



CYPRESS

# Independent Clock Quad HOTLink II™ Transceiver

CYP15G0403DXB  
CYV15G0403DXB  
CYW15G0403DXB

## Features

- Second-generation HOTLink® technology
- Compliant to multiple standards
  - ESCON, DVB-ASI, SMPTE-292M, SMPTE-259M, Fibre Channel and Gigabit Ethernet (IEEE802.3z)
  - CPRI™ compliant
  - CYW15G0403DXB compliant to OBSAI-RP3
  - 8B/10B coded data or 10 bit uncoded data
- Quad channel transceiver operates from 195 to 1500 MBaud serial data rate
  - CYW15G0403DXB operates from 195 to 1540 MBaud
  - Aggregate throughput of up to 12 Gbits/second
- Second-generation HOTLink® technology
- Truly independent channels
  - Each channel can operate at a different signaling rate
  - Each channel can transport a different type of data
- Selectable input/output clocking options
- Internal phase-locked loops (PLLs) with no external PLL components
- Dual differential PECL-compatible serial inputs per channel
  - Internal DC-restoration
- Dual differential PECL-compatible serial outputs per channel
  - Source matched for 50Ω transmission lines
  - No external bias resistors required
  - Signaling-rate controlled edge-rates
- MultiFrame™ Receive Framer provides alignment options
  - Bit and byte alignment
  - Comma or Full K28.5 detect
- Single or Multi-byte Framer for byte alignment
- Low-latency option
- Synchronous LVTTTL parallel interface
- JTAG boundary scan
- Built-In Self-Test (BIST) for at-speed link testing
- Compatible with
  - Fiber-optic modules
  - Copper cables
  - Circuit board traces
- Per-channel Link Quality Indicator
  - Analog signal detect
  - Digital signal detect
- Low-power 3W @ 3.3V typical
- Single 3.3V supply
- 256-ball thermally enhanced BGA
- Pb-Free package option available
- 0.25μ BiCMOS technology

## Functional Description

The CYP(V)15G0403DXB<sup>[1]</sup> Independent Clock Quad HOTLink II™ Transceiver is a point-to-point or point-to-multi-point communications building block enabling transfer of data over a variety of high-speed serial links like optical fiber, balanced, and unbalanced copper transmission lines. The signaling rate can be anywhere in the range of 195 to 1500 MBaud per serial link. Each channel operates independently with its own reference clock allowing different rates. Each transmit channel accepts parallel characters in an Input Register, encodes each character for transport, and then converts it to serial data. Each receive channel accepts serial data and converts it to parallel data, decodes the data into characters, and presents these characters to an Output Register. *Figure 1* illustrates typical connections between independent host systems and corresponding CYP(V)(W)15G0403DXB chips.

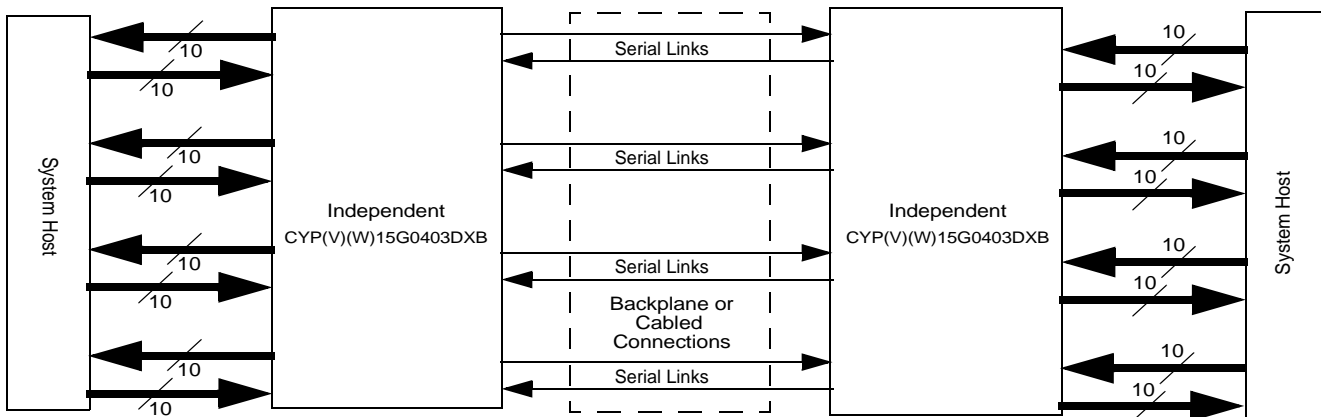


Figure 1. HOTLink II™ System Connections

### Note:

1. CYV15G0403DXB refers to SMPTE 259M and SMPTE 292M compliant devices. CYW15G0403DXB refers to OBSAI RP3 compliant devices (maximum operating data rate is 1540 MBaud). CYP15G0403DXB refers to devices not compliant to SMPTE 259M and SMPTE 292M pathological test requirements and also OBSAI RP3 operating data rate of 1536 MBaud. CYP(V)(W)15G0403DXB refers to all three devices.

The CYW15G0403DXB<sup>[1]</sup> operates from 195 to 1540 MBaud, which includes operation at the OBSAI RP3 datarate of both 1536 MBaud and 768 MBaud.

The CYV15G0403DXB satisfies the SMPTE-259M and SMPTE-292M compliance as per SMPTE EG34-1999 Pathological Test Requirements.

As a second-generation HOTLink device, the CYP(V)(W)15G0403DXB extends the HOTLink family with enhanced levels of integration and faster data rates, while maintaining serial-link compatibility (data, command, and BIST) with other HOTLink devices. The transmit (TX) section of the CYP(V)(W)15G0403DXB Quad HOTLink II consists of four independent byte-wide channels. Each channel can accept either 8-bit data characters or preencoded 10-bit transmission characters. Data characters may be passed from the Transmit Input Register to an integrated 8B/10B Encoder to improve their serial transmission characteristics. These encoded characters are then serialized and output from dual Positive ECL (PECL) compatible differential transmission-line drivers at a bit-rate of either 10- or 20-times the input reference clock for that channel.

The receive (RX) section of the CYP(V)(W)15G0403DXB Quad HOTLink II consists of four independent byte-wide channels. Each channel accepts a serial bit-stream from one of two PECL-compatible differential line receivers, and using a completely integrated Clock and Data Recovery PLL,

recovers the timing information necessary for data reconstruction. Each recovered bit-stream is deserialized and framed into characters, 8B/10B decoded, and checked for transmission errors. Recovered decoded characters are then written to an internal Elasticity Buffer, and presented to the destination host system.

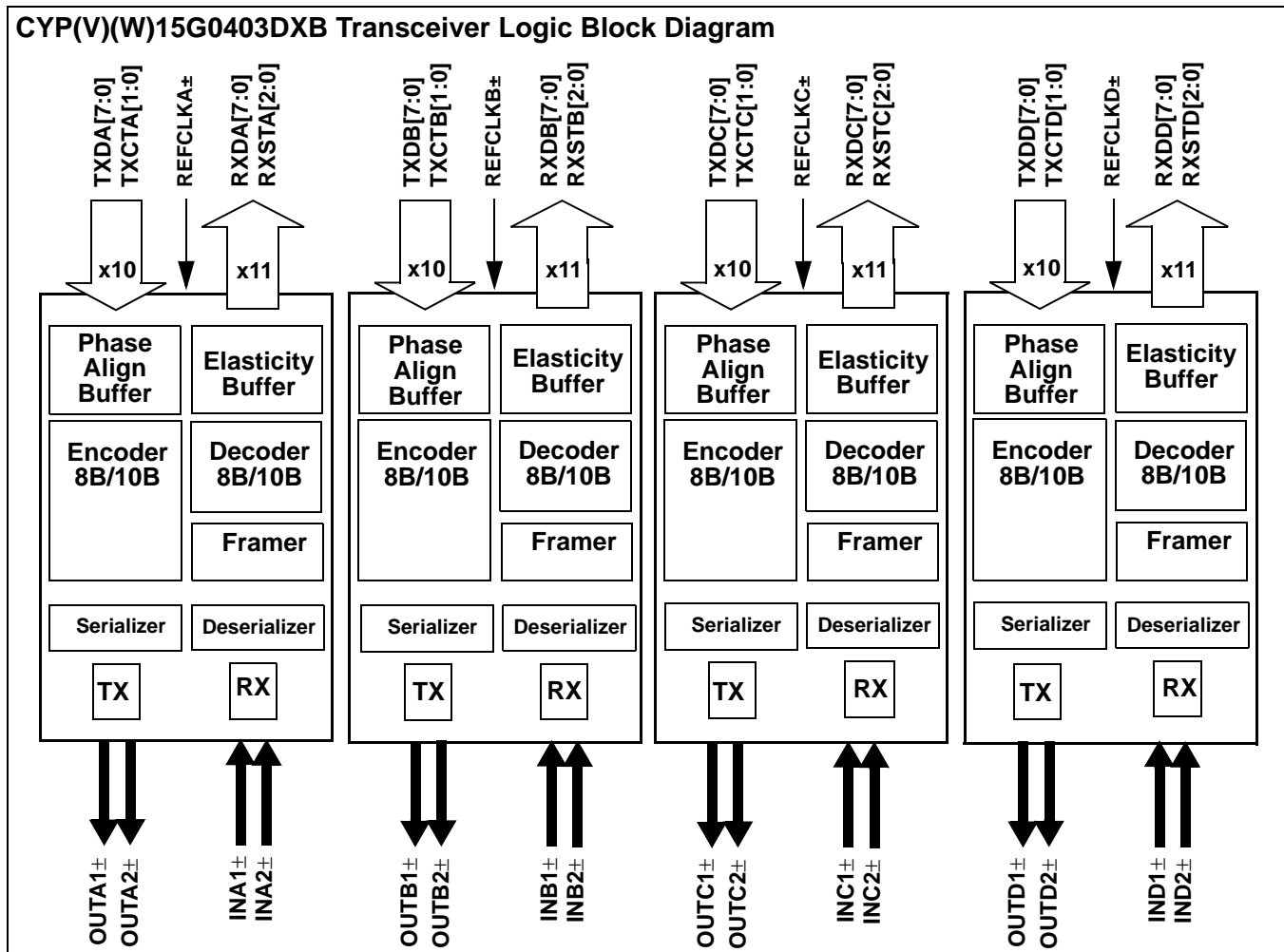
The integrated 8B/10B encoder/decoder may be bypassed for systems that present externally encoded or scrambled data at the parallel interface.

The parallel I/O interface may be configured for numerous forms of clocking to provide the highest flexibility in system architecture. In addition to clocking the transmit path with a local reference clock, the receive interface may also be configured to present data relative to a recovered clock or to a local reference clock.

Each transmit and receive channel contains an independent BIST pattern generator and checker. This BIST hardware allows at-speed testing of the high-speed serial data paths in each transmit and receive section, and across the interconnecting links.

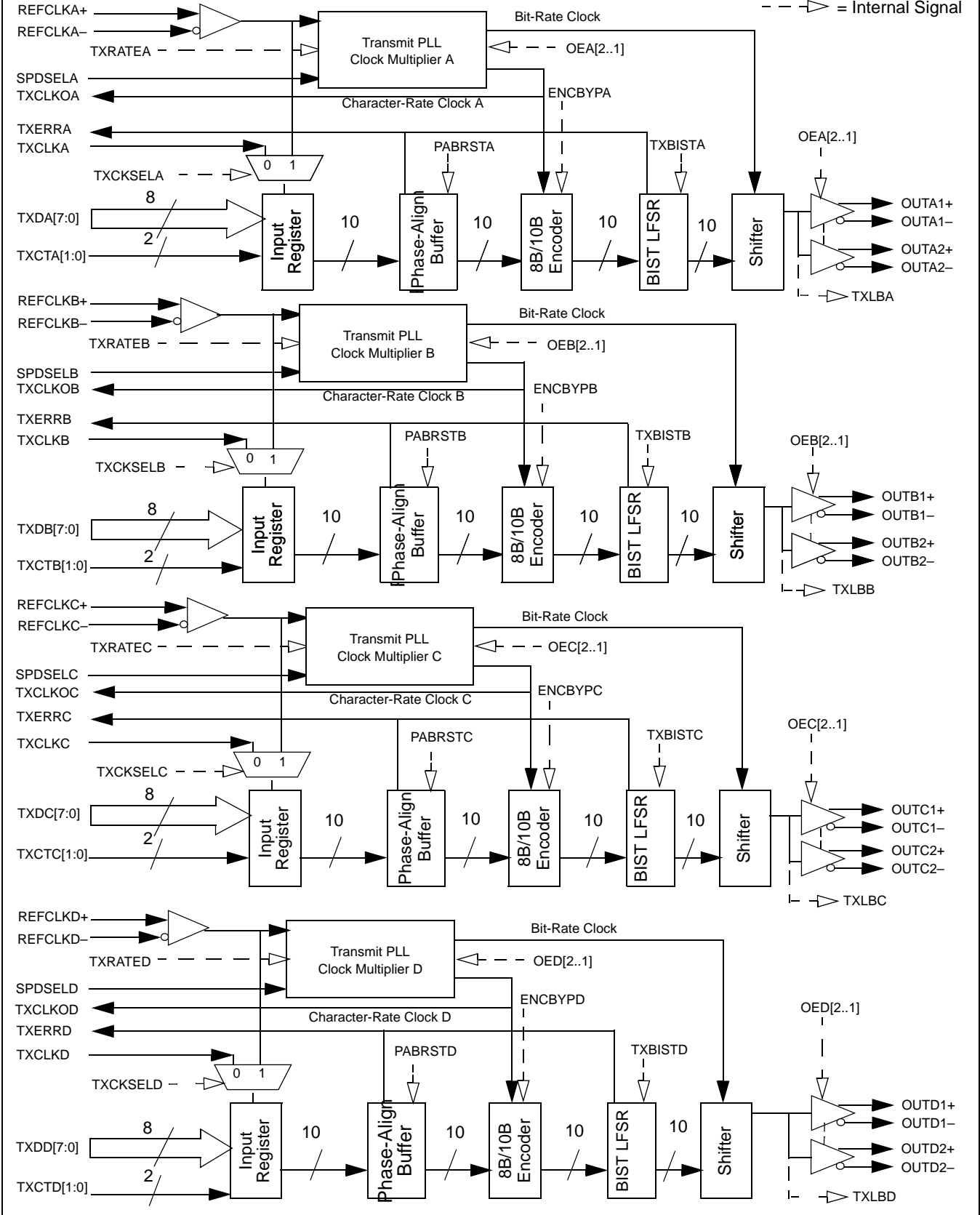
The CYP(V)(W)15G0403DXB is ideal for port applications where different data rates and serial interface standards are necessary for each channel. Some applications include multi-protocol routers, aggregation equipment, and switches.

**CYP(V)(W)15G0403DXB Transceiver Logic Block Diagram**



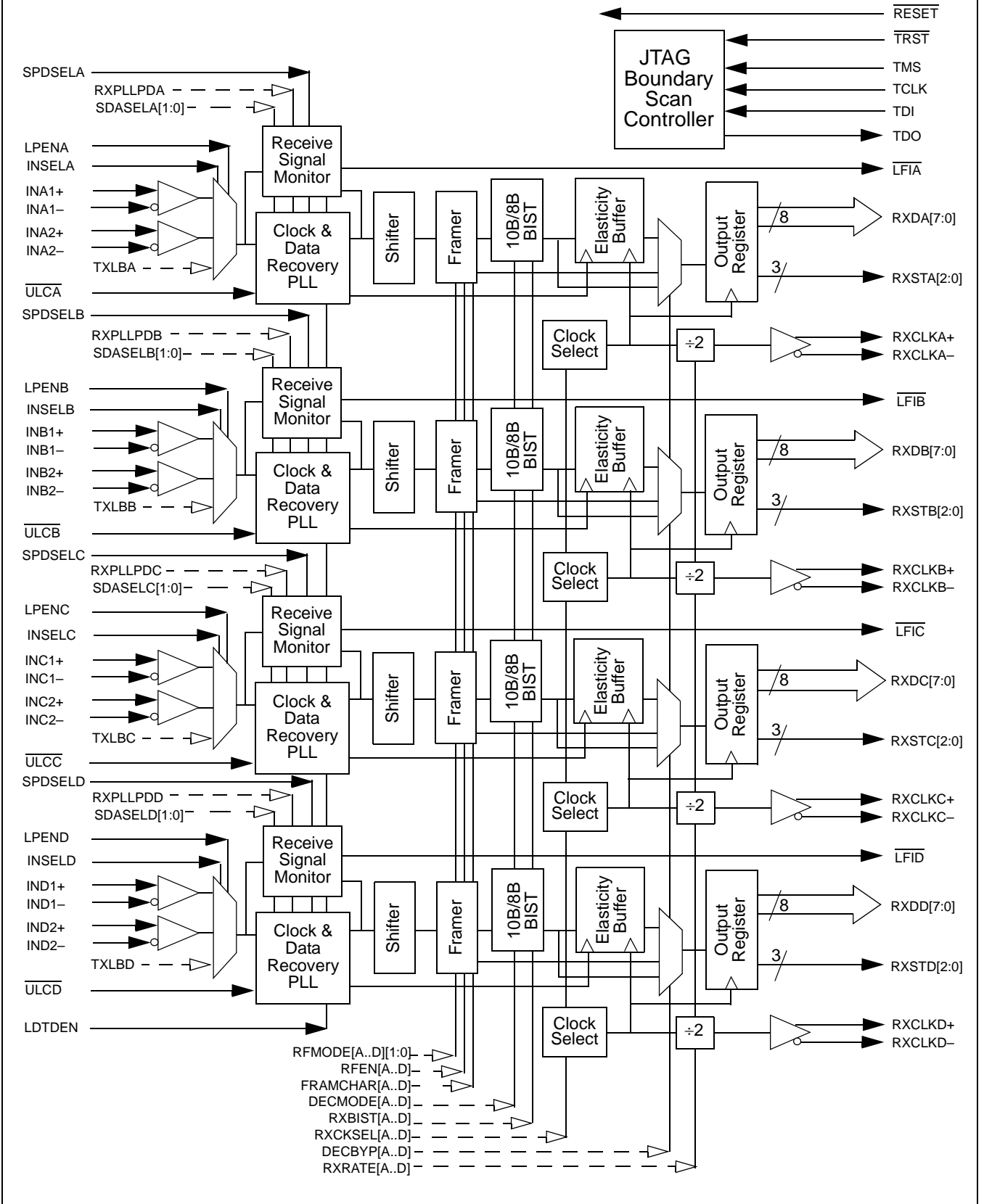
### Transmit Path Block Diagram

TXLB[A..D] are Internal Serial Loopback Signals  
 ---▷ = Internal Signal



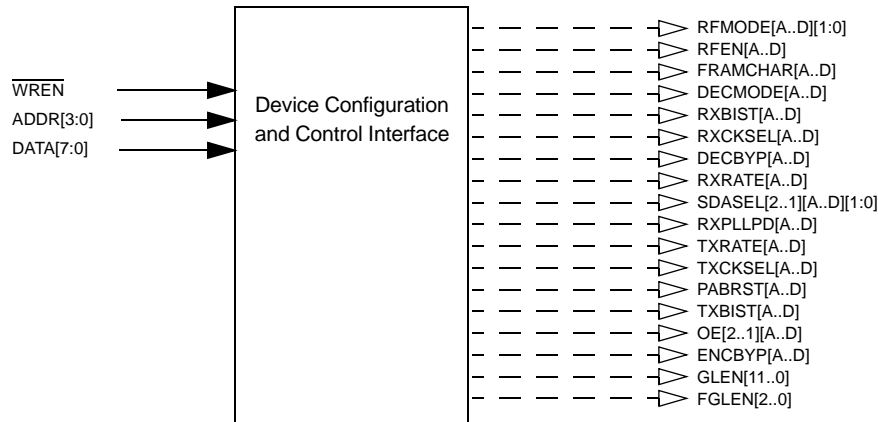
**Receive Path Block Diagram**

TXLB[A..D] are Internal Serial Loopback Signals  
 - - - ▷ = Internal Signal



**Device Configuration and Control Block Diagram**

-- &gt; = Internal Signal


**Pin Configuration (Top View)**

|   | 1               | 2               | 3                | 4               | 5               | 6        | 7         | 8   | 9         | 10       | 11        | 12       | 13  | 14       | 15        | 16              | 17               | 18              | 19              | 20               |
|---|-----------------|-----------------|------------------|-----------------|-----------------|----------|-----------|-----|-----------|----------|-----------|----------|-----|----------|-----------|-----------------|------------------|-----------------|-----------------|------------------|
| A | IN C1-          | OUT C1-         | IN C2-           | OUT C2-         | V <sub>CC</sub> | IN D1-   | OUT D1-   | GND | IN D2-    | OUT D2-  | IN A1-    | OUT A1-  | GND | IN A2-   | OUT A2-   | V <sub>CC</sub> | IN B1-           | OUT B1-         | IN B2-          | OUT B2-          |
| B | IN C1+          | OUT C1+         | IN C2+           | OUT C2+         | V <sub>CC</sub> | IN D1+   | OUT D1+   | GND | IN D2+    | OUT D2+  | IN A1+    | OUT A1+  | GND | IN A2+   | OUT A2+   | V <sub>CC</sub> | IN B1+           | OUT B1+         | IN B2+          | OUT B2+          |
| C | TDI             | TMS             | INSEL            | INSELB          | V <sub>CC</sub> | ULCD     | ULCC      | GND | DATA [7]  | DATA [5] | DATA [3]  | DATA [1] | GND | NC       | SPD SELD  | V <sub>CC</sub> | LDTD EN          | TRST            | LPEND           | TDO              |
| D | TCLK            | RESET           | INSEL            | INSELA          | V <sub>CC</sub> | ULCA     | SPD SELC  | GND | DATA [6]  | DATA [4] | DATA [2]  | DATA [0] | GND | LPENB    | ULCB      | V <sub>CC</sub> | LPENA            | LTEN1           | SCAN EN2        | TMEN3            |
| E | V <sub>CC</sub> | V <sub>CC</sub> | V <sub>CC</sub>  | V <sub>CC</sub> |                 |          |           |     |           |          |           |          |     |          |           |                 | V <sub>CC</sub>  | V <sub>CC</sub> | V <sub>CC</sub> | V <sub>CC</sub>  |
| F | RX DC[6]        | RX DC[7]        | TX DC[0]         | NC              |                 |          |           |     |           |          |           |          |     |          |           |                 | NC               | RX STB[1]       | TX CLKOB        | RX STB[0]        |
| G | TX DC[7]        | WREN            | TX DC[4]         | TX DC[1]        |                 |          |           |     |           |          |           |          |     |          |           |                 | SPD SELB         | LP ENC          | SPD SELA        | RX DB[1]         |
| H | GND             | GND             | GND              | GND             |                 |          |           |     |           |          |           |          |     |          |           |                 | GND              | GND             | GND             | GND              |
| J | TX CTC[1]       | TX DC[5]        | TX DC[2]         | TX DC[3]        |                 |          |           |     |           |          |           |          |     |          |           |                 | RX STB[2]        | RX DB[0]        | RX DB[5]        | RX DB[2]         |
| K | RX DC[2]        | REF CLKC-       | TX CTC[0]        | TX CLKC         |                 |          |           |     |           |          |           |          |     |          |           |                 | RX DB[3]         | RX DB[4]        | RX DB[7]        | LFI <sub>B</sub> |
| L | RX DC[3]        | REF CLKC+       | LFI <sub>C</sub> | TX DC[6]        |                 |          |           |     |           |          |           |          |     |          |           |                 | RX DB[6]         | RX CLKB+        | RX CLKB-        | TX DB[6]         |
| M | RX DC[4]        | RX DC[5]        | NC               | TX ERRC         |                 |          |           |     |           |          |           |          |     |          |           |                 | REF CLKB+        | REF CLKB-       | TX ERRB         | TX CLKB          |
| N | GND             | GND             | GND              | GND             |                 |          |           |     |           |          |           |          |     |          |           |                 | GND              | GND             | GND             | GND              |
| P | RX DC[1]        | RX DC[0]        | RX STC[0]        | RX STC[1]       |                 |          |           |     |           |          |           |          |     |          |           |                 | TX DB[5]         | TX DB[4]        | TX DB[3]        | TX DB[2]         |
| R | RX STC[2]       | TX CLKOC        | RX CLKC+         | RX CLKC-        |                 |          |           |     |           |          |           |          |     |          |           |                 | TX DB[1]         | TX DB[0]        | TX CTB[1]       | TX DB[7]         |
| T | V <sub>CC</sub> | V <sub>CC</sub> | V <sub>CC</sub>  | V <sub>CC</sub> |                 |          |           |     |           |          |           |          |     |          |           |                 | V <sub>CC</sub>  | V <sub>CC</sub> | V <sub>CC</sub> | V <sub>CC</sub>  |
| U | TX DD[0]        | TX DD[1]        | TX DD[2]         | TX CTD[1]       | V <sub>CC</sub> | RX DD[2] | RX DD[1]  | GND | TX CTA[1] | ADDR [0] | REF CLKD- | TX DA[1] | GND | TX DA[4] | TX CTA[0] | V <sub>CC</sub> | RX DA[2]         | TX CTB[0]       | RX STA[2]       | RX STA[1]        |
| V | TX DD[3]        | TX DD[4]        | TX CTD[0]        | RX DD[6]        | V <sub>CC</sub> | RX DD[3] | RX STD[0] | GND | RX STD[2] | ADDR [2] | REF CLKD+ | TX CLKOA | GND | TX DA[3] | TX DA[7]  | V <sub>CC</sub> | RX DA[7]         | RX DA[3]        | RX DA[0]        | RX STA[0]        |
| W | TX DD[5]        | TX DD[7]        | LFI <sub>D</sub> | RX CLKD-        | V <sub>CC</sub> | RX DD[4] | RX STD[1] | GND | ADDR [3]  | ADDR [1] | RX CLKA+  | TX ERRA  | GND | TX DA[2] | TX DA[6]  | V <sub>CC</sub> | LFI <sub>A</sub> | REF CLKA+       | RX DA[4]        | RX DA[1]         |
| Y | TX DD[6]        | TX CLKD         | RX DD[7]         | RX CLKD+        | V <sub>CC</sub> | RX DD[5] | RX DD[0]  | GND | TX CLKOD  | NC       | TX CLKA   | RX CLKA- | GND | TX DA[0] | TX DA[5]  | V <sub>CC</sub> | TX ERRD          | REF CLKA-       | RX DA[6]        | RX DA[5]         |



