Serial Extension for 800/1600 Mb/s Computer Interconnect

C. Yen, R. Walker, W. McFarland, D. Cunningham, G. Kwan, T. Hornak

Hewlett-Packard Labs, 1501 Page Mill Road, Palo Alto, CA 94304

ABSTRACT This paper describes the interface units required by Serial-HIPPI, an extender for transmitting data serially between HIPPI nodes at 800 Mb/s, or 1600 Mb/s. HIPPI is an ANSI standard (X3.183-1991) for transmitting data in parallel between data processing equipment.

I. INTRODUCTION

For the past few years, there has been a growing need for Gb/s interconnect between computers. An ANSI Standard X3T9.3 was drafted for 800 Mb/s and 1600 Mb/s channels. This standard, the HIgh Performance Parallel Interface, or HIPPI [1], uses cables of 50 twisted pairs with a maximum length of 25 meters. This is a serious limitation for many applications, which, in some cases, may require connection lengths up to 10 km.

Optical fiber has the potential of transmitting data over much longer distances with very little signal degradation. One could conceivably use 50 fibers to replace the 50 twisted pairs to achieve longer distances. However, signal skew is a serious problem, and becomes unmanageable except for very short distances. A practical solution is to serialize the input signals and carry them in a single high speed fiber. At the receiving end, the serial data is converted back to the parallel form.

II. HIPPI/SERIAL-HIPPI

To meet the need for transmitting HIPPI data serially on fiber or coaxial cable, Serial-HIPPI was created. The Serial-HIPPI Specification was agreed to by numerous computer interconnect users and vendors in May, 1991 [2]. Figure 1 shows the 44 signals defined by the 800 Mb/s HIPPI. The signals can be divided into four groups:

1. Data related signals (36): 32 data and 4 parity
2. Handshake signals (5): Request, Connect, Ready, Packet, and Burst
3. Clock (1): Clock
4 Status signals (2): Interconnect S->D and D->S.

The 41 Signals of the first two groups must be transmitted over the serial link, whereas the 3 signals of the last two groups need not. Two of the 41 signals, Connect and Ready, are handshake signals that transmit in the reverse direction, i.e. from the Destination to the Source. Since serial links are

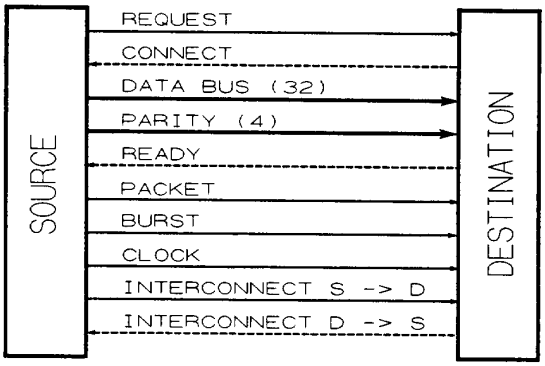

Figure 1. 800 Mb/s HIPPI Signals

usually uni-directional, it is necessary to use a second serial link to carry these signals in the reverse direction. Hence a HIPPI extension between node 1 and node 2 requires two serial links as shown in Figure 2. XMUX and XDEMUX are the HIPPI Source and Destination Adapters, and TLI and RLI are the Transmitter (TX) and Receiver (RX) Link Interface units. Data is transmitted from Node 1 to Node 2 via the upper link, while Connect and Ready are transmitted from Node 2 to Node 1 via the lower link. Note that a duplex HIPPI can be supported by exactly the same configuration with no additional hardware required: Each link carries the reverse handshake signals for the other.

At the Source end, the XMUX is responsible for converting the 41 signals to 20. The TLI converts the 20 signals to the high speed serial data stream. The reverse function is performed by the RLI and the XDEMUX at the Destination end. To facilitate automatic startup at power-on or reset, a Control unit is required at both ends of the serial links as shown in Figure 2. They communicate with the Link Interface units and the HIPPI ports to make sure that the serial links are ready before data transmission is allowed.

