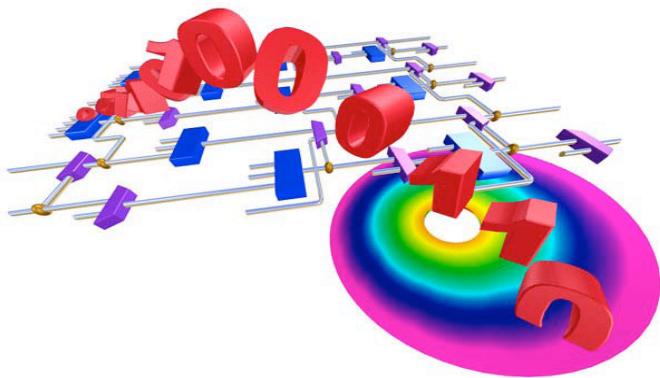




High Speed Serial IOs



Basics

Peter Thorwartl

01_so_hssio_basics.odp Date Oct 23, 2009 Page 1

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Why do we need HSSIO ?

Speed 1 Gbits/s and 11 Gbits/s
Reduced pin count, 2wires for each direction
=> 4 wires for one link needed
Simultaneous Switching Outputs SSO
Less EMI
Less Costs
Predefined Protocols
SATA
PClexpress 1x, 4x, 8x, 16x, 32 x Lanes, Gen1, Gen2, Gen3
Gigabit Ethernet 1 GE , 10 GE, 100 GE
Serial Rapid IO
Infiniband 1x, 4x, 12x
Hypertransport



NOTES

What is the chief advantage of gigabit serial I/O?

Speed

For getting data on and off of chips, boards, or boxes, nothing beats a high-speed serial link. With wire speeds from 1 to 12 Gb/s and payloads from 0.8 to 10Gb, that is a lot of data transfer. And with fewer pins, no massive simultaneous switching output (SSO) problems, lower EMI, and lower cost, high-speed serial is the clear choice. Multigigabit transceivers (MGTs) are the way to go when we need to move lots of data fast. Let's examine some of the advantages of gigabit serial I/O.

Pin Count

Pin count is the first problem encountered when trying to move a lot of data in and out of a chip or a board. The number of input and output pins is always limited. Although pin count tends to increase over time, it is never enough to keep up.

To be fair, there are some pin issues for which we are not accounting. For example, some MGTs need more power and ground pins than a pair of slower pins. And a parallel interface may require special reference pins.

Simultaneous Switching Outputs

A designer should consider SSO when using single-ended parallel buses. However, some of those outputs are going to toggle at the same time. When too many switch simultaneously, ground bounce creates a lot of noise. A designer could also employ differential signal processing on all I/O to get rid of the SSO problems, but that doubles the pin count. And if the data flow needs are more modest, the designer could use a parallel interface with a usable pin count.



Where Will HSSIO Be Used?

Chip to Chip

Pin Count: smaller cheaper packages, fewer layers on PCB

Power

Board To Board / Backplanes

Higher Bandwidth

Reduced Pin Count

Point to Point Connection

Box to Box

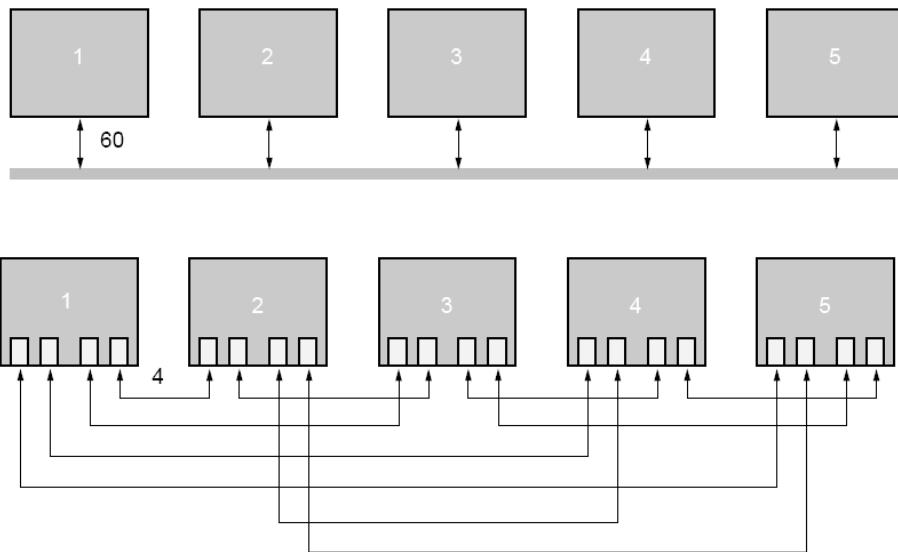


NOTES

SERDES was initially used to talk box-to-box. But it exploded into the marketplace because of how nicely it handles chip-to-chip communication on the same circuit board. Chip-to-chip communication had previously been almost exclusively a parallel domain. The amount of logic needed to serialize and deserialize far outweighed any savings that come from pin count reduction. But with deep sub-micron geometry, an incredible amount of logic can be achieved in a very small amount of silicon. SERDES can be included on parts for a very low silicon cost. Add to that the ever increasing need for I/O bandwidth, and SERDES quickly becomes the logical choice for moving any significant amount of data chip-to-chip.



Parallel vs Serial



NOTES

Although they were once the best available, parallel architectures are at their limits. Most parallel bus protocols have evolved to the point where adding data bits is physically impractical because of pin counts on connectors. Clock skew, data skew, rise and fall times, and jitter limit the ability to increase clock frequency. Doubling the data rate can help, but it often requires moving to differential signaling, and that drastically increases pin count. Also, controlling the cross-talk issues on parallel buses is difficult.

New serial backplanes are somewhat different than parallel backplanes. They typically have dedicated serial links from each node to every other node. Figure 2-4 illustrates the basic architecture of an old parallel bus and a new serial bus.

Serial bus architectures have a lot to offer. The pin count of a serial bus is a function of the number of nodes. For most practical node numbers, a serial architecture has fewer pins than the old parallel architectures.



The Future of HSSIO

At first glance, multi-gigabit communication seems to impose unacceptable restrictions.

Serial designers must contend with signal integrity, smaller time bases, and possibly the need for extra gates and additional CPU cycles.

However, multi-gigabit advantages in box-to-box and chip-to-chip communication far outweigh the perceived shortcomings.

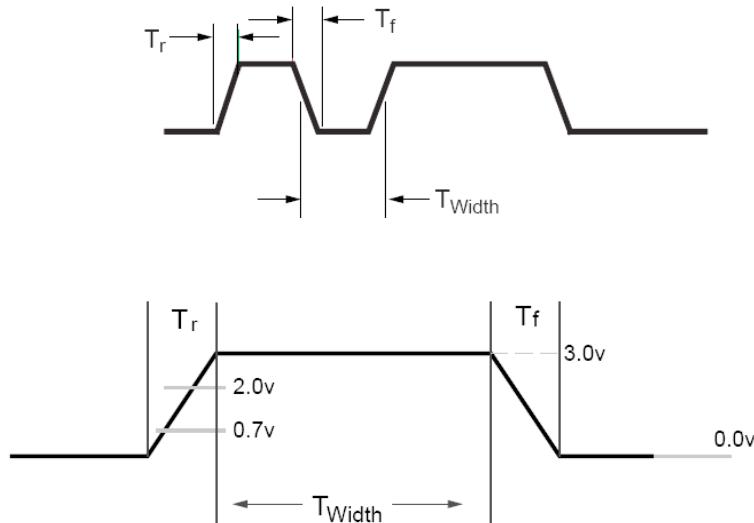
For example, high speed, fewer pins, lower EMI, and lower cost make it the ideal choice in many communication designs. These advantages will ensure its continued use in communication applications far into the future.



NOTES



Typical Digital Signal



01_so_hssio_basics.odp Date Oct 23, 2009 Page 6

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

I/O Performance Limitations

Input/output (I/O) has always played a crucial role in computer and industrial applications. But as signal processing became more sophisticated, problems arose that prevented reliable I/O communication.

In early parallel I/O buses, interface alignment problems prevented effective communication with outside devices. And as higher speeds became prevalent in digital design, managing signal delays became problematic.

Digital Design Solutions for I/O

Digital designers turned to a host of methods to increase signal speed and eliminate I/O problems. For example, differential signal processing was employed to increase speed in chip-to-chip communications.

And design methods such as signal-, source-, and self-synchronization refined inter-IC (integrated circuit) communication to provide reliable I/O at speeds demanded by the computer industry

Notice the values of the time measurements listed on the diagram:

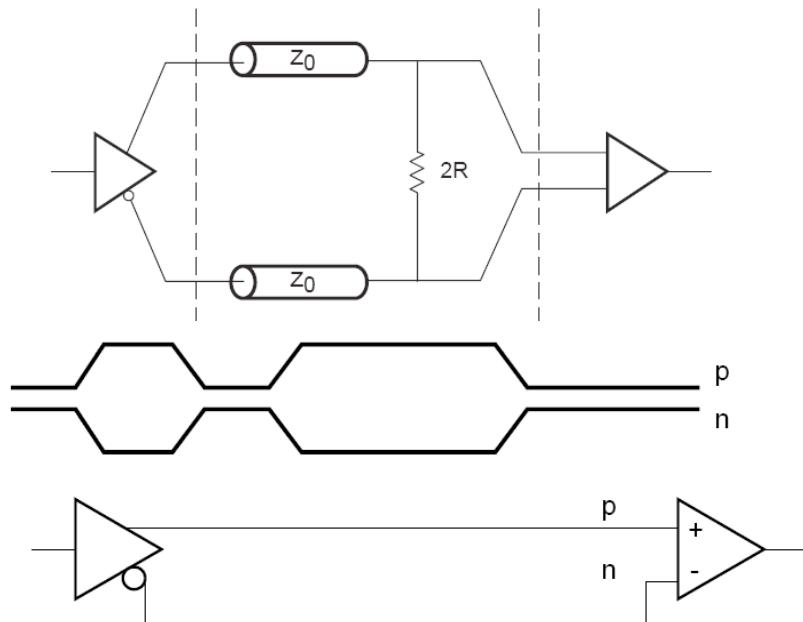
TR = 20 ps

TF = 20 ps

TWIDTH = 0.10 ns



Differential Signals



01_so_hssio_basics.odp Date Oct 23, 2009 Page 7

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



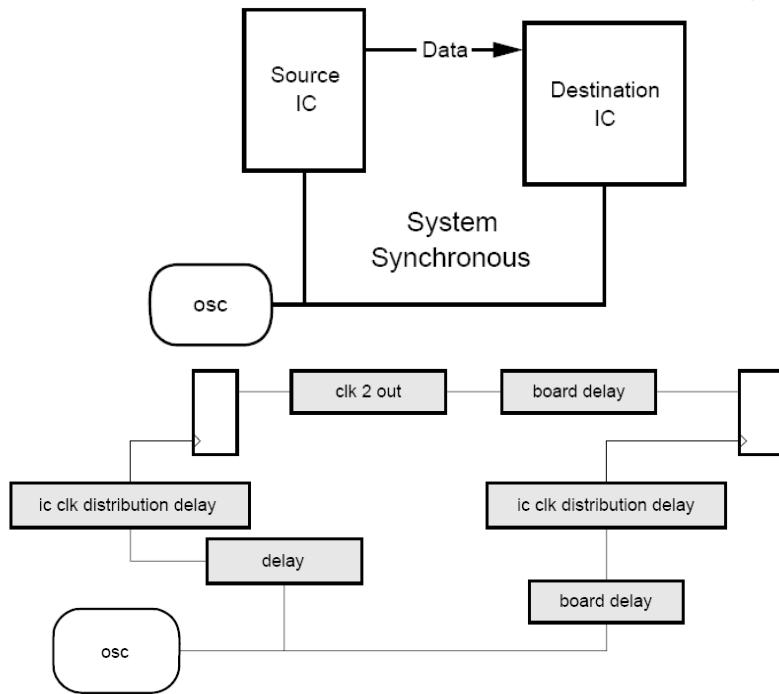
NOTES

About the time HSTL and other low voltage swings became popular, a differential signal method began to appear on chip-to-chip communications. Differential signals had long been available, but they had been used for long transmissions, not for chip-to-chip communication on PCBs .

As IC communication speeds increased, system and IC designers began to look for signaling methods that could handle higher speed . Differential signaling was such a method. It has several advantages over single-ended signaling. For example, it is much less susceptible to noise. It helps to maintain a constant current flow into the driving IC. And rather than comparing a voltage to a set value or reference voltage, it compares two signals to each other. Thus, if the signal referenced as the positive node has a higher voltage than the one referenced negative, the signal is high, or one. If the negative referenced signal is more positive, the signal is low, or zero. The positive and negative pins are driven with exact complementary signals as shown below.



System Synchronous



01_so_hssio_basics.odp Date Oct 23, 2009 Page 8

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

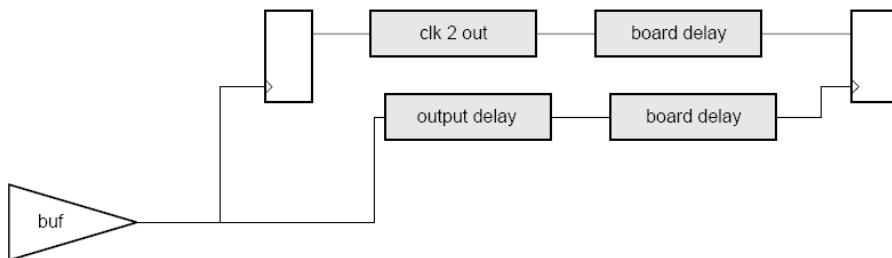
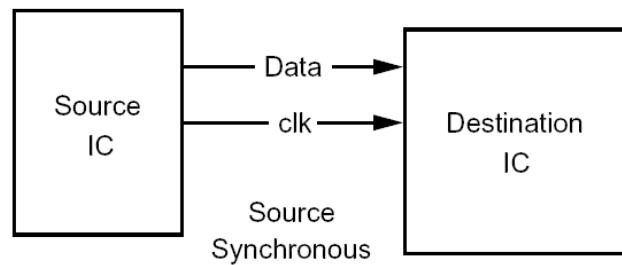


NOTES

This method as shown in figure was the most common for many years. It seems very simple until we look at the timing model. The shaded boxes represent delays that must be accounted for and balanced to ensure a reliable receiving circuit. **System-Synchronous:** Communication between two ICs where a common clock is applied to both ICs and is used for data transmission and reception
System-Synchronous: Communication between two ICs where a common clock is applied to both ICs and is used for data transmission and reception.



Source Synchronous



NOTES

For years most signal delays were ignored because they were so small compared to the available time. But as speeds increased, managing delays became more difficult, then impossible. One way to improve the problem was to send a copy of the clock along with the data. This method is called source-synchronous and it greatly simplified the timing parameters.

The output time of the forwarded clock is adjusted so that the clock transitions in the middle of the data cell. Then the trace lengths of the data and clock lines must be matched. But there are some drawbacks. The received data on the destination IC must be moved from the received clock domain to a global IC clock.

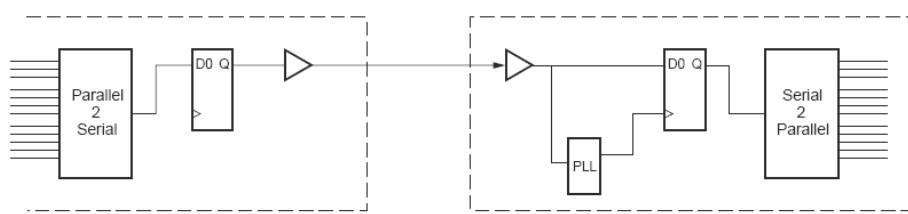
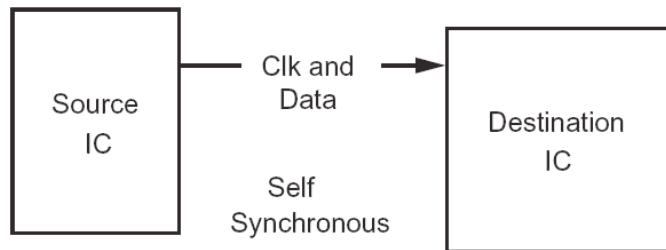
Source-synchronous design results in a marked increase in the number of clock domains. This introduces timing constraint and analysis complications for devices such as a Field Programmable Gate Array (FPGA) with limited clock buffers, and an Application-Specific Integrated Circuit (ASIC) where each clock tree must be custom designed. The problem is aggravated on large parallel buses where board design limitations often force the use of more than one forwarded clock per data bus. Hence, a 32-bit bus may require four, or even eight forwarded clocks

Source-Synchronous: Communication between two ICs where the transmitting IC generates a clock that accompanies the data. The receiving IC uses this forwarded clock for data reception.

Clock Forwarded: Another term for source-synchronous.



Self Synchronous



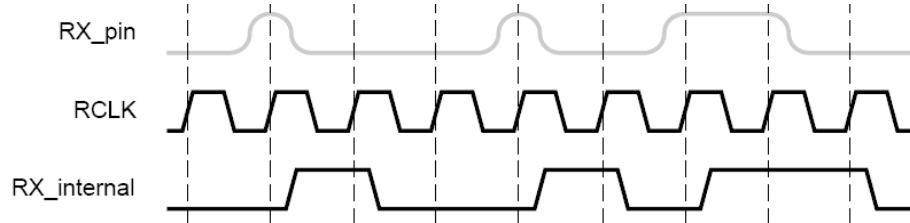
NOTES

Here, the data stream contains both the data and the clock.

Self-Synchronous: Communication between two ICs where the transmitting IC generates a stream that contains both the data and the clock.



Clock Data Recovery Unit



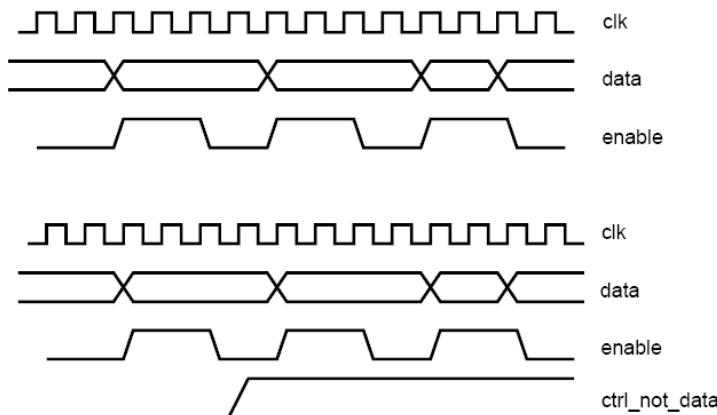
NOTES

The clock recovery process (Figure 1-15) does not provide a common clock or send the clock with the data. Instead, a phased locked loop (PLL) is used to synthesizes a clock that matches the frequency of the clock that generates the incoming serial data stream.

PLL: A phased locked loop is a circuit that takes a reference clock and an incoming signal and creates a new clock that is locked to the incoming signal.



Parallel Data Transfer

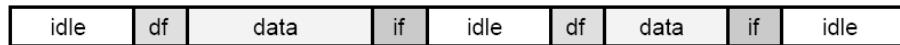


NOTES

In parallel transfers, additional control lines are often used to give different meanings to the data. Examples include data enables and multiplexing both data and control data onto the same bus.



Serial Data Transfer



Data Flag marks start of data



Control Flag marks start of control



Idle Flag marks end of valid data and control

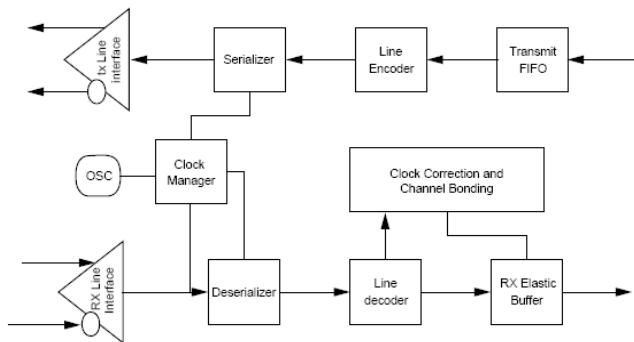


NOTES

In the serial domain, flags or markers are created to set data apart from non-data that is normally referred to as idle. Flags can also be used to mark different types of information such as data and control



SERDES



NOTES

Serializer: Takes n bits of parallel data changing at rate y and transforms them into a serial stream at a rate of n times y.

Deserializer: Takes serial stream at a rate of n times y and changes it into parallel data of width n changing at rate y.

Rx (Receive) Align: Aligns the incoming data into the proper word boundaries. Several different mechanisms can be used from automatic detection and alignment of a special reserved bit sequence (often called a comma) to user-controlled bit slips. Clock Manager: Manages various clocking needs including clock multiplication, clock division, and clock recovery.

Transmit FIFO (First In First Out): Allows for storing of incoming data before transmission.

Receive FIFO: Allows for storing of received data before removal; is essential in a system where clock correction is required.

Receive Line Interface: Analog receive circuitry includes differential receiver and may include active or passive equalization.

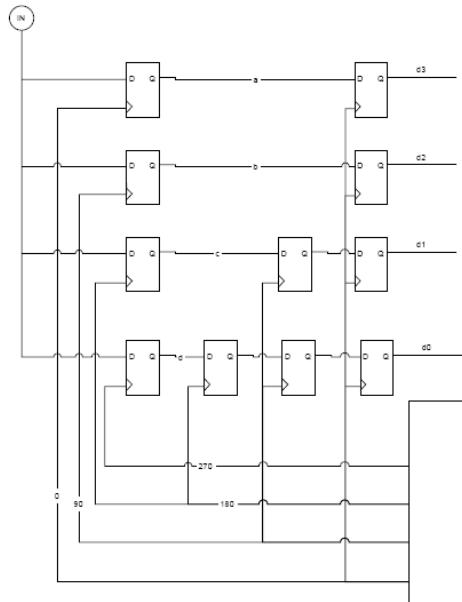
Transmit Line Interface: Analog transmission circuit often allows varying drive strengths. It may also allow for pre-emphasis of transitions.

Line Encoder: Encodes the data into a more line-friendly format. This usually involves eliminating long sequences of non-changing bits. May also adjust data for an even balance of ones and zeros. (This is an optional block sometimes not included in a SERDES.)

Line Decoder: Decodes from line encoded data to plain data. (This is an optional block that is sometimes done outside of the SERDES.)



Why are they so fast?



01_so_hssio_basics.odp Date Oct 23, 2009 Page 15

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

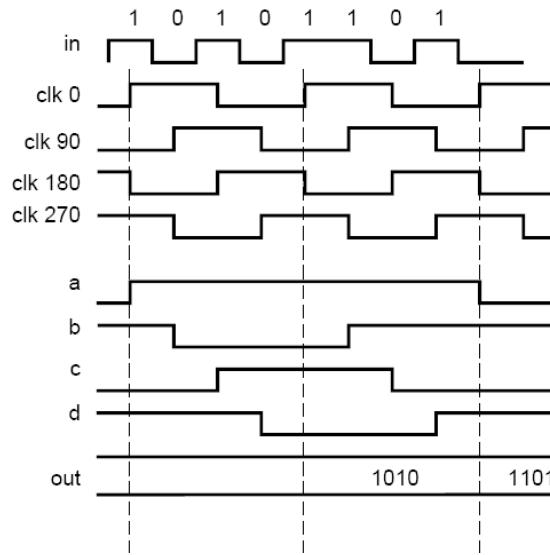
An unsettling aspect of the Gigabit SERDES is that they appear to be almost magical. They work with 3, 5, and even 10+ gigabits. How is that kind of speed possible? There are several techniques that provide this speed.

A common element of most of these techniques is multiple phases. We can get an idea of how multiple phases can help us by looking at a multiphase data extraction circuit.

If we have an incoming serial stream with a bit rate of x , we can recover the stream with a clock of $x/4$ by using multiple phases of the slow clock. The incoming stream is directed into four flip-flops, each running off a different phase of the clock (0, 90, 180, and 270).



Phase



01_so_hssio_basics.odp Date Oct 23, 2009 Page 16

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Each flip-flop then feeds into a flip-flop clocked by the next lowest phase until it is clocked off the zero-phase clock. This deserializes the incoming stream into a 4-bit word running at 1/4 the clock rate of the incoming stream.

In the previous example the phase was lined up and the clock was exactly 1/4 the rate of the incoming stream. How does that happen? We must lock to the incoming stream. We could do it with a classic phase-locked loop (PLL), but that would require a full-rate clock and defeat the purpose. One of the biggest advances in high-speed SERDES involves the PLLs used in clock and data recovery. A normal PLL requires a clock running at the data speed, but there are several techniques that can be used to avoid this requirement, including fractional rate phase detectors, multi-phase PLLs, parallel sampling, and over-sampling data recovery.



Line Encoding Schemes

- Word alignment
- Clock correction mechanism
- Channel bonding mechanism
- Sub-channel creation
- 3B4B
- 5B6B
- 8B10B
- 64B66B
- 64B67B
- Scrambling



NOTES

Line encoding schemes modify raw data into a form that the receiver can accept. Specifically, the line encode scheme ensures that there are enough transitions for the clock recovery circuit to operate. They provide a means of aligning the data into words with a good direct current (DC) balance on the line.