Pin Configuration (Bottom View)

|   | 20                       | 19              | 18                       | 17                       | 16              | 15                       | 14       | 13  | 12       | 11        | 10       | 9         | 8   | 7                        | 6                        | 5               | 4               | 3                        | 2                         | 1               |
|---|--------------------------|-----------------|--------------------------|--------------------------|-----------------|--------------------------|----------|-----|----------|-----------|----------|-----------|-----|--------------------------|--------------------------|-----------------|-----------------|--------------------------|---------------------------|-----------------|
| A | OUT B2-                  | IN B2-          | OUT B1-                  | IN B1-                   | V <sub>CC</sub> | OUT A2-                  | IN A2-   | GND | OUT A1-  | IN A1-    | OUT D2-  | IN D2-    | GND | OUT D1-                  | IN D1-                   | V <sub>CC</sub> | OUT C2-         | IN C2-                   | OUT C1-                   | IN C1-          |
| B | OUT B2+                  | IN B2+          | OUT B1+                  | IN B1+                   | V <sub>CC</sub> | OUT A2+                  | IN A2+   | GND | OUT A1+  | IN A1+    | OUT D2+  | IN D2+    | GND | OUT D1+                  | IN D1+                   | V <sub>CC</sub> | OUT C2+         | IN C2+                   | OUT C1+                   | IN C1+          |
| C | TDO                      | LP END          | $\overline{\text{TRST}}$ | LDTD EN                  | V <sub>CC</sub> | SPD SELD                 | NC       | GND | DATA [1] | DATA [3]  | DATA [5] | DATA [7]  | GND | $\overline{\text{ULCC}}$ | $\overline{\text{ULCD}}$ | V <sub>CC</sub> | IN SELB         | IN SELC                  | TMS                       | TDI             |
| D | TMEN3                    | SCAN EN2        | LTEN1                    | LP ENA                   | V <sub>CC</sub> | $\overline{\text{ULCB}}$ | LP ENB   | GND | DATA [0] | DATA [2]  | DATA [4] | DATA [6]  | GND | SPD SELC                 | $\overline{\text{ULCA}}$ | V <sub>CC</sub> | IN SELA         | IN SELD                  | $\overline{\text{RESET}}$ | TCLK            |
| E | V <sub>CC</sub>          | V <sub>CC</sub> | V <sub>CC</sub>          | V <sub>CC</sub>          |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | V <sub>CC</sub> | V <sub>CC</sub>          | V <sub>CC</sub>           | V <sub>CC</sub> |
| F | RX STB[0]                | TX CLKOB        | RX STB[1]                | NC                       |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | NC              | TX DC[0]                 | RX DC[7]                  | Rx DC[6]        |
| G | RX DB[1]                 | SPD SELA        | LP ENC                   | SPD SELB                 |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | TX DC[1]        | TX DC[4]                 | $\overline{\text{WREN}}$  | TX DC[7]        |
| H | GND                      | GND             | GND                      | GND                      |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | GND             | GND                      | GND                       | GND             |
| J | RX DB[2]                 | RX DB[5]        | RX DB[0]                 | RX STB[2]                |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | TX DC[3]        | TX DC[2]                 | TX DC[5]                  | TX CTC[1]       |
| K | $\overline{\text{LFIB}}$ | RX DB[7]        | RX DB[4]                 | RX DB[3]                 |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | TX CLKC         | TX CTC[0]                | REF CLKC-                 | RX DC[2]        |
| L | TX DB[6]                 | RX CLKB-        | RX CLKB+                 | RX DB[6]                 |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | TX DC[6]        | $\overline{\text{LFIC}}$ | REF CLKC+                 | RX DC[3]        |
| M | TX CLKB                  | TX ERRB         | REF CLKB-                | REF CLKB+                |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | TX ERRC         | NC                       | RX DC[5]                  | RX DC[4]        |
| N | GND                      | GND             | GND                      | GND                      |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | GND             | GND                      | GND                       | GND             |
| P | TX DB[2]                 | TX DB[3]        | TX DB[4]                 | TX DB[5]                 |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | RX STC[1]       | RX STC[0]                | RX DC[0]                  | RX DC[1]        |
| R | TX DB[7]                 | TX CTB[1]       | TX DB[0]                 | TX DB[1]                 |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | RX CLKC-        | RX CLKC+                 | TX CLKOC                  | RX STC[2]       |
| T | V <sub>CC</sub>          | V <sub>CC</sub> | V <sub>CC</sub>          | V <sub>CC</sub>          |                 |                          |          |     |          |           |          |           |     |                          |                          |                 | V <sub>CC</sub> | V <sub>CC</sub>          | V <sub>CC</sub>           | V <sub>CC</sub> |
| U | RX STA[1]                | RX STA[2]       | TX CTB[0]                | RX DA[2]                 | V <sub>CC</sub> | TX CTA[0]                | TX DA[4] | GND | TX DA[1] | REF CLKD- | ADDR [0] | TXC TA[1] | GND | RX DD[1]                 | RX DD[2]                 | V <sub>CC</sub> | TX CTD[1]       | TX DD[2]                 | TX DD[1]                  | TX DD[0]        |
| V | RX STA[0]                | RX DA[0]        | RX DA[3]                 | RX DA[7]                 | V <sub>CC</sub> | TX DA[7]                 | TX DA[3] | GND | TX CLKOA | REF CLKD+ | ADDR [2] | RX STD[2] | GND | RX STD[0]                | RX DD[3]                 | V <sub>CC</sub> | RX DD[6]        | TX CTD[0]                | TX DD[4]                  | TX DD[3]        |
| W | RX DA[1]                 | RX DA[4]        | REF CLKA+                | $\overline{\text{LFIA}}$ | V <sub>CC</sub> | TX DA[6]                 | TX DA[2] | GND | TX ERRA  | RX CLKA+  | ADDR [1] | ADDR [3]  | GND | RX STD[1]                | RX DD[4]                 | V <sub>CC</sub> | RX CLKD-        | $\overline{\text{LFID}}$ | TX DD[7]                  | TX DD[5]        |
| Y | RX DA[5]                 | RX DA[6]        | REF CLKA-                | TX ERRD                  | V <sub>CC</sub> | TX DA[5]                 | TX DA[0] | GND | RX CLKA- | TX CLKA   | NC       | TX CLKOD  | GND | RX DD[0]                 | RX DD[5]                 | V <sub>CC</sub> | RX CLKD+        | RX DD[7]                 | TX CLKD                   | TX DD[6]        |



**Pin Descriptions**  
**CYP(V)(W)15G0403DXB Quad HOTLink II Transceiver**

| Name   | I/O Characteristics   | Signal Description   |
|--|---|--|
| <b>Transmit Path Data and Status Signals</b>   |   |  |
| TXDA[7:0]<br>TXDB[7:0]<br>TXDC[7:0]<br>TXDD[7:0]   | LVTTTL Input, synchronous, sampled by the associated TXCLKx <sup>↑</sup> or REFCLKx <sup>↑</sup> [2]  | <b>Transmit Data Inputs.</b> TXDx[7:0] data inputs are captured on the rising edge of the transmit interface clock. The transmit interface clock is selected by the TXCKSELx latch via the device configuration interface, and passed to the encoder or Transmit Shifter. When the Encoder is enabled, TXDx[7:0] specifies the specific data or command character sent.  |
| TXCTA[1:0]<br>TXCTB[1:0]<br>TXCTC[1:0]<br>TXCTD[1:0]   | LVTTTL Input, synchronous, sampled by the associated TXCLKx <sup>↑</sup> or REFCLKx <sup>↑</sup> [2]  | <b>Transmit Control.</b> TXCTx[1:0] inputs are captured on the rising edge of the transmit interface clock. The transmit interface clock is selected by the TXCKSELx latch via the device configuration interface, and passed to the Encoder or Transmit Shifter. The TXCTA[1:0] inputs identify how the associated TXDx[7:0] characters are interpreted. When the Encoder is bypassed, these inputs are interpreted as data bits. When the Encoder is enabled, these inputs determine if the TXDx[7:0] character is encoded as Data, a Special Character code, or replaced with other Special Character codes. See <i>Table 3</i> for details.  |
| TXERRA<br>TXERRB<br>TXERRC<br>TXERRD   | LVTTTL Output, synchronous to REFCLKx <sup>↑</sup> [3], synchronous to RXCLKx when selected as REFCLKx, asynchronous to transmit channel enable / disable, asynchronous to loss or return of REFCLKx <sup>±</sup> | <b>Transmit Path Error.</b> TXERRx is asserted HIGH to indicate detection of a transmit Phase-Align Buffer underflow or overflow. If an underflow or overflow condition is detected, TXERRx, for the channel in error, is asserted HIGH and remains asserted until either a Word Sync Sequence is transmitted on that channel, or the transmit Phase-Align Buffer is re-centered with the PABRSTx latch via the device configuration interface. When TXBISTx = 0, the BIST progress is presented on the associated TXERRx output. The TXERRx signal pulses HIGH for one transmit-character clock period to indicate a pass through the BIST sequence once every 511 or 527 (depending on RXCKSELx) character times. If RXCKSELx = 1, a one character pulse occurs every 527 character times. If RXCKSELx = 0, a one character pulse occurs every 511 character times.<br><br>TXERRx is also asserted HIGH, when any of the following conditions is true: <ul style="list-style-type: none"> <li>• The TXPLL for the associated channel is powered down. This occurs when OE2x and OE1x for a given channel are both disabled by setting OE2x = 0 and OE1x = 0.</li> <li>• The absence of the REFCLKx<sup>±</sup> signal</li> </ul> |
| <b>Transmit Path Clock Signals</b>   |   |  |
| REFCLKA <sup>±</sup><br>REFCLKB <sup>±</sup><br>REFCLKC <sup>±</sup><br>REFCLKD <sup>±</sup> | Differential LVPECL or single-ended LVTTTL input clock  | <b>Reference Clock.</b> REFCLKx <sup>±</sup> clock inputs are used as the timing references for the transmit and receive PLLs. These input clocks may also be selected to clock the transmit and receive parallel interfaces. When driven by a single-ended LVCMOS or LVTTTL clock source, connect the clock source to either the true or complement REFCLKx input, and leave the alternate REFCLKx input open (floating). When driven by an LVPECL clock source, the clock must be a differential clock, using both inputs.   |
| TXCLKA<br>TXCLKB<br>TXCLKC<br>TXCLKD   | LVTTTL Clock Input, internal pull-down  | <b>Transmit Path Input Clock.</b> When configuration latch TXCKSELx = 0, the associated TXCLKx input is selected as the character-rate input clock for the TXDx[7:0] and TXCTx[1:0] inputs. In this mode, the TXCLKx input must be frequency-coherent to its associated TXCLKOx output clock, but may be offset in phase by any amount. Once initialized, TXCLKx is allowed to drift in phase as much as ±180 degrees. If the input phase of TXCLKx drifts beyond the handling capacity of the Phase Align Buffer, TXERRx is asserted to indicate the loss of data, and remains asserted until the Phase Align Buffer is initialized. The phase of the TXCLKx input clock relative to its associated REFCLKx <sup>±</sup> is initialized when the configuration latch PABRSTx is written as 0. When the associated TXERRx is deasserted, the Phase Align Buffer is initialized and input characters are correctly captured.  |

**Notes:**

2. When REFCLKx<sup>±</sup> is configured for half-rate operation, these inputs are sampled relative to both the rising and falling edges of the associated REFCLKx<sup>±</sup>.
3. When REFCLKx<sup>±</sup> is configured for half-rate operation, these outputs are presented relative to both the rising and falling edges of the associated REFCLKx<sup>±</sup>.



**Pin Descriptions** (continued)  
**CYP(V)(W)15G0403DXB Quad HOTLink II Transceiver**

| Name   | I/O Characteristics  | Signal Description   |
|--|--|--|
| TXCLKOA<br>TXCLKOB<br>TXCLKOC<br>TXCLKOD             | LVTTTL Output  | <b>Transmit Clock Output.</b> TXCLKOx output clock is synthesized by each channel's transmit PLL and operates synchronous to the internal transmit character clock. TXCLKOx operates at either the same frequency as REFCLKx± (TXRATEx = 0), or at twice the frequency of REFCLKx± (TXRATEx = 1). The transmit clock outputs have no fixed phase relationship to REFCLKx±.   |
| <b>Receive Path Data and Status Signals</b>          |  |  |
| RXDA[7:0]<br>RXDB[7:0]<br>RXDC[7:0]<br>RXDD[7:0]     | LVTTTL Output, synchronous to the selected RXCLK± output or REFCLKx± input | <b>Parallel Data Output.</b> RXDx[7:0] parallel data outputs change relative to the receive interface clock. The receive interface clock is selected by the RXCKSELx latch. If RXCLKx± is a full-rate clock, the RXCLKx± clock outputs are complementary clocks operating at the character rate. The RXDx[7:0] outputs for the associated receive channels follow rising edge of RXCLKx+ or falling edge of RXCLKx-. If RXCLKx± is a half-rate clock, the RXCLKx± clock outputs are complementary clocks operating at half the character rate. The RXDx[7:0] outputs for the associated receive channels follow both the falling and rising edges of the associated RXCLKx± clock outputs.   |
| RXSTA[2:0]<br>RXSTB[2:0]<br>RXSTC[2:0]<br>RXSTD[2:0] | LVTTTL Output, synchronous to the selected RXCLK± output or REFCLKx± input | <b>Parallel Status Output.</b> RXSTA[2:0] status outputs change relative to the receive interface clock. The receive interface clock is selected by the RXCKSELx latch. If RXCLKx± is a full-rate clock, the RXCLKx± clock outputs are complementary clocks operating at the character rate. The RXSTAx[2:0] outputs for the associated receive channels follow rising edge of RXCLKx+ or falling edge of RXCLKx-. If RXCLKx± is a half-rate clock, the RXCLKx± clock outputs are complementary clocks operating at half the character rate. The RXSTAx[2:0] outputs for the associated receive channels follow both the falling and rising edges of the associated RXCLKx± clock outputs. When the decoder is bypassed, RXSTx[1:0] become the two low-order bits of the 10-bit received character. RXSTx[2] = HIGH indicates the presence of a Comma character in the Output Register. When the decoder is enabled, RXSTx[2:0] provide status of the received signal. See <i>Table 11</i> for a list of received character status.  |
| <b>Receive Path Clock Signals</b>                    |  |  |
| RXCLKA±<br>RXCLKB±<br>RXCLKC±<br>RXCLKD±             | LVTTTL Output Clock  | <b>Receive Clock Output.</b> RXCLKx± is the receive interface clock used to control timing of the RXDx[7:0] and RXSTA[2:0] parallel outputs. The source of the RXCLKx± outputs is selected by the RXCKSELx latch via the device configuration interface. These true and complement clocks are used to control timing of data output transfers. These clocks are output continuously at either the dual-character rate (1/20 <sup>th</sup> the serial bit-rate) or character rate (1/10 <sup>th</sup> the serial bit-rate) of the data being received, as selected by RXRATEx. When configured such that the output data path is clocked by the REFCLKx± instead of a recovered clock, the RXCLKx± output drivers present a buffered or divided form (depending on RXRATEx) of the associated REFCLKx± that are delayed in phase to align with the data. This phase difference allows the user to select the optimal clock (REFCLKx± or RXCLK±) for setup/hold timing for their specific system.<br><br>When REFCLKx± is a full-rate clock, the RXCLKx± rate depends on the value of RXRATEx.<br><br>When REFCLKx± is a half-rate clock and RXCKSELx = 0, the RXCLKx± rate depends on the value of RXRATEx.<br><br>When REFCLKx± is a half-rate clock and RXCKSELx = 1, the RXCLKx± rate does not depend on the value of RXRATEx and operates at the same rate as REFCLKx±. |
| <b>Device Control Signals</b>                        |  |  |
| RESET  | LVTTTL Input, asynchronous, internal pull-up                               | <b>Asynchronous Device Reset.</b> RESET initializes all <u>state</u> machines, counters, and configuration latches in the device to a known state. RESET must be asserted LOW for a minimum pulse width. When the reset is removed, all state machines, counters and configuration latches are at an initial state. See <i>Table 9</i> for the initialize values of the device configuration latches.  |





Pin Descriptions (continued)  
 CYP(V)15G0403DXB Quad HOTLink II Transceiver

| Name   | I/O Characteristics                                | Signal Description   |
|--|--|--|
| LDTDEN   | LVTTL Input, internal pull-up                      | <b>Level Detect Transition Density Enable.</b> When LDTDEN is HIGH, the Signal Level Detector, Range Controller, and Transition Density Detector are all enabled to determine if the RXPLL tracks REFCLKx± or the selected input serial data stream. If the Signal Level Detector, Range Controller, or Transition Density Detector are out of their respective limits while LDTDEN is HIGH, the RXPLL locks to REFCLK± until such a time they become valid. The (SDASEL[A..D][1:0]) are used to configure the trip level of the Signal Level Detector. The Transition Density Detector limit is one transition in every 60 consecutive bits. When LDTDEN is LOW, only the Range Controller is used to determine if the RXPLL tracks REFCLKx± or the selected input serial data stream. For the cases when RXCKSELx = 0 (recovered clock), it is recommended to set LDTDEN = HIGH. |
| <u>ULCA</u><br><u>ULCB</u><br><u>ULCC</u><br><u>ULCD</u> | LVTTL Input, internal pull-up                      | <b>Use Local Clock.</b> When <u>ULCx</u> is LOW, the RXPLL is forced to lock to REFCLKx± instead of the received serial data stream. While <u>ULCx</u> is LOW, the <u>LFix</u> for the associated channel is LOW indicating a link fault.<br><br>When <u>ULCx</u> is HIGH, the RXPLL performs Clock and Data Recovery functions on the input data streams. This function is used in applications in which a stable RXCLKx± is needed. In cases when there is an absence of valid data transitions for a long period of time, or the high-gain differential serial inputs (INx±) are left floating, there may be brief frequency excursions of the RXCLKx± outputs from REFCLKx±.   |
| SPDSELA<br>SPDSELB<br>SPDSELC<br>SPDSELD                 | 3-Level Select <sup>[4]</sup> static control input | <b>Serial Rate Select.</b> The SPDSELx inputs specify the operating signaling-rate range of each channel's transmit and receive PLL.<br><br>LOW = 195 – 400 MBaud<br>MID = 400 – 800 MBaud<br>HIGH = 800 – 1500 MBaud (800–1540 MBaud for CYW15G0403DXB)   |
| INSELA<br>INSELB<br>INSELC<br>INSELD                     | LVTTL Input, asynchronous                          | <b>Receive Input Selector.</b> The INSELx input determines which external serial bit stream is passed to the receiver's Clock and Data Recovery circuit. When INSELx is HIGH, the Primary Differential Serial Data Input, INx1±, is selected for the associated receive channel. When INSELx is LOW, the Secondary Differential Serial Data Input, INx2±, is selected for the associated receive channel.  |
| LPENA<br>LPENB<br>LPENC<br>LPEND                         | LVTTL Input, asynchronous, internal pull-down      | <b>Loop-Back-Enable.</b> The LPENx input enables the internal serial loop-back for the associated channel. When LPENx is HIGH, the transmit serial data from the associated channel is internally routed to the associated receive Clock and Data Recovery (CDR) circuit. All enabled serial drivers on the channel are forced to differential logic-1, and the serial data inputs are ignored. When LPENx is LOW, the internal serial loop-back function is disabled.   |
| <u>LFIA</u><br><u>LFIB</u><br><u>LFIC</u><br><u>LFID</u> | LVTTL Output, asynchronous                         | <b>Link Fault Indication Output.</b> <u>LFix</u> is an output status indicator signal. <u>LFix</u> is the logical OR of six internal conditions. <u>LFix</u> is asserted LOW when any of the following conditions is true: <ul style="list-style-type: none"> <li>• Received serial data rate outside expected range</li> <li>• Analog amplitude below expected levels</li> <li>• Transition density lower than expected</li> <li>• Receive channel disabled</li> <li>• <u>ULCx</u> is LOW</li> <li>• Absence of REFCLKx±.</li> </ul>  |
| <b>Device Configuration and Control Bus Signals</b>      |  |  |
| <u>WREN</u>  | LVTTL input, asynchronous, internal pull-up        | <b>Control Write Enable.</b> The <u>WREN</u> input writes the values of the DATA[7:0] bus into the latch specified by the address location on the ADDR[3:0] bus. <sup>[5]</sup>  |

Notes:

4. 3-Level Select inputs are used for static configuration. These are ternary inputs that make use of logic levels of LOW, MID, and HIGH. The LOW level is usually implemented by direct connection to V<sub>SS</sub> (ground). The HIGH level is usually implemented by direct connection to V<sub>CC</sub> (power). The MID level is usually implemented by not connecting the input (left floating), which allows it to self bias to the proper level.
5. See *Device Configuration and Control Interface* for detailed information on the operation of the Configuration Interface.



**Pin Descriptions** (continued)  
**CYP(V)(W)15G0403DXB Quad HOTLink II Transceiver**

| Name   | I/O Characteristics                        | Signal Description   |
|--|--|--|
| ADDR[3:0]                                    | LVTTL input asynchronous, internal pull-up | <b>Control Addressing Bus.</b> The ADDR[3:0] bus is the input address bus used to configure the device. The WREN input writes the values of the DATA[7:0] bus into the latch specified by the address location on the ADDR[3:0] bus. <sup>[5]</sup> Table 9 lists the configuration latches within the device, and the initialization value of the latches upon the assertion of RESET. Table 10 shows how the latches are mapped in the device. |
| DATA[7:0]                                    | LVTTL input asynchronous, internal pull-up | <b>Control Data Bus.</b> The DATA[7:0] bus is the input data bus used to configure the device. The WREN input writes the values of the DATA[7:0] bus into the latch specified by address location on the ADDR[3:0] bus. <sup>[5]</sup> Table 9 lists the configuration latches within the device, and the initialization value of the latches upon the assertion of RESET. Table 10 shows how the latches are mapped in the device.              |
| <b>Internal Device Configuration Latches</b> |  |  |
| RFMODE[A..D][1:0]                            | Internal Latch <sup>[6]</sup>              | <b>Reframe Mode Select.</b>  |
| FRAMCHAR[A..D]                               | Internal Latch <sup>[6]</sup>              | <b>Framing Character Select.</b>   |
| DECMODE[A..D]                                | Internal Latch <sup>[6]</sup>              | <b>Receiver Decoder Mode Select.</b>   |
| DECBYP[A..D]                                 | Internal Latch <sup>[6]</sup>              | <b>Receiver Decoder Bypass.</b>  |
| RXCKSEL[A..D]                                | Internal Latch <sup>[6]</sup>              | <b>Receive Clock Select.</b>   |
| RXRATE[A..D]                                 | Internal Latch <sup>[6]</sup>              | <b>Receive Clock Rate Select.</b>  |
| SDASEL[A..D][1:0]                            | Internal Latch <sup>[6]</sup>              | <b>Signal Detect Amplitude Select.</b>   |
| ENCBYP[A..D]                                 | Internal Latch <sup>[6]</sup>              | <b>Transmit Encoder Bypassed.</b>  |
| TXCKSEL[A..D]                                | Internal Latch <sup>[6]</sup>              | <b>Transmit Clock Select.</b>  |
| TXRATE[A..D]                                 | Internal Latch <sup>[6]</sup>              | <b>Transmit PLL Clock Rate Select.</b>   |
| RFEN[A..D]                                   | Internal Latch <sup>[6]</sup>              | <b>Reframe Enable.</b>   |
| RXPLLPD[A..D]                                | Internal Latch <sup>[6]</sup>              | <b>Receive Channel Power Control.</b>  |
| RXBIST[A..D]                                 | Internal Latch <sup>[6]</sup>              | <b>Receive Bist Disabled.</b>  |
| TXBIST[A..D]                                 | Internal Latch <sup>[6]</sup>              | <b>Transmit Bist Disabled.</b>   |
| OE2[A..D]                                    | Internal Latch <sup>[6]</sup>              | <b>Differential Serial Output Driver 2 Enable.</b>   |
| OE1[A..D]                                    | Internal Latch <sup>[6]</sup>              | <b>Differential Serial Output Driver 1 Enable.</b>   |
| PABRST[A..D]                                 | Internal Latch <sup>[6]</sup>              | <b>Transmit Clock Phase Alignment Buffer Reset.</b>  |
| GLEN[11..0]                                  | Internal Latch <sup>[6]</sup>              | <b>Global Latch Enable.</b>  |
| FGLEN[2..0]                                  | Internal Latch <sup>[6]</sup>              | <b>Force Global Latch Enable.</b>  |
| <b>Factory Test Modes</b>                    |  |  |
| LTEN1  | LVTTL input, internal pull-down            | <b>Factory Test 1.</b> LTEN1 input is for factory testing only. This input may be left as a NO CONNECT, or GND only.   |
| SCANEN2                                      | LVTTL input, internal pull-down            | <b>Factory Test 2.</b> SCANEN2 input is for factory testing only. This input may be left as a NO CONNECT, or GND only.   |
| TMEN3  | LVTTL input, internal pull-down            | <b>Factory Test 3.</b> TMEN3 input is for factory testing only. This input may be left as a NO CONNECT, or GND only.   |

**Note:**

6. See *Device Configuration and Control Interface* for detailed information on the internal latches.



Pin Descriptions (continued)  
 CYP(V)(W)15G0403DXB Quad HOTLink II Transceiver

| Name                                 | I/O Characteristics             | Signal Description  |
|--------------------------------------|---------------------------------|---|
| <b>Analog I/O</b>                    |                                 |   |
| OUTA1±<br>OUTB1±<br>OUTC1±<br>OUTD1± | CML Differential Output         | <b>Primary Differential Serial Data Output.</b> The OUTx1± PECL-compatible CML outputs (+3.3V referenced) are capable of driving terminated transmission lines or standard fiber-optic transmitter modules, and must be AC-coupled for PECL-compatible connections.   |
| OUTA2±<br>OUTB2±<br>OUTC2±<br>OUTD2± | CML Differential Output         | <b>Secondary Differential Serial Data Output.</b> The OUTx2± PECL-compatible CML outputs (+3.3V referenced) are capable of driving terminated transmission lines or standard fiber-optic transmitter modules, and must be AC-coupled for PECL-compatible connections. |
| INA1±<br>INB1±<br>INC1±<br>IND1±     | Differential Input              | <b>Primary Differential Serial Data Input.</b> The INx1± input accepts the serial data stream for deserialization and decoding. The INx1± serial stream is passed to the receive CDR circuit to extract the data content when INSELx = HIGH.                          |
| INA2±<br>INB2±<br>INC2±<br>IND2±     | Differential Input              | <b>Secondary Differential Serial Data Input.</b> The INx2± input accepts the serial data stream for deserialization and decoding. The INx2± serial stream is passed to the receiver CDR circuit to extract the data content when INSELx = LOW.                        |
| <b>JTAG Interface</b>                |                                 |   |
| TMS                                  | LVTTL Input, internal pull-up   | <b>Test Mode Select.</b> Used to control access to the JTAG Test Modes. If maintained high for ≥5 TCLK cycles, the JTAG test controller is reset.   |
| TCLK                                 | LVTTL Input, internal pull-down | <b>JTAG Test Clock.</b>   |
| TDO                                  | 3-State LVTTL Output            | <b>Test Data Out.</b> JTAG data output buffer. High-Z while JTAG test mode is not selected.   |
| TDI                                  | LVTTL Input, internal pull-up   | <b>Test Data In.</b> JTAG data input port.  |
| TRST                                 | LVTTL Input, internal pull-up   | <b>JTAG reset signal.</b> When asserted (LOW), this input asynchronously resets the JTAG test access port controller.   |
| <b>Power</b>                         |                                 |   |
| V <sub>CC</sub>                      |                                 | <b>+3.3V Power.</b>   |
| GND                                  |                                 | <b>Signal and Power Ground for all internal circuits.</b>   |

**CYP(V)(W)15G0403DXB HOTLink II Operation**

The CYP(V)(W)15G0403DXB is a highly configurable, independent clocking, quad-channel transceiver designed to support reliable transfer of large quantities of data, using high-speed serial links from multiple sources to multiple destinations. This device supports four single-byte channels.

**CYP(V)(W)15G0403DXB Transmit Data Path**

*Input Register*

The bits in the Input Register for each channel support different assignments, based on if the input data is encoded or unencoded. These assignments are shown in *Table 1*.

When the ENCODER is enabled, each input register captures eight data bits and two control bits on each input clock cycle. When the Encoder is bypassed, the control bits are part of the pre-encoded 10-bit character.

When the Encoder is enabled, the TXCTx[1:0] bits are interpreted along with the associated TXDx[7:0] character to generate a specific 10-bit transmission character.

*Phase-Align Buffer*

Data from each Input Register is passed to the associated Phase-Align Buffer, when the TXDx[7:0] and TXCTx[1:0] input registers are clocked using TXCLKx! (TXCKSELx = 0 and TXRATEx = 0). When the TXDx[7:0] and TXCTx[1:0] input registers are clocked using REFCLKx± (TXCKSELx = 1) and REFCLKx± is a full-rate clock, the associated Phase Alignment Buffer in the transmit path is bypassed. These buffers are used to absorb clock phase differences between the TXCLKx input clock and the internal character clock for that channel.

