Frequently Asked Questions about HOTLink™

The following questions are frequently asked by customers who are evaluating HOTLink™ products. These cursory answers will serve as an introduction for each topic. Separate application notes cover these topics in more complete detail.

1. How far can HOTLink communicate over various media?

HOTLink has no intrinsic distance limit. The two issues that determine the distances over which data can be sent using HOTLink are: (1) the choice of interconnect media (fiber-optic cable, coaxial cable, twisted-pair cable, etc.); and (2) the jitter that accumulates or is injected while the data is in transit over the selected media.

HOTLink can drive all standard fiber-optic interface modules that support standard PECL interface signals. These electro-optical modules are suitable for communicating over distances from a few meters to several kilometers. Fiber-optic interconnect offers the longest distances and the lowest interference potential of all transmission media.

For lower-cost applications, HOTLink can directly drive wire transmission lines. The main distance determining factors when using wire links are related to the characteristics of the cable. Wire transmission lines have significant frequency-dependent attenuation that causes jitter as a direct function of the data rate and the media length. Uncompensated transmission line lengths are limited much more by jitter (and the jitter tolerance of the receiver) than by actual signal attenuation. The detrimental effect of jitter can be lessened with the addition of a suitable attenuation compensation filter that matches the attenuation characteristics of the cable. This filter trades receiver differential voltage amplitude for jitter reduction and increases the possible transmission distance. When using wire transmission lines, other issues beyond transmission distance often determine transmission line suitability. These issues include both radiated emissions and susceptibility to external disturbance that must be examined prior to selection of a link media type.

Some typical wire types and uncompensated transmission distances over which HOTLink can communicate are shown in *Table 1*. A simple compensation filter, built from passive components, can increase reliable transmission distance to more than twice these distances.

For more information see the application note “HOTLink Copper Interconnect—Maximum Length vs. Frequency.”

Table 1. Coaxial Cable Types

Coaxial Cable	50Ω	75Ω	75Ω	93Ω
160 Mbaud	RG−58 A/U − 350 ft	RG−6 A/U − 900 ft	RG−59 A/U − 525 ft	RG−62 A/U − 675 ft
266 Mbaud	RG−58 A/U − 225 ft	RG−6 A/U − 600 ft	RG−59 A/U − 350 ft	RG−62 A/U − 400 ft
330 Mbaud	RG−58 A/U − 115 ft	RG−6 A/U − 500 ft	RG−59 A/U − 250 ft	RG−62 A/U − 325 ft

Table 2. Twisted Pair Cable Types

Shielded Twisted Pair	150Ω
160 Mbaud	IBM®−Type 1 − 550 ft
266 Mbaud	IBM−Type 1 − 350 ft
330 Mbaud	IBM−Type 1 − 275 ft

Unshielded Twisted Pair	UTP3	UTP5
160 Mbaud	140 ft	280 ft
266 Mbaud	80 ft	180 ft
330 Mbaud	60 ft	130 ft

2. Can the PECL inputs and outputs of HOTLink products be connected to ECL (−5.2V) products?

The +5.0V PECL inputs and outputs are directly compatible with true ECL (10K, 10KH, 100K, etc.) running on +5V power supplies. Connections between the HOTLink PECL I/O and ECL running on −5.2V is easily accomplished by capacitor-coupling the serial data lines. Details on this coupling technique are included in the Cypress application note "HOTLink Design Considerations."

3. What happens when the ECL inputs of the HOTLink Receiver are left open?

All of the ECL inputs on the HOTLink Receiver have internal pull-down resistors to assure that ECL-emitter follower outputs will see a positive input current (approximately 250 µA into the pin) at all normal ECL voltages. Thus, all single-ended ECL inputs (i.e., A/$\overline{B}$, SI, INB) will float to a logical LOW level. (These pull-downs will not sink enough current to act as the normal ECL output termination. They are only intended to prevent the emitter-follower oscillations caused by negative input-impedance that are possible in some less robust designs.) Open inputs will be interpreted as follows: A/$\overline{B}$ = LOW will cause the Receiver to accept data from the INB serial inputs; SI = LOW will cause the SO output to assume a LOW output state; INB = LOW will be interpreted as an input with no data (assuming A/$\overline{B}$ is also LOW). No data is interpreted as an error (RVS=HIGH & C0.7 in Encoded mode, and Qa−j outputs LOW in Bypass mode) and will cause the internal clock-synchronizer phase-locked loop (PLL) to track the REFCLK input frequency.

