
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 | 1568 |
| Total RAM Bits | 25600 |
| Number of I/O | 221 |
| Number of Gates | 48000 |
| Voltage - Supply | 3V ~ 3.6V |
| Mounting Type | Surface Mount |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Package / Case | 256-BGA |
| Supplier Device Package | 256-FPBGA (17x17) |
| Purchase URL | https://www.e-xfl.com/product-detail/lattice-semiconductor/or3t306ba256i-db |

# Table of Contents

## Programmable Logic Cells (continued)

### Look-Up Table Operating Modes

The operating mode affects the functionality of the PFU input and output ports and internal PFU routing. For example, in some operating modes, the DIN[7:0] inputs are direct data inputs to the PFU latches/FFs. In memory mode, the same DIN[7:0] inputs are used as a 4-bit write data input bus and a 4-bit write address input bus into LUT memory.

Table 3 lists the basic operating modes of the LUT. Figure 4—Figure 10 show block diagrams of the LUT operating modes. The accompanying descriptions demonstrate each mode's use for generating logic.

**Table 3. Look-Up Table Operating Modes**

| Mode | Function |
|---|---|
| Logic | 4- and 5-input LUTs; softwired LUTs; latches/FFs with direct input or LUT input; CIN as direct input to ninth FF or as pass through to COUT. |
| Half Logic/ Half Ripple | Upper four LUTs and latches/FFs in logic mode; lower four LUTs and latches/FFs in ripple mode; CIN and ninth FF for logic or ripple functions. |
| Ripple | All LUTs combined to perform ripple-through data functions. Eight LUT registers available for direct-in use or to register ripple output. Ninth FF dedicated to ripple out, if used. The submodes of ripple mode are adder/subtractor, counter, multiplier, and comparator. |
| Memory | All LUTs and latches/FFs used to create a 32 x 4 synchronous dual-port RAM. Can be used as single-port or as ROM. |

#### PFU Control Inputs

Each PFU has five routable control inputs and an active-low, asynchronous global set/reset (GSRN) signal that affects all latches and FFs in the device. The five control inputs are CLK, LSR, CE, ASWE, and SEL, and their functionality for each logic mode of the PFU (discussed subsequently) is shown in Table 4. The clock signal to the PFU is CLK, CE stands for clock enable, which is its primary function. LSR is the local set/reset signal that can be configured as synchronous or asynchronous. The selection of set or reset is made for each latch/FF and is not a function of the signal itself. ASWE stands for add/subtract/write enable, which are its functions, along with being an optional clock enable, and SEL is used to dynamically select between direct PFU input and LUT output data as the input to the latches/FFs.

All of the control signals can be disabled and/or inverted via the configuration logic. A disabled clock enable indicates that the clock is always enabled. A disabled LSR indicates that the latch/FF never sets/resets (except from GSRN). A disabled SEL input indicates that DIN[7:0] PFU inputs are routed to the latches/FFs. For logic and ripple modes of the PFU, the LSR, CE, and ASWE (as a clock enable) inputs can be disabled individually for each nibble (latch/FF[3:0], latch/FF[7:4]) and for the ninth FF.

## 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.

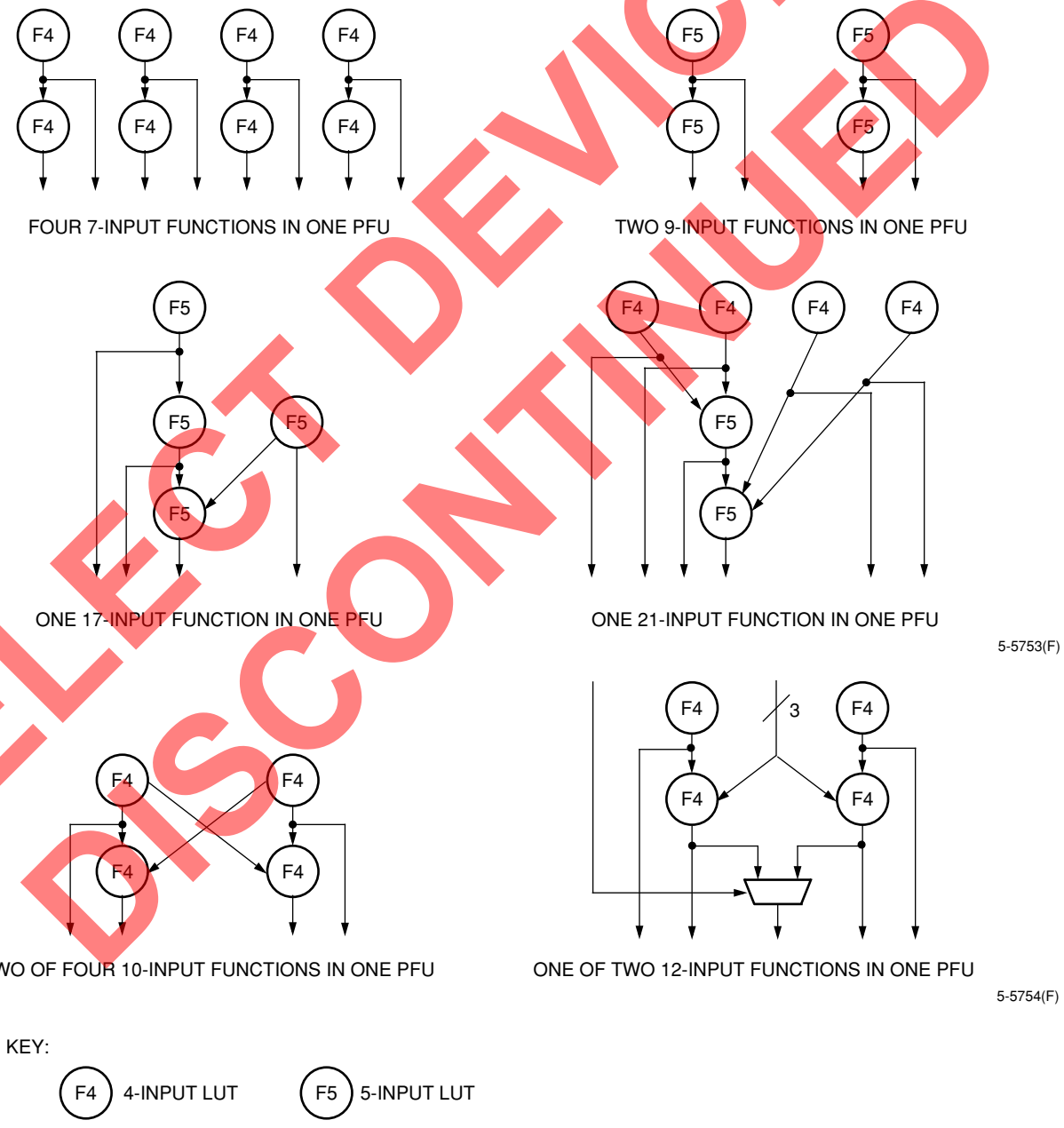



**Figure 5. Softwired LUT Topology Examples**

## Programmable Input/Output Cells

The programmable input/output cells (PICs) are located along the perimeter of the device. The PIC's name is represented by a two-letter designation to indicate on which side of the device it is located followed by a number to indicate in which row or column it is located. The first letter, P, designates that the cell is a PIC and not a PLC. The second letter indicates the side of the array where the PIC is located. The four sides are left (L), right (R), top (T), and bottom (B). The individual I/O pad is indicated by a single letter (either A, B, C, or D) placed at the end of the PIC name. As an example, PL10A indicates a pad located on the left side of the array in the tenth row.

Each PIC interfaces to four bond pads and contains the necessary routing resources to provide an interface between I/O pads and the PLCs. Each PIC is composed of four programmable I/Os (PIOs) and significant routing resources. Each PIO contains input buffers, output buffers, routing resources, latches/FFs, and logic and can be configured as an input, output, or bidirectional I/O.

