# HP's Link Interface Chipset for Serial-HIPPI

W. McFarland, R. Walker, C. Stout, J. Wu, B. Lai, G. Kwan, C. Yen

Hewlett-Packard

## Abstract

*A link interface chipset that conforms to the Serial-HIPPI specification is presented. The chipset contains all portions of the Link Interface and Link Control functions specified in Serial-HIPPI. The simple additional circuitry required to create a complete Serial-HIPPI link using this chipset is detailed. The two-chip set can also serve as a general purpose link with no additional circuitry. It transfers parallel data across a serial channel at rates from 960Mb/s to 1440Mb/s. It provides automatic link startup, continuous frame synchronization, and requires no adjustments or off-package circuitry.*

## 1: Introduction

Many areas of computer and data communications could benefit from inexpensive, compact, easy-to-use gigabit-rate data links. High speed serial links can reduce board space and wiring requirements across short spans between and within computers and peripherals. With the use of fiber optics, high speed serial data can be transmitted much further than is possible on metallic cables. In response to these needs, the Serial-HIPPI specification[1] was agreed upon by over 40 vendors and users. The two chip set presented in this paper meets these requirements. It is an easy to use generic chipset that will be produced in high volumes at low cost, and can form a complete Serial-HIPPI system with the addition of some simple circuitry.

Serial-HIPPI specifies three major functional units: HIPPI Interface, Link Interface, and Link Control. The chipset presented in this paper (referred to as the "G-Link") implements the Link Interface and Link Control functional units. Figure 1 shows a diagram of a complete Serial-HIPPI system using the G-Link. Serial-HIPPI transmits data from a HIPPI-PH[2] source port to a HIPPI-PH destination port by encoding and serializing the data, transmitting it over fiber or coaxial cable, and recovering the data at the other end of the link. The on chip Link Control Unit is a state machine that controls the startup and operation of the complete link. The HIPPI Interface Unit is composed of an XMUX at the source and an XDEMUX at the destination. These circuits take the HIPPI-PH parallel data words and convert them into two 20 bit data fields. The XMUX and XDEMUX are external circuits that customize the G-Link to the Serial-HIPPI application.

This paper is divided into three major parts. The first part describes the internal operation of the G-Link chipset. The second part shows details of the design of the HIPPI Interface XMUX and XDEMUX. The last part describes features incorporated into the chipset to make it easier to use.

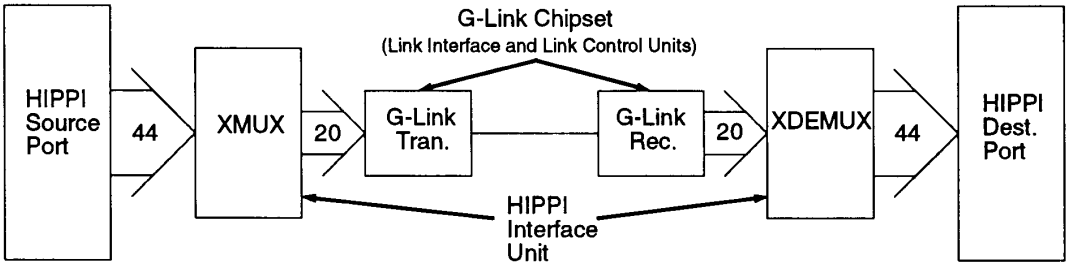



**Figure 1. Serial-HIPPI system**



0-8186-2655-0/92 $3.00 © 1992 IEEE

## 2: G-Link chipset

The G-Link chipset is composed of a transmitter chip and a receiver chip. The transmitter generates a high speed clock, encodes parallel data, then serializes the data so it can be transmitted on fiber or coaxial cable. At the receiver, the clock is extracted from the incoming serial data. The data is then deserialized, decoded, and passed out as parallel data.