Once initialized, TXCLKx is allowed to drift in phase as much as ±180 degrees. If the input phase of TXCLKx drifts beyond the handling capacity of the Phase Align Buffer, TXERRx is

asserted to indicate the loss of data, and remains asserted until the Phase Align Buffer is initialized. The phase of the TXCLKx relative to its associated internal character rate clock is initialized when the configuration latch PABRSTx is written as 0. When the associated TXERRx is deasserted, the Phase Align Buffer is initialized and input characters are correctly captured.

**Table 1. Input Register Bit Assignments<sup>[7]</sup>**

| Signal Name    | Unencoded | Encoded  |
|----------------|-----------|----------|
| TXDx[0] (LSB)  | DINx[0]   | TXDx[0]  |
| TXDx[1]        | DINx[1]   | TXDx[1]  |
| TXDx[2]        | DINx[2]   | TXDx[2]  |
| TXDx[3]        | DINx[3]   | TXDx[3]  |
| TXDx[4]        | DINx[4]   | TXDx[4]  |
| TXDx[5]        | DINx[5]   | TXDx[5]  |
| TXDx[6]        | DINx[6]   | TXDx[6]  |
| TXDx[7]        | DINx[7]   | TXDx[7]  |
| TXCTx[0]       | DINx[8]   | TXCTx[0] |
| TXCTx[1] (MSB) | DINx[9]   | TXCTx[1] |

If the phase offset, between the initialized location of the input clock and REFCLKx<sub>i</sub>, exceeds the skew handling capabilities of the Phase-Align Buffer, an error is reported on that channel's TXERRx output. This output indicates an error continuously until the Phase-Align Buffer for that channel is reset. While the error remains active, the transmitter for that channel outputs a continuous C0.7 character to indicate to the remote receiver that an error condition is present in the link.

Each Phase-Align Buffer may be individually reset with minimal disruption of the serial data stream. When a Phase-Align Buffer error is present, the transmission of a Word Sync Sequence re-centers the Phase-Align Buffer and clears the error indication.

**Note.** K28.5 characters may be added or removed from the data stream during the Phase Align Buffer reset operation. When used with non-Cypress devices that require a complete 16-character Word Sync Sequence for proper receive Elasticity Buffer Operation, it is recommend that the Phase Alignment Buffer reset be followed by a Word Sync Sequence to ensure proper operation.

#### Encoder

Each character received from the Input Register or Phase-Align Buffer is passed to the Encoder logic. This block interprets each character and any associated control bits, and outputs a 10-bit transmission character.

Depending on the operational mode, the generated transmission character may be

- the 10-bit pre-encoded character accepted in the Input Register.
- the 10-bit equivalent of the 8-bit Data character accepted in the Input Register.
- the 10-bit equivalent of the 8-bit Special Character code accepted in the Input Register.

**Note:**

7. LSB shifted out first.

- the 10-bit equivalent of the C0.7 violation character if a Phase-Align Buffer overflow or underflow error is present.
- a character that is part of the 511-character BIST sequence.
- a K28.5 character generated as an individual character or as part of the 16-character Word Sync Sequence.

#### Data Encoding

Raw data, as received directly from the Transmit Input Register, is seldom in a form suitable for transmission across a serial link. The characters must usually be processed or transformed to guarantee

- a minimum transition density (to allow the receive PLL to extract a clock from the serial data stream).
- a DC-balance in the signaling (to prevent baseline wander).
- run-length limits in the serial data (to limit the bandwidth requirements of the serial link).
- the remote receiver a way of determining the correct character boundaries (framing).

When the Encoder is enabled (ENCBYPx = 1), the characters transmitted are converted from Data or Special Character codes to 10-bit transmission characters, using an integrated 8B/10B encoder. When directed to encode the character as a Special Character code, the encoder uses the Special Character encoding rules listed in *Table 16*. When directed to encode the character as a Data character, it is encoded using the Data Character encoding rules in *Table 15*.

The 8B/10B encoder is standards compliant with ANSI/NCITS ASC X3.230-1994 Fibre Channel, IEEE 802.3z Gigabit Ethernet, the IBM<sup>®</sup> ESCON<sup>®</sup> and FICON<sup>™</sup> channels, ETSI DVB-ASI, and ATM Forum standards for data transport.

Many of the Special Character codes listed in *Table 16* may be generated by more than one input character. The CYP(V)(W)15G0403DXB is designed to support two independent (but non-overlapping) Special Character code tables. This allows the CYP(V)(W)15G0403DXB to operate in mixed environments with other Cypress HOTLink devices using the enhanced Cypress command code set, and the reduced command sets of other non-Cypress devices. Even when used in an environment that normally uses non-Cypress Special Character codes, the selective use of Cypress command codes can permit operation where running disparity and error handling must be managed.

Following conversion of each input character from eight bits to a 10-bit transmission character, it is passed to the Transmit Shifter and is shifted out LSB first, as required by ANSI and IEEE standards for 8B/10B coded serial data streams.

#### Transmit Modes

##### Encoder Bypass

When the Encoder is bypassed, the character captured from the TXDx[7:0] and TXCTx[1:0] input register is passed directly to the transmit shifter without modification. With the encoder bypassed, the TXCTx[1:0] inputs are considered part of the data character and do not perform a control function that would otherwise modify the interpretation of the TXDx[7:0] bits. The bit usage and mapping of these control bits when the Encoder is bypassed is shown in *Table 2*.

**Table 2. Encoder Bypass Mode**

| Signal Name    | Bus Weight     | 10B Name         |
|----------------|----------------|------------------|
| TXDx[0] (LSB)  | 2 <sup>0</sup> | a <sup>[7]</sup> |
| TXDx[1]        | 2 <sup>1</sup> | b                |
| TXDx[2]        | 2 <sup>2</sup> | c                |
| TXDx[3]        | 2 <sup>3</sup> | d                |
| TXDx[4]        | 2 <sup>4</sup> | e                |
| TXDx[5]        | 2 <sup>5</sup> | i                |
| TXDx[6]        | 2 <sup>6</sup> | f                |
| TXDx[7]        | 2 <sup>7</sup> | g                |
| TXCTx[0]       | 2 <sup>8</sup> | h                |
| TXCTx[1] (MSB) | 2 <sup>9</sup> | j                |

When the encoder is enabled, the TXCTx[1:0] data control bits control the interpretation of the TXDx[7:0] bits and the characters generated by them. These bits are interpreted as listed in *Table 3*.

**Table 3. Transmit Modes**

| TXCTx[1] | TXCTx[0] | Characters Generated            |
|----------|----------|---------------------------------|
| 0        | 0        | Encoded data character          |
| 0        | 1        | K28.5 fill character            |
| 1        | 0        | Special character code          |
| 1        | 1        | 16-character Word Sync Sequence |

#### Word Sync Sequence

When TXCTx[1:0] = 11, a 16-character sequence of K28.5 characters, known as a Word Sync Sequence, is generated on the associated channel. This sequence of K28.5 characters may start with either a positive or negative disparity K28.5 (as determined by the current running disparity and the 8B/10B coding rules). The disparity of the second and third K28.5 characters in this sequence are reversed from what normal 8B/10B coding rules would generate. The remaining K28.5 characters in the sequence follow all 8B/10B coding rules. The disparity of the generated K28.5 characters in this sequence follow a pattern of either ++---+---+---+---+--- or ---+---+---+---+---+---.

The generation of this sequence, once started, cannot be stopped until all 16 characters have been sent. The content of the associated input registers are ignored for the duration of this sequence. At the end of this sequence, if the TXCTx[1:0] = 11 condition is sampled again, the sequence restarts and remains uninterrupted for the following 15 character clocks.

#### Transmit BIST

Each transmit channel contains an internal pattern generator that can be used to validate both the link and device operation. These generators are enabled by the associated TXBISTx latch via the device configuration interface. When enabled, a register in the associated transmit channel becomes a signature pattern generator by logically converting to a Linear Feedback Shift Register (LFSR). This LFSR generates a 511-character (or 526-character) sequence that includes all Data and Special Character codes, including the explicit violation symbols. This provides a predictable yet pseudo-random

sequence that can be matched to an identical LFSR in the attached Receiver(s).

A device reset ( $\overline{\text{RESET}}$  sampled LOW) presets the BIST Enable Latches to disable BIST on all channels.

All data and data-control information present at the associated TXDx[7:0] and TXCTx[1:0] inputs are ignored when BIST is active on that channel. If the receive channels are configured for reference clock operation, each pass is preceded by a 16-character Word Sync Sequence to allow Elasticity Buffer alignment and management of clock-frequency variations.

#### Transmit PLL Clock Multiplier

Each Transmit PLL Clock Multiplier accepts a character-rate or half-character-rate external clock at the associated REFCLKx± input, and that clock is multiplied by 10 or 20 (as selected by TXRATEx) to generate a bit-rate clock for use by the transmit shifter. It also provides a character-rate clock used by the transmit paths, and outputs this character rate clock as TXCLKOx.

Each clock multiplier PLL can accept a REFCLKx± input between 19.5 MHz and 150 MHz (19.5 MHz and 154 MHz for CYW15G0403DXB), however, this clock range is limited by the operating mode of the CYP(V)(W)15G0403DXB clock multiplier (TXRATEx) and by the level on the associated SPDSELx input.

SPDSELx are 3-level select<sup>[4]</sup> inputs that select one of three operating ranges for the serial data outputs and inputs of the associated channel. The operating serial signaling-rate and allowable range of REFCLKx± frequencies are listed in *Table 4*.

**Table 4. Operating Speed Settings**

| SPDSELx    | TXRATE | REFCLKx± Frequency (MHz) | Signaling Rate (MBAud)                   |
|------------|--------|--------------------------|--|
| LOW        | 1      | reserved                 | 195–400                                  |
|            | 0      | 19.5–40                  |  |
| MID (Open) | 1      | 20–40                    | 400–800                                  |
|            | 0      | 40–80                    |  |
| HIGH       | 1      | 40–75                    | 800–1500<br>(800–1540 for CYW15G0403DXB) |
|            | 0      | 80–150                   |  |

The REFCLKx± inputs are differential inputs with each input internally biased to 1.4V. If the REFCLKx+ input is connected to a TTL, LVTTTL, or LVCMOS clock source, the input signal is recognized when it passes through the internally biased reference point. When driven by a single-ended TTL, LVTTTL, or LVCMOS clock source, connect the clock source to either the true or complement REFCLKx input, and leave the alternate REFCLKx input open (floating).

When both the REFCLKx+ and REFCLKx– inputs are connected, the clock source must be a differential clock. This can either be a differential LVPECL clock that is DC-or AC-coupled or a differential LVTTTL or LVCMOS clock.

By connecting the REFCLKx– input to an external voltage source, it is possible to adjust the reference point of the REFCLKx+ input for alternate logic levels. When doing so it is necessary to ensure that the input differential crossing point remains within the parametric range supported by the input.

## Serial Output Drivers

The serial output interface drivers use differential Current Mode Logic (CML) drivers to provide source-matched drivers for transmission lines. These drivers accept data from the Transmit Shifters. These drivers have signal swings equivalent to that of standard PECL drivers, and are capable of driving AC-coupled optical modules or transmission lines. When configured for local loopback (LPENx = HIGH), all enabled serial drivers are configured to drive a static differential logic 1. To achieve OBSAI RP3 compliancy, the serial output drivers must be AC-coupled to the transmission medium.

### Transmit Channels Enabled

Each driver can be enabled or disabled separately via the device configuration interface.

When a driver is disabled via the configuration interface, it is internally powered down to reduce device power. If both serial drivers for a channel are in this disabled state, the associated internal logic for that channel is also powered down. A device reset (RESET sampled LOW) disables all output drivers.

**Note.** When a disabled transmit channel (i.e., both outputs disabled) is re-enabled:

- data on the serial outputs may not meet all timing specifications for up to 250  $\mu$ s
- the state of the phase-align buffer cannot be guaranteed, and a phase-align reset is required if the phase-align buffer is used

## CYP(V)(W)15G0403DXB Receive Data Path

### Serial Line Receivers

Two differential Line Receivers, INx1 $\pm$  and INx2 $\pm$ , are available on each channel for accepting serial data streams. The active Serial Line Receiver on a channel is selected using the associated INSELx input. The Serial Line Receiver inputs are differential, and can accommodate wire interconnect and filtering losses or transmission line attenuation greater than 16 dB. For normal operation, these inputs should receive a signal of at least  $V_{DIFF} > 100$  mV, or 200 mV peak-to-peak differential. Each Line Receiver can be DC- or AC-coupled to +3.3V powered fiber-optic interface modules (any ECL/PECL family, not limited to 100K PECL) or AC-coupled to +5V powered optical modules. The common-mode tolerance of these line receivers accommodates a wide range of signal termination voltages. Each receiver provides internal DC-restoration, to the center of the receiver's common mode range, for AC-coupled signals.

The local internal loopback (LPENx) allows the serial transmit data outputs to be routed internally back to the Clock and Data Recovery circuit associated with each channel. When configured for local loopback, the associated transmit serial driver outputs are forced to output a differential logic-1. This prevents local diagnostic patterns from being broadcast to attached remote receivers.

### Signal Detect/Link Fault

Each selected Line Receiver (i.e., that routed to the clock and data recovery PLL) is simultaneously monitored for

#### Note:

8. The peak amplitudes listed in this table are for typical waveforms that have generally 3–4 transitions for every ten bits. In a worse case environment the signals may have a sine-wave appearance (highest transition density with repeating 0101...). Signal peak amplitudes levels within this environment type could increase the values in the table above by approximately 100 mV.

- analog amplitude above amplitude level selected by SDASELx
- transition density above the specified limit
- range controls report the received data stream inside normal frequency range ( $\pm 1500$  ppm<sup>[30]</sup>)
- receive channel enabled
- Presence of reference clock
- $\overline{ULCx}$  is not asserted.

All of these conditions must be valid for the Signal Detect block to indicate a valid signal is present. This status is presented on the LFlx (Link Fault Indicator) output associated with each receive channel, which changes synchronous to the selected receive interface clock.

### Analog Amplitude

While most signal monitors are based on fixed constants, the analog amplitude level detection is adjustable to allow operation with highly attenuated signals, or in high-noise environments. The analog amplitude level detection is set by the SDASELx latch via device configuration interface. The SDASELx latch sets the trip point for the detection of a valid signal at one of three levels, as listed in Table 5. This control input affects the analog monitors for all receive channels.

**Table 5. Analog Amplitude Detect Valid Signal Levels<sup>[8]</sup>**

| SDASEL | Typical Signal with Peak Amplitudes Above |
|--------|---|
| 00     | Analog Signal Detector is disabled        |
| 01     | 140 mV p-p differential                   |
| 10     | 280 mV p-p differential                   |
| 11     | 420 mV p-p differential                   |

The Analog Signal Detect monitors are active for the Line Receiver as selected by the associated INSELx input. When configured for local loopback, no input receivers are selected, and the LFlx output for each channel reports only the receive VCO frequency out-of-range and transition density status of the associated transmit signal. When local loopback is active, the associated Analog Signal Detect Monitor is disabled.

### Transition Density

The Transition Detection logic checks for the absence of transitions spanning greater than six transmission characters (60 bits). If no transitions are present in the data received, the Detection logic for that channel asserts LFlx.

### Range Controls

The CDR circuit includes logic to monitor the frequency of the PLL Voltage Controlled Oscillator (VCO) used to sample the incoming data stream. This logic ensures that the VCO operates at, or near the rate of the incoming data stream for two primary cases:

- when the incoming data stream resumes after a time in which it has been "missing."
- when the incoming data stream is outside the acceptable signaling rate range.

To perform this function, the frequency of the RXPLL VCO is periodically compared to the frequency of the REFCLKx $\pm$



input. If the VCO is running at a frequency beyond  $\pm 1500 \text{ ppm}^{[30]}$  as defined by the REFCLKx $\pm$  frequency, it is periodically forced to the correct frequency (as defined by REFCLKx $\pm$ , SPDSELx, and TXRATEx) and then released in an attempt to lock to the input data stream.

The sampling and relock period of the Range Control is calculated as follows: RANGE\_CONTROL\_SAMPLING\_PERIOD = (RECOVERED BYTE CLOCK PERIOD) \* (4096).

During the time that the Range Control forces the RXPLL VCO to track REFCLKx $\pm$ , the LFIx output is asserted LOW. After a valid serial data stream is applied, it may take up to one RANGE CONTROL SAMPLING PERIOD before the PLL locks to the input data stream, after which LFIx should be HIGH.

*Receive Channel Enabled*

The CYP(V)(W)15G0403DXB contains four receive channels that can be independently enabled and disabled. Each channel can be enabled or disabled separately through the RXPLLPDx input latch as controlled by the device configuration interface. When the RXPLLPDx latch = 0, the associated PLL and analog circuitry of the channel is disabled. Any disabled channel indicates a constant link fault condition on the LFIx output. When RXPLLPDx = 1, the associated PLL and receive channel is enabled to receive and decode a serial stream.

**Note.** When a disabled receive channel is reenabled, the status of the associated LFIx output and data on the parallel outputs for the associated channel may be indeterminate for up to 2 ms.

**Clock/Data Recovery**

The extraction of a bit-rate clock and recovery of bits from each received serial stream is performed by a separate CDR block within each receive channel. The clock extraction function is performed by an integrated PLL that tracks the frequency of the transitions in the incoming bit stream and align the phase of the internal bit-rate clock to the transitions in the selected serial data stream.

Each CDR accepts a character-rate (bit-rate  $\div$  10) or half-character-rate (bit-rate  $\div$  20) reference clock from the associated REFCLKx $\pm$  input. This REFCLKx $\pm$  input is used to

- ensure that the VCO (within the CDR) is operating at the correct frequency (rather than a harmonic of the bit-rate)
- reduce PLL acquisition time
- limit unlocked frequency excursions of the CDR VCO when there is no input data present at the selected Serial Line Receiver.

Regardless of the type of signal present, the CDR attempts to recover a data stream from it. If the signalling rate of the recovered data stream is outside the limits set by the range control monitors, the CDR tracks REFCLKx $\pm$  instead of the data stream. Once the CDR output (RXCLK $\pm$ ) frequency returns back close to REFCLKx $\pm$  frequency, the CDR input is switched back to the input data stream. If no data is present at the selected line receiver, this switching behavior may result in brief RXCLK $\pm$  frequency excursions from REFCLKx $\pm$ . However, the validity of the input data stream is indicated by the LFIx output. The frequency of REFCLKx $\pm$  is required to be

**Note:**

9. The standard definition of a Comma contains only seven bits. However, since all valid Comma characters within the 8B/10B character set also have the eighth bit as an inversion of the seventh bit, the compare pattern is extended to a full eight bits to reduce the possibility of a framing error.

within  $\pm 1500 \text{ ppm}^{[30]}$  of the frequency of the clock that drives the REFCLKx $\pm$  input of the remote transmitter to ensure a lock to the incoming data stream.

For systems using multiple or redundant connections, the LFIx output can be used to select an alternate data stream. When an LFIx indication is detected, external logic can toggle selection of the associated INx1 $\pm$  and INx2 $\pm$  input through the associated INSELx input. When a port switch takes place, it is necessary for the receive PLL for that channel to reacquire the new serial stream and frame to the incoming character boundaries.

**Deserialzer/Framer**

Each CDR circuit extracts bits from the associated serial data stream and clocks these bits into the Shifter/Framer at the bit-clock rate. When enabled, the Framer examines the data stream looking for one or more COMMA or K28.5 characters at all possible bit positions. The location of this character in the data stream is used to determine the character boundaries of all following characters.

*Framing Character*

The CYP(V)(W)15G0403DXB allows selection of different framing characters on each channel. Two combinations of framing characters are supported to meet the requirements of different interfaces. The selection of the framing character is made through the FRAMCHARx latches via the configuration interface.

The specific bit combinations of these framing characters are listed in Table 6. When the specific bit combination of the selected framing character is detected by the framer, the boundaries of the characters present in the received data stream are known.

**Table 6. Framing Character Selector**

| FRAMCHARx | Bits detected in framer |  |
|-----------|-------------------------|--|
|           | Character Name          | Bits Detected                              |
| 0         | COMMA+<br>COMMA-        | 00111110XX <sup>[9]</sup><br>or 11000001XX |
| 1         | -K28.5<br>+K28.5        | 0011111010 or<br>1100000101                |

*Framer*

The framer on each channel operates in one of three different modes. Each framer may be enabled or disabled using the RFENx latches via the configuration interface. When the framer is disabled (RFENx = 0), no combination of received bits alters the frame information.

When the Low-Latency framer is selected (RFMODEx[1:0] = 00), the framer operates by stretching the recovered character clock until it aligns with the received character boundaries. In this mode the framer starts its alignment process on the first detection of the selected framing character. To reduce the impact on external circuits that use the recovered clock, the clock period is not stretched by more than two bit-periods in any one clock cycle. When operated with a character-rate output clock, the output of properly framed characters may be delayed by up to nine character-clock cycles from the detection of the selected framing character. When operated

with a half-character-rate output clock, the output of properly framed characters may be delayed by up to 14 character-clock cycles from the detection of the framing character.

**Note.** When Receive BIST is enabled on a channel, the Low-Latency Framer must not be enabled. The BIST sequence contains an aliased K28.5 framing character, which causes the Receiver to update its character boundaries incorrectly.

When RFMODEx[1:0] = 10, the Cypress-Mode Multi-Byte framer is selected. The required detection of multiple framing characters makes the associated link much more robust to incorrect framing due to aliased SYNC characters in the data stream. In this mode, the framer does not adjust the character clock boundary, but instead aligns the character to the already recovered character clock. This ensures that the recovered clock does not contain any significant phase changes or hops during normal operation or framing, and allows the recovered clock to be replicated and distributed to other external circuits or components using PLL-based clock distribution elements. In this framing mode the character boundaries are only adjusted if the selected framing character is detected at least twice within a span of 50 bits, with both instances on identical 10-bit character boundaries.

When RFMODEx[1:0] = 01, the Alternate-mode Multi-Byte Framer is enabled. Like the Cypress-mode Multi-Byte Framer, multiple framing characters must be detected before the character boundary is adjusted. In this mode, the data stream must contain a minimum of four of the selected framing characters, received as consecutive characters, on identical 10-bit boundaries, before character framing is adjusted.

### 10B/8B Decoder Block

The decoder logic block performs two primary functions:

- decoding the received transmission characters to Data and Special Character codes
- comparing generated BIST patterns with received characters to permit at-speed link and device testing.

The framed parallel output of each deserializer shifter is passed to its associated 10B/8B Decoder where, if the decoder is enabled, the input data is transformed from a 10-bit transmission character back to the original Data or Special Character code. This block uses the 10B/8B decoder patterns in *Table 15* and *Table 16*. Received Special Code characters are decoded using *Table 16*. Valid data characters are indicated by a 000b bit-combination on the associated RXSTx[2:0] status bits, and Special Character codes are indicated by a 001b bit-combination of these status outputs. Framing characters, Invalid patterns, disparity errors, and synchronization status are presented as alternate combinations of these status bits.

When DECBYPx = 0, the 10B/8B decoder is bypassed via the configuration interface. When bypassed, raw 10-bit characters are passed through the receiver and presented at the RXDx[7:0] and the RXSTA[1:0] outputs as 10-bit wide characters.

When the decoder is enabled by setting DECBYPx = 1 via the configuration interface, the 10-bit transmission characters are decoded using *Table 15* and *Table 16*. Received Special characters are decoded using *Table 16*. The columns used in *Table 16* are determined by the DECMODEx latch via the device configuration interface. When DECMODEx = 0 the ALTERNATE table is used and when DECMODEx = 1 the CYPRESS table is used.

### Receive BIST Operation

The receiver channel contains an internal pattern checker that can be used to validate both device and link operation. These pattern checkers are enabled by the associated RXBISTx latch via the device configuration interface. When enabled, a register in the associated receive channel becomes a signature pattern generator and checker by logically converting to a Linear Feedback Shift Register (LFSR). This LFSR generates a 511-character or 526-character sequence that includes all Data and Special Character codes, including the explicit violation symbols. This provides a predictable yet pseudo-random sequence that can be matched to an identical LFSR in the attached Transmitter(s). When synchronized with the received data stream, the associated Receiver checks each character in the Decoder with each character generated by the LFSR and indicates compare errors and BIST status at the RXSTx[2:0] bits of the Output Register.

When BIST is first recognized as being enabled in the Receiver, the LFSR is preset to the BIST-loop start-code of D0.0. This code D0.0 is sent only once per BIST loop. The status of the BIST progress and any character mismatches are presented on the RXSTx[2:0] status outputs.

Code rule violations or running disparity errors that occur as part of the BIST loop do not cause an error indication. RXSTx[2:0] indicates 010b or 100b for one character period per BIST loop to indicate loop completion. This status can be used to check test pattern progress. These same status values are presented when the decoder is bypassed and BIST is enabled on a receive channel.

The specific status reported by the BIST state machine are listed in *Table 11*. These same codes are reported on the receive status outputs.

The specific patterns checked by each receiver are described in detail in the Cypress application note "HOTLink Built-In Self-Test." The sequence compared by the CYP(V)(W)15G0403DXB is identical to that in the CY7B933, CY7C924DX, and CYP(V)(W)15G0401DXB, allowing interoperable systems to be built when used at compatible serial signaling rates.

If the number of invalid characters received ever exceeds the number of valid characters by 16, the receive BIST state machine aborts the compare operations and resets the LFSR to the D0.0 state to look for the start of the BIST sequence again.

When the receive paths are configured for REFCLKx± operation, each pass must be preceded by a 16-character Word Sync Sequence to allow management of clock frequency variations.

The receive BIST state machine requires the characters to be correctly framed for it to detect the BIST sequence. If the Low Latency Framer is enabled, the Framer misaligns to an aliased SYNC character within the BIST sequence. If the Alternate Multi-Byte Framer is enabled and the Receiver outputs are clocked relative to a recovered clock, it is generally necessary to frame the receiver before BIST is enabled. If the receive outputs are clocked relative to REFCLKx±, the transmitter precedes every 511 character BIST sequence with a 16 character-character Word Sync Sequence.

A device reset ( $\overline{\text{RESET}}$  sampled LOW) presets the BIST Enable Latches to disable BIST on all channels.



### Receive Elasticity Buffer

Each receive channel contains an Elasticity Buffer that is designed to support multiple clocking modes. These buffers allow data to be read using a clock that is asynchronous in both frequency and phase from the Elasticity Buffer write clock, or to be read using a clock that is frequency coherent but with uncontrolled phase relative to the Elasticity Buffer write clock.

If the chip is configured for operation with a recovered clock, the Elasticity Buffer is bypassed.

Each Elasticity Buffer is 10 characters deep, and supports and an 11 bit wide data path. It is capable of supporting a decoded character and three status bits for each character present in the buffer. The write clock for these buffers is always the recovered clock for the associated read channel.

### Receive Modes

When the receive channel is clocked by REFCLKx±, the RXCLKx± outputs present a buffered or divided (depending on RXRATEx) and delayed form of REFCLKx±. In this mode, the receive Elasticity Buffers are enabled. For REFCLKx± clocking, the Elasticity Buffers must be able to insert K28.5 characters and delete framing characters as appropriate.

The insertion of a K28.5 or deletion of a framing character can occur at any time on any channel, however, the actual timing of these insertions and deletions is controlled in part by how the transmitter sends its data. Insertion of a K28.5 character can only occur when the receiver has a framing character in the Elasticity Buffer. Likewise, to delete a framing character, one must also be in the Elasticity Buffer. To prevent a buffer overflow or underflow on a receive channel, a minimum density of framing characters must be present in the received data streams.