PICs in the Series 3 FPGAs have significant local routing resources, similar to routing in the PLCs. This new routing increases the ability to fix user pinouts prior to placement and routing of a design and still maintain routability. The flexibility provided by the routing also provides for increased signal speed due to a greater variety of signal paths possible.

Included in the PIC routing is a fast path from the input pins to the SLICs in each of the three adjacent PLCs (one orthogonal and two diagonal). This feature allows for input signals to be very quickly processed by the SLIC decoder function and used on-chip or sent back off of the FPGA. Also new to the Series 3 PIOs are latches and FFs and options for using fast, dedicated clocks called ExpressCLKs. These features will all be discussed in subsequent sections.

A diagram of a single PIO (one of four in a PIC) is shown in Figure 22. Table 9 provides an overview of the programmable functions in an I/O cell.

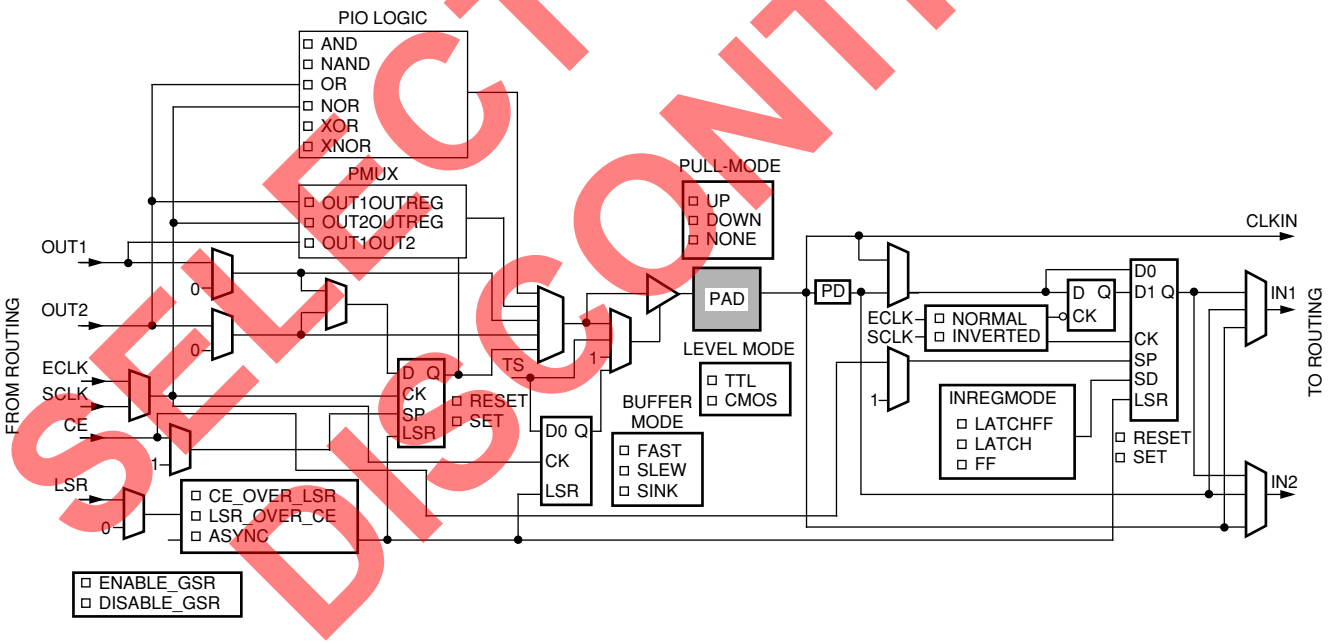
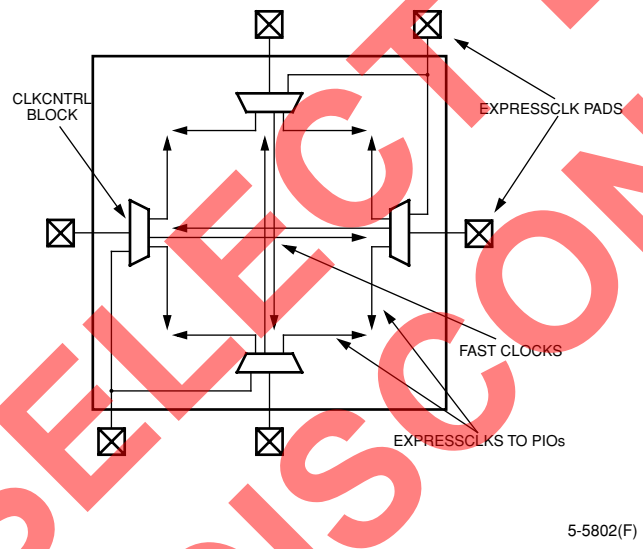



**Figure 22. OR3C/Txxx Programmable Input/Output (PIO) Image from ispLEVER**

## Clock Distribution Network (continued)

### ExpressCLK Inputs

There are four dedicated ExpressCLK pads on each Series 3 device: one in the middle of each side. Two other user I/O pads can also be used as corner ExpressCLK inputs, one on the lower-left corner, and one on the upper-right corner. The corner ExpressCLK pads feed the ExpressCLK to the two sides of the array that are adjacent to that corner, always driving the same signal in both directions. The ExpressCLK route from the middle pad and from the corner pad associated with that side are multiplexed and can be glitchlessly stopped/started under user control using the StopCLK feature of the CLKCNTRL function block (described under Special Function Blocks) on that side. The ExpressCLK output of the programmable clock manager (PCM) is programmably connected to the corner ExpressCLK routes. PCM blocks are found in the same corners as the corner ExpressCLK signals and are described in the Special Function Blocks section. The ExpressCLK structure is shown in Figure 34 (PCM blocks are not shown).




Note: All multiplexers are set during configuration.

**Figure 34. ExpressCLK and Fast Clock Distribution**

### Selecting Clock Input Pins

Any user I/O pin on an *ORCA* FPGA can be used as a fast, low-skew system clock input. Since the four dedicated ExpressCLK inputs can only be used to distribute global signals into the FPGA, these pins should be selected first as clock pins. Within the interquad region of the device, these clocks sourced by the ExpressCLK inputs are called fast clocks. Choosing the next clock pin is completely arbitrary, but using a pin that is near the center of an edge of the device will provide the lowest skew system clock network. The pin-to-pin timing numbers in the Timing Characteristics section assume that the clock pin is in one of the PICs at the center of any side of the device next to an ExpressCLK pad. For actual timing characteristics for a given clock pin, use the timing analyzer results from ispLEVER.

To select subsequent clock pins, certain rules should be followed. As discussed in the Programmable Input/Output Cells section, PICs are grouped into adjacent pairs. Each of these pairs contains eight I/Os, but only one of the eight I/Os in a PIC pair can be routed directly onto a system clock spine. Therefore, to achieve top performance, the next clock input chosen should not be one of the pins from a PIC pair previously used for a clock input. If it is necessary to have a second input in the same PIC pair route onto global system clock routing, the input can be routed to a free clock spine using the PIC switching segment (pSW) connections to the clock spine network at some small sacrifice in speed. Alternatively, if global distribution of the secondary clock is not required, the signal can be routed on long lines (xL) and input to the PFU clock input without using a clock spine.

