Replace Your TAXI™–125 and TAXI–175

This application note will explain how to replace TAXIchip™ devices with the HOTLink™ devices from Cypress Semiconductor. This note begins with an introduction to HOTLink and then gives advantages and replacement suggestions for the TAXI–125 and TAXI–175 devices.

HOTLink Introduction

The HOTLink family of devices transfers data from point to point over high-speed serial links at 160 to 330 Mbits/second (*Figure 1*). The CY7B923 Transmitter (*Figure 2*) takes an 8-bit parallel data stream and encodes it using the Fibre Channel compliant and ESCON™-compliant 8B/10B code. This code maps all 8-bit data characters into a 10-bit transmission code that insures the transmission signal contains suitable transitions for recovery by the receiving device. The transmitter takes this 10-bit data word and converts it to a serial bit-stream and transmits it at 10 times the byte rate over a serial transmission link.

The CY7B933 HOTLink Receiver (*Figure 3*) lies on the other end of a transmission link that may consist of anything from a few inches of printed circuit-board trace to several kilometers of fiberoptic cable. The receiver decodes the incoming bit stream and reconstructs the original parallel data character, which is presented at the outputs aligned with the recovered clock. The receiver, in addition to these tasks, checks the incoming data stream for errors that may have occurred in the serial transmission.

The SC/$\overline{D}$ (Special Character/$\overline{\text{Data}}$) pin provides the ability to transmit command codes in addition to sending data characters. These codes are mapped

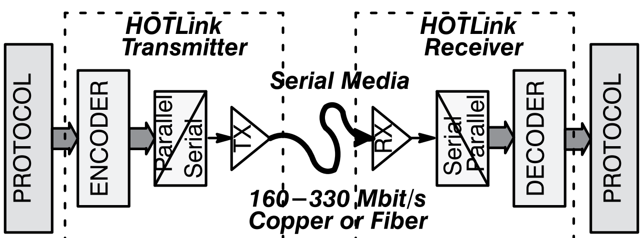

Figure 1. CY7B923 Transmitter Logic Diagram

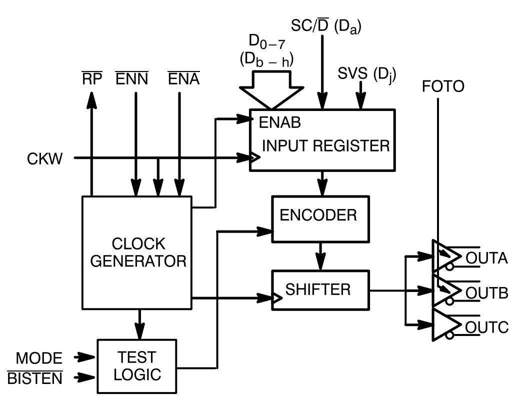

Figure 2. CY7B923 Transmitter Logic Diagram

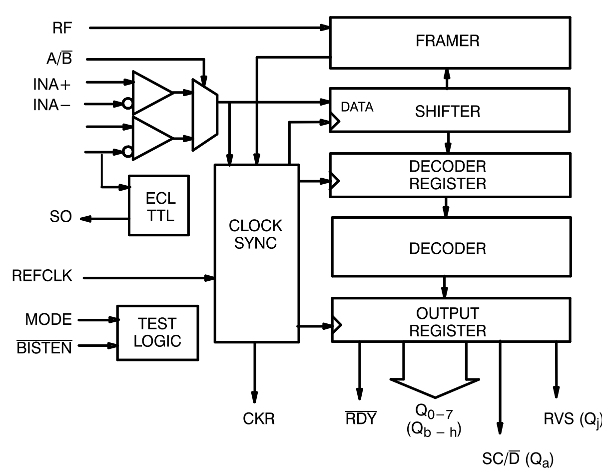

Figure 3. CY7B933 Receiver Logic Diagram

to 10-bit transmission characters defined in the 8B/10B codes of the Fibre Channel standard. This provides the ability to send commands as part of the transmission stream signaling events such as Idle, Start-of-frame, End-of-frame, etc.

Other features include Built-In Self-Test for in-system diagnostic testing, unencoded mode for sending 10-bit data in systems that use a different encoding method, and a seamless parallel interface for connection with both asynchronous and clocked FIFOs. A brief description of the various features of HOTLink is given below with a more detailed discussion found in the CY7B923/CY7B933 HOTLink Transmitter/Receiver datasheet. The PLCC pinouts for these devices are shown in *Figure 4*.

Replacement of TAXI−1 Devices

The following section shows how to upgrade a system using the TAXI−1 (Am7968/Am7969−125 or the Am7968/Am7969−175) with HOTLink. This section begins with a brief explanation of the TAXI−1 devices. It then shows how HOTLink simplifies systems that either use, or plan to use, these devices. It ends with a discussion on how to modify systems that use some of the features of these TAXI devices that are different from HOTLink.

Brief Description of TAXI−1

The Am7968/AM7969 provide a method of connecting systems over a serial link. These devices accept 8-, 9-, or 10-bit parallel data words on the transmitting side of the link (*Figure 5*) and convert the data to a serial bit stream using 4B/5B and 5B/6B NRZI (Non Return to Zero, Invert on 1s) codes. These codes convert 4 input bits into 5 transmission bits in the case of the 4B/5B code, or 5 input bits into 6 transmission bits in the case of the 5B/6B code. These codes insure that enough signal transitions occur on the link for the receiving device to recover the data. On the receiving end of the system (*Figure 6*), the serial data is decoded and presented to the outputs along with the recovered transmitted clock. The pinouts of the Am7968 TAXI Transmitter and Am7969 TAXI Receiver are shown in *Figure 7*.

Simplifying Your System with HOTLink

HOTLink offers an extensive feature list that provides a host of benefits when designing systems that perform point-to-point serial communication.