The internal resistor network used to pull the differential serial data inputs (i.e., INA± and INB±) will cause unconnected inputs to rest at approximately 2.0V. This resting voltage is a byproduct of the internal resistive attenuator used to enhance input-common mode range. If both inputs of a differential pair are left unconnected, the inputs will be in an undefined state and HOTLink receiver behavior will be unpredictable. Stray, non-differential noise that appears on these unconnected inputs will be amplified and interpreted as serial data. This will cause random parallel-data output changes, and may cause the PLL to wander or drift away from the REFCLK frequency. One input of an intentionally unused differential-pair should be terminated to V_{CC} through a 1−5 KΩ resistor to assure that no data transitions are accidentally created.

4. What special power-supply bypassing is required for HOTLink products?

HOTLink requires no special considerations for power-supply bypassing beyond that normally associated with high speed logic. This typically includes the use of a ground plane, a split V_{CC} plane, and multiple chip bypassing using RF quality capacitors. Each of the ground pins of a HOTLink IC should connect directly to the ground plane using short (<.25") traces and vias. All of the V_{CC} pins should connect to a V_{CC} pad under the HOTLink and then connect to the board V_{CC} through a single via. Connect one 22-nF capacitor for each V_{CC} pin directly from the pin to GND. For more information see the "Using Decoupling Capacitors" application note.

5. If the HOTLink Receiver is switched from INA to INB, how long will it take for the PLL to re-lock?

Assuming that the data on both INA and INB are within the ±0.1% frequency offset described in the HOTLink datasheet, the phase-locked loop (PLL) will acquire and lock to the new data stream within a few byte times. The exact time required involves statistical probabilities related to phase, frequency, and jitter, and cannot be exactly predicted. Empirical testing using normal data patterns shows that the time required to achieve absolute minimum phase error with the new data stream will vary from zero to about ten bytes.

An operational serial link will produce valid parallel data much earlier than the amount of time required to achieve minimum phase error, since instantaneous phase error is accommodated as jitter. The wide jitter tolerance offered by the HOTLink Receiver will minimize the time that data is incorrectly interpreted during phase acquisition. The larger problem facing a system protocol that allows switching of serial data streams, is byte synchronization (byte-framing). After the data-stream has been switched, it must be reframed. This requires that a K28.5 (or two K28.5s within five bytes if multibyte framing is enabled) must be received. The time that elapses before this happens depends on the system protocol and the timing of the data input switch. Correct data might not come out of the HOTLink Receiver for hundreds of byte times due to reframing regardless of speed of phase acquisition.

For more information, refer to the Receiver Data-Phase Acquisition Time section of the "HOTLink Jitter Characteristics" application note.

6. If the connection between the HOTLink Transmitter and Receiver is briefly interrupted, how long will it take for the PLL to re-lock?

The exact behavior of the HOTLink Receiver depends on the length and cause of the interruption. If the interruption is synchronous with the data (i.e., data bits disappear without any significant disturbance to the placement of the final few data transitions), and lasts for less than a few dozen bytes, it is probable that the PLL will relock on the very first bit. If the interruption is asynchronous (i.e., the timing of the final few transitions is disturbed) or if the synchronous interruption lasts longer than a few dozen bytes, the PLL will relock within the first one or two bytes after resumption of the data stream. If a long interruption occurs that is not synchronous to byte boundaries, the receiver may lose byte synchronization when the PLL relocks. In this case, the data will need to be reframed.