III. INTERCONNECTS AND CLOCK

The Interconnect signals do not need to be transmitted over the serial links since they can be derived from the status signals of the Control units. Figure 3 shows the detailed communication between the HIPPI nodes and the Control units. A false Interconnect S->D signal at the Source will, via the Control

92CH3132-8/0000-0981 $3.00 © 1992 IEEE

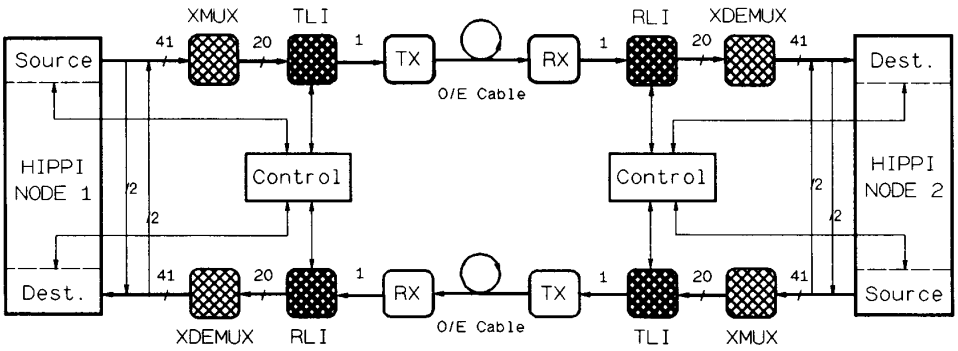

Figure 2. Duplex Serial Link for HIPPI Extension (Simplex or Duplex)
XMUX, XDEMUX: HIPPI Source, Destination Adapter
TLI, RLI: Transmitter(TX), Receiver(RX) Link Interface

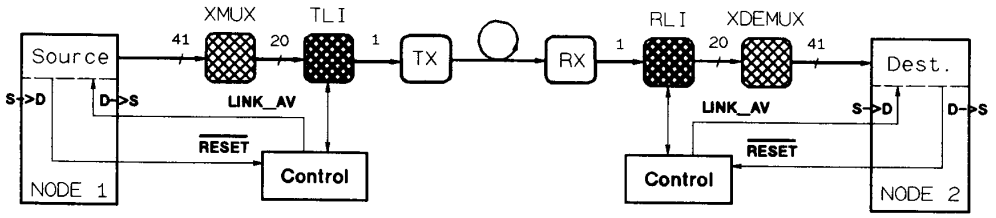

Figure 3. Interconnect Signals Communicated through Link Controls

unit, cause the serial link to reset. In this state, the Control unit holds the Link_Av line low, and hence the Interconnect signals to the HIPPI ports. Similarly, a false Interconnect D->S signal at the Destination is delivered to the Source via the same mechanism.

The Clock signal is not transmitted through the serial link, but the clock information is carried by the serial data stream. On the transmitter side, the 25 MHz clock supplied by the HIPPI Source is used as a reference for the high speed bit clock. On the receiver side, the recovered bit clock is divided down to form the 25 MHz clock for the HIPPI Destination.

IV. HIPPI ADAPTERS, XMUX AND XDEMUX

The primary function of the Source Adapter, XMUX, is to convert the 41 parallel HIPPI output bits at 25 MHz into 2 frames of 20 bits each at 50 MHz. To do this, the three forward handshake signals, Request, Packet, and Burst, are first encoded into two signals F0, F1 as shown in Table 1 [3]. This reduces the 41 signals to 40, which can be easily multiplexed into two frames of 20 bits each as shown in Figure 4. The XMUX also generates a 21st or flag bit, which is transmitted with the 20 bit frame to distinguish between the first and second half of the HIPPI frame.

Table 1. Encoding of Request, Packet, and Burst.

Request	Packet	Burst	F1	F0	State
0	0	0	0	0	Idle
1	0	0	0	1	Request
1	1	0	1	0	Packet
1	1	1	1	1	Burst
0	0	1	0	0	Idle
0	1	0	0	0	Idle
0	1	1	0	0	Idle
1	0	1	0	0	Idle

The Destination Adapter, XDEMUX, reverses the encoding and multiplexing functions of the XMUX, Figure 5. It takes the data and flag bits from the Receiver Link Interface unit, RLI, and converts them to the 41 bit format required by the HIPPI Destination. Because of the asynchronous clocking of the signals transmitted in the reverse direction, there is some chance that a Ready signal pulse may be narrowed. Therefore, a filter is inserted in the Ready signal line whose function is to correct any signal pulse that does not meet the pulse width specification.