### 2.1: CIMT code

The chipset uses a Conditional frame Invert, Master Transition code (CIMT)[3]. This code insures DC balance in the serial bit stream. This allows transmission across AC coupled fiber or coaxial cable systems. The code also provides a transition in a fixed location of each data frame. This "master transition" allows for a simple, adjustment free, fully integrated clock recovery system. In addition, the master transition is used to provide framing information. Since every transmitted frame contains framing information, it is not necessary to send special framing characters. The code also supports $2^{18}$ "reserved" and Fill Frames (FF0 and FF1). These frames are distinct from normal data frames and are used for link startup, or by the user for any other function where non-data signalling is desired. A special "flag bit" is included in the code. This bit can be used as an independent data bit, or can be automatically alternated in the transmitter. In Serial-HIPPI applications, the flag bit is used to distinguish the first 20 bit data field of a HIPPI-PH word from the second. Since it is continuously alternating, the flag bit is used as an additional check for proper link operation.

### 2.2: Transmitter

A simplified block diagram of the transmitter chip is shown in Figure 2. The PLL/Clock Generator generates the

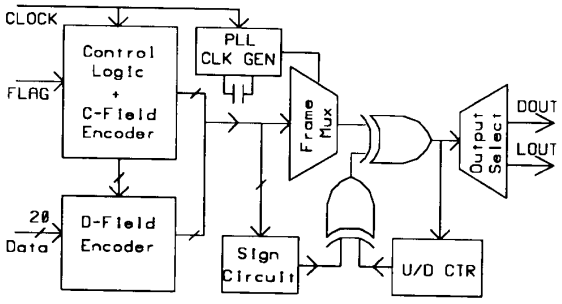
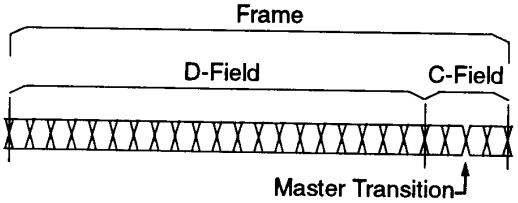



**Figure 2. Transmitter**

high-speed serial clock from the low frequency input clock. The low frequency clock may be at the frame rate, or one-half the frame rate, which is very convenient for Serial-HIPPI. The conditional frame invert code is generated by transmitting data fields in true or complement form, as needed, to maintain DC balance on the line. To make the decision, the transmitter uses a sign circuit to compute the net balance of the incoming word and compares it to the sign of an up/down counter which keeps track of the overall disparity of all transmitted bits. If the two signs agree, the word is sent inverted. Otherwise it is sent non-inverted. Four extra coding bits (the C-Field) are appended to the data field (D-Field). The central pair of bits in the C-Field are always complementary and provide the "master transition". The polarity of this master transition is used to encode the extra FLAG bit. The other two bits in the C-Field are used to signal whether a given frame is representing inverted data, non-inverted data, or reserved words. Figure 3 shows the resulting code.




| C-Field | D-Field | Flag bit |
|---|---|---|
| 1 1 0 1 | true | 0 |
| 0 0 1 0 | inverted | 0 |
| 1 0 1 1 | true | 1 |
| 0 1 0 0 | inverted | 1 |

**Figure 3. Coded frame structure**

### 2.3: Receiver

A simplified block diagram of the receiver is shown in Figure 4. The data path consists of an input selector, two input sampling latches, a demultiplexer, a control field decoder and a data field decoder. The input selector allows either normal data input (DIN), equalized input (EQIN), or a loop-back data input (LIN) to be chosen. Local loop-back is useful for system debugging.

Clock recovery in the G-Link receiver is accomplished with a phase locked loop (PLL) driving an on-chip VCO [4]. The incoming data stream is latched on both the rising and falling edges of the bit-rate VCO clock. When the loop is locked, the latch driven by the non-inverted VCO clock