Another rule for choosing clock pins has to do with the alternating nature of clock spine connections to the xL and pxL routing segments. Starting at the left side of the device, the first vertical clock spine from the top connects to hxL[0] (horizontal xL[0]), and the first vertical clock spine from the bottom connects to hxL[5] in all PLC rows. The next vertical clock spine from the top connects to hxL[1], and the next one from the bottom connects to hxL[6]. This progression continues across the device, and after a spine connects to hxL[9], the next spine connects to hxL[0] again. Similar connections are made from horizontal clock spines to vxL (vertical xL) lines from the top to the bottom of the device. Because the *ORCA* Series 3 clock routing only requires the use of an xL line in every other row or column, even two inputs chosen 20 PLCs apart on the same xL line will not conflict, but it is always better to avoid these choices, if possible. The fast clock spines in the interquad routing region also connect to xL[8] and xL[9] for each set of xL lines, so it is better to avoid user I/Os that connect to xL[8] or xL[9] when a fast clock is used that might share one of these connections. Another reason to use the fast clock spines is that since they use only the xL[9:8] lines, they will not conflict with internal data buses which typically use xL[7:0]. For more details on clock selection, refer to application notes on clock distribution in *ORCA* Series 3 devices.

## Special Function Blocks (continued)

The external test (EXTEST) instruction allows the interconnections between ICs in a system to be tested for opens and stuck-at faults. If an EXTEST instruction is performed for the system shown in Figure 36, the connections between U1 and U2 (shown by nets a, b, and c) can be tested by driving a value onto the given nets from one device and then determining whether the same value is seen at the other device. This is determined by shifting 2 bits of data for each pin (one for the output value and one for the 3-state value) through the BSR until each one aligns to the appropriate pin. Then, based upon the value of the 3-state signal, either the I/O pad is driven to the value given in the BSR, or the BSR is updated with the input value from the I/O pad, which allows it to be shifted out TDO.

The SAMPLE/PRELOAD instruction is useful for system debugging and fault diagnosis by allowing the data at the FPGA's I/Os to be observed during normal operation or written during test operation. The data for all of the I/Os is captured simultaneously into the BSR, allowing them to be shifted-out TDO to the test host. Since each I/O buffer in the PICs is bidirectional, two pieces of data are captured for each I/O pad: the value at the I/O pad and the value of the 3-state control signal. For preload operation, data is written from the BSR to all of the I/Os simultaneously.

There are five *ORCA*-defined instructions. The PLC scan rings 1 and 2 (PSR1, PSR2) allow user-defined internal scan paths using the PLC latches/FFs. The RAM_Write Enable (RAM_W) instruction allows the user to serially configure the FPGA through TDI. The RAM_Read Enable (RAM_R) allows the user to read back RAM contents on TDO after configuration. The IDCODE instruction allows the user to capture a 32-bit identification code that is unique to each device and serially output it at TDO. The IDCODE format is shown in Table 14.

**Table 14. Boundary-Scan ID Code**

| Device | Version (4 bits) | Part* (10 bits) | Family (6 bits) | Manufacturer (11 bits) | LSB (1 bit) |
|---|---|---|---|---|---|
| OR3T20 | 0000 | 0011000000 | 110000 | 00000011101 | 1 |
| OR3T30 | 0000 | 0111000000 | 110000 | 00000011101 | 1 |
| OR3T55 | 0000 | 0100100000 | 110000 | 00000011101 | 1 |
| OR3C/T80 | 0000 | 0110100000 | 110000 | 00000011101 | 1 |
| OR3T125 | 0000 | 0011100000 | 110000 | 00000011101 | 1 |

* PLC array size of FPGA, reverse bit order.

Note: Table assumes version 0.

## 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.

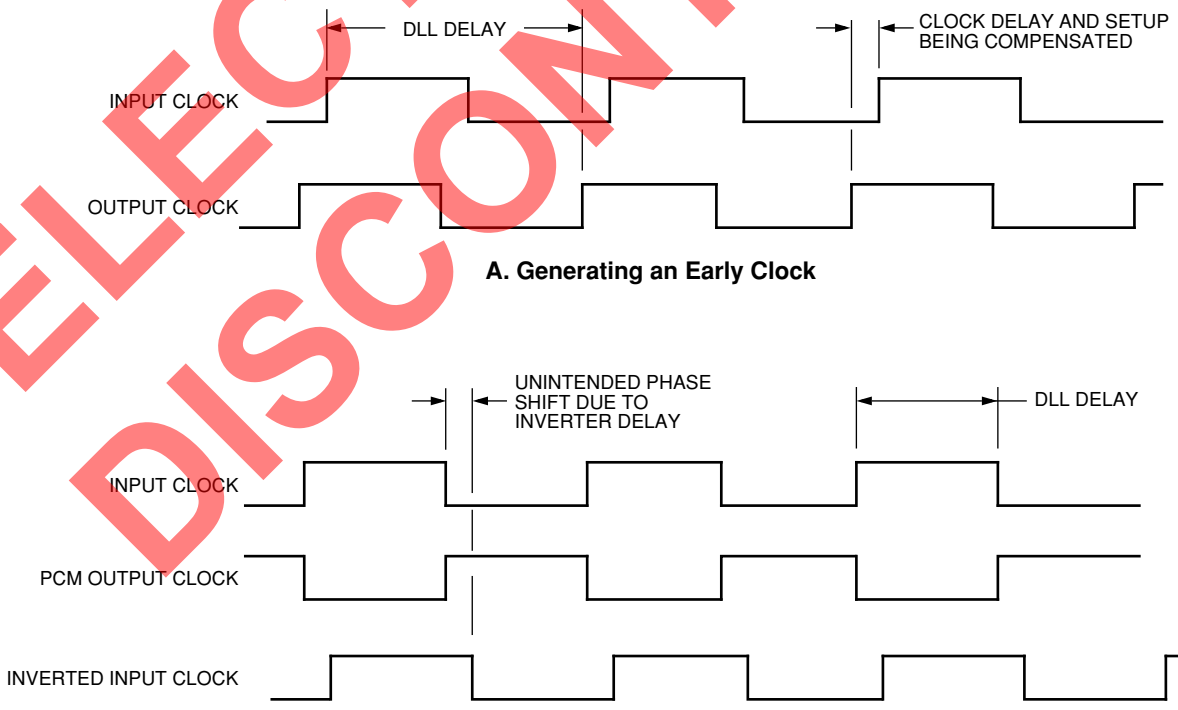
In some systems, it is desirable to operate logic from several clocks that operate at different phases. This technique is often used in microprocessor-based systems to transfer and process data synchronously between functional areas, but without incurring excessive delays. Figure 48B shows an input clock 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





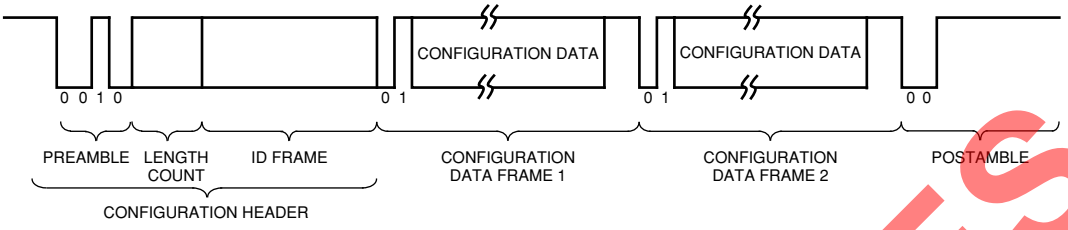

**A. Generating an Early Clock**

**B. Multiphase Clock Generation Using the DLL**

5-5979(F)