If the interruption is asynchronous, and the link interface allows noise to be injected into the serial inputs of the HOTLink Receiver, the time to relock the PLL becomes much harder to predict. If the noise that is being injected causes the PLL to track within its frequency offset limits (approximately ±0.25% of the REFCLK frequency) the PLL will reacquire in a few bytes (typically less than ten) after a good data stream reappears. If the PLL frequency has been moved to its offset limits by the input noise, it may take more than 60−70 bytes before the PLL locks to the good data. When the PLL hits the frequency offset limit, it will recenter itself at the REFCLK frequency and then attempt to lock to the data. While the PLL is out of lock (after experiencing a data stream interruption) the frequency of CKR will not wander beyond the offset limits.

For more information, refer to the Receiver Data-Phase Acquisition Time section of the "HOTLink Jitter Characteristics" application note.

7. If the connection between HOTLink Transmitter and Receiver is broken, what will come out of the receiver?

The exact behavior of HOTLink Receiver is difficult to predict when the serial data link is broken, since there are so many ways that the link itself can behave. The following behaviors are most common;

Bypass Mode−Reframe−OFF (RF = LOW) Clean link break with no extraneous noise input into serial inputs:

- CKR runs at REFCLK frequency.
- $\overline{RDY}$ is always HIGH.
- Qa−j all go LOW or HIGH depending on exact offsets built into transmission line termination. If the terminations are exactly matched, then Qa−j may be indeterminate.

Bypass Mode−Reframe−OFF Noise injection into serial inputs:

- CKR runs at REFCLK frequency ±<1.0% (typically <±0.25%) and may wander between its range limits and the center frequency, randomly controlled by the injected noise.
- $\overline{RDY}$ may rest HIGH or may pulse randomly as false K28.5s are decoded from the noise.
- Qa−j will be indeterminate and may switch randomly.

Encoded Mode−Reframe−OFF Clean break with no extraneous noise input into serial inputs:

- CKR runs at REFCLK frequency.
- $\overline{RDY}$ pulses once per byte.
- Q_{0-7} indicate C0.7, SC/$\overline{D}$ is always HIGH, RVS is always HIGH if there are any offsets built into transmission line termination. If the terminations are exactly matched, then Q_{0-7}, SC/$\overline{D}$ and RVS may be indeterminate.

Encoded Mode−Reframe−OFF Noise injection into serial inputs:

- CKR runs at REFCLK frequency ±<1.0% (typically <±0.25%) and may wander between its range limits and the center frequency randomly controlled by the injected noise.
- $\overline{RDY}$ may pulse randomly or once per byte.
- Q_{0-7}, SC/$\overline{D}$ and RVS may be indeterminate and may switch randomly.

Either Mode−Reframe−ON Noise injection into serial inputs:

- CKR runs at REFCLK frequency ±<1.0% (typically <±0.25%) and may wander between its range limits and the center frequency randomly controlled by the injected noise. If RF has been HIGH for less than 2048 bytes, CKR will stretch randomly as false K28.5s are decoded from the noise. If RF has been HIGH for more than 2048 byte-times, CKR will only stretch when a multiple K28.5 string is decoded from the noise.
- $\overline{RDY}$ may pulse randomly or once per byte.
- Q_{0-7}, SC/$\overline{D}$ and RVS may be indeterminate and may switch randomly.

8. What is the correct operation of the RF input on the receiver? What is the minimum number of K28.5 characters required to insure proper framing? How can I tell if the receiver is framed properly?

Recovery of information from a serial data stream requires recovery of the bit clock (accomplished by the receiver PLL) and byte synchronization (accomplished by the receiver framer). The HOTLink framer is enabled or disabled by the RF input. In well behaved, standardized point-to-point protocols that are seldom switched, the control of the byte framer is managed as a service in the protocol controller. This service monitors when some error criteria have been exceeded, and goes to a framing subroutine. This framer service sets RF=HIGH while framing and LOW during normal message transactions.