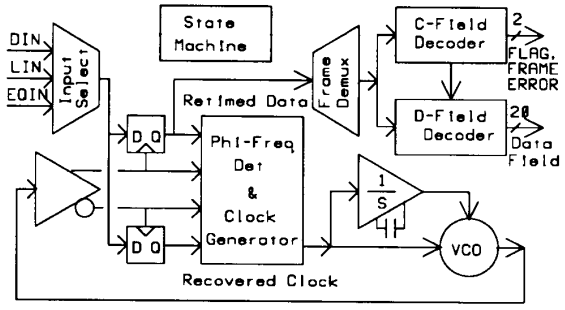



**Figure 4. Receiver**

samples the center of each data bit and produces retimed data. The latch driven by the inverted VCO clock samples the transitions between bits. For example, a rising transition sampled too early results in a low output. A rising transition sampled too late results in a high output. The sample corresponding to the master transition is corrected for transition polarity by being XORed with the immediately preceding data bit to derive a binary-quantized (bang-bang) phase error indication. This phase error indication is used to drive the VCO. Because the two latches are matched, the VCO clock phase is inherently aligned within the bit cell over process, temperature, and frequency variations.

The circuitry operates from 600Mb/s to 1500Mb/s. All of the circuitry is on chip, except for one capacitor in the PLL feedback loop. This capacitor is placed within the G-Link package.

### 2.4: Technology

The two chips were implemented in a 3-level metal, 25GHz ft, Silicon Bipolar process [5] using full-custom differential 4.5V ECL design. The 1.8W transmitter and 2.0W receiver are both 3.5x3.5mm in size and utilize ~6500 devices. The chips are housed in custom 68-pin surface-mount packages that include de-coupling and PLL integrating capacitors.

### 2.5: Experimental results

The chipset is fully functional and has been tested at speed. With a low-jitter input, the receiver recovered clock jitter is below 19ps rms. The PLL acquisition time is under 2ms with a full scale (600Mb/s to 1500Mb/s) data rate step at the input.

## 3: HIPPI interface

The G-Link chipset can be customized to the Serial-HIPPI application by adding external circuitry. This circuitry corresponds to the HIPPI Interface XMUX and XDEMUX as defined in the Serial-HIPPI specification. The major function of these circuits is to adapt the 44 parallel HIPPI-PH signals to the G-Link's 20-bit data fields.

In general, data flows from a HIPPI-PH Source to a HIPPI-PH Destination. However, the CONNECT and READY signals flow in reverse, from Destination to Source. It is not convenient to transmit signals in both directions on a single fiber. Therefore, these reverse direction signals are routed by the XMUX to a neighboring link that is transmitting in the correct direction, and rerouted by the XDEMUX at the far end.

### 3.1: XMUX

Figure 5 shows a diagram of the XMUX. The signals

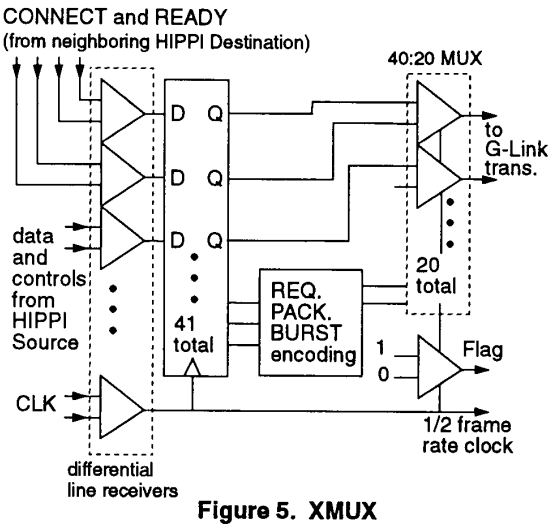



**Figure 5. XMUX**

from the HIPPI-PH Source port is captured by differential ECL receivers followed by a latch. The latch is clocked by the HIPPI-PH Source clock. Because of the structure of the HIPPI-PH REQUEST, PACKET, and BURST signals, only four combinations are legal. Therefore, these three lines are encoded into two lines according to the Serial-HIPPI standard with a few simple logic gates. The HIPPI-PH signals are then multiplexed 40:20 by the HIPPI-PH Source clock. Since the G-Link chipset can generate its high speed clock from a 1/2 frame rate clock, and the XMUX can do all its functions with the HIPPI-PH Source clock, the frame rate clock is not used in the system.

