits of the read address are decoded in the SLIC and are used to control the 3-state buffers through which the read data passes. The write data bus is common, with separate nibbles for width expansion, across all PLCs, and the read data bus is common (again, with separate nibbles) to all PLCs at the output of the 3-state buffers.

Figure 10 also shows a new optional capability to provide a read enable for RAMs/ROMs in Series 3 using the SLIC cell. The read enable will 3-state the read data bus when inactive, allowing the write data and read data buses to be tied together if desired.

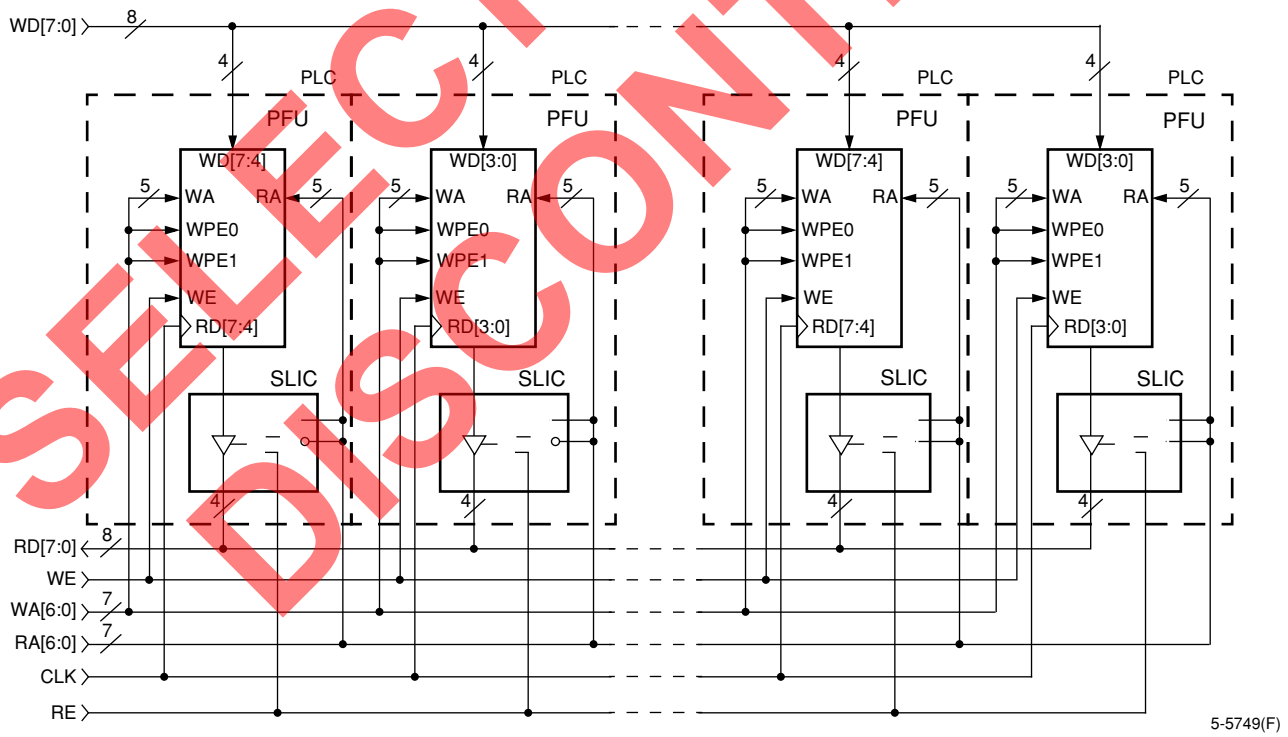


Figure 10. Memory Mode Expansion Example—128 x 8 RAM

Programmable Logic Cells (continued)

PLC Routing Resources

Generally, the ispLEVER Development System is used to automatically route interconnections. Interactive routing with the ispLEVER design editor (EPIC) is also available for design optimization. To use EPIC for interactive layout, an understanding of the routing resources is needed and is provided in this section.

The routing resources consist of switching circuitry and metal interconnect segments. Generally, the metal lines which carry the signals are designated as routing segments. The switching circuitry connects the routing segments, providing one or more of three basic functions: signal switching, amplification, and isolation. A net running from a PFU or PIC output (source) to a PLC or PIC input (destination) consists of one or more routing segments, connected by switching circuitry called configurable interconnect points (CIPs).

The following sections discuss PLC, PIC, and interquad routing resources. This section discusses the PLC switching circuitry, intra-PLC routing, inter-PLC routing, and clock distribution.

Configurable Interconnect Points

The process of connecting routing segments uses three basic types of switching circuits: two types of configurable interconnect points (CIPs) and bidirectional buffers (BIDIs). The basic element in CIPs is one or more pass transistors, each controlled by a configuration RAM bit. The two types of CIPs are the mutually exclusive (or multiplexed) CIP and the independent CIP.

A mutually exclusive set of CIPs contains two or more CIPs, only one of which can be on at a time. An independent CIP has no such restrictions and can be on independent of the state of other CIPs. Figure 18 shows an example of both types of CIPs.