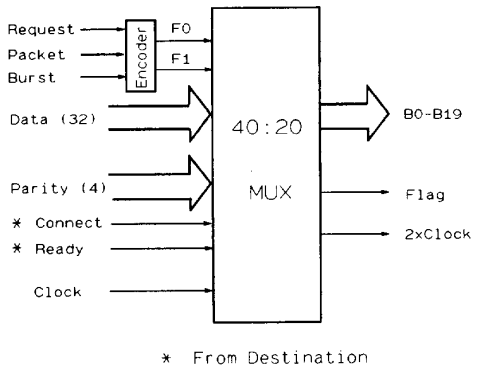

Figure 4. Source Adapter, XMUX, Converting 41 HIPPI Signals to 2 frames of 20 Bits

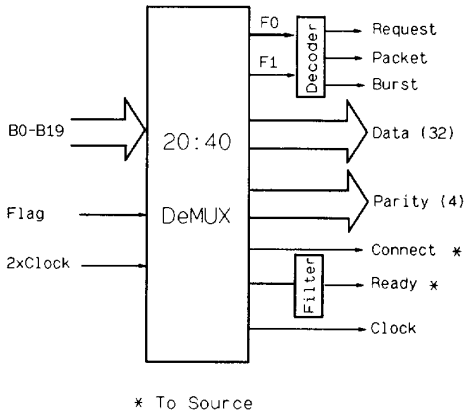

Figure 5. Destination Adapter, XDEMUX, Converting 2 frames of 20 Bits to 41 HIPPI Signals

V. LINK INTERFACE UNITS, TLI AND RLI

The Transmitter and Receiver Link Interface units, TLI and RLI, provide the basic serial data transmission for both the flag bit and the 20 parallel data bits [4]. Two important sets of functions are performed by these two units: Encoding/Decoding, and Multiplexing/Demultiplexing. Encoding is required to guarantee a dc free code and to provide sufficient transitions for clock recovery independent of the data pattern transmitted. Multiplexing is required for parallel to serial conversion. On the receiver side, the serial data stream is demultiplexed and decoded to bring the signals back to their original form.

The Serial-HIPPI encoding scheme, used by the Link Interface units, is simple and effective. As shown in Figure 6, data is sent either inverted or non-inverted by comparing (COMP) the accumulated disparity (ACCUM) of the data already sent and the disparity of the incoming word (SIGN) to be sent. If the incoming word in its original form can reduce the accumulated disparity, it is sent non-inverted. Otherwise, it is sent inverted. Table 2 shows the 24 bit Data Frame structure obtained by

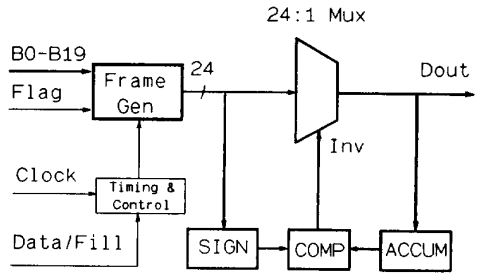

Figure 6. Transmitter Link Interface, TLI, and the HP 21B/24B Encoding Scheme

appending four coding bits to the data word B0-B19. The four coding bits indicate whether data has been inverted or not, and whether the flag bit is a "1" or a "0". They also indicate two kinds of non-data frames: the Fill Frame are used for link startup, and the Reserved Frame intended for user-defined applications (Reserved Frame is not utilized by Serial-HIPPI). The four coding bits are structured so that the middle two bits are always complementary to each other, providing a fixed transition for clock synchronization. In a frame of 24 bits, there are 21 user controllable data bits. This coding scheme is called the HP 21B/24B code. Its coding efficiency is slightly higher than either the 4B/5B (FDDI) or the 8B/10B (Fiber Channel) code. An important benefit of this code is that it provides automatic frame synchronization: no special framing characters are required.

Table 2. 24 Bit Data Frame Structure.

Data Status	Flag bit	Data Field	Appended Bits
True	0	$B_0 - B_{19}$	1 1 0 1
Inverted	0	$\overline{B_0 - B_{19}}$	0 0 1 0
True	1	$B_0 - B_{19}$	1 0 1 1
Inverted	1	$\overline{B_0 - B_{19}}$	0 1 0 0

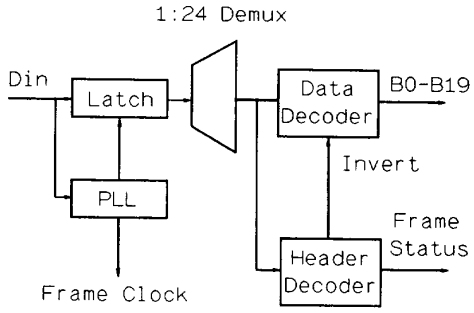

Figure 7. Receiver Link Interface, RLI

Decoding is extremely simple. The coding bits indicate whether the data is inverted. If it is, The RLI inverts the data back to its original form. Figure 7 shows a simplified block diagram of the Receiver Link Interface unit.

Signals from HIPPI are interleaved in a special way [5] to form the two 20-bit data fields, Table 3. This bit arrangement helps to maintain the error detecting capabilities of the parity bits for two potential sources of error.