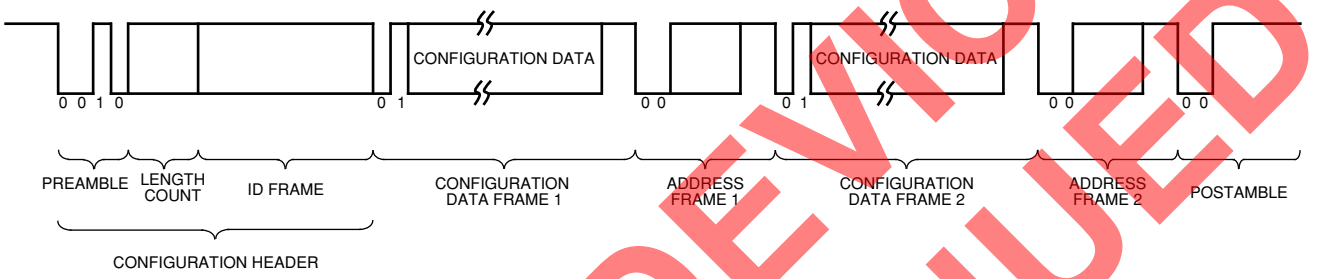
**Figure 48. Clock Phase Adjustment Using the PCM**

## 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**

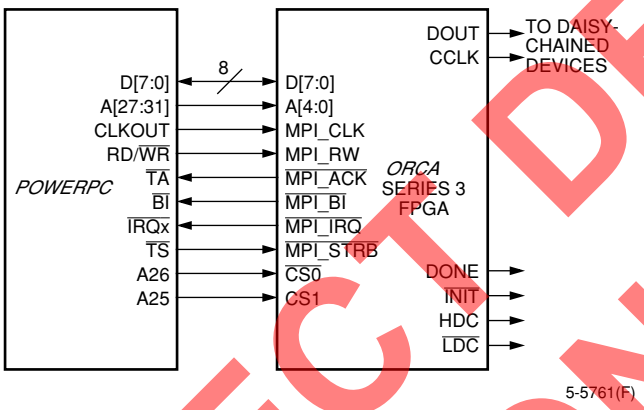
| | | |
|---|---|---|
| **Header** | 11110010 | Preamble |
| | 24-bit Length Count | Configuration frame length. |
| | 11111111 | Trailing header—8 bits. |
| **ID Frame** | 0101 1111 1111 1111 | ID frame header. |
| | Configuration Mode | 00 = autoincrement, 01 = explicit. |
| | Reserved [41:0] | Reserved 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.

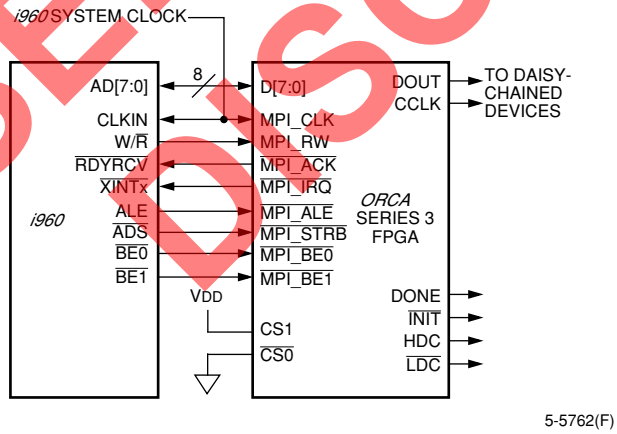
## FPGA Configuration Modes (continued)

There are two options for using the host interrupt request in configuration mode. The configuration control register offers control bits to enable the interrupt on either a bit stream error or to notify the host processor when the FPGA is ready for more configuration data. The MPI status register may be used in conjunction with, or in place of, the interrupt request options. The status register contains a 2-bit field to indicate the bit stream error status. As previously mentioned, there is also a bit to indicate the MPI's readiness to receive another byte of configuration data. A flow chart of the MPI configuration process is shown in Figure 59. The MPI status and configuration register bit maps can be found in the Special Function Blocks section and MPI configuration timing information is available in the Timing Characteristics section of this data sheet.




Note: FPGA shown as a memory-mapped peripheral using CS0 and CS1. Other decoding schemes are possible using CS0 and/or CS1.

**Figure 57. *PowerPC*/MPI Configuration Schematic**




Note: FPGA shown as only system peripheral with fixed chip select signals. For multiperipheral systems, address decoding and/or latching can be used to implement chip selects.

**Figure 58. *i960*/MPI Configuration Schematic**

Configuration readback can also be performed via the MPI when it is in user mode. The MPI is enabled in user mode by setting the MP_USER bit to 1 in the configuration control register prior to the start of configuration or through a configuration option. To perform readback, the host processor writes the 14-bit readback start address to the readback address registers and sets the RD_CFG bit to 0 in the configuration control register. Readback data is returned 8 bits at a time to the readback data register and is valid when the DATA_RDY bit of the status register is 1. There is no error checking during readback. A flow chart of the MPI readback operation is shown in Figure 60. The RD_DATA pin used for dedicated FPGA readback is invalid during MPI readback.

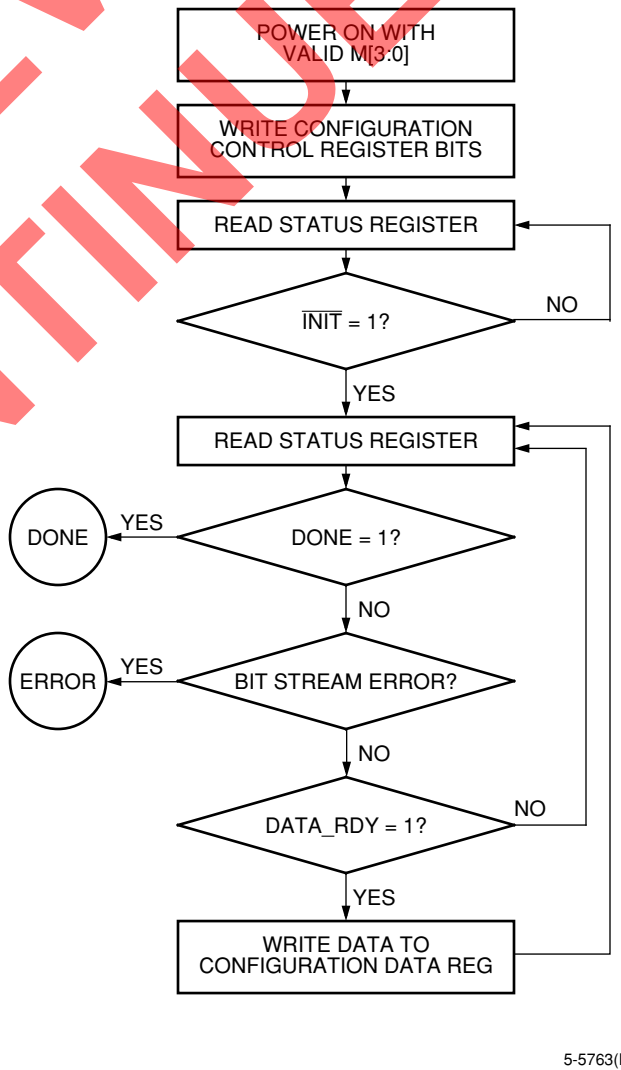



**Figure 59. Configuration Through MPI**

## FPGA Configuration Modes (continued)

### Slave Serial Mode

The slave serial mode is primarily used when multiple FPGAs are configured in a daisy-chain (see the Daisy-Chaining section). It is also used on the FPGA evaluation board that interfaces to the download cable. A device in the slave serial mode can be used as the lead device in a daisy-chain. Figure 61 shows the connections for the slave serial configuration mode.

The configuration data is provided into the FPGA's DIN input synchronous with the configuration clock CCLK input. After the FPGA has loaded its configuration data, it retransmits the incoming configuration data on DOUT. CCLK is routed into all slave serial mode devices in parallel.

Multiple slave FPGAs can be loaded with identical configurations simultaneously. This is done by loading the configuration data into the DIN inputs in parallel.