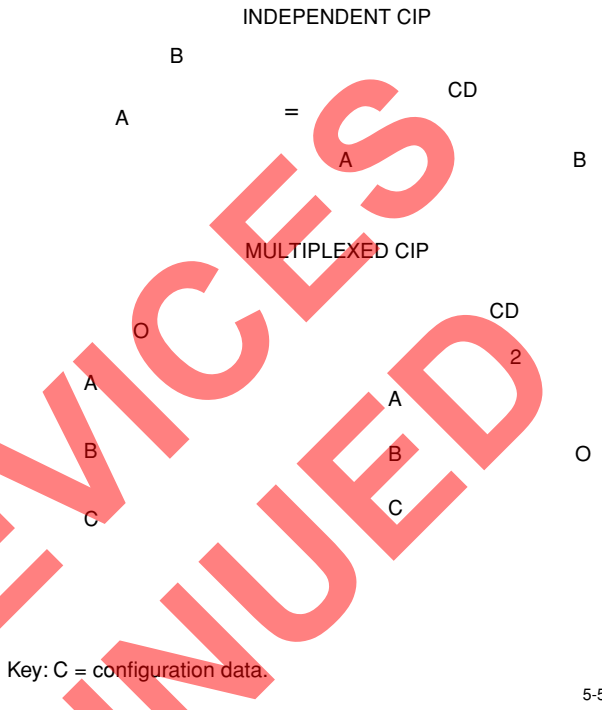


Figure 18. Configurable Interconnect Point

3-Statable Bidirectional Buffers

Bidirectional buffers, previously described in the SLIC section of the programmable logic cell discussion, provide isolation as well as amplification for signals routed a long distance. Bidirectional buffers are also used to route signals diagonally in the PLC (described later in the subsection entitled Intra-PLC Routing), and BIDIs can be used to indirectly route signals through the switching routing (xSW) segments. Any number from zero to ten BIDIs can be used in a given PLC.

Programmable Clock Manager (PCM)

The *ORCA* programmable clock manager (PCM) is a special function block that is used to modify or condition clock signals for optimum system performance. Some of the functions that can be performed with the PCM are clock skew reduction (both internal and board level), duty-cycle adjustment, clock delay reduction, clock phase adjustment, and clock frequency multiplication/division. Due to the different capabilities required by customer application, each PCM contains both a PLL (phase-locked loop) and a DLL (delayed-locked loop) mode. By using PLC logic resources in conjunction with the PCM, many other functions, such as frequency synthesis, are possible.

There are two PCMs on each Series 3 device, one in the lower left corner and one in the upper right corner. Each can drive two different, but interrelated clock networks inside the FPGA. Each PCM can take a clock input from the ExpressCLK pad in its corner or from general routing resources. There are also two input sources that provide feedback to the PCM from the PLC array. One of these is a dedicated corner ExpressCLK feedback, and the other is from general routing. Each PCM sources two clock outputs, one to the corner ExpressCLK that feeds the CLKCNTRL blocks on the two sides adjacent to the PCM, and one to the system clock spine network through general routing. Figure 45 shows a high-level block diagram of the PCM.

Functionality of the PCM is programmed during operation through a read/write interface internal to the FPGA array or via the configuration bit stream. The internal FPGA interface comprises write enable and read enable signals, a 3-bit address bus, an 8-bit input (to the PCM) data bus, and an 8-bit output data bus. There is also a PCM output signal, LOCK, that indicates a stable output clock state. These signals are used to program a series of registers to configure the PCM functional core for the desired functionality.

Operation of the PCM is divided into two modes, delay-locked loop (DLL) and phase-locked loop (PLL). Some operations can be performed by either mode and some are specific to a particular mode. These will be described in each individual mode section. In general, DLL mode is preferable to PLL mode for the same function because it is less sensitive to input clock noise.

In the discussions that follow, the duty cycle is the percent of the clock period during which the output clock is high.

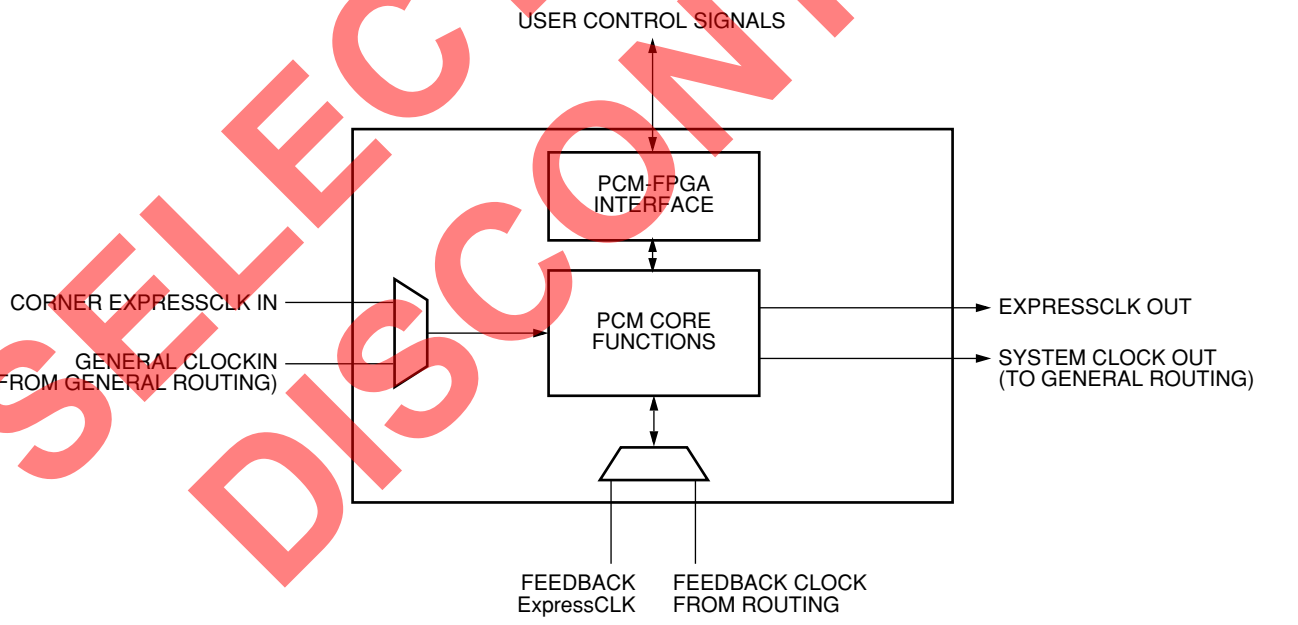


Figure 45. PCM Block Diagram

Programmable Clock Manager (PCM)
(continued)

PCM Applications

The applications discussed below are only a small sampling of the possible uses for the PCM. Check the Lattice website for additional application notes.