When the receive channel Output Register is clocked by a recovered clock, no characters are added or deleted and the receiver Elasticity Buffer is bypassed.

### Power Control

The CYP(V)(W)15G0403DXB supports user control of the powered up or down state of each transmit and receive channel. The receive channels are controlled by the RXPLLPDx latch via the device configuration interface. When RXPLLPDx = 0, the associated PLL and analog circuitry of the channel is disabled. The transmit channels are controlled by the OE1x and the OE2x latches via the device configuration interface. When a driver is disabled via the configuration interface, it is internally powered down to reduce device power. If both serial drivers for a channel are in this disabled state, the associated internal logic for that channel is also powered down.

#### Device Reset State

When the CYP(V)(W)15G0403DXB is reset by assertion of RESET, all state machines, counters, and configuration latches in the device are initialized to a reset state, and the Elasticity Buffer pointers are set to a nominal offset. See Table 9 for the initialize values of the configuration latches.

Following a device reset, it is necessary to enable the transmit and receive channels used for normal operation. This can be done by sequencing the appropriate values on the device configuration interface.<sup>[5]</sup>

### Output Bus

Each receive channel presents an 11-signal output bus consisting of

- an 8-bit data bus
- a 3-bit status bus.

The signals present on this output bus are modified by the present operating mode of the CYP(V)(W)15G0403DXB as selected by the DECBYPx configuration latch. This mapping is shown in Table 7.

**Table 7. Output Register Bit Assignments**

| Signal Name                | BYPASS ACTIVE (DECBYPx = 0) | DECODER (DECBYP = 1) |
|----------------------------|-----------------------------|----------------------|
| RXSTx[2] (LSB)             | COMDET <sub>x</sub>         | RXSTx[2]             |
| RXSTx[1]                   | DOUTx[0]                    | RXSTx[1]             |
| RXSTx[0]                   | DOUTx[1]                    | RXSTx[0]             |
| RXD <sub>x</sub> [0]       | DOUTx[2]                    | RXD <sub>x</sub> [0] |
| RXD <sub>x</sub> [1]       | DOUTx[3]                    | RXD <sub>x</sub> [1] |
| RXD <sub>x</sub> [2]       | DOUTx[4]                    | RXD <sub>x</sub> [2] |
| RXD <sub>x</sub> [3]       | DOUTx[5]                    | RXD <sub>x</sub> [3] |
| RXD <sub>x</sub> [4]       | DOUTx[6]                    | RXD <sub>x</sub> [4] |
| RXD <sub>x</sub> [5]       | DOUTx[7]                    | RXD <sub>x</sub> [5] |
| RXD <sub>x</sub> [6]       | DOUTx[8]                    | RXD <sub>x</sub> [6] |
| RXD <sub>x</sub> [7] (MSB) | DOUTx[9]                    | RXD <sub>x</sub> [7] |

When the 10B/8B decoder is bypassed, the framed 10-bit value is presented to the associated Output Register, along with a status output signal indicating if the character in the Output Register is one of the selected framing characters. The bit usage and mapping of the external signals to the raw 10B transmission character is shown in Table 8.

**Table 8. Decoder Bypass Mode**

| Signal Name                | Bus Weight          | 10 Bit Name |
|----------------------------|---------------------|-------------|
| RXSTx[2] (LSB)             | COMDET <sub>x</sub> |             |
| RXSTx[1]                   | 2 <sup>0</sup>      | a           |
| RXSTx[0]                   | 2 <sup>1</sup>      | b           |
| RXD <sub>x</sub> [0]       | 2 <sup>2</sup>      | c           |
| RXD <sub>x</sub> [1]       | 2 <sup>3</sup>      | d           |
| RXD <sub>x</sub> [2]       | 2 <sup>4</sup>      | e           |
| RXD <sub>x</sub> [3]       | 2 <sup>5</sup>      | i           |
| RXD <sub>x</sub> [4]       | 2 <sup>6</sup>      | f           |
| RXD <sub>x</sub> [5]       | 2 <sup>7</sup>      | g           |
| RXD <sub>x</sub> [6]       | 2 <sup>8</sup>      | h           |
| RXD <sub>x</sub> [7] (MSB) | 2 <sup>9</sup>      | j           |

The COMDET<sub>x</sub> status output operates the same regardless of the bit combination selected for character framing by the FRAMCHAR<sub>x</sub> latch. COMDET<sub>x</sub> is HIGH when the character in the output register contains the selected framing character at the proper character boundary, and LOW for all other bit combinations.

When the low-latency framer and half-rate receive port clocking are also enabled, the framer stretches the recovered clock to the nearest 20-bit boundary such that the rising edge of RXCLKx+ occurs when COMDET<sub>x</sub> is present on the associated output bus.

When the Cypress or Alternate Mode Framer is enabled and half-rate receive port clocking is also enabled, the output clock is not modified when framing is detected, but a single pipeline stage may be added or subtracted from the data stream by the framer logic such that the rising edge of RXCLKx+ occurs when COMDET<sub>x</sub> is present on the associated output bus.

This adjustment only occurs when the framer is enabled. When the framer is disabled, the clock boundaries are not adjusted, and COMDET<sub>x</sub> may be asserted during the rising edge of RXCLKx- (if an odd number of characters were received following the initial framing).

#### Receive Status Bits

When the 10B/8B decoder is enabled, each character presented at the Output Register includes three associated status bits. These bits are used to identify

- if the contents of the data bus are valid,
- the type of character present,
- the state of receive BIST operations,
- character violations.

These conditions often overlap; e.g. a valid data character received with incorrect running disparity is not reported as a valid data character. It is instead reported as a decoder violation of some specific type. This implies a hierarchy or priority level to the various status bit combinations. The hierarchy and value of each status are listed in *Table 11*.

A second status mapping, listed in *Table 11*, is used when the receive channel is configured for BIST operation. This status is used to report receive BIST status and progress.

#### BIST Status State Machine

When a receive path is enabled to look for and compare the received data stream with the BIST pattern, the RXSTx[2:0] bits identify the present state of the BIST compare operation.

The BIST state machine has multiple states, as shown in *Figure 2* and *Table 11*. When the receive PLL detects an out-of-lock condition, the BIST state is forced to the Start-of-BIST state, regardless of the present state of the BIST state machine. If the number of detected errors ever exceeds the number of valid matches by greater than 16, the state machine is forced to the WAIT\_FOR\_BIST state where it monitors the receive path for the first character of the next BIST sequence (D0.0). Also, if the Elasticity Buffer ever hits an overflow/underflow condition, the status is forced to the BIST\_START until the buffer is re-centered (approximately nine character periods).

To ensure compatibility between the source and destination systems when operating in BIST modes, the sending and receiving ends of the link must use the same receive clock configuration.

#### Device Configuration and Control Interface

The CYP(V)(W)15G0403DXB is highly configurable via the configuration interface. The configuration interface allows the device to be configured globally or allows each channel to be configured independently. *Table 9* lists the configuration

latches within the device including the initialization value of the latches upon the assertion of RESET. *Table 10* shows how the latches are mapped in the device. Each row in the *Table 10* maps to a 8-bit latch bank. There are 16 such write-only latch banks. When WREN = 0, the logic value in the DATA[7:0] is latched to the latch bank specified by the values in ADDR[3:0]. The second column of *Table 10* specifies the channels associated with the corresponding latch bank. For example, the first three latch banks (0, 1 and 2) consist of configuration bits for channel A. The latch banks 12, 13 and 14 consist of Global configuration bits and the last latch bank (15) is the Mask latch bank that can be configured to perform bit-by-bit configuration.

#### Global Enable Function

The global enable function, controlled by the GLENx bits, is a feature that can be used to reduce the number of write operations needed to setup the latch banks. This function is beneficial in systems that use a common configuration in multiple channels. The GLENx bit is present in bit 0 of latch banks 0 through 11 only. Its default value (1) enables the global update of the latch bank's contents. Setting the GLENx bit to 0 disables this functionality.

Latch Banks 12, 13, and 14 are used to load values in the related latch banks in a global manner. A write operation to latch bank 12 could do a global write to latch banks 0, 3, 6, and 9 depending on the value of GLENx in these latch banks; latch bank 13 could do a global write to latch banks 1, 4, 7 and 10; and latch banks 14 could do a global write to latch banks 2, 5, 8 and 11. The GLENx bit cannot be modified by a global write operation.

#### Force Global Enable Function

FGLENx forces the global update of the target latch banks, but does not change the contents of the GLENx bits. If FGLENx = 1 for the associated global channel, FGLENx forces the global update of the target latch banks.

#### Mask Function

An additional latch bank (15) is used as a global mask vector to control the update of the configuration latch banks on a bit-by-bit basis. A logic 1 in a bit location allows for the update of that same location of the target latch bank(s), whereas a logic 0 disables it. The reset value of this latch bank is FFh, thereby making its use optional by default. The mask latch bank is not maskable. The FGLEN functionality is not affected by the bit 0 value of the mask latch bank.

#### Latch Types

There are two types of latch banks: static (S) and dynamic (D). Each channel is configured by 2 static and 1 dynamic latch banks. The S type contain those settings that normally do not change for a given application, whereas the D type controls the settings that could change dynamically during the application's lifetime. The first row of latches for each channel (address numbers 0, 3, 7, and 10) are the static receiver control latches. The second row of latches for each channel (address numbers 1, 4, 8, and 11) are the static transmitter control latches. The third row of latches for each channel (address numbers 2, 5, 9, and 12) are the dynamic control latches that are associated with enabling dynamic functions within the device.



Latch Bank 14 is also useful for those users that do not need the latch-based programmable feature of the device. This latch bank could be used in those applications that do not need to modify the default value of the static latch banks, and that can afford a global (i.e., not independent) control of the

dynamic signals. In this case, this feature becomes available when ADDR[3:0] is left unchanged with a value of "1110" and WREN is left asserted. The signals present in DATA[7:0] effectively become global control pins, and for the latch banks 2, 5, 8 and 11.

**Table 9. Device Configuration and Control Latch Descriptions**

| Name   | Signal Description  |
|--|---|
| RFMODEA[1:0]<br>RFMODEB[1:0]<br>RFMODEC[1:0]<br>RFMODED[1:0] | <b>Reframe Mode Select.</b> The initialization value of the RFMODEx [1:0] latches = 10. RFMODEx is used to select the operating mode of the framer. When RFMODEx[1:0] = 00, the low-latency framer is selected. This frames on each occurrence of the selected framing character(s) in the received data stream. This mode of framing stretches the recovered clock for one or multiple cycles to align that clock with the recovered data. When RFMODEx[1:0] = 01, the alternate mode Multi-Byte parallel framer is selected. This requires detection of the selected framing character(s) in the received serial bit stream, on identical 10-bit boundaries, on four directly adjacent characters. The recovered character clock remains in the same phasing regardless of character offset. When RFMODEx[1:0] = 10, the Cypress-mode Multi-Byte parallel framer is selected. This requires a pair of the selected framing character(s), on identical 10-bit boundaries, within a span of 50 bits, before the character boundaries are adjusted. The recovered character clock remains in the same phasing regardless of character offset. RFMODEx[1:0] = 11 is reserved for test.  |
| FRAMCHARA<br>FRAMCHARB<br>FRAMCHARC<br>FRAMCHARD             | <b>Framing Character Select.</b> The initialization value of the FRAMCHARx latch = 1. FRAMCHARx is used to select the character or portion of a character used for framing of each channel's received data stream. When FRAMCHARx = 1, the framer looks for either disparity of the K28.5 character. When FRAMCHARx = 0, the framer looks for either disparity of the 8-bit Comma characters. The specific bit combinations of these framing characters are listed in <i>Table 6</i> .  |
| DECMODEA<br>DECMODEB<br>DECMODEC<br>DECMODED                 | <b>Receiver Decoder Mode Select.</b> The initialization value of the DECMODEx latch = 1. DECMODEx selects the Decoder Mode used for the associated channel. When DECMODEx = 1 and decoder is enabled, the Cypress Decoding Mode is used. When DECMODEx = 0 and decoder is enabled, the Alternate Decoding mode is used. When the decoder is enabled (DECByPx = 1), the 10-bit transmission characters are decoded using <i>Table 15</i> and <i>Table 16</i> . The column used in the Special Characters <i>Table 16</i> is determined by the DECMODEx latch.  |
| DECBYPA<br>DECBYPB<br>DECBYPC<br>DECBYPD                     | <b>Receiver Decoder Bypass.</b> The initialization value of the DECByPx latch = 1. DECByPx selects if the Receiver Decoder is enabled or bypassed. When DECByPx = 1, the decoder is enabled and the Decoder Mode is selected by DECMODEx. When DECByPx = 0, the decoder is bypassed and raw 10-bit characters are passed through the receiver.  |
| RXCKSELA<br>RXCKSELB<br>RXCKSELC<br>RXCKSELD                 | <b>Receive Clock Select.</b> The initialization value of the RXCKSELx latch = 1. RXCKSELx selects the receive clock source used to transfer data to the Output Registers and the clock source for the RXCLK± output. When RXCKSELx = 1, the associated Output Registers, are clocked by REFCLKx± at the associated RXCLKx± output buffer. When RXCKSELx = 0, the associated Output Registers, are clocked by the Recovered Byte clock at the associated RXCLKx± output buffer. These output clocks may operate at the character-rate or half the character-rate as selected by RXRATEx.   |
| RXRATEA<br>RXRATEB<br>RXRATEC<br>RXRATED                     | <b>Receive Clock Rate Select.</b> The initialization value of the RXRATEx latch = 1. RXRATEx is used to select the rate of the RXCLKx± clock output.<br>When RXRATEx = 1 and RXCKSELx = 0, the RXCLKx± clock outputs are complementary clocks that follow the recovered clock operating at half the character rate. Data for the associated receive channels should be latched alternately on the rising edge of RXCLKx+ and RXCLKx-.<br>When RXRATEx = 0 and RXCKSELx = 0, the RXCLKx± clock outputs are complementary clocks that follow the recovered clock operating at the character rate. Data for the associated receive channels should be latched on the rising edge of RXCLKx+ or falling edge of RXCLKx-.<br>When RXRATEx = 1 with RXCKSELx = 1 and REFCLKx± is a full-rate clock, the RXCLKx± clock outputs are complementary clocks that follow the reference clock operating at half the character rate. Data for the associated receive channels should be latched alternately on the rising edge of RXCLKx+ and RXCLKx-.<br>When RXRATEx = 0 with RXCKSELx = 1 and REFCLKx± is a full-rate clock, the RXCLKx± clock outputs are complementary clocks that follow the reference clock operating at the character rate. Data for the associated receive channels should be latched on the rising edge of RXCLKx+ or falling edge of RXCLKx-.<br>When RXCKSELx = 1 and REFCLKx± is a half-rate clock, the value of RXRATEx is not interpreted and the RXCLKx± clock outputs are complementary clocks that follow the reference clock operating at half the character rate. Data for the associated receive channels should be latched alternately on the rising edge of RXCLKx+ and RXCLKx-. |

**Table 9. Device Configuration and Control Latch Descriptions** (continued)

| Name   | Signal Description   |
|--|--|
| SDASEL1A[1:0]<br>SDASEL1B[1:0]<br>SDASEL1C[1:0]<br>SDASEL1D[1:0] | <b>Primary Serial Data Input Signal Detector Amplitude Select.</b> The initialization value of the SDASEL1x[1:0] latch = 10. SDASEL1x[1:0] selects the trip point for the detection of a valid signal for the INx1± Primary Differential Serial Data Inputs.<br>When SDASEL1x[1:0] = 00, the Analog Signal Detector is disabled.<br>When SDASEL1x[1:0] = 01, the typical p-p differential voltage threshold level is 140 mV.<br>When SDASEL1x[1:0] = 10, the typical p-p differential voltage threshold level is 280 mV.<br>When SDASEL1x[1:0] = 11, the typical p-p differential voltage threshold level is 420 mV.   |
| SDASEL2A[1:0]<br>SDASEL2B[1:0]<br>SDASEL2C[1:0]<br>SDASEL2D[1:0] | <b>Secondary Serial Data Input Signal Detector Amplitude Select.</b> The initialization value of the SDASEL2x[1:0] latch = 10. SDASEL2x[1:0] selects the trip point for the detection of a valid signal for the INx2± Secondary Differential Serial Data Inputs.<br>When SDASEL2x[1:0] = 00, the Analog Signal Detector is disabled.<br>When SDASEL2x[1:0] = 01, the typical p-p differential voltage threshold level is 140 mV.<br>When SDASEL2x[1:0] = 10, the typical p-p differential voltage threshold level is 280 mV.<br>When SDASEL2x[1:0] = 11, the typical p-p differential voltage threshold level is 420 mV.   |
| ENCBYPA<br>ENCBYPB<br>ENCBYPC<br>ENCBYPD                         | <b>Transmit Encoder Bypassed.</b> The initialization value of the ENCBYPx latch = 1. ENCBYPx selects if the Transmit Encoder is enabled or bypassed. When ENCBYPx = 1, the Transmit encoder is enabled. When ENCBYPx = 0, the Transmit Encoder is bypassed and raw 10-bit characters are transmitted.  |
| TXCKSELA<br>TXCKSELB<br>TXCKSELC<br>TXCKSELD                     | <b>Transmit Clock Select.</b> The initialization value of the TXCKSELx latch = 1. TXCKSELx selects the clock source used to write data into the Transmit Input Register. When TXCKSELx = 1, the associated input register, TXDx[7:0] and TXCTx[1:0], is clocked by REFCLKx↑. In this mode, the phase alignment buffer in the transmit path is bypassed. When TXCKSELx = 0, the associated TXCLKx↑ is used to clock in the input registers, TXDx[7:0] and TXCTx[1:0].   |
| TXRATEA<br>TXRATEB<br>TXRATEC<br>TXRATED                         | <b>Transmit PLL Clock Rate Select.</b> The initialization value of the TXRATEx latch = 0. TXRATEx is used to select the clock multiplier for the Transmit PLL. When TXRATEx = 0, each transmit PLL multiplies the associated REFCLKx± input by 10 to generate the serial bit-rate clock. When TXRATEx = 0, the TXCLKOx output clocks are full-rate clocks and follow the frequency and duty cycle of the associated REFCLKx± input. When TXRATEx = 1, each Transmit PLL multiplies the associated REFCLKx± input by 20 to generate the serial bit-rate clock. When TXRATEx = 1, the TXCLKOx output clocks are twice the frequency rate of the REFCLKx± input. When TXCKSELx = 1 and TXRATEx = 1, the Transmit Data Inputs are captured using both the rising and falling edges of REFCLKx. TXRATEx = 1 and SPDSELx is LOW, is an invalid state and this combination is reserved. |
| RFENA<br>RFENB<br>RFENC<br>RFEND                                 | <b>Reframe Enable.</b> The initialization value of the RFENx latch = 1. RFENx selects if the receiver framer is enabled or disabled. When RFENx = 1, the associated channel's framer is enabled to frame per the presently enabled framing mode and selected framing character. When RFENx = 0, the associated channel's framer is disabled, and no received bits alters the frame offset.   |
| RXPLLPDA<br>RXPLLPDB<br>RXPLLPDC<br>RXPLLPDD                     | <b>Receive Channel Enable.</b> The initialization value of the RXPLLPDx latch = 0. RXPLLPDx selects if the associated receive channel is enabled or powered-down. When RXPLLPDx = 0, the associated PLL and analog circuitry is powered-down. When RXPLLPDx = 1, the associated PLL and analog circuitry is enabled.   |
| RXBISTA<br>RXBISTB<br>RXBISTC<br>RXBISTD                         | <b>Receive Bist Disabled.</b> The initialization value of the RXBISTx latch = 1. RXBISTx selects if receive BIST is disabled or enabled. When RXBISTx = 1, the receiver BIST function is disabled. When RXBISTx = 0, the receive BIST function is enabled.   |
| TXBISTA<br>TXBISTB<br>TXBISTC<br>TXBISTD                         | <b>Transmit Bist Disabled.</b> The initialization value of the TXBISTx latch = 1. TXBISTx selects if the transmit BIST is disabled or enabled. When TXBISTx = 1, the transmit BIST function is disabled. When TXBISTx = 0, the transmit BIST function is enabled.  |
| OE2A<br>OE2B<br>OE2C<br>OE2D                                     | <b>Secondary Differential Serial Data Output Driver Enable.</b> The initialization value of the OE2x latch = 0. OE2x selects if the OUT2± secondary differential output drivers are enabled or disabled. When OE2x = 1, the associated serial data output driver is enabled allowing data to be transmitted from the transmit shifter. When OE2x = 0, the associated serial data output driver is disabled. When a driver is disabled via the configuration interface, it is internally powered down to reduce device power. If both serial drivers for a channel are in this disabled state, the associated internal logic for that channel is also powered down. A device reset (RESET sampled LOW) disables all output drivers.   |

**Table 9. Device Configuration and Control Latch Descriptions** (continued)

| Name                                     | Signal Description   |
|--|--|
| OE1A<br>OE1B<br>OE1C<br>OE1D             | <b>Primary Differential Serial Data Output Driver Enable.</b> The initialization value of the OE1x latch = 0. OE1x selects if the OUT1± primary differential output drivers are enabled or disabled. When OE1x = 1, the associated serial data output driver is enabled allowing data to be transmitted from the transmit shifter. When OE1x = 0, the associated serial data output driver is disabled. When a driver is disabled via the configuration interface, it is internally powered down to reduce device power. If both serial drivers for a channel are in this disabled state, the associated internal logic for that channel is also powered down. A device reset (RESET sampled LOW) disables all output drivers. |
| PABRSTA<br>PABRSTB<br>PABRSTC<br>PABRSTD | <b>Transmit Clock Phase Alignment Buffer Reset.</b> The initialization value of the PABRSTx latch = 1. The PABRSTx is used to re-center the Transmit Phase Align Buffer. When the configuration latch PABRSTx is written as a 0, the phase of the TXCLKx input clock relative to its associated REFCLKx± is initialized. PABRST is an asynchronous input, but is sampled by each TXCLKx↑ to synchronize it to the internal clock domain. PABRSTx is a self clearing latch. This eliminates the requirement of writing a 1 to complete the initialization of the Phase Alignment Buffer.  |
| GLEN[11..0]                              | <b>Global Enable.</b> The initialization value of the GLENx latch = 1. The GLENx is used to reconfigure several channels simultaneously in applications where several channels may have the same configuration. When GLENx = 1 for a given address, that address is allowed to participate in a global configuration. When GLENx = 0 for a given address, that address is disabled from participating in a global configuration.   |
| FGLEN[2..0]                              | <b>Force Global Enable.</b> The initialization value of the FGLENx latch is NA. The FGLENx latch forces a Global Enable no matter what the setting is on the GLENx latch. If FGLENx = 1 for the associated Global channel, FGLEN forces the global update of the target latch banks.   |

#### Device Configuration Strategy

The following is a series of ordered events needed to load the configuration latches on a per channel basis:

1. Pulse  $\overline{\text{RESET}}$  Low after device power-up. This operation resets all four channels.
2. Set the static receiver latch bank for the target channel. May be performed using a global operation, if the application permits it. [Optional step if the default settings match the desired configuration.]
3. Set the static transmitter latch bank for the target channel. May be performed using a global operation, if the application permits it. [Optional step if the default settings match the desired configuration.]

4. Set the dynamic bank of latches for the target channel. Enable the Receive PLLs and transmit channels. May be performed using a global operation, if the application permits it. [Required step.]
5. Reset the Phase Alignment Buffer for the target channel. May be performed using a global operation, if the application permits it. [Optional if phase align buffer is bypassed.]

When a receive channel is configured with the decoder bypassed and the receive clock selected as recovered clock in half-rate mode (DECBYPx = 0, RXRATEx = 1, RXCKSELx = 0), the channel cannot be dynamically reconfigured to enable the decoder with RXCLKx selected as the REFCLKx (DECBYPx = 1, RXCKSELx = 1). If such a change is desired, a global reset should be performed and all channels should be reconfigured to the desired settings.



**Table 10. Device Control Latch Configuration Table**

| ADDR          | Channel | Type | DATA7        | DATA6        | DATA5        | DATA4        | DATA3    | DATA2     | DATA1    | DATA0  | Reset Value |
|---------------|---------|------|--------------|--------------|--------------|--------------|----------|-----------|----------|--------|-------------|
| 0<br>(0000b)  | A       | S    | RFMODEA[1]   | RFMODEA[0]   | FRAMCHARA    | DECMODEA     | DECBYPA  | RXCKSELA  | RXRATEA  | GLEN0  | 10111111    |
| 1<br>(0001b)  | A       | S    | SDASEL2A[1]  | SDASEL2A[0]  | SDASEL1A[1]  | SDASEL1A[0]  | ENCBYPA  | TXCKSELA  | TXRATEA  | GLEN1  | 10101101    |
| 2<br>(0010b)  | A       | D    | RFENA        | RXPLLPDA     | RXBISTA      | TXBISTA      | OE2A     | OE1A      | PABRSTA  | GLEN2  | 10110011    |
| 3<br>(0011b)  | B       | S    | RFMODEB[1]   | RFMODEB[0]   | FRAMCHARB    | DECMODEB     | DECBYPB  | RXCKSELB  | RXRATEB  | GLEN3  | 10111111    |
| 4<br>(0100b)  | B       | S    | SDASEL2B[1]  | SDASEL2B[0]  | SDASEL1B[1]  | SDASEL1B[0]  | ENCBYPB  | TXCKSELB  | TXRATEB  | GLEN4  | 10101101    |
| 5<br>(0101b)  | B       | D    | RFENB        | RXPLLPDB     | RXBISTB      | TXBISTB      | OE2B     | OE1B      | PABRSTB  | GLEN5  | 10110011    |
| 6<br>(0110b)  | C       | S    | RFMODEC[1]   | RFMODEC[0]   | FRAMCHARC    | DECMODEC     | DECBYPC  | RXCKSELC  | RXRATEC  | GLEN6  | 10111111    |
| 7<br>(0111b)  | C       | S    | SDASEL2C[1]  | SDASEL2C[0]  | SDASEL1C[1]  | SDASEL1C[0]  | ENCBYPC  | TXCKSELC  | TXRATEC  | GLEN7  | 10101101    |
| 8<br>(1000b)  | C       | D    | RFENC        | RXPLLPDC     | RXBISTC      | TXBISTC      | OE2C     | OE1C      | PABRSTC  | GLEN8  | 10110011    |
| 9<br>(1001b)  | D       | S    | RFMODED[1]   | RFMODED[0]   | FRAMCHARD    | DECMODED     | DECBYPD  | RXCKSELD  | RXRATE D | GLEN9  | 10111111    |
| 10<br>(1010b) | D       | S    | SDASEL2D[1]  | SDASEL2D[0]  | SDASEL1D[1]  | SDASEL1D[0]  | ENCBYPD  | TXCKSELD  | TXRATED  | GLEN10 | 10101101    |
| 11<br>(1011b) | D       | D    | RFEND        | RXPLLPDD     | RXBISTD      | TXBISTD      | OE2D     | OE1D      | PABRSTD  | GLEN11 | 10110011    |
| 12<br>(1100b) | GLOBAL  | S    | RFMODEGL[1]  | RFMODEGL[0]  | FRAMCHARGL   | DECMODEGL    | DECBYPGL | RXCKSELGL | RXRATEGL | FGLEN0 | N/A         |
| 13<br>(1101b) | GLOBAL  | S    | SDASEL2GL[1] | SDASEL2GL[0] | SDASEL1GL[1] | SDASEL1GL[0] | ENCBPGL  | TXCKSELGL | TXRATEGL | FGLEN1 | N/A         |
| 14<br>(1110b) | GLOBAL  | D    | RFENGL       | RXPLLPDGL    | RXBISTGL     | TXBISTGL     | OE2GL    | OE1GL     | PABRSTGL | FGLEN2 | N/A         |
| 15<br>(1111b) | MASK    | D    | D7           | D6           | D5           | D4           | D3       | D2        | D1       | D0     | 11111111    |

**JTAG Support**

The CYP(V)(W)15G0403DXB contains a JTAG port to allow system level diagnosis of device interconnect. Of the available JTAG modes, boundary scan, and bypass are supported. This capability is present only on the LVTTTL inputs and outputs and the REFCLKx± clock input. The high-speed serial inputs and outputs are not part of the JTAG test chain.