In less well behaved systems, or systems that switch data sources often, it may be necessary to leave RF=HIGH for long periods (or permanently). Leaving RF HIGH opens the system to the problem of data corruption in the serial link caused by data patterns that happen to match the SYNC character. Since this Alias SYNC is unlikely to be aligned to the normal byte boundaries, it will cause the framer to align the parallel data to the wrong byte boundary resulting in long running data corruption. When RF is set HIGH, the receiver searches the received data stream for the bit pattern matching K28.5 (001111 1010 or 110000 0101). When it is found, the internal bit counter that controls byte translation is reset and the byte boundaries are aligned to the SYNC character.

HOTLink minimizes the alias SYNC problem by incorporating a multi-byte framer into the receiver. If RF has been HIGH for less than 2048 bytes, as would be typical in protocol driven framing control, a single K28.5 will align the byte boundaries. If RF has been HIGH for more than 2048 bytes, as would be typical in packet switched systems, the multi-byte framer is enabled and a single K28.5 is no longer sufficient to align the byte boundaries. To minimize the risk of alias SYNC, reframing is only allowed when two K28.5s are detected. These two K28.5s can be adjacent, or separated by exactly one, two, or three transmission characters. Any other spacing (i.e., non-integral character separation, or too far between K28.5) is assumed to be caused by transmission errors and will be ignored for framing purposes.

In addition to the upper level protocol error detection mechanisms common in communication links, the HOTLink Receiver offers several indications that a link is misframed. For example, in Bypass mode the $\overline{\text{RDY}}$ output pulses once per K28.5 detected. If RF is LOW, the only K28.5 that can be detected is one that is properly framed, and all others will just pass through as part of the received data. If the protocol in use has a maximum packet size or a minimum number of K28.5s, a simple retriggerable-one-shot can be used to detect when framing has been lost. In this example, if the one-shot is retriggered by the properly spaced K28.5s, then the data is properly framed. If the one-shot times-out, indicating that too much time had elapsed between SYNC characters, the data would automatically be reframed by raising RF till the next K28.5 indication.

Another example of HOTLink's indication of a misframed link occurs during Encoded mode. In Encoded mode, the RVS output serves a similar if not quite as obvious function. Normal data being sent over typical data links will have a very low error rate (e.g., bit-error-rates of $1x10^{-12}$ are quite common. BER=$1x10^{-12}$ $\approx$ one error per hour at 266 MHz). Therefore, if RVS is asserted often it can be assumed that the cause is misframing. Another retriggerable-one-shot could be used to detect this condition, or it could be detected by a simple synchronous state machine constructed in a PLD.

For more information, refer to the "HOTLink CY7B933 $\overline{\text{RDY}}$ Pin Description" application note.

9. What happens to the receiver's clock and parallel outputs when it reframes?

When a byte boundary realignment occurs, the external timing of the HOTLink Receiver changes to match the new byte alignment. Logic internal to the receiver guarantees that the clock outputs (CKR and $\overline{\text{RDY}}$) never glitch. They will stretch to the new byte alignment by adding to the HIGH or LOW time of the output pulse. The exact width of the high or low times of these clock outputs will depend on the exact timing of the realignment, but neither will ever be less than that of a nominal, normally running output (i.e., five bit times, each, minimum).

The data outputs (Q_{0-7}, SC/$\overline{\text{D}}$, and RVS) all change at a time determined by internal bit-rate counters, and are timed to assure maximum set-up and hold times to down-stream logic. Since realignment will reset the cycle of the internal counter, it is possible that the outputs will change, and then change again between clock edges when byte realignment happens. Since the clock-cycle stretches, this glitch on the data output remains outside the specified data-access and hold times.

For more information, refer to the "HOTLink CY7B933 $\overline{\text{RDY}}$ Pin Description" application note.

10. What does BIST do? How can I add BIST to my system without redoing all calculations for my critical interface timing? What functionality does the BIST test and guarantee?

The HOTLink built-in self-test allows a clear and unambiguous check of the HOTLink Transmitter and Receiver, and the serial link connecting them. As part of an offline diagnostic, this feature allows the user to insure that the interconnect link is fully operational and that any other diagnostic failure indications are caused by system blocks above the physical layer. BIST allows the HOTLink adapter card manufacturer to do a quick link quality test (or node quality test with the use of the loop-back functionality of HOTLink) without the necessity of bringing up a fully functional system to do link testing.