Clock Phase Adjustment

The PCM may be used to adjust the phase of the input clock. The result is an output clock which has its active edge either preceding or following the active edge of the input clock. Clock phase adjustment is accomplished in DLL mode by delaying the clock. This is discussed in the Delay-Locked Loop (DLL) Mode section. Examples of using the delayed clock as an early or late phase-adjusted clock are outlined in the following paragraphs.

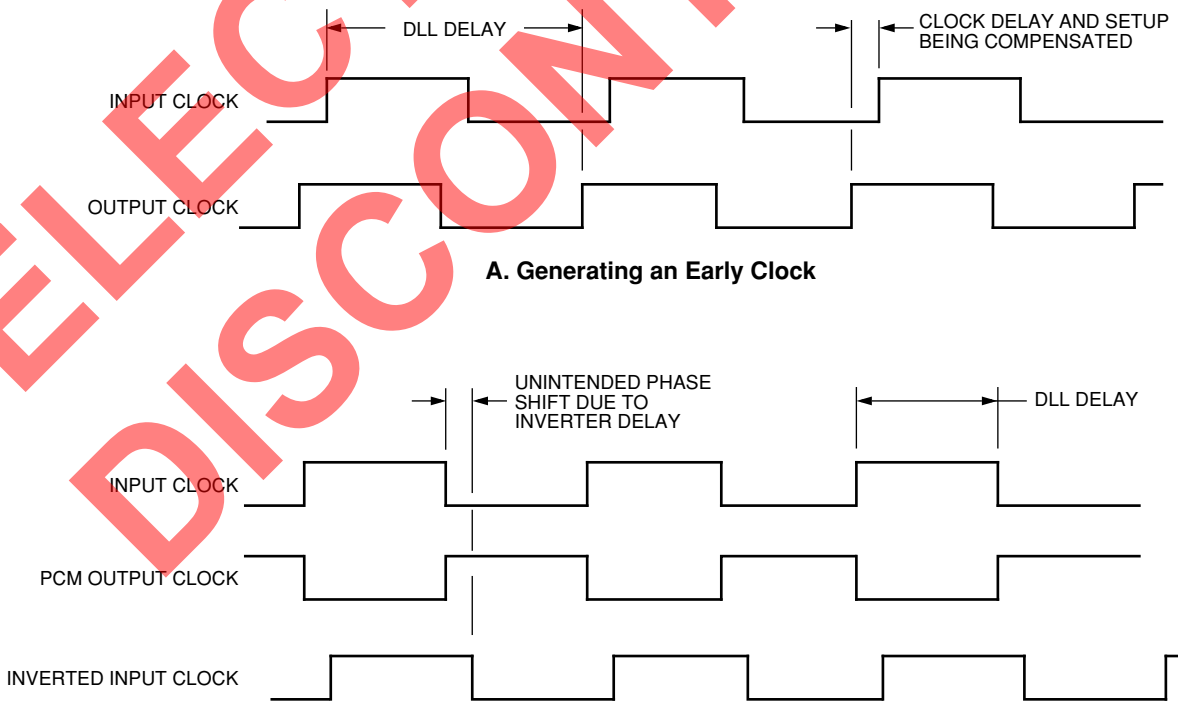
An output clock that precedes the input clock can be used to compensate for clock delay that is largely due to excessive loading. The preceding output clock is really not early relative to the input clock, but is delayed almost a full cycle. This is shown in Figure 48A. The amount of delay that is being compensated for, plus clock setup time and some margin, is the amount **less** than one full clock cycle that the output clock is delayed from the input clock.

In some systems, it is desirable to operate logic from several clocks that operate at different phases. This technique is often used in microprocessor-based systems to transfer and process data synchronously between functional areas, but without incurring excessive delays. Figure 48B shows an input clock and an output clock operating 180° out of phase. It also shows a version of the input clock that was shifted approximately 180° using logic gates to create an inverter. Note that the inverted clock is really shifted more than 180° due to the propagation delay of the inverter. The PCM output clock does not suffer from this delay. Additionally, the 180° shifted PCM output could be shifted by some smaller amount to effect an early 180° shifted clock that also accounts for loading effects.

In terms of degrees of phase shift, the phase of a clock is adjustable in DLL mode with resolution relative to the delay increment (see Table 27):

Phase Adjustment = (Delay)* 11.25, Delay < 16

Phase Adjustment = ((Delay)* 11.25) – 360, Delay > 16



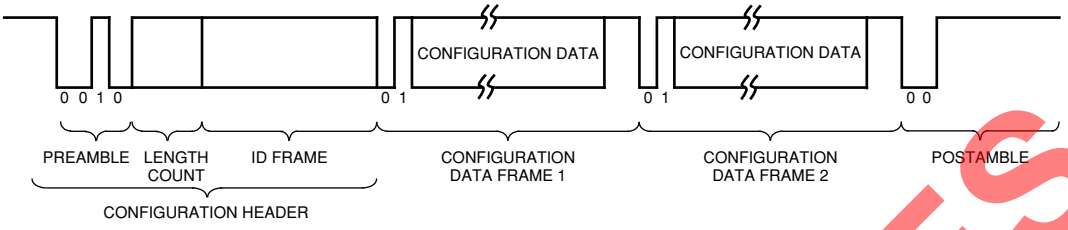
A. Generating an Early Clock

B. Multiphase Clock Generation Using the DLL

5-5979(F)