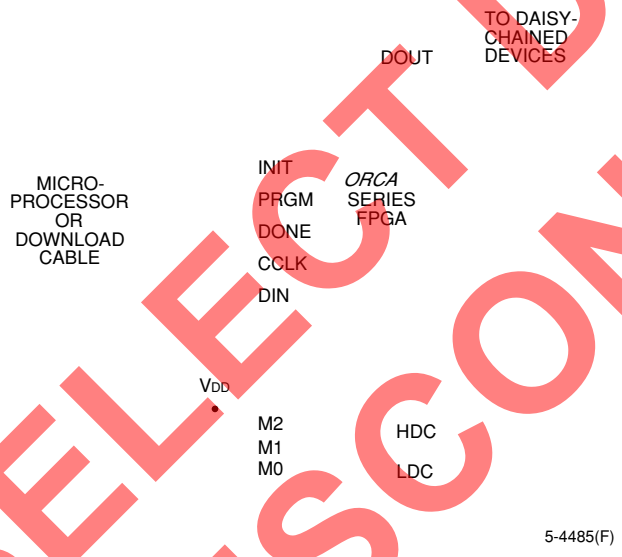



**Figure 61. Slave Serial Configuration Schematic**

### Slave Parallel Mode

The slave parallel mode is essentially the same as the slave serial mode except that 8 bits of data are input on pins D[7:0] for each CCLK cycle. Due to 8 bits of data being input per CCLK cycle, the DOUT pin does not contain a valid bit stream for slave parallel mode. As a result, the lead device cannot be used in the slave parallel mode in a daisy-chain configuration.

Figure 62 is a schematic of the connections for the slave parallel configuration mode. $\overline{WR}$ and $\overline{CS0}$ are active-low chip select signals, and CS1 is an active-high chip select signal. These chip selects allow the user to configure multiple FPGAs in slave parallel mode using an 8-bit data bus common to all of the FPGAs. These chip selects can then be used to select the FPGA(s) to be configured with a given bit stream. The chip selects must be active for each valid CCLK cycle until the device has been completely programmed. They can be inactive between cycles but must meet the setup and hold times for each valid positive CCLK. D[7:0] of the FPGA can be connected to D[7:0] of the microprocessor only if a standard prom file format is used. If a .bit or .rbt file is used from ispLEVER, then the user must mirror the bytes in the .bit or .rbt file OR leave the .bit or .rbt file unchanged and connect D[7:0] of the FPGA to D[0:7] of the microprocessor.

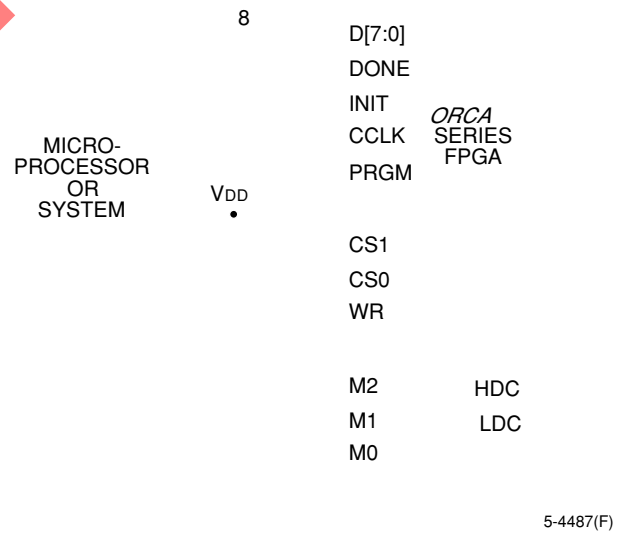



**Figure 62. Slave Parallel Configuration Schematic**

## Timing Characteristics (continued)

### Table 48. Programmable I/O (PIO) Timing Characteristics (continued)

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 | Speed | | | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | -4 | | -5 | | -6 | | -7 | | |
| | | Min | Max | Min | Max | Min | Max | Min | Max | |
| **Output Delays** ($T_J$ = 85 °C, $V_{DD}$ = min, $C_L$ = 50 pF) | | | | | | | | | | |
| Output to Pad (OUT2, OUT1 direct to pad): | | | | | | | | | | |
| Fast | OUTF_DEL | — | 5.09 | — | 4.21 | — | 2.63 | — | 2.17 | ns |
| Slewlim | OUTSL_DEL | — | 7.86 | — | 6.49 | — | 3.49 | — | 2.91 | ns |
| Sinklim | OUTSI_DEL | — | 9.41 | — | 7.98 | — | 8.08 | — | 7.32 | ns |
| 3-state Enable/Disable Delay (TS to pad): | | | | | | | | | | |
| Fast | TSF_DEL | — | 4.93 | — | 4.09 | — | 2.33 | — | 1.88 | ns |
| Slewlim | TSSL_DEL | — | 7.70 | — | 6.37 | — | 3.00 | — | 2.41 | ns |
| Sinklim | TSSI_DEL | — | 9.25 | — | 7.86 | — | 7.95 | — | 7.23 | ns |
| Local Set/Reset (async) to Pad (LSR to pad): | | | | | | | | | | |
| Fast | OUTLSRF_DEL | — | 9.03 | — | 7.25 | — | 4.96 | — | 3.94 | ns |
| Slewlim | OUTLSRSL_DEL | — | 11.79 | — | 9.53 | — | 5.82 | — | 4.67 | ns |
| Sinklim | OUTLSRSI_DEL | — | 13.35 | — | 11.02 | — | 10.38 | — | 9.10 | ns |
| Global Set/Reset to Pad (GSRN to pad): | | | | | | | | | | |
| Fast | OUTGSRF_DEL | — | 8.30 | — | 6.69 | — | 4.39 | — | 3.46 | ns |
| Slewlim | OUTGSRSL_DEL | — | 11.06 | — | 8.97 | — | 5.07 | — | 3.99 | ns |
| Sinklim | OUTGSRSI_DEL | — | 12.62 | — | 10.46 | — | 10.02 | — | 8.81 | ns |
| Output FF Setup Timing: | | | | | | | | | | |
| Out to ExpressCLK (OUT[2:1] to ECLK) | OUTE_SET | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| Out to Clock (OUT[2:1] to CLK) | OUT_SET | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| Clock Enable to Clock (CE to CLK) | OUTCE_SET | 0.91 | — | 0.67 | — | 0.56 | — | 0.45 | — | ns |
| Local Set/Reset (sync) to Clock (LSR to CLK) | OUTLSR_SET | 0.41 | — | 0.32 | — | 0.26 | — | 0.24 | — | ns |
| Output FF Hold Timing: | | | | | | | | | | |
| Out from ExpressCLK (OUT[2:1] from ECLK) | OUTE_HLD | 0.73 | — | 0.58 | — | 0.36 | — | 0.29 | — | ns |
| Out from Clock (OUT[2:1] from CLK) | OUT_HLD | 0.73 | — | 0.58 | — | 0.36 | — | 0.29 | — | ns |
| Clock Enable from Clock (CE from CLK) | OUTCE_HLD | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| Local Set/Reset (sync) from Clock (LSR from CLK) | OUTLSR_HLD | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| Clock to Pad Delay (ECLK, SCLK to pad): | | | | | | | | | | |
| Fast | OUTREGF_DEL | — | 6.71 | — | 5.44 | — | 3.56 | — | 2.78 | ns |
| Slewlim | OUTREGSL_DEL | — | 9.47 | — | 7.71 | — | 4.42 | — | 3.52 | ns |
| Sinklim | OUTREGSI_DEL | — | 11.03 | — | 9.20 | — | 8.98 | — | 7.94 | ns |
| Additional Delay If Using Open Drain | OD_DEL | — | 0.20 | — | 0.16 | — | 0.10 | — | 0.08 | ns |

Note: The delays for all input buffers assume an input rise/fall time of <1 V/ns.