**3-Level Select Inputs**

Each 3-Level select inputs reports as two bits in the scan register. These bits report the LOW, MID, and HIGH state of the associated input as 00, 10, and 11 respectively

**JTAG ID**

The JTAG device ID for the CYP(V)(W)15G0403DXB is '0C810069'x.

**Table 11. Receive Character Status Bits**

| RXSTx[2:0] | Priority | Description   |  |
|------------|----------|---|--|
|            |          | Normal Status   | Receive BIST Status<br>(Receive BIST = Enabled)  |
| 000        | 7        | <b>Normal character received.</b> The valid Data character on the output bus meets all the formatting requirements of Data characters listed in <i>Table 15</i> .   | <b>BIST Data Compare.</b> Character compared correctly.  |
| 001        | 7        | <b>Special code detected.</b> The valid special character on the output bus meets all the formatting requirements of Special Code characters listed in <i>Table 16</i> , but is not the presently selected framing character or a decoder violation indication. | <b>BIST Command Compare.</b> Character compared correctly.   |
| 010        | 2        | <b>Receive Elasticity buffer underrun/overflow error.</b> The receive buffer was not able to add/drop a K28.5 or framing character  | <b>BIST Last Good.</b> Last Character of BIST sequence detected and valid.   |
| 011        | 5        | <b>Framing character detected.</b> This indicates that a character matching the patterns identified as a framing character (as selected by FRAMCHARx) was detected. The decoded value of this character is present in the associated output bus.                |  |
| 100        | 4        | <b>Codeword violation.</b> The character on the output bus is a C0.7. This indicates that the received character cannot be decoded into any valid character.  | <b>BIST Last Bad.</b> Last Character of BIST sequence detected invalid.  |
| 101        | 1        | <b>Loss of sync.</b> This indicates a PLL Out of Lock condition   | <b>BIST Start.</b> Receive BIST is enabled on this channel, but character compares have not yet commenced. This also indicates a PLL Out of Lock condition, and Elasticity Buffer overflow/underflow conditions. |
| 110        | 6        | <b>Running disparity error.</b> The character on the output bus is a C4.7, C1.7, or C2.7.   | <b>BIST Error.</b> While comparing characters, a mismatch was found in one or more of the decoded character bits.  |
| 111        | 3        | Reserved  | <b>BIST Wait.</b> The receiver is comparing characters. but has not yet found the start of BIST character to enable the LFSR.  |

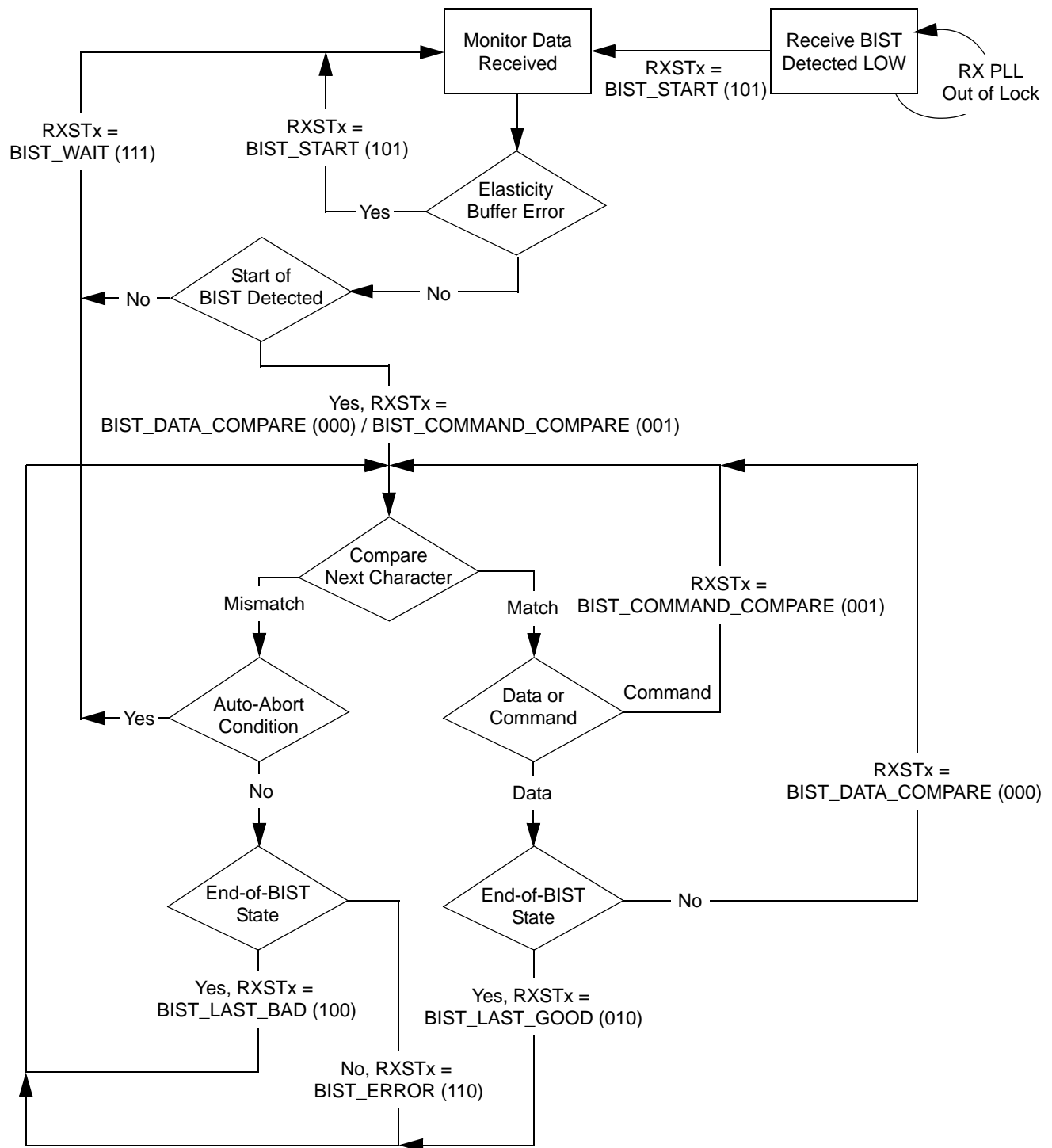


Figure 2. Receive BIST State Machine





### Maximum Ratings

Above which the useful life may be impaired. User guidelines only, not tested

|  |                          |
|--|--------------------------|
| Storage Temperature .....                                  | -65°C to +150°C          |
| Ambient Temperature with Power Applied.....                | -55°C to +125°C          |
| Supply Voltage to Ground Potential .....                   | -0.5V to +3.8V           |
| DC Voltage Applied to LVTTTL Outputs in High-Z State ..... | -0.5V to $V_{CC} + 0.5V$ |
| Output Current into LVTTTL Outputs (LOW).....              | 60 mA                    |
| DC Input Voltage.....                                      | -0.5V to $V_{CC} + 0.5V$ |

Static Discharge Voltage..... > 2000 V (per MIL-STD-883, Method 3015)

Latch-up Current..... > 200 mA

### Power-up Requirements

The CYP(V)(W)15G0403DXB requires one power-supply. The Voltage on any input or I/O pin cannot exceed the power pin during power-up.

### Operating Range

| Range      | Ambient Temperature | $V_{CC}$  |
|------------|---------------------|-----------|
| Commercial | 0°C to +70°C        | +3.3V ±5% |
| Industrial | -40°C to +85°C      | +3.3V ±5% |

### CYP(V)(W)15G0403DXB DC Electrical Characteristics

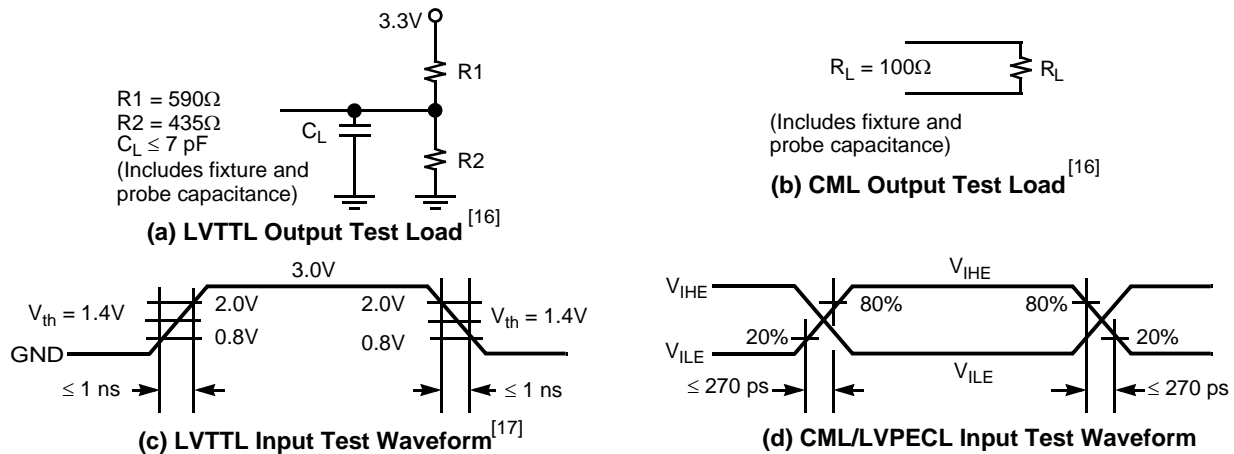
| Parameter  | Description                                | Test Conditions                                   | Min.            | Max.            | Unit |
|--|--|---|-----------------|-----------------|------|
| <b>LVTTTL-compatible Outputs</b>   |  |   |                 |                 |      |
| $V_{OHT}$  | Output HIGH Voltage                        | $I_{OH} = -4 \text{ mA}$ , $V_{CC} = \text{Min.}$ | 2.4             |                 | V    |
| $V_{OLT}$  | Output LOW Voltage                         | $I_{OL} = 4 \text{ mA}$ , $V_{CC} = \text{Min.}$  |                 | 0.4             | V    |
| $I_{OST}$  | Output Short Circuit Current               | $V_{OUT} = 0V^{[10]}$ , $V_{CC} = 3.3V$           | -20             | -100            | mA   |
| $I_{OZL}$  | High-Z Output Leakage Current              | $V_{OUT} = 0V$ , $V_{CC}$                         | -20             | 20              | µA   |
| <b>LVTTTL-compatible Inputs</b>  |  |   |                 |                 |      |
| $V_{IHT}$  | Input HIGH Voltage                         |   | 2.0             | $V_{CC} + 0.3$  | V    |
| $V_{ILT}$  | Input LOW Voltage                          |   | -0.5            | 0.8             | V    |
| $I_{IHT}$  | Input HIGH Current                         | REFCLKx Input, $V_{IN} = V_{CC}$                  |                 | 1.5             | mA   |
|  |  | Other Inputs, $V_{IN} = V_{CC}$                   |                 | +40             | µA   |
| $I_{ILT}$  | Input LOW Current                          | REFCLKx Input, $V_{IN} = 0.0V$                    |                 | -1.5            | mA   |
|  |  | Other Inputs, $V_{IN} = 0.0V$                     |                 | -40             | µA   |
| $I_{IHPDT}$  | Input HIGH Current with internal pull-down | $V_{IN} = V_{CC}$                                 |                 | +200            | µA   |
| $I_{ILPUT}$  | Input LOW Current with internal pull-up    | $V_{IN} = 0.0V$                                   |                 | -200            | µA   |
| <b>LVDIFF Inputs: REFCLKx±</b>   |  |   |                 |                 |      |
| $V_{DIFF}^{[11]}$  | Input Differential Voltage                 |   | 400             | $V_{CC}$        | mV   |
| $V_{IHHP}$   | Highest Input HIGH Voltage                 |   | 1.2             | $V_{CC}$        | V    |
| $V_{ILLP}$   | Lowest Input LOW voltage                   |   | 0.0             | $V_{CC}/2$      | V    |
| $V_{COMREF}^{[12]}$  | Common Mode Range                          |   | 1.0             | $V_{CC} - 1.2V$ | V    |
| <b>3-Level Inputs</b>  |  |   |                 |                 |      |
| $V_{IHH}$  | Three-Level Input HIGH Voltage             | $\text{Min.} \leq V_{CC} \leq \text{Max.}$        | $0.87 * V_{CC}$ | $V_{CC}$        | V    |
| $V_{IMM}$  | Three-Level Input MID Voltage              | $\text{Min.} \leq V_{CC} \leq \text{Max.}$        | $0.47 * V_{CC}$ | $0.53 * V_{CC}$ | V    |
| $V_{ILL}$  | Three-Level Input LOW Voltage              | $\text{Min.} \leq V_{CC} \leq \text{Max.}$        | 0.0             | $0.13 * V_{CC}$ | V    |
| $I_{IHH}$  | Input HIGH Current                         | $V_{IN} = V_{CC}$                                 |                 | 200             | µA   |
| $I_{IMM}$  | Input MID current                          | $V_{IN} = V_{CC}/2$                               | -50             | 50              | µA   |
| $I_{ILL}$  | Input LOW current                          | $V_{IN} = \text{GND}$                             |                 | -200            | µA   |
| <b>Differential CML Serial Outputs: OUTA1±, OUTA2±, OUTB1±, OUTB2±, OUTC1±, OUTC2±, OUTD1±, OUTD2±</b> |  |   |                 |                 |      |
| $V_{OHC}$  | Output HIGH Voltage ( $V_{CC}$ Referenced) | 100Ω differential load                            | $V_{CC} - 0.5$  | $V_{CC} - 0.2$  | V    |
|  |  | 150Ω differential load                            | $V_{CC} - 0.5$  | $V_{CC} - 0.2$  | V    |

**Notes:**

- Tested one output at a time, output shorted for less than one second, less than 10% duty cycle.
- This is the minimum difference in voltage between the true and complement inputs required to ensure detection of a logic-1 or logic-0. A logic-1 exists when the true (+) input is more positive than the complement (-) input. A logic-0 exists when the complement (-) input is more positive than true (+) input.
- The common mode range defines the allowable range of REFCLKx+ and REFCLKx- when REFCLKx+ = REFCLKx-. This marks the zero-crossing between the true and complement inputs as the signal switches between a logic-1 and a logic-0.

**CYP(V)(W)15G0403DXB DC Electrical Characteristics** (continued)

| Parameter   | Description  | Test Conditions   | Min.                  | Max.                  | Unit |
|---|--|---|-----------------------|-----------------------|------|
| V <sub>OLC</sub>  | Output LOW Voltage<br>(V <sub>CC</sub> Referenced) | 100Ω differential load  | V <sub>CC</sub> - 1.4 | V <sub>CC</sub> - 0.7 | V    |
|   |  | 150Ω differential load  | V <sub>CC</sub> - 1.4 | V <sub>CC</sub> - 0.7 | V    |
| V <sub>ODIF</sub>   | Output Differential Voltage<br> (OUT+) - (OUT-)    | 100Ω differential load  | 450                   | 900                   | mV   |
|   |  | 150Ω differential load  | 560                   | 1000                  | mV   |
| <b>Differential Serial Line Receiver Inputs: INA1±, INA2±, INB1±, INB2±, INC1±, INC2±, IND1±, IND2±</b> |  |   |                       |                       |      |
| V <sub>DIFFs</sub> <sup>[11]</sup>  | Input Differential Voltage  (IN+) - (IN-)          |   | 100                   | 1200                  | mV   |
| V <sub>IHE</sub>  | Highest Input HIGH Voltage                         |   |                       | V <sub>CC</sub>       | V    |
| V <sub>ILE</sub>  | Lowest Input LOW Voltage                           |   | V <sub>CC</sub> - 2.0 |                       | V    |
| I <sub>IHE</sub>  | Input HIGH Current                                 | V <sub>IN</sub> = V <sub>IHE</sub> Max.                             |                       | 1350                  | μA   |
| I <sub>ILE</sub>  | Input LOW Current                                  | V <sub>IN</sub> = V <sub>ILE</sub> Min.                             | -700                  |                       | μA   |
| V <sub>ICOM</sub> <sup>[13]</sup>   | Common Mode input range                            | ((V <sub>CC</sub> - 2.0V)+0.5)min,<br>(V <sub>CC</sub> - 0.5V) max. | +1.25                 | +3.1                  | V    |
| <b>Power Supply</b>   |  |   | <b>Typ.</b>           | <b>Max.</b>           |      |
| I <sub>CC</sub> <sup>[14, 15]</sup>   | Max Power Supply Current                           | REFCLKx = Commercial  | 910                   | 1270                  | mA   |
|   |  | REFCLKx = Industrial  |                       | 1320                  | mA   |
| I <sub>CC</sub> <sup>[14, 15]</sup>   | Typical Power Supply Current                       | REFCLKx = Commercial  | 900                   | 1270                  | mA   |
|   |  | REFCLKx = Industrial  |                       | 1320                  | mA   |

**AC Test Loads and Waveforms**

**Notes:**

- The common mode range defines the allowable range of INPUT+ and INPUT- when INPUT+ = INPUT-. This marks the zero-crossing between the true and complement inputs as the signal switches between a logic-1 and a logic-0.
- Maximum I<sub>CC</sub> is measured with V<sub>CC</sub> = MAX, RFENx = 0, T<sub>A</sub> = 25°C, with all channels and Serial Line Drivers enabled, sending a continuous alternating 01 pattern, and outputs unloaded.
- Typical I<sub>CC</sub> is measured under similar conditions except with V<sub>CC</sub> = 3.3V, T<sub>A</sub> = 25°C, RFENx = 0, with all channels enabled and one Serial Line Driver per transmit channel sending a continuous alternating 01 pattern. The redundant outputs on each channel are powered down and the parallel outputs are unloaded.
- Cypress uses constant current (ATE) load configurations and forcing functions. This figure is for reference only.
- The LVTTTL switching threshold is 1.4V. All timing references are made relative to where the signal edges cross the threshold voltage.



**CYP(V)(W)15G0403DXB AC Electrical Characteristics**

| Parameter  | Description   | Min.                      | Max                 | Unit |
|--|---|---------------------------|---------------------|------|
| <b>CYP(V)(W)15G0403DXB Transmitter LVTTTL Switching Characteristics Over the Operating Range</b> |   |                           |                     |      |
| $f_{TS}$   | TXCLKx Clock Cycle Frequency  | 19.5                      | 150 <sup>[18]</sup> | MHz  |
| $t_{TXCLK}$  | TXCLKx Period=1/ $f_{TS}$   | 6.66 <sup>[19]</sup>      | 51.28               | ns   |
| $t_{TXCLKH}$ <sup>[20]</sup>   | TXCLKx HIGH Time  | 2.2                       |                     | ns   |
| $t_{TXCLKL}$ <sup>[20]</sup>   | TXCLKx LOW Time   | 2.2                       |                     | ns   |
| $t_{TXCLKR}$ <sup>[20, 21, 22, 23]</sup>   | TXCLKx Rise Time  | 0.2                       | 1.7                 | ns   |
| $t_{TXCLKF}$ <sup>[20, 21, 22, 23]</sup>   | TXCLKx Fall Time  | 0.2                       | 1.7                 | ns   |
| $t_{TXDS}$   | Transmit Data Set-up Time to TXCLKx $\uparrow$ (TXCKSELx $\neq$ 0)                  | 2.2                       |                     | ns   |
| $t_{TXDH}$   | Transmit Data Hold Time from TXCLKx $\uparrow$ (TXCKSELx $\neq$ 0)                  | 1.0                       |                     | ns   |
| $f_{TOS}$  | TXCLKOx Clock Frequency = 1x or 2x REFCLKx Frequency                                | 19.5                      | 150 <sup>[18]</sup> | MHz  |
| $t_{TXCLKO}$   | TXCLKOx Period=1/ $f_{TOS}$   | 6.66 <sup>[19]</sup>      | 51.28               | ns   |
| $t_{TXCLKOD}$  | TXCLKO Duty Cycle centered at 60% HIGH time   | -1.9                      | 0                   | ns   |
| <b>CYP(V)(W)15G0403DXB Receiver LVTTTL Switching Characteristics Over the Operating Range</b>    |   |                           |                     |      |
| $f_{RS}$   | RXCLKx $\pm$ Clock Output Frequency   | 9.75                      | 150 <sup>[18]</sup> | MHz  |
| $t_{RXCLKP}$   | RXCLKx $\pm$ Period = 1/ $f_{RS}$   | 6.66 <sup>[19]</sup>      | 102.56              | ns   |
| $t_{RXCLKD}$   | RXCLKx $\pm$ Duty Cycle Centered at 50% (Full Rate and Half Rate when RXCKSELx = 0) | -1.0                      | +1.0                | ns   |
| $t_{RXCLKR}$ <sup>[20]</sup>   | RXCLKx $\pm$ Rise Time  | 0.3                       | 1.2                 | ns   |
| $t_{RXCLKF}$ <sup>[20]</sup>   | RXCLKx $\pm$ Fall Time  | 0.3                       | 1.2                 | ns   |
| $t_{RXDV-}$ <sup>[24]</sup>  | Status and Data Valid Time to RXCLKx $\pm$ (RXRATEx = 0, RXCKSELx = 0) (Full Rate)  | 5UI - 2.0 <sup>[25]</sup> |                     | ns   |
|  | Status and Data Valid Time to RXCLKx $\pm$ (RXRATEx = 1, RXCKSELx = 0) (Half Rate)  | 5UI - 1.3 <sup>[25]</sup> |                     | ns   |
| $t_{RXDV+}$ <sup>[24]</sup>  | Status and Data Valid Time to RXCLKx $\pm$ (RXRATEx = 0, RXCKSELx = 0) (Full Rate)  | 5UI-1.8 <sup>[25]</sup>   |                     | ns   |
|  | Status and Data Valid Time to RXCLKx $\pm$ (RXRATEx = 1, RXCKSELx = 0) (Half Rate)  | 5UI - 2.6 <sup>[25]</sup> |                     | ns   |
| <b>CYP(V)(W)15G0403DXB REFCLKx Switching Characteristics Over the Operating Range</b>            |   |                           |                     |      |
| $f_{REF}$  | REFCLKx Clock Frequency   | 19.5                      | 150 <sup>[18]</sup> | MHz  |
| $t_{REFCLK}$   | REFCLKx Period = 1/ $f_{REF}$   | 6.66 <sup>[19]</sup>      | 51.28               | ns   |
| $t_{REFH}$   | REFCLKx HIGH Time (TXRATEx = 1)(Half Rate)  | 5.9 <sup>[26]</sup>       |                     | ns   |
|  | REFCLKx HIGH Time (TXRATEx = 0)(Full Rate)  | 2.9 <sup>[20]</sup>       |                     | ns   |
| $t_{REFL}$   | REFCLKx LOW Time (TXRATEx = 1)(Half Rate)   | 5.9 <sup>[26]</sup>       |                     | ns   |
|  | REFCLKx LOW Time (TXRATEx = 0)(Full Rate)   | 2.9 <sup>[20]</sup>       |                     | ns   |
| $t_{REFD}$ <sup>[27]</sup>   | REFCLKx Duty Cycle  | 30                        | 70                  | %    |
| $t_{REFR}$ <sup>[20, 21, 22, 23]</sup>   | REFCLKx Rise Time (20%–80%)   |                           | 2                   | ns   |
| $t_{REFF}$ <sup>[20, 21, 22, 23]</sup>   | REFCLKx Fall Time (20%–80%)   |                           | 2                   | ns   |

**Notes:**

18. This parameter is 154 MHz for CYW15G0403DXB.
19. This parameter is 6.49 ns for CYW15G0403DXB.
20. Tested initially and after any design or process changes that may affect these parameters, but not 100% tested.
21. The ratio of rise time to falling time must not vary by greater than 2:1.
22. For a given operating frequency, neither rise or fall specification can be greater than 20% of the clock-cycle period or the data sheet maximum time.
23. All transmit AC timing parameters measured with 1 ns typical rise time and fall time.
24. Parallel data output specifications are only valid if all outputs are loaded with similar DC and AC loads.
25. Receiver UI (Unit Interval) is calculated as 1/( $f_{REF} * 20$ ) (when TXRATEx = 1) or 1/( $f_{REF} * 10$ ) (when TXRATEx = 0). In an operating link this is equivalent to  $t_{IB}$ .
26. If REFCLKx is selected as receive interface clock (RXCKSELx=1), then this parameter has to be greater than or equal to 6.3 ns.
27. The duty cycle specification is a simultaneous condition with the  $t_{REFH}$  and  $t_{REFL}$  parameters. This means that at faster character rates the REFCLKx $\pm$  duty cycle cannot be as large as 30%–70%.