Optionally, the line encoding scheme may also provide for implementation of clock correction, block synchronization and channel bonding, and division of the bandwidth into sub-channels. There are two main line encoding schemes—value lookup schemes and self-modifying streams, or scramblers.



8B10B Encoding/Decoding

DC balanced
Running Disparity
Control Characters
Comma Detection
Channel Bonding

8-bit Value	10-bit Symbol
00000000	1001110100
00000001	0111010100



NOTES

The 8b/10b encoding scheme was developed by IBM and has been widely adapted. It is the encoding scheme used in Infiniband, Gigabit Ethernet, FiberChannel, and the XAUI interface to 10 Gigabit Ethernet. It is a value lookup-type encoding scheme where 8-bit words are translated into 10-bit symbols. These symbols ensure a good number of transitions for the clock recovery. Table 3-1 gives a few examples of 8-bit values that would result in long runs without transitions. 8b/10b allows for 12 special characters that decode into 12 control characters commonly called K characters. We will look at K-characters in more detail, but first let's examine how 8b/10b ensures a good DC balance.



Running Disparity

Name	Hex	8 Bits	RD -	RD +
D10.7	EA	11101010	0101011110	0101010001
D31.7	FF	11111111	1010110001	0101001110
D4.5	A4	10100100	1101011010	0010101010
D0.0	00	00000000	1001110100	0110001011
D23.0	17	00010111	1110100100	0001011011



NOTES

DC balance is achieved in the 8b/10b through a method called running disparity. The easiest way to achieve DC balance would be to only allow symbols that have the same number of ones and zeros, but that would limit the number of symbols.

Instead, 8b/10b uses two different symbols assigned to each data value. In most cases, one of the symbols has six zeros and four ones, and the other has four zeros and six ones. The total number of ones and zeros is monitored and the next symbol is chosen based on what is needed to bring the DC balance back in line. The two symbols are normally referred to as + and - symbols.



Control Characters

Name	Hex	8 Bits	RD -	RD +
K28.0	1C	00011100	0011110100	1100001011
K28.1	3C	00111100	0011111001	1100000110
K28.2	5C	01011100	0011110101	1100001010
K28.3	7C	01111100	0011110011	1100001100
K28.4	9C	10011100	0011110010	1100001101
K28.5	BC	10111100	0011111010	1100000101
K28.6	DC	11011100	0011111011	11000001001
K28.7	FC	11111100	0011111000	1100000111
K23.7	F7	11110111	1110101000	0001010111
K27.7	FB	11111011	1101101000	0010010111
K29.7	FD	11111101	1011101000	0100010111
K30.7	FE	11111110	0111101000	1000010111

01_so_hssio_basics.odp Date Oct 23, 2009 Page 20

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

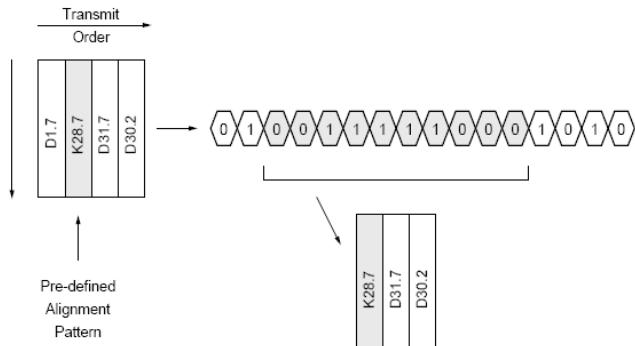


NOTES

This table lists the encoding of 12 special symbols known as control characters or K-characters. These control characters are used for alignment, control, and dividing the bandwidth into subchannels.



Comma Detection



NOTES

Alignment of data is an important function of the deserializer. This figure represents valid 8b/10b data in a serial stream.

How do we know where the symbol boundaries are? Symbols are delineated by a comma. Here, a comma is one or two symbols specified to be the comma or alignment sequence. This sequence is usually settable in the transceiver, but in some cases it may be predefined.

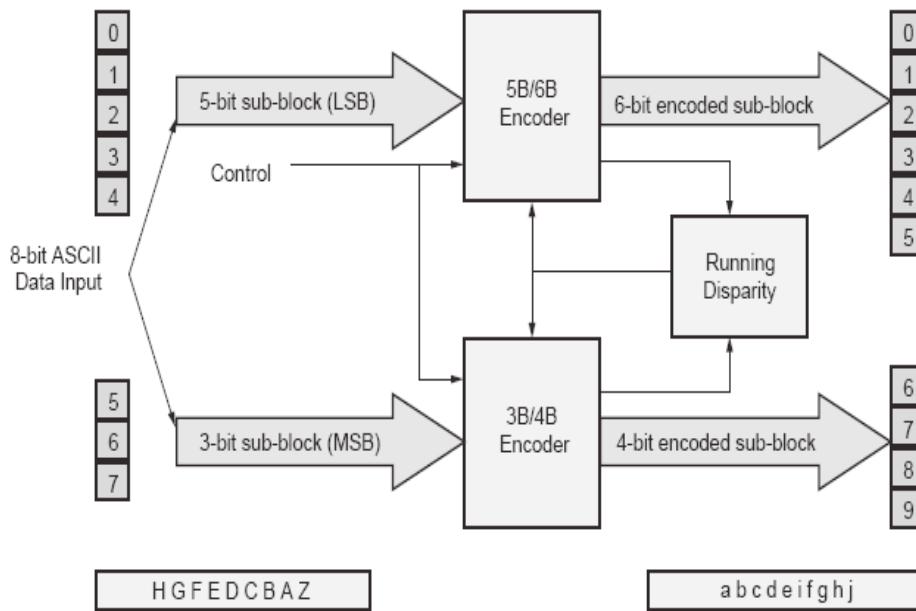
Comma: One or two symbols specified to be the alignment sequence.

The receiver scans the incoming data stream for the specified bit sequence. If it finds the sequence, the deserializer resets the word boundaries to match the detected comma sequence. This is a continuous scan. Once the alignment has been made, all subsequent commas detected should find the alignment already set. Of course, the comma sequence must be unique within any combination of sequences.

For example, if we are using a signal symbol c for the comma, then we must be certain that no ordered set of symbols xy contains the bit sequence c. Using a predefined protocol is not a problem since the comma characters have already been defined. One or more of a special subset of K-characters is often used. The subset consists of K28.1, K28.5, and K28.7, all of which have 1100000 as the first seven bits. This pattern is only found in these characters; no ordered set of data and no other K-characters will ever contain this sequence. Hence, it is ideal for alignment use. In cases where a custom protocol is built, the safest and most common solution is to “borrow” a sequence from a well-known protocol. Gigabit Ethernet uses K28.5 as its comma. Because of this it is often referred to as the comma symbol even though there are technically other choices.



Encoder Decoder Block Diagram



01_so_hssio_basics.odp Date Oct 23, 2009 Page 22

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

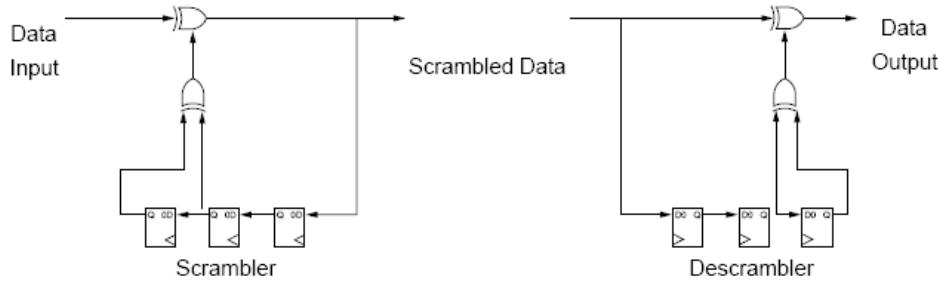
The names used—such as D0.3 and K28.5—are derived from the way the encoders and decoder scan be built.

The 8-input bits are broken into 5- and 3-bit buses; that is how the names were developed. For example, the name Dx.y describes the data symbol for the input byte where the five least significant bits have a decimal value of x and the three most significant bits have a decimal value of y.

A K indicates the control character. The three bits turn into four bits and the five bits turn into six. Another naming convention refers to the 8-bit bits as HGF EDCBA and 10-bit bits as abcdei fghj. Overhead is one of the drawbacks to the 8b/10b scheme. To get 2.5 gigabits of bandwidth requires a wire speed of 3.125 Gb/s. Scrambling techniques can easily handle the clock transition and DC bias problems without a need for increased bandwidth.



Scrambling



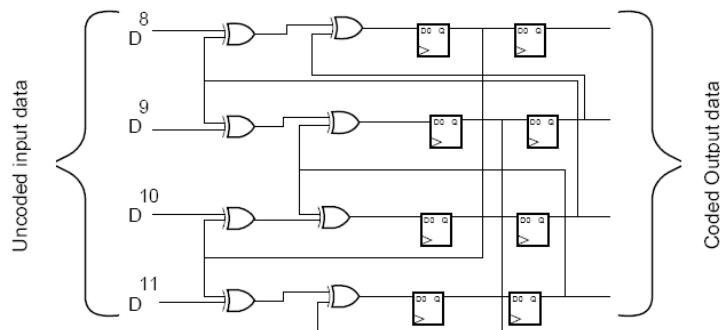
NOTES

Scrambling is a way of reordering or encoding the data so that it appears to be random, but it can still be unscrambled. We want randomizers that break up long runs of zeros and ones. Obviously, we want the descrambler to unscramble the bits without requiring any special alignment information. This characteristic is called a self-synchronizing code.

The scrambling method is usually referred to as a polynomial because of the mathematics involved. Polynomials are chosen based on scrambling properties such as how random a stream they create, and how well they break up long runs of zeros and ones. They must also avoid generating long run lengths.



Parallel Scrambling

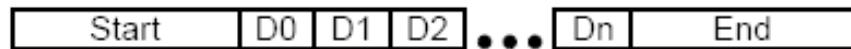


NOTES

Increasing the clock rate of the flip-flops is desirable. But obtaining a high rate such as 10 Gb/s is simply not attainable. However, there is a way to parallel any serial coefficient into a y-size parallel word to speed up the process as shown here..



Word Alignment



Start = 12 bits = 3F 00 00

End = 12 bits = 3F 3F 00

Data = 12 bits range is 04 - 3B

00,01,02,03,3c,3d,3e,3f, are forbidden in Data fields.



NOTES

Scrambling eliminates long runs and works to eliminate other patterns that may have a negative impact on the receiver's ability to decode the signal. There are, however, other tasks provided by line encoding schemes such as 8b/10b that are not supplied by scrambling:

- Word alignment
- Clock correction mechanism
- Channel bonding mechanism
- Sub-channel creation

While the last three may not be needed in some circumstances, word alignment is always needed. If scrambling is used as the line encoding method, then another method must be used for word alignment.

For example, we can exclude some values from the allowed values of the data or the payload.

Then we can use these disallowed values to create a stream of bits that could not occur in the data portion of the sequence.

Normally, this would involve designing long run lengths that cannot occur in the data stream because of the disallowed values. The long runs will be broken by the scrambling and then restored when the stream is unscrambled. Downstream unscrambler logic looks for these patterns and aligns the data. Similar techniques can be used to install any of the other characteristi



4B5B Encoder/ Decoder

Simpler Implementation
AES, MADI
Fiber Channel
Same overhead as 8B10B 20%, but less features
No DC balancing
Less control characters

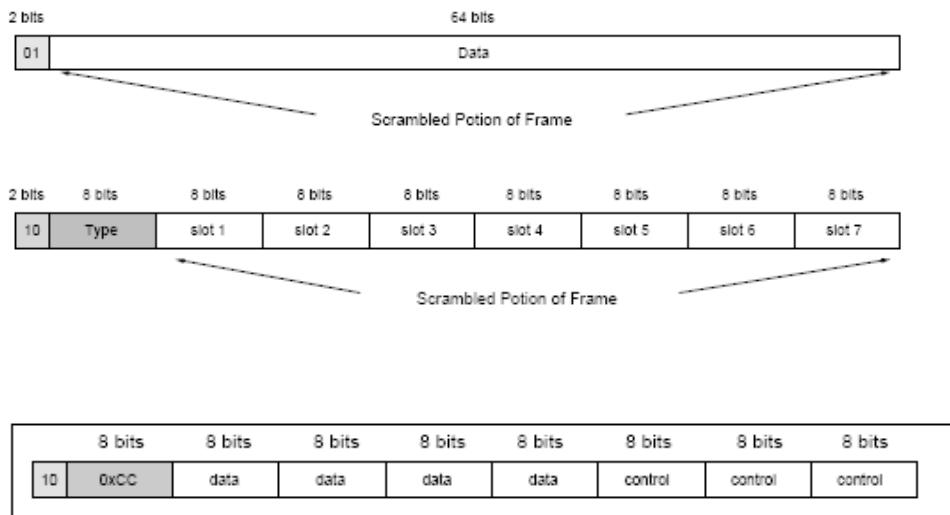


NOTES

4b/5b is similar to 8b/10b, but simpler. As the name implies, four bits are encoded into five bits with this scheme. 4b/5b offers simpler encoders and decoders than 8b/10b. But there are few control characters and it does not handle the DC balance or disparity problem. With the same coding overhead and less functionality, 4b/5b is not often used anymore. Its main advantage was implementation size, but gates are so cheap now that it is not much of an advantage. 4b/5b is still used in various standards including low bit rate versions of FiberChannel and Audio Engineering Society-10 (AES-10) or Multichannel Audio Digital Interface (MADI), a digital audio multiplexing standard.



64B66B Encoder/Decoder



NOTES

One of the new encoding methods is known as 64b/66b. We might think that it is simply a version of 8b/10b that has less coding overhead, but the details are vastly different. 64b/66b came about as a result of user needs not being met by current technology. The 10 Gigabit Ethernet community had a need for Ethernet-based communication at 10 Gb/s. And while they

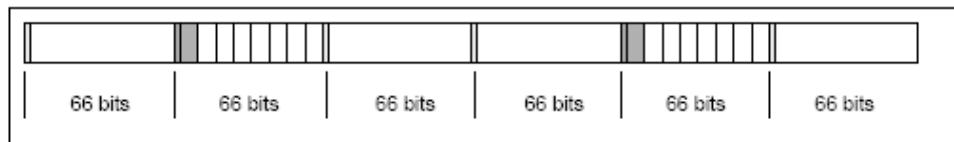
could use four links at a 2.5 Gb payload and 3.125-Gb/s wire speed, XERDES was approaching the ultimate 10 Gb solution in a single link. There were new SERDES that could run at just over 10 Gb/s, but could not be pushed to the 12.5 Gb needed to support 8b/10b overhead. The laser driving diode was another issue. The telecommunications standard Synchronous Optical Network (SONET) used lasers capable of just over 10 Gb. Faster lasers were much more expensive.

The Gigabit Ethernet community could either give up or create something with a significantly lower overhead to replace 8b/10b. They chose 64b/66b. 64b/66b: A line encoding scheme developed for 10 Gigabit Ethernet that uses a scrambling method combined with a non-scrambled sync pattern and control type. Rather than using a 8b/10b-type lookup table, 64b/66b uses a scrambling method combined with a non-scrambled sync pattern and control type.

There are two main frame types. The simple main frame consists of a 2-bit sync pattern of 01 followed by 64 bits of data. The data is scrambled but the sync bits are not. The other frame type allows for control information as well as data. Control frames start with the 2-bit pattern 10. The eight bits in the type field define the format of the 56-bit payload. For example, if the type is hex 0xcc, then the pattern contains four bytes of data and three bytes of control.



64B66B Alignment



NOTES

Sufficient Transitions

Scrambling of the payload section will provide adequate transition for clock recovery. Careful selection of the scramblers will also handle DC bias problems. The scrambler used in 64b/66b is X58 + X19 + 1.

Alignment

64b/66b differs from other methods in the alignment procedure. Figure 3-16 shows how it works. There will be a sync value of 01 or 10 every 66 bits. Those same bit combinations will appear in many other places as well. The alignment procedure selects a random starting point. It first looks for a valid sync (01 or 10 combination). If there isn't one, it slips a bit and rechecks. Once a 01 or 10 combination is found, the position 66 bits later is checked. If that is a valid sync also, the process increments the counter and checks the location 66 bits later. If enough sync markers are found in a row without any misses, the alignment is considered found. Any misses during the sequence forces the counter back to zero. Once the alignment has been locked, missed syncs are considered errors. If enough errors occur in a period of time, the alignment is re-evaluated. At first glance, it appears that this algorithm would obtain lock within the maximum number of valid sync tries (+66 or less). But the high likelihood of the 01 and 10 sequences showing up in the data window can mean many false paths are taken for a long time before they are abandoned. To speed lock time, some optional or alternative protocols have been suggested. They involve the replacement of data with special training or locking sequences that can ease alignment.



64B66

Clock Alignment or Clock Correction

Byte

Multibyte

66 bit

Channel Alignment

Sub Channel Alignment



NOTES

Clock Correction

Clock correction can be handled on byte or multi-byte boundaries, or on a 66-bit code word. A special type could be defined to be the clock correction symbol. The entire payload of the code word would not contain useful information and could be deleted or repeated as necessary. Alternatively, a bytewise clock correction symbol could be defined as any unused value. Of course, if the SERDES we are using only supports one of these methods, we will need to use that particular method. The byte-wide method is the most common since it allows for smaller receive FIFO buffers and matches up better with legacy protocols.

Channel alignment

Channel alignment can be handled much like clock alignment, either as a special type or a sequence found within the control data.

Sub-Channels

Sub-channels can be handled like clock alignment, either as a special type or a sequence found within the control data.



Line Encoder Decoder Trade Offs

Efficiency / Overhead

4B5B	25 %
8B10B	25%
64B66B	3 %
Scrambling	0 %

Complexity

4B5B	low
8B10B	low
64B66B	high (turn scrambling on and off for parts)
Scrambling	medium

Alignment Time

4B5B	fast
8B10B	fast
64B66B	slow
Scrambling	medium



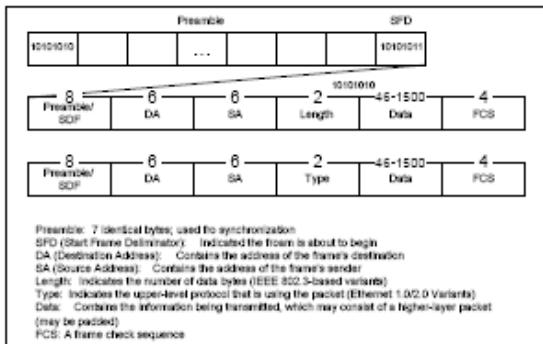
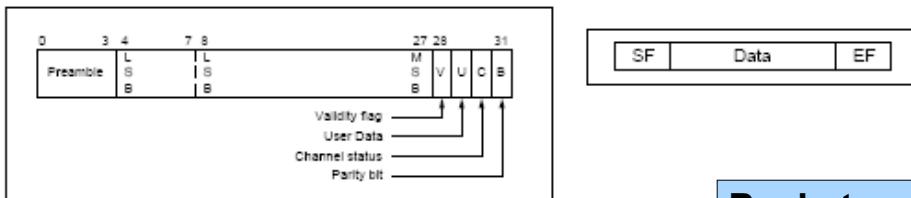
NOTES

These functions comprise the overhead coding method that allowed 10 Gigabit Ethernet to use existing SONET class laser diodes. Laser diodes are not the only similarity that this method has in common with SONET; SONET uses many of the same principles for alignment but is even more complicated than 64b/66b.

The price for the lower overhead is longer alignment times, the possibility of a slight DC bias, and more complicated encoders and decoders. Complications such as turning the scramblers on and off for payload vs. sync and type fields make 64b/66b circuits more complicated than their 8b/10b cousins. There is also a complexity cost for using and setting up the encoder.



Introduction to Packets



Packet:
A well-defined collection
of bytes consisting of a
header, data, and trailer.

01_so_hssio_basics.odp Date Oct 23, 2009 Page 31

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Some designers feel that sending data over packets for anything but a local area network (LAN) is a complete waste. Let's address that issue by first defining a packet.

Notice that there is nothing in the definition about source and destination addresses, CRCs, minimum lengths, or Open Systems Interconnection (OSI) protocol layers. A packet is simply a data structure with defined starting and ending points. While LAN packets often have many of these characteristics, there are many other uses of packets that are much simpler.

Packets are used everywhere to transfer information—automobile wiring harnesses, cell phones, and home entertainment centers, to name a few.

But what do packets have to do with gigabit serial links?

Most data transferred across a gigabit serial link is embedded in some sort of packet. It's only natural that a SERDES requires a method for aligning the incoming stream into words. This special bit sequence or comma must be sent if the system requires clock correction. The comma could be a natural marker for the beginning or end of a frame. If clock correction is required, the clock correction sequence is usually the ideal character. After adding a couple of ordered sets to indicate the end or start of the packet, and an ordered set to indicate a special type of packet, we have a simple, powerful transmission path.

The idle symbol, or sequence, is another important packet concept. This symbol is sent whenever there is no information to send. Continuous transmission of data ensures that the link stays aligned and that the PLL keeps the recovered clock locked. This slide illustrates some sample packet formats from various standards.



Reference Clock

Frequency

Output Voltage

Single Ended or Differential

PPM: Parts per million; a way of describing a very small ratio

UI: Unit intervals; same as length of time as a symbol,
i.e., 0.2 UI = 20% of the symbol time.

Jitter: Variation of the ideal transition placement



NOTES

The input, or reference clock, of a Multi-Gigabit Transceiver (MGT) has very tight specifications. It includes a tight frequency requirement usually specified in allowable parts per million (PPM) of frequency error. It will also have strict jitter requirements defined in terms of time units (picoseconds) or unit intervals (UI).

Such tight requirements enable the PLL and clock extraction circuits to work. This often requires an accurate crystal oscillator on each printed circuit board (PCB) in the system that uses MGTs. These crystal oscillators are a step above most used for digital systems and will cost more. In many cases, clock generation chips and PLLs have too much jitter to be used.

The tight jitter requirements on a transmit clock normally prevent a Gigabit SERDES from using a recovered clock as a transmit clock. Each PCB assembly has a unique oscillator and a unique frequency. If the two oscillators are just 1 PPM off frequency from each other at 1 GHz, and we supply a 1/20th rate reference clock, clock one of the streams will be faster or slower 20,000 times per second. Hence, in an 8b/10b encoded system there would be twenty thousand extra or missing symbols every second.



Clock Correction Table

						Max Cycles before Correction			
Oscillator Frequency (MHz)	OSC Accuracy (PPM)	Line Speed (GB/s)	Fmax (MHz)	Fmin (MHz)	Diff/Cycle (ps)	Remove 1 Sequence	Remove 2 Sequences	Remove 3 Sequences	Remove 4 Sequences
156.25	100	3.125	156.2656	156.2344	1.2800	4,999	9,999	14,998	19,998
156.25	50	3.125	156.2578	156.2422	0.6400	9,999	19,998	29,998	'39,997
156.25	20	3.125	156.2531	156.2469	0.2560	24,999	49,999	74,998	99,998
125	100	2.500	125.0125	124.9875	1.6000	4,999	9,998	14,998	19,997
125	50	2.500	125.0063	124.9938	0.8000	9,999	19,998	29,998	'39,997
125	20	2.500	125.0025	124.9975	0.3200	24,999	49,998	74,998	99,997
62.5	100	1.250	62.5063	62.4938	3.2000	4,999	9,998	14,998	19,997
62.5	50	1.250	62.5031	62.4969	1.6000	9,999	19,998	29,998	'39,997
62.5	20	1.250	62.5013	62.4988	0.6400	24,999	49,998	74,998	99,997

01_so_hssio_basics.odp Date Oct 23, 2009 Page 33

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Most SERDES have clock correction options built in. Clock correction involves a unique symbol or sequence of symbols not found elsewhere in the data stream. Since clock correction is downstream from alignment, this can easily be accomplished by reserving one K-character or ordered pairs of K-characters and/or data characters as the clock correction sequence. In some cases a four-symbol clock correction sequence may be desired. Clock correction works by monitoring the receive FIFO. If the FIFO is getting close to full, it simply looks for the next clock correction sequence and does not write that data sequence into the FIFO. This is called dropped. Conversely, if the FIFO is getting close to empty, the next time a clock correction sequence is found it will be written into the FIFO twice. This is commonly referred to as repeating. The clock correction must happen often enough to allow dropping or repeating to compensate for the differences in the clocks. Often the clock correction sequence will also be the same as the idle sequence. Some systems do not require clock correction. In many chip-to-chip applications, for example, the same oscillator will provide the reference clock to all transmitters. Using the same reference clock and same rate means there is no need for clock correction. Also, a clock correction is not needed when all of the receive circuitry is clocked from the recovered clock. If the FIFO is emptied at the same rate it is filled, there is no need for clock correction. Also, clock correction is not required when all transmit reference clocks are locked using an external PLL to a common reference. This is a common architecture for high definition serial digital video links. All transmit clocks are derived from a common video reference. Failure to lock to this signal will usually result in a free running video stream that tends to roll in respect to the rest of the locked signals. While achieving this at one or two gigabits is easily possible, designing PLLs with enough accuracy to provide the input reference clocks for 10-Gb links is quite challenging.