The XMUX generates a flag bit with an additional 2:1 multiplexer that has its inputs fixed to a "1" and a "0". This bit distinguishes the first 20 bit data field of a HIPPI-PH word from the second 20 bit data field of a HIPPI-PH word.

## 3.2: XDEMUX

Figure 6 shows a diagram of the XDEMUX. The 20 bit

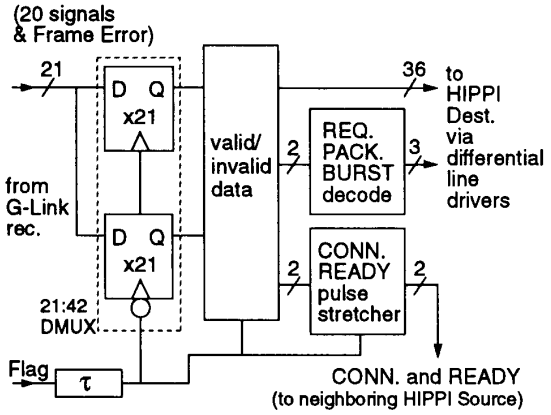



**Figure 6. XDEMUX**

data fields, along with the Frame Error signal from the G-Link receiver, are first demultiplexed into 42 bit words. This is done by delaying the flag bit, and using it to clock two sets of 21 bit registers on opposite edges.

If the data is not marked with a Frame Error indication from the receiver, it is considered valid, and is passed unaltered through the valid/invalid data circuit. If the receiver detected a frame error when the data was received, the data is considered invalid. On invalid data, a parity error is forced into the data to insure that the HIPPI-PH port will not use the data. This is done by setting all the data bits and parity bits to "0"s. Because HIPPI-PH uses odd parity, all four bytes will appear to have parity errors. This is implemented with a rank of 2:1 multiplexers that switch from data to all "0"s when Frame Error is true. In addition, all five HIPPI-PH control signals (REQUEST, PACKET, BURST, CONNECT, and READY) are held to the values they had on the last valid data frame. This is done with enable flip-flops that are disabled when the Frame Error signal is true.

Because the CONNECT and READY signals travel across a neighboring link, they are clocked and transmitted with arbitrary phase relative to the transmit clock. These signals are nominally at least four clock cycles long, but they may be narrowed to three clock cycles when transmitted. The pulse stretcher circuitry restores these signals to at least four clock cycles. The pulse stretcher is designed to ignore single bit errors that might occur on the CONNECT or READY signals. Figure 7 shows the circuit (CONNECT is handled in a similar way). A shift register is monitored

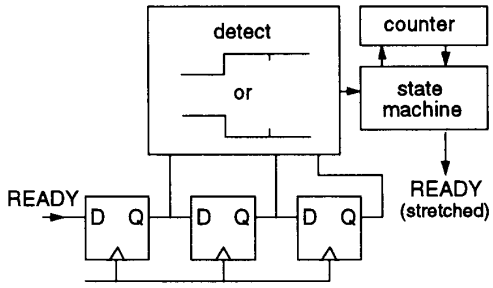



**Figure 7. Pulse stretcher**

for two specific bit patterns. A "0" followed by two "1"s indicates a valid positive transition, a "1" followed by two "0"s indicates a valid negative transition. This information is relayed to a very simple state machine which loads a counter. The counter and state machine act to insure that there are at least four clock cycles between transitions.

The XMUX and XDEMUX are presently implemented in discrete TTL logic. A complete Serial-HIPPI XMUX/transmitter, XDEMUX/receiver pair fits on a single 8x14" PC Board. The system conforms to the Serial-HIPPI specification, and forms a fully HIPPI-PH compliant serial extender.

## 4: Features for ease-of-use

A state machine is included on the G-Link receiver which transparently handles the link startup protocol. Figure 8 shows the state diagram of the fully handshaked startup sequence. G-link chipsets are assumed to be arranged in a duplex pair. FF0 and FF1 are special non-data signals. At a given node, the transmitter sends one of these signals, while the receiver detects one of these signals from the other end of the link. The end to end handshake ensures that both ends of the link are operating properly before any data is transmitted. This start-up procedure is extremely robust. It requires no time-outs, and makes no assumptions about time delays or status of the other link. Both control units act as peers. The state machine conforms to the Serial-HIPPI specification, and can be modified or bypassed externally.

