

March 10, 2002

**EE273 Class Project
6.67Gb/s differential links for
High Bandwidth Routers**

**Elad Alon
Andrew Ho
Younggon Kim
Hae-Chang Lee**

1. Overall Design Approach

The goal of our design was to improve the bandwidth of a differential 2Gb/s signaling system by at least two times. We tackled this problem by first designing the 20cm PCB traces at either end of the backplane. The next step was to investigate the properties of the complete channel. Finding that the channel alone would not support a higher data rate, we decided to implement equalization and crosstalk cancellation. These were implemented as a Matlab program that invoked Perl and Spice to automate the generation of taps for the FIR filters (equalization and crosstalk) and the plotting of the eye diagram. The design uses a differential, bipolar, current-mode signaling system to achieve a data rate of 6.67Gb/s. BER is calculated to be on the order of 10^{-21} . To obtain tight control of the timing margin, per-pin closed loop timing is implemented. Timing uncertainty was found to be 50ps.

2. Description of the 20cm transmission line on the line card and switch card

2.1. Board specification, line width and spacing, and layer “stack up”

To reduce the cost, we use the basic circuit boards with FR4 dielectric and eight layers. The structure accommodates 3 signaling layers, and each layer is vertically shielded by ground/power planes to reduce crosstalk between the different layers. The stack up is shown in Figure 1.

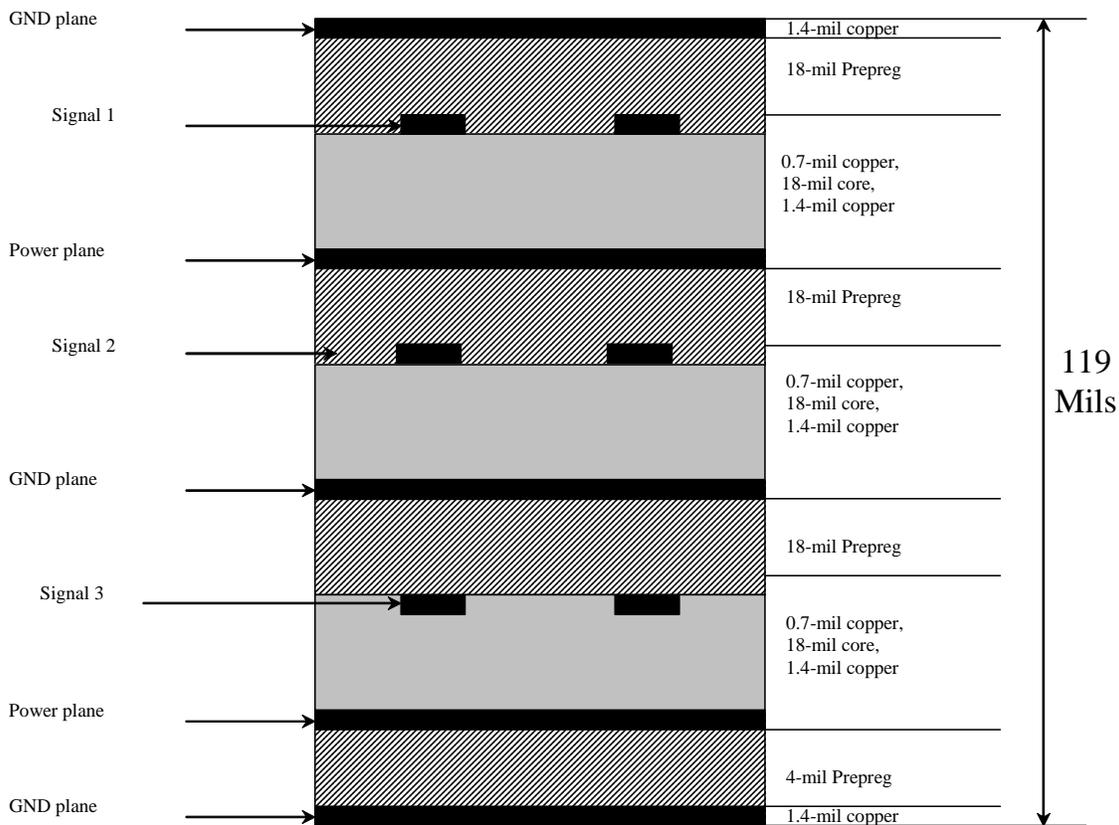


Figure 1 -- board stack up

The total length of the PCB trace is 40cm and it causes a large attenuation at high frequencies. The width of 16mils is used to reduce the attenuation. The thickness of the board also becomes larger as we increase the width of the wire, because we want to set the Z_0 to be 50. However, the cost overhead would be small since the cost is mostly depends on the area of the board. The worst case spacing of the wire is 16mils.

Size of the board

The width of the board is constrained by the Teradyne connector. We decided to use 8-row connectors since they fit into our layer stack up. On the connector, one column has three pairs of differential lines and two shielding lines between them. The switch card has 256 differential pairs; therefore the total number of column will be $256/3 = 86$. By multiplying this number by the pin pitch, we find the width of the connector to be 17.2cm. By adding some margin, the estimated width of the switch card is 20cm. According to the project handout, the line and switch cards are drawn to have same width. We will assume that the line card has the same width.

The height of the board is hard to estimate because it depends on the number and size of the other chips. Using the estimated size of the switch card chip which will be described later (3.9cm by 3.9cm), we will assume that the height is roughly 10cm.

Routing Issue

We have to make sure that the signaling wires are routable. The two most congested areas are the connector and chip footprint.

At the connector side (shown in Figure 2), three differential pairs are connected to one column. Since we have three different layers, the wires can be routed with minimum bit pitch of 32mils.

Figure 2 Connector Pin layout and Routing

At the chip side, escape routing is tricky since the pin pitch is small compared to the connector pitch. The 6 outer rows are populated and one layer should accommodate the escape path of two lines. Two wires with a 32 mils pitch will not fit in 1mm pin pitch. Therefore min spacing will be used for escaping only. The wire whose spacing is reduced has a different characteristic impedance. However the difference is less than 3%. Since the chip is small and the escape path is short (<5mm), the degradation of the transmission

lines is negligible. Using more sophisticated algorithms and a software router, the degradation can be further reduced.

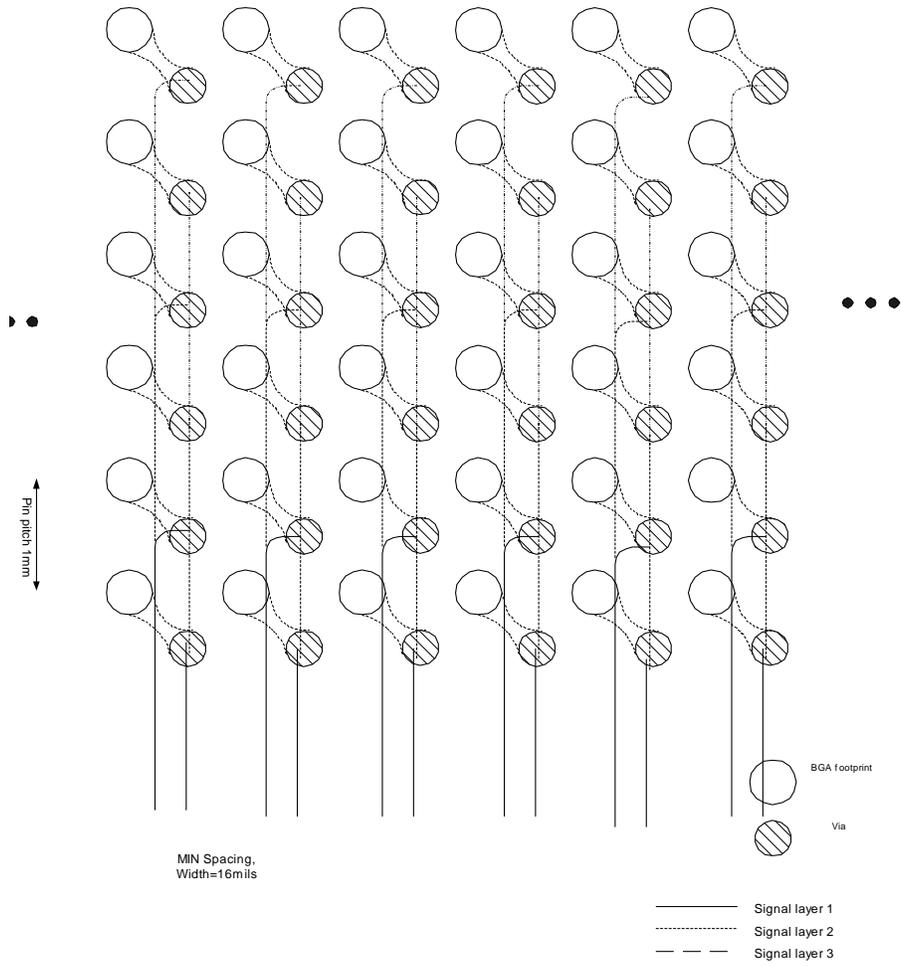


Figure 3 Chip Pin layout and Routing

2.2. Electrical model of the line

The following parameters are extracted from LinPar.