Receive and Transmit Buffer

Interface to MGTs

Writing fifo_full, fifo_write

Reading fifo_empty, fifo_read

Writing Side can overrun

Reading Side can underrun

Compensate clock differences => Clock Compensation

Asynchronous FIFO, two different clock domains

Isochronous: Matched in frequency but not necessarily matched in phase.



NOTES

The receive and transmit buffers, or FIFOs, are the main digital interface of the Multi-Gigabit Transceiver. This is normally where data is written and read. On the transmit side it is common to have a small FIFO that requires the read and the write clock to be isochronous (matched in frequency but not necessarily matched in phase).

A different scheme is used in cases where the tx_write and tx_read strobes are not of the exact same frequency. Here, a larger FIFO is used and its current status is constantly monitored. If the FIFO is filling it will eventually overrun. In this case the incoming stream is monitored for idle symbols. When encountered they are not written into the FIFO.

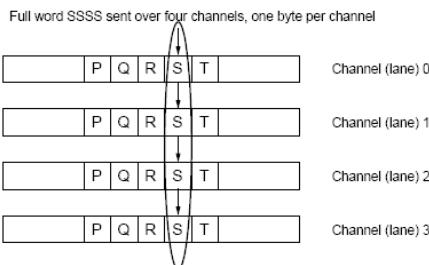
Conversely, if the FIFO is running low when an idle is found on the output, the data is brought to the user. The write pointer is not moved causing the idle to be repeated. It is important for idle symbols to be used instead of byte alignment, comma symbols, clock correction sequences, or channel bonding sequences. All these are needed downstream at some guaranteed delivery rate.

The receive FIFO built into an MGT is usually considerably deeper than the transmit (Tx) buffer. Its main purpose is to allow for clock correction and channel bonding.

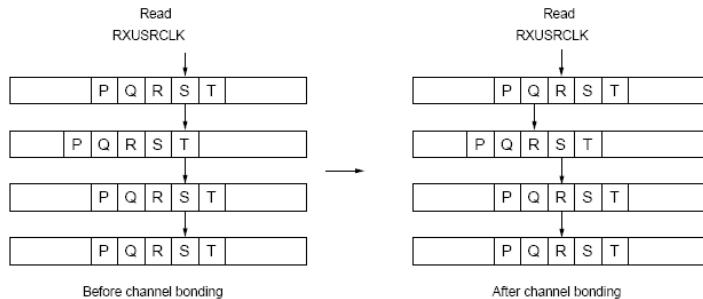


Channel Bonding

In Transmitters:



In Receivers:



01_so_hssio_basics.odp Date Oct 23, 2009 Page 35

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Channel Bonding: Absorbs the skew between two or more MGTs and presents the data to the user as if it were transmitted over a single link.

Sometimes there is a need to move more data than can fit on one serial link. In these cases multiple links are used in parallel to transmit the data. When this is done, incoming streams must be aligned. This process is commonly referred to as channel bonding. Channel bonding absorbs the skew between two or more MGTs and presents the data to the user as if it were transmitted over a single link.

There are several causes of data skew between multiple MGTs:

- Differences in transmission path length
- Active repeaters in transmission path
- Differences because of clock correction
- Differences in time to lock/byte alignment

Since channel bonding requires communication between transceivers, the exact details will vary from vendor to vendor and part to part. Some common traits are designation of one channel as the master channel, designation of slaves, and possibly the designation of forwarding slaves. Three-level channel bonding that includes a master and forwarding slaves is sometimes referred to as two-hop channel bonding. The channel bonding sequence must be unique and expandable and it must be ignored downstream because it may be added or dropped. There are normally a minimum number of symbols between a clock correction sequence and a channel bonding sequence. Many 8b/10b-based standard protocols specify a minimum of four symbols between clock correction and channel bonding sequences. Hence, four symbols or bytes is a common separation distance.



Physical Signaling

CML: Current Mode Logic; a differential-based electrical interface well suited to the gigabit link.

Front-end and back-end together make up the physical interface.

Voltage Standard

LVDS

LVPECL

CML



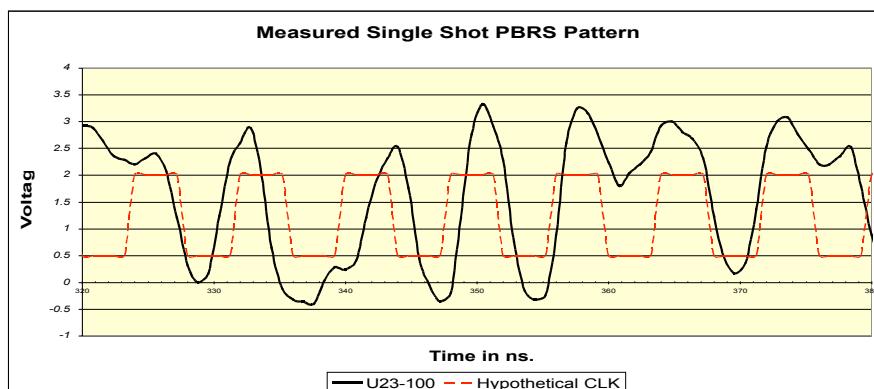
NOTES

The physical implementation of multi-gigabit SERDES universally takes the form of differential based electrical interfaces. There are three common differential signal methods—Low-Voltage Differential Signaling (LVDS), Low Voltage Pseudo Emitter-Coupled Logic (LVPECL), and Current Mode Logic (CML). CML is preferred for the gigabit link. It has the most common interface type and often provides for either AC or DC termination and selectable output drive. Some inputs provide built-in line equalization and/or internal termination. Often the termination impedance is selectable as well.



Real World Digital Signals

- High-speed design must contend with the analog world
 - The concept of a “1” and “0” is an idealization
 - Typical signals can look like the following



01_so_hssio_basics.odp Date Oct 23, 2009 Page 37

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

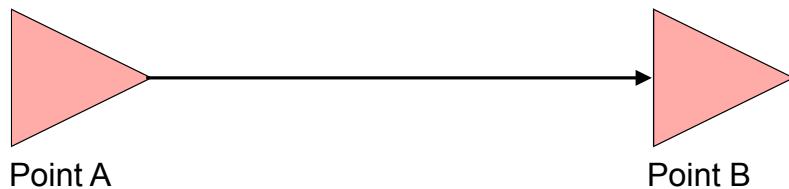
The dashed line represents the underlying theoretical clock. This theoretical signal resembles the classic form of a digital waveform. Clearly, this signal has a high and low level. The “wavy” signal represents an SDRAM data pattern (DQ) bit.

The analog nature of the signal is also clearly apparent. No longer do you live in the strictly digital realm. Analog effects must be accounted to ensure proper digital operation. Thus, the concept of signal integrity.



Why All the Fuss?

- Schematically, you are doing this



- What is the reality?



NOTES

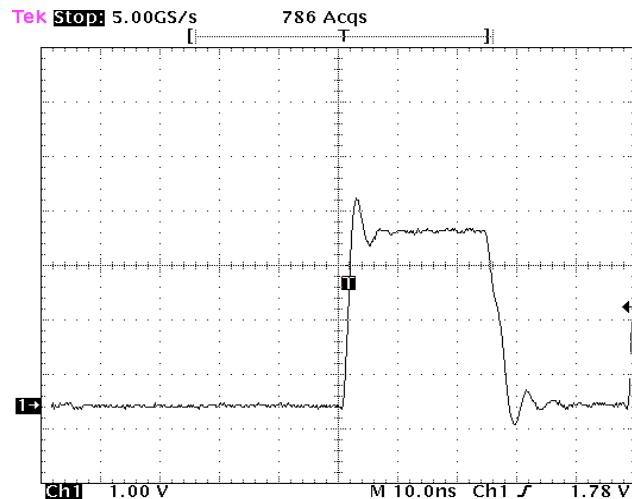
How can sending energy from point A to point B be difficult?

Note to Facilitator:

This slide establishes signal integrity as an aspect of the simple, direct transmission of energy from point A to B. The waveforms shown in the following slides show the reality: that the energy, represented by the signal, undergoes changes along the way. Topics in the SI part detail the sources for these changes.



A Bit of This



01_so_hssio_basics.odp Date Oct 23, 2009 Page 39

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

At times, you see overshoot and undershoot characteristics on your lab waveforms. Overshoot can reduce reliability and, in extreme cases, inflict damage. Undershoot can do the same, as well as causing reverse biasing of the substrate — making the device work in ways in which it was never intended!

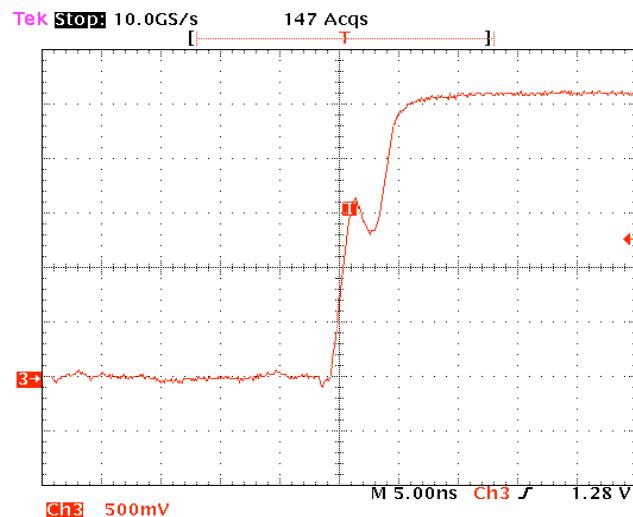
Note to Facilitator:

While the topics in Part I, Signal Integrity discuss the sources for the changes in energy transmission between point A and B, you can engage students' interest early on, and show them that the course is personally relevant by asking them if they have seen results like these, and what were their sources. Repeat your requests for the waveforms on the next two slides.

For this slide and the two that follow, point out the critical peaks in the waveforms (max ratings from the data sheet).



A Little of That



01_so_hssio_basics.odp Date Oct 23, 2009 Page 40

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

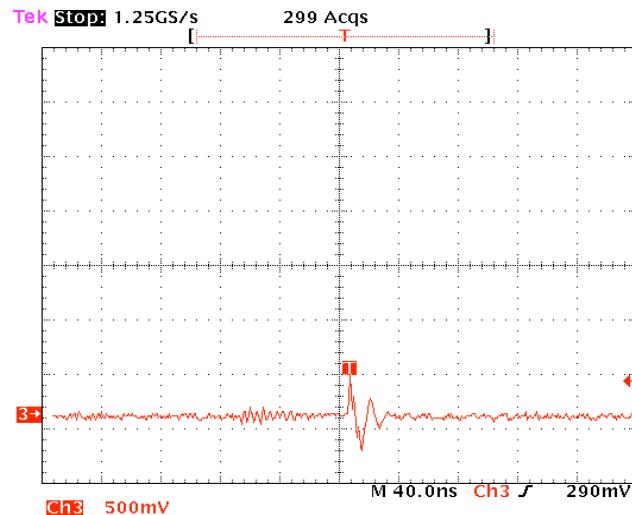


NOTES

Non-monatomic behavior is the classic cause of double-clocking.



And a Sprinkle of the Other



01_so_hssio_basics.odp Date Oct 23, 2009 Page 41

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

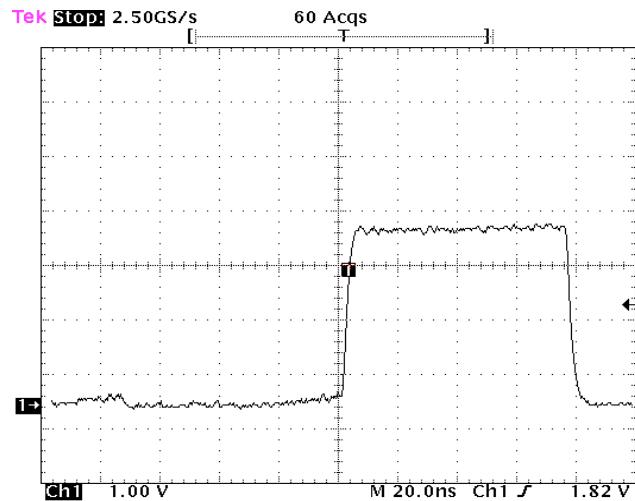


NOTES

A glitch like this in an otherwise quiet zone indicates either Simultaneous Switching Output (SSO) noise or crosstalk.



But You Really Wanted This



01_so_hssio_basics.odp Date Oct 23, 2009 Page 42

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



So What is *High Speed*?

- High speed is not high clock rate
 - The clock rate can give an orientation only
- Propagation delay is proportional to the square root of the dielectric constant
 - Outer layer: 55 ps/cm
 - Inner layer: 70 ps/cm
- The main issue is the edge speed (rise/fall time) in relation to the line length
 - Length of the rising edge (L) = rise time [ps] / delay [ps/cm]
 - Traces smaller than $L/5$ are lumped traces
 - Traces greater than $L/5$ must be transmission lines
 - 1 ns rise time with 70 ps/cm has a length of about 14 cm
 - Traces greater than about 3 cm must be transmission lines



NOTES



Signal Integrity Issues

- Reflection
 - Overshoots and undershoots
 - Multi-crossing errors
 - Threshold errors
 - Signal oscillation
- Crosstalk
- System timing
- EMI (Electro-Magnetic Interference)
 - Radiation
- Power distribution system
 - Bypassing

01_so_hssio_basics.odp Date Oct 23, 2009 Page 44

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

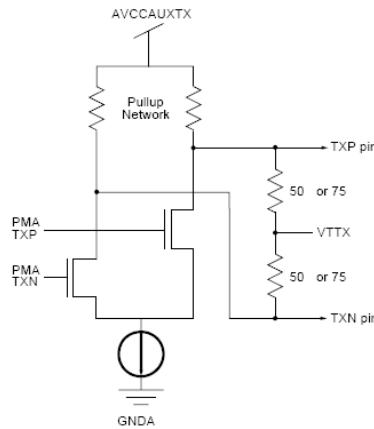


NOTES

This slide gives some topics of interest for investigating signal integrity.



CML Driver



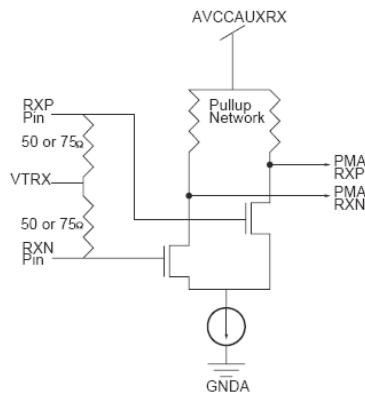
NOTES

The concept behind this high-speed driver is quite simple. One of the two resistors always has a current running through it that is different than the current running through the other.

Parameter	Min	Typ	Max	Units	Conditions
V_{OUT}	Serial output differential peak to peak (TXP/TXN)	800		mV	Output differential voltage is programmable
V_{TTX}	Output termination voltage supply	1.8		V	
V_{TCM}	Common mode output voltage range (no transmission line connected)	1.1		V	
V_{TCM}	Common mode output voltage range (transmission line connected)	1.1		V	The common mode depends on coupling (DC or AC), VTTX, VTRX, and differential swing. Spice simulation gives the exact common mode voltage for any given system.
V_{ISKEW}	Differential output skew		15	ps	



CML Receiver



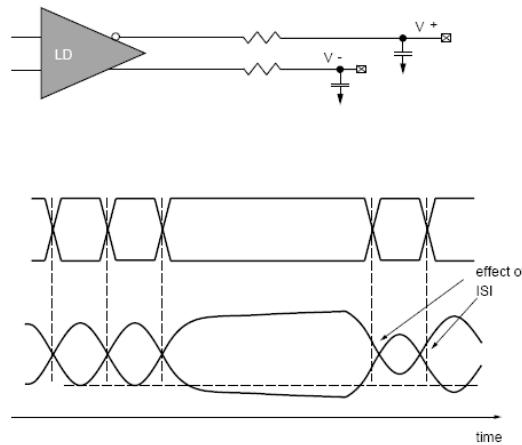
NOTES

Parameter		Min	Typ	Max	Units	Conditions
V_{IN}	Serial input differential peak to peak (RXP/RXN)	175		2000	mV	
V_{ICM}	Common mode input voltage range	500		2500	mV	
T_{ISKEW}	Differential input skew			75	ps	
T_{JTOL}	Receive data total jitter tolerance (peak to peak)			0.65	UI ⁽¹⁾	
T_{DJTOL}	Receive data deterministic jitter tolerance (peak to peak)			0.41	UI	



Tx Pre-Emphasis

Pre-emphasis: Intentional overdriving at the first of a transition

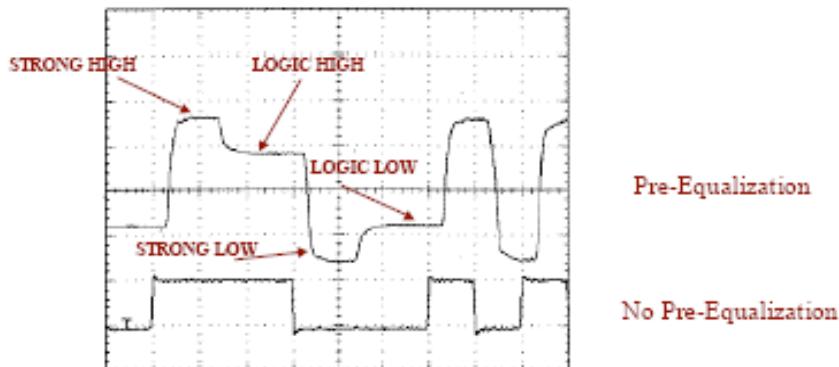


NOTES

Perhaps the most important characteristic of a multi-gigabit driver is its ability to perform preemphasis. Pre-emphasis is the intentional overdriving at the beginning of a transition. To the inexperienced eye it looks like a fault; it looks like overshoot and undershoot that can indicate a bad design. To understand why this is done, we need to understand inter-symbol interference (ISI).



Tx Pre-Emphasis



NOTES

ISI: Inter-symbol interference: Occurs when the serial stream contains a number of bit times of the same value followed by short bit times of the opposite value.

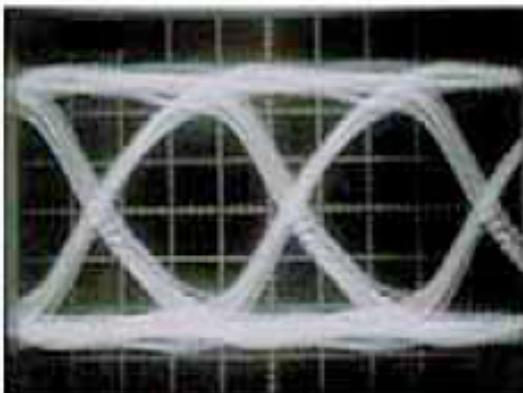
ISI occurs when the serial stream contains a number of bit times of the same value followed by short (1 or 2) bit times of the opposite value. The medium (transmission path capacitance) has less time to charge during the shorter value time, so it produces lower amplitude.

With ISI, the larger runs allow for maximum charge but the single bit time cannot compensate. It is at risk of not being detected. The solution to this problem is to overdrive the first of each transition, or underdrive any consecutive bit times of the same value. This is sometimes called de-emphasis.

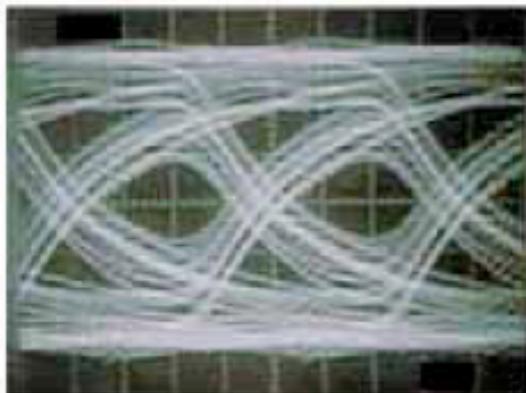


Eye Pattern

Common waveform viewed on digital sampling scopes. It is an indication of the quality of the signal. Jitter, impedance matching, and amplitude can all be characterized through eye patterns.



Pre-Emphasis



No Pre-Emphasis

01_so_hssio_basics.odp Date Oct 23, 2009 Page 49

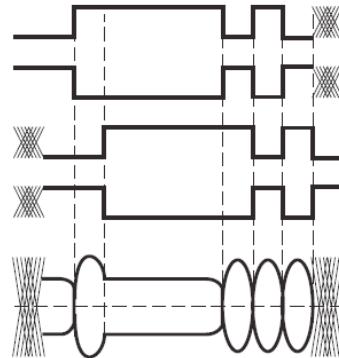
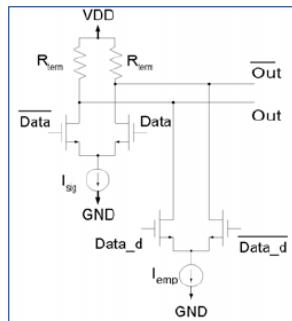
SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Pre-Emphasis Implementation

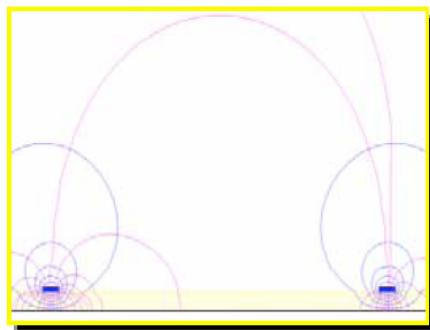


NOTES

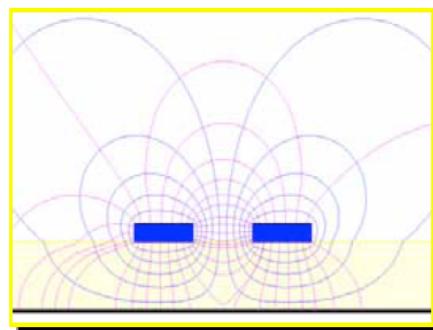
Pre-emphasis can be implemented by using two CML drivers in parallel where one is delayed one bit time after the other.



Transmission Lines



Weakly Coupled



Strongly Coupled

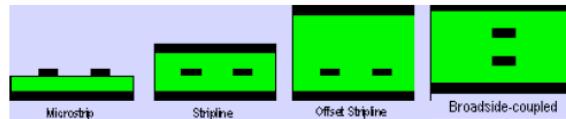
NOTES

Digital design engineers and PCB designers once thought of traces as simple interconnects or wires. In fact, prototypes were built using a technique called wire warping. Transmission lines and transmission line theory were not necessarily applied. When the propagation delay of the trace was a tiny fraction of the rise time of the signal, this was satisfactory. But as signals increased in frequency, transmission line theory had to move into the PCB design process.

For multi-gigabit operation this includes not only transmission lines and controlled impedance, but differential pair controlled impedance as well. Differential pair impedance matched traces are two traces that run adjacent to each other. The spacing between the pair allows for a coupling to occur between the traces. The coupling is called weak on left hand side figure, if the traces are relatively far apart. If the traces are closer it is called strongly coupled on the right hand side.



Controlled Impedance



Type	Pros	Cons
Microstrip	Less loss than internal traces	Only two layers (top and bottom) More susceptible to interference
Stripline	Better shielding More possible layers	More amplitude loss per inch in high frequency signals than microstrip
Offset Stripline	Useful if non-symmetrical Stack-up is needed. Can be used to limit the number of power/gnd planes.	If used to save layers, the offset area above the traces should be kept free of other traces and must be free of parallel traces.
Broadside-coupled	Very tight coupling	The broadside coupled is difficult to manufacture because of tight tolerances and it is not recommended for multi-gigabit operation.

01_so_hssio_basics.odp Date Oct 23, 2009 Page 52

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



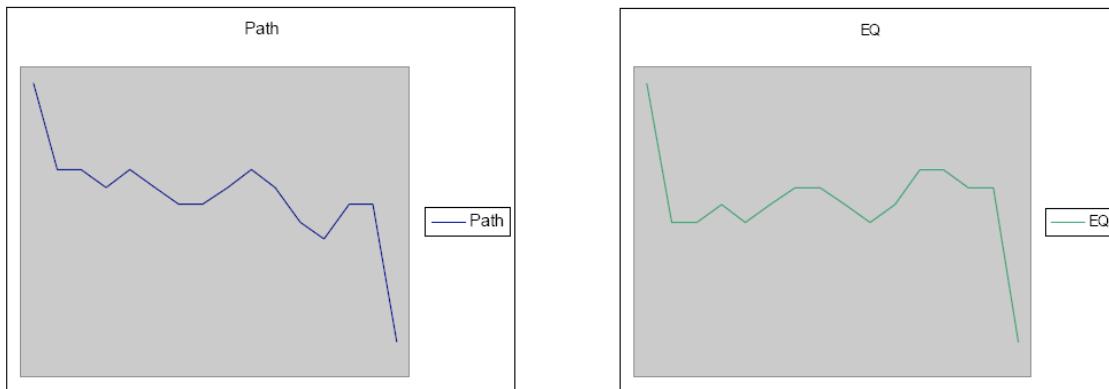
NOTES

The coupling also affects the impedance of the trace if a given trace length and layer stack will make a given impedance. The same geometry for a differential pair will have a different impedance. The exact dimensions for any given impedance varies on material, but the board manufacturer can often provide the exact dimensions.