## Timing Characteristics (continued)

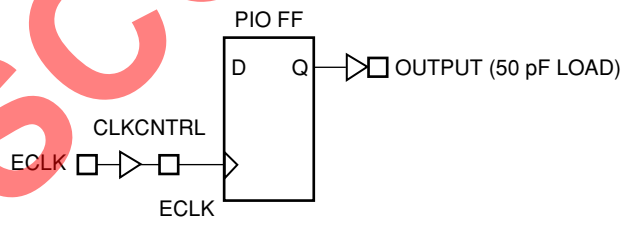
### Table 54. OR3Cxx ExpressCLK to Output Delay (Pin-to-Pin)

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; $C_L$ = 50 pF.
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; $C_L$ = 50 pF.

| Description ($T_J$ = 85 °C, $V_{DD}$ = min) | Device | Speed | | | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | -4 | | -5 | | -6 | | -7 | | |
| | | Min | Max | Min | Max | Min | Max | Min | Max | |
| ECLK Middle Input Pin→OUTPUT Pin (Fast) | OR3T20 | — | — | — | 7.78 | — | 5.40 | — | 4.38 | ns |
| | OR3T30 | — | — | — | 7.84 | — | 5.43 | — | 4.40 | ns |
| | OR3T55 | — | 9.93 | — | 7.96 | — | 5.48 | — | 4.44 | ns |
| | OR3C/T80 | — | 10.10 | — | 8.08 | — | 5.54 | — | 4.49 | ns |
| | OR3T125 | — | — | — | 8.28 | — | 5.64 | — | 4.58 | ns |
| ECLK Middle Input Pin→OUTPUT Pin (Slewlim) | OR3T20 | — | — | — | 9.77 | — | 6.07 | — | 4.91 | ns |
| | OR3T30 | — | — | — | 9.83 | — | 6.10 | — | 4.93 | ns |
| | OR3T55 | — | 12.37 | — | 9.95 | — | 6.15 | — | 4.97 | ns |
| | OR3C/T80 | — | 12.54 | — | 10.07 | — | 6.21 | — | 5.02 | ns |
| | OR3T125 | — | — | — | 10.27 | — | 6.31 | — | 5.11 | ns |
| ECLK Middle Input Pin→OUTPUT Pin (Sinklim) | OR3T20 | — | — | — | 11.12 | — | 10.92 | — | 9.65 | ns |
| | OR3T30 | — | — | — | 11.18 | — | 10.95 | — | 9.67 | ns |
| | OR3T55 | — | 13.73 | — | 11.30 | — | 11.00 | — | 9.71 | ns |
| | OR3C/T80 | — | 13.90 | — | 11.42 | — | 11.06 | — | 9.76 | ns |
| | OR3T125 | — | — | — | 11.62 | — | 11.16 | — | 9.85 | ns |
| Additional Delay if ECLK Corner Pin Used | OR3T20 | — | — | — | 1.91 | — | 1.80 | — | 1.58 | ns |
| | OR3T30 | — | — | — | 1.91 | — | 1.90 | — | 1.67 | ns |
| | OR3T55 | — | 1.97 | — | 1.91 | — | 2.09 | — | 1.84 | ns |
| | OR3C/T80 | — | 1.97 | — | 1.91 | — | 2.28 | — | 2.02 | ns |
| | OR3T125 | — | — | — | 1.90 | — | 2.57 | — | 2.29 | ns |

Notes:
Timing is without the use of the programmable clock manager (PCM).

This clock delay is for a fully routed clock tree that uses the ExpressCLK network. It includes both the input buffer delay, the clock routing to the PIO CLK input, the clock→Q of the FF, and the delay through the output buffer. The given timing requires that the input clock pin be located at one of the six ExpressCLK inputs of the device, and that a PIO FF be used.




5-4846(F).a

**Figure 76. ExpressCLK to Output Delay**

## Timing Characteristics (continued)

### Table 58. OR3C/Txxx Input to Fast Clock Setup/Hold Time (Pin-to-Pin)

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.

| Description ($T_J$ = 85 °C, $V_{DD}$ = min) | Device | Speed | | | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | -4 | | -5 | | -6 | | -7 | | |
| | | Min | Max | Min | Max | Min | Max | Min | Max | |
| **Output Not on Same Side of Device As Input Clock (Fast Clock Delays Using ExpressCLK Inputs)** | | | | | | | | | | |
| Input to FCLK Setup Time (middle ECLK pin) | OR3T20 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T30 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T55 | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3C/T80 | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T125 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| Input to FCLK Setup Time (middle ECLK pin, delayed data input) | OR3T20 | — | — | 0.80 | — | 0.58 | — | 2.20 | — | ns |
| | OR3T30 | — | — | 0.74 | — | 0.55 | — | 2.17 | — | ns |
| | OR3T55 | 0.29 | — | 0.62 | — | 0.51 | — | 2.11 | — | ns |
| | OR3C/T80 | 0.14 | — | 0.50 | — | 0.46 | — | 2.06 | — | ns |
| | OR3T125 | — | — | 0.22 | — | 0.33 | — | 1.90 | — | ns |
| Input to FCLK Setup Time (corner ECLK pin) | OR3T20 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T30 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T55 | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3C/T80 | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T125 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| Input to FCLK Setup Time (corner ECLK pin, delayed data input) | OR3T20 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T30 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T55 | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3C/T80 | 0.00 | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| | OR3T125 | — | — | 0.00 | — | 0.00 | — | 0.00 | — | ns |
| Input to FCLK Hold Time (middle ECLK pin) | OR3T20 | — | — | 4.29 | — | 3.72 | — | 3.27 | — | ns |
| | OR3T30 | — | — | 4.50 | — | 3.80 | — | 3.35 | — | ns |
| | OR3T55 | 6.33 | — | 4.97 | — | 3.96 | — | 3.52 | — | ns |
| | OR3C/T80 | 6.95 | — | 5.49 | — | 4.15 | — | 3.72 | — | ns |
| | OR3T125 | — | — | 6.36 | — | 4.47 | — | 4.05 | — | ns |

Notes:
The pin-to-pin timing parameters in this table should be used instead of results reported by ispLEVER.

The FCLK delays are for a fully routed clock tree that uses the ExpressCLK input into the fast clock network. It includes both the input buffer delay and the clock routing to the PFU CLK input. The delay will be reduced if any of the clock branches are not used.

## Timing Characteristics (continued)

### Table 63. Asynchronous Peripheral Configuration Mode Timing Characteristics

OR3Cxx Commercial: VDD = 5.0 V ± 5%, 0 °C < TA < 70 °C; Industrial: VDD = 5.0 V ± 10%, –40 °C < TA < +85 °C.
OR3Txxx Commercial: VDD = 3.0 V to 3.6 V, 0 °C < TA < 70 °C; Industrial: VDD = 3.0 V to 3.6 V, –40 °C < TA < +85 °C.

| Parameter | Symbol | Min | Max | Unit |
|---|---|---|---|---|
| $\overline{WR}$, $\overline{CS0}$, and CS1 Pulse Width | TWR | 50.00 | — | ns |
| D[7:0] Setup Time:<br>3Cxx<br>3Txxx | TS | <br>20.00<br>10.50 | <br>—<br>— | <br>ns<br>ns |
| D[7:0] Hold Time | TH | 0.00 | — | ns |
| RDY Delay | TRDY | — | 40.00 | ns |
| RDY Low | TB | 1.00 | 8.00 | CCLK Periods |
| Earliest $\overline{WR}$ After RDY Goes High* | TWR2 | 0.00 | — | ns |
| $\overline{RD}$ to D7 Enable/Disable | TDEN | — | 40.00 | ns |
| CCLK to DOUT | TD | — | 5.00 | ns |

* This parameter is valid whether the end of not RDY is determined from the RDY pin or from the D7 pin.