* PCB trace model

* N = number of lines

2

* Lo = inductance matrix

3.585E-7

3.090E-8 3.585E-7

* Co = Capacitance matrix

1.485E-10

-1.280E-11 1.485E-10

* Ro = Resistance matrix

0

0 0

* Go = conductance matrix

0

0 0

* Rs = skin effect matrix

0.002

0 0.002

* Gd = dielectric loss matrix

0

0 0

3. Signal Scheme

3.0 Voltage vs. Current Mode

Our goal was to reach as high of a signaling rate as realistically possible, and in order to achieve this goal it was necessary to minimize power supply and reference noise.

Therefore, we decided to do current mode signaling. Additionally, we were initially considering leaving the transmitter un-terminated. It is advantageous to do this with a current mode scheme because it is easier to build current sources with extremely high output impedances than it is to build voltage sources with very low output impedances.

3.1 Unidirectional vs. Bidirectional Signaling

In order to evaluate the feasibility of bidirectional signaling, we put a current mode pulse into the line and examined the response at the near end. As seen below, the pulse actually causes an oscillation – which means that generating a replica of the transmitted signal to subtract out would be very difficult to do. For this reason, we decided to stick to unidirectional signaling.

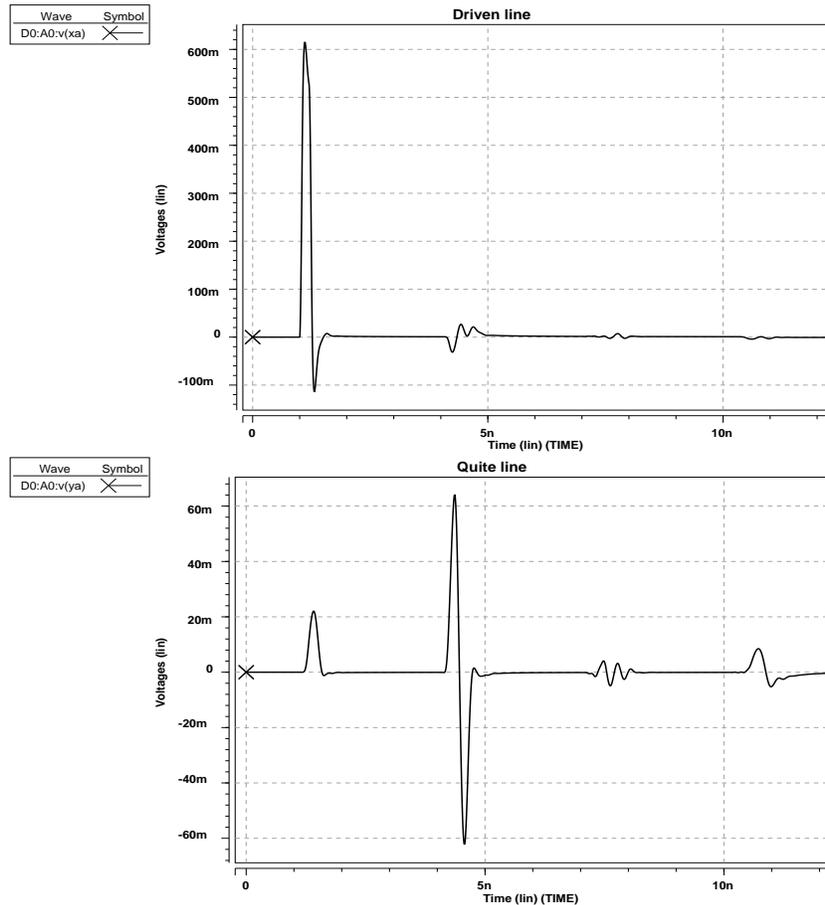


Figure 4 Near End Crosstalk

3.2 Single-ended vs. Differential Signaling

The tradeoffs between single-ended and differential signaling are fairly clear. Single-ended has the advantage that it can potentially be more pin efficient, and therefore run at lower bit rates per pin in order to achieve the same specified 2x speedup in bandwidth. Optimistically assuming that we can use 3 signals pins per ground return, we would need to run each pin at 2.67Gbps, as opposed to running each differential pair at 4Gbps to achieve the same aggregate bandwidth. On the other hand, single ended designs are very sensitive to all forms of noise – particularly to power supply, reference, and cross talk. In order to avoid these issues, particularly reference and power supply noise, we decided to stick with the original differential scheme. Additionally, the task of modeling the system is much easier with a differential signal, because you truly only need to look at 2 conductors to get all the information you need about the system.

3.3 Number of Signal Levels

The only reason to use anything other than a binary signaling scheme is if the channel is severely band-limited, and by using multiple signal levels you can remain below the frequencies of high attenuation. This is because if you signal with 2^N levels, the symbol rate is reduced by N , but your noise margin is reduced by $2^N - 1$. A plot of the frequency response of the channel is shown below:

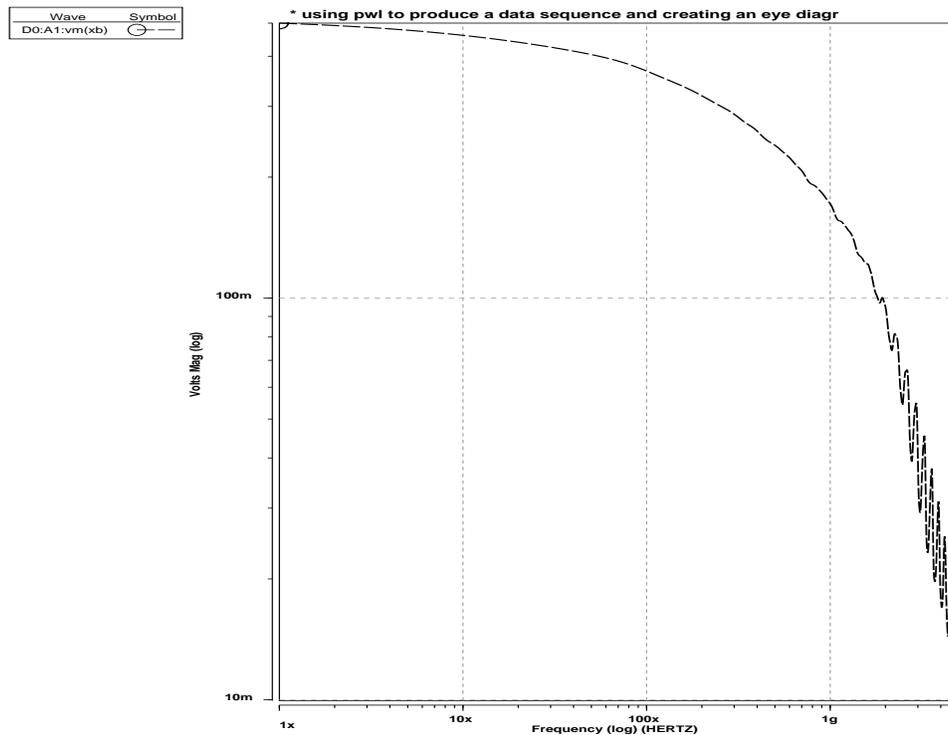


Figure 5 Frequency Response of the channel

Clearly, the channel has a high degree of attenuation at even 1+Ghz. Therefore, even if we used a 4 level scheme, we would still need to do some sort of equalization in order to overcome the attenuation. Equalizing a multi level signal is not trivial (nor is clock recovery and reference generation), therefore we chose to stick with conventional binary signaling.

