Introduction

Cypress's entire line of backplane PHYs (the HOTLink® families) incorporates a unique function called multi-byte framing. This feature, which has been included in Cypress backplane transceivers since the original HOTLink family was developed, helps when using any of the HOTLink devices in a non-standards-compliant system where it is a possibility that bits can be flipped during transmission. Multi-byte framing helps the system perform in a much more robust way and avoid misinterpretation of data that causes retransmission of large blocks of data. This paper gives a brief explanation of the purpose and use of this function and how it can help you in your system.

A Little Background

As the earliest synchronous-serial communications systems were developed, the designers faced the problem of "how to recover data that has been serialized and then transmitted over an imperfect transmission channel." The first step involved "encoding" the data to ensure that it had sufficient transition density to allow a Phase-Locked Loop (PLL) to extract the clock from the data stream. This step has been accomplished using various types of scramblers, 4B/5B and later 8B/10B encoder/decoders, as well as other more exotic methods.

The second step involved finding the bit position that represents the start of the data stream (i.e., framing the data). In a serial stream arriving at its destination, one bit looks pretty much like any other, so some "trick" is needed to find the exact bit that was sent "first."

All of the transmission methods in use today include a special string of bits that can be used to discover this bit alignment. In the FDDI-4B/5B code it is called JK, in Fibre-Channel-8B/10B it's called K28.5, and in SONET it's called A1A2. The bit pattern is chosen such that legally transmitted data does not contain this sequence in any bit-alignment except for properly aligned data bits. Thus, the receiver logic can simply look for the "secret code," and when it's found, set internal pointers to match it, and the bit sequence will be "framed" (i.e., output in the same groupings as it had when it was transmitted).

Standards Define the Sync Character

International Standards bodies have adopted and documented the Sync-Characters mentioned above (along with the underlying encoding method) to ensure that systems can interoperate. The published standards include the basic definition of the "minimum" necessary behavior to insure interoperability. They are all silent on the finer points of how this minimum responds to worst case events (or even worse, non-standard uses) in the system environment.

For example, the 8B/10B code tables (the ten-bit codes used to represent eight-bit data with a few control codes left over) used by the Fibre-Channel standard define the sync character to be a code sequence called a *comma*. This seven-bit sequence (1100000 or 0011111) can be used to determine byte boundaries. It occurs in three of the code patterns that are defined as legal out of band signaling characters (**K28.5**, K28.1, and K28.7) only the first one of which is used by Fibre Channel.

In the Fibre Channel standard, the letter of the law states that it is legal to frame on the seven-bit comma character (either 1100000 **or** 0011111). In later standards, this has been refined to allow framing to occur on only one of the bit patterns, (only 0011111 but **not** 1100000).

The Problem

These simplifications are included in the standards because they are written to allow users to claim conformance to the standard with silicon that was built before the standard was finalized.

After a standard is published, this low-level detail is seldom if ever revisited, and is usually used in later standards without comment because nobody wants to challenge a published standard. In this case, the affected silicon traded decreased framing complexity for improved power, without fully comprehending the effects this trade-off would have on noisy-system behavior.

The problem arises when a serial stream nears its limits for distance or speed and starts to stress the receiving data recovery circuits. Jitter and pulse degradation causes the serial stream to be misinterpreted, causing random data bits to be flipped. In general, the system can tolerate this data corruption by detecting erroneous bits and either discarding (retransmit bad data) or fixing them (on-the-fly error correction). However, the effect of a bad bit on the framing logic is that a bit transposition can cause the packet recovery to be misframed (i.e., realigned to the wrong byte boundary). This will corrupt the data stream forever (or at least until a higher level protocol comes down to force reframing).

The simplification of using a single seven-bit comma opens the door for single bit errors to create an alias-Sync character, which will cause the data stream to be misframed. This problem arises because there are many characters (and adjacent character combinations) that include the string 0011111 or 1100000 if an unfortunate bit transposition occurs.

Some Solutions

The problem of generating alias-Sync characters is exacerbated by the fact that there are MANY opportunities to create a seven-bit alias-comma by corrupting otherwise good data bytes (or byte pairs) with an unlucky single-bit transposition. The problem is slightly alleviated by using a more complete ten-bit K28.5 to frame the data; this improvement, however, will not solve the problem because even though the error rate is less than before, there are still a huge number of combinations that can create the alias-Sync. If the system is operating in a noisy enough environment these errors will occur, and if it is very sensitive to framing errors, they will disrupt the system.

The protocol designers for Fibre Channel were aware of this problem, and defined a 65-state FSM (Finite State Machine) to manage the framer logic. This state-machine detects that the recovered data is corrupt, and enables the framer. The deserializer logic then examines the incoming data stream looking for commas, and aligns the internal byte boundaries to them. When good data starts flowing out of the decoder, the FSM disables the framer. This technique limits the misframing exposure to only that time during which the framer is enabled.

Multi-Byte Sync in HOTLink

Since the original HOTLink devices (CY7B923 and CY7B933) were intended to be Fibre-Channel-compliant (i.e., used in systems in which the framer is controlled by the protocol logic)and also operate in the more general non-standard environment (i.e., systems where no logical intelligence had been assumed), a solution was called for that did not require a complex external state-machine.

The problem of bit flipping as a result of noise is what causes the alias-Sync problem. This effect is rare in that it affects a small group of bits at a time, and can go a long way before the error is repeated. This characteristic was used to support the assumption that if an alias-Sync character was decoded (can't tell if its real or not in the decoder), it could be detected if the condition necessary for reframe was expanded to require a second sync before reframing was allowed to happen. As it happens, Fibre Channel interpacket gaps are filled with four-byte ordered-sets, each of which starts with a K28.5. ESCON® interpacket gaps are filled with multiple-K28.5, so requiring two K28.5 is no hardship to the target standards.

The pipelined high-speed logic in the CY7B933 was expanded to record the arrival of a K28.5 and then decide to reframe only on the arrival of another if it arrived exactly 10, 20, 30 or 40 bits later (i.e., exactly byte-aligned with 0–4 intervening bytes). This would allow a system to be implemented without an FSM and ensure that it was not troubled by erroneous framing events. (The cost was the addition of about eight high-speed flip-flops.)

To cover all bases, both a single-K28.5 framing mode and a multi-K28.5 mode were implemented, just in case there was some unforeseen system requirement to force framing on a single K28.5. The selection switch for this feature is internal and time-based. If the framing control pin (RF on the HOTLink, RFEN on the HOTLink DX™) is toggled, and true (framer enabled) for less than an arbitrary 2500 byte times as it would be in systems using the F.C. FSM, then the framer logic will reframe on every single K28.5 found. If RF/RFEN is not toggled, and is held HIGH indefinitely, after 2500 byte times (as it might be in nonstandard systems), it will take two K28.5s, byte-aligned and spaced less than five bytes apart to effect a reframe.

This feature has also been implemented on the newest HOTLink II family of backplane PHYs. Setting RFMODE to LOW (Low-Latency Framer) will cause the device to reframe on every framing character found. If RFMODE is set to MID (Cypress-mode Multi-Byte Framer), the device will reframe only if two framing characters are detected within 50 bits on identical 10-bit character boundaries. If RFMODE is set to HIGH (Alternate-mode Multi-Byte Framer), the device will reframe when at least four consecutive framing characters are detected on identical 10-bit character boundaries.

Conclusion

The HOTLink Multi-Byte Framer is a solution to a real-system problem that is not delineated in any of the standards documents for which it is compliant. This type of better than standard behavior is what makes HOTLink devices an important part of a robust serial transmission system architecture.

Approved AN061 9/25/03 kkv