Notes:
Serial data is transmitted out on DOUT on the falling edge of CCLK after the byte is input on D[7:0].
D[6:0] timing is the same as the write data portion of the D7 waveform because D[6:0] are not enabled by $\overline{RD}$.

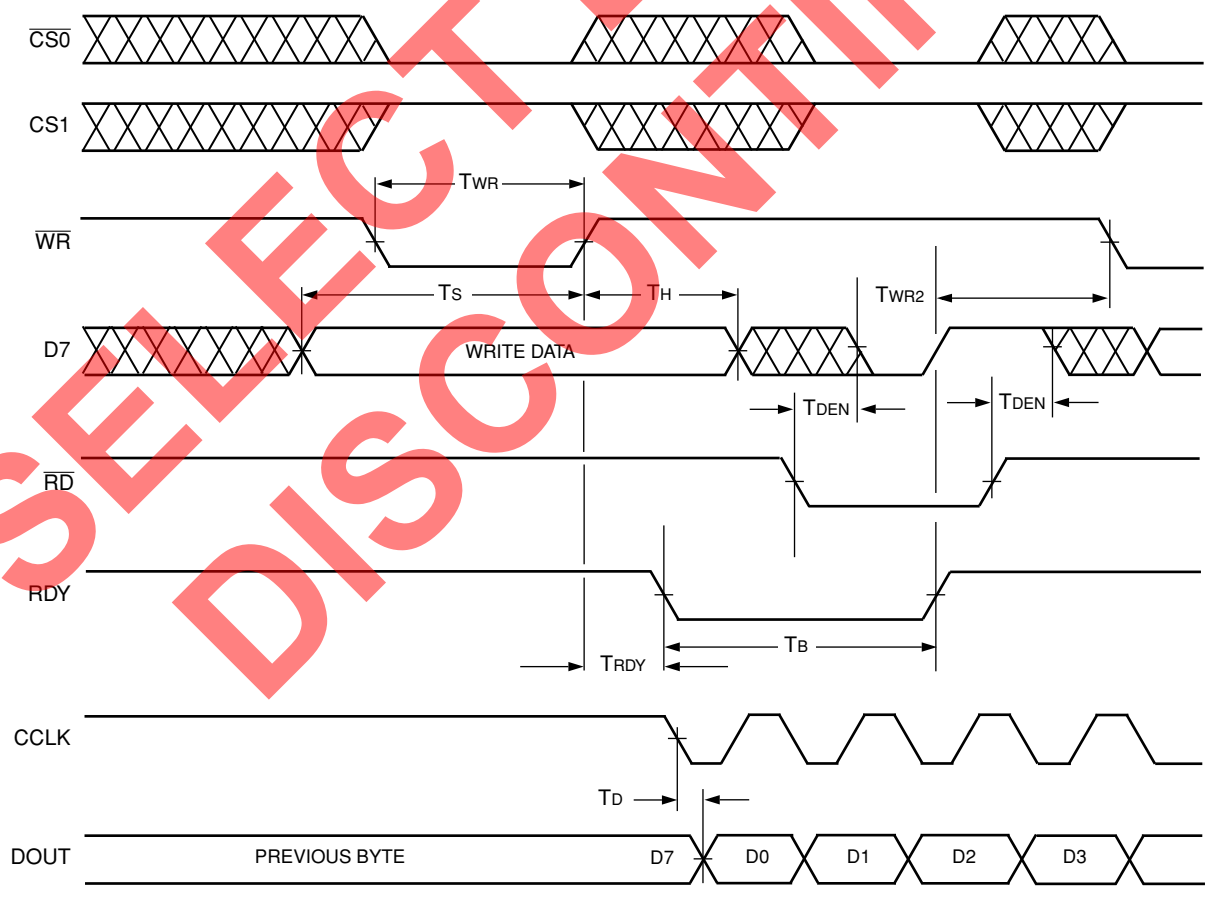



**Figure 85. Asynchronous Peripheral Configuration Mode Timing Diagram**

## Timing Characteristics (continued)

### Readback Timing

**Table 66. Readback Timing Characteristics**

OR3Cxx Commercial: VDD = 5.0 V ± 5%, 0 °C < TA < 70 °C; Industrial: VDD = 5.0 V ± 10%, –40 °C < TA < +85 °C.
OR3Txxx Commercial: VDD = 3.0 V to 3.6 V, 0 °C < TA < 70 °C; Industrial: VDD = 3.0 V to 3.6 V, –40 °C < TA < +85 °C.

| Parameter | Symbol | Min | Max | Unit |
|---|---|---|---|---|
| $\overline{\text{RD\_CFG}}$ to CCLK Setup Time | TS | 50.00 | — | ns |
| $\overline{\text{RD\_CFG}}$ High Width to Abort Readback | TRBA | 2 | — | CCLK cycles |
| CCLK Low Time | TCL | 40.00 | — | ns |
| CCLK High Time | TCH | 40.00 | — | ns |
| CCLK Frequency | FC | — | 12.50 | MHz |
| CCLK to RD_DATA Delay | TD | — | 40.00 | ns |

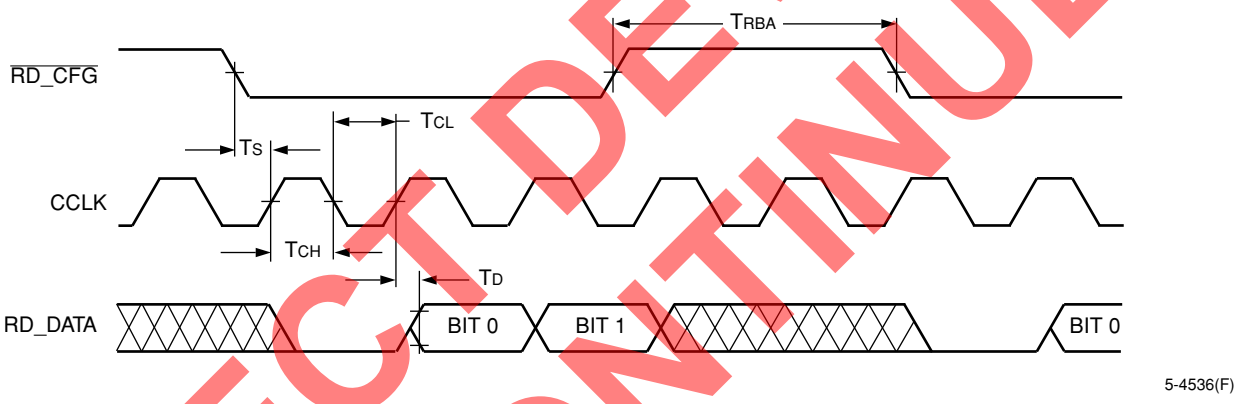



**Figure 88. Readback Timing Diagram**

**Table 70. OR3T20 144-Pin TQFP Pinout**

| Pin | OR3T20 Pad | Function |
|---|---|---|
| 1 | VDD | VDD |
| 2 | VSS | VSS |
| 3 | PL1A | I/O-A0/$\overline{\text{MPI\_BE0}}$ |
| 4 | PL2D | I/O |
| 5 | PL2A | I/O-A1/$\overline{\text{MPI\_BE1}}$ |
| 6 | PL3D | I/O-A2 |
| 7 | PL3A | I/O-A3 |
| 8 | PL4D | I/O |
| 9 | PL4C | I/O |
| 10 | PL4A | I/O-A4 |
| 11 | PL5D | I/O-A5 |
| 12 | PL5C | I/O |
| 13 | PL5A | I/O-A6 |
| 14 | VSS | VSS |
| 15 | PECKL | I-ECKL |
| 16 | PL6C | I/O |
| 17 | PL6A | I/O-A7/MPI_CLK |
| 18 | VDD | VDD |
| 19 | PL7D | I/O |
| 20 | PL7C | I/O |
| 21 | PL7A | I/O-A8/MPI_RW |
| 22 | VSS | VSS |
| 23 | PL8D | I/O-A9/$\overline{\text{MPI\_ACK}}$ |
| 24 | PL8A | I/O-A10/$\overline{\text{MPI\_BI}}$ |
| 25 | PL9D | I/O |
| 26 | PL9C | I/O |
| 27 | PL9A | I/O-A11/$\overline{\text{MPI\_IRQ}}$ |
| 28 | PL10D | I/O-A12 |
| 29 | PL10C | I/O |
| 30 | PL10A | I/O-A13 |
| 31 | PL11A | I/O-A14 |
| 32 | PL12D | I/O |
| 33 | PL12B | I/O-SECKLL |
| 34 | PL12A | I/O-A15 |
| 35 | VSS | VSS |
| 36 | PCCLK | CCLK |
| 37 | VDD | VDD |
| 38 | VSS | VSS |
| 39 | PB1A | I/O-A16 |
| 40 | PB1D | I/O |
| 41 | PB2A | I/O-A17 |
| 42 | PB3A | I/O |