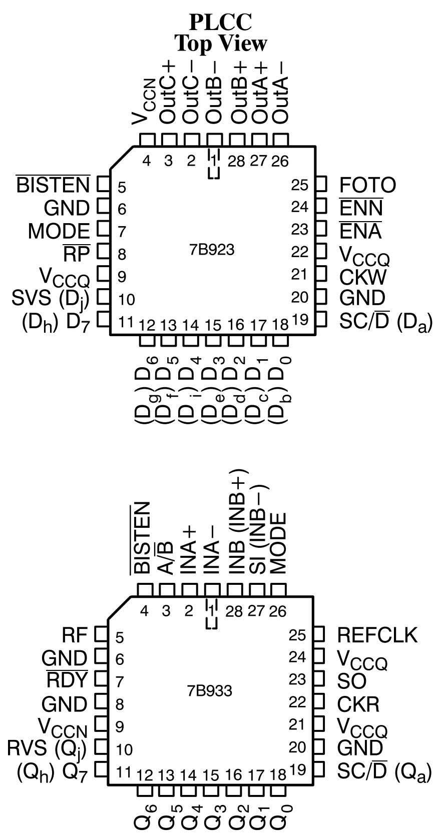

Figure 4. CY7B923 and CY7B933 Pin Configurations

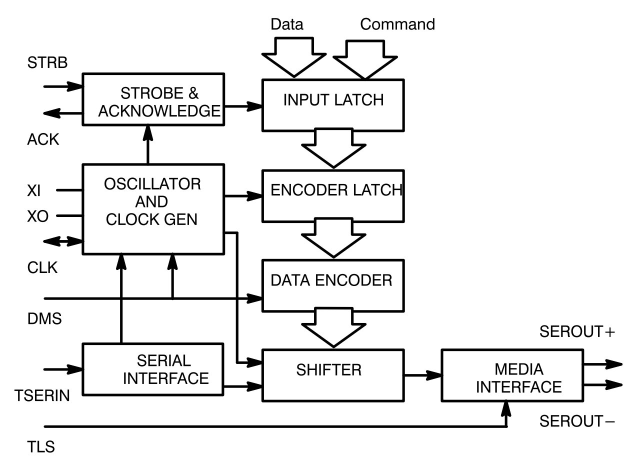

Figure 5. Am7968 Logic Diagram

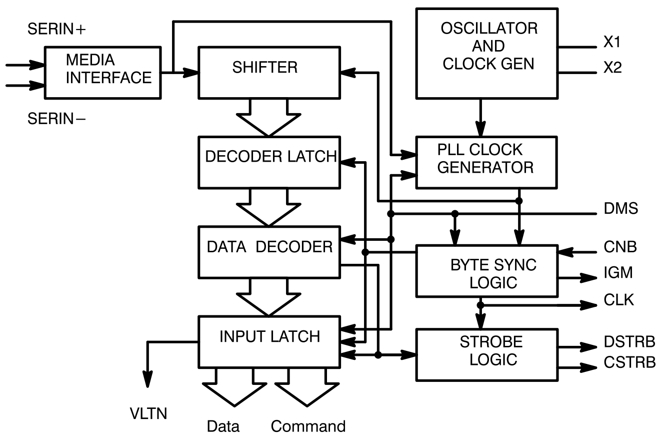

Figure 6. Am7969 Logic Diagram

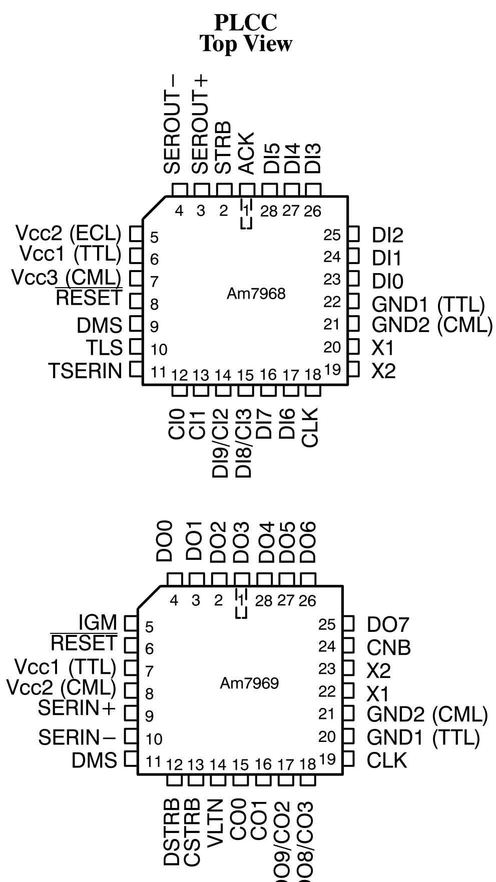

Figure 7. Am7968 and Am7969 Pin Configurations

While many of these features are offered in the TAXI−1 devices, they are difficult to use. These features include multiplexed command and data, multiple inputs and outputs, self-test operation, and many others. Below is a list of HOTLink features and advantage offered to the system designer when compared with the TAXI−1 devices.

Multiplexed Command and Data

One of the major differences between the HOTLink and the TAXI devices is the parallel data interface. The TAXI devices have separate inputs for command and data, while the HOTLink devices have an integrated command and data path. An external controller connected to the command inputs determines what command or data is to be sent. HOTLink determines if a Special Character (Command) should be sent by the status on SC/$\overline{D}$ pin (Special Character/$\overline{\text{Data}}$).

The integrated command and data paths of HOTLink allow a simpler, more conventional controller architecture. Instead of creating a separate command path, command codes can be integrated within the data stream and a ninth bit (the SC/$\overline{D}$ bit) can be added to indicate the status of the associated 8 bits of information.

More Serial Outputs

TAXI has one pair of differential ECL outputs. The HOTLink transmitter has three identical differential Pseudo ECL (PECL) serial output ports, any number of which can be disabled to conserve power. Additionally, two of these outputs may be switched off with the use of the FOTO (Fiberoptic Transmitter Off) pin. The AMD™ devices, on the other hand, have only one differential PECL output pair.

The additional HOTLink outputs can be used in a system to provide redundant data paths, for loopback testing, or for building complete networks where a single transmitter is received by multiple receivers.

More Serial Inputs

The TAXI Receiver has a single pair of differential inputs (SERIN±). The HOTLink Receiver has multiple interfaces to the serial transmission medium (INA± and INB±). As in the case of the

HOTLink Transmitter, the additional media inputs of the HOTLink Receiver can be used to provide loop-back testing, redundant transmission paths, or more complex network configurations. The TAXI SERIN− is used to control Test Mode function of the part. In addition to limiting the “in-circuit” test ability of TAXI, this limits the common mode range of the TAXI receiver.