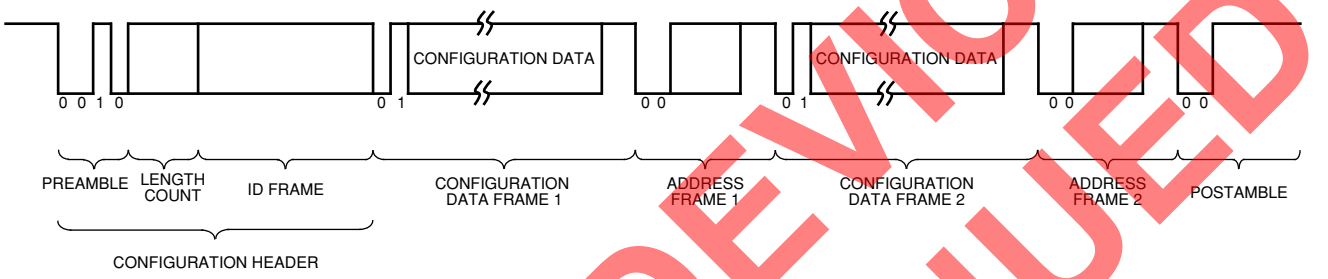
Figure 48. Clock Phase Adjustment Using the PCM

Configuration Data Format (continued)



5-5759(F)

Figure 52. Serial Configuration Data Format—Autoincrement Mode



5-5760(F)

Figure 53. Serial Configuration Data Format—Explicit Mode

Table 32. Configuration Frame Format and Contents

Header	11110010	Preamble
	24-bit Length Count	Configuration frame length.
	11111111	Trailing header—8 bits.
ID Frame	0101 1111 1111 1111	ID frame header.
	Configuration Mode	00 = autoincrement, 01 = explicit.
	Reserved [41:0]	Reserved bits set to 0.
	ID	20-bit part ID.
	Checksum	8-bit checksum.
	11111111	Eight stop bits (high) to separate frames.
Configuration Data Frame (repeated for each data frame)	01	Data frame header.
	Data Bits	Number of data bits depends upon device.
	Alignment Bits = 0	String of 0 bits added to bit stream to make frame header, plus data bits reach a byte boundary.
	Checksum	8-bit checksum.
	11111111	Eight stop bits (high) to separate frames.
Configuration Address Frame	00	Address frame header.
	14 Address Bits	14-bit address of location to start data storage.
	Checksum	8-bit checksum.
	11111111	Eight stop bits (high) to separate frames.
Postamble	00	Postamble header.
	11111111 111111	Dummy address.
	1111111111111111	16 stop bits.*

* In MPI configuration mode, the number of stop bits = 32.

Note: For slave parallel mode, the byte containing the preamble must be 11110010. The number of leading header dummy bits must be (n * 8) + 4, where n is any nonnegative integer and the number of trailing dummy bits must be (n * 8), where n is any positive integer. The number of stop bits/frame for slave parallel mode must be (x * 8), where x is a positive integer. Note also that the bit stream generator tool supplies a bit stream that is compatible with all configuration modes, including slave parallel mode.

Absolute Maximum Ratings

Stresses in excess of the absolute maximum ratings can cause permanent damage to the device. These are absolute stress ratings only. Functional operation of the device is not implied at these or any other conditions in excess of those given in the operations sections of this data sheet. Exposure to absolute maximum ratings for extended periods can adversely affect device reliability.

The *ORCA* Series FPGAs include circuitry designed to protect the chips from damaging substrate injection currents and to prevent accumulations of static charge. Nevertheless, conventional precautions should be observed during storage, handling, and use to avoid exposure to excessive electrical stress.

Table 35. Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Unit
Storage Temperature	T_{stg}	–65	150	°C
Supply Voltage with Respect to Ground	V_{DD}	–0.5	7.0	V
Input Signal with Respect to Ground	—	–0.5	V_{DD} + 0.3	V
Signal Applied to High-impedance Output	—	–0.5	V_{DD} + 0.3	V
Maximum Package Body Temperature	—	—	220	°C

Recommended Operating Conditions

Table 36. Recommended Operating Conditions

Mode	OR3Cxx		OR3Txxx	
	Temperature Range (Ambient)	Supply Voltage (V_{DD})	Temperature Range (Ambient)	Supply Voltage (V_{DD})
Commercial	0 °C to 70 °C	5 V ± 5%	0 °C to 70 °C	3.0 V to 3.6 V
Industrial	–40 °C to +85 °C	5 V ± 10%	–40 °C to +85 °C	3.0 V to 3.6 V

Note: The maximum recommended junction temperature (T_J) during operation is 125 °C.

Timing Characteristics (continued)

Table 62. Master Parallel Configuration Mode Timing Characteristics

OR3Cxx Commercial: V_{DD} = 5.0 V ± 5%, 0 °C < T_A < 70 °C; Industrial: V_{DD} = 5.0 V ± 10%, –40 °C < T_A < +85 °C.
OR3Txxx Commercial: V_{DD} = 3.0 V to 3.6 V, 0 °C < T_A < 70 °C; Industrial: V_{DD} = 3.0 V to 3.6 V, –40 °C < T_A < +85 °C.

Parameter	Symbol	Min	Max	Unit
RCLK to Address Valid	T_{AV}	—	60.00	ns
D[7:0] Setup Time to RCLK High	T_S	60.00	—	ns
D[7:0] Hold Time to RCLK High	T_H	0.00	—	ns
RCLK Low Time (M3 = 0)	T_{CL}	7.00	7.00	CCLK cycles
RCLK High Time (M3 = 0)	T_{CH}	1.00	1.00	CCLK cycles
RCLK Low Time (M3 = 1)	T_{CL}	7.00	7.00	CCLK cycles
RCLK High Time (M3 = 1)	T_{CH}	1.00	1.00	CCLK cycles
CCLK to DOUT	T_D	—	5.00	ns

Notes:
The RCLK period consists of seven CCLKs for RCLK low and one CCLK for RCLK high.