3.4 Signaling Rate

Our design objective was to maximize the signaling rate. In the end, we achieved 6.67Gbps.

3.5 Signaling Convention

For all of the reasons discussed in the previous sections, we chose to do unidirectional differential bipolar current mode signaling.

3.6 Equalization

As shown in Figure 6, the channel is highly attenuating at the frequencies we are interested in running the channel. In order to compensate for the attenuation, equalization was necessary to achieve any sort of eye opening. The first step in the design of

equalization is to characterize the channel. The following model allows us to include termination mismatches and current mismatches:

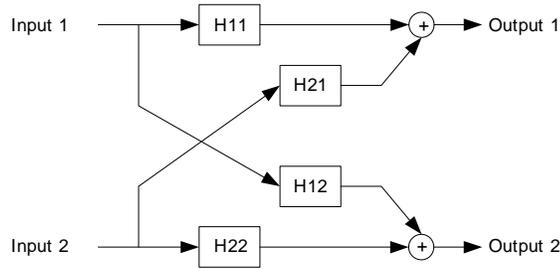


Figure 6 Characterization of the Channel

We calculated the coefficients for the taps by sending a lone pulse down the channel (Input1) and measuring the response at the far end (Output1). The measured response is H11, the pulse response of the channel. We then sample the channel response at a spacing of 1 bit period. Finally, the FIR tap coefficients are calculated by finding the least squares approximation for the inverse of the channel response¹. All of this was implemented using a combination of Spice, Perl, and Matlab. The approach is shown below:

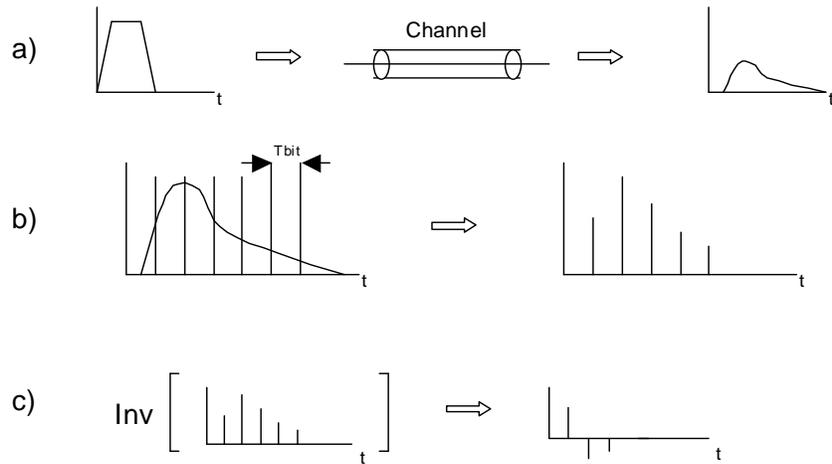


Figure 7 Method of Equalization

(a) In Hspice, a lone pulse of 1 bit period wide is sent down the channel model. The response at the far end of the line is recorded. (b) The data points are ported into Matlab and sampled at rate = 1/Tbit. (c) The sampled channel response is put into a convolution matrix. The psuedo-inverse of the convolution matrix is then matrix-multiplied to an ideal channel response (single impulse) to get the FIR coefficients.

¹ The coefficients are scaled to have a total magnitude of 1 in order to ensure that the maximum current is bounded to $\pm 20\text{mA}$ after equalization.

3.7 Crosstalk Control

After the equalization FIR taps are generated, we sent an equalized lone pulse down the aggressor line (Input1) and measured the response at the far end of the victim line (Output2).

Comparing H11 and H12, we realized that H12 has significant components at approximately half the bit period. This means that in order to effectively reduce the crosstalk, we would need to run the crosstalk control taps at twice the bit rate. The equalizer remains at bit rate to reduce the hardware implementation complexity. A block diagram of the equalizer is shown below:

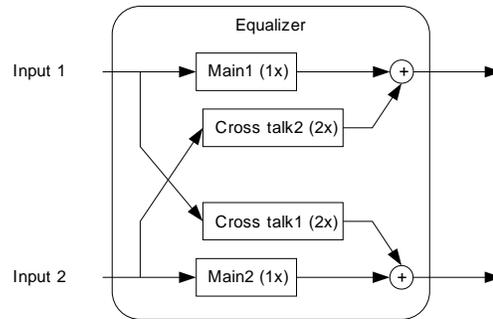


Figure 8 Equalizer Block Diagram

The crosstalk response (H12) is then sampled at a spacing of $\frac{1}{2}$ bit period. H11 is also re-sampled at $\frac{1}{2}$ bit period, in order to keep the time domain representations the same. The crosstalk control FIR tap coefficients are determined by solving the following equation for W:

$$H_{11} \cdot W = -H_{12}$$

Unfortunately, there are several details we neglected in designing the equalizer and crosstalk control in this manner. For example, this method does not consider the self-induced crosstalk on the aggressor line that occurs because of the cross talk canceling current sent on the victim line. In reality, the channel is a multiple- input, multiple-output system, and we can make use of this fact to not only cancel crosstalk but also boost the received signal by transmitting through **both** paths to the output.

3.8 Advanced Equalizer and Crosstalk Control Design

Finding the equalizer and crosstalk control FIR taps sequentially has some drawbacks. First, it requires traversing of the channel on two separate occasions:

- i. To determine the equalizer FIR taps.
- ii. To determine the crosstalk control FIR taps, after the equalizer taps are determined.

Second, the sequential design assumes that interaction between the aggressor and victim is necessarily an adverse effect, and therefore is designed only to cancel it. However, by looking at the aggressor line and its crosstalk on the victim line as a channel network, we could actually design the equalizer and crosstalk control together in the following fashion:

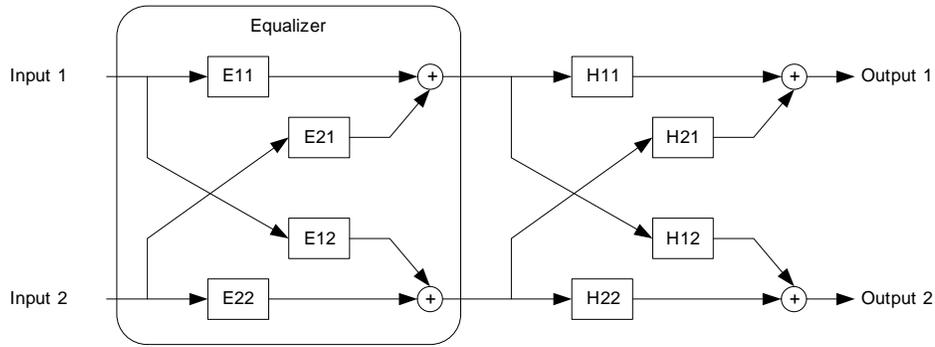


Figure 9 Better Characterization of the System

The procedure for calculating the equalizer taps is shown below:

A)

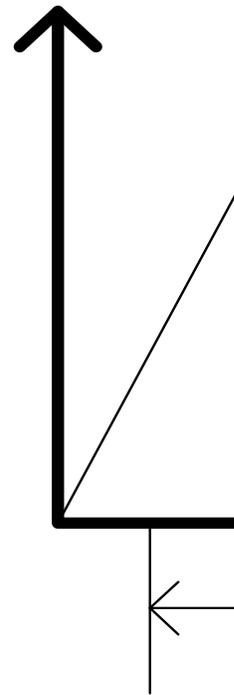


Figure 10 Finding the Equalizer Taps

With the termination mismatches and current mismatches included, we first send a pulse down line 1 (A). The responses at the far end of line 1 (H11) and the far end of the line 2 (H12) are recorded. We then send a pulse down the other line and measure H22 and H21 in the same manner. H11, H12, H22, and H21 are all sampled at half the bit period (B). The four sampled responses are combined in Matlab in the following way:

$$\begin{bmatrix} H_{11} & H_{21} \\ H_{12} & H_{22} \end{bmatrix} \cdot \begin{bmatrix} E_{11} & E_{21} \\ E_{12} & E_{22} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The FIR taps E11, E21, E12, and E22 can be found by solving the above equation in the Least Square Sense. The “1” in the right matrix represents an ideal impulse response of the appropriate length. “0” is a vector of zeros (implying that a pulse on line 1 should cause no response at the far end of line 2, and vice versa) (C).