A tool/mathematical model called a field solver can calculate the numbers as well. This slide shows the main types of controlled impedance differential traces—microstrip, stripline, offset stripline, and broadside coupled. The table includes controlled impedance differential trace types.



Rx Equalization



NOTES

Equalization is an attempt to compensate for differences in impedance/losses relative to frequency. Equalizers come in many forms but can generally be divided into passive and active types.

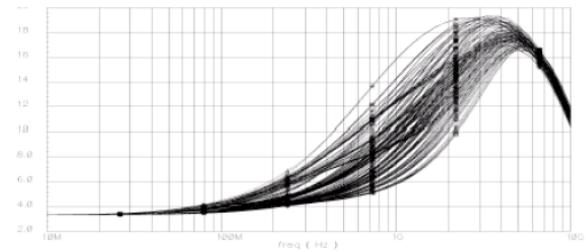
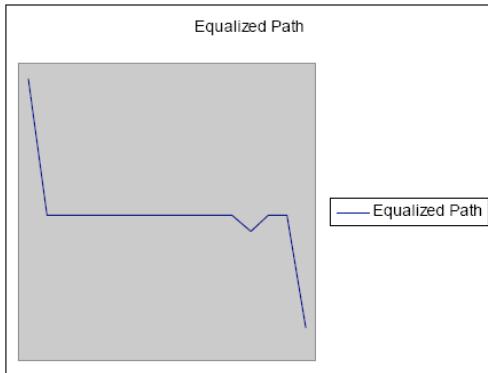
Active equalizer: Frequency dependant amplifiers/attenuators.

Passive equalizer: A passive circuit with a frequency response that is complementary to the transmission losses; similar to a filter

A passive equalizer is a passive circuit that has a frequency response that is complementary to the transmission losses. A passive equalizer can be thought of as a filter. If we filter out the frequencies that the transmission line passes, and not filter those that it does not pass, we can flatten the overall response as shown in on the left hand side. The active equalizers can be thought of as frequency-dependant amplifiers/attenuators. There are two types of active equalizers—fixed pattern and self-adjusting. No matter what the incoming data stream looks like, the fixed pattern active equalizer has the same frequency response



Rx Equalization



NOTES

This set gain attenuation pattern may be user-selectable or programmable. Some have a simple control— n settings with high or low gain. They are similar to the bass control on a simple audio system. Or they could allow for individual settings at various frequency bands much like the equalizer settings on a more complex audio system. A chart showing the possible frequency response for one such equalizer is shown in the figure on the right hand site.

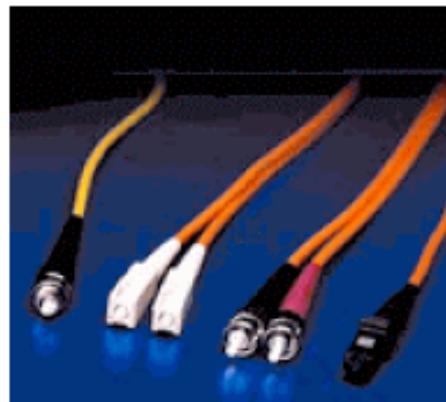
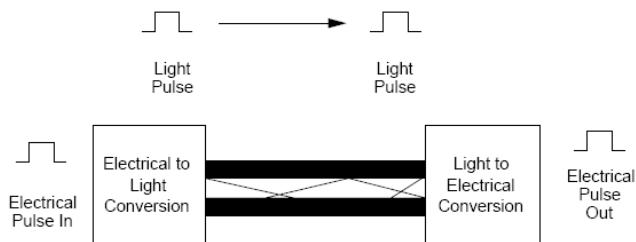
This set gain attenuation pattern may be user-selectable or programmable. Some have a simple control— n settings with high or low gain. They are similar to the bass control on a simple audio system. Or they could allow for individual settings at various frequency bands much like the equalizer settings on a more complex audio system.

Cables can also be equalized. The most common cable equalization technique is to add a passive equalizing circuit in the cable assembly, usually in the connector. Some higher-end cables obtain equalized-type characteristics through novel cable construction techniques involving silver plated solid copper cables



Optical Transmission

A basic optical system consists of a transmitter or source, the fiber, and a receiver.



NOTES

The design solution will likely be optical if cables go much further than the adjacent chassis. With optical, there are a wide variety of optical choices to pass signals upstairs, across the building, around the block, or across town.

Fiber optic systems use light instead of electricity to transport information. The basic systems consist of a transmitter or source, the fiber, and a receiver that converts the light pulse back into an electrical signal. The source is usually an injection laser diode (ILD) or a light emitting diode (LED) as shown in this figure.

Fiber allows transport of light pulses because of the principle of total internal reflection. This principle states that when the angle of incidence exceeds a critical value, light cannot get out of the glass. Instead, it bounces back in. In simple terms, fiber is like a long flexible paper towel-sized tube lined with a mirror and a flashlight. When shining a flashlight down the tube, even if the tube is bent around a corner, the light will continue to the end.

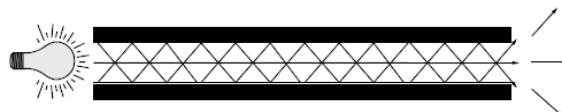


Fibers

Total Internal Reflection: When the angle of incidence exceeds a critical value, light cannot get out of the glass. Instead, the light bounces back in.



Single Mode Fiber SMF Path



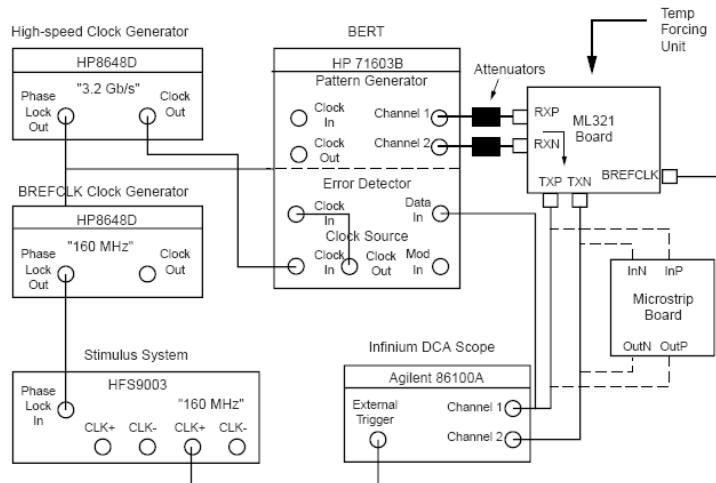
Multi Mode Fiber MMF Path

NOTES

Total Internal Reflection: When the angle of incidence exceeds a critical value, light cannot get out of the glass. Instead, the light bounces back in.



Bit Error Testing



01_so_hssio_basics.odp Date Oct 23, 2009 Page 57

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

So why are serial links different?

1. Cosmic rays can cause errors especially if they happen to hit during a transition. The faster the signal, the more transitions and the more likely a cosmic ray will occur during a transition.
2. For any given BER, the faster the signal, the more likelihood of an error.
3. High-speed clock data recovery is not an exact science. Jitter, ISI, and a host of other real world interferences can cause a bad data decision that results in an error. For example, PLLs, are constantly trying to adjust to the changing incoming signal. And as oscillators drift with temperature, errors can occur. The bit error rate (BER) is a concern for gigabit links designers, especially when moving from a parallel to serial backplane system. No link has a BER of zero because there is always some potential for errors. In many lower rate systems, the likelihood of errors is due to cosmic ray interference. And the likelihood of that is so small that it is essentially zero.

Realities of Testing

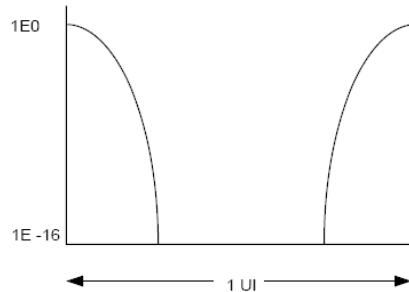
While the above reasons have a real effect, careful analysis of parallel backplanes, source synchronous links, or any communication channel could find similar faults. But for the most part, these are routinely assumed to be close to zero and ignored.

Why shouldn't we have the same concerns in Gigabit SERDES? Because their original environment is the communication industry with long and short haul optic transports. This is an industry that has always worried, tested, designed, and specified Bit Error Rates.

Some of the standards such as XAUI and some of the SONET variations specify a maximum BER. Unfortunately, testing for BER is difficult, boring, and time consuming. And it gets exponentially more difficult to get a unit of improvement. BERs are normally expressed in 10-x notation, so to move from 10-8 to a 10-9 takes 10-x the time. Testing becomes impractical at some point. Hence, most manufacturers test to the tightest BER in a published standard and no further.



Bath Tube Curve



NOTES

The bathtub curve is a plot that shows the bit error rate relative to sampling position within the unit interval. The bottom of the curve is not zero, but however close to zero the testing stopped, normally somewhere in the 10-12 to 10-16 range. The upper limit is a 100% bit error rate or 1. A bit error tester is often used to generate the bathtub curves. Under some circumstances, a relationship between an eye pattern and the bathtub curve can be demonstrated like this one from a BER test vendor called Wavecrest.



WaveCrest

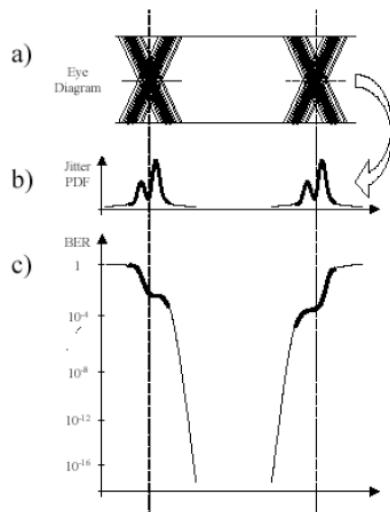


Illustration of relationship between eye diagram, jitter PDF, and bathtub curve.

- a.) Eye diagram indicating data transition threshold.
- b.) Jitter PDF (think line) with TailFit™ extrapolation (thin line).
- c.) Bathtub curves found from jitter PDF (thick line) and TailFit extrapolation (thin line).

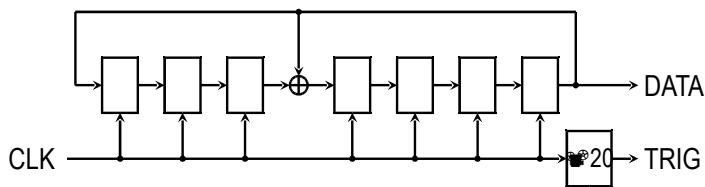
NOTES

This relationship is only valid under the correct conditions. Trying to deduce information about a bathtub curve or BER from an eye pattern from a transmission path external to an MGT that includes an active equalizer would not be valid.



Cyclic Redundancy Check CRC

- Use Pseudo Random Bit Sequence (PRBS)
 - Highlight data-dependent jitter—ISI for example
- Example: 2^7-1 PRBS
 - Polynomial: $1 + x^3 + x^7$
 - Similar to 8B/10B (RLL = 7, 63 zeros/64 ones)
 - Repeats every 127 bits
 - Trigger at 1/20th bit rate with infinite “persistence”



NOTES

But a designer must still design a system that is robust. To do so, he must first examine the system requirements to see if he can use the same commonly-used methods that contributed to the problem. One method is error detection data retransmission. The incoming data is examined for errors. If any are discovered, a message is sent to the sender to retransmit. The preferred method for error detection is CRC. This is so common that many SERDES include CRC generation and checking hardware directly within the SERDES. Often, the retransmission request is built into an upper level protocol.

This is the best solution if the protocol used supports CRCs and retransmission, or if the data requirements are such that they can be implemented. If this is not possible, there are other options. The designer could simply build and test the system and see if it works. The published BER for the selected SERDES is a specification of how far it was tested, so the designer has some room to maneuver. It is possible that he can build a system far better than the published number. Besides being a specified testing stop point, the testing was probably done at the extremes (input jitter very near the maximum, etc.). If he designs the system to provide a better input stream, he will get better results.

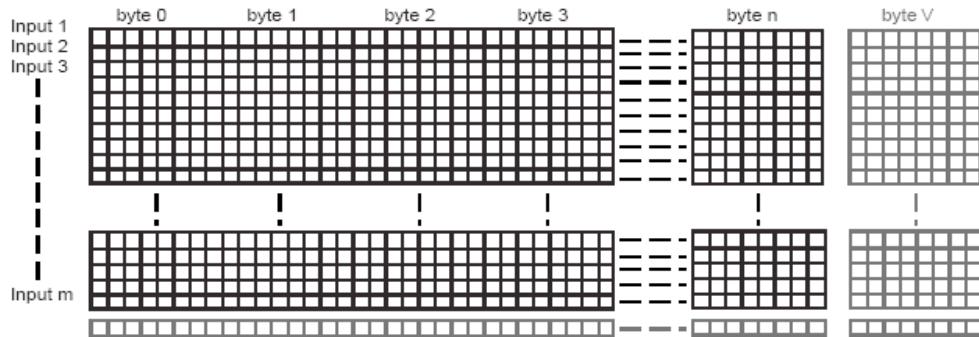
Data offers another option to consider. Most data streams have a pattern and they are much more predictable than the pseudo-random bit streams used for BER testing. This can be good or bad depending on how the transmission path and equalizers react to the stream. This must be tested and adjusted.

So it is not completely far-fetched to build a system and see if it will work. However, if doing this presents a management concern, forward error correction (FEC) can alleviate concerns.



Forward Error Correction FEC

FEC Forward Error Correction: Extra bits are added to data to help recover from an error.



NOTES

Since the designer knows that errors are going to occur, he can prepare to recover from those error by providing extra data bits.

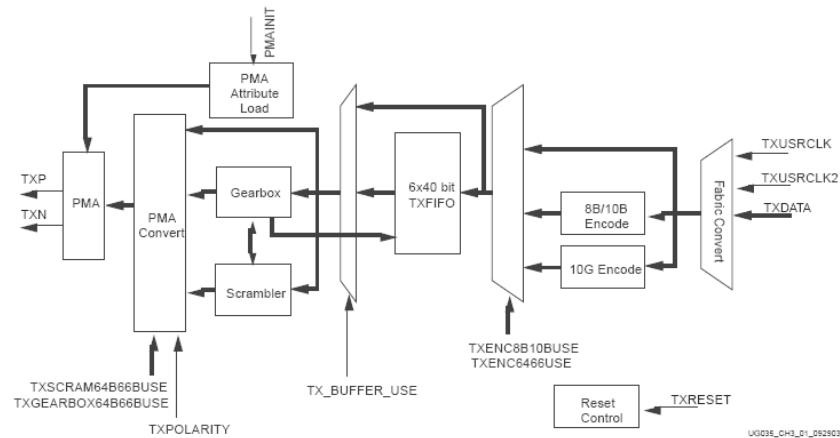
Let's examine how FEC works. Consider a block of data to be transmitted $N \times R$ bytes long and divide it into a matrix N bytes by R rows. Now add one extra byte to each row and one extra row to the matrix. These are the extra slots. Additional information about the data block will be put in these slots. In this example, the extra information is parity bits. Each bit of the extra byte on each row represents the parity of that specific bit for each byte on the row. That is, $P[1][0]$ is the parity of $D[1.1][0] D[1.2][0] D[1.3][0] D[1.N][0]$. Then for the extra row, the parity of the bits directly above are taken. That is, $P[R+1.0][0]$ is the parity of $D[0.0][0], D[1.0][0], D[2.0][0], D[N.0][0]$. A diagram of this matrix is shown in the slide.

The data and extra bits are transmitted over the link. On the other side, the matrix is examined for parity errors. If any one bit of data is the wrong value, it will be flagged and identified by row and column. This bit could then be corrected by a simple inversion. Depending on where errors occur, multiple errors could either be corrected or they could cause confusion and prevent the correction of other errors. This method is known as a simple parity matrix and was the first type of FEC. It is the basic building block for most FEC methods. While this example is straightforward, it does have limitations. Some FEC methods have been developed for harsh environments or dirty channels like Viterbi, Reed-Solomon, or Turbo Product codes. All have powerful correction, but that correction comes at a cost:

- They do not go very fast. Gigabit SERDES are faster than most of these methods can handle in their normal construction.
- They are too big. The encoders and decoders may consist of ten times as much logic as the MGT and/or the rest of the design.
- The coding overhead is too great. The coding overhead is the added bits. Often the coding overhead can completely eliminate the feasibility of the FEC method.



Transmitter Architecture



01_so_hssio_basics.odp Date Oct 23, 2009 Page 62

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



MGT Debugging

Debugging your multi-gigabit design can sometimes be a challenge. Some debugging hints you can use cover the following areas:

- Low Signal Amplitude
- Low Eye Pattern Height
- Excessive Jitter
- Using SI Tools
- Power Supply
- Cross talk
- Asymmetrical rise/fall times
- Common mode distortions
- Oscillator wander and jitter
- Noise



NOTES

Low Signal Amplitude at the Receive Pins

If our amplitude is too low, we may be able to crank up the voltage of the output driver. If we cannot fix the problem with output drive, we have too much loss in our boards and connectors; at this point we will really wish we had done those analog simulations because we are looking at a board redesign. Before we concede, we will want to make sure it is not a test setup problem or a manufacturing defect. Check all connections, part numbers, component values, and so on. We may also want to check the amplitude at various points along the path to get a feel for where the loss is occurring.

Low Eye Pattern Height

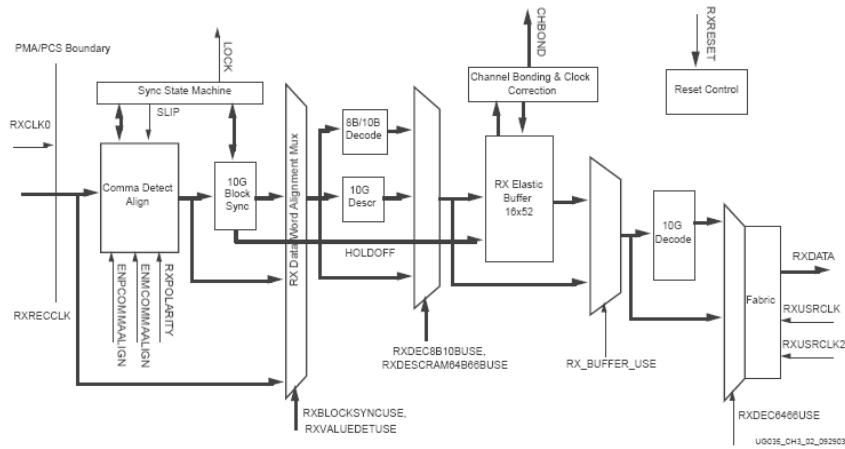
If the overall amplitude is high enough but the height of the eye pattern is small, then some bits are getting high enough, but others are not. This is often a result of a difference in gain attenuation of the path or transmitter at some frequencies. Usually the easiest thing to try is to check our pre-emphasis settings. It could be that we are just not getting high enough on single bit transitions. If we have an equalizer or equalized cable in the path, we will want to check and make sure they are the correct values. If we can adjust the equalization, we should try that.

Excessive Jitter for Receive

This is the most common problem of non-working links. Low eye pattern height will often accompany jitter problems, so all of the suggestions for low eye pattern height apply here as well. If it is not a preemphasis problem and the associated jitter from internal signal integrity or our equalizer settings, it is time to start looking for other sources of jitter. Some likely candidates are:



Receiver Architecture



01_so_hssio_basics.odp Date Oct 23, 2009 Page 64

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Simulation

- Digital Simulation
 - Only Protocol, very limited
- IBIS Model
 - not always sufficient for SERDES modeling
- Encrypted SPICE has been used
 - ELDO / HSPICE transistor-level models
 - Greater flexibility ↔ greater complexity
 - Longer simulation times than IBIS
- AMS models
 - Modeling language with sequential, concurrent, simultaneous programming
 - Internationally recognized standard
 - VHDL-AMS and Verilog-AMS
- Signal Integrity Analysis
 - Board Level
 - Transmission Lines
 - StackUp

01_so_hssio_basics.odp Date Oct 23, 2009 Page 65

SO-LOGIC electronic consulting Austria & Brazil, Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



What are IBIS Files?

I I/O
B Buffer
I Information
S Specification



- IBIS is a standard for describing the analog behavior of the buffers of digital devices using plain ASCII text formatted data
 - IBIS home page: www.eigroup.org/ibis
- Used by modern simulation tools, such as the HyperLynx, SPECCTRAQuest, HSPICE, ICX, and Quad software
- Originated in the early 1990s for promoting tool-independent I/O models for system-level SI design

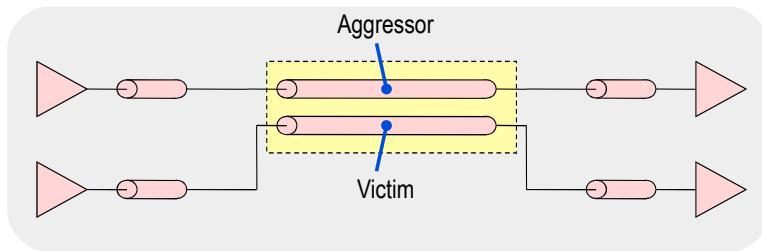


NOTES



Crosstalk (XTalk)

- Energy from one trace is transferred to an adjacent trace based on electro-magnetic coupling
- This transfer occurs during the switching process only
- Topology



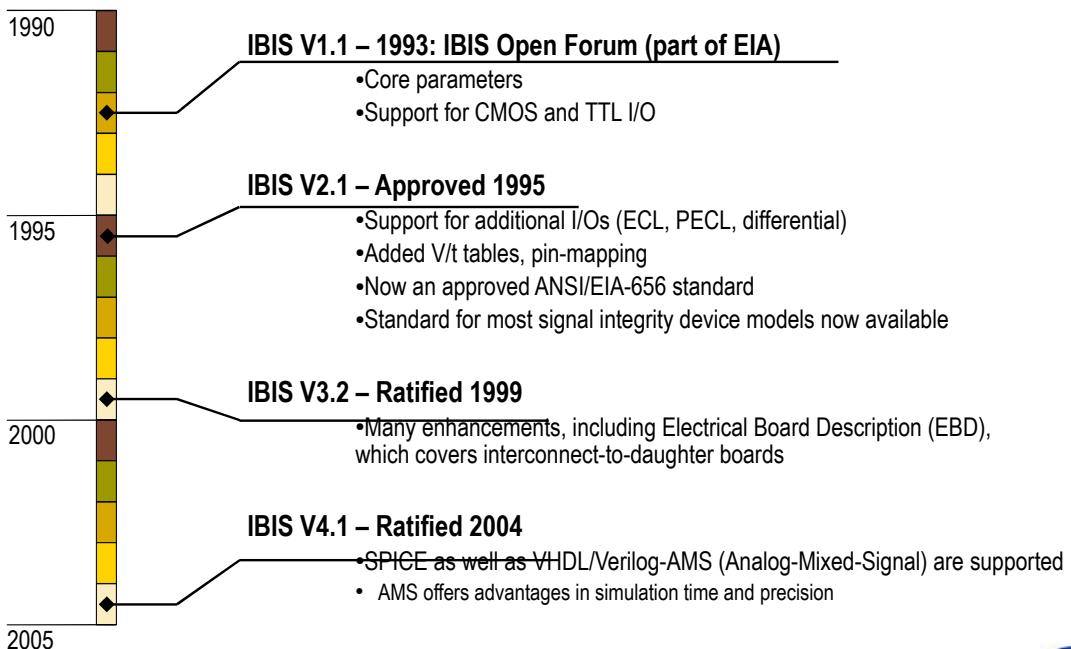
- Signal on the victim depends on
 - Coupling distance and coupling length
 - Impedance
 - Slew rate ($\Delta V / \Delta t$)



NOTES



Evolution of IBIS Standard



01_so_hssio_basics.odp Date Oct 23, 2009 Page 68

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

AMS is an improvement over IEEE1076. This simulation language is based on VHDL (or Verilog) extended by a description of time and value contiguous models. Additionally, it can describe digital as well as analog circuits.

Newer IBIS versions are:

- V4.2 (ratified 2006) adds more technical advances and a few editorial changes are documented.
- V5.0 (ratified 2008) adds more technical advances and a few editorial changes are documented.



IBIS: Good and Bad

- SPICE models were usually difficult to obtain
- IBIS models offer tool-independent I/O models for SI
 - IBIS files are really *not* models; they just contain the data that will be used by the behavioral models and algorithms inside the simulators
- IBIS simulates rapidly
 - Static characteristic curve
 - “Fast” simulation
- IBIS models on the Internet are error prone
 - Always need to check
- IBIS models come from either HSPICE derivations or IC measurements
- IBIS does not account for second-order effects



NOTES

Second-order effects involve the feedback of reflected signal energy into the driver that originally sent the wave out. The driver's parasitic or purposely designed capacitances can become re-biased, sending the driver into a different operating region. IBIS does not account for this effect. Furthermore, IBIS has limited capability to model staged I/O drivers that are implemented in the silicon.