**CYP(V)(W)15G0403DXB AC Electrical Characteristics** (continued)

| Parameter  | Description  | Min.                        | Max                 | Unit |      |
|--|--|-----------------------------|---------------------|------|------|
| t <sub>TREFDs</sub>  | Transmit Data Set-up Time to REFCLKx - Full Rate (TXRATE <sub>x</sub> = 0, TXCKSEL <sub>x</sub> = 1)                 | 2.4                         |                     | ns   |      |
|  | Transmit Data Set-up Time to REFCLKx - Half Rate (TXRATE <sub>x</sub> = 1, TXCKSEL <sub>x</sub> = 1)                 | 2.3                         |                     | ns   |      |
| t <sub>TREFDH</sub>  | Transmit Data Hold Time from REFCLKx - Full Rate (TXRATE <sub>x</sub> = 0, TXCKSEL <sub>x</sub> = 1)                 | 1.0                         |                     | ns   |      |
|  | Transmit Data Hold Time from REFCLKx - Half Rate (TXRATE <sub>x</sub> = 1, TXCKSEL <sub>x</sub> = 1)                 | 1.6                         |                     | ns   |      |
| t <sub>RREFDA</sub>  | Receive Data Access Time to REFCLKx (RXCKSEL <sub>x</sub> = 1)   |                             | 9.7 <sup>[28]</sup> | ns   |      |
| t <sub>RREFDW</sub>  | Receive Data Valid Time Window (RXCKSEL <sub>x</sub> = 1)  | 10UI – 5.8                  |                     | ns   |      |
| t <sub>REFxDV–</sub>   | Received Data Valid Time to RXCLK when RXCKSEL <sub>x</sub> = 1 (TXRATE <sub>x</sub> = 0, RXRATE <sub>x</sub> = 0)   | 10UI <sup>[25]</sup> – 6.16 |                     | ns   |      |
|  | Received Data Valid Time to RXCLK when RXCKSEL <sub>x</sub> = 1 (TXRATE <sub>x</sub> = 0, RXRATE <sub>x</sub> = 1)   | 5UI – 2.53 <sup>[29]</sup>  |                     | ns   |      |
|  | Received Data Valid Time to RXCLK when RXCKSEL <sub>x</sub> = 1 (TXRATE <sub>x</sub> = 1)                            | 10UI – 5.86 <sup>[29]</sup> |                     | ns   |      |
| t <sub>REFxDV+</sub>   | Received Data Valid Time from RXCLK when RXCKSEL <sub>x</sub> = 1 (TXRATE <sub>x</sub> = 0, RXRATE <sub>x</sub> = 0) | 1.4                         |                     | ns   |      |
|  | Received Data Valid Time from RXCLK when RXCKSEL <sub>x</sub> = 1 (TXRATE <sub>x</sub> = 0, RXRATE <sub>x</sub> = 1) | 5UI – 1.83 <sup>[29]</sup>  |                     | ns   |      |
|  | Received Data Valid Time from RXCLK when RXCKSEL <sub>x</sub> = 1 (TXRATE <sub>x</sub> = 1)                          | 1.0 <sup>[29]</sup>         |                     | ns   |      |
| t <sub>REFRX</sub> <sup>[30]</sup>   | REFCLKx Frequency Referenced to Received Clock Period  | –0.15                       | +0.15               | %    |      |
| <b>CYP(V)(W)15G0403DXB Bus Configuration Write Timing Characteristics Over the Operating Range</b>     |  |                             |                     |      |      |
| t <sub>DATAH</sub>   | Bus Configuration Data Hold  | 0                           |                     | ns   |      |
| t <sub>DATAS</sub>   | Bus Configuration Data Set-up  | 10                          |                     | ns   |      |
| t <sub>WRENp</sub>   | Bus Configuration WREN Pulse Width   | 10                          |                     | ns   |      |
| <b>CYP(V)(W)15G0403DXB JTAG Test Clock Characteristics Over the Operating Range</b>                    |  |                             |                     |      |      |
| f <sub>TCLK</sub>  | JTAG Test Clock Frequency  |                             | 20                  | MHz  |      |
| t <sub>TCLK</sub>  | JTAG Test Clock Period   | 50                          |                     | ns   |      |
| <b>CYP(V)(W)15G0403DXB Device RESET Characteristics Over the Operating Range</b>                       |  |                             |                     |      |      |
| t <sub>RST</sub>   | Device RESET Pulse Width   | 30                          |                     | ns   |      |
| <b>CYP(V)(W)15G0403DXB Transmit Serial Outputs and TX PLL Characteristics Over the Operating Range</b> |  |                             |                     |      |      |
| Parameter  | Description  | Condition                   | Min.                | Max. | Unit |
| t <sub>B</sub>   | Bit Time   |                             | 5128                | 666  | ps   |
| t <sub>RISE</sub> <sup>[20]</sup>  | CML Output Rise Time 20–80% (CML Test Load)  | SPDSEL <sub>x</sub> = HIGH  | 60                  | 270  | ps   |
|  |  | SPDSEL <sub>x</sub> = MID   | 100                 | 500  | ps   |
|  |  | SPDSEL <sub>x</sub> = LOW   | 180                 | 1000 | ps   |

**Notes:**

28. Since this timing parameter is greater than the minimum time period of REFCLK it sets an upper limit to the frequency in which REFCLKx can be used to clock the receive data out of the output register. For predictable timing, users can use this parameter only if REFCLK period is greater than sum of t<sub>RREFDA</sub> and set-up time of the upstream device. When this condition is not true, RXCLKx± (a buffered or divided version of REFCLK when RXCKSEL<sub>x</sub> = 1) could be used to clock the receive data out of the device.
29. Measured using a 50% duty cycle reference clock.
30. REFCLKx has no phase or frequency relationship with the recovered clock and only acts as a centering reference to reduce clock synchronization time. REFCLKx must be within ±1500 ppm (±0.15%) of the remote transmitter's PLL reference (REFCLKx) frequency. Although transmitting to a HOTLink II receiver channel necessitates the frequency difference between the transmitter and receiver reference clocks to be within ±1500 ppm, the stability of the crystal needs to be within the limits specified by the appropriate standard when transmitting to a remote receiver that is compliant to that standard. For example, to be IEEE 802.3z Gigabit Ethernet compliant, the frequency stability of the crystal needs to be within ±100 ppm.

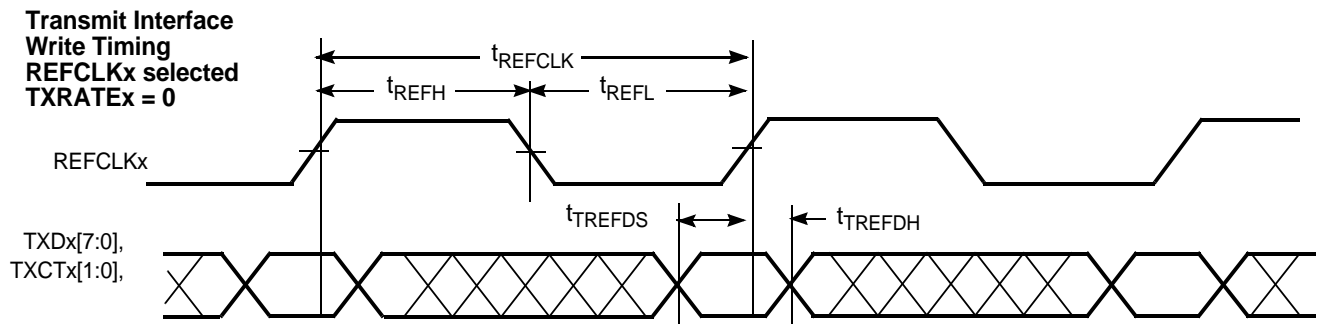
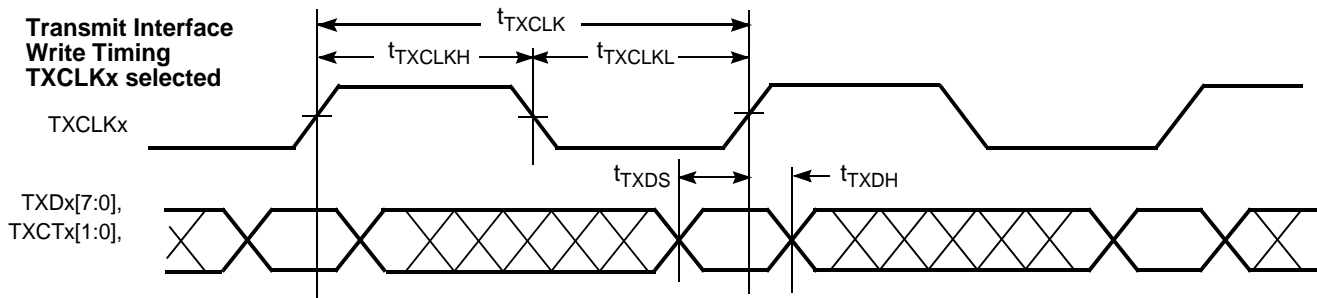
**CYP(V)(W)15G0403DXB AC Electrical Characteristics** (continued)

| Parameter   | Description  | Min.           | Max  | Unit |    |
|---|--|----------------|------|------|----|
| $t_{FALL}^{[20]}$   | CML Output Fall Time 80–20% (CML Test Load)        | SPDSELx = HIGH | 60   | 270  | ps |
|   |  | SPDSELx = MID  | 100  | 500  | ps |
|   |  | SPDSELx = LOW  | 180  | 1000 | ps |
| $t_{DJ}^{[20, 31, 33]}$   | Deterministic Jitter (peak-peak) <sup>[34]</sup>   | IEEE 802.3z    | 27   | ps   |    |
| $t_{RJ}^{[20, 32, 33]}$   | Random Jitter ( $\sigma$ ) <sup>[34]</sup>         | IEEE 802.3z    | 11   | ps   |    |
| $t_{REFJ}^{[20]}$   | REFCLKx jitter tolerance / Phase noise limits      |                | TBD  |      |    |
| $t_{TXLOCK}$  | Transmit PLLx lock to REFCLKx±                     |                | 200  | μs   |    |
| <b>CYP(V)(W)15G0403DXB Receive Serial Inputs and CDR PLL Characteristics Over the Operating Range</b> |  |                |      |      |    |
| $t_{RXLOCK}$  | Receive PLL lock to input data stream (cold start) |                | 376k | UI   |    |
|   | Receive PLL lock to input data stream              |                | 376k | UI   |    |
| $t_{RXUNLOCK}$  | Receive PLL Unlock Rate                            |                | 46   | UI   |    |
| $t_{JTOL}^{[20]}$   | Total Jitter Tolerance <sup>[34]</sup>             | IEEE 802.3z    | 600  | ps   |    |
| $t_{DJTOL}^{[20]}$  | Deterministic Jitter Tolerance <sup>[34]</sup>     | IEEE 802.3z    | 370  | ps   |    |

**Capacitance**<sup>[20]</sup>

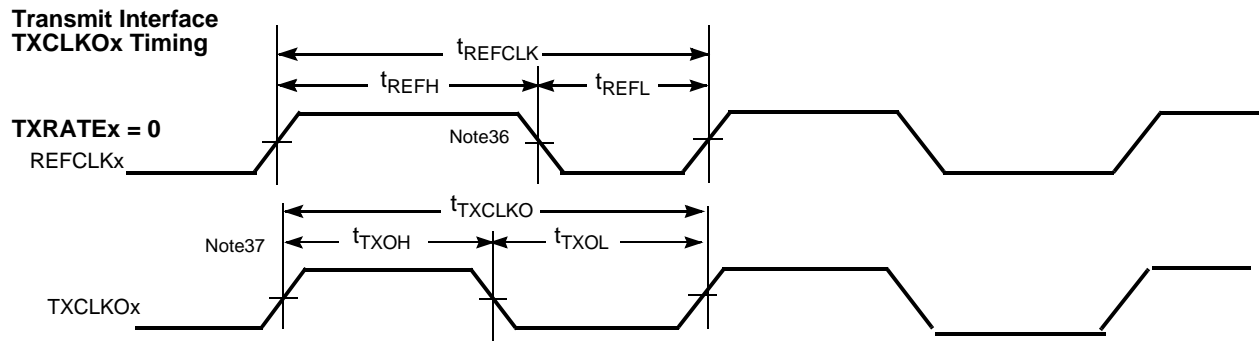
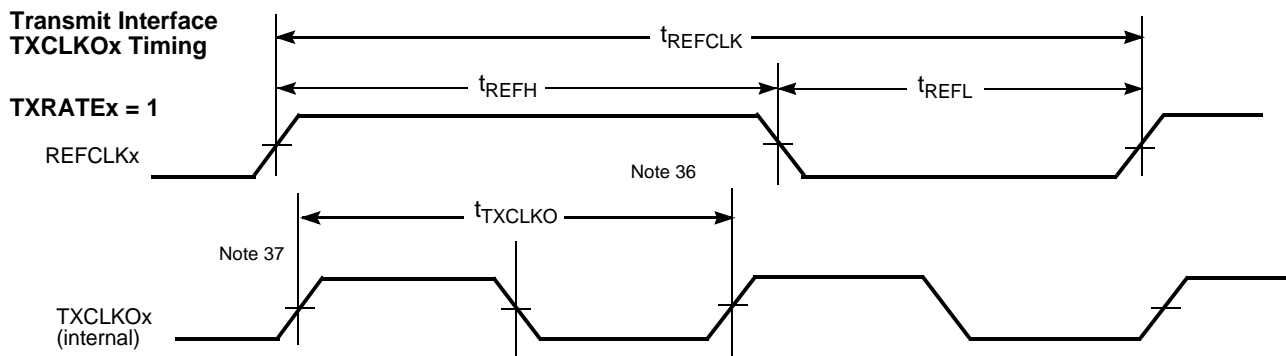
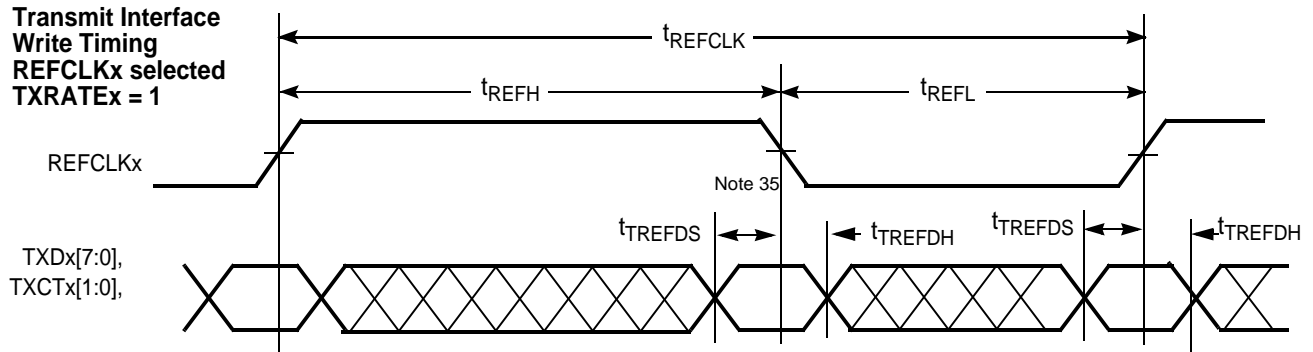
| Parameter    | Description            | Test Conditions  | Max. | Unit |
|--------------|------------------------|--|------|------|
| $C_{INTTL}$  | TTL Input Capacitance  | $T_A = 25^\circ\text{C}$ , $f_0 = 1\text{ MHz}$ , $V_{CC} = 3.3\text{V}$ | 7    | pF   |
| $C_{INPECL}$ | PECL input Capacitance | $T_A = 25^\circ\text{C}$ , $f_0 = 1\text{ MHz}$ , $V_{CC} = 3.3\text{V}$ | 4    | pF   |

**CYP(V)(W)15G0403DXB HOTLink II Transmitter Switching Waveforms**

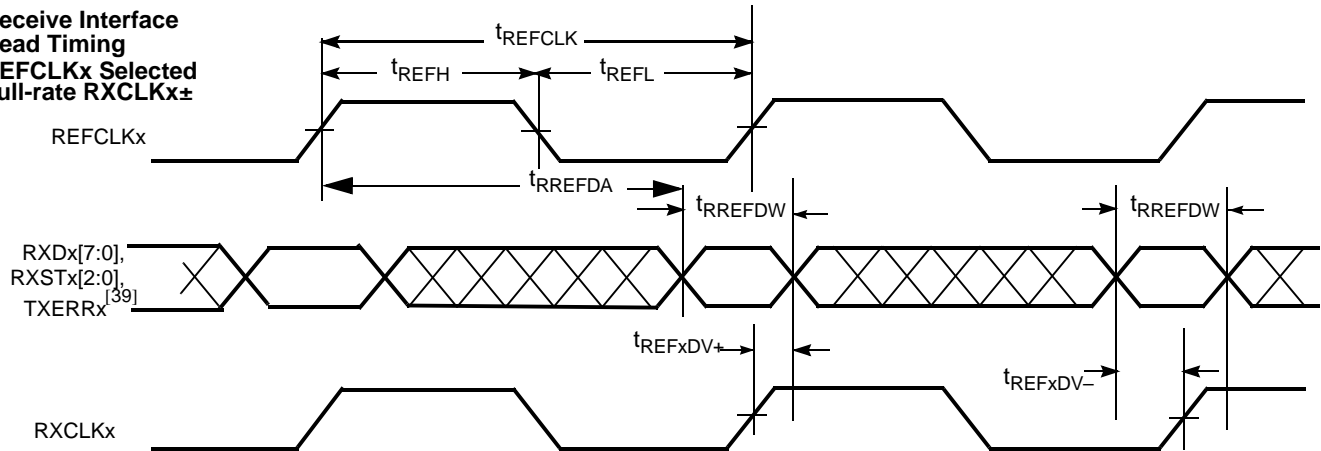
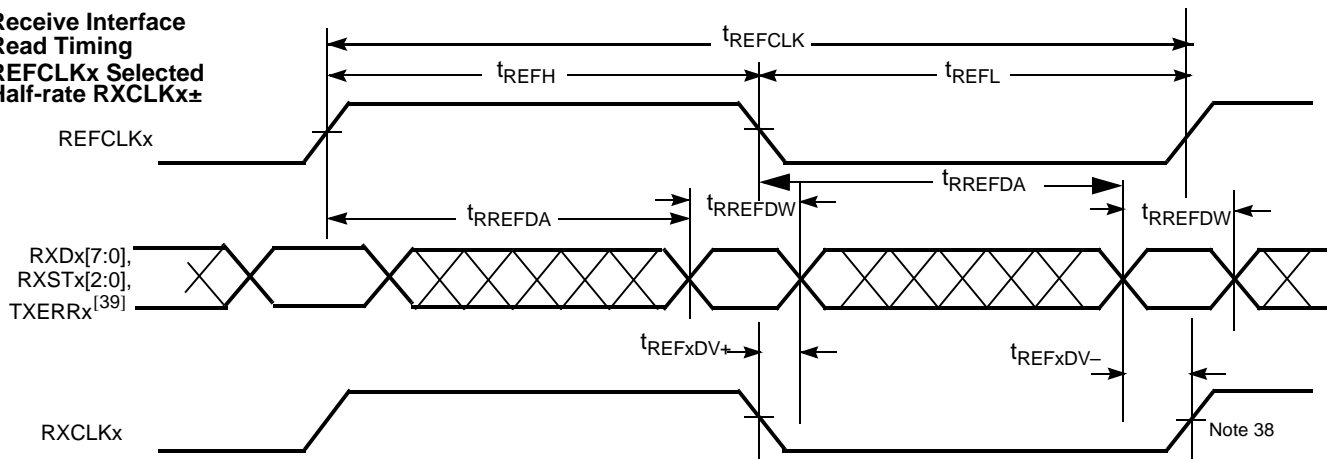
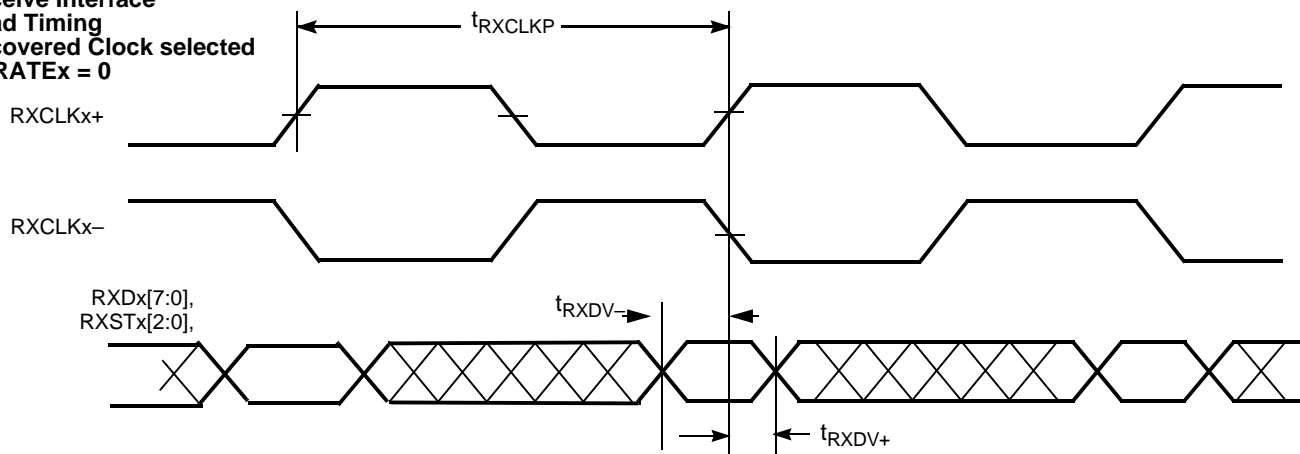


**Notes:**

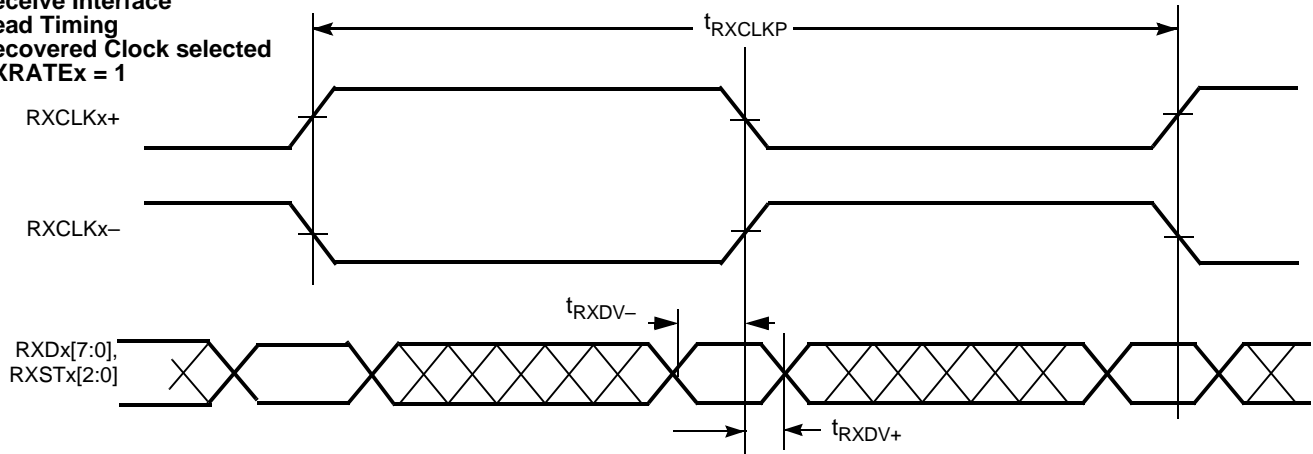
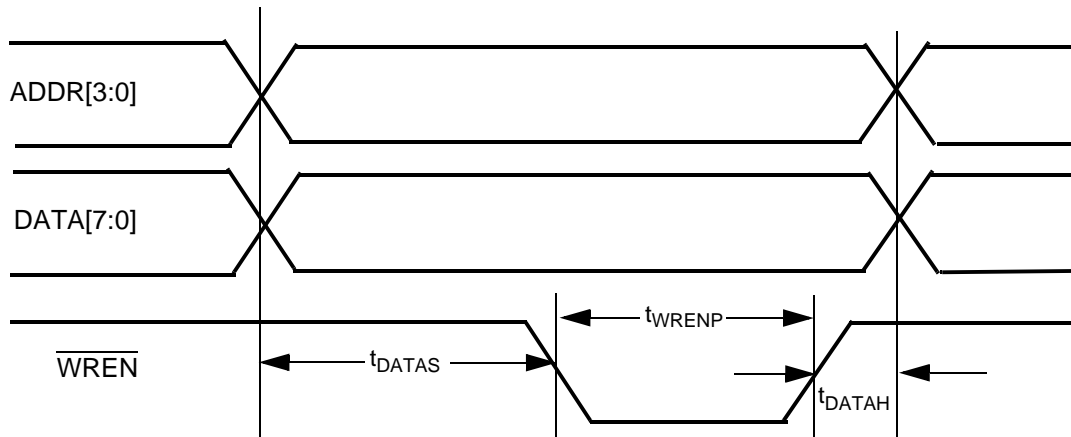
- 31. While sending continuous K28.5s, outputs loaded to a balanced 100Ω load, measured at the cross point of differential outputs, over the operating range.
- 32. While sending continuous K28.7s, after 100,000 samples measured at the cross point of differential outputs, time referenced to REFCLKx± input, over the operating range.
- 33. Total jitter is calculated at an assumed BER of 1E -12. Hence: Total Jitter ( $t_j$ ) = ( $t_{RJ} * 14$ ) +  $t_{DJ}$ .
- 34. Also meets all Jitter Generation and Jitter Tolerance requirements as specified by SMPTE 259M, SMPTE 292M, ESCON, FICON, Fibre Channel, and DVB-ASI.

**CYP(V)(W)15G0403DXB HOTLink II Transmitter Switching Waveforms (continued)**

**Notes:**

35. When REFCLKx± is configured for half-rate operation (TXRATE = 1) and data is captured using REFCLKx instead of a TXCLKx clock. Data is captured using both the rising and falling edges of REFCLKx.
36. The TXCLKOx output remains at the character rate regardless of the state of TXRATE and does not follow the duty cycle of REFCLKx±.
37. The rising edge of TXCLKOx output has no direct phase relationship to the REFCLKx± input.

**Switching Waveforms for the CYP(V)(W)15G0403DXB HOTLink II Receiver**
**Receive Interface  
 Read Timing  
 REFCLKx Selected  
 Full-rate RXCLKx±**

**Receive Interface  
 Read Timing  
 REFCLKx Selected  
 Half-rate RXCLKx±**

**Receive Interface  
 Read Timing  
 Recovered Clock selected  
 RXRATEx = 0**

**Note:**

38. When operated with a half-rate REFCLKx±, the set-up and hold specifications for data relative to RXCLKx are relative to both rising and falling edges of the respective clock output
39. TXERRx is synchronous to RXCLKx only when RXCLKx is selected as REFCLK.