BIST is controlled by unused HOTLink data-enable inputs. Only a few connections and minimal external logic are necessary to add BIST to an otherwise complete system. (See the Cypress application note "HOTLink Built-In Self-Test.") BIST status indications appear on the $\overline{RP}$, RVS(Qj) and $\overline{RDY}$ outputs which are easily monitored by logic internal or external to the data flow controller.

In BIST mode, the HOTLink Transmitter generates a 2^9-1 (511 byte) pseudo-random pattern using its Input register configured as a Linear Feedback Shift register. The HOTLink Receiver compares the serial BIST data stream with identical BIST patterns generated in its Output register. All of the logic in the transmitter (except the input pins) and all of the logic in the receiver (including the output pins and their attached loads) are checked by BIST. All of the serial link interconnect components are exercised with normal data patterns, which are checked byte-by-byte in real time.

11. What fiber-optic components are compatible with HOTLink products?

All standard fiber-optic interface components are compatible with HOTLink products. The following table is a representative but not comprehensive list of optical interface manufacturers. A more complete list of vendors and products is included in the "HOTLink Design Considerations" application note.

AMP/Lytel Division
61 Chubb Way
P.O. Box 1300
Somerville, NJ 08876
(908) 685-2000

CTS Corp
1201 Cumberland Ave
West Lafayette, IN 47906−1388
(317) 463-2565

Hewlett-Packard
Components Division
370 West Trimble Road
San Jose, CA 95131
(800) 535-7449 or (408) 435-6342

Siemens Fiber Optic Components
20F Commerce Way
Totowa, NJ 07512
(201) 890-1606

Sumitomo Electric
Fiber Optics Corporation
777 Old Sawmill River Road
Tarrytown, NY 10591-6725
(914) 347-3770

12. What is the significance of the HOTLink claim of "no external PLL components"?

HOTLink Transmitter and Receiver have completely integrated the PLL clock multiplier and data separator functions. These functions are implemented with high-performance phase-locked loops (PLLs) that have been tuned for maximum performance and minimum system noise sensitivity. In competitive products that purport to offer similar functions, these PLLs are often implemented with external filter and frequency setting components with the goal of achieving maximum performance. But these very same external components are the largest cause of end-user complaints and random system failures because they expose the most critical analog signals in the circuit to the external noises that abound in normal systems. External components require critical, costly and time consuming printed circuit board layout as well as high-speed analog and digital design techniques that are unfamiliar to many system integrators. HOTLink products are designed and built using fully differential analog and digital circuits to give the lowest possible output jitter and highest possible jitter tolerance. There are no external components to compromise system performance in unexpected and unpredictable ways. For more information, refer to the HOTLink Transmitter Jitter section of the "HOTLink Jitter Characteristics" application note.

13. What is the intrinsic bit-error-rate of HOTLink Transmitter and Receiver?

HOTLink BER=Zero. HOTLink Transmitter and Receiver have no intrinsic failure modes. If their power is maintained and if the interface to the link connecting them has reasonable design margin, the total error rate will be exactly that of the interconnect media. Link error rates of $<<1x10^{-15}$ are common and easily achieved. Even with worst-case design derating and end-of-life derating, BER $<<1x10^{-12}$ presents no significant challenge.

The real question being asked is, "What will be my link BER when using HOTLink?" The answer to this question involves the design of the serial transmission link and the margins designed into it. HOTLink will not significantly degrade the BER of the link. For more information, refer to the "Understanding Bit-Error-Rate with HOTLink" application note.

14. How much jitter is created by the transmitter? How much jitter is created by the receiver? What is the significance of the HOTLink Transmitter requirement for a crystal-stable clock source?