Loop-back testing is easily accomplished with the multiple media interface features of the HOTLink devices. In a typical network-style configuration, both a transmitter and a receiver will exist for each node. By connecting one of the three output pairs of the transmitter to the the second input pair of the receiver as shown in *Figure 8*, the system gains the ability to perform a complete self-check upon system initialization or when an excessive amount of errors are received over the transmission link. HOTLink provides the ability to check the transmitting and receiving devices as well as the associated serial transmission link.

Neither complete system diagnostics nor integrated loop-back testing can be accomplished with the TAXI−1 devices. Many TAXI−1 based systems attempt this feature with single-ended ECL multi-

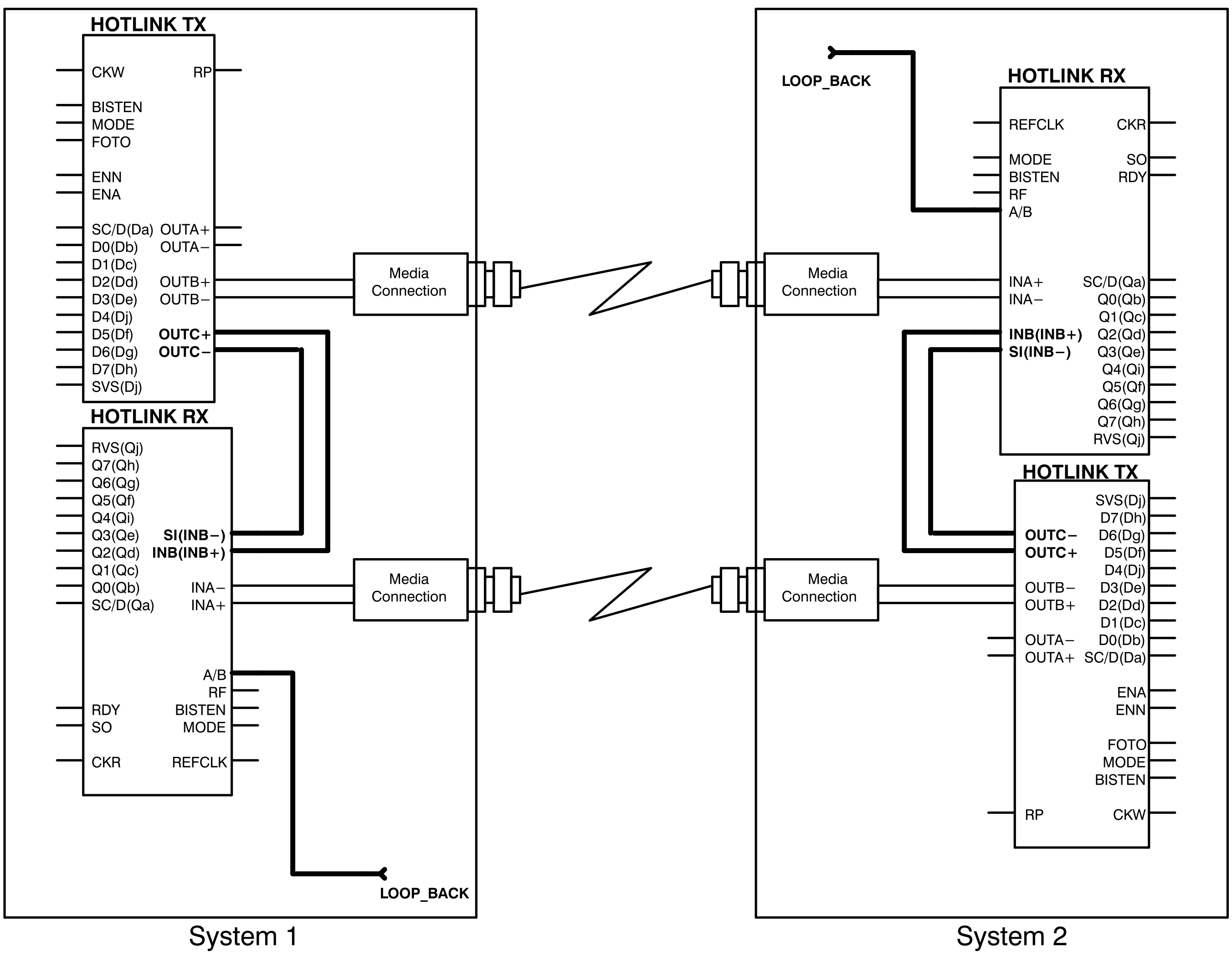

Figure 8. Example HOTLink Loop-Back System Connection

plexers, as shown in *Figure 9*. This solution compromises system reliability and performance.

Superior Data Encoding

Both the HOTLink and TAXI devices map the 8 bits of incoming transmission data into 10-bit transmission characters. The TAXI−1 devices accomplish this task by changing each pair of 4-bit nibbles of data into a pair of 5-bit transmission symbols according to the ANSI X3T9.5 (FDDI) standard. HOTLink, on the other hand, converts each 8 bits of data into a 10-bit transmission symbol according to the ANSI X3T9.3 Fibre Channel and ESCON (Enterprise System CONnection) specifications.

The primary purpose of converting the 8 data bits to 10 transmission bits is to include clock information in the data stream. A code is selected that maps each user character to a transmission character. This mapping insures that the data stream contains enough signal transitions to insure that the receiver PLL stays frequency and phase locked with the incoming data. By including the clock along with the data, the receiver is able to sample the incoming stream of data at the correct rate and position. For example, without this embedded clock information there would be no way of knowing if 1000, 999, or 1001 1s were sent in a row.

While the 4B/5B code used in TAXI−1 merely insures that the transition density of the serial bit stream is maintained, the 8B/10B code used in HOTLink also maintains the DC balance of the signal on the transmission line. This code maintains DC balance by insuring that, on the average, the number of 1s sent is equal to the number of 0s. This improves system performance by reducing the low-frequency “base-line” wander that causes jitter.