A possible hardware implementation of a system that would be able to automatically calculate the tap values is shown below:

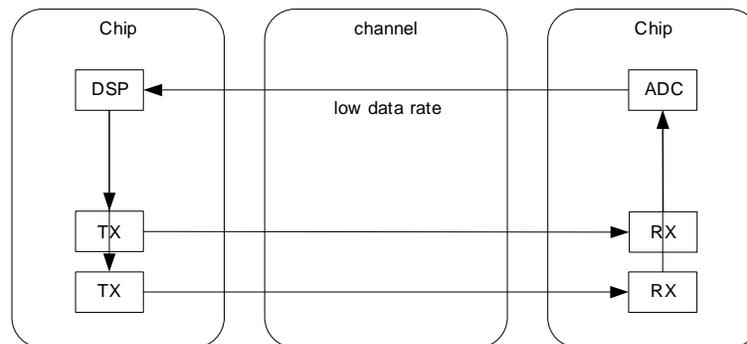


Figure 11 Hardware needed for equalizer and xtalk cancellation

When the system first starts running, the transmitter will send a training sequence of lone pulses across the channel in order to generate the response at the receiver. In order to measure this response, an ADC will convert the analog voltages into digital values. These bits will be sent back to the transmitter over a slow back channel and an on or off chip DSP will do the required operations to calculate the correct taps for each transmitter. Plots of the previous sequential design’s pulse response and the advanced design’s pulse response are shown below. Note that the advanced design has both a higher main pulse and significantly less crosstalk.

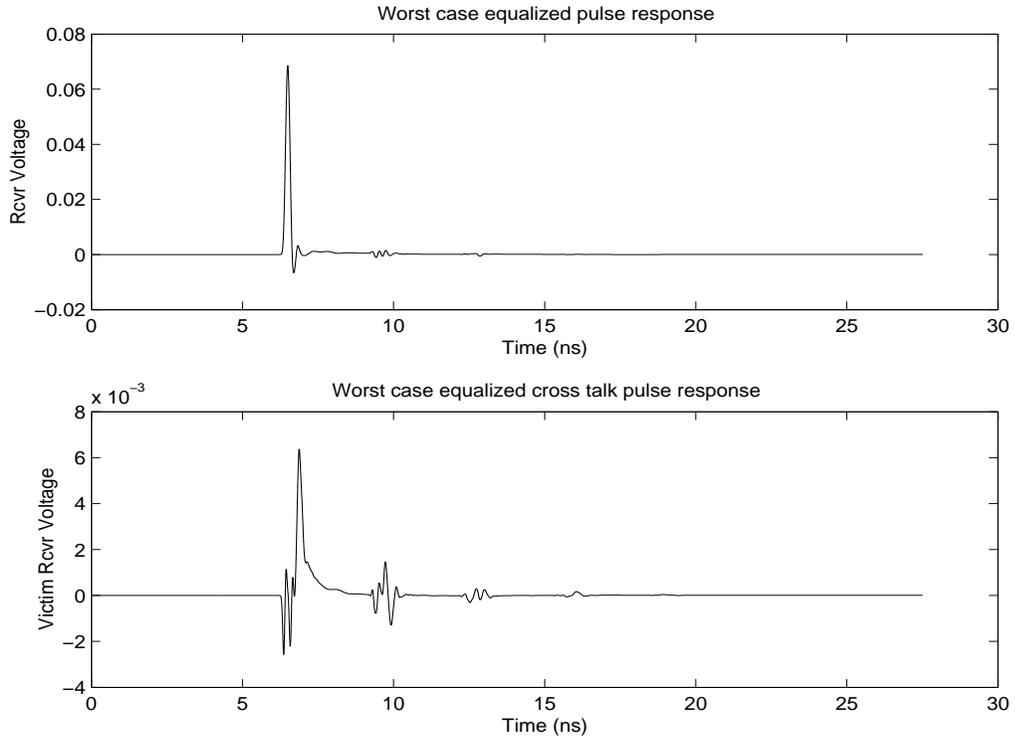


Figure 12 Sequential Equalizer/Crosstalk Control Design Pulse Response (3 taps E11, 7 taps E12)

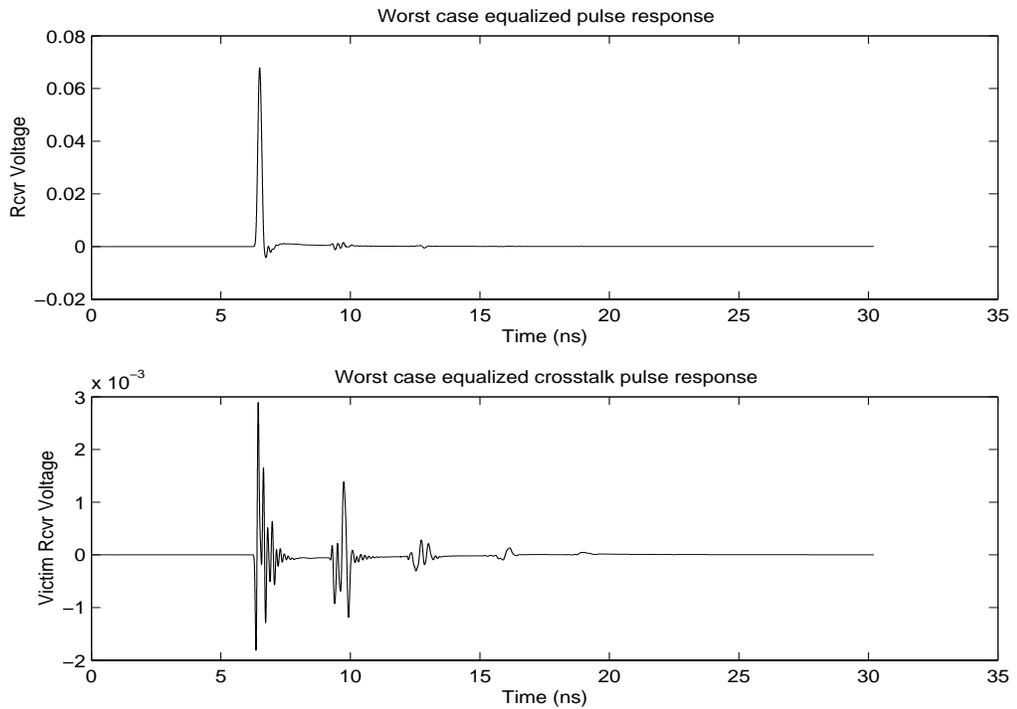


Figure 13 Advanced Equalizer/Crosstalk Control Design Pulse Response (3 taps E11, 7 taps E12)

3.9 Signaling Details

Our system uses bipolar current mode drivers with a rise time of 50ps, bit time of 150ps, 50Ω termination resistor, and a **peak** current of +/-30mA. This peak current value may seem excessive, but due to equalization the average value of current is actually much lower. This is because the peak current is only supplied for lone pulses which have a worst case cross talk pattern on them. For a random sequence of 300 bits, the average power supplied by the driver was 10.7mA (as measured in spice). Finally, we chose to use 3 taps for the bit-rate equalizers (E11 and E22) and 7 taps for the double-rate equalizers (E12 and E21).

4. Timing and Synchronization

4.1 Overview

We perform per-pin closed-loop timing. There is one crystal oscillator (20ppm, 20ps p-p jitter) per board. This crystal oscillator generates a differential clock that is fed into each chip on the board (Figure 14). Within the chip, this differential clock is buffered up and fed to the CDR (clock and data recovery) at each pin. Each CDR has one on-chip VCO. As the off-chip oscillator has a limited frequency range (max. 300 MHz), it is necessary to use a ‘global’ PLL (an additional on chip VCO) to multiply the clock to drive the rest of the chip including the transmitters as shown below.