Serial data is transmitted out on DOUT 1.5 CCLK cycles after the byte is input on D[7:0].

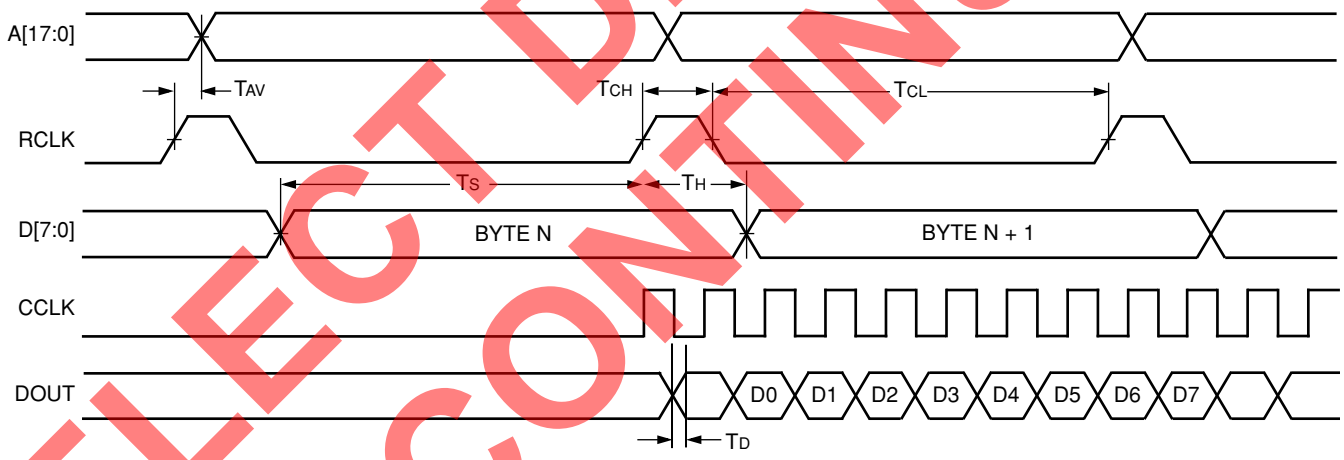


Figure 84. Master Parallel Configuration Mode Timing Diagram

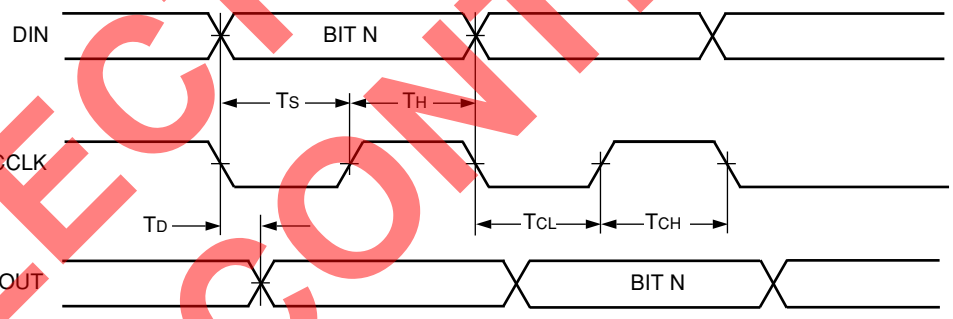
Timing Characteristics (continued)

Table 64. Slave Serial Configuration Mode Timing Characteristics

OR3Cxx Commercial: V_{DD} = 5.0 V ± 5%, 0 °C < T_A < 70 °C; Industrial: V_{DD} = 5.0 V ± 10%, –40 °C < T_A < +85 °C.
OR3Txxx Commercial: V_{DD} = 3.0 V to 3.6 V, 0 °C < T_A < 70 °C; Industrial: V_{DD} = 3.0 V to 3.6 V, –40 °C < T_A < +85 °C.

Parameter	Symbol	Min	Max	Unit
DIN Setup Time: 3Cxx 3Txxx	T_S	 20.00 10.50	 — —	 ns ns
DIN Hold Time	T_H	0.00	—	ns
CCLK High Time: 3Cxx 3Txxx	T_{CH}	 20.00 7.00	 — —	 ns ns
CCLK Low Time: 3Cxx 3Txxx	T_{CL}	 20.00 7.00	 — —	 ns ns
CCLK Frequency: 3Cxx 3Txxx	F_C	 — —	 25.00 66.00	 MHz MHz
CCLK to DOUT	T_D	—	20.00	ns

Note: Serial configuration data is transmitted out on DOUT on the falling edge of CCLK after it is input on DIN.



5-4535(F).