The phase-locked loops (PLLs) in the HOTLink Transmitter and Receiver act like low-pass filters to jitter that is embedded in the data or clock signal source. For the transmitter, the signal source is the CKW input. Any jitter that appears at CKW will be passed unattenuated if it has frequency components below the natural frequency of the PLL filter (approximately 500 kHz). Frequency components above the natural frequency will be attenuated at about 6 dB/octave. Frequency components that fall very near the natural frequency of the filter will be slightly amplified (approximately 0.5 dB). These are the normal characteristics of a Type-2, second-order PLL filter. When the transmitter is fed by a low jitter clock source, typical output jitter will be less than 20 ps RMS and 200 ps peak-to-peak. It is possible to measure significantly more jitter than that which is actually present if the complete system is not well understood. A few hundred millivolts of V_{CC} noise, while insignificant to the logic of a normal system board, will add imaginary jitter to the measured output. This imaginary jitter appears because a single ended oscilloscope sees the waveform as if it were measured against a fixed threshold, while the differential serial interface sees V_{CC} noise as a common mode signal to be ignored (e.g., 100 mV of V_{CC} noise could create 100−200 ps of imaginary jitter). Likewise, the normal method of measuring peak-to-peak jitter, an infinite persistence scope trace, will show larger jitter than that contributed by the HOTLink Transmitter. Low frequency jitter (wander) in the oscillator, scope trigger, temperature, and voltage related delay variations will all contribute to the width of the stored scope trace. Delay variations include TTL threshold variations that cause apparent delay variation (e.g., 100 mV of TTL threshold change can cause 100−200 ps of apparent jitter).

The signal source for the receiver is the serial data stream and, like the transmitter, it passes the frequency components of received jitter that fall below the natural frequency of its filter (approximately 300 kHz to 1000 kHz depending on actual data transition density being received). Frequency components above the natural frequency will be attenuated and there is minor jitter peaking at about the natural frequency of the PLL. Since the characteristics of the input jitter will determine the jitter content on the receiver CKR output (the only place to directly measure Rx-PLL jitter) it is somewhat difficult to predict the output jitter. Maximum CKR output jitter is less than 200 ps (peak-to-peak) when the receiver is tracking normal data (BIST data is typical) that exhibits maximum tolerable peak-to-peak jitter. Jitter from normal data is wide-bandwidth, has a significantly high-frequency content, and can have peak-to-peak amplitude of up to about 90% of a bit time. If the serial data contains a significant low frequency jitter component (typical of crystal oscillators and some pulse generators) the output jitter measured on the CKR pin could be much higher. Jitter measurements at the receiver output can be more misleading than those associated with the transmitter serial outputs, since all measurements are made on TTL outputs.

The jitter characteristics mentioned above affect system performance in the following ways. Any low-frequency jitter (below the bandwidth of either transmitter or receiver PLL) will be treated as wander.

For purposes of the PLLs, wander (usually caused by low frequency power supply variations or temperature fluctuations within the timing ICs) will not reduce the system timing margins and will not contribute to bit-error-rate. Wander can affect system timing at interfaces where the transmitter clock source is used to clock information received from a receiver tracking data from another clock source. The variation in clock frequencies may violate set-up and hold times, the exact problems usually solved by FIFO memories in typical communication systems.

High-frequency jitter (at or above the natural frequency of the PLL filters) may contribute to BER. High-frequency jitter can be caused by the clock source, media transfer characteristics, or external noise. The recovered internal bit-rate clock will not track high-frequency jitter above the PLL natural frequency. High-frequency jitter, therefore, may cause a bit edge to move into the receiver sampling window causing the bit to be erroneously sampled (a bit error).

A suitable clock source should be selected with the above effects in mind. The only clock source guaranteed to offer the required stability and high-frequency specifications is a crystal oscillator. High-frequency jitter is minimal, and low-frequency wander is usually small and very low frequency. Frequency accuracy is easily guaranteed by mechanical means, and high accuracy devices are relatively low cost. Free-running resistor-capacitor (RC) oscillators, logic gate ring oscillators or inductor-capacitor (LC) oscillators include too much high-frequency jitter, experience wide frequency variation as a function of process and environmental conditions and thus are unsuitable for this application. See the "HOTLink Jitter Characteristics" application note for more information.