**Switching Waveforms for the CYP(V)(W)15G0403DXB HOTLink II Receiver**
**Receive Interface  
 Read Timing  
 Recovered Clock selected  
 RXRATE<sub>x</sub> = 1**

**Bus Configuration  
 Write Timing**






**Table 12. Package Coordinate Signal Allocation**

| Ball ID | Signal Name | Signal Type  | Ball ID | Signal Name | Signal Type    | Ball ID | Signal Name | Signal Type  |
|---------|-------------|--------------|---------|-------------|----------------|---------|-------------|--------------|
| A01     | INC1-       | CML IN       | C07     | ULCC        | LVTTTL IN PU   | F17     | NC          | NO CONNECT   |
| A02     | OUTC1-      | CML OUT      | C08     | GND         | GROUND         | F18     | RXSTB[1]    | LVTTTL OUT   |
| A03     | INC2-       | CML IN       | C09     | DATA[7]     | LVTTTL IN PU   | F19     | TXCLKOB     | LVTTTL OUT   |
| A04     | OUTC2-      | CML OUT      | C10     | DATA[5]     | LVTTTL IN PU   | F20     | RXSTB[0]    | LVTTTL OUT   |
| A05     | VCC         | POWER        | C11     | DATA[3]     | LVTTTL IN PU   | G01     | TXDC[7]     | LVTTTL IN    |
| A06     | IND1-       | CML IN       | C12     | DATA[1]     | LVTTTL IN PU   | G02     | WREN        | LVTTTL IN PU |
| A07     | OUTD1-      | CML OUT      | C13     | GND         | GROUND         | G03     | TXDC[4]     | LVTTTL IN    |
| A08     | GND         | GROUND       | C14     | NC          | NO CONNECT     | G04     | TXDC[1]     | LVTTTL IN    |
| A09     | IND2-       | CML IN       | C15     | SPDSELD     | 3-LEVEL SEL    | G17     | SPDSELB     | 3-LEVEL SEL  |
| A10     | OUTD2-      | CML OUT      | C16     | VCC         | POWER          | G18     | LPENC       | LVTTTL IN PD |
| A11     | INA1-       | CML IN       | C17     | LDTDEN      | LVTTTL IN PU   | G19     | SPDSELA     | 3-LEVEL SEL  |
| A12     | OUTA1-      | CML OUT      | C18     | TRST        | LVTTTL IN PU   | G20     | RXDB[1]     | LVTTTL OUT   |
| A13     | GND         | GROUND       | C19     | LPEND       | LVTTTL IN PD   | H01     | GND         | GROUND       |
| A14     | INA2-       | CML IN       | C20     | TDO         | LVTTTL 3-S OUT | H02     | GND         | GROUND       |
| A15     | OUTA2-      | CML OUT      | D01     | TCLK        | LVTTTL IN PD   | H03     | GND         | GROUND       |
| A16     | VCC         | POWER        | D02     | RESET       | LVTTTL IN PU   | H04     | GND         | GROUND       |
| A17     | INB1-       | CML IN       | D03     | INSELD      | LVTTTL IN      | H17     | GND         | GROUND       |
| A18     | OUTB1-      | CML OUT      | D04     | INSELA      | LVTTTL IN      | H18     | GND         | GROUND       |
| A19     | INB2-       | CML IN       | D05     | VCC         | POWER          | H19     | GND         | GROUND       |
| A20     | OUTB2-      | CML OUT      | D06     | ULCA        | LVTTTL IN PU   | H20     | GND         | GROUND       |
| B01     | INC1+       | CML IN       | D07     | SPDSELC     | 3-LEVEL SEL    | J01     | TXCTC[1]    | LVTTTL IN    |
| B02     | OUTC1+      | CML OUT      | D08     | GND         | GROUND         | J02     | TXDC[5]     | LVTTTL IN    |
| B03     | INC2+       | CML IN       | D09     | DATA[6]     | LVTTTL IN PU   | J03     | TXDC[2]     | LVTTTL IN    |
| B04     | OUTC2+      | CML OUT      | D10     | DATA[4]     | LVTTTL IN PU   | J04     | TXDC[3]     | LVTTTL IN    |
| B05     | VCC         | POWER        | D11     | DATA[2]     | LVTTTL IN PU   | J17     | RXSTB[2]    | LVTTTL OUT   |
| B06     | IND1+       | CML IN       | D12     | DATA[0]     | LVTTTL IN PU   | J18     | RXDB[0]     | LVTTTL OUT   |
| B07     | OUTD1+      | CML OUT      | D13     | GND         | GROUND         | J19     | RXDB[5]     | LVTTTL OUT   |
| B08     | GND         | GROUND       | D14     | LPENB       | LVTTTL IN PD   | J20     | RXDB[2]     | LVTTTL OUT   |
| B09     | IND2+       | CML IN       | D15     | ULCB        | LVTTTL IN PU   | K01     | RXDC[2]     | LVTTTL OUT   |
| B10     | OUTD2+      | CML OUT      | D16     | VCC         | POWER          | K02     | REFCLKC-    | PECL IN      |
| B11     | INA1+       | CML IN       | D17     | LPENA       | LVTTTL IN PD   | K03     | TXCTC[0]    | LVTTTL IN    |
| B12     | OUTA1+      | CML OUT      | D18     | LTEN1       | LVTTTL IN PD   | K04     | TXCLKC      | LVTTTL IN PD |
| B13     | GND         | GROUND       | D19     | SCANEN2     | LVTTTL IN PD   | K17     | RXDB[3]     | LVTTTL OUT   |
| B14     | INA2+       | CML IN       | D20     | TMEN3       | LVTTTL IN PD   | K18     | RXDB[4]     | LVTTTL OUT   |
| B15     | OUTA2+      | CML OUT      | E01     | VCC         | POWER          | K19     | RXDB[7]     | LVTTTL OUT   |
| B16     | VCC         | POWER        | E02     | VCC         | POWER          | K20     | LFIB        | LVTTTL OUT   |
| B17     | INB1+       | CML IN       | E03     | VCC         | POWER          | L01     | RXDC[3]     | LVTTTL OUT   |
| B18     | OUTB1+      | CML OUT      | E04     | VCC         | POWER          | L02     | REFCLKC+    | PECL IN      |
| B19     | INB2+       | CML IN       | E17     | VCC         | POWER          | L03     | LFIC        | LVTTTL OUT   |
| B20     | OUTB2+      | CML OUT      | E18     | VCC         | POWER          | L04     | TXDC[6]     | LVTTTL IN    |
| C01     | TDI         | LVTTTL IN PU | E19     | VCC         | POWER          | L17     | RXDB[6]     | LVTTTL OUT   |
| C02     | TMS         | LVTTTL IN PU | E20     | VCC         | POWER          | L18     | RXCLKB+     | LVTTTL OUT   |
| C03     | INSELC      | LVTTTL IN    | F01     | RXDC[6]     | LVTTTL OUT     | L19     | RXCLKB-     | LVTTTL OUT   |



**Table 12. Package Coordinate Signal Allocation** (continued)

| Ball ID | Signal Name | Signal Type  | Ball ID | Signal Name | Signal Type  | Ball ID | Signal Name | Signal Type  |
|---------|-------------|--------------|---------|-------------|--------------|---------|-------------|--------------|
| C04     | INSELB      | LVTTTL IN    | F02     | RXDC[7]     | LVTTTL OUT   | L20     | TXDB[6]     | LVTTTL IN    |
| C05     | VCC         | POWER        | F03     | TXDC[0]     | LVTTTL IN    | M01     | RXDC[4]     | LVTTTL OUT   |
| C06     | ULCD        | LVTTTL IN PU | F04     | NC          | NO CONNECT   | M02     | RXDC[5]     | LVTTTL OUT   |
| M03     | NC          | NO CONNECT   | U03     | TXDD[2]     | LVTTTL IN    | W03     | LFID        | LVTTTL OUT   |
| M04     | TXERRC      | LVTTTL OUT   | U04     | TXCTD[1]    | LVTTTL IN    | W04     | RXCLKD-     | LVTTTL OUT   |
| M17     | REFCLKB+    | PECL IN      | U05     | VCC         | POWER        | W05     | VCC         | POWER        |
| M18     | REFCLKB-    | PECL IN      | U06     | RXDD[2]     | LVTTTL OUT   | W06     | RXDD[4]     | LVTTTL OUT   |
| M19     | TXERRB      | LVTTTL OUT   | U07     | RXDD[1]     | LVTTTL OUT   | W07     | RXSTD[1]    | LVTTTL OUT   |
| M20     | TXCLKB      | LVTTTL IN PD | U08     | GND         | GROUND       | W08     | GND         | GROUND       |
| N01     | GND         | GROUND       | U09     | TXCTA[1]    | LVTTTL IN    | W09     | ADDR [3]    | LVTTTL IN PU |
| N02     | GND         | GROUND       | U10     | ADDR [0]    | LVTTTL IN PU | W10     | ADDR [1]    | LVTTTL IN PU |
| N03     | GND         | GROUND       | U11     | REFCLKD-    | PECL IN      | W11     | RXCLKA+     | LVTTTL OUT   |
| N04     | GND         | GROUND       | U12     | TXDA[1]     | LVTTTL IN    | W12     | TXERRA      | LVTTTL OUT   |
| N17     | GND         | GROUND       | U13     | GND         | GROUND       | W13     | GND         | GROUND       |
| N18     | GND         | GROUND       | U14     | TXDA[4]     | LVTTTL IN    | W14     | TXDA[2]     | LVTTTL IN    |
| N19     | GND         | GROUND       | U15     | TXCTA[0]    | LVTTTL IN    | W15     | TXDA[6]     | LVTTTL IN    |
| N20     | GND         | GROUND       | U16     | VCC         | POWER        | W16     | VCC         | POWER        |
| P01     | RXDC[1]     | LVTTTL OUT   | U17     | RXDA[2]     | LVTTTL OUT   | W17     | LFIA        | LVTTTL OUT   |
| P02     | RXDC[0]     | LVTTTL OUT   | U18     | TXCTB[0]    | LVTTTL IN    | W18     | REFCLKA+    | PECL IN      |
| P03     | RXSTC[0]    | LVTTTL OUT   | U19     | RXSTA[2]    | LVTTTL OUT   | W19     | RXDA[4]     | LVTTTL OUT   |
| P04     | RXSTC[1]    | LVTTTL OUT   | U20     | RXSTA[1]    | LVTTTL OUT   | W20     | RXDA[1]     | LVTTTL OUT   |
| P17     | TXDB[5]     | LVTTTL IN    | V01     | TXDD[3]     | LVTTTL IN    | Y01     | TXDD[6]     | LVTTTL IN    |
| P18     | TXDB[4]     | LVTTTL IN    | V02     | TXDD[4]     | LVTTTL IN    | Y02     | TXCLKD      | LVTTTL IN PD |
| P19     | TXDB[3]     | LVTTTL IN    | V03     | TXCTD[0]    | LVTTTL IN    | Y03     | RXDD[7]     | LVTTTL OUT   |
| P20     | TXDB[2]     | LVTTTL IN    | V04     | RXDD[6]     | LVTTTL OUT   | Y04     | RXCLKD+     | LVTTTL OUT   |
| R01     | RXSTC[2]    | LVTTTL OUT   | V05     | VCC         | POWER        | Y05     | VCC         | POWER        |
| R02     | TXCLKOC     | LVTTTL OUT   | V06     | RXDD[3]     | LVTTTL OUT   | Y06     | RXDD[5]     | LVTTTL OUT   |
| R03     | RXCLKC+     | LVTTTL OUT   | V07     | RXSTD[0]    | LVTTTL OUT   | Y07     | RXDD[0]     | LVTTTL OUT   |
| R04     | RXCLKC-     | LVTTTL OUT   | V08     | GND         | GROUND       | Y08     | GND         | GROUND       |
| R17     | TXDB[1]     | LVTTTL IN    | V09     | RXSTD[2]    | LVTTTL OUT   | Y09     | TXCLKOD     | LVTTTL OUT   |
| R18     | TXDB[0]     | LVTTTL IN    | V10     | ADDR [2]    | LVTTTL IN PU | Y10     | NC          | NO CONNECT   |
| R19     | TXCTB[1]    | LVTTTL IN    | V11     | REFCLKD+    | PECL IN      | Y11     | TXCLKA      | LVTTTL IN PD |
| R20     | TXDB[7]     | LVTTTL IN    | V12     | TXCLKOA     | LVTTTL OUT   | Y12     | RXCLKA-     | LVTTTL OUT   |
| T01     | VCC         | POWER        | V13     | GND         | GROUND       | Y13     | GND         | GROUND       |
| T02     | VCC         | POWER        | V14     | TXDA[3]     | LVTTTL IN    | Y14     | TXDA[0]     | LVTTTL IN    |
| T03     | VCC         | POWER        | V15     | TXDA[7]     | LVTTTL IN    | Y15     | TXDA[5]     | LVTTTL IN    |
| T04     | VCC         | POWER        | V16     | VCC         | POWER        | Y16     | VCC         | POWER        |
| T17     | VCC         | POWER        | V17     | RXDA[7]     | LVTTTL OUT   | Y17     | TXERRD      | LVTTTL OUT   |
| T18     | VCC         | POWER        | V18     | RXDA[3]     | LVTTTL OUT   | Y18     | REFCLKA-    | PECL IN      |
| T19     | VCC         | POWER        | V19     | RXDA[0]     | LVTTTL OUT   | Y19     | RXDA[6]     | LVTTTL OUT   |
| T20     | VCC         | POWER        | V20     | RXSTA[0]    | LVTTTL OUT   | Y20     | RXDA[5]     | LVTTTL OUT   |
| U01     | TXDD[0]     | LVTTTL IN    | W01     | TXDD[5]     | LVTTTL IN    |         |             |              |
| U02     | TXDD[1]     | LVTTTL IN    | W02     | TXDD[7]     | LVTTTL IN    |         |             |              |



### X3.230 Codes and Notation Conventions

Information transmitted over a serial link is encoded eight bits at a time into a 10-bit Transmission Character and then sent serially, bit by bit. Information received over a serial link is collected ten bits at a time, and those Transmission Characters that are used for data characters are decoded into the correct eight-bit codes. The 10-bit Transmission Code supports all 256 8-bit combinations. Some of the remaining Transmission Characters (Special Characters) are used for functions other than data transmission.

The primary use of a Transmission Code is to improve the transmission characteristics of a serial link. The encoding defined by the Transmission Code ensures that sufficient transitions are present in the serial bit stream to make clock recovery possible at the Receiver. Such encoding also greatly increases the likelihood of detecting any single or multiple bit errors that may occur during transmission and reception of information. In addition, some Special Characters of the Transmission Code selected by Fibre Channel Standard contain a distinct and easily recognizable bit pattern that assists the receiver in achieving character alignment on the incoming bit stream.

#### Notation Conventions

The documentation for the 8B/10B Transmission Code uses letter notation for the bits in an 8-bit byte. Fibre Channel Standard notation uses a bit notation of A, B, C, D, E, F, G, H for the 8-bit byte for the raw 8-bit data, and the letters a, b, c, d, e, i, f, g, h, j for encoded 10-bit data. There is a correspondence between bit A and bit a, B and b, C and c, D and d, E and e, F and f, G and g, and H and h. Bits i and j are derived, respectively, from (A,B,C,D,E) and (F,G,H).

The bit labeled A in the description of the 8B/10B Transmission Code corresponds to bit 0 in the numbering scheme of the FC-2 specification, B corresponds to bit 1, as shown below.

|                          |   |   |   |   |   |   |   |   |
|--------------------------|---|---|---|---|---|---|---|---|
| FC-2 bit designation—    | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| HOTLink D/Q designation— | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 8B/10B bit designation—  | H | G | F | E | D | C | B | A |

To clarify this correspondence, the following example shows the conversion from an FC-2 Valid Data Byte to a Transmission Character.

FC-2 45H  
Bits: 7654 3210  
0100 0101

Converted to 8B/10B notation, note that the order of bits has been reversed):

Data Byte Name D5.2  
Bits: ABCDE FGH  
10100 010

Translated to a transmission Character in the 8B/10B Transmission Code:

Bits: abcdei fghj  
101001 0101

Each valid Transmission Character of the 8B/10B Transmission Code has been given a name using the following convention: cxx.y, where c is used to show whether the Transmission Character is a Data Character (c is set to D, and SC/D = LOW) or a Special Character (c is set to K, and SC/D = HIGH). When c is set to D, xx is the decimal value of the binary

number composed of the bits E, D, C, B, and A in that order, and the y is the decimal value of the binary number composed of the bits H, G, and F in that order. When c is set to K, xx and y are derived by comparing the encoded bit patterns of the Special Character to those patterns derived from encoded Valid Data bytes and selecting the names of the patterns most similar to the encoded bit patterns of the Special Character.

Under the above conventions, the Transmission Character used for the examples above, is referred to by the name D5.2. The Special Character K29.7 is so named because the first six bits (abcdei) of this character make up a bit pattern similar to that resulting from the encoding of the unencoded 11101 pattern (29), and because the second four bits (fghj) make up a bit pattern similar to that resulting from the encoding of the unencoded 111 pattern (7).

**Note.** This definition of the 10-bit Transmission Code is based on the following references, which describe the same 10-bit transmission code.

A.X. Widmer and P.A. Franaszek. "A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code" IBM Journal of Research and Development, 27, No. 5: 440-451 (September, 1983).

U.S. Patent 4,486,739. Peter A. Franaszek and Albert X. Widmer. "Byte-Oriented DC Balanced (0.4) 8B/10B Partitioned Block Transmission Code" (December 4, 1984).

Fibre Channel Physical and Signaling Interface (ANS X3.230-1994 ANSI FC-PH Standard).

IBM Enterprise Systems Architecture/390 ESCON I/O Interface (document number SA22-7202).

#### 8B/10B Transmission Code

The following information describes how the tables are used for both generating valid Transmission Characters (encoding) and checking the validity of received Transmission Characters (decoding). It also specifies the ordering rules followed when transmitting the bits within a character and the characters within any higher-level constructs specified by a standard.

#### Transmission Order

Within the definition of the 8B/10B Transmission Code, the bit positions of the Transmission Characters are labeled a, b, c, d, e, i, f, g, h, j. Bit "a" is transmitted first followed by bits b, c, d, e, i, f, g, h, and j in that order.

Note that bit i is transmitted between bit e and bit f, rather than in alphabetical order.

#### Valid and Invalid Transmission Characters

The following tables define the valid Data Characters and valid Special Characters (K characters), respectively. The tables are used for both generating valid Transmission Characters and checking the validity of received Transmission Characters. In the tables, each Valid-Data-byte or Special-Character-code entry has two columns that represent two Transmission Characters. The two columns correspond to the current value of the running disparity. Running disparity is a binary parameter with either a negative (-) or positive (+) value.

After powering on, the Transmitter may assume either a positive or negative value for its initial running disparity. Upon transmission of any Transmission Character, the transmitter selects the proper version of the Transmission Character



based on the current running disparity value, and the Transmitter calculates a new value for its running disparity based on the contents of the transmitted character. Special Character codes C1.7 and C2.7 can be used to force the transmission of a specific Special Character with a specific running disparity as required for some special sequences in X3.230.

After powering on, the Receiver may assume either a positive or negative value for its initial running disparity. Upon reception of any Transmission Character, the Receiver decides whether the Transmission Character is valid or invalid according to the following rules and tables and calculates a new value for its Running Disparity based on the contents of the received character.

The following rules for running disparity are used to calculate the new running-disparity value for Transmission Characters that have been transmitted and received.

Running disparity for a Transmission Character is calculated from sub-blocks, where the first six bits (abcdei) form one sub-block and the second four bits (fghj) form the other sub-block. Running disparity at the beginning of the 6-bit sub-block is the running disparity at the end of the previous Transmission Character. Running disparity at the beginning of the 4-bit sub-block is the running disparity at the end of the 6-bit sub-block. Running disparity at the end of the Transmission Character is the running disparity at the end of the 4-bit sub-block.

Running disparity for the sub-blocks is calculated as follows:

1. Running disparity at the end of any sub-block is positive if the sub-block contains more ones than zeros. It is also positive at the end of the 6-bit sub-block if the 6-bit sub-block is 000111, and it is positive at the end of the 4-bit sub-block if the 4-bit sub-block is 0011.
2. Running disparity at the end of any sub-block is negative if the sub-block contains more zeros than ones. It is also negative at the end of the 6-bit sub-block if the 6-bit sub-block is 111000, and it is negative at the end of the 4-bit sub-block if the 4-bit sub-block is 1100.
3. Otherwise, running disparity at the end of the sub-block is the same as at the beginning of the sub-block.

**Use of the Tables for Generating Transmission Characters**

The appropriate entry in *Table 15* for the Valid Data byte or *Table 16* for Special Character byte identify which Transmission Character is generated. The current value of the Transmitter's running disparity is used to select the Transmission Character from its corresponding column. For each Transmission Character transmitted, a new value of the running disparity is calculated. This new value is used as the Transmitter's current running disparity for the next Valid Data

**Table 14. Code Violations Resulting from Prior Errors**

|                            | RD | Character   | RD | Character   | RD | Character      | RD |
|----------------------------|----|-------------|----|-------------|----|----------------|----|
| Transmitted data character | –  | D21.1       | –  | D10.2       | –  | D23.5          | +  |
| Transmitted bit stream     | –  | 101010 1001 | –  | 010101 0101 | –  | 111010 1010    | +  |
| Bit stream after error     | –  | 101010 1011 | +  | 010101 0101 | +  | 111010 1010    | +  |
| Decoded data character     | –  | D21.0       | +  | D10.2       | +  | Code Violation | +  |

byte or Special Character byte encoded and transmitted. *Table 13* shows naming notations and examples of valid transmission characters.

**Use of the Tables for Checking the Validity of Received Transmission Characters**

The column corresponding to the current value of the Receiver's running disparity is searched for the received Transmission Character. If the received Transmission Character is found in the proper column, then the Transmission Character is valid and the associated Data byte or Special Character code is determined (decoded). If the received Transmission Character is not found in that column, then the Transmission Character is invalid. This is called a code violation. Independent of the Transmission Character's validity, the received Transmission Character is used to calculate a new value of running disparity. The new value is used as the Receiver's current running disparity for the next received Transmission Character.

**Table 13. Valid Transmission Characters**

| Byte Name | Data                                |       | Hex Value |
|-----------|-------------------------------------|-------|-----------|
|           | D <sub>IN</sub> or Q <sub>OUT</sub> |       |           |
|           | 765                                 | 43210 |           |
| D0.0      | 000                                 | 00000 | 00        |
| D1.0      | 000                                 | 00001 | 01        |
| D2.0      | 000                                 | 00010 | 02        |
| .         | .                                   | .     | .         |
| .         | .                                   | .     | .         |
| D5.2      | 010                                 | 00101 | 45        |
| .         | .                                   | .     | .         |
| .         | .                                   | .     | .         |
| D30.7     | 111                                 | 11110 | FE        |
| D31.7     | 111                                 | 11111 | FF        |

Detection of a code violation does not necessarily show that the Transmission Character in which the code violation was detected is in error. Code violations may result from a prior error that altered the running disparity of the bit stream which did not result in a detectable error at the Transmission Character in which the error occurred. *Table 14* shows an example of this behavior.



**Table 15. Valid Data Characters (TXCTx[0] = 0, RXSTx[2:0] = 000)**

| Data Byte Name | Bits      | Current RD- | Current RD+ | Data Byte Name | Bits      | Current RD- | Current RD+ |
|----------------|-----------|-------------|-------------|----------------|-----------|-------------|-------------|
|                | HGF EDCBA | abcdei fghj | abcdei fghj |                | HGF EDCBA | abcdei fghj | abcdei fghj |
| D0.0           | 000 00000 | 100111 0100 | 011000 1011 | D0.1           | 001 00000 | 100111 1001 | 011000 1001 |
| D1.0           | 000 00001 | 011101 0100 | 100010 1011 | D1.1           | 001 00001 | 011101 1001 | 100010 1001 |
| D2.0           | 000 00010 | 101101 0100 | 010010 1011 | D2.1           | 001 00010 | 101101 1001 | 010010 1001 |
| D3.0           | 000 00011 | 110001 1011 | 110001 0100 | D3.1           | 001 00011 | 110001 1001 | 110001 1001 |
| D4.0           | 000 00100 | 110101 0100 | 001010 1011 | D4.1           | 001 00100 | 110101 1001 | 001010 1001 |
| D5.0           | 000 00101 | 101001 1011 | 101001 0100 | D5.1           | 001 00101 | 101001 1001 | 101001 1001 |
| D6.0           | 000 00110 | 011001 1011 | 011001 0100 | D6.1           | 001 00110 | 011001 1001 | 011001 1001 |
| D7.0           | 000 00111 | 111000 1011 | 000111 0100 | D7.1           | 001 00111 | 111000 1001 | 000111 1001 |
| D8.0           | 000 01000 | 111001 0100 | 000110 1011 | D8.1           | 001 01000 | 111001 1001 | 000110 1001 |
| D9.0           | 000 01001 | 100101 1011 | 100101 0100 | D9.1           | 001 01001 | 100101 1001 | 100101 1001 |
| D10.0          | 000 01010 | 010101 1011 | 010101 0100 | D10.1          | 001 01010 | 010101 1001 | 010101 1001 |
| D11.0          | 000 01011 | 110100 1011 | 110100 0100 | D11.1          | 001 01011 | 110100 1001 | 110100 1001 |
| D12.0          | 000 01100 | 001101 1011 | 001101 0100 | D12.1          | 001 01100 | 001101 1001 | 001101 1001 |
| D13.0          | 000 01101 | 101100 1011 | 101100 0100 | D13.1          | 001 01101 | 101100 1001 | 101100 1001 |
| D14.0          | 000 01110 | 011100 1011 | 011100 0100 | D14.1          | 001 01110 | 011100 1001 | 011100 1001 |
| D15.0          | 000 01111 | 010111 0100 | 101000 1011 | D15.1          | 001 01111 | 010111 1001 | 101000 1001 |
| D16.0          | 000 10000 | 011011 0100 | 100100 1011 | D16.1          | 001 10000 | 011011 1001 | 100100 1001 |
| D17.0          | 000 10001 | 100011 1011 | 100011 0100 | D17.1          | 001 10001 | 100011 1001 | 100011 1001 |
| D18.0          | 000 10010 | 010011 1011 | 010011 0100 | D18.1          | 001 10010 | 010011 1001 | 010011 1001 |
| D19.0          | 000 10011 | 110010 1011 | 110010 0100 | D19.1          | 001 10011 | 110010 1001 | 110010 1001 |
| D20.0          | 000 10100 | 001011 1011 | 001011 0100 | D20.1          | 001 10100 | 001011 1001 | 001011 1001 |
| D21.0          | 000 10101 | 101010 1011 | 101010 0100 | D21.1          | 001 10101 | 101010 1001 | 101010 1001 |
| D22.0          | 000 10110 | 011010 1011 | 011010 0100 | D22.1          | 001 10110 | 011010 1001 | 011010 1001 |
| D23.0          | 000 10111 | 111010 0100 | 000101 1011 | D23.1          | 001 10111 | 111010 1001 | 000101 1001 |
| D24.0          | 000 11000 | 110011 0100 | 001100 1011 | D24.1          | 001 11000 | 110011 1001 | 001100 1001 |
| D25.0          | 000 11001 | 100110 1011 | 100110 0100 | D25.1          | 001 11001 | 100110 1001 | 100110 1001 |
| D26.0          | 000 11010 | 010110 1011 | 010110 0100 | D26.1          | 001 11010 | 010110 1001 | 010110 1001 |
| D27.0          | 000 11011 | 110110 0100 | 001001 1011 | D27.1          | 001 11011 | 110110 1001 | 001001 1001 |
| D28.0          | 000 11100 | 001110 1011 | 001110 0100 | D28.1          | 001 11100 | 001110 1001 | 001110 1001 |
| D29.0          | 000 11101 | 101110 0100 | 010001 1011 | D29.1          | 001 11101 | 101110 1001 | 010001 1001 |
| D30.0          | 000 11110 | 011110 0100 | 100001 1011 | D30.1          | 001 11110 | 011110 1001 | 100001 1001 |
| D31.0          | 000 11111 | 101011 0100 | 010100 1011 | D31.1          | 001 11111 | 101011 1001 | 010100 1001 |



**Table 15. Valid Data Characters (TXCTx[0] = 0, RXSTx[2:0] = 000) (continued)**

| Data Byte Name | Bits      | Current RD- | Current RD+ | Data Byte Name | Bits      | Current RD- | Current RD+ |
|----------------|-----------|-------------|-------------|----------------|-----------|-------------|-------------|
|                | HGF EDCBA | abcdei fghj | abcdei fghj |                | HGF EDCBA | abcdei fghj | abcdei fghj |
| D0.2           | 010 00000 | 100111 0101 | 011000 0101 | D0.3           | 011 00000 | 100111 0011 | 011000 1100 |
| D1.2           | 010 00001 | 011101 0101 | 100010 0101 | D1.3           | 011 00001 | 011101 0011 | 100010 1100 |
| D2.2           | 010 00010 | 101101 0101 | 010010 0101 | D2.3           | 011 00010 | 101101 0011 | 010010 1100 |
| D3.2           | 010 00011 | 110001 0101 | 110001 0101 | D3.3           | 011 00011 | 110001 1100 | 110001 0011 |
| D4.2           | 010 00100 | 110101 0101 | 001010 0101 | D4.3           | 011 00100 | 110101 0011 | 001010 1100 |
| D5.2           | 010 00101 | 101001 0101 | 101001 0101 | D5.3           | 011 00101 | 101001 1100 | 101001 0011 |
| D6.2           | 010 00110 | 011001 0101 | 011001 0101 | D6.3           | 011 00110 | 011001 1100 | 011001 0011 |
| D7.2           | 010 00111 | 111000 0101 | 000111 0101 | D7.3           | 011 00111 | 111000 1100 | 000111 0011 |
| D8.2           | 010 01000 | 111001 0101 | 000110 0101 | D8.3           | 011 01000 | 111001 0011 | 000110 1100 |
| D9.2           | 010 01001 | 100101 0101 | 100101 0101 | D9.3           | 011 01001 | 100101 1100 | 100101 0011 |
| D10.2          | 010 01010 | 010101 0101 | 010101 0101 | D10.3          | 011 01010 | 010101 1100 | 010101 0011 |
| D11.2          | 010 01011 | 110100 0101 | 110100 0101 | D11.3          | 011 01011 | 110100 1100 | 110100 0011 |
| D12.2          | 010 01100 | 001101 0101 | 001101 0101 | D12.3          | 011 01100 | 001101 1100 | 001101 0011 |
| D13.2          | 010 01101 | 101100 0101 | 101100 0101 | D13.3          | 011 01101 | 101100 1100 | 101100 0011 |
| D14.2          | 010 01110 | 011100 0101 | 011100 0101 | D14.3          | 011 01110 | 011100 1100 | 011100 0011 |
| D15.2          | 010 01111 | 010111 0101 | 101000 0101 | D15.3          | 011 01111 | 010111 0011 | 101000 1100 |
| D16.2          | 010 10000 | 011011 0101 | 100100 0101 | D16.3          | 011 10000 | 011011 0011 | 100100 1100 |
| D17.2          | 010 10001 | 100011 0101 | 100011 0101 | D17.3          | 011 10001 | 100011 1100 | 100011 0011 |
| D18.2          | 010 10010 | 010011 0101 | 010011 0101 | D18.3          | 011 10010 | 010011 1100 | 010011 0011 |
| D19.2          | 010 10011 | 110010 0101 | 110010 0101 | D19.3          | 011 10011 | 110010 1100 | 110010 0011 |
| D20.2          | 010 10100 | 001011 0101 | 001011 0101 | D20.3          | 011 10100 | 001011 1100 | 001011 0011 |
| D21.2          | 010 10101 | 101010 0101 | 101010 0101 | D21.3          | 011 10101 | 101010 1100 | 101010 0011 |
| D22.2          | 010 10110 | 011010 0101 | 011010 0101 | D22.3          | 011 10110 | 011010 1100 | 011010 0011 |
| D23.2          | 010 10111 | 111010 0101 | 000101 0101 | D23.3          | 011 10111 | 111010 0011 | 000101 1100 |
| D24.2          | 010 11000 | 110011 0101 | 001100 0101 | D24.3          | 011 11000 | 110011 0011 | 001100 1100 |
| D25.2          | 010 11001 | 100110 0101 | 100110 0101 | D25.3          | 011 11001 | 100110 1100 | 100110 0011 |
| D26.2          | 010 11010 | 010110 0101 | 010110 0101 | D26.3          | 011 11010 | 010110 1100 | 010110 0011 |
| D27.2          | 010 11011 | 110110 0101 | 001001 0101 | D27.3          | 011 11011 | 110110 0011 | 001001 1100 |
| D28.2          | 010 11100 | 001110 0101 | 001110 0101 | D28.3          | 011 11100 | 001110 1100 | 001110 0011 |
| D29.2          | 010 11101 | 101110 0101 | 010001 0101 | D29.3          | 011 11101 | 101110 0011 | 010001 1100 |
| D30.2          | 010 11110 | 011110 0101 | 100001 0101 | D30.3          | 011 11110 | 011110 0011 | 100001 1100 |
| D31.2          | 010 11111 | 101011 0101 | 010100 0101 | D31.3          | 011 11111 | 101011 0011 | 010100 1100 |