Figure 86. Slave Serial Configuration Mode Timing Diagram

Pin Information (continued)

Table 67. Pin Descriptions (continued)

Symbol	I/O	Description
Special-Purpose Pins (continued)		
M3	I	During powerup and initialization, M3 is used to select the speed of the internal oscillator during configuration with their values latched on the rising edge of $\overline{\text{INIT}}$. When M3 is low, the oscillator frequency is 10 MHz. When M3 is high, the oscillator is 1.25 MHz. During configuration, a pull-up is enabled.
	I/O	After configuration, this pin is a user-programmable I/O pin (see Note).
TDI, TCK, TMS	I	If boundary scan is used, these pins are test data in, test clock, and test mode select inputs. If boundary scan is not selected, all boundary-scan functions are inhibited once configuration is complete. Even if boundary scan is not used, either TCK or TMS must be held at logic 1 during configuration. Each pin has a pull-up enabled during configuration.
	I/O	After configuration, these pins are user-programmable I/O (see Note).
RDY/RCLK/ MPI_ALE	O	During configuration in peripheral mode, RDY/RCLK indicates another byte can be written to the FPGA. If a read operation is done when the device is selected, the same status is also available on D7 in asynchronous peripheral mode.
	O	During the master parallel configuration mode, RCLK is a read output signal to an external memory. This output is not normally used.
	I	In *i960* microprocessor mode, this pin acts as the address latch enable (ALE) input.
	I/O	After configuration, if the MPI is not used, this pin is a user-programmable I/O pin (see Note).
HDC	O	High During Configuration is output high until configuration is complete. It is used as a control output, indicating that configuration is not complete.
	I/O	After configuration, this pin is a user-programmable I/O pin (see Note).
$\overline{\text{LDC}}$	O	$\overline{\text{Low During Configuration}}$ is output low until configuration is complete. It is used as a control output, indicating that configuration is not complete.
	I/O	After configuration, this pin is a user-programmable I/O pin (see Note).
$\overline{\text{INIT}}$	I/O	$\overline{\text{INIT}}$ is a bidirectional signal before and during configuration. During configuration, a pull-up is enabled, but an external pull-up resistor is recommended. As an active-low open-drain output, $\overline{\text{INIT}}$ is held low during power stabilization and internal clearing of memory. As an active-low input, $\overline{\text{INIT}}$ holds the FPGA in the wait-state before the start of configuration.
	I/O	After configuration, this pin is a user-programmable I/O pin (see Note).

Note: The FPGA States of Operation section contains more information on how to control these signals during start-up. The timing of DONE release is controlled by one set of bit stream options, and the timing of the simultaneous release of all other configuration pins (and the activation of all user I/Os) is controlled by a second set of options.

Pin	OR3T20 Pad	Function
43	PB3B	I/O
44	PB3D	I/O
45	VDD	VDD
46	PB4A	I/O
47	PB4D	I/O
48	PB5A	I/O
49	PB5C	I/O
50	PB5D	I/O
51	PB6A	I/O
52	PB6C	I/O
53	PB6D	I/O
54	VSS	VSS
55	PECKB	I-ECKB
56	PB7C	I/O
57	PB7D	I/O
58	PB8A	I/O
59	PB8D	I/O
60	PB9A	I/O-HDC
61	PB9C	I/O
62	PB9D	I/O
63	VDD	VDD
64	PB10A	I/O-$\overline{LDC}$
65	PB10C	I/O
66	PB10D	I/O
67	PB11A	I/O-$\overline{INIT}$
68	PB11D	I/O
69	PB12A	I/O
70	VSS	VSS
71	PDONE	DONE
72	VDD	VDD
73	VSS	VSS
74	PRESETN	$\overline{RESET}$
75	PPRGMN	$\overline{PRGM}$
76	PR12A	I/O-M0
77	PR12D	I/O
78	PR11A	I/O
79	PR10A	I/O-M1
80	PR10C	I/O
81	PR10D	I/O
82	PR9A	I/O-M2
83	PR9B	I/O
84	PR9D	I/O
85	PR8A	I/O-M3

Pin	OR3T30 Pad	OR3T55 Pad	OR3C/T80 Pad	OR3T125 Pad	Function
42	PL10B	PL12B	PL15B	PL19D	I/O
43	PL10A	PL12A	PL15A	PL19A	I/O-A11/$\overline{\text{MPI\_IRQ}}$
44	VDD	VDD	VDD	VDD	VDD
45	PL11D	PL13D	PL16D	PL20D	I/O-A12
46	PL11C	PL13B	PL16B	PL20B	I/O
47	PL11B	PL14D	PL17D	PL21D	I/O
48	PL11A	PL14B	PL17B	PL21B	I/O-A13
49	PL12D	PL14A	PL17A	PL21A	I/O
50	PL12C	PL15D	PL18D	PL22D	I/O
51	PL12B	PL15B	PL18B	PL23D	I/O
52	PL12A	PL16D	PL19D	PL24A	I/O-A14
53	VSS	VSS	VSS	VSS	VSS
54	PL13D	PL17D	PL20D	PL26D	I/O
55	PL13A	PL17A	PL21D	PL27D	I/O
56	PL14C	PL18C	PL21A	PL27A	I/O-SECKLL
57	PL14A	PL18A	PL22A	PL28A	I/O-A15
58	VSS	VSS	VSS	VSS	VSS
59	PCCLK	PCCLK	PCCLK	PCCLK	CCLK
60	VDD	VDD	VDD	VDD	VDD
61	VSS	VSS	VSS	VSS	VSS
62	VSS	VSS	VSS	VSS	VSS
63	PB1A	PB1A	PB1A	PB1A	I/O-A16
64	PB1D	PB1D	PB2A	PB2A	I/O
65	PB2A	PB2A	PB2D	PB2D	I/O
66	PB2D	PB2D	PB3D	PB3D	I/O
67	VSS	VSS	VSS	VSS	VSS
68	PB3A	PB3D	PB4D	PB4D	I/O-A17
69	PB3B	PB4D	PB5D	PB5D	I/O
70	PB3C	PB5A	PB6A	PB6A	I/O
71	PB3D	PB5B	PB6B	PB6D	I/O
72	PB4A	PB5D	PB6D	PB7D	I/O
73	PB4B	PB6A	PB7A	PB8A	I/O
74	PB4C	PB6B	PB7B	PB8D	I/O
75	PB4D	PB6D	PB7D	PB9D	I/O
76	VDD	VDD	VDD	VDD	VDD
77	PB5A	PB7A	PB8A	PB10A	I/O
78	PB5B	PB7B	PB8D	PB10D	I/O
79	PB5C	PB7C	PB9A	PB11A	I/O
80	PB5D	PB7D	PB9C	PB11D	I/O
81	PB6A	PB8A	PB9D	PB12A	I/O
82	PB6B	PB8B	PB10A	PB12D	I/O
83	PB6C	PB8C	PB10B	PB13A	I/O
84	PB6D	PB8D	PB10D	PB13D	I/O