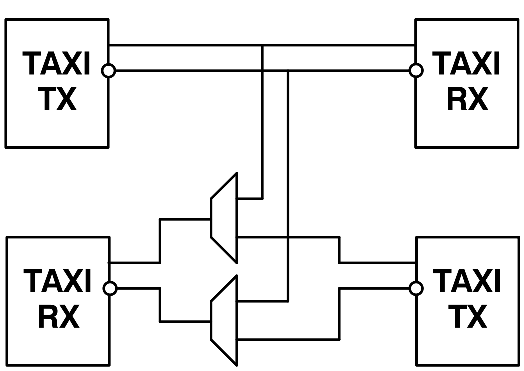

Figure 9. Loop-Back Testing with Multiplexers

More Robust Reframing Capabilities

To reassemble the incoming data stream into parallel data words, the receiver must know which bit location is the beginning of each byte. The transmitter must send SYNC characters to let the receiver know the location of byte boundaries. The TAXI−1 Transmitter sends a SYNC character when neither Data or Command is strobed into the part. At the receiver, this character is decoded as a command and the command strobe is pulsed.

The HOTLink Transmitter also sends a SYNC character when neither data nor command information is latched into the device. And again, at the receiver this character is decoded and the SC/$\overline{D}$ is held HIGH. HOTLink, however, differs in some very important ways from TAXI−1 devices.

TAXI−1 does not have a method for sending a SYNC character as part of the user character stream. HOTLink has a dedicated code that forces a SYNC character to be sent. This is important for controllers that wish to send a SYNC character at the beginning of each packet to insure that previous framing errors do not affect the current packet of data. This simplifies the controller and parallel data interface since the code can be embedded in a stream of other data.

Both TAXI−1 and HOTLink pad the spaces between data packets with SYNC characters. When the “No STRB” condition exists with TAXI−1 or the “No Enable” condition exists with HOTLink, the transmitter fills the unused bandwidth with JK (TAXI−1) or K28.5 (HOTLink). This pad string must be identified at the receiver so that the receiving system is not forced to process this information.

TAXI−1 has no method for ignoring multiple SYNC characters and preventing them from being passed to the receiving system. This is important in systems that have bursty data transmission or transmit data slower than the maximum TAXI−1 data operating frequency. If multiple SYNCs are passed to the outputs of the receiver, the receive FIFO will overflow with useless SYNC characters and this will require external decoder logic to discard the extraneous information. HOTLink eliminates this problem by only presenting the last SYNC character in a string of SYNC characters (the first SYNC character of a new packet of information) to the outputs of the receiver. This prevents redundant information from being passed to the receive system, yet maintains packet boundaries for easy packet identification.

Occasionally transmission links will experience noise that transforms part of the information stream into a SYNC character (an alias SYNC). This may cause the receiver to incorrectly identify the byte boundary and cause all of the following information to be misframed. This will continue to occur until the transmitter sends an intentional SYNC symbol. The TAXI−1 devices have no method to prevent this unintended reframing. HOTLink can prevent this in two ways.

The first way HOTLink prevents misalignment is provided by its ability to disengage reframing under user control with the RF (reframe) pin. In systems that need reframing only between packets, or only during supervisory functions, the reframe option can be selectively activated or deactivated depending on the system needs.

The second way HOTLink prevents misalignment is provided by its multi-byte framing capability. After the initial start-up phase, approximately 2K bytes after reframe (RF=HIGH) has been activated, the receiver will no longer frame on just one SYNC character, but instead requires at least two SYNC characters separated by exactly 0, 10, 20, or 30 bits of valid data as shown in *Figure 10*.

The multi-byte reframe option is useful in systems that wish to keep the Reframe option activated continuously, but do not want to suffer the data corruption consequences of erroneous misalignment. Systems that stay connected for long communication sessions (e.g., point-to-point data recovery) rarely need to be Reframed since the receiver will rarely lose byte alignment. In these systems, the protocol, or an external timer, can control reframing and only enable framing when it is required. On the HOTLink Receiver this will save 50 mW of power.

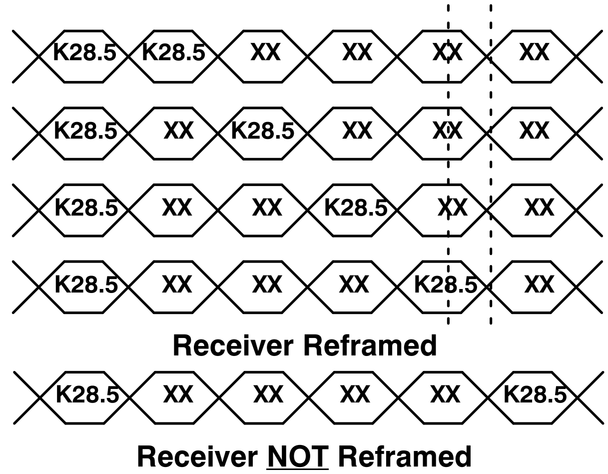

Figure 10. Double-Byte Reframing

For systems that are reconnected often (switched systems) the need to quickly reacquire byte synchronization requires that Reframe be continuously enabled. Multibyte framing available with HOTLink protects these systems from alias SYNC characters.

Higher Operating Frequency

HOTLink provides the biggest improvement in a system upgrade by allowing operation at nearly twice the rate of the TAXI−175 devices and nearly 2.5 times the rate of the TAXI−125s. The range of the TAXI−1 devices is 40 to 175 MBaud whereas HOTLink operates from 160 to 330 MBaud. This increased operating frequency range provides the ability to transfer data at over twice the rate of an equivalent TAXI system.