| Pin | OR3T20 Pad | OR3T30 Pad | OR3T55 Pad | Function |
|---|---|---|---|---|
| Y12 | PB7D | PB8D | PB10D | I/O |
| W12 | PB8A | PB9A | PB11A | I/O |
| V12 | PB8B | PB9B | PB11B | I/O |
| U12 | PB8C | PB9C | PB11C | I/O |
| Y13 | PB8D | PB9D | PB11D | I/O |
| W13 | PB9A | PB10A | PB12A | I/O-HDC |
| V13 | PB9B | PB10B | PB12B | I/O |
| Y14 | PB9C | PB10C | PB12C | I/O |
| W14 | PB9D | PB10D | PB12D | I/O |
| Y15 | PB10A | PB11A | PB13A | I/O-$\overline{\text{LDC}}$ |
| V14 | PB10B | PB11B | PB13B | I/O |
| W15 | PB10C | PB11C | PB13C | I/O |
| Y16 | PB10D | PB11D | PB13D | I/O |
| U14 | — | PB12A | PB14A | I/O |
| V15 | — | PB12B | PB14D | I/O |
| W16 | PB11A | PB12C | PB15A | I/O-$\overline{\text{INIT}}$ |
| Y17 | — | — | PB15D | I/O |
| V16 | — | PB12D | PB16A | I/O |
| W17 | PB11B | PB13A | PB16D | I/O |
| Y18 | PB11C | PB13B | PB17A | I/O |
| U16 | PB11D | PB13C | PB17 | I/O |
| V17 | PB12A | PB13D | PB17D | I/O |
| W18 | PB12B | PB14A | PB18A | I/O |
| Y19 | PB12C | PB14B | PB18B | I/O |
| V18 | PB12D | PB1 | P | I/O |
| W19 | — | PB 1D | PB18D | I/O |
| Y20 | PDONE | PDON | DONE | DONE |
| W20 | $\overline{\text{PRESETN}}$ | $\overline{\text{PRESETN}}$ | $\overline{\text{PRESETN}}$ | $\overline{\text{PRES}}$ |
| V19 | $\overline{\text{PPRGMN}}$ | $\overline{\text{PRGM}}$ | $\overline{\text{PPRGMN}}$ | $\overline{\text{PRGM}}$ |
| U19 | PR12A | PR14A | PR18A | I/O-M0 |
| U18 | | PR14C | PR18C | I/O |
| T | — | PR14D | PR | I/O |
| R | — | PR13A | PR17A | I/O |
| U20 | PR12B | PR13B | PR17B | I/O |
| T18 | PR12C | PR13C | PR17C | I/O |
| T19 | PR12D | PR13D | PR17D | I/O |
| T20 | PR11A | PR | PR16A | I/O |
| R18 | PR11B | PR12B | PR16D | I/O |
| P17 | PR11C | PR12C | PR15A | I/O |
| R19 | PR11D | PR12D | PR15C | I/O |
| R20 | PR10A | PR11A | PR15D | I/O-M1 |
| P18 | PR10B | PR11B | PR14A | I/O |
| P19 | PR10C | PR11C | PR14D | I/O |
| P20 | PR10D | PR11D | PR13A | I/O |
| N18 | PR9A | PR10A | PR12A | I/O-M2 |

| Pin | OR3T20 Pad | OR3T30 Pad | OR3T55 Pad | Function |
|---|---|---|---|---|
| N19 | PR9B | PR10B | PR12B | I/O |
| N20 | PR9C | PR10C | PR12C | I/O |
| M17 | PR9D | PR10D | PR12D | I/O |
| M18 | PR8A | PR9A | PR11A | I/O-M3 |
| M19 | PR8B | PR9B | PR11B | I/O |
| M20 | PR8C | PR9C | PR11C | I/O |
| L19 | PR8D | PR9D | PR11D | I/O |
| L18 | PR7A | PR8A | PR10A | I/O |
| L20 | PR7B | PR8B | PR10B | I/O |
| K20 | PR7C | PR8C | PR10C | I/O |
| K19 | PR7D | PR8D | PR10D | I/O |
| K18 | PECKR | PECKR | PECKR | I-ECKR |
| K17 | PR6B | PR7B | PR9B | I/O |
| J20 | PR6C | PR7C | PR9C | I/O |
| J19 | PR6D | PR7D | PR9D | I/O |
| J18 | PR5A | PR6A | PR8A | I/O |
| J17 | PR5B | PR6B | PR8B | I/O |
| H20 | PR5C | PR6C | PR8C | I/O |
| H19 | PR5D | PR6D | PR8D | I/O |
| H18 | PR4A | PR5A | PR7A | I/O-CS1 |
| G20 | PR4B | PR5B | PR7B | I/O |
| G19 | PR4C | PR5C | PR7C | I/O |
| F20 | PR4D | PR5D | PR7D | I/O |
| G18 | PR3A | PR4A | PR6A | I/O-$\overline{CS0}$ |
| F19 | PR3B | PR4B | PR6B | I/O |
| E20 | PR3C | PR4C | PR5B | I/O |
| G17 | PR3D | PR4D | PR5D | I/O |
| F18 | PR2A | PR3A | PR4A | I/O-$\overline{RD}$/MPI_$\overline{STRB}$ |
| E19 | PR2B | PR3B | PR4B | I/O |
| D20 | PR2C | PR3C | PR4D | I/O |
| E18 | PR2D | PR3D | PR3A | I/O |
| D19 | PR1A | PR2A | PR2A | I/O-$\overline{WR}$ |
| C20 | PR1B | PR2B | PR2B | I/O |
| E17 | PR1C | PR2C | PR2C | I/O |
| D18 | PR1D | PR2D | PR2D | I/O |
| C19 | — | PR1A | PR1A | I/O |
| B20 | — | PR1B | PR1B | I/O |
| C18 | — | PR1C | PR1C | I/O |
| B19 | — | PR1D | PR1D | I/O |
| A20 | $\overline{PRD\_CFGN}$ | $\overline{PRD\_CFGN}$ | $\overline{PRD\_CFGN}$ | $\overline{RD\_CFG}$ |
| A19 | PT12D | PT14D | PT18D | I/O-SECKUR |
| B18 | — | PT14C | PT18C | I/O |
| B17 | PT12C | PT14B | PT18B | I/O |
| C17 | PT12B | PT14A | PT18A | I/O |
| D16 | PT12A | PT13D | PT17D | I/O-RDY/RCLK/MPI_ALE |
| A18 | — | PT13C | PT17A | 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.

## Package Outline Diagrams (continued)

### 208-Pin SQFP2

Dimensions are in millimeters.



5-3828(F)

DETAIL C (SQFP2 CHIP-UP)

5-3828(F).a