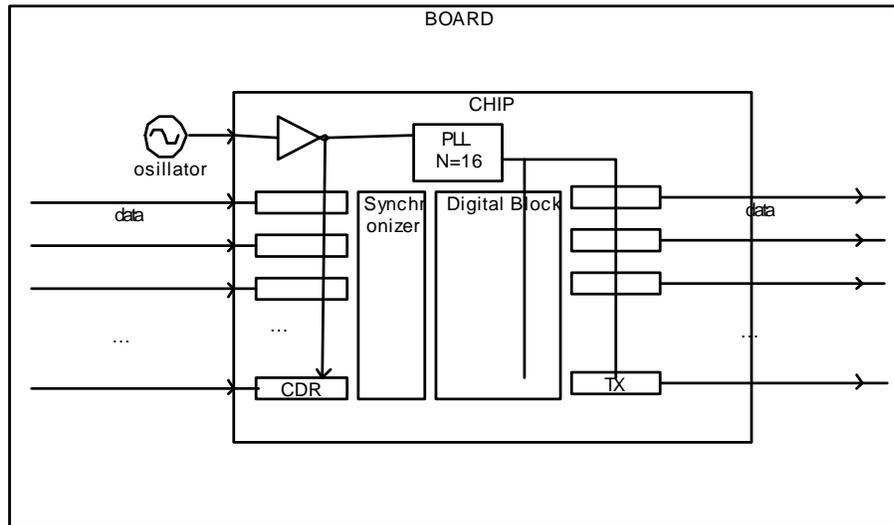


Figure 14 Board/Chip level Timing

The block diagram of the CDR is given in Figure 15. The clock from the crystal oscillator is an input to the CDR. The CDR consists of two loops: frequency acquisition and phase acquisition. (Note that the frequency acquisition loop is what the global PLL for the rest of the chip looks like, which is why we have omitted a description of it.) At start up, a controller (not shown) closes the frequency acquisition loop (top loop). Once the VCO is running at 16 times the crystal oscillator, the phase/frequency comparator

(PFC) notifies the controller that it is locked. This prompts the controller to switch to the phase acquisition loop.

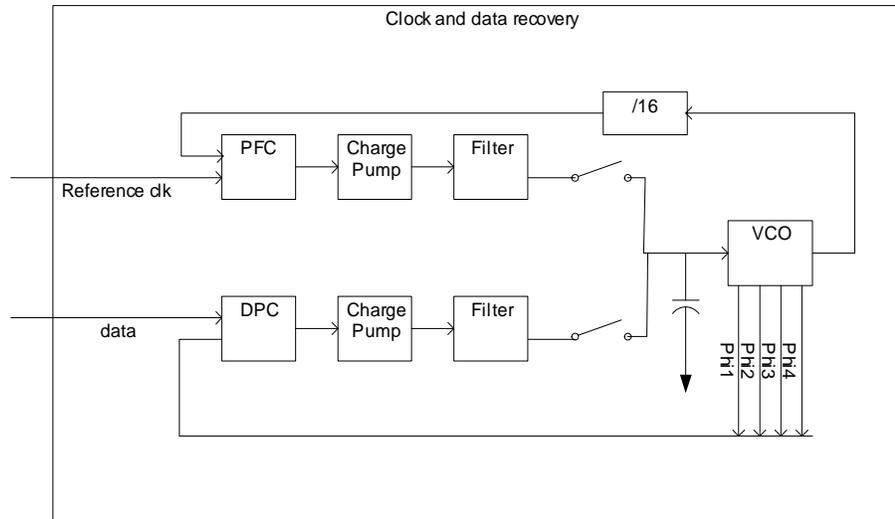


Figure 15 CDR at each pin

The key to the phase acquisition loop is the design of the Data/Phase comparator (DPC) shown below. Since our receivers are differential sense-amp implementations, it is not realistic to run them double edge trigger at clock frequencies of 2GHz and above – enough time must be given to not only resolve the output, but also to reset it. We use four receivers that are triggered off the four phases generated from the VCO as shown above such that each receiver is only triggered once every clock cycle. The DPC not only acquires phase information (which is translated into an up or down pulse), but as a by-product of phase acquisition recovers the data as well. The logic can obtain phase information by comparing adjacent data bits to see if a transition has occurred. If a transition has occurred the logic can decide whether the clock is sampling early or late by whether the edge sample is equal to the previous or current bit. An interesting side note is that a data bit from the previous clock cycle must be stored to make full use of the edge information. Furthermore, while not shown here, it is necessary to include some sort of synchronization circuit after the data receivers to settle possible metastable edge samples. This control loop centers the sampling clock on the middle of the eye.

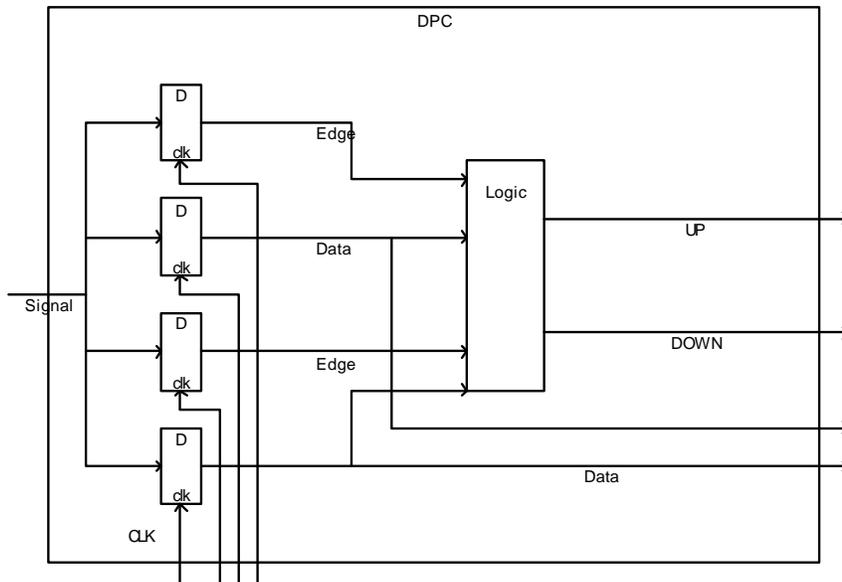


Figure 16 Schematic of DPC

Due to the divide by 16 in the feedback path, the VCO runs at a frequency that is 16 times faster than the reference clock. It should be mentioned that it is very difficult for a full swing CMOS ring oscillator to run faster than $6FO_4$. This means that the maximum realistic frequency using a ring oscillator is about 3.33GHz for our technology ($FO_4=50ps$). While faster oscillators are possible with small signal swings, full swing ring oscillators are more convenient as CMOS logic must be driven with the outputs of the oscillators and as larger signals are more immune to noise. As a result, if we send one bit per edge of the clock, the maximum realistic bit rate is 6.67Gb/s.

4.2 Timing Budget

As we implement per-pin clock recovery, the following sources of timing uncertainty are removed:

- skew due to difference in channel length
- skew in the transmit clock between the numerous transmitters
- skew due to fixed differences in propagation delay at TX/RX output
- frequency difference between TX and RX (mesochronous)

In order to handle the frequency offset between the TX and RX, transmitted data needs to be encoded to ensure a transition density. 64B/66B encoding, which has been proposed for 10G Ethernet, is a viable option since the difference in frequency is only 20ppm. Furthermore, it is attractive as the overhead is minute (about 2% of bandwidth). With 64/66 encoding, the maximum phase drift between transitions is less than 0.3ps. We will thus ignore this factor.

4.2.1 Transmitter

The jitter in the transmitter is determined by the jitter in the 'global' clock. If we assume that the global PLL has a small enough bandwidth such that jitter at the input (i.e. of crystal oscillator) is completely filtered whereas the jitter of the VCO shows up without attenuation, then jitter of the transmitted data is simply characterized by jitter of the VCO which is 10% of the cycle time. Thus, the jitter of the transmitted data is $0.1 \cdot T_{\text{cycle}}$ (p-p) or equivalently $0.2 / (\text{Data Rate})$.