Built-In Self-Test Capabilities

BIST (Built-In Self-Test) can be used to test the transmitter, receiver, and the serial data link connecting them. During BIST (See *Figure 11*), the transmitter repeats a pattern representing all possible data and command characters, decodes them into transmission symbols and passes them to its outputs. The receiver, while in BIST, waits for the symbol that represents the beginning of the BIST pattern. It then decodes this symbol and every following symbol and compares it with an internally generated pattern that matches those produced by the transmitter's pattern generator. Error signals are indicated with pulses on the RVS (Received

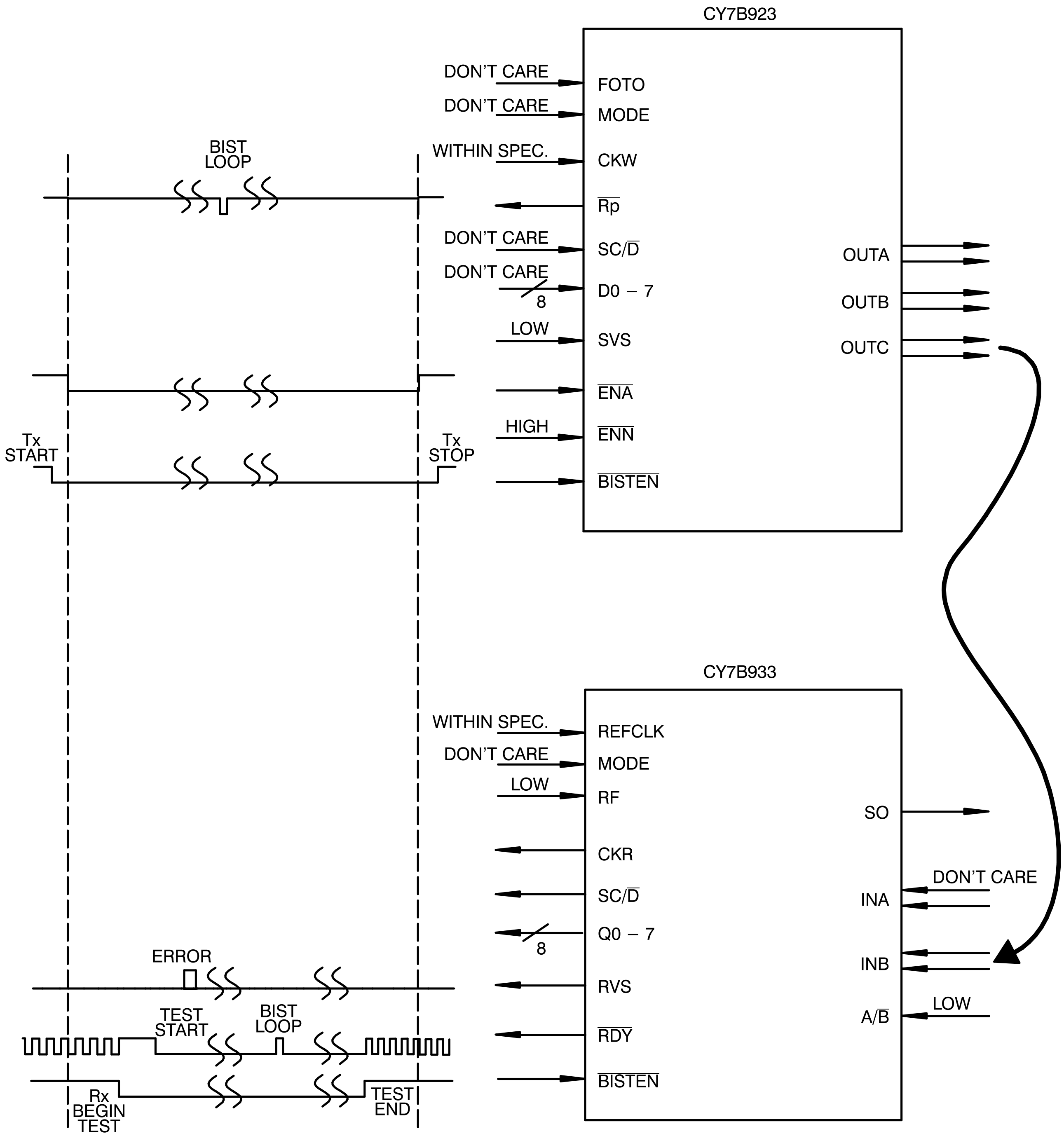

Figure 11. Built-In Self-Test

Violation Symbol) while completed BIST loops are indicated with pulses on the $\overline{\text{RDY}}$ line. The BIST function, therefore, checks the entire function of the transmitter (except the transmitter input pins and the bypass function in the Encoder), the serial link, and the receiver.

These functions can not be implemented with the TAXI devices. A substantial amount of additional circuitry would need to be added to a TAXI system to imitate this function. This type of testing is necessary for many types of diagnostics including device functionality and link integrity.

Simplified Synchronous Interface

The TAXI−1 devices have two methods of strobing data into the devices, synchronous and asynchronous. In the asynchronous mode of operation, a strobe line is used in conjunction with an acknowledge line to present data to the device. In this mode of operation the maximum byte-rate frequency for the TAXI−175 devices under the most ideal conditions is no faster than 14 MB/sec. The synchronous strobing feature of the TAXI−1 devices is also cumbersome. This method involves connecting the strobe to the clock line.

HOTLink has a very simple interface that allows seamless connection to both asynchronous and clocked FIFOs. On the transmitter, two enable inputs control when data is to be transmitted. When the $\overline{\text{ENA}}$ input is asserted, data on the data lines is serialized and transmitted. When the $\overline{\text{ENN}}$ line is asserted, data that is presented on the data lines during the next rising edge of the CLK input is transmitted. This allows efficient, synchronous state machines to control the flow of data over the serial link. In addition, the $\overline{\text{RP}}$ (read pulse) output can be connected to the $\overline{\text{R}}$ (read) input of asynchronous FIFOs, as shown in *Figure 12*, to provide a seamless asynchronous interface. The $\overline{\text{RP}}$ signal has timing that matches the timing required by asynchronous FIFOs. For clocked FIFO designs like that shown in *Figure 13*, the $\overline{\text{ENN}}$ input is used to read data from a Clocked FIFO like the Cypress CY7C453 as well as latch data into the transmitter on the next rising edge of CKW.

The receiver has an $\overline{\text{RDY}}$ output that pulses LOW each time new data has been received. The $\overline{\text{RDY}}$ output has timing that allows the receiver to be seamlessly interfaced with both asynchronous and clocked FIFOs as shown in *Figures 12* and *13*. The TAXI devices require a significant amount of additional circuitry to allow interfacing with FIFOs.

Better DC Specifications

The maximum current specifications of the TAXI−1 Transmitter operating at 17.5 MB/sec is 265 mA. The maximum current specification of the HOTLink Transmitter, on the other hand is 80 mA even when operating at 33 MB/sec.