1. Noise hit of adjacent bits, and
2. Inversion of the 20 bit data field due to an error in the coding bits.

Table 3. Interleaving of HIPPI Signals

Frame = 0; Flag = 0

00	08	16	24	01	09	17	25	02	10
18	26	03	11	F0	19	27	P0	P1	F1

Frame = 1; Flag = 1

P2	P3	04	12	20	28	05	13	21	29
06	14	22	30	RY	07	15	23	31	CN

00-31: HIPPI Data Bits P0-P3: Parity Bits

F0,F1: Source State CN,RY: Destination State

00	=	Idle	CN	=	Connect
01	=	Request	RY	=	Ready
10	=	Packet			
11	=	Burst			

VI. CONTROL

The Control unit oversees the startup of the serial link at power-on or reset. It communicates with the TLI and the RLI to determine which special Fill frames are being received and which should be transmitted. From this information, it facilitates an orderly transition of states until the link is ready for data transmission.

Figure 8 shows the three states of the Control unit. In state 0, the RLI is frequency locking to the incoming signal, and the TLI is instructed to send a special Fill frame, called FF0. When the RLI has achieved lock, the Control unit progresses to state 1. In this state, the Control unit waits for the other end to lock also, and the TLI indicates to the other side that it is ready by sending a second Fill frame, called FF1. Upon receipt of either FF1, or data, the Control unit progresses to the state 2. In this state, the TLI sends data frames, and Link_Av is true. The HIPPI ports are then signalled via the Interconnect signals that data transmission can begin.

VII. OPTICAL/ELECTRICAL LINKS

For the 800 Mb/s HIPPI, The baud rate of the serial link is 1.2 Gbaud/s, including the handshake signals and the encoding overhead. For short distances, coaxial cable can be used to transmit the serial stream. With a simple equalization circuit, the distance of an RG-8/U cable can be increased to about 36 meters. For longer distances, fiber is the medium of choice, and lasers are the only optical sources usable at gigabit rates and higher. Measurements show that the low bit-error rates specified by Serial-HIPPI ($<10^{-12}$) can be achieved with commercially available lasers. Work is ongoing at many companies to produce low cost, reliable optical sources appropriate for gigabit rate links.

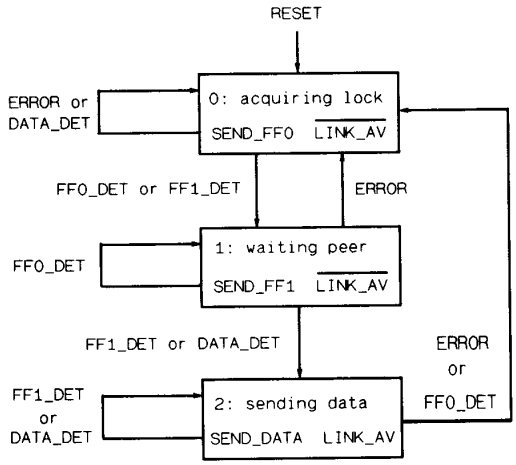

Figure 8. Three States of Link Control

VIII. 1600 MB/S SERIAL-HIPPI

To support the 1600 Mb/s HIPPI standard, which transmits data, in duplex, on 4 cables of 50 twisted pairs, two parallel 800 Mb/s Serial-HIPPI links are required for each direction. In this case, one must pay attention to the signal skews between the two parallel paths [2]. A better solution will be to use higher speed Link Interface units operating at 2.4 Gb/s when they become available.

IX. REFERENCES:

[1] "High-Performance Parallel Interface - Mechanical, Electrical, and Signalling Protocol Specification (HIPPI-PH)", American National Standard (ANSI) X3.183-1991.

[2] "Serial-HIPPI Specification", Revision 1.0, Serial-HIPPI Implementors Group, May 17, 1991.

[3] First presented to Serial-HIPPI Implementors Group by K. Annamalai, Dec. 4, 1991. Similar coding suggested by Eric Fromm, private communication, Oct. 9, 1990.

[4] "A 1.5 Gb/s Link Interface Chipset for Computer Data Transmission", Richard C. Walker, Thomas Hornak, Chu-Sun Yen, Joey Doernberg, Kent Springer, IEEE Journal on Selected Areas in Communication, Vol. 9, No. 5, June 1991, pp.698-703.

[5] Similar schemes presented to Serial-HIPPI Implementors Group by K. Annamalai, Dec. 4, 1991, also suggested by Eric Fromm, private communication, Oct. 9, 1990.