15. Can I use HOTLink for anything other than Fibre Channel/ESCON™ interconnect?

HOTLink has been designed to implement the required performance and specifications of Fibre Channel and ESCON, but has additional user features that encourage use beyond these specifications. The specific timing of the parallel I/O and clock signals allow efficient interconnect with typical generic controllers and FIFO memories. The built-in self-test and the included 8B/10B encoder functions allow users to implement custom protocols that are suitable to any data-movement application. HOTLink is compatible with all common link interconnect media and interfaces. It is a low-cost, low-power, high-performance tool that enables otherwise impractical system innovation. If there is data to move, HOTLink can carry it.

16. Is HOTLink compatible with ATM?

HOTLink is compatible with the 194.40 Mbaud (155.52 MBit/second), 8B/10B interface defined by the ATM Forum. It offers all of the data, special characters and framing behaviors described in the ATM Forum User-Network Interface (UNI) Specification. In particular HOTLink serves as the physical layer interface for the physical layer for 155 Mbps Interface (and its copper variant). When operating in this capacity, HOTLink runs at 194.40 Mbaud and uses the built-in 8B/10B encoder. All required data and special codes and responses are included in HOTLink.

17. Is HOTLink compatible with SONET?

HOTLink is not directly compatible with SONET for at least the following reasons:

- There are no standard SONET frequencies within its operating range of 160−330 Mbaud.
- HOTLink has a 10-bit unencoded interface, and SONET systems use an 8-bit interface.
- SONET requires a much slower rate-of-change of frequency during loss of signal than HOTLink can achieve.

The HOTLink Receiver can tolerate the long strings of zeros contained in SONET serial streams, and future designs will directly accommodate SONET specifications.

18. What is the latency through a HOTLink Transmitter and Receiver?

The input data is stored in the Transmitter Input register on the rising edge of CKW, so this becomes time-zero. Approximately 21 bit-times (i.e., 21 times the period of CKW ÷ 10) minus the t_{PD} of a TTL output buffer (approximately 10 ns) later, the first bit of that data will emerge from the OUTA±, B±, and C± pins. After the transit time of the serial link, which can be significant, that bit will appear at the receiver. Transit times for typical serial links include the propagation delay of the optical modules (typically 5−10 ns for the pair), if any, and the propagation rate in the link media (i.e., approximately 1 ns/ft in copper, and 2 ns/ft in multi-mode optical cable). Approximately 24 bit-times plus the t_{PD} of a TTL output buffer (approximately 10 ns) after the first data bit is received at the input of the receiver, it appears at the Q_{0-7} outputs. Eight bit-times later CKR rises and the data transfer is complete. The total latency of a HOTLink Tx/Rx pair is approximately link delay plus 45 bit-times.

19. Is there a VERILOG or VHDL model of HOTLink?

Logic Modeling offers full function logic models of both the HOTLink Transmitter (CY7B923) and the HOTLink Receiver (CY7B933). These models perform all of the normal chip functions including BIST, Encoded, and Bypass modes of operation. The models accurately model the "real" parts and have been validated by having them run the actual-chip design-simulation vectors and the outgoing-test vectors. Logic Modeling offers a wide variety of standard product logic models that run on various simulations platforms. They can be reached at:

Logic Modeling
19500 N.W. Gibbs Drive
P.O Box 310
Beaverton, OR 97006
Telephone (503) 690−6900
Fax (503) 690−6906

20. I need to estimate the reliability of HOTLink in my design. How many components does it contain?

Table 3. HOTLink Reliability Data

	CY7B923	CY7B933
Number of components	4285	7988
Number of transistors	3813	6855
Number of gates	2072	2960
Percent digital by gate count	85	90
Percent analog by die area	30	20
Die size	96 x 116 mils	126 x 131 mils

Built on Cypress Standard 0.8-micron BiCMOS. Designed for reliable operation at temperatures −55°C < T_j < 155°C. All pins characterized to withstand ESD >4400V (HBM). Wafer Fab Capability in San Jose, CA; Round Rock, TX.