SSTL and HSTL are generally well modeled. CML, LVDS, and LVPECL can be problematic. S parameters are not currently supported by IBIS.

What is an IBIS model?

- IBIS models represent the I/V characteristics and dV/dt for the best case, typical case, and worst case inputs and outputs. Simulations of complete I/O chains can describe printed circuit board transmission line effects. Using IBIS, you can predict, correct, and control ringing, bounce, and crosstalk.
- The Xilinx IBIS models are available at www.xilinx.com/support/download.



Test Equipment

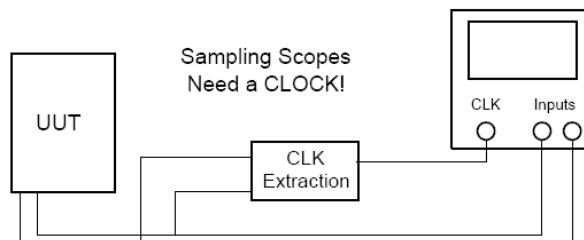
Digital Oscilloscope
Sampling Scope
Digital Communication Analyzer
Time Delay Reflectrometer
Bit Error Ratio TesterBERT
Can take a long time



NOTES



Scope



Sampling Scope: Digitizes the information and stores it. To capture signals faster than the analog-to-digital converters can go, the scope captures only a few samples of each period. Moving the sampling each time allows it to capture enough signals to represent a repetitive signal

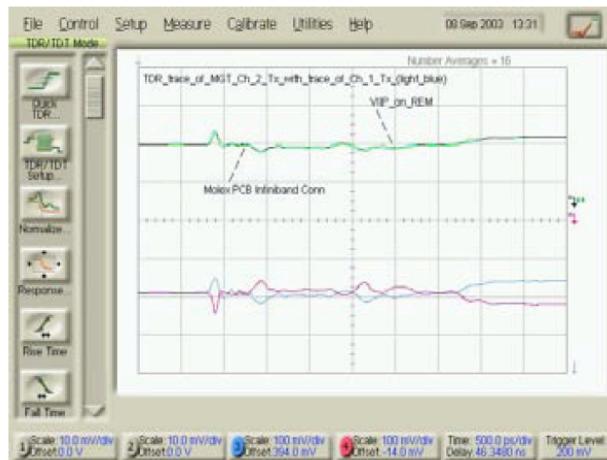
Digital Storage Scope: Converts the incoming signal to digital samples that are stored and then used to recreate the signal on a display

DCA: Digital communication analyzer; takes the sampling scope and adds a bunch of other features.

NOTES



TDR Time Delay Reflectometer



01_so_hssio_basics.odp Date Oct 23, 2009 Page 72

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

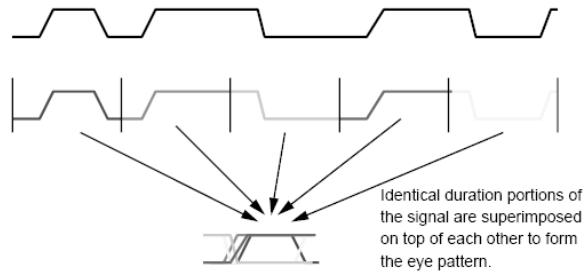


NOTES

One of the first things to do when we get our prototypes back is to go to the lab and check our transmission paths using a TDR. The TDR allows us to check our transmission paths for impedance increases. The smoother the path, the fewer problems. A TDR works by sending a pulse down the transmission path and measuring the reflections that come back. This determines the location and severity of the impedance discontinuity. A TDR and DSO module can be used together to create a TDT which looks at the pulse as it is received at the other end rather than the reflections. TDT can be useful for finding trace length mismatches



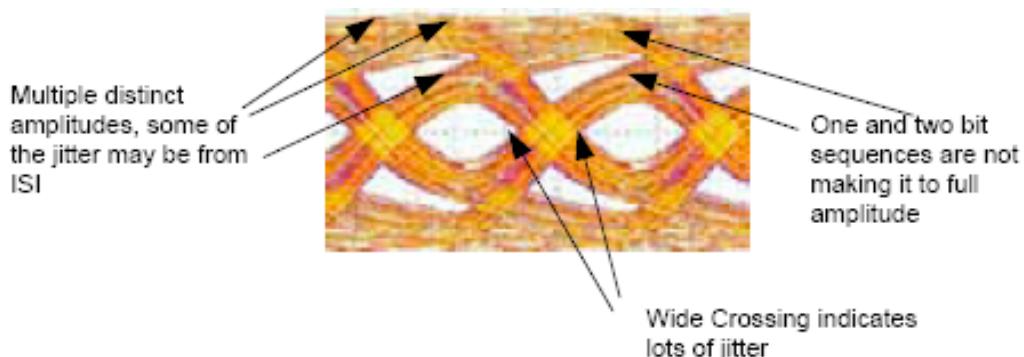
Eye Pattern



NOTES



Eye Pattern

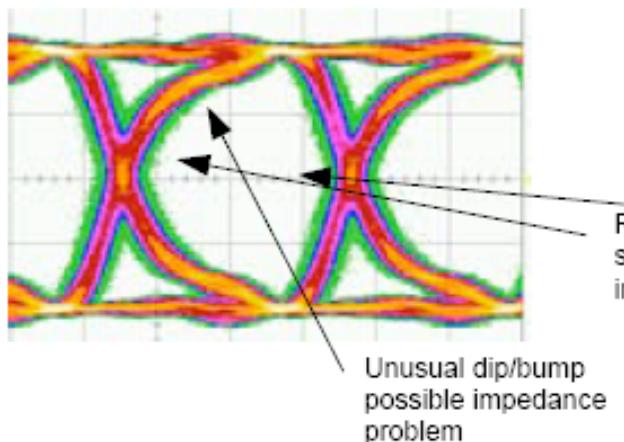


NOTES

By analyzing the eye pattern, we can discover much about the signal and the path it is traveling. The height and width of the eye shape correspond to the ability of a receiver to receive the signal. Often a receiver will have a published eye mask. If the eye pattern is within the mask, the receiver can detect the signal. The width of the fill (cross over) in between the eyes is a representation of the jitter of the system.



Eye Pattern



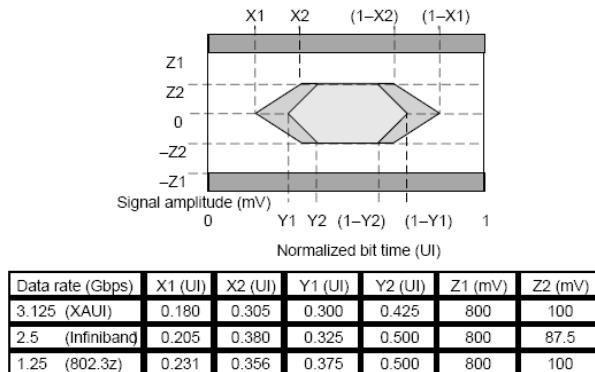
NOTES

Other details, such as too little or too much pre-emphasis and impedance mismatches that are not symmetrical on both sides of the differential pair can be identified through anomalies in the shape of the eye.

Another important aspect of eye patterns is color. Most modern equipment uses color as a way of signifying intensity. The darker or “hotter” the color, the more data samples have landed at that location. In this eye pattern , the orange shows many data samples (“hits”) and the green is just a few “hits.”



Eye Mask Pattern



NOTES

Another term associated with eye patterns is eye mask. This is simply a definition of how good or open an eye pattern needs to be for a receiver to operate correctly. An eye mask might look something like this figure.



Jitter

Jitter: The difference between the ideal zero crossing and the actual zero crossing.



NOTES

Mathematically, we could talk about jitter as a variation in the period of our signal. For example, if we had a sine wave clock, we could define a perfect zero jitter clock as: $\cos(w(t))$. Then a description of the jittery signal would be:

$$\cos(w(t) + j(t))$$

where $j(t)$ is a function describing the jitter. Jitter is often categorized into two types: deterministic and random:

- Random jitter: The component of the jitter resulting from differential and common mode stochastic noise processes such as power supply noise and thermal noise. Also known as rj , RJ , and called indeterministic jitter.
- Deterministic jitter: The component of the jitter attributable to specific patterns or events. Includes jitter resulting from sources such as asymmetric rise/fall times, inter-symbol interference, power supply feed through, oscillator wand, and cross-talk from other signals. Often abbreviated as DJ or dj .

When investigating why a multi-gigabit link is not working, the most likely problem is excessive jitter. It is a good idea to get a feel for how much jitter there is on the incoming signal of the receivers and compare that to the specification of the receiver. It is also a good idea to check the amplitude/eye height while we are looking. If everything looks acceptable, jitter is most likely not the problem.



Protocols

Data formats
Sub-channels
Data striping
Embedding
Errors detection and handling
Flow control
Addressing/switching/forwarding
Physical interface



NOTES

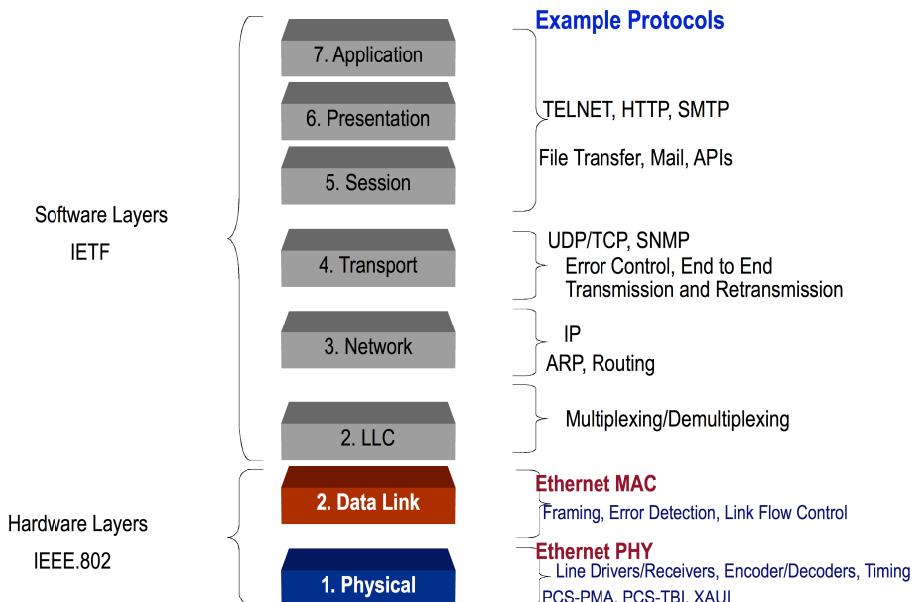
Data formats: Value definitions for video and audio protocols; how we use the ones and zeros to represent specific values or meanings.

- Sub-channels: Often there is a need for several different channels over the same link. Some of the common uses of sub-channels are control, status, and auxiliary data path.
- Data striping: A common function of a protocol is to define of how and where the data is separated from the overhead. This is commonly referred to as striping or de-embedding.
- Embedding: A protocol often defines how and where the data is embedded into the protocol streams or packets. This is especially true of protocols that follow the protocol stack model.
- Errors detection and handling: A protocol defines how errors are detected and what happens if there is an error.
- Flow control: Protocols may also define flow control. This can vary, from defining a way of dynamically scaling sub-channel bandwidth allocation to varying the idle insertion rate to match the clock correction needs.
- Addressing/switching/forwarding: While the direct point-to-point nature of a serial protocol eliminates many of the needs for an addressing scheme, some of the more complex protocols include addressing schemes. With addressing comes the possibility for forwarding and switching.
- Physical interface: Drive levels, pre-emphasis, and more, are specified by the protocol to ensure compatibility between devices.

Often the protocol choice is simple. When building a PCI Express card, simply run the PCI Express protocol. But when building a proprietary system, the system architect must decide whether to use a predefined protocol or design a custom protocol.



Ethernet in the OSI Model



01_so_hssio_basics.odp Date Oct 23, 2009 Page 79

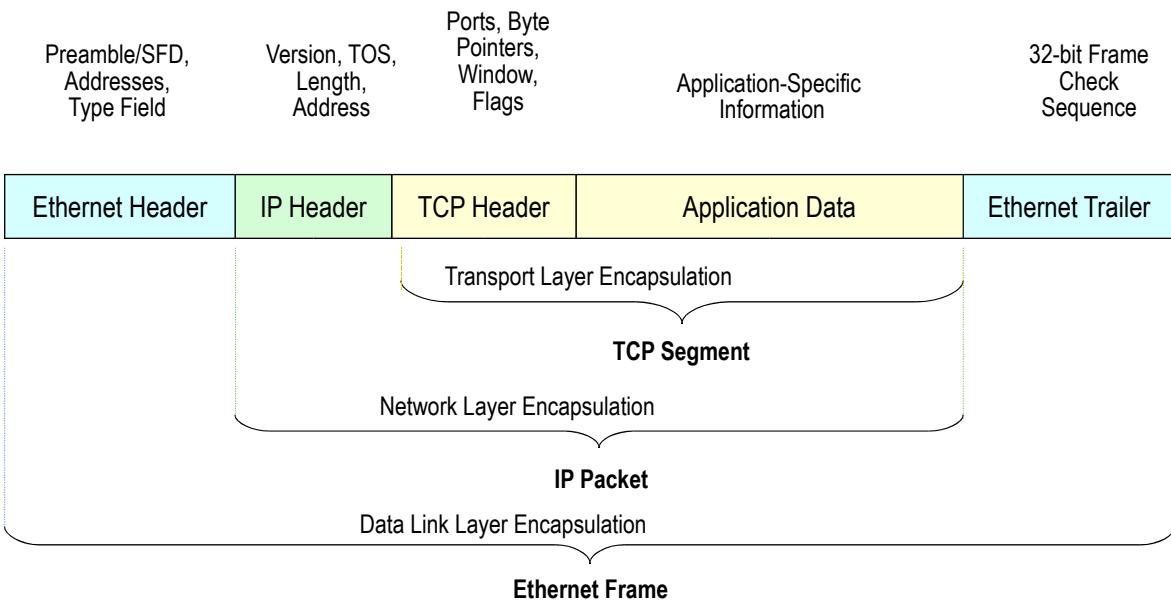
SO-LOGIC electronic consulting Austria & Brazil Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Network Protocols



01_so_hssio_basics.odp Date Oct 23, 2009 Page 80

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

This diagram shows the successive encapsulation of the layers of TCP/IP/Ethernet networking protocols. The innermost structure is the TCP segment, which is enveloped in the IP packet, which is in turn enveloped in the Ethernet frame.



Standard Protocols

XAUl

A 4-channel interface (2.5 Gb/s payload, 3.125 Gb/s wire speed) for 10-Gigabit Ethernet.

PCI Express

Gen1 2.5 Gbit/s, Gen2 5 Gbit/s, Gen3 8 Gbits/s

Serial RapidIO

Another serial version of an older parallel spec,

FiberChannel

FiberChannel has always been a serial standard, but its speeds have increased over the years..

Infiniband

A box-to-box protocol run over either copper or fiber.

Aurora

Open protocol from Xilinx for use with MGTs



NOTES

XAUI: A 4-channel interface (2.5 Gb/s payload, 3.125 Gb/s wire speed) for 10-Gigabit Ethernet. PCI Express: Takes the old parallel PCI structure and updates it to a high-speed serial structure. Upper levels of the protocol remain compatible, providing an easy adaptation into legacy PCI systems.

Serial RapidIO: Another serial version of an older parallel spec, RapidIO is quite flexible and sometimes used as a method of interfacing to multiple protocols such as PCI and Infiniband.

FiberChannel: FiberChannel has always been a serial standard, but its speeds have increased over the years. As copper interconnects have advanced, it has also become available on copper as well as fiber optics.

Infiniband: A box-to-box protocol run over either copper or fiber. Infiniband-style cables have become highly popular for multi-gigabit links of a few meters range. The specification allows for a variety of devices and complexity, and includes specifications for repeaters, and switches or hubs to

Aurora is a relatively simple protocol that handles only link-layer and physical issues. It has been designed to allow other protocols such as TCP/IP or Ethernet to ride easily on top of it. It uses one or more high-speed serial lanes



Ethernet

- What is Ethernet?
 - Communications protocol
 - Designed for networking
 - System-to-system communication
 - Not traditionally considered reliable
 - Adapted beyond system-to-system communication
 - Board-to-board
- GTPs are used for Gigabit / 10 Gigabit Ethernet (GE / 10GE)



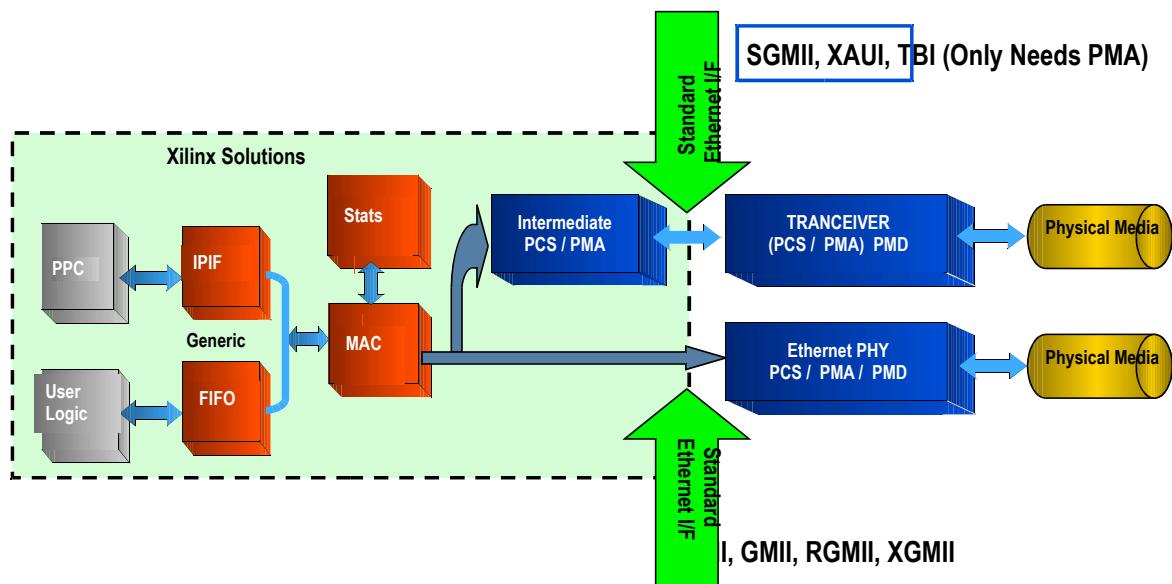
NOTES

Specifications:

- GE: IEEE 802.3-2002 – IEEE 802.3 Ethernet Working Group
- 10GE: IEEE 802.3ae-2002 – IEEE 802.3 Ethernet Working Group



Tri-Mode Ethernet Systems Overview



01_so_hssio_basics.odp Date Oct 23, 2009 Page 83

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Many interfaces in Ethernet:

- Some are for a parallel interface – MII, GMII, RGMII, and XGMII.
- Some are for serial interfaces – SGMII, XAUI, and TBI.



Ethernet MAC Responsibilities

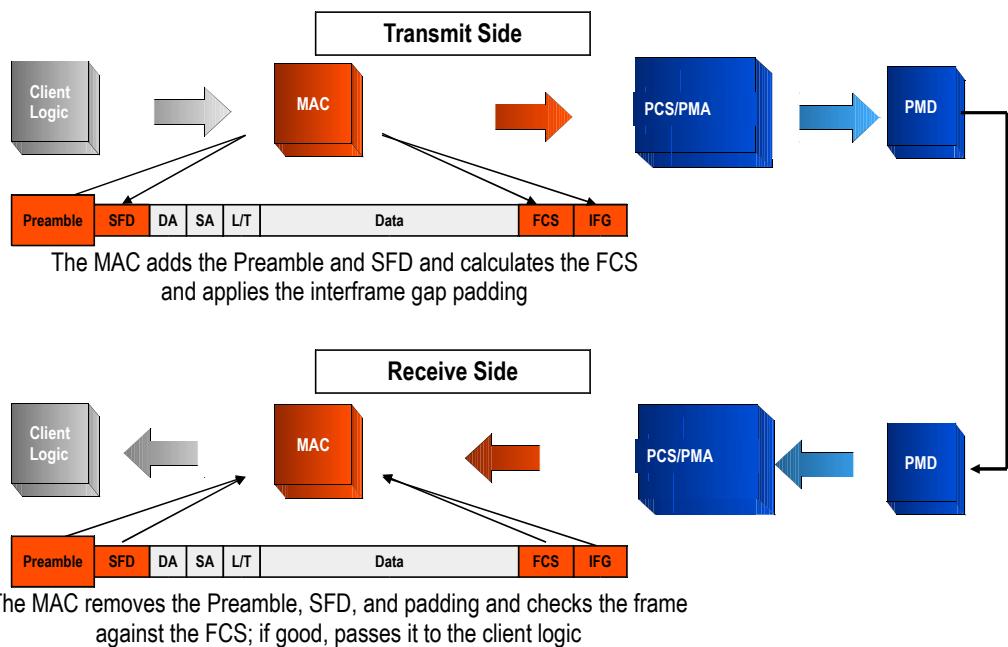
- Transmission
 - Package the Ethernet frame and communicate with the physical layer for the correct interface
 - Handle flow control (pause frames) and collisions
 - Generate and append the Frame Check Sequence (FCS) on the Ethernet frame
 - Handle the timing of the interframe gap and back off
- Reception
 - Receive and extract the Ethernet frame
 - Check the destination address and ignore the frame if it is not for this device
 - Can be set to accept all frames (promiscuous mode)
 - Check the FCS and protocol for errors



NOTES



Packet Transport



01_so_hssio_basics.odp Date Oct 23, 2009 Page 85

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



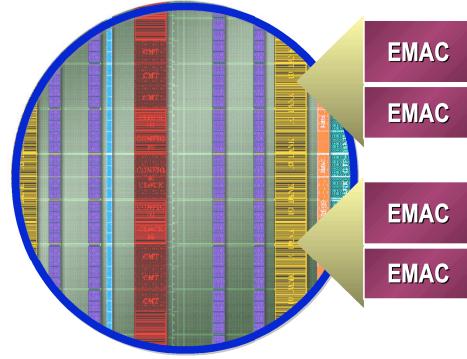
NOTES

The diagram illustrates the process of sending and receiving a frame.



Virtex-5 FPGA Tri-Mode EMAC

- Tri-mode 10/100/1000 Mb/s – full or half duplex
- IEEE 802.3 compliant
- Four integrated EMACs per chip
- Receive address filter
- Supports VLAN and Jumbo frames
- Programmable PHY interfaces
 - Compatible with SelectIO™ and serial I/O interfaces



Available in all LXT/SXT devices

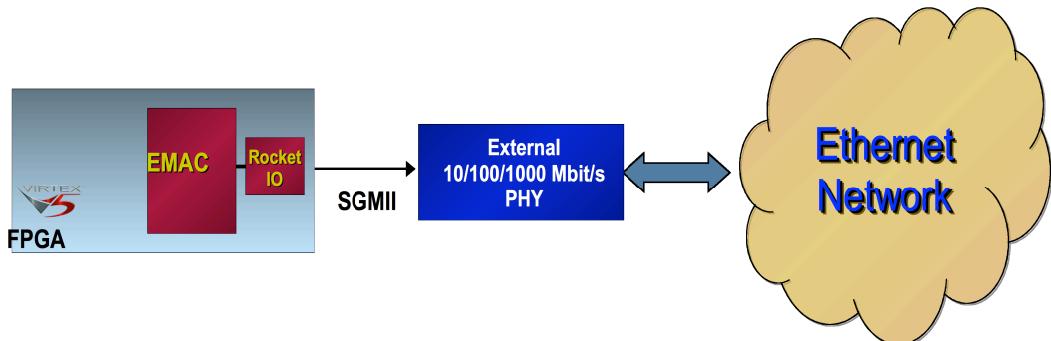


NOTES



Physical Interface

- A GTP can be used to connect to external PHYs through SGMII



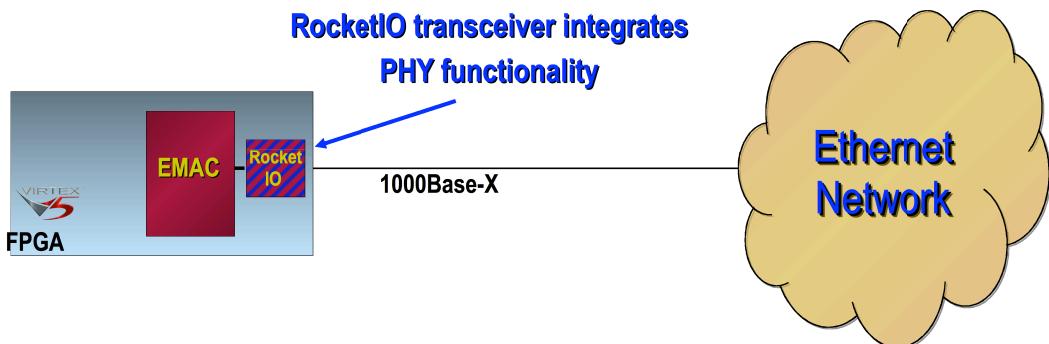
NOTES

The MGT can be used to connect to an external PHY through the SGMII interface.