**Table 15. Valid Data Characters (TXCTx[0] = 0, RXSTx[2:0] = 000) (continued)**

| Data Byte Name | Bits      | Current RD- | Current RD+ | Data Byte Name | Bits      | Current RD- | Current RD+ |
|----------------|-----------|-------------|-------------|----------------|-----------|-------------|-------------|
|                | HGF EDCBA | abcdei fghj | abcdei fghj |                | HGF EDCBA | abcdei fghj | abcdei fghj |
| D0.4           | 100 00000 | 100111 0010 | 011000 1101 | D0.5           | 101 00000 | 100111 1010 | 011000 1010 |
| D1.4           | 100 00001 | 011101 0010 | 100010 1101 | D1.5           | 101 00001 | 011101 1010 | 100010 1010 |
| D2.4           | 100 00010 | 101101 0010 | 010010 1101 | D2.5           | 101 00010 | 101101 1010 | 010010 1010 |
| D3.4           | 100 00011 | 110001 1101 | 110001 0010 | D3.5           | 101 00011 | 110001 1010 | 110001 1010 |
| D4.4           | 100 00100 | 110101 0010 | 001010 1101 | D4.5           | 101 00100 | 110101 1010 | 001010 1010 |
| D5.4           | 100 00101 | 101001 1101 | 101001 0010 | D5.5           | 101 00101 | 101001 1010 | 101001 1010 |
| D6.4           | 100 00110 | 011001 1101 | 011001 0010 | D6.5           | 101 00110 | 011001 1010 | 011001 1010 |
| D7.4           | 100 00111 | 111000 1101 | 000111 0010 | D7.5           | 101 00111 | 111000 1010 | 000111 1010 |
| D8.4           | 100 01000 | 111001 0010 | 000110 1101 | D8.5           | 101 01000 | 111001 1010 | 000110 1010 |
| D9.4           | 100 01001 | 100101 1101 | 100101 0010 | D9.5           | 101 01001 | 100101 1010 | 100101 1010 |
| D10.4          | 100 01010 | 010101 1101 | 010101 0010 | D10.5          | 101 01010 | 010101 1010 | 010101 1010 |
| D11.4          | 100 01011 | 110100 1101 | 110100 0010 | D11.5          | 101 01011 | 110100 1010 | 110100 1010 |
| D12.4          | 100 01100 | 001101 1101 | 001101 0010 | D12.5          | 101 01100 | 001101 1010 | 001101 1010 |
| D13.4          | 100 01101 | 101100 1101 | 101100 0010 | D13.5          | 101 01101 | 101100 1010 | 101100 1010 |
| D14.4          | 100 01110 | 011100 1101 | 011100 0010 | D14.5          | 101 01110 | 011100 1010 | 011100 1010 |
| D15.4          | 100 01111 | 010111 0010 | 101000 1101 | D15.5          | 101 01111 | 010111 1010 | 101000 1010 |
| D16.4          | 100 10000 | 011011 0010 | 100100 1101 | D16.5          | 101 10000 | 011011 1010 | 100100 1010 |
| D17.4          | 100 10001 | 100011 1101 | 100011 0010 | D17.5          | 101 10001 | 100011 1010 | 100011 1010 |
| D18.4          | 100 10010 | 010011 1101 | 010011 0010 | D18.5          | 101 10010 | 010011 1010 | 010011 1010 |
| D19.4          | 100 10011 | 110010 1101 | 110010 0010 | D19.5          | 101 10011 | 110010 1010 | 110010 1010 |
| D20.4          | 100 10100 | 001011 1101 | 001011 0010 | D20.5          | 101 10100 | 001011 1010 | 001011 1010 |
| D21.4          | 100 10101 | 101010 1101 | 101010 0010 | D21.5          | 101 10101 | 101010 1010 | 101010 1010 |
| D22.4          | 100 10110 | 011010 1101 | 011010 0010 | D22.5          | 101 10110 | 011010 1010 | 011010 1010 |
| D23.4          | 100 10111 | 111010 0010 | 000101 1101 | D23.5          | 101 10111 | 111010 1010 | 000101 1010 |
| D24.4          | 100 11000 | 110011 0010 | 001100 1101 | D24.5          | 101 11000 | 110011 1010 | 001100 1010 |
| D25.4          | 100 11001 | 100110 1101 | 100110 0010 | D25.5          | 101 11001 | 100110 1010 | 100110 1010 |
| D26.4          | 100 11010 | 010110 1101 | 010110 0010 | D26.5          | 101 11010 | 010110 1010 | 010110 1010 |
| D27.4          | 100 11011 | 110110 0010 | 001001 1101 | D27.5          | 101 11011 | 110110 1010 | 001001 1010 |
| D28.4          | 100 11100 | 001110 1101 | 001110 0010 | D28.5          | 101 11100 | 001110 1010 | 001110 1010 |
| D29.4          | 100 11101 | 101110 0010 | 010001 1101 | D29.5          | 101 11101 | 101110 1010 | 010001 1010 |
| D30.4          | 100 11110 | 011110 0010 | 100001 1101 | D30.5          | 101 11110 | 011110 1010 | 100001 1010 |
| D31.4          | 100 11111 | 101011 0010 | 010100 1101 | D31.5          | 101 11111 | 101011 1010 | 010100 1010 |



**Table 15. Valid Data Characters (TXCTx[0] = 0, RXSTx[2:0] = 000) (continued)**

| Data Byte Name | Bits      | Current RD- | Current RD+ | Data Byte Name | Bits      | Current RD- | Current RD+ |
|----------------|-----------|-------------|-------------|----------------|-----------|-------------|-------------|
|                | HGF EDCBA | abcdei fghj | abcdei fghj |                | HGF EDCBA | abcdei fghj | abcdei fghj |
| D0.6           | 110 00000 | 100111 0110 | 011000 0110 | D0.7           | 111 00000 | 100111 0001 | 011000 1110 |
| D1.6           | 110 00001 | 011101 0110 | 100010 0110 | D1.7           | 111 00001 | 011101 0001 | 100010 1110 |
| D2.6           | 110 00010 | 101101 0110 | 010010 0110 | D2.7           | 111 00010 | 101101 0001 | 010010 1110 |
| D3.6           | 110 00011 | 110001 0110 | 110001 0110 | D3.7           | 111 00011 | 110001 1110 | 110001 0001 |
| D4.6           | 110 00100 | 110101 0110 | 001010 0110 | D4.7           | 111 00100 | 110101 0001 | 001010 1110 |
| D5.6           | 110 00101 | 101001 0110 | 101001 0110 | D5.7           | 111 00101 | 101001 1110 | 101001 0001 |
| D6.6           | 110 00110 | 011001 0110 | 011001 0110 | D6.7           | 111 00110 | 011001 1110 | 011001 0001 |
| D7.6           | 110 00111 | 111000 0110 | 000111 0110 | D7.7           | 111 00111 | 111000 1110 | 000111 0001 |
| D8.6           | 110 01000 | 111001 0110 | 000110 0110 | D8.7           | 111 01000 | 111001 0001 | 000110 1110 |
| D9.6           | 110 01001 | 100101 0110 | 100101 0110 | D9.7           | 111 01001 | 100101 1110 | 100101 0001 |
| D10.6          | 110 01010 | 010101 0110 | 010101 0110 | D10.7          | 111 01010 | 010101 1110 | 010101 0001 |
| D11.6          | 110 01011 | 110100 0110 | 110100 0110 | D11.7          | 111 01011 | 110100 1110 | 110100 1000 |
| D12.6          | 110 01100 | 001101 0110 | 001101 0110 | D12.7          | 111 01100 | 001101 1110 | 001101 0001 |
| D13.6          | 110 01101 | 101100 0110 | 101100 0110 | D13.7          | 111 01101 | 101100 1110 | 101100 1000 |
| D14.6          | 110 01110 | 011100 0110 | 011100 0110 | D14.7          | 111 01110 | 011100 1110 | 011100 1000 |
| D15.6          | 110 01111 | 010111 0110 | 101000 0110 | D15.7          | 111 01111 | 010111 0001 | 101000 1110 |
| D16.6          | 110 10000 | 011011 0110 | 100100 0110 | D16.7          | 111 10000 | 011011 0001 | 100100 1110 |
| D17.6          | 110 10001 | 100011 0110 | 100011 0110 | D17.7          | 111 10001 | 100011 0111 | 100011 0001 |
| D18.6          | 110 10010 | 010011 0110 | 010011 0110 | D18.7          | 111 10010 | 010011 0111 | 010011 0001 |
| D19.6          | 110 10011 | 110010 0110 | 110010 0110 | D19.7          | 111 10011 | 110010 1110 | 110010 0001 |
| D20.6          | 110 10100 | 001011 0110 | 001011 0110 | D20.7          | 111 10100 | 001011 0111 | 001011 0001 |
| D21.6          | 110 10101 | 101010 0110 | 101010 0110 | D21.7          | 111 10101 | 101010 1110 | 101010 0001 |
| D22.6          | 110 10110 | 011010 0110 | 011010 0110 | D22.7          | 111 10110 | 011010 1110 | 011010 0001 |
| D23.6          | 110 10111 | 111010 0110 | 000101 0110 | D23.7          | 111 10111 | 111010 0001 | 000101 1110 |
| D24.6          | 110 11000 | 110011 0110 | 001100 0110 | D24.7          | 111 11000 | 110011 0001 | 001100 1110 |
| D25.6          | 110 11001 | 100110 0110 | 100110 0110 | D25.7          | 111 11001 | 100110 1110 | 100110 0001 |
| D26.6          | 110 11010 | 010110 0110 | 010110 0110 | D26.7          | 111 11010 | 010110 1110 | 010110 0001 |
| D27.6          | 110 11011 | 110110 0110 | 001001 0110 | D27.7          | 111 11011 | 110110 0001 | 001001 1110 |
| D28.6          | 110 11100 | 001110 0110 | 001110 0110 | D28.7          | 111 11100 | 001110 1110 | 001110 0001 |
| D29.6          | 110 11101 | 101110 0110 | 010001 0110 | D29.7          | 111 11101 | 101110 0001 | 010001 1110 |
| D30.6          | 110 11110 | 011110 0110 | 100001 0110 | D30.7          | 111 11110 | 011110 0001 | 100001 1110 |
| D31.6          | 110 11111 | 101011 0110 | 010100 0110 | D31.7          | 111 11111 | 101011 0001 | 010100 1110 |





Table 16. Valid Special Character Codes and Sequences (TXCTx = special character code or RXSTx[2:0] = 001)<sup>[40, 41]</sup>

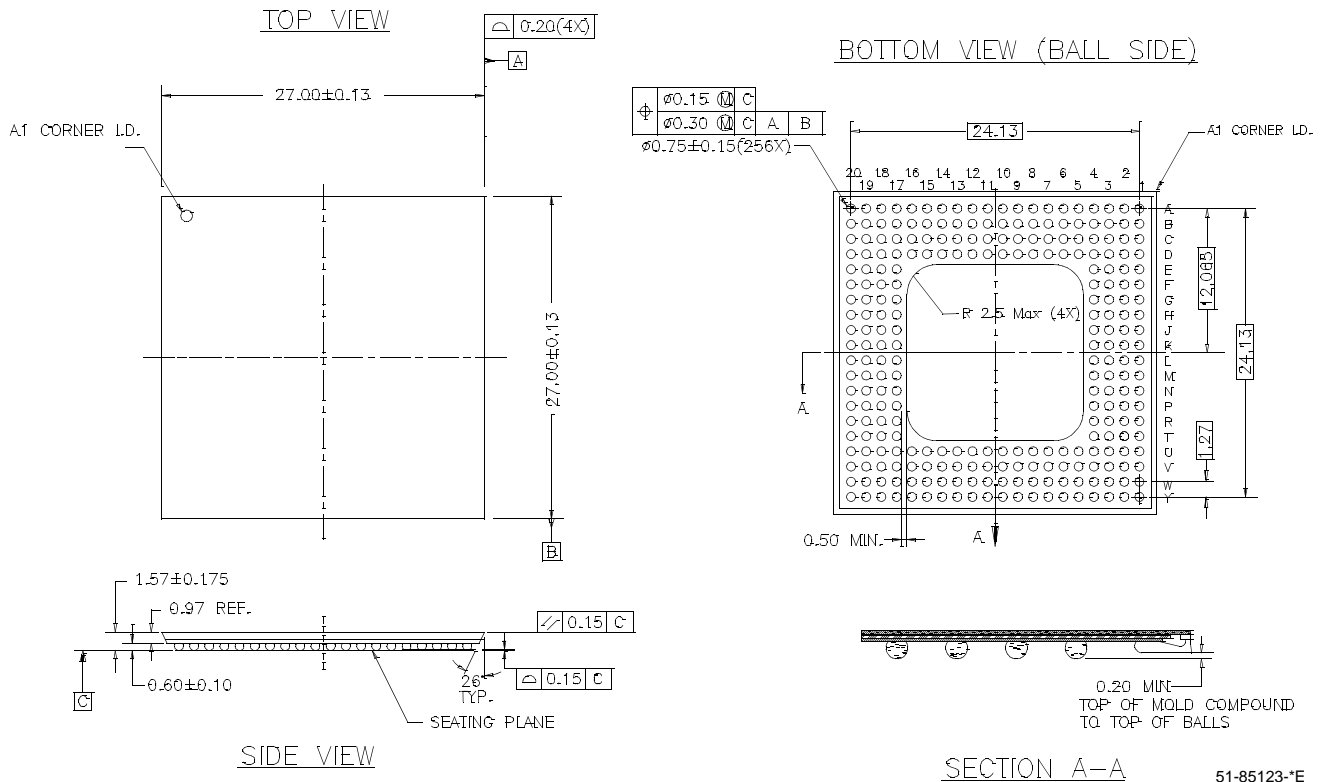
| S.C. Code Name                                | S.C. Byte Name                 |                   |                                |                   | Current RD-<br>abcdei fgjh     | Current RD+<br>abcdei fgjh     |
|---|--------------------------------|-------------------|--------------------------------|-------------------|--------------------------------|--------------------------------|
|   | Cypress                        |                   | Alternate                      |                   |                                |                                |
|   | S.C. Byte Name <sub>[42]</sub> | Bits<br>HGF EDCBA | S.C. Byte Name <sub>[42]</sub> | Bits<br>HGF EDCBA |                                |                                |
| K28.0   | C0.0 (C00)                     | 000 00000         | C28.0 (C1C)                    | 000 11100         | 001111 0100                    | 110000 1011                    |
| K28.1 <sup>[43]</sup>                         | C1.0 (C01)                     | 000 00001         | C28.1 (C3C)                    | 001 11100         | 001111 1001                    | 110000 0110                    |
| K28.2 <sup>[43]</sup>                         | C2.0 (C02)                     | 000 00010         | C28.2 (C5C)                    | 010 11100         | 001111 0101                    | 110000 1010                    |
| K28.3   | C3.0 (C03)                     | 000 00011         | C28.3 (C7C)                    | 011 11100         | 001111 0011                    | 110000 1100                    |
| K28.4 <sup>[43]</sup>                         | C4.0 (C04)                     | 000 00100         | C28.4 (C9C)                    | 100 11100         | 001111 0010                    | 110000 1101                    |
| K28.5 <sup>[43, 44]</sup>                     | C5.0 (C05)                     | 000 00101         | C28.5 (CBC)                    | 101 11100         | 001111 1010                    | 110000 0101                    |
| K28.6 <sup>[43]</sup>                         | C6.0 (C06)                     | 000 00110         | C28.6 (CDC)                    | 110 11100         | 001111 0110                    | 110000 1001                    |
| K28.7 <sup>[43, 45]</sup>                     | C7.0 (C07)                     | 000 00111         | C28.7 (CFC)                    | 111 11100         | 001111 1000                    | 110000 0111                    |
| K23.7   | C8.0 (C08)                     | 000 01000         | C23.7 (CF7)                    | 111 10111         | 111010 1000                    | 000101 0111                    |
| K27.7   | C9.0 (C09)                     | 000 01001         | C27.7 (CFB)                    | 111 11011         | 110110 1000                    | 001001 0111                    |
| K29.7   | C10.0 (C0A)                    | 000 01010         | C29.7 (CFD)                    | 111 11101         | 101110 1000                    | 010001 0111                    |
| K30.7   | C11.0 (C0B)                    | 000 01011         | C30.7 (CFE)                    | 111 11110         | 011110 1000                    | 100001 0111                    |
| <b>End of Frame Sequence</b>                  |                                |                   |                                |                   |                                |                                |
| EOFxx   | C2.1 (C22)                     | 001 00010         | C2.1 (C22)                     | 001 00010         | -K28.5,Dn.xxx0 <sup>[46]</sup> | +K28.5,Dn.xxx1 <sup>[46]</sup> |
| <b>Code Rule Violation and SVS Tx Pattern</b> |                                |                   |                                |                   |                                |                                |
| Exception <sup>[45, 47]</sup>                 | C0.7 (CE0)                     | 111 00000         | C0.7 (CE0)                     | 111 00000         | 100111 1000                    | 011000 0111                    |
| -K28.5 <sup>[48]</sup>                        | C1.7 (CE1)                     | 111 00001         | C1.7 (CE1)                     | 111 00001         | 001111 1010                    | 001111 1010                    |
| +K28.5 <sup>[49]</sup>                        | C2.7 (CE2)                     | 111 00010         | C2.7 (CE2)                     | 111 00010         | 110000 0101                    | 110000 0101                    |
| <b>Running Disparity Violation Pattern</b>    |                                |                   |                                |                   |                                |                                |
| Exception <sup>[50]</sup>                     | C4.7 (CE4)                     | 111 00100         | C4.7 (CE4)                     | 111 00100         | 110111 0101                    | 001000 1010                    |

**Notes:**

40. All codes not shown are reserved.
41. Notation for Special Character Code Name is consistent with Fibre Channel and ESCON naming conventions. Special Character Code Name is intended to describe binary information present on I/O pins. Common usage for the name can either be in the form used for describing Data patterns (i.e., C0.0 through C31.7), or in hex notation (i.e., Cnn where nn = the specified value between 00 and FF).
42. Both the Cypress and alternate encodings may be used for data transmission to generate specific Special Character Codes. The decoding process for received characters generates Cypress codes or Alternate codes as selected by the BOE[7:0] configuration inputs.
43. These characters are used for control of ESCON interfaces. They can be sent as embedded commands or other markers when not operating using ESCON protocols.
44. The K28.5 character is used for framing operations by the receiver. It is also the pad or fill character transmitted to maintain the serial link when no user data is available.
45. Care must be taken when using this Special Character code. When a C7.0 or a C0.7 is followed by a D11.x or D20.x, an alias K28.5 sync character is created. These sequences can cause erroneous framing and should be avoided while RFENx = 1.
46. C2.1 = Transmit either -K28.5+ or +K28.5- as determined by Current RD and modify the Transmission Character that follows, by setting its least significant bit to 1 or 0. If Current RD at the start of the following character is plus (+) the LSB is set to 0, and if Current RD is minus (-) the LSB becomes 1. This modification allows construction of X3.230 "EOF" frame delimiters wherein the second data byte is determined by the Current RD. For example, to send "EOFdt" the controller could issue the sequence C2.1-D21.4- D21.4-D21.4, and the HOTLink Transmitter sends either K28.5-D21.4-D21.4-D21.4 or K28.5-D21.5- D21.4-D21.4 based on Current RD. Likewise to send "EOFdti" the controller could issue the sequence C2.1-D10.4-D21.4-D21.4, and the HOTLink Transmitter sends either K28.5-D10.4-D21.4- D21.4 or K28.5-D10.5-D21.4-D21.4 based on Current RD. The receiver never outputs this Special Character, since K28.5 is decoded as C5.0, C1.7, or C2.7, and the subsequent bytes are decoded as data.
47. C0.7 = Transmit a deliberate code rule violation. The code chosen for this function follows the normal Running Disparity rules. Transmission of this Special Character has the same effect as asserting TXSVS = HIGH. The receiver only outputs this Special Character if the Transmission Character being decoded is not found in the tables.
48. C1.7 = Transmit Negative K28.5 (-K28.5+) disregarding Current RD. The receiver only outputs this Special Character if K28.5 is received with the wrong running disparity. The receiver outputs C1.7 if -K28.5 is received with RD+, otherwise K28.5 is decoded as C5.0 or C2.7.
49. C2.7 = Transmit Positive K28.5 (+K28.5-) disregarding Current RD. The receiver only outputs this Special Character if K28.5 is received with the wrong running disparity. The receiver outputs C2.7 if +K28.5 is received with RD-, otherwise K28.5 is decoded as C5.0 or C1.7.
50. C4.7 = Transmit a deliberate code rule violation to indicate a Running Disparity violation. The receiver only outputs this Special Character if the Transmission Character being decoded is found in the tables, but Running Disparity does not match. This might indicate that an error occurred in a prior byte.

**Ordering Information**

| Speed    | Ordering Code      | Package Name | Package Type  | Operating Range |
|----------|--------------------|--------------|---|-----------------|
| Standard | CYP15G0403DXB-BGC  | BL256        | 256-Ball Thermally Enhanced Ball Grid Array         | Commercial      |
| Standard | CYP15G0403DXB-BGI  | BL256        | 256-Ball Thermally Enhanced Ball Grid Array         | Industrial      |
| Standard | CYV15G0403DXB-BGC  | BL256        | 256-Ball Thermally Enhanced Ball Grid Array         | Commercial      |
| Standard | CYV15G0403DXB-BGI  | BL256        | 256-Ball Thermally Enhanced Ball Grid Array         | Industrial      |
| OBSAI    | CYW15G0403DXB-BGC  | BL256        | 256-Ball Thermally Enhanced Ball Grid Array         | Commercial      |
| OBSAI    | CYW15G0403DXB-BGI  | BL256        | 256-Ball Thermally Enhanced Ball Grid Array         | Industrial      |
| Standard | CYP15G0403DXB-BGXC | BL256        | Pb-Free 256-Ball Thermally Enhanced Ball Grid Array | Commercial      |
| Standard | CYP15G0403DXB-BGXI | BL256        | Pb-Free 256-Ball Thermally Enhanced Ball Grid Array | Industrial      |
| Standard | CYV15G0403DXB-BGXC | BL256        | Pb-Free 256-Ball Thermally Enhanced Ball Grid Array | Commercial      |
| Standard | CYV15G0403DXB-BGXI | BL256        | Pb-Free 256-Ball Thermally Enhanced Ball Grid Array | Industrial      |
| OBSAI    | CYW15G0403DXB-BGXC | BL256        | Pb-Free 256-Ball Thermally Enhanced Ball Grid Array | Commercial      |
| OBSAI    | CYW15G0403DXB-BGXI | BL256        | Pb-Free 256-Ball Thermally Enhanced Ball Grid Array | Industrial      |

**Package Diagram**
**256-Lead L2 Ball Grid Array (27 x 27 x 1.57 mm) BL256**


51-85123-1E

HOTLink is a registered trademark and HOTLink II and MultiFrame are trademarks of Cypress Semiconductor. CPRI is a trademark of Siemens AG. IBM and ESCON are registered trademarks, and FICON is a trademark, of International Business Machines. All product and company names mentioned in this document may be the trademarks of their respective holders.



Document History Page

| Document Title: CYP(V)(W)15G0403DXB Independent Clock Quad HOTLink II™ Transceiver<br>Document Number: 38-02065 |         |            |                 |   |
|---|---------|------------|-----------------|---|
| REV.  | ECN NO. | ISSUE DATE | ORIG. OF CHANGE | DESCRIPTION OF CHANGE   |
| **  | 118422  | 09/24/02   | LNМ             | New Data Sheet  |
| *A  | 125289  | 04/04/03   | CGX             | Revised entire data sheet<br>Redefined device   |
| *B  | 128692  | 08/14/03   | PDS             | Provided AC timing information for TXERRx<br>Added additional information regarding the availability of half-rate RXCLKx± when REFCLKx is a full-rate clock with RXCKSELx = 1<br>Added influence of ULCx input on LFLx status<br>Added influence of DECMODEx on decoder bypass<br>Revised the text for “Device Configuration and Control Interface” for better clarity<br>Removed the timing parameter t <sub>RREFDV</sub> and added the timing parameter t <sub>RREFDW</sub> instead. This change was done to provide a more meaningful timing parameter<br>Revised t <sub>RREFDA</sub> from 9.5 ns to 9.7 ns<br>Added additional information to “Device Configuration Strategy” |
| *C  | 234390  | See ECN    | PDS             | Removed dependence of DECMODEx on decoder bypass.<br>Revised AC timing parameters (AC Electrical Characteristics) to match final device characterization.<br>Expanded the CDR Range Controller’s permissible frequency offset between incoming serial signalling rate and Reference clock from ±200-PPM to ±1500-PPM (changed parameter t <sub>REFRX</sub> ).   |
| *D  | 338721  | See ECN    | SUA             | Added CYW15G0403DXB part number for OBSAI RP3 compliance to support operating data rate up to 1540 MBaud. Made changes to reflect OBSAI RP3 and CPR compliance. Added Pb-Free Package option for all parts listed in the datasheet.<br>Modified Timing Parameters<br>Changed MBd to MBaud in SPDSEL pin description   |
| *E  | 384307  | See ECN    | AGT             | Revised setup and hold time parameters (t <sub>TXDH</sub> , t <sub>TREFDS</sub> , t <sub>TREFDH</sub> , t <sub>RXDv-</sub> , t <sub>RXDv+</sub> , t <sub>RXDv+</sub> , t <sub>REFxDV-</sub> , t <sub>REFxDV+</sub> )  |