The TAXI−1 Receiver requires a maximum of 350 mA to operate at 17.5 MB/sec whereas the HOTLink Receiver requires only 150 mA when operating at 33 MB/sec.

The TAXI−1 devices require 300 mV of differential input voltage at the receiver to accurately recover the clock and data from the input serial data stream. The HOTLink Receiver requires only 50 mV of differential input voltage. This translates into lower error rates, increased noise margins, higher jitter tolerance, and longer transmission distances when compared with the TAXI−1 devices.

Loop-Back Testing Capabilities

TAXI−1 has no loop-back testing capabilities. As mentioned previously, the redundant inputs and outputs on the HOTLink devices allow in-system loop-back testing to be performed. An additional output from the transmitter can be connected to an unused input of the receiver. The transmitter/receiver pair of an individual port can be tested together by simply switching the receiver from the link input to the Loop-back input as shown in *Figure 8*.

Ability to Send Violations

The TAXI−1 Transmitter has no method of sending violations. The TAXI Receiver has no unambiguous violation indication. Many, but not all, errors will be indicated as C0 (SYNC) while others will be indicated as other commands. In many systems it is important to explicitly send violations. In normal system operation, a violation can be caused by either a received symbol having no corresponding decode

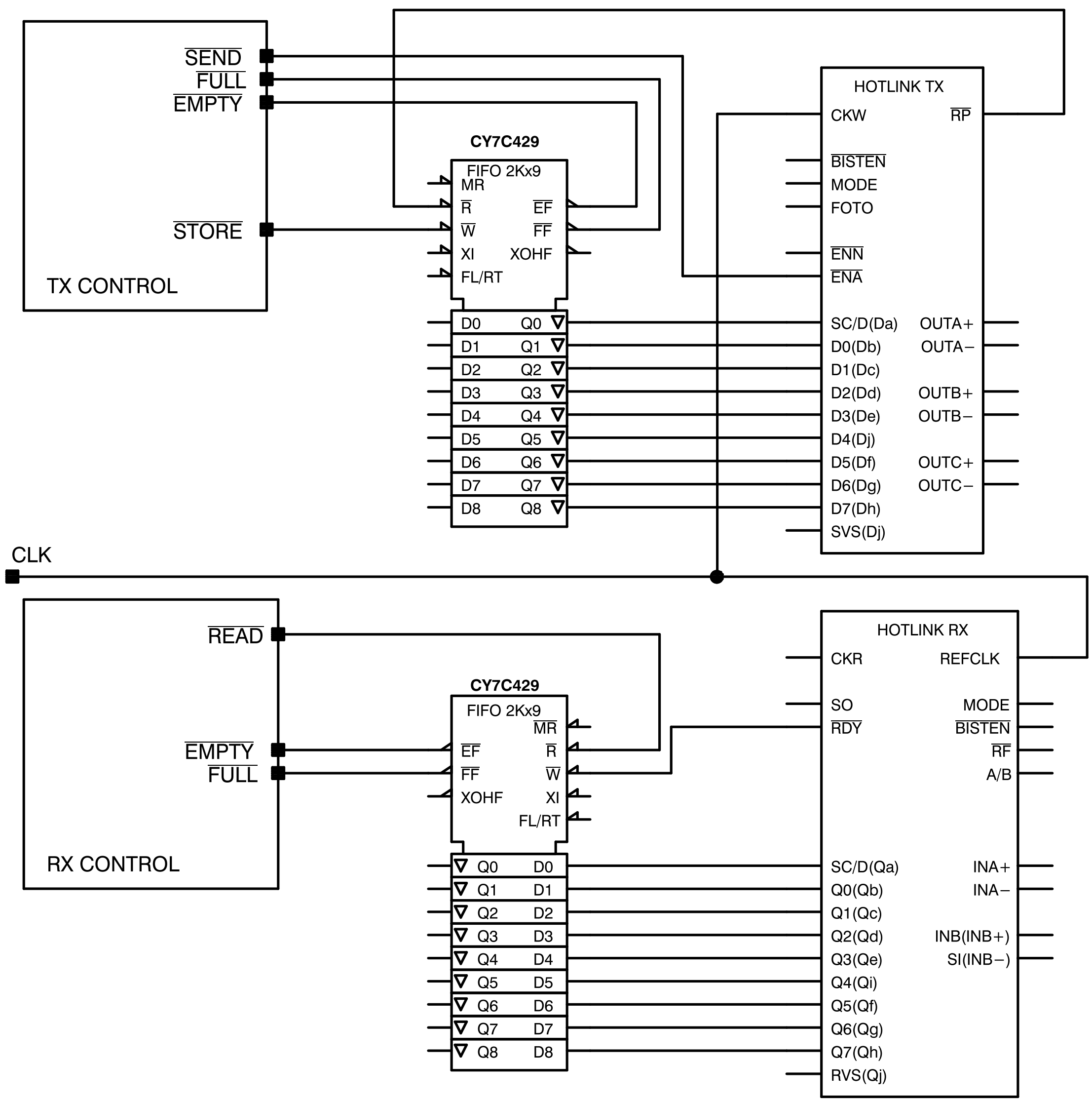

Figure 12. Asynchronous FIFO Interface

value in the receiver, or a valid code received with the wrong running disparity. Sending a violation code on purpose is useful for testing, signaling, and interrupting the receiving system.

The HOTLink Transmitter, on the other hand, provides two mechanisms to allow a system to send a pattern that will translate into a Code Rule Violation at the receiver. Various codes are included in the Special Character (SC) codes to send code rule and Running Disparity (RD) violations as part of the normal data stream. The SVS (Send Violation Symbol) pin allows an external supervisory system to force errors on an otherwise undisturbed data stream. Received errors are unambiguously indicated in the received data stream. All errors also generate an indication on the RVS (Received Violation Symbol) pin to be used by external supervisory logic.