Physical Interface

- The Virtex®-5 FPGA Ethernet MAC connects directly to the 1000-Mb/s Ethernet network using one of the available RocketIO transceivers



NOTES

The Virtex-4 FPGA EMAC can also connect directly to the 1000-Mb/s network using the RocketIO MGT, which integrates PHY functionality. It does not require an external PHY.



Ethernet IP Cores

- 10-Gigabit Ethernet solution designed to IEEE 802.3ae
 - Single-speed, full-duplex 10 GE MAC
 - XGMII interface and XAUI interface
- TEMAC and Gigabit Ethernet solution designed to IEEE 802.3-2000
 - Full-duplex GE MAC with PCS/PMA
 - Full-duplex or half-duplex GE MAC with GMII
 - Full-duplex GE MAC with PCS/Ten-Bit Interface (TBI)
 - Includes processor interface for use in the Embedded Development Kit (EDK)
- Legacy 10/100 Ethernet solution designed to IEEE 802.3
 - Full-duplex or half-duplex EMAC with MII
 - Includes processor interface for use in the EDK

01_so_hssio_basics.odp Date Oct 23, 2009 Page 89

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

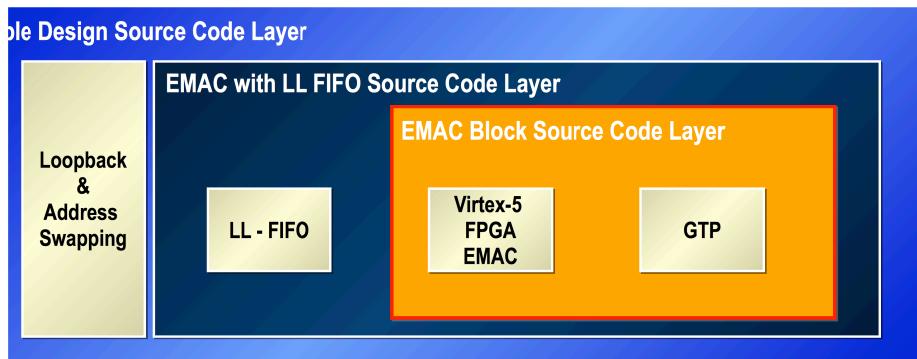


NOTES



TEMAC Wrapper: Ease of Use

- Enhanced hierarchy for easier design insertion
- Added Local Link FIFO
- Updated CORE Generator tool GUI and user guide



01_so_hssio_basics.odp Date Oct 23, 2009 Page 90

SO-LOGIC electronic consulting Austria & Brazil Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

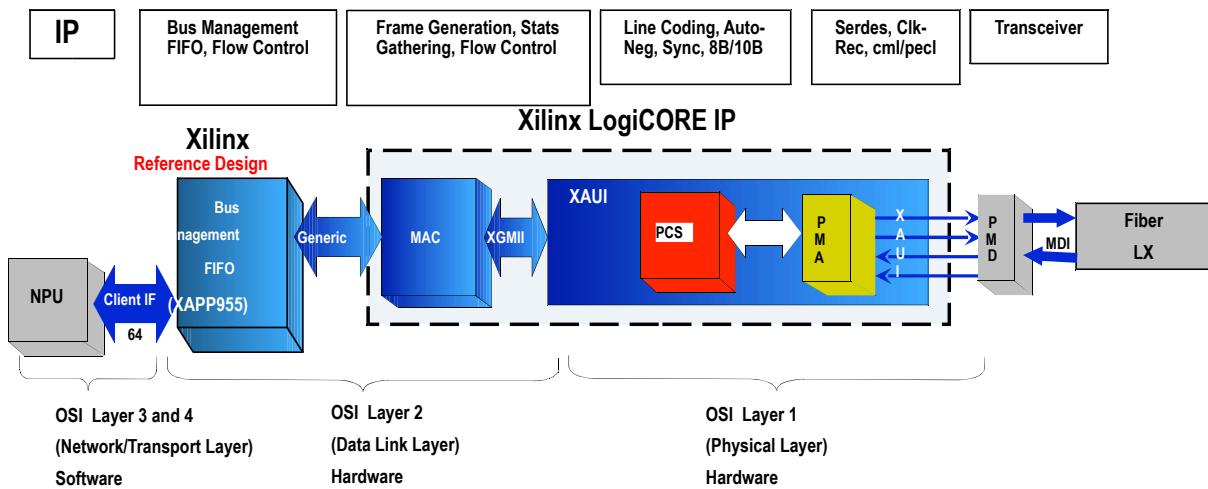


NOTES

LL_TEMAC is the Local Link TEMAC used by EPD; that is, it includes a TOE offload engine.



10-Gigabit Ethernet System



01_so_hssio_basics.odp Date Oct 23, 2009 Page 91

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



10GE MAC Core

- Designed to 10-Gigabit Ethernet specification IEEE 802.3ae-2002
- Choice of external XGMII or internal FPGA interface to PHY layer
- Supports deficit idle count for maximum data throughput; maintains minimum interframe gap under all conditions and provides line-rate performance
- Configured and monitored through a microprocessor-neutral management interface
- Comprehensive statistics gathering with statistic vector outputs
- Supports flow control in both directions
- MDIO STA master interface to manage PHY layers
- Extremely customizable; trade resource usage against functionality
- Available under SignOnce license program
- Supports VLAN, jumbo frames, and WAN mode
- Custom preamble mode
- Delivered through the CORE Generator tool



NOTES



Interlaken

- Interlaken is a highly scalable chip-to-chip protocol defined by Cisco and Cortina Systems as a follow on to SPI-4.2
- Interlaken features SERDES-based links constructed from 1 to n RocketIO transceivers running at speeds from 3.125 Gb/s to 6.25 Gb/s
- Interlaken features efficient encoding, robust error protection, and high reliability hardware attributes
- Interlaken is designed to handle future high-speed (10 Gb/s, 100 Gb/s and beyond) network connections



NOTES

Specification:

- www.interlakenalliance.com

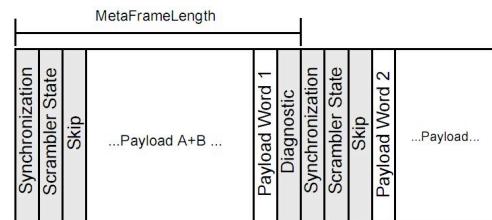
White paper:

- www.siliconlogic.com



Interlaken Functionality

- Functionality
 - Striping data for low latency
 - Eight-byte words are distributed to all lanes
 - Two modes for operation
 - Non-interleaved transfer (packet-by-packet transfer)
 - Interleaved transfer (fragments of multiple packets)
 - Flow control for backpressure
 - Data integrity using CRC24 over 256 bytes
 - Detection if single, double, triple, quadruple, and all odd errors
 - Meta frames
 - Synchronization, scrambling, and diagnostics, for example
- Interlaken frame structure
 - Data is packed into frames
 - Frames have base words with 67 bits



01_so_hssio_basics.odp Date Oct 23, 2009 Page 94

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



PCI Express Technology Performance

- Transmission rate is 2.5 Gb/s per lane per direction
- Numbers represent bidirectional aggregate throughput of link bits
- Payload data throughput will be lower
- x4 example $[2.5 \text{ Gb/s} * 2 * 4] / [10 \text{ bits/byte}] = 2 \text{ GB/s}$

PCI Express® Technology Link Width	x1	x2	x4	x8	x12	x16	x32
Aggregate Bandwidth (GB/s)	0.5	1	2	4	6	8	16
PCI™ Technology Rough Equivalent	PCI 64/66	PCI-X 64/133	PCI-X 64/266	PCI-X 64/533	N/A	N/A	N/A

01_so_hssio_basics.odp Date Oct 23, 2009 Page 95

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

The PCI Express standard uses 8B/10B encoding, meaning for every 1 byte (8 bits) of data, 10 bits are actually transmitted, which results in 25 percent additional overhead per byte transferred.

The above numbers are found as follows:

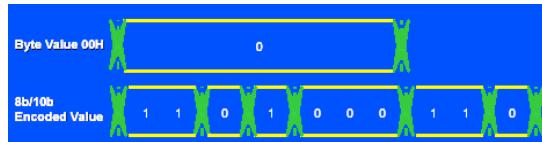
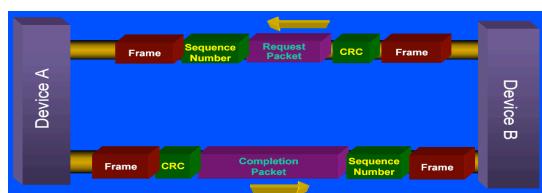
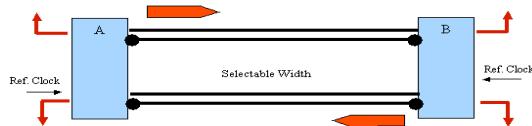
- Aggregate bandwidth = $[2.5 \text{ Gb/s} * 2 \text{ (for each direction)} * \text{Lane Width}] / [10 \text{ bits/byte}]$ (for 8B/10B encoding)
x4 example:
 - $[2.5 \text{ Gb/s} * 2 * 4] / [10 \text{ bits/byte}] = 2 \text{ Gb/s}$

The actual payload data transfer will be lower due to packet overhead, flow control, and other link maintenance-type DLLPs and ordered sets being transferred.



PCI Express Technology

- Differential low voltage
- Point-to-point dual simplex
- Packetized split transaction
- Embedded clock (8B/10B)
- PIPE (Phy Interface PCI Express)
 - Gen 1 2.5 GB
 - 250-MHz, 8-bit interface



01_so_hssio_basics.odp Date Oct 23, 2009 Page 96

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

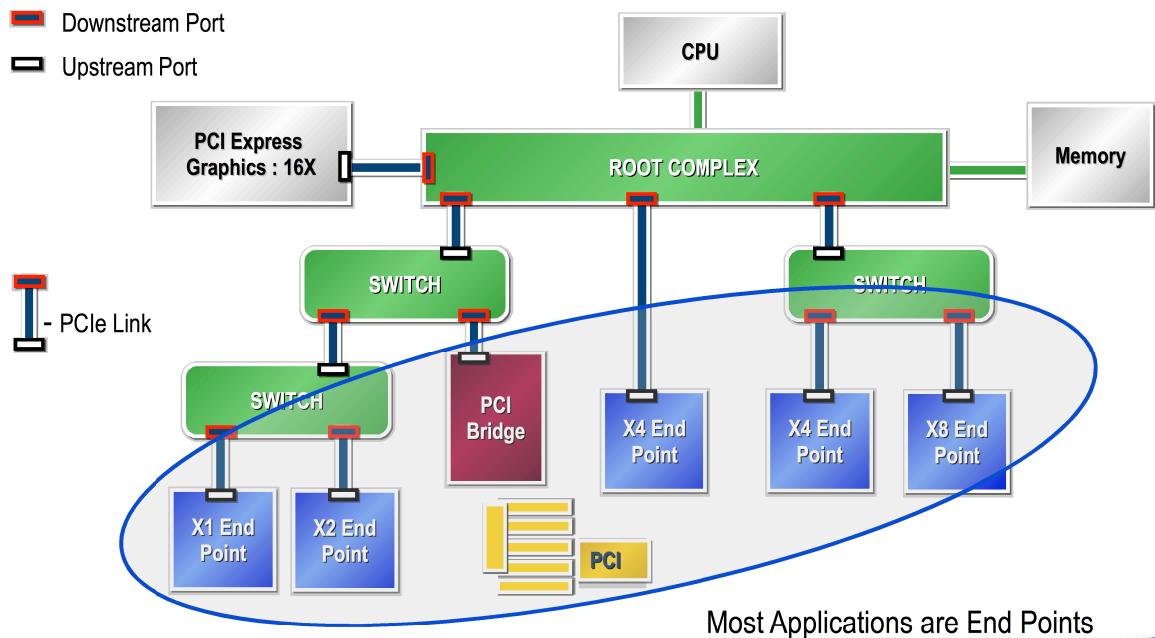
The basic technology is low-voltage differential signaling, which provides minimized signals on the connector. It is point-to-point dual simplex.

Unlike the PCI standard, read and write transactions can be performed simultaneously. It is a packetized split transaction, a non-blocking architecture. For example, for a read from memory, there is a completion for that read, and then sometime later you will get the data and you are not blocking the channel.



Topology

- Downstream Port
- Upstream Port



01_so_hssio_basics.odp Date Oct 23, 2009 Page 97

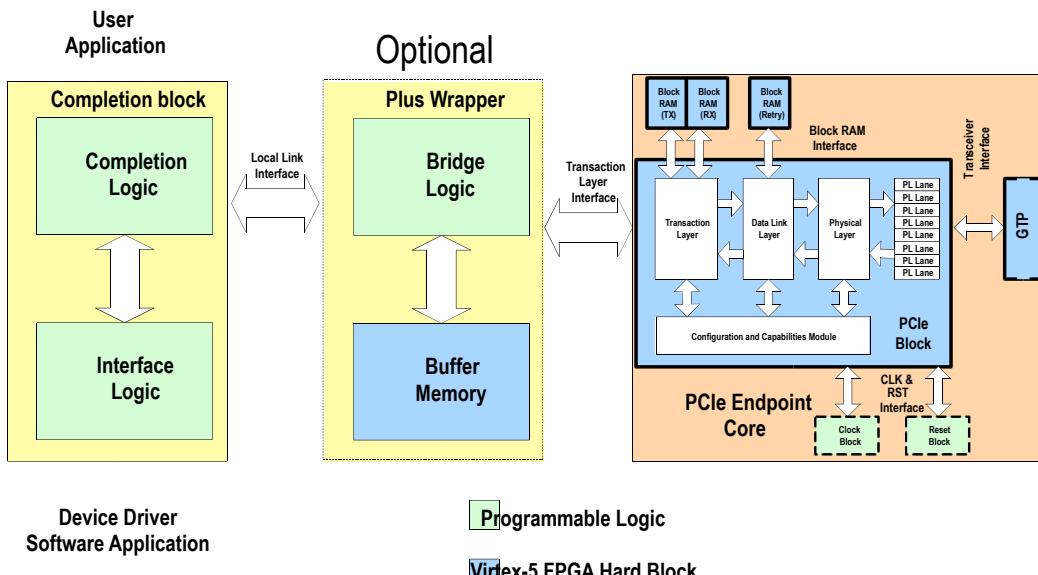
SO-LOGIC electronic consulting Austria & Brazil Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



FPGA Block PCIe Endpoint Block Diagram



01_so_hssio_basics.odp Date Oct 23, 2009 Page 98

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



What is Aurora?

- An open communication standard from Xilinx
- An area-efficient, scalable, data-transfer protocol for high-speed serial links
 - Specification is freely licensed
 - Available from www.xilinx.com/aurora
 - Bus Functional Model (BFM) for simulation is included
- Xilinx Aurora LogiCORE™ IP implementation
 - Free core; HDL licensed for Xilinx FPGAs
 - Fully parameterized across feature and device set



NOTES



Aurora Features Overview

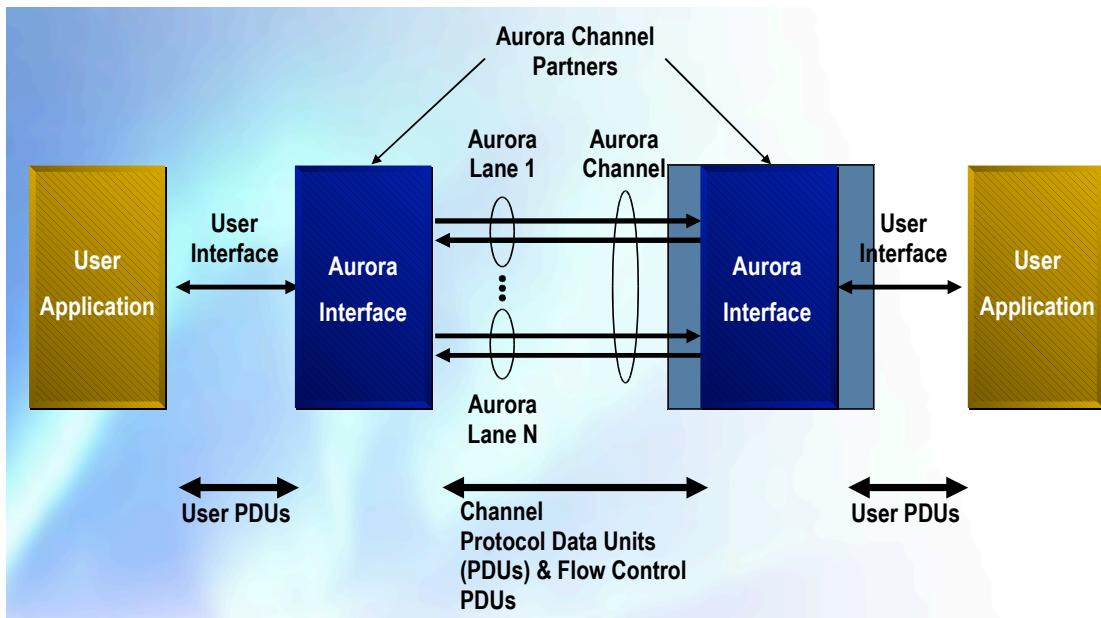
- Designed for small footprint implementations
- A scalable, high-speed serial, link-level interface
 - Common protocol for single and multi-lane channels
 - No protocol restriction on the number lanes
 - Serial full duplex or serial simplex operation
 - System-synchronous or asynchronous operation
 - Arbitrary data transfers: packets or words
 - Optional flow control and expedited messaging
- Aurora is available in two flavors
 - Aurora 8B/10B
 - Aurora 64B/66B



NOTES



Aurora Lanes and Channels



01_so_hssio_basics.odp Date Oct 23, 2009 Page 101

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

The Aurora protocol describes the transfer of user data across an Aurora channel. An Aurora channel consists of one or more high-speed serial links. The devices that communicate across the channel are called channel partners.

The Aurora interfaces transfer data to and from user application functions by way of the user interface. The user interface is not part of the Aurora protocol specification. Data flow consists of the transfer of user Protocol Data Units (PDUs) between the user application and the Aurora interface, and the transfer of channel PDUs across the Aurora channel. The format of user PDUs is user defined.



Aurora Encoding

- Data in an Aurora channel is encoded
 - 8B10B: Each byte mapped to 10-bit codes—25% encoding overhead
 - Additional overhead is required for framing, clock correction, and channel bonding, for example
 - 64B66B: Eight-byte blocks mapped to 66-bit blocks—3% encoding overhead
 - Additional overhead is required for framing, clock correction, and channel bonding, for example
- Encoding provides
 - Transitions for PLL lock
 - Control characters
 - Limited bit-error detection
 - DC balance and reduced data-dependent jitter



NOTES



Aurora Features

- Initialization

- Before channel partners exchange PDUs, the channel must be initialized
 - Lane initialization: Transceivers are reset and aligned
 - Lanes are bonded for multi-lane channels if necessary
 - Channel verified to check for correct operation and allow for post-bonding word alignment

- Clock Correction

- Aurora includes Clock Correction (CC) to allow transceivers to use separate reference clocks
 - Allows 200-ppm difference between reference clocks
 - Aurora receivers include an elastic buffer

- Flow Control

- Aurora supports two types of flow control
 - Native Flow Control (NFC): link layer rate control
 - User Flow Control (UFC): high-priority messages



NOTES



OBSAI

- OBSAI: Open Base Station Architecture Initiative
 - Provides the architecture, function descriptions, and minimum requirements for integration of a set of common modules into a Base Transceiver Station (BTS)
 - Hyundai, LGE, Nokia, Samsung, and ZTE
 - Reference points in the model
 - RP1 - Control to RF/Processing/Transport Modules (100 Mb/s)
 - RP2 - Processing Module to Transport Module (1000 Mb/s)
 - RP3 - Processing Module to Radio Module (768, 1536, 3072 Mb/s)
 - The OBSAI Reference Point 3 (RP3) specification defines the interface between the Baseband module and the RF module of a base station
 - The specification allows for line rates of $i \times 768$ Mb/s; $i=1, 2, 4$
 - Uses an optical link between the Baseband Module and the RF module
- Specification
 - www.obsai.org



NOTES



OBSAI LogiCORE IP Features

- Designed to the OBSAI RP3 v4.0 spec
- Operates at 768, 1536, or 3072 Mb/s
- Supports CDMA, WCDMA, WIMAX, 802.16, and LTE
- Supports RP3-01
 - Modular design concept
- Implements full data link and physical layer functions
- Supports both master or slave operation
- Supports auto-negotiation and daisy chaining
- Supports Virtex-5 *XT families
- Available



NOTES



CPRI

- CPRI: Common Public Radio Interface
 - Defines the interface of base stations between the Radio Equipment Controller (REC) to local or remote radio units, known as Radio Equipment (RE)
 - Ericsson, Huawei, NEC, Nortel, Siemens
 - Processing module to Radio module (614, 1228, 2456 Mb/s)
- Specification
 - www.cpri.info



NOTES



CPRI LogiCORE IP Features

- Designed to the CPRI V2.1 spec
- Operates at 614.4, 1228.8, or 2457.6 Mb/s
- Suitable for Radio Equipment Controller (REC) or Radio Equipment (RE)
- Supports multi-hop systems
- Customizable to support 1–24 antenna carriers
- Implements full data link and physical layer functions
- Supports both master or slave operation
- Supports auto-negotiation
- Supports Virtex-5 *XT families
- Available



NOTES



Serial RapidIO

- RapidIO is a packet-switched interconnect standard for
 - Connecting chips on a PCB
 - Connecting PCBs to each other using a backplane
- Application domains are embedded systems, primarily for the signal processing, networking, and communications markets
- RapidIO trade association
 - Alcatel-Lucent, AMCC, EMC Corporation, Ericsson, Freescale, GDA Technologies, Mercury Computer Systems, PMC-Sierra, Texas Instruments, and Tundra
- Specification
 - www.rapidio.org/home

01_so_hssio_basics.odp Date Oct 23, 2009 Page 108

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Protocol Overview

- Data packets provide a logical transaction interface between endpoints
 - Example: read, write, message
 - Packets contain a transport address (source and destination) that allows the packet to be routed through the fabric
- Control symbols provide link reliability and management
 - Example: acknowledge, retry, end of packet
 - Control symbols provide the basis for hardware-based error recovery
 - Control symbols can be embedded within packets
 - Generate and terminate in the physical layer; 32 bits



01_so_hssio_basics.odp Date Oct 23, 2009 Page 109

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Serial RapidIO LogiCORE IP

- Capability for x1 and x4 lane Serial RapidIO bus links
- Support for 1.25, 2.5, or 3.125 Gb/s
- Complete endpoint includes
 - Physical layer LogiCORE IP
 - Logical (I/O) and transport layer LogiCORE IP
 - Buffer and register manager reference design
 - Approximately 20 percent less LUT utilization than Virtex-4 FPGA platform
 - Approximately 6.4K LUTs in the Virtex-5 LXT family)
- Compliant with Serial RapidIO Specification v1.3
- Delivered via the CORE Generator tool

01_so_hssio_basics.odp Date Oct 23, 2009 Page 110

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Custom Protocol



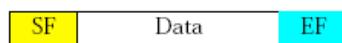
Transmitter

Takes Incoming parallel stream and places it in packets of 2048 bytes.
Uses an MGT to 8b10b encode, serialize data, and drive on wire.
Wire speed is 2.5 Gbits/sec
Inserts variable number of Idle characters between packets

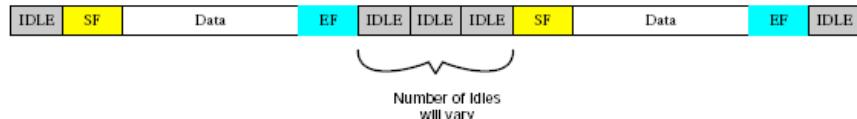
Receiver

Takes Incoming serial stream strips start flags, end flags and idles.
Converts 8 bit received data into 12bit data and stores in a fifo.
Data is read from fifo at 150MHz

Basic Packet Structure



Representation of multiple packets on wire



NOTES

There may be times when you want to define your own protocol. This would most often make sense when the standard protocols do not fit your needs, and/or is too extensive for your application. Of course, there are also times when a new complex protocol is needed, but that is most often left to committees of people who are experts.

Some of the things to consider when defining your own protocol are best illustrated by looking at a simple example. In your sample application we need to transfer a constant 1.8 GHz stream from one board to another. The data in and out of the system will be a 12-bit bus, changing at 150 MHz. With this simple requirement, all you really need from a protocol is a definition of a data frame, alignment, and idle character. In this example, we will use 8b/10b as the line-encoding scheme and borrow from other 8b/10b standards for our markers and comma choices.