An equalizing amplifier on the receiver can be selected by the user if desired. This equalizer implements a shelf filter providing a 3dB boost at 600MHz to compensate for

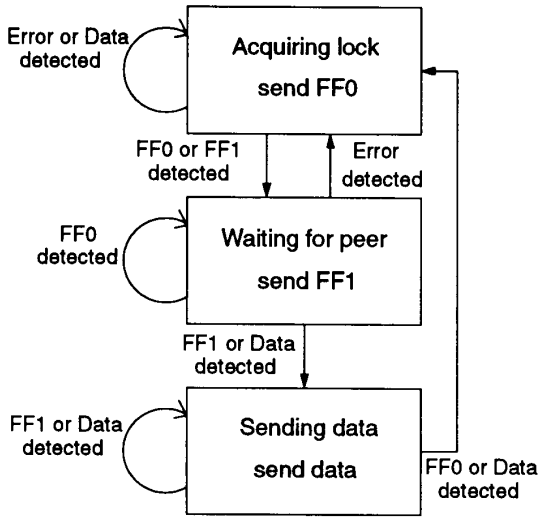



**Figure 8. Startup state diagram**

skin-loss in long coaxial lines. When selected, the maximum distance data can be transmitted over various metallic cables is increased by over 50%. Table 1 shows maximum range on various cables with and without equalization.

**Table 1: Maximum cable lengths**

| Coaxial cable type | Maximum length | |
|---|---|---|
| | Un-equalized | Equalized |
| RG 174U | 5 Meters | 9 Meters |
| RG 58A/U | 10 Meters | 18 Meters |
| RG 8/U | 20 Meters | 36 Meters |

The G-Link chipset includes several features to increase its robustness. The receiver automatically detects and flags code violations, or failures in flag bit alternation if that feature is selected. If code violations are persistent, the state machine can automatically restart the links.

The G-Link chipset provides a very flexible parallel interface. Using the Flag bit for data transmission, data words of width 16, 17, 20, or 21 can be handled directly.

These ease-of-use features allow the G-Link to support a wide variety of applications. Because it handles its own link startup and framing, it can be used in almost any application as a "virtual ribbon cable". Its code has been accepted as the IEEE P-1596 Scalable Coherent Interface standard. The chipset easily implements the Serial-HIPPI specification. The wide variety of applications provide high volume and thereby low cost to the user.

## 5: Conclusion

A chipset conforming to the Serial-HIPPI specification has been developed. When combined with simple off chip circuitry, the chipset implements a complete Serial-HIPPI system. The chipset has several features that make it easy to use for Serial-HIPPI and other applications. These features include: a variable width parallel interface, automatic link startup, continuous framing, transmit clock generation, and on-chip cable equalization. The compact chipset's flexibility and ease-of-use make it attractive for a wide variety of applications.

## 6: Acknowledgments

The authors thank D. Pettengill, D. Yoo, S. Chiang, and the staff at HP's Circuit Technology Research and Development group.

## 7: References

[1] "Serial-HIPPI Specification, Revision 1.0, Serial HIPPI Implementors Group." Available via anonymous ftp as file pub/hippi/serial_hippi_1.0.ps on hplsci.hpl.hp.com (15.255.176.57) or by contacting Chu Yen at 415-857-4689.

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

[3] U.S. Patent #4,926,447, "Phase Locked Loop for Clock Extraction." U.S. Patent #5,022,051, "DC-Free Line Code for Arbitrary Data Transmission."

[4] R.C. Walker, et. al., "A 1.5 Gb/s Link Interface Chipset for Computer Data Transmission," IEEE Journal on Selected Areas in Communications, vol. 9, June 1991, pp. 698-703.

[5] W. M. Huang et al., "A high-speed bipolar technology featuring self-aligned single-poly base and submicrometer emitter contacts," IEEE Electron Device Letters, vol. 11, pp. 412-414, 1990.