Ability to Turn-Off Serial Output Stream

There is no method of turning off the serial output of the TAXI−1 devices. The FOTO (Fiberoptic Transmitter Off) is an input found on the HOTLink Transmitter that allows the OUTA and OUTB differential outputs to be logically turned off. Laser

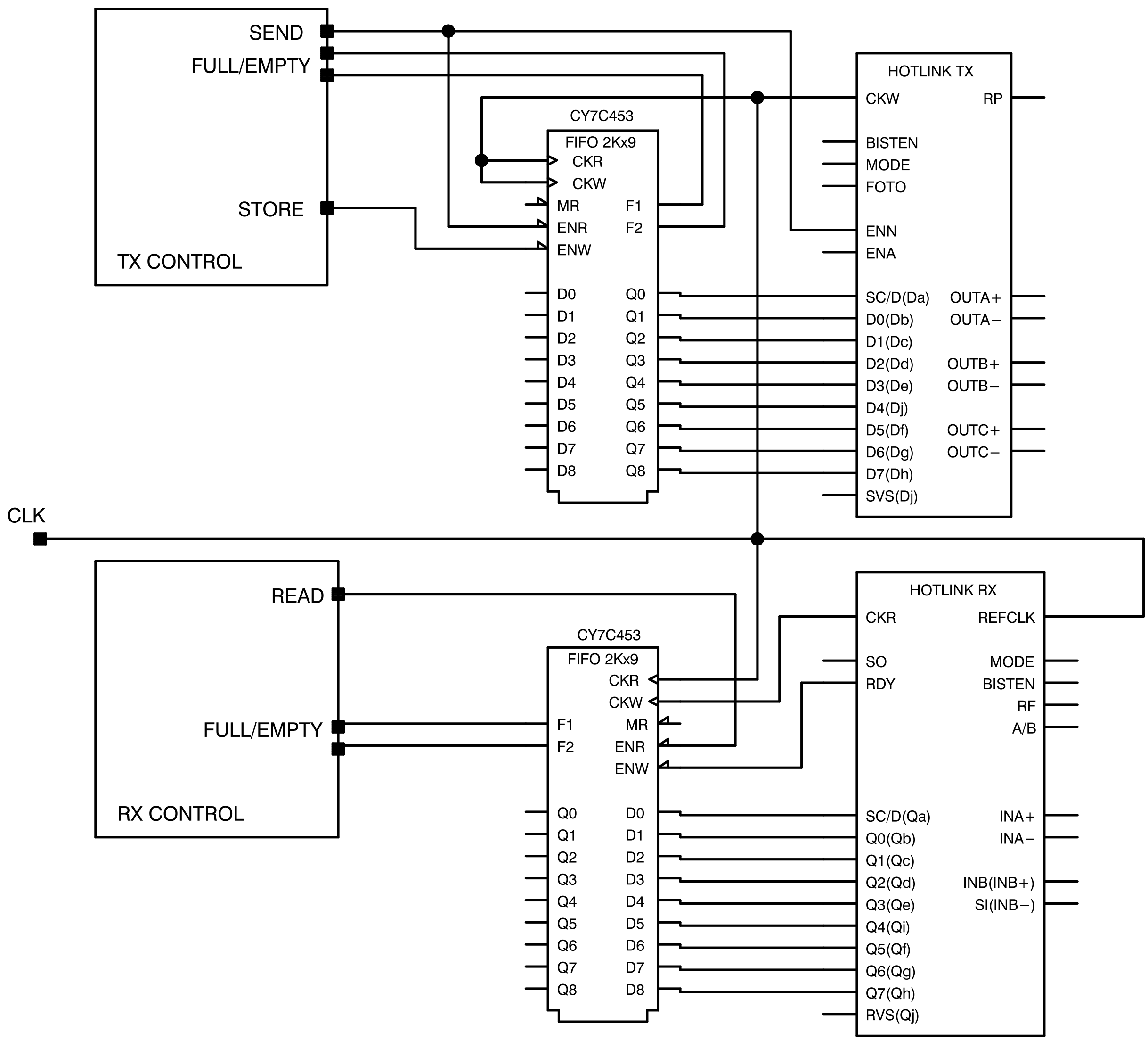

Figure 13. Clocked FIFO Interface

safety systems can use this input to shut off the lasers in the case of a fiber disconnect.

ECL to TTL Translator

The TAXI devices have no ECL to TTL translator. The HOTLink devices have a built-in ECL to TTL translator. The SI input takes the single-ended ECL 100K (+5V referenced) input and the translated TTL signal is presented at the SO output. The system can utilize this translator to convert a carrier-detect signal into its TTL equivalent for use by a controller.

Modifying the System for HOTLink

Listed below are some simple system modifications that can be performed in lieu of modifying the entire system architecture for designers who currently use the TAXI−1 devices and wish to easily upgrade to the HOTLink devices in order to take advantage of its performance and architectural improvements.

Multiplexed Command and Data

Most systems have, at some level, an integrated command and data path, much like that used in HOTLink. These systems explicitly demultiplex these paths to make themselves compatible with the TAXI architecture. These systems can easily take advantage of the HOTLink architecture by removing the unnecessary multiplexing circuitry and allowing the demultiplexer control line and the data/command lines to drive HOTLink directly.

Some systems, however, send command codes out-of-band with respect to the data stream. These systems can be easily modified by adding a simple multiplexer external to HOTLink or external to the FIFO that drives HOTLink. The MUX select can be driven by the AND of the command lines.

Data Words Longer than 8 Bits

Most data words that need to be encoded are 8 bits in length. In a few cases, however, the data that needs to be encoded is 9, or even 10, bits in length. In these cases, an external multiplexer can be used external to a FIFO that would put each half of the 9- or 10-bit data word into the FIFO separately. At the receiving end, the same operation would be performed in reverse. This is possible due to the extended operating frequency of HOTLink.

Asynchronous Strobing

HOTLink provides a very user-friendly synchronous interface. For asynchronous operation, a FIFO can be used to interface the two asynchronous entities.

Mapping Command Codes

In 8-bit mode, TAXI-1 has 15 different command codes (see *Table 1*), while HOTLink can transmit and receive 12 specific codes (see *Table 2*). Several of these TAXI command codes have restrictions on their usage. HOTLink has no restrictions on the use of any codes. If the system must use more than 12 codes, an easy way to expand the command set is by utilizing a specific code that indicates that the next data word is also a command code. Using this method, or a simple extension of this method, allows nearly an infinite command code set to be transmitted and received.