Once we define a character or ordered set of characters for sf (start of frame), ef (end of frame) and idle, we need to determine line speed and data frame size. The size of the data frame should be chosen so that we can guarantee enough sf symbols to align to and idle symbols to provide for clock correction. In this case, since we need a data payload of 1.8 GHz, an easy choice is to run the wire at 2.5 Gb/s.

This gives us a “wire payload” of 2 Gb/s that we can use for our 1.8 GHz data needs, with excess capacity for our overhead needs. Since we are going board-to-board, we would definitely have different oscillators driving the transmit clocks of the transceivers, so we must account for clock correction. When considering how big to make the data frame, we need to balance two contradicting needs. The bigger the data frame, the less overhead and the more bandwidth available for data. The smaller the data frame, the more alignment and clock correction characters.



Custom Protocol

SF	K28.5, K28.5	8B10B Assignments
EF	K28.0, D0.0	
IDLE	K29.7, K28.5	
Data	2048 Bytes encoded into 2048 8b10b data symbols	

Alignment sequence = K28.5, K28.5 (SF) **MGT settings**

Clock Correction sequence = K29.7, K28.5 (IDLE)

Physical Details: Pre-emphasis, termination, AC or DC coupling, and amplitude can vary according to application.



NOTES

Clearly, we must balance the two needs. We could calculate both limits and pick something in the middle. Data handling capabilities are easy to calculate. We need 1.8 Gb/s out of 2 Gb/s available. So as long as the overhead bits fit into the available space, we will be fine. Now we must pick a convenient size and see if it works.

If the data frame holds 2048 bytes and the overhead is 10 bytes (2 for sf, 2 for ef, 6 for idles) then our over head rate would be 10/2058 or about 0.5%. Our available overhead is 0.2/2 or 10%. We could make the data frame 20 times smaller and still be within our overhead budget. We want to go smaller if necessary for clock corrections, so we need to look at that. Our particular MGT requires a reference clock with a 20 ppm accuracy. If we program the MGT to a

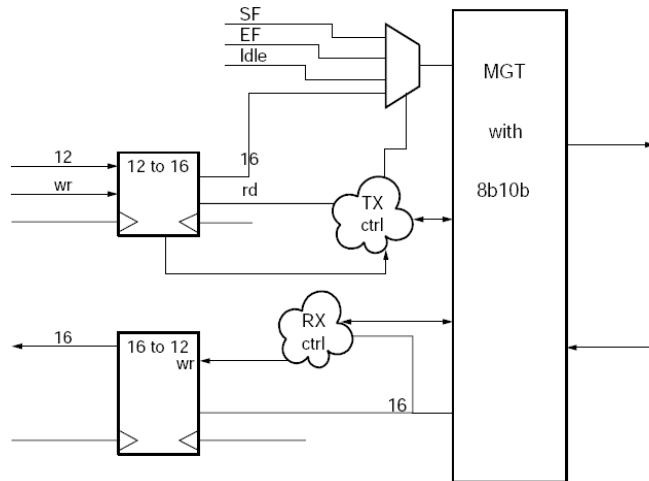
2-symbol-wide idle sequence, the maximum correction would be every 49,999 symbols. The distance between idle characters must be less than this. In most cases, we will want it to be smaller than 1/3 of the maximum distance between idles. We are about 1/24th, so we have our very simple protocol almost defined. The only thing we need to deal with is a flow control issue.

Remember how we were only using 0.5% of the time for overhead but had 10% available? We need to define what is going to fill that space and who manages the fill. So we need to fill that extra time with idles so the wire will look like the illustration in this figure.

Notice that the exact number of idles between frames will need to vary slightly depending on the reference oscillators. How this happens should be defined in the protocol, as should the stripping of idles. We can do this simply in the protocol by defining that it is the transmitter's responsibility to add enough idles to fill in the wire time, and that it is the receiver's responsibility to strip all idles, start of frames, and end of frames, from the incoming data stream.



Custom Hardware



NOTES

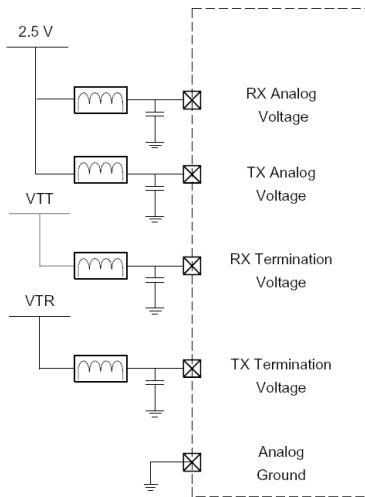
Notice that the exact number of idles between frames will need to vary slightly depending on the reference oscillators. How this happens should be defined in the protocol, as should the stripping of idles. We can do this simply in the protocol by defining that it is the transmitter's responsibility to add enough idles to fill in the wire time, and that it is the receiver's responsibility to strip all idles, start of frames, and end of frames, from the incoming data stream.

So that is all it takes to define a simple protocol in about a page. But what about implementation?

Most of the work will be done by an 8b/10b-enabled MGT. We will need to add a bit of custom logic to the interface, but no processor or software will be needed; not even a complex state machine.



Power Supply



NOTES

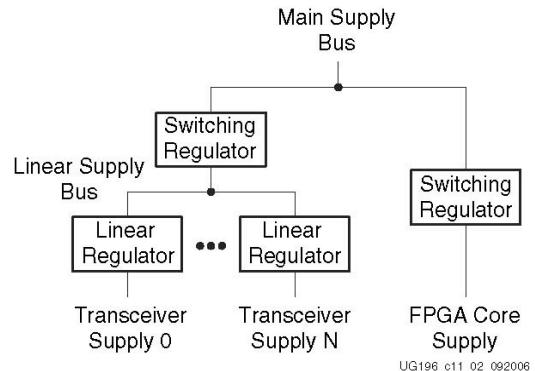
It is critical that both the analog transmit and receive power supplies, and the associated analog ground be extremely clean. As such, it is common for the MGT manufacturer to define that specific circuits to be used. This will almost always call for separate analog voltage regulators for each voltage, if not each MGT, and a passive power filter consisting of a capacitor and a ferrite bead.

In some MGTs (especially those in flip chip packages) the capacitor will be included inside the package of the part. In this case, often only the ferrite bead is needed. If a manufacturer recommends a specific circuit, it is normally best to follow the exact recommendations. One reason for this is that, in cases where multiple MGTs are in a common part, it is normal to require only a single linear regulator. And while we think of our filter circuit as filtering power supply noise from reaching our MGT, it also has some value in keeping the noise from one MGT from filtering to another MGT. The filter becomes both an input filter and an output filter. Sometimes a manufacturer will make a trade-off between input and output filter capabilities based on internal knowledge of how much output filtering is needed.



MGT Power Supply

- Use linear regulators
 - Better noise behavior
- Characterized by
 - Input voltage range
 - Output voltage range
 - Output voltage current
 - Output voltage tolerance
 - Output noise voltage
 - Power Supply Ripple Rejection (PSRR)



UG196_c11_02_092006

01_so_hssio_basics.odp Date Oct 23, 2009 Page 115

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

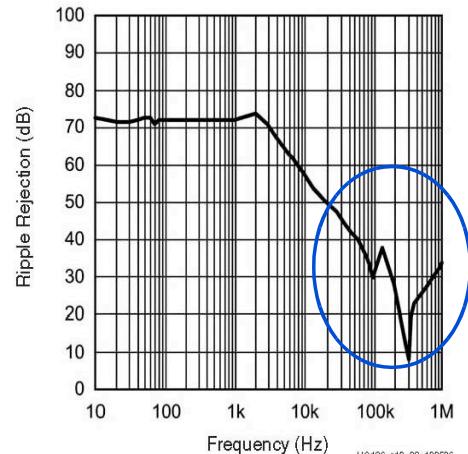
Noise can propagate from the FPGA core, other supplies, or transceiver-to-transceiver supplies. Using linear regulators helps to isolate transceiver supplies from noise sources.

Although PSRR is usually Power Supply Rejection Ratio, some authors (Linear, TI) use Power Supply Ripple Rejection Ratio.



Regulator Selection

- Must meet or exceed the characteristics given in the data sheet
 - Noise for MGTAVCC and MGTAVCCPLL is critical
- Ripple rejection must provide attenuation in a frequency range where sourcing power is noisy
- Additional filter network will be required in most cases
 - Ferrite – capacitor network



NOTES

The selection criteria for the linear regulator are:

- Meet or exceed the characteristics specified in the Virtex-5 FPGA data sheet.
- The PSRR of the linear regulator must provide attenuation in a frequency range where the sourcing power supply or regulator emits noise. Use an adjustable regulator so that the voltage can be changed, if necessary.

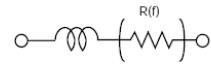
The PSRR over frequency of a linear regulator is temperature and load current dependent. Because the PSRR of the regulator in this example has a local minimum of rejection around 300 KHz, extra care must be taken if the sourcing power supply has spurs or high-amplitude noise in this frequency range.

If the sourcing power supply cannot be changed or a different power supply cannot be selected, an additional filter network between the output of the sourcing power supply and the input of the linear regulator is required to prevent substantial noise from passing through the linear regulator at the minimum of attenuation. Because the capacitor on the output of the regulator is part of the regulator control loop, this capacitor not only impacts the regulator stability but the PSSR as well.

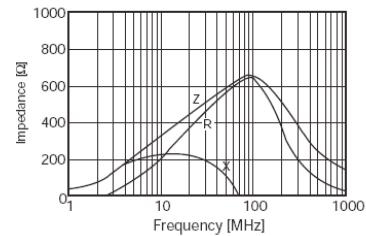


Ferrite Selection

- Criteria are
 - Low DC resistance
 - Maximum current
 - But do not max out the current rating
 - High impedance in a frequency range where you expect the highest spurs of noise levels



[Impedance-Frequency Characteristics (typical)]



R : Real Part (Resistive Portion) X : Imaginary part (Inductive Portion)

01_so_hssio_basics.odp Date Oct 23, 2009 Page 117

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES



Power Filtering: Why?

- Clock recovery
 - Clock Data Recovery (CDR) circuitry is analog; thus, very sensitive to noise
 - Too much noise and the clock cannot be recovered
- I/O timing
 - A noisy power supply translates directly to jitter
 - At 3.125 Gb/s, the eye opening is only 320-ps wide
 - Very little margin for error



NOTES

The power distribution system guidelines for the specific receiver must be strictly followed in order for the specified performance to be attained.



Power Filtering

- Need to isolate transceiver supplies from
 - Each other
 - Xilinx recommends the use of separate (adjustable) voltage regulators for each supply circuit until full characterization results are available
 - FPGA noise
 - System and board noise
- Accomplished two ways
 - Dedicated voltage regulators
 - Passive filtering networks
 - Ferrite beads
 - Ceramic capacitors (like bypass)



NOTES

This section focuses on the voltage regulators that directly source each dedicated filter network connected to one of the analog power supply pins of the GTP_DUAL primitive.

A voltage regulator is characterized by:

- Input voltage range
- Output voltage range
- Output voltage current
- Output voltage tolerance
- Output noise voltage
- Power Supply Ripple Rejection (PSRR)

These characteristics are the selection criteria when choosing a voltage regulator for a design with GTP transceivers. The output voltage noise and the PSRR over frequency are often neglected but are very important selection criteria.

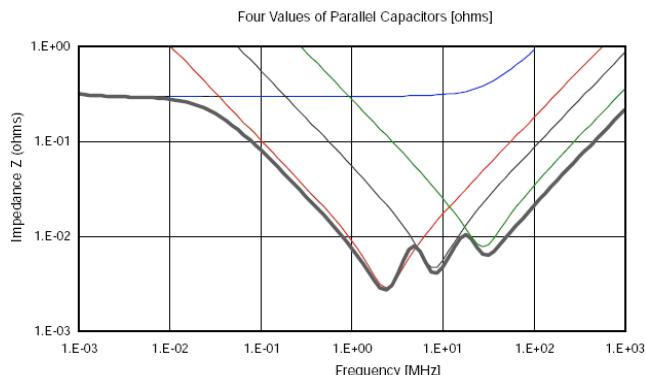
As a rule-of-thumb guideline, any substantial noise in the frequency range of 1 MHz and above on the power supply lines contributes to jitter. Depending on the frequency range and amplitude, this noise can degrade the overall system performance. The AVCC supply pins, which source the internal analog circuits of the transceivers, and AVCC_PLL, which sources the shared PMA PLL of the GTP_DUAL primitive, are especially sensitive to power supply noise.

When designing a complete Power Distribution System (PDS), the PSRR of the whole system and of each regulator is load current and frequency dependent.



Capacitor

- Criteria are
 - Low inductance capacitor
 - Dielectric material with a low-temperature coefficient
 - Dielectric material with a low-frequency coefficient
- Take care with capacitor placement
 - Reduce mounting inductance



01_so_hssio_basics.odp Date Oct 23, 2009 Page 120

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

In addition to the analog power supplies, the digital supplies must also be considered. Often the digital supplies for the MGT will be the common supply for all the digital logic of the devices. As with any switching circuit, bypassing is critical. But, at these speeds, we cannot just insert few capacitors and say the bypassing is complete. That approach used to work a few years ago, so why not now?

It still can work if we can find some ideal capacitors (no inductance or resistance) and get them on the board using ideal routes and vias (no inductance or resistance), and the package is ideal, and so on. As switching frequency and current needs have increased, the ESR and ESL that at one time could be ignored now have to be considered.

The goal of a power distribution and bypassing network is to be able to deliver the correct voltage in varying amounts of current. Bypassing circuits need to be designed to meet the specific needs of each application.

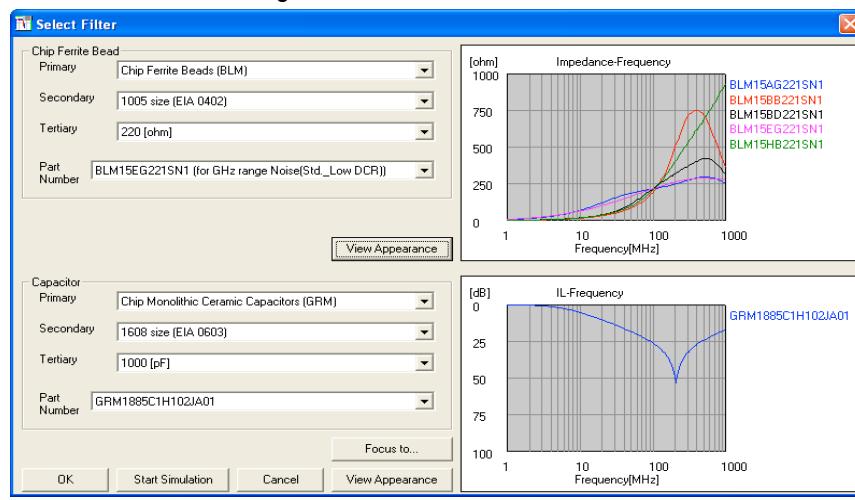
One method of analyzing this is to look at the impedance of our power system and its associated frequency. This figure shows the frequency response for three capacitors commonly recommended in standard applications. Notice two main problems. One problem is the large impedance spikes between the values. If our system happens to need power supplied in that frequency range, we will have a problem.

Notice two main problems. One problem is the large impedance spikes between the values. If our system happens to need power supplied in that frequency range, we will have a problem. Part of designing our bypassing circuit is to make sure these spikes are in areas not critical to our particular design. This can be accomplished by using different capacitors



Calculating a Power Supply Filter

- Example: Murata EMI Filter Selection Simulator
 - www.murata.com/designlib/mefss/index.html



01_so_hssio_basics.odp Date Oct 23, 2009 Page 121

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

The selection criteria for the filter network are:

- Place the filter network as close as possible to the device power pin.
- Ensure a low-inductance connection between the capacitor and the power pin.
- Simulate the filter circuit and optimize it if possible.
- Isolate the analog supply plane between the filter and the FPGA pin. Make sure that no signals can capacitively or inductively couple into this supply.



Power Supply Guidelines

- Regulator design guidelines
 - Use a linear regulator with low ripple
 - Consider the ripple rejection of the linear regulator
 - Ripple rejection depends on frequency and output load current
 - Check your power supply against noise in this frequency range
 - Do not operate the regulator with VIN that is just slightly over VOUT + VDROPOUT
 - Use the correct decoupling for the regulator
 - Data sheet: Value, ESR, dielectric for input, and output and bypass pins
 - Follow the layout design rules of the regulator data sheet
 - Do not maximize the output current
 - Place the regulator close to the filter network
 - Place the filter network close to the analog power supply pins



NOTES



Power Supply Guidelines

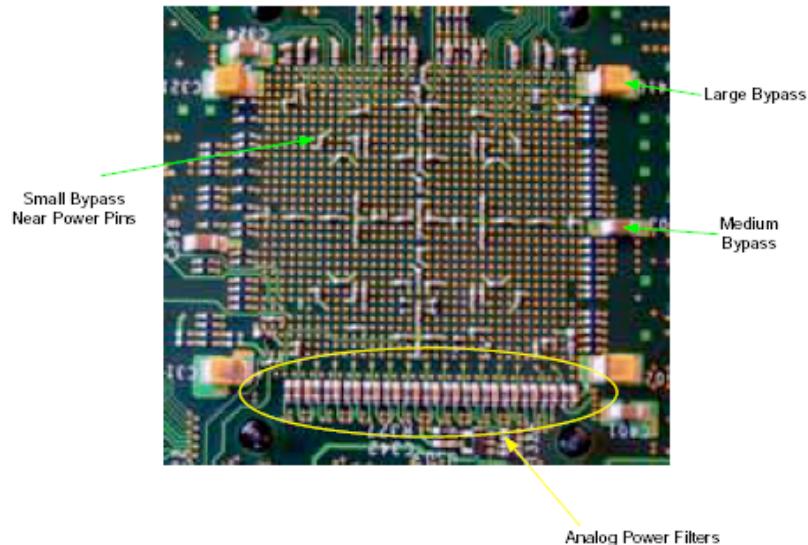
- Ferrite selection guidelines
 - Choose a ferrite with a low DC resistance
 - Do not work at the maximum ferrite current
 - Choose a ferrite with a high impedance in a frequency range where you expect or measure the highest spurs or noise levels
- Capacitor selection guidelines
 - Choose a low-inductance capacitor
 - Use a dielectric material with a low-temperature and low-frequency coefficient
- Filter network design guidelines
 - Place the filter network as close as possible to the analog power pins
 - Ensure low-inductance connection and mounting
 - Simulate and optimize the filter if possible



NOTES



Printed Circuit Board PCB



01_so_hssio_basics.odp Date Oct 23, 2009 Page 124

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Another important aspect of bypassing is placement. As a general rule, the larger the cap value, the less critical the placement. The smallest values want to go as near a power and ground pin as possible. One way to do this that is often available when using MGTS inside FPGAs is to remove the trace and via of unused general IO to make room for the bypassing.

Shielding

Any multi-gigabit signal needs to be isolated from interfering, and being interfered on, by other signals, whether the signal is on a board, cable, or going through a connector. This is accomplished by isolation and shielding with connectors and cables. On PCBs, multi-gigabit signals should be isolated from other signals by using extra space, and should be isolated from parallel traces on other layers by ground or power planes.



PCB Design

- Material selection
- Stack-up/board thickness
- Power and ground planes
- Differential pairs
- Differential trace width and spacing
- Vias
- Space between pairs
- Ground guards between pairs
- Power layout



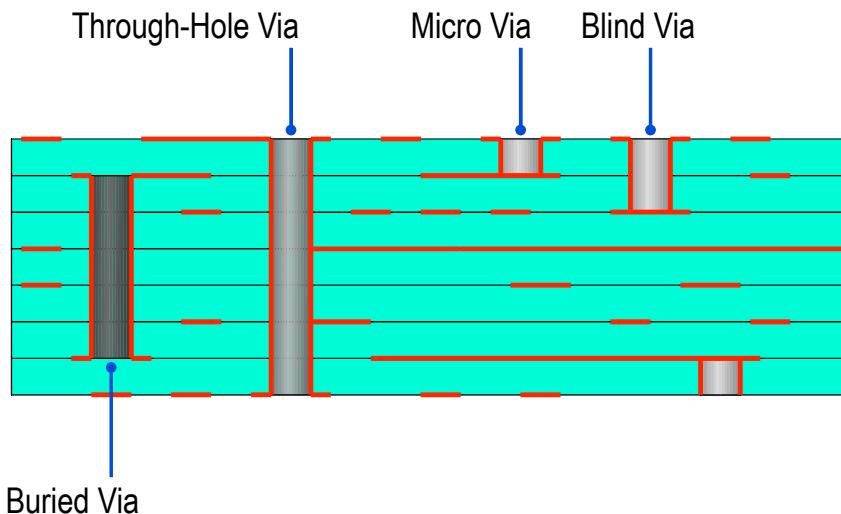
NOTES

Material Selection

While FR-4 has become the standard board material for a number of years, some lower loss alternatives have become readily available. A general guideline is that for total trace length less than 20 inches and speed at or below 3.125 Gb/s, FR-4 may be acceptable. If we need longer traces or faster speed, we should seriously consider using a high-speed material such as ROGERS 3450.



PCB Via Types



NOTES

The holes on a PCB can be classified as:

- Assembly holes: Fixing of the board and fixing of components on the board, for example; the diameter can be from 1.7 mm to about 10 mm.
- Holes for component pin: Fixing of components on the boards using through holes; the diameter can be from 0.6 mm to about 1.7 mm.
- Vias: Galvano-technical contact of several levels of the board; the diameter can be from 0.05 mm to about 0.5 mm.

The aspect ratio of a via is the ratio of board thickness to via diameter. The larger the aspect ratio, the more difficult it is to achieve reliable plating. Premium charge for aspect ratios > 8.

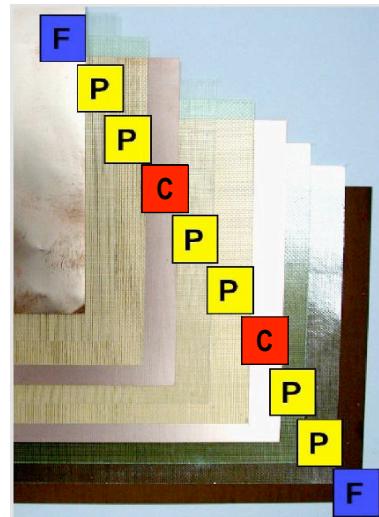
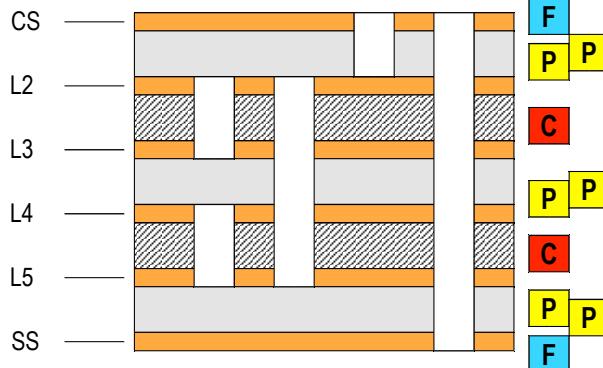
Notes to Facilitator:

Be sure to mention the aspect ratio; it is an important parameter.



Multi-Layer PCB

- Six-layer standard PCB with two inner laminates



01_so_hssio_basics.odp Date Oct 23, 2009 Page 127

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

CS: Component Side

SS: Solder Side

F: Foil

P: Prepreg

Stack-up/Board Thickness

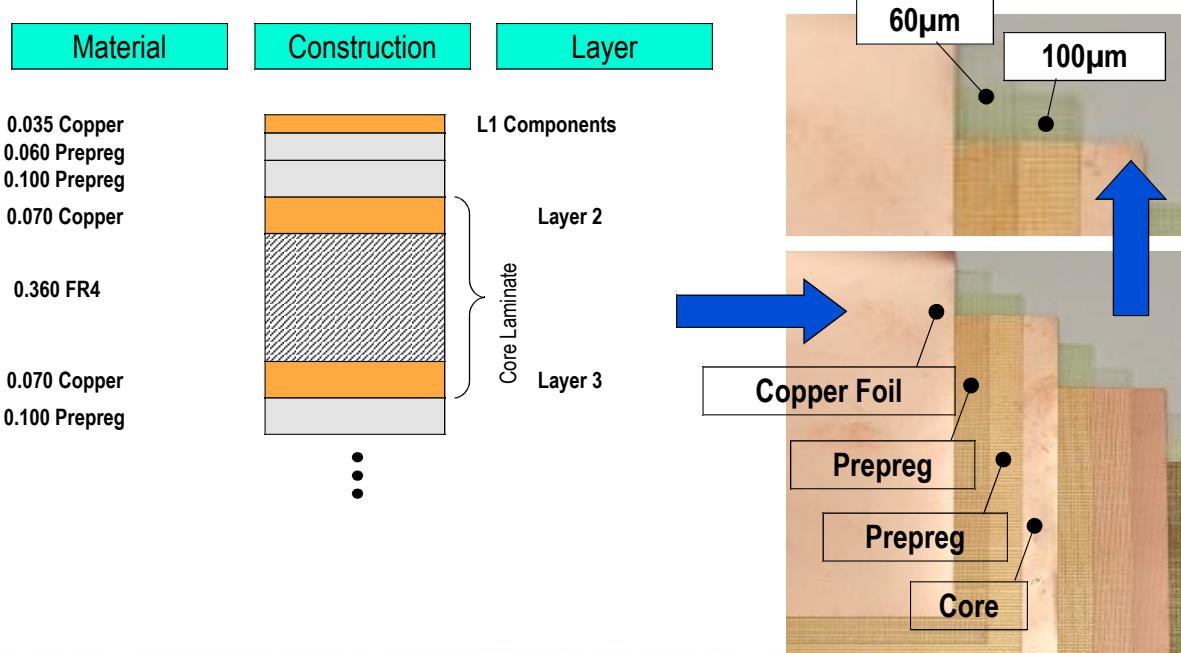
Once we have selected a material, the next step will be to devise a general stack-up plan. This may change as the number of signal layers is determined, but we will need to keep our stack-up in mind throughout the processes. Do not forget to add an adjacent power and ground plane layer to improve bypassing.