Pin	OR3T55 Pad	OR3C/T80 Pad	OR3T125 Pad	Function
AF7	PB4C	PB5C	PB5C	I/O
AD7	PB4D	PB5D	PB5D	I/O
AE8	PB5A	PB6A	PB6A	I/O
AC9	PB5B	PB6B	PB6D	I/O
AF8	PB5C	PB6C	PB7A	I/O
AD8	PB5D	PB6D	PB7D	I/O
AE9	PB6A	PB7A	PB8A	I/O
AF9	PB6B	PB7B	PB8D	I/O
AE10	PB6C	PB7C	PB9A	I/O
AD9	PB6D	PB7D	PB9D	I/O
AF10	PB7A	PB8A	PB10A	I/O
AC10	PB7B	PB8D	PB10D	I/O
AE11	PB7C	PB9A	PB11A	I/O
AD10	PB7D	PB9C	PB11D	I/O
AF11	PB8A	PB9D	PB12A	I/O
AE12	PB8B	PB10A	PB12D	I/O
AF12	PB8C	PB10B	PB13A	I/O
AD11	PB8D	PB10D	PB13D	I/O
AE13	PB9A	PB11A	PB14A	I/O
AC12	PB9B	PB11B	PB14B	I/O
AF13	PB9C	PB11C	PB14C	I/O
AD12	PB9D	PB11D	PB14D	I/O
AE14	PECKB	PECKB	PECKB	I-ECKB
AC14	PB10B	PB12B	PB15B	I/O
AF14	PB10C	PB12C	PB15C	I/O
AD13	PB10D	PB12D	PB15D	I/O
AE15	PB11A	PB13A	PB16A	I/O
AD14	PB11B	PB13B	PB16D	I/O
AF15	PB11C	PB13C	PB17A	I/O
AE16	PB11D	PB14A	PB17D	I/O
AD15	PB12A	PB14B	PB18A	I/O-HDC
AF16	PB12B	PB14D	PB18D	I/O
AC15	PB12C	PB15A	PB19A	I/O
AE17	PB12D	PB15D	PB19D	I/O
AD16	PB13A	PB16A	PB20A	I/O-$\overline{\text{LDC}}$
AF17	PB13B	PB16B	PB20D	I/O
AC17	PB13C	PB16C	PB21A	I/O
AE18	PB13D	PB16D	PB21D	I/O
AD17	PB14A	PB17A	PB22A	I/O
AF18	PB14B	PB17B	PB23A	I/O
AE19	PB14C	PB17C	PB23C	I/O
AF19	PB14D	PB17D	PB23D	I/O
AD18	PB15A	PB18A	PB24A	I/O-$\overline{\text{INIT}}$
AE20	PB15B	PB18B	PB24B	I/O
AC19	PB15C	PB18C	PB24C	I/O
AF20	PB15D	PB18D	PB24D	I/O

Pin	OR3T55 Pad	OR3C/T80 Pad	OR3T125 Pad	Function
R16	VSS	VSS	VSS	VSS*
T11	VSS	VSS	VSS	VSS*
T12	VSS	VSS	VSS	VSS*
T13	VSS	VSS	VSS	VSS*
T14	VSS	VSS	VSS	VSS*
T15	VSS	VSS	VSS	VSS*
T16	VSS	VSS	VSS	VSS*
AA23	VDD	VDD	VDD	VDD
AA4	VDD	VDD	VDD	VDD
AC11	VDD	VDD	VDD	VDD
AC16	VDD	VDD	VDD	VDD
AC21	VDD	VDD	VDD	VDD
AC6	VDD	VDD	VDD	VDD
D11	VDD	VDD	VDD	VDD
D16	VDD	VDD	VDD	VDD
D21	VDD	VDD	VDD	VDD
D6	VDD	VDD	VDD	VDD
F23	VDD	VDD	VDD	VDD
F4	VDD	VDD	VDD	VDD
L23	VDD	VDD	VDD	VDD
L4	VDD	VDD	VDD	VDD
T23	VDD	VDD	VDD	VDD
T4	VDD	VDD	VDD	VDD

*Thermally enhanced connection.

Pin	OR3C/T80 Pad	OR3T125 Pad	Function
AA28	VDD	VDD	VDD
AA4	VDD	VDD	VDD
AE28	VDD	VDD	VDD
AE4	VDD	VDD	VDD
AH11	VDD	VDD	VDD
AH15	VDD	VDD	VDD
AH17	VDD	VDD	VDD
AH21	VDD	VDD	VDD
AH25	VDD	VDD	VDD
AH28	VDD	VDD	VDD
AH4	VDD	VDD	VDD
AH7	VDD	VDD	VDD
AJ29	VDD	VDD	VDD
AJ3	VDD	VDD	VDD
AK2	VDD	VDD	VDD
AK30	VDD	VDD	VDD
AL1	VDD	VDD	VDD
AL31	VDD	VDD	VDD
B2	VDD	VDD	VDD
B30	VDD	VDD	VDD
C29	VDD	VDD	VDD
C3	VDD	VDD	VDD
D11	VDD	VDD	VDD
D15	VDD	VDD	VDD
D17	VDD	VDD	VDD
D21	VDD	VDD	VDD
D25	VDD	VDD	VDD
D28	VDD	VDD	VDD
D4	VDD	VDD	VDD
D7	VDD	VDD	VDD
G28	VDD	VDD	VDD
G4	VDD	VDD	VDD
L28	VDD	VDD	VDD
L4	VDD	VDD	VDD
R28	VDD	VDD	VDD
R4	VDD	VDD	VDD
U28	VDD	VDD	VDD
U4	VDD	VDD	VDD