4.2.2 Receiver

We assume that a data receiver has the same phase offset of $\pm 10\text{ps}$ as a phase comparator. The phase offset can cause a timing uncertainty of 20ps if the offsets of the edge and data samplers are skewed in opposite directions. In addition, the CDR at the pin also has jitter from the VCO like the transmitter $- 0.2/(\text{data rate})$.

When the RX is locked to the TX, the worst-case timing offset between the two due to jitter is the peak to peak jitter of a VCO (not 2 times this quantity). Thus, the total timing uncertainty is $0.2/(\text{Data Rate}) + 20\text{ps}$ which is equal to 50ps when the data rate is 6.67Gb/s . As the total aperture time of the RX is 20ps , we need to make sure that we have at least 70ps of timing margin. This is clearly seen in the eye diagram of Figure 17.

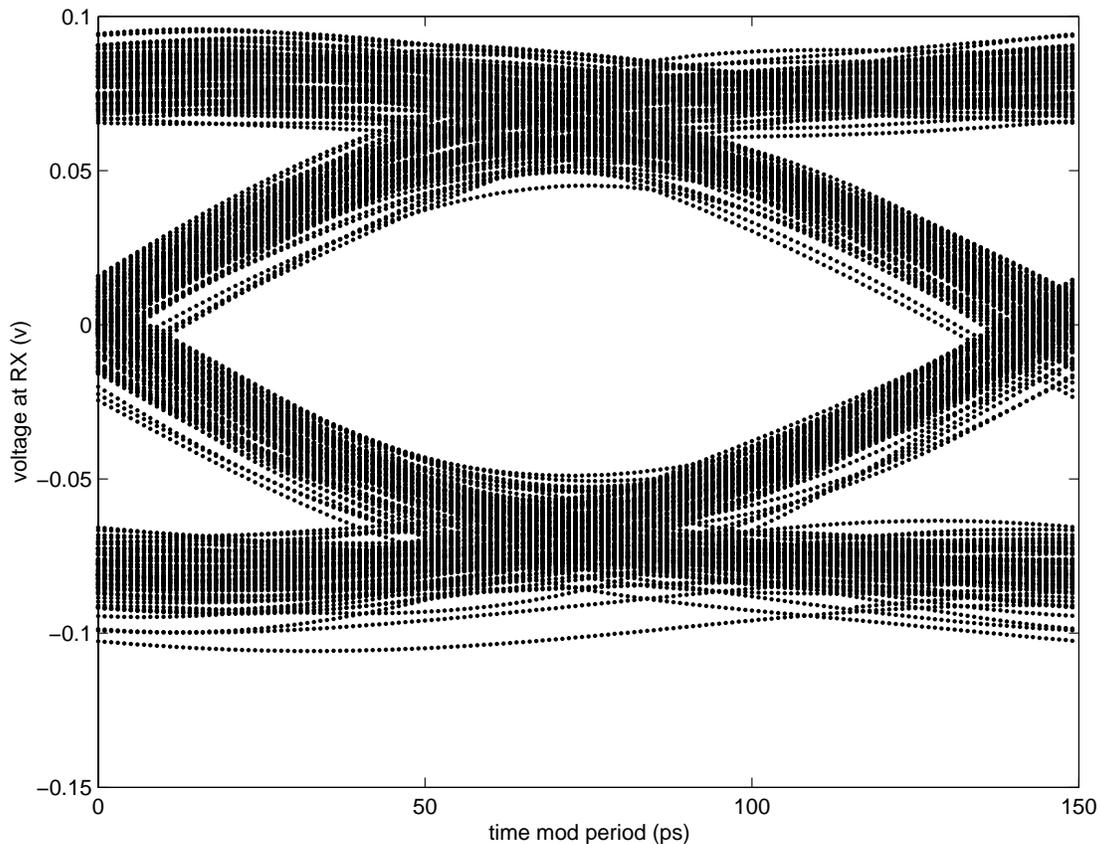


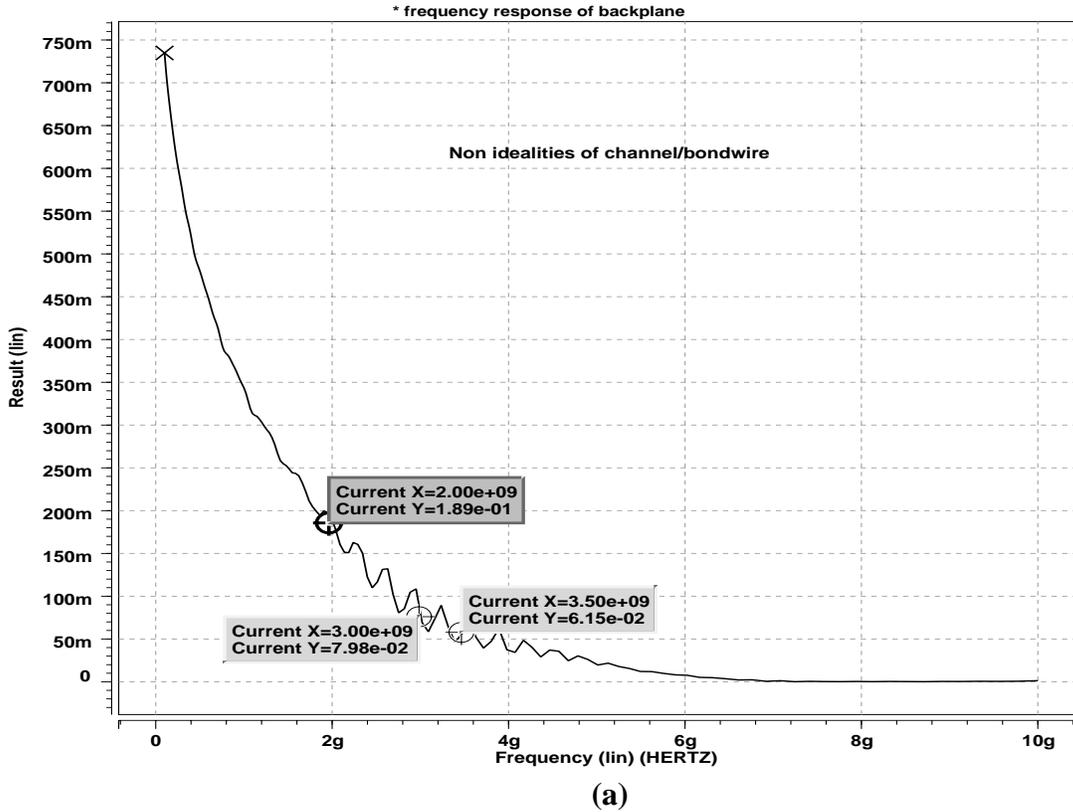
Figure 17 Eyediagram @ 6.67Gb/s, 300 hits

5. Noise Budget and BER

It is more convenient to perform the noise budget at the receiver as equalization is done at the transmitter. This is because it is difficult to define signal swing and thus gross margin at the transmitter where equalized pulses have varying amplitudes that depend on the past history of bits (i.e. frequency content of the data). After equalization, the amplitudes of all signals should be nominally the same. By referencing to the receiver we can do a much more straight-forward analysis of our noise budget.

5.1 Channel Attenuation

Two plots are included that show the frequency dependent attenuation of our channel for two different chip packages (bond-wire and flip-chip). There is no performance difference between the two up to 4Gb/s. However, at 6Gb/s and higher, flip-chip packages give about 25% less attenuation than the bond-wire package. We decided on the flip-chip package as we wanted to push our system to the limits of the technology. As far as noise analysis goes, this channel attenuation does not need to be considered explicitly as we implicitly include it by considering the signal swing at the receiver.



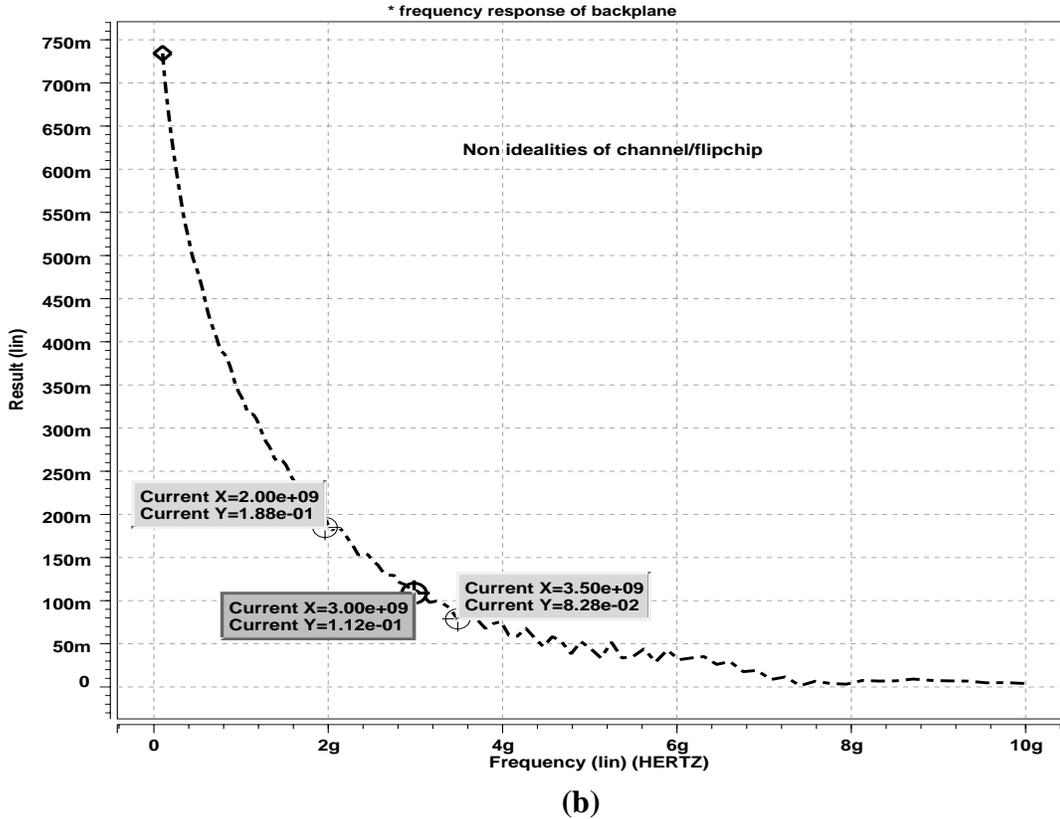


Figure 18 Frequency response of channel with a) bond-wire b) flip-chip

5.2 Gross Margin

In a system with equalization at the transmitter, defining gross margin is not as straightforward as simply measuring the signal swing. The equalization attenuates the DC components of the signal, so at the transmitter side a lone pulse will have a much larger “swing” than a steady stream of values. Similarly, at the receiver end, the magnitude of received signals strongly depends on previous data values because our equalization is not ideal. In our case, a lone pulse had the smallest swing, and for this reason we decided to define the gross margin as being the height of a lone equalized pulse. The gross margin at 6.67Gb/s is **143.6mV** (see 5.3).

5.3 Transmitter and TX/RX termination mismatch

Current mode drivers are specified to within 5% of their nominal value. The worst case occurs when the current is 95% of its nominal value. Similarly, the worst-case terminations are when both the RX/TX terminations are 7% less than their expected values (see 5.3.1). These offsets all reduce the signal swing and thus the eye opening. This worst case arrangement of current and termination mismatch is shown in Figure 19. Calculation shows that these three effects combine to reduce the eye by 11.8%. Only the first reflection due to the termination mismatch is considered, as the channel attenuation is large (5.1). This means that the K for these offsets is 5.9% as this is a proportional noise source. In calculating our worst case margins, we defined the gross margin to be the signal swing of the equalized pulse with the resistor and current mismatches **already included**. We defined the gross margin in this manner because these effects cause the actual swing to be reduced – and hence all of the proportional noise sources are reduced

by this same amount. Additionally, the equalizer is designed so that it will compensate for differences between the two channels, and hence the 5.9% we calculated by hand will not be fully observed. Therefore, our gross margin, as stated above, is **143.6mV**.

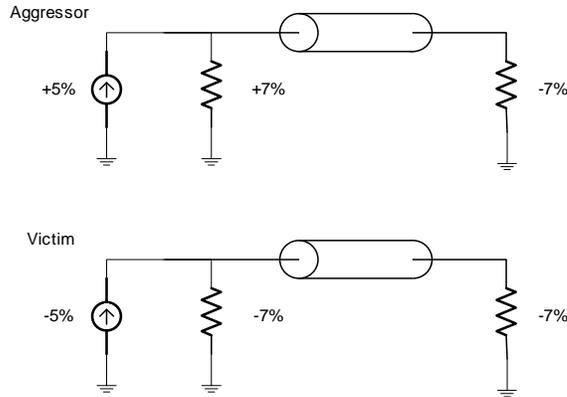


Figure 19 Worst Case setup

5.3.1 Programmable Termination Resistors

The transceivers use programmable termination resistors whose mismatch occurs because of inter and intra-die process variations. Inter-die variations (+/- 50%) can be reduced via calibration. In this scheme, an off-chip high precision resistor is used as a reference. A current source drives the reference resistor and the ensuing voltage is measured by an on chip ADC. The same current source is then switched to drive an on-chip programmable resistor. The control logic adjusts the on-chip resistor until it develops the same voltage as the off-chip resistor. After calibration, the same control setting is fed to all on-chip programmable terminations.

The area and power overhead is small because only one circuit block is needed per chip. To minimize the effect of supply noise on our calibration, dedicated power and ground pins are assigned. After calibration, the total mismatch will be:
 5% from the incremental value of resistor + 2% from on die variation = 7%

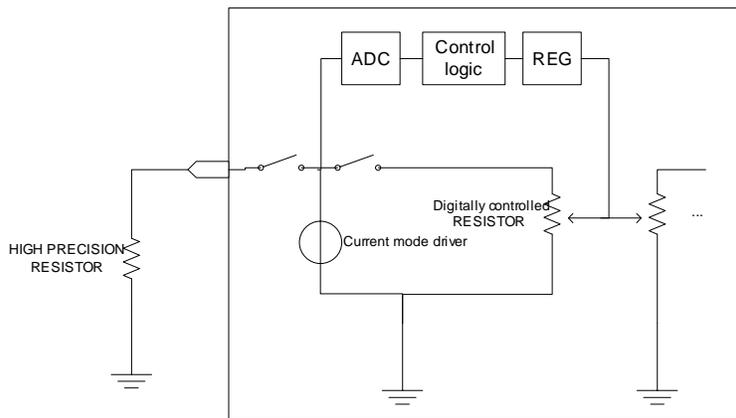


Figure 20 Automatic Termination Calibration

5.4 Crosstalk

Usually, one would find the crosstalk coefficients K_{fx} and K_{rx} from the L and C parameters of a transmission line to model LC coupling. However, as our system implements crosstalk cancellation whose effectiveness is not easily calculated (see 3.8), it is difficult to do a meaningful hand estimate.

In order to measure the worst-case crosstalk, we send a lone pulse into the aggressor. Due to the crosstalk cancellation scheme described in 3.8, a series of pulses are injected into the victim to cancel the crosstalk induced by the aggressor. As the cancellation is not perfect, there is a signal remaining on the victim. This is what is measured to assess the worst possible crosstalk. Bit periods are defined to begin from the moment in time that the aggressor reaches its peak. For each bit period, the maximum crosstalk amplitude within the period is found and accumulated. This assumes that cycle to cycle jitter is large enough at the receiver that it will detect the peak crosstalk signal no matter where it occurs within a bit period. This is clearly pessimistic and is not very realistic given the timing uncertainty found in section 4. However, this measurement gives us an upper bound on the crosstalk noise.

The crosstalk waveform observed on the victim due to an equalized lone pulse on the aggressor at 6.67Gb/s is in Figure 13. The measured worst-case crosstalk at this speed is 14.9mV. This means that the eye is reduced by twice this value (29.8mV). We treat this as a proportional noise source. **kn = .1038.**

5.5 Receiver Offset and Sensitivity

Clocked receivers are specified to have a sensitivity of 10mV. This is a fixed noise source that cannot be improved. The maximum offset is specified to be +/-40mV. However, the offset can be reduced via the calibration method of 5.5.1 to 5mV. The two properties of the receiver pose a total fixed noise of 15mV.

5.5.1 Clocked Receiver Offset Cancellation

Offsets in the receiver can be reduced using sense amplifiers with trimmed capacitors. The sense amplifiers are trimmed by placing binary-weighted pMOS capacitors on the two output nodes. Digitally adjusting the capacitance while shorting the inputs unbalances the amplifier to cancel the offset voltage.

In the calibration stage, the input nodes are shorted together using a transistor. By measuring the output, the control logic decides which side needs more or less capacitance. Iterating this step will eventually trim down the offset to the resolution of the calibration circuit. Moreover, this process can be applied to each amplifier individually by an on-chip controller without any external control.

M.J. Lee et. al show that the offset can be trimmed to within 8mV with any untrimmed offset <120mV. Since we have smaller maximum offset of 40mV, it can be trimmed to about $8*40/120 = 2.66\text{mV}$ (assuming comparable resolution). However, we will be pessimistic and use 5mV in our noise analysis. Also it is shown that the area overhead for the control logic is 0.13mm^2 (0.25um technology). Since we are using 0.13um technology, the area will be about $0.13*(0.13/0.25)^2 = 0.0352\text{mm}^2$ per control. Assuming that there is offset calibration at each pin, the total area overhead will be $32*0.0352 = 1.1264\text{mm}^2$ for line card, and $256*0.0352 = 9.01\text{mm}^2$ for the switch card.

Precharge and equalization offset compensation

Figure 21 Receiver Offset compensation

Lee, M.-J.E.; Dally, W.; Chiang, P., "A 90mW 4Gb/s equalized I/O circuit with input offset cancellation", Solid-State Circuits Conference, 2000. Digest of Technical Papers. ISSCC. 2000 IEEE International , 2000

5.6 Inter-Symbol Interference

Due to the frequency dependent attenuation and imperfect equalization, there is significant data-dependent ISI. This ISI causes timing jitter as well. We characterize ISI as a proportional noise source. The proportionality constant is found from measuring the ISI of an equalized eye without crosstalk over 300 bits. The measured ISI is referenced to the full signal swing to obtain the K factor. **kn = 0.1727**

5.7 Noise through the power network

As we use a differential bipolar current mode signaling, various types of noise from the power supply are eliminated. Signal return crosstalk does not exist as signal return current is 0. Single supply noise is made common-mode due to differential signaling. There is no reference noise as our references are not generated from a power supply but rather from the complement signal.

5.8 Gaussian Noise

Due to differential signaling, gaussian noise due to perpendicular crosstalk is made common-mode. This leaves two sources of Gaussian noise. Unrelated backplane noise (5mV) and resistor thermal noise. Resistor thermal noise is found by $V_n^2 = 4KTR\Delta B$, where ΔB is $\frac{1}{2}$ the data rate. For our application, this number is very small compared to 5mV but is included for the sake of completeness. It must be remembered that for Gaussian noise, the variances add rather than the standard deviations.

5.9 BER calculation

The following table is the spreadsheet used to find the BER given the noise sources listed in this section. Only the result for the fastest data rate deemed practical is given. Our BER was found to be $9.5e-22$.

Data Rate	6.67E+09	Bandwidth	3.34E+09		
Gross Margin (mV)	143.6	Swing (mV)	287.2		
Clean Eye (just channel)	94				
crosstalk (mV)	29.8				
Proportional Noise (%)		Fixed Noise		Gaussian Noise	
crosstalk	10.38			Backplane	5.00E+00
Inter-symbol interference	17.27	RX offset/Sens.	15	Termination	5.25E-02
Total Proportional	27.65	Total Fixed mV	15	Total Gauss. mV	5.00E+00
Total Bounded (mV)	94.4				
Net Margin (mV)	49.2	VSNR	9.84E+00	BER	9.482E-22

6. SPICE Simulation

6.1 TDR Simulation of Transmission Line

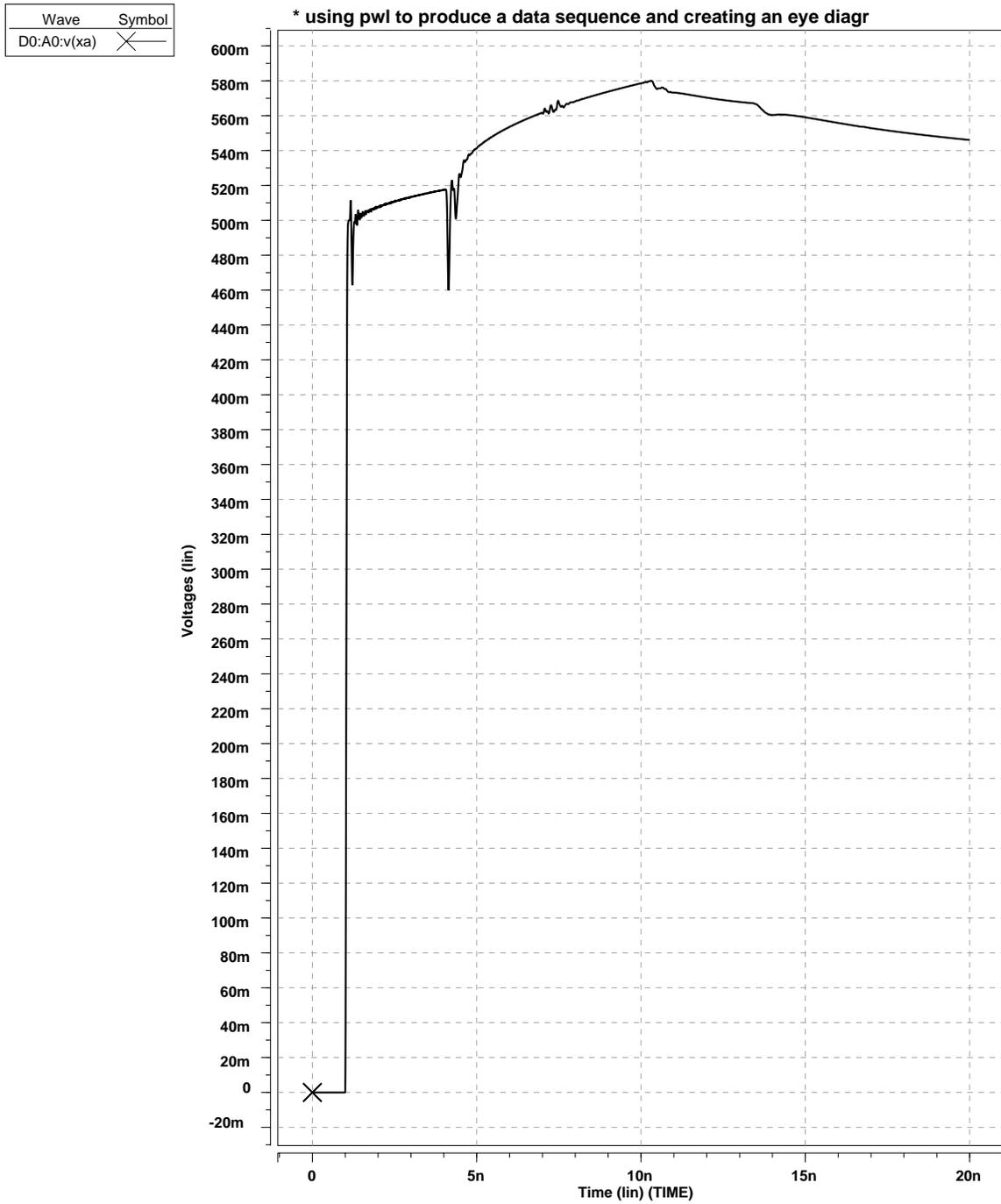


Figure 22 TDR

6.2 TDT Simulation of Transmission Line