Power and Ground Planes

We need to think about how we are going to distribute all those special analog voltages. We may need to consider separate planes for each analog power. Isolating and filtering ground planes that are the reference plane for the multi-gigabit signals might be a good idea. We could also consider eliminating the digital power supply plane from signal areas that operate at less than gigabit speed.



Multi-Layer PCB



01_so_hssio_basics.odp Date Oct 23, 2009 Page 128



SO-LOGIC electronic consulting Austria & Brazil Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

NOTES

Generally, there are three materials:

- Conductor: Metallic copper in the form of thin foils, usually 17 µm and 35 µm (1/2 oz. and 1 oz.).
- Substrate: As an insulator, such as paper, glass, and ceramic.
- Binder: Mainly resin systems based on epoxy, polyester, and polimide.

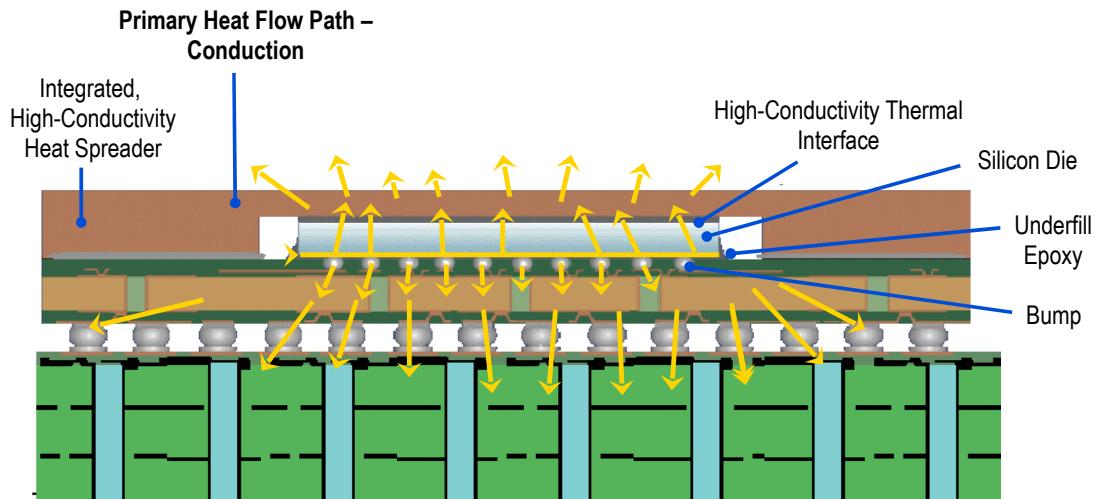
There are also “pre-constructed” materials available:

- Prepregs: Consisting of a substrate plus binder; typically fiberglass epoxy-resin (FR4).
- Core laminate: Copper foil plus substrate/binder or copper foil plus substrate/binder plus copper foil.



Heat Flow in a Flip-Chip Package

- Heat source and other heat-carrying components



01_so_hssio_basics.odp Date Oct 23, 2009 Page 129

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

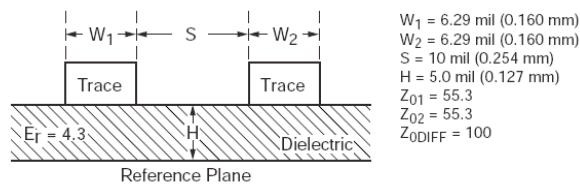


NOTES

Typical heat flow path of a Flip-Chip Ball Gate Array (FCBGA). All Virtex®-4 device packages use the FCBGA surface mount, as do all high-performance board packages.



Microstrip Edge Coupled Differential Pair



NOTES

Differential Pairs

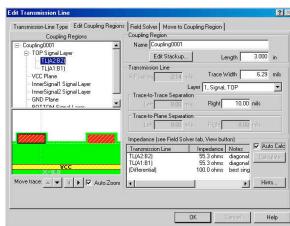
For best results, we should run differential pairs tightly coupled and closely matched. Trace length matching is essential. In FR-4, a 100-mil (1 tenth of an inch) difference in trace length results in approximately 18 picoseconds of difference between the positive and negative signal. This is also enough skew to start causing problems. And, while a tenth of an inch may sound like a lot if we just use normal trace routing from one BGA to another, it is easy to end up with 300 - 400 mils of difference.

If our PCB tool has an auto-trace matching, we need to use it. In general, we will want 50 mils or less difference in differential trace lengths. Differential Trace Width and Spacing This will need to be worked out for each particular stack-up. The board foundry can be a valuable resource, but we need to make sure they know what they are doing. Some published guidelines recommend against letting the PCB vendor do these calculations. We need to make sure they are using a field solver tool to figure the width and spacing of tightly coupled pairs. Then we need to adjust our boards accordingly. One technique we definitely should not use is just choosing a close geometry and then letting the board foundry adjust the impedance with over- or under-etching. If we have a local, in-house field solving program and the expertise to use it, that is even better.

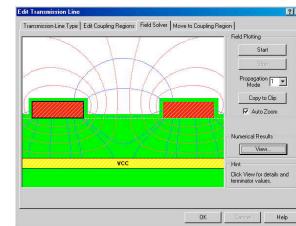


Differential Trace Design

- 50-ohm termination 100-ohm differential traces
- Tight coupling required
 - Traces must be close together
 - Traces must maintain constant spacing to each other
- Use a field solver to determine geometry
 - Two 50-ohm traces do not make 100-ohm differential traces
 - Many possible geometries add up to 100 ohms



----- Impedance Summary
----Differential Z = 100.0 ohm
Common-mode Z = 60.6 ohm
Line-to-ground Z = 55.3 ohm



01_so_hssio_basics.odp Date Oct 23, 2009 Page 131

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



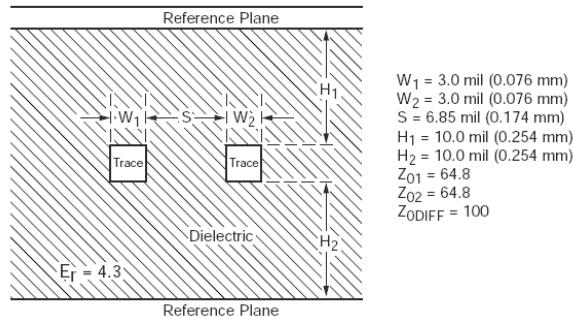
NOTES

Because the transceivers use differential signaling, the most useful trace configurations are differential edge-coupled center strip line and differential microstrip. While some backplanes use the differential broadside-coupled strip line configuration, it is not recommended for 10 Gb/s operation, because the P and N vias are asymmetrical and introduce common-mode non-idealities.

With few exceptions, 50Ω characteristic impedance (Z_0) is used for transmission lines in the channel. In general, when the width/spacing (W/S) ratio is greater than 0.4 (8 mil wide traces with 20 mil separation), coupling between the P and N signals affects the trace impedance. In this case, the differential traces must be designed to have an odd mode impedance (Z_{OO}) of 50Ω , resulting in a differential impedance (Z_{DIFF}) of 100Ω , because $Z_{DIFF} = 2 \times Z_{OO}$.



Stripline Edge Coupled Differential Pair



01_so_hssio_basics.odp Date Oct 23, 2009 Page 132

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net

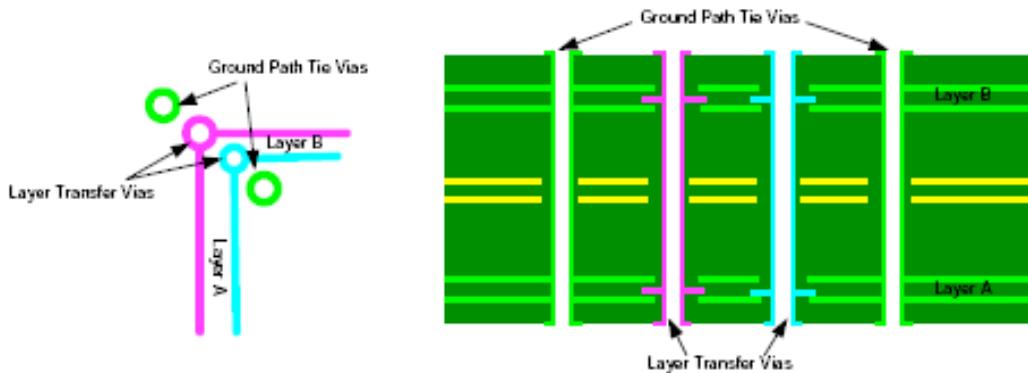


NOTES



Vias

Common Reference Plane



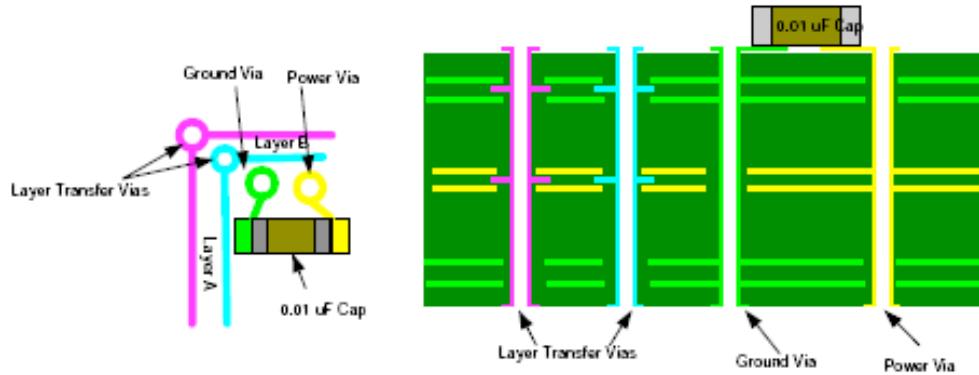
NOTES

Vias

Changing layers on the multi-gigabit differential traces should be avoided whenever possible. If a layer transition is required, we must be extra careful. First, we must provide an intact return path. To do this, we must couple the reference plane of layer A to the reference plane of layer B. The ideal situation is to have both reference planes be ground. In this case, the return path is created by placing a via connecting the planes in close proximity to the via used to make the transition. Figure 4-22 illustrates the technique.



Via Reference Plane not Common



01_so_hssio_basics.odp Date Oct 23, 2009 Page 134

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



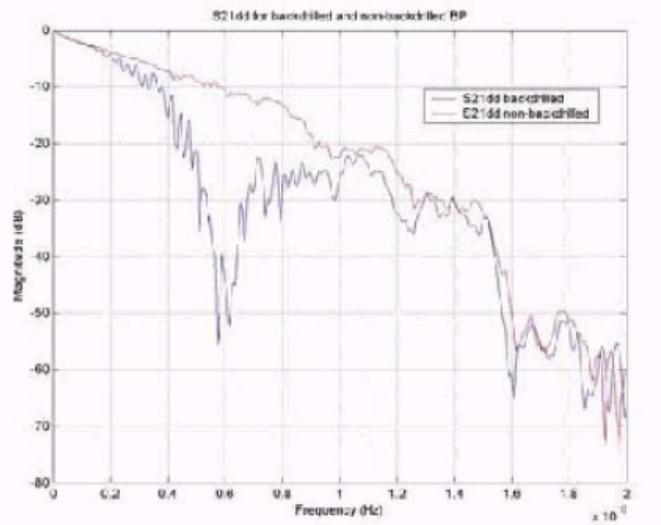
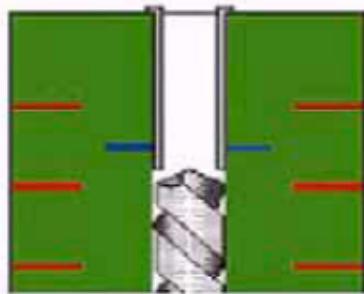
NOTES

If reference planes are not common (one is gnd and one is pwr), then a $0.01 \mu F$ capacitor should be placed across the two planes as close to the transition via as possible.

Another problem with vias is that they represent a stub. Clearly, we know it is a bad idea to introduce stubs in our transmission line



Via Backdrilling



01_so_hssio_basics.odp Date Oct 23, 2009 Page 135

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



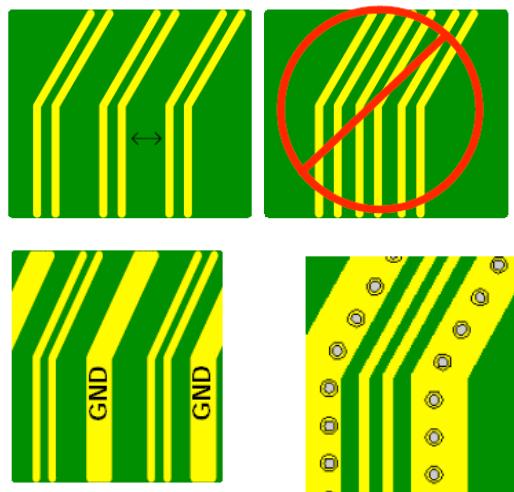
NOTES

Consider a via that transfers a signal from an inner layer to the top layer. The via also goes to the bottom layer, and that unused portion of the via is a stub. One method to avoid this stub is a technique called back drilling. After plating, the unused portion of the via is removed by drilling (as shown with drill bit in lower portion of the figure5

Any design over 5 Gb/s should seriously consider back drilling vias



Space between pairs



NOTES

It is important to maintain a good amount of distance between differential pairs carrying multi-gigabit signals and other traces. One general rule is that at least five times the space between the two signals of the pair should be placed between adjacent pairs.

Another technique is to route a ground guard in parallel to the differential traces. Tying the guard plane back to the reference plane using a via in parallel to the trace often improves this shielding Method.

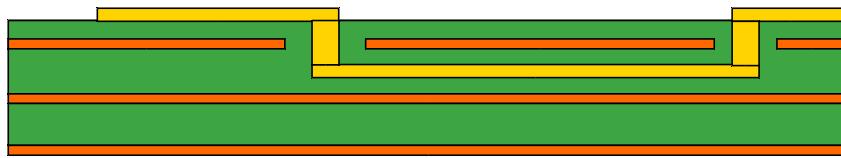
Power layout

Many of the items discussed in the Powering MGT section have board layout implications as well. The placement of the ferrite beads and capacitors that filter the analog power supplies relative to the supply pins and the signal traces must be carefully considered.



Trace Routing

- Avoid vias
 - Vias are impedance discontinuities
 - Route on as few signal layers as possible
 - The same trace routed on multiple layers should be on signal layers that share a reference plane
- Space pairs from all other signals by at least five trace separation widths



NOTES



Trace Routing

- Twenty inches and two high-speed connectors is the maximum FR4 trace length
- Fifty-ohm termination
 - Use 50 ohm for all chip-to-chip connections
 - Overwrite this value for test purposes only
- Length matching
 - Unmatched traces lead to noise, radiated emission, and jitter
 - Match traces of differential pairs to within 50 mils (9 ps)
- Intact reference plane beneath traces
 - No fewer than five trace widths on each side
 - **Never** route traces over a plane split or hole

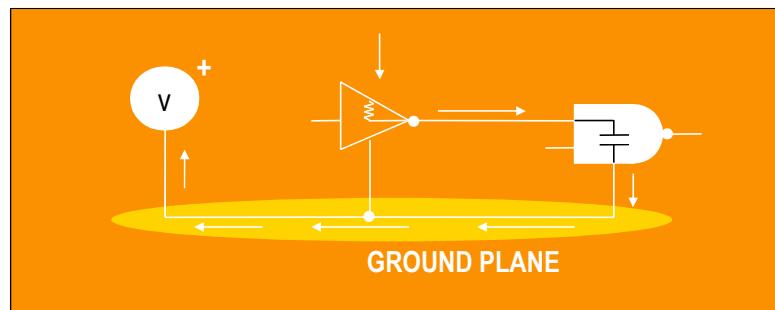


NOTES



Return Currents

- Every trace current induces an equal but opposite current in the reference plane, called a *return current*
- High-frequency return currents run directly under the trace
- Obstructions to the return current (holes or splits in the reference plane) cause the current to take a circuitous route away from the trace, creating a larger inductive loop



01_so_hssio_basics.odp Date Oct 23, 2009 Page 139

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Routing over plane splits also creates issues with the return current. High-speed signals travel near the surface of the trace due to the skin effect. Meanwhile, the return current also travels near the surface of the tightly coupled reference plane.

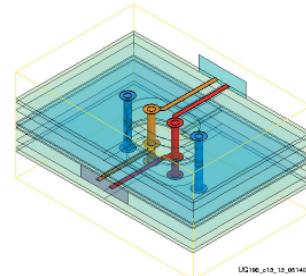
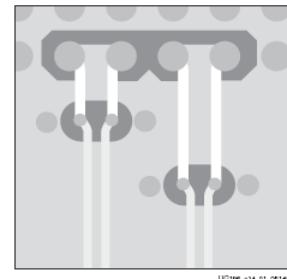
Because of the tight coupling, the return current has the tendency to travel close to the original signal-carrying trace. At the plane split, the return current can no longer follow the same path parallel to the trace, but must instead find an alternative route.

A plane split causes a suboptimal current return path and increases the current loop area, thereby increasing the inductance of the trace at the plane split, changing the impedance of the trace.



MGT Transceiver Board Trace Guidelines

- Use impedance-controlled traces
- MGT signal traces must be routed with the highest priority
 - Traces must be kept straight, short, and with as few layer changes as possible
- MGT signal traces must be avoided near to other traces with noise
- Strip lines are better than microstrip structures (no forward crosstalk)
 - Use the uppermost and lowermost layers (minimize via stubs)
- Use 45° bends instead of 90° turns
- The two traces of a differential pair must be length matched to eliminate skew
- Do not split reference planes over the length of the trace
- Use differential vias



01_so_hssio_basics.odp Date Oct 23, 2009 Page 140

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

This slide summarizes routing guidelines.

The figures show BGA escape routing with 45° bends and differential vias.

For more information, see Chapter 13, “Design of Transitions,” in the user guide.



Trace Guidelines I

- Layer stackup definition
 - High-speed signals should be routed near the top of the board
 - Best case is on the top layer
 - Microstrip line with wider traces with 6 ... 12 mil widths
- Do not press the differential traces
 - Use wider traces with higher distance
 - 5 mil width / 5 mil spacing ↗ 7 mil width / 12 mil spacing with the same impedance
- Separate high-speed differential traces from other traces
 - Limit crosstalk



NOTES

This high-level summary provides a quick reference to some of the guidelines already covered in previous sections, and also introduces some general strategies when designing high-speed serial channels.

When defining the stackup, high-speed strip line layers are kept near the bottom of the board. If all high-speed traces can be routed on the top and/or the bottom microstrip layers, there is no need for a strip line layer. Wider traces are preferred and widths of 6 mils to 12 mils are typical.

Unless there are tight space constraints, the differential trace pairs do not need to be coupled closely. For example, instead of using a 5 mil width with 5 mil spacing, the same characteristic trace impedance can be obtained using a 7 mil trace width with 12 mil spacing.

High-speed differential pairs and transitions must be spread apart on adjacent channels generously to limit crosstalk, even if the paths become longer. In most cases, they eventually have to be spread out to match connector pin spacing.

For transitions, large clearances of planes must be provided around and below transitions to limit excess capacitance. Transitions are spaced apart within the same channel. For example, differential vias typically are not placed next to DC blocking capacitors or connectors. However, in some specific cases, performance was acceptable with this placement.



Trace Guidelines II

- Use plane overlapping around and below transitions
 - Limit excess capacitance
- Remove unused pads on vias
 - Limit excess capacitance
- Keep the via stub as short as possible
 - Use back-drilling if possible



NOTES

To further limit excess capacitance in vias, the unused pads on vias should be removed and the via stub length is kept to a minimum. By routing from the top microstrip to the bottom microstrip, the via stub can be eliminated. Routing from the top microstrip to the bottommost strip line layer results in a negligible via stub. If the lowest layers are not available for high-speed strip lines, other strip lines can be used. However, the via stub should be removed by back-drilling the vias.

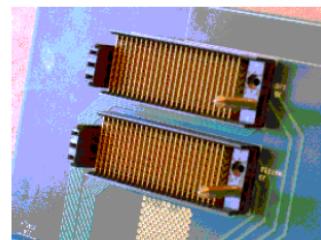
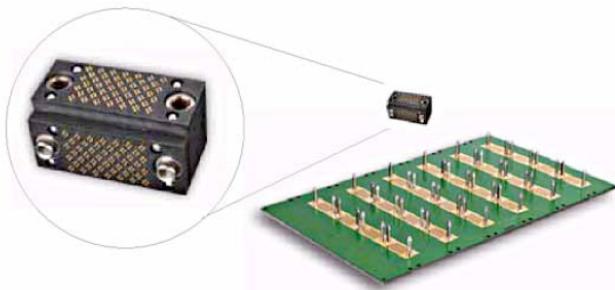
Use of minimum spacing and clearance design rules is to be avoided, such as 5 mil pad clearances. These clearances can be detrimental to performance even at lower multi-gigabit rates due to the excess capacitance from the tight spacing.

Most transitions shown in module have 40 fF to 200 fF of excess capacitance. One exception is a press-fit connector with the PCB pin array having about 500 fF to 800 fF of excess capacitance using these guidelines with a via stub less than 10 mils. With smaller antipads or longer via stubs, the excess capacitance is much greater.

These guidelines are recommended to be followed even for designs slower than 10 Gb/s, allowing for more margin at lower speeds such that a smaller output signal swing can be used. Having a 10 Gb/s capable channel also provides the option to upgrade the bandwidth of the system for the next generation product.



Connectors



01_so_hssio_basics.odp Date Oct 23, 2009 Page 143

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

Only high-speed connectors should be used for multi-gigabit signals. Like everything else in the path, a high-speed connector has controlled impedance. While the connector impedance is never as continuous as a PCB trace, high-speed connectors are much better than normal connectors. Early high-speed connectors were designed for both single-ended and differential signals. The latest, fastest connectors are designed specifically for differential pairs. Here are a few examples of high-speed connectors:

- Gbx
- VHDM-HSD
- VHDM
- HDM
- High Density Plus
- Z-PACK HM-Zd
- Z-PACK HS3

Shielding

Consider how the signals are shielded from each other and from other outside influences. There may be shielding issues on the sides of the connectors.

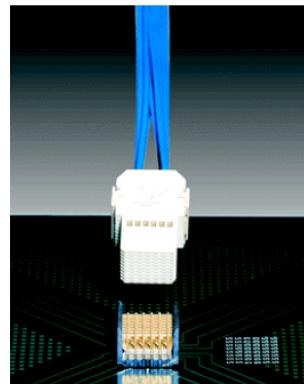
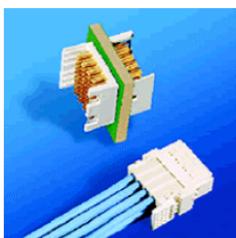
Differential Pairs

Was the connector designed for differential pairs or is it only adaptable to differential pairs?

Maximum Edge rate Have we considered the maximum edge rate? A common source of cross-talk is found in connectors if the edges of the signals entering the connector are too fast. We must know what our connector can handle and what we expect to send through it.



Cables



01_so_hssio_basics.odp Date Oct 23, 2009 Page 144

SO-LOGIC electronic consulting Austria & Brazil. Worldwide Technical Training and Consulting www.so-logic.net thor@so-logic.net



NOTES

If going box-to-box in a custom application, we will need to select a cable/connector scheme. The first thing to consider is how far the signals will travel, and if the signal can go that distance using copper or if we will have to convert to optical. If distance is under 20 meters and speed under 6 gigabits, then copper may work.

One cable used in many multi-gigabit applications is Infiniband cables. Originally designed for 2.5 Gb/s operation in Infiniband applications, the cable has been adapted and slightly modified for FiberChannel, CX4 (10-Gigabit Ethernet) and other uses. It comes in 1, 4, and 12-pair

Variations.

Another interesting cabling option is cable assemblies designed to plug into backplane-type connectors . These assemblies can be used inside cabinets and some include EMI shielding to allow box- to-box connectivity.

Many other cables are being investigated for multi-gigabit uses, including coax and the familiar Cat 5 twisted pair.