ψJC

This JEDEC designated parameter correlates the junction temperature to the case temperature. It is generally used to infer the junction temperature while the device is operating in the system. It is not considered a true thermal resistance, and it is defined by:

$$\psi_{JC} = \frac{T_J - T_C}{Q}$$

where T_C is the case temperature at top dead center, T_J is the junction temperature, and Q is the chip power. During the Θ_{JA} measurements described above, besides the other parameters measured, an additional temperature reading, T_C, is made with a thermocouple attached at top-dead-center of the case. ψ_{JC} is also expressed in units of °C/watt.

ΘJC

This is the thermal resistance from junction to case. It is most often used when attaching a heat sink to the top of the package. It is defined by:

$$\Theta_{JC} = \frac{T_J - T_C}{Q}$$

The parameters in this equation have been defined above. However, the measurements are performed with the case of the part pressed against a water-cooled heat sink so as to draw most of the heat generated by the chip out the top of the package. It is this difference in the measurement process that differentiates Θ_{JC} from ψ_{JC}. Θ_{JC} is a true thermal resistance and is expressed in units of °C/watt.

ΘJB

This is the thermal resistance from junction to board (a.k.a. Θ_{JL}). It is defined by:

$$\Theta_{JB} = \frac{T_J - T_B}{Q}$$

where T_B is the temperature of the board adjacent to a lead measured with a thermocouple. The other parameters on the right-hand side have been defined above. This is considered a true thermal resistance, and the measurement is made with a water-cooled heat sink pressed against the board so as to draw most of the heat out of the leads. Note that Θ_{JB} is expressed in units of °C/watt, and that this parameter and the way it is measured is still in JEDEC committee.

Package Thermal Characteristics (continued)

FPGA Maximum Junction Temperature

Once the power dissipated by the FPGA has been determined (see the Estimating Power Dissipation section), the maximum junction temperature of the FPGA can be found. This is needed to determine if speed derating of the device from the 85 °C junction temperature used in all of the delay tables is needed. Using the maximum ambient temperature, T_{Amax}, and the power dissipated by the device, Q (expressed in °C), the maximum junction temperature is approximated by:

$$T_{Jmax} = T_{Amax} + (Q \bullet \Theta_{JA})$$

Table 76 lists the plastic package thermal characteristics for the *ORCA* Series FPGAs.

Table 76. Plastic Package Thermal Characteristics for the *ORCA* Series[1]

Package	Θ_{JA} (°C/W)			T_A = 70 °C max T_J = 125 °C max @ 0 fpm (W)
	0 fpm	200 fpm	500 fpm	
144-Pin TQFP[1]	52.0	39.0	–	1.1
208-Pin SQFP[1]	26.5	23.0	21.0	2.1
208-Pin SQFP2[1]	12.8	10.3	9.1	4.3
240-Pin SQFP[1]	25.5	22.5	21.0	2.2
240-Pin SQFP2[1]	13.0	10.0	9.0	4.2
256-Pin PBGA[1, 2]	22.5	19.0	17.5	2.4
256-Pin PBGA[1, 3]	26.0	22.0	20.5	2.1
352-Pin PBGA[1, 2]	19.0	16.0	15.0	2.9
352-Pin PBGA[1, 3]	25.5	22.0	20.5	2.1
432-Pin EBGA[1]	11.0	8.5	7.5	5.0

1. Mounted on 4-layer JEDEC standard test board with two power/ground planes.
2. With thermal balls connected to board ground plane.
3. Without thermal balls connected to board ground plane.

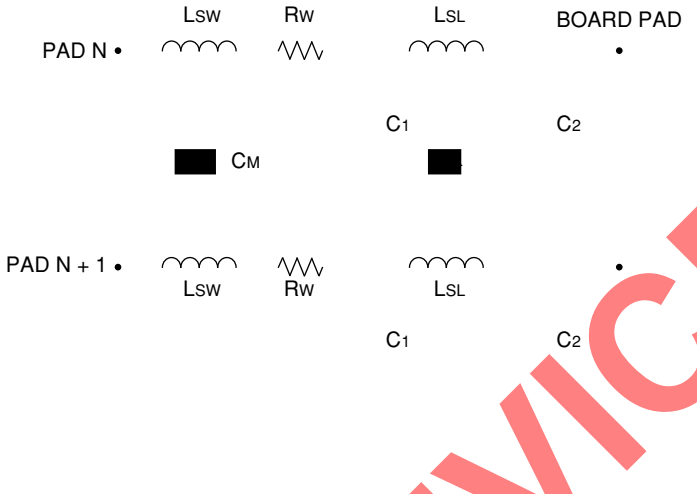


Figure 104. Package Parasitics

Package Outline Diagrams

Terms and Definitions

Basic Size (BSC): The basic size of a dimension is the size from which the limits for that dimension are derived by the application of the allowance and the tolerance.

Design Size: The design size of a dimension is the actual size of the design, including an allowance for fit and tolerance.

Typical (TYP): When specified after a dimension, this indicates the repeated design size if a tolerance is specified or repeated basic size if a tolerance is not specified.

Reference (REF): The reference dimension is an untoleranced dimension used for informational purposes only. It is a repeated dimension or one that can be derived from other values in the drawing.

Minimum (MIN) or Maximum (MAX): Indicates the minimum or maximum allowable size of a dimension.