Operating Frequency

The operating frequency of HOTLink is much faster than the TAXI devices. No design issues need to be considered in systems that wish to operate their parallel side at the same rate and take advantage of the increased system flexibility and functionality that HOTLink offers. When the system has no data to send over the transmission link, HOTLink simply sends strings of SYNC characters automatically. These SYNC characters are ignored on the receiving end. So, whenever the transmitting side of the link does not present data to the transmitter, a SYNC character will be sent. These characters, although used to keep the receiver in lock with the transmission stream, will not be presented as a character to the outputs.

Conclusion

The HOTLink Transmitter and Receiver have many advantages over the AMD Am7968 Transmitter and the Am7969 Receiver (TAXI−1). These advantages include those listed below.

- Multiplexed command and data
- More serial outputs
- More serial inputs
- Superior data encoding
- More robust reframing
- Higher operating frequency
- Built-in self-test
- Simplified synchronous interface
- Reduced power consumption
- Loop-back testing capabilities
- Ability to send violations
- Ability to turn off serial output stream
- ECL-to-TTL translator

Table 1. TAXIchip Command Symbols

Am7968 Transmitter				Am7969 Receiver	
Command Input				Command Output	
HEX	Binary	Encoded Symbol	Mnemonic	HEX	Binary
8-Bit Mode					
0	0000	XXXXX XXXXX	Data	No Change	No Change
No STRB	No STRB	11000 10001	JK (8-bit Sync)	0	0000
1	0001	11111 11111	II	1	0001
2	0010	01101 01101	TT	2	0010
3	0011	01101 11001	TS	3	0011
4	0100	11111 00100	IH	4	0100
5	0101	01101 00111	TR	5	0101
6	0110	11001 00111	SR	6	0110
7	0111	11001 11001	SS	7	0111
8[1]	1000	00100 00100	HH	8	1000
9	1001	00100 11111	HI	9	1001
A[1]	1010	00100 00000	HQ	A	1010
B	1011	00111 00111	RR	B	1011
C	1100	00111 11001	RS	C	1100
D[1]	1101	00000 00100	QH	D	1101
E[1]	1110	00000 11111	QI	E	1110
F[1]	1111	00000 00000	QQ	F	1111

Note

1. While these Commands are legal data and will not disrupt normal operation if used occasionally, they may cause data errors if grouped into recurrent fields. Normal PLL operation cannot be guaranteed if one or more of these commands is continuously repeated.

Table 2. HOTLink Valid Special Character Codes and Sequences (SC/$\overline{D}$ = HIGH)

HOTLink Special Code Byte Name	Special Code Code Name	Bits HGF	Bits EDCBA	Current RD− abcdei	Current RD− fghj	Current RD+ abcdei	Current RD+ fghj	Receiver Output Code Name
K28.0	C0.0 (C00)	000	00000	001111	0100	110000	1011	C0.0
K28.1	C1.0 (C01)	000	00001	001111	1001	110000	0110	C1.0
K28.2	C2.0 (C02)	000	00010	001111	0101	110000	1010	C2.0
K28.3	C3.0 (C03)	000	00011	001111	0011	110000	1100	C3.0
K28.4	C4.0 (C04)	000	00100	001111	0010	110000	1101	C4.0
K28.5	C5.0 (C05)	000	00101	001111	1010	110000	0101	C5.0
K28.6	C6.0 (C06)	000	00110	001111	0110	110000	1001	C6.0
K28.7	C7.0 (C07)	000	00111	001111	1000	110000	0111	C7.0
K23.7	C8.0 (C08)	000	01000	111010	1000	000101	0111	C8.0
K27.7	C9.0 (C09)	000	01001	110110	1000	001001	0111	C9.0
K29.7	C10.0 (C0A)	000	01010	101110	1000	010001	0111	C10.0
K30.7	C11.0 (C0B)	000	01011	011110	1000	100001	0111	C11.0
Sequences								
Idle	C0.1 (C20)	001	00000	−K28.5+,D21.4,D21.5,D21.5, repeat				C5.0, D21.4, D21.5, D21.5
R_RDY	C1.1 (C21)	001	00001	−K28.5+,D21.4,D10.2,D10.2, repeat				C5.0, D21.4, D10.2, D10.2
EOFxx	C2.1 (C22)	001	00010	−K28.5,Dn.xxx0		+K28.5,Dn.xxx1		C5.0,Dn.xxx0 or C5.0,Dn.xxx1
Follows K28.1 for ESCON Connect-SOF (Rx indication only)								
C−SOF	C7.1 (C27)	001	00111	001111	1000	110000	0111	C7.1
Follows K28.5 for ESCON Passive-SOF (Rx indication only)								
P−SOF	C7.2 (C47)	010	00111	001111	1000	110000	0111	C7.2
Code Rule Violation and SVS Tx Pattern								
Exception	C0.7 (CE0)	111	00000	100111	1000	011000	0111	C0.7
−K28.5	C1.7 (CE1)	111	00001	001111	1010	001111	1010	C5.0 or C1.7
+K28.5	C2.7 (CE2)	111	00010	110000	0101	110000	0101	C5.0 or C2.7
Running Disparity Violation Pattern								
Exception	C4.7 (CE4)	111	00100	110111	0101	001000	1010	C4.7

References

1. Cypress Semiconductor, *CY7B923/CY7B933 HOTLink Transmitter/Receiver Preliminary Datasheet*, Cypress Semiconductor High Performance Data Book, August 1, 1993.
2. Cypress Semiconductor, *HOTLink Design Considerations Application Note*, October 1993.
3. Advanced Micro Devices, TAXIchip Integrated Circuits Transparent Asynchronous Transmitter/Receiver Interface Am7968/Am7969−125 Am7968/Am7969−175 Data Sheet and Technical Manual, 1992
4. Advanced Micro Devices, Am79168/Am79169−275 TAXI−275 Integrated Circuits Technical Manual, Rev. 1.0, 1993.
5. Advanced Micro Devices, Am79168/Am79169−275 TAXI−275 Transmitter/Receiver Transparent Asynchronous Transmitter/Receiver Interface Preliminary Data Sheet, March 1993 Rev B.

Document #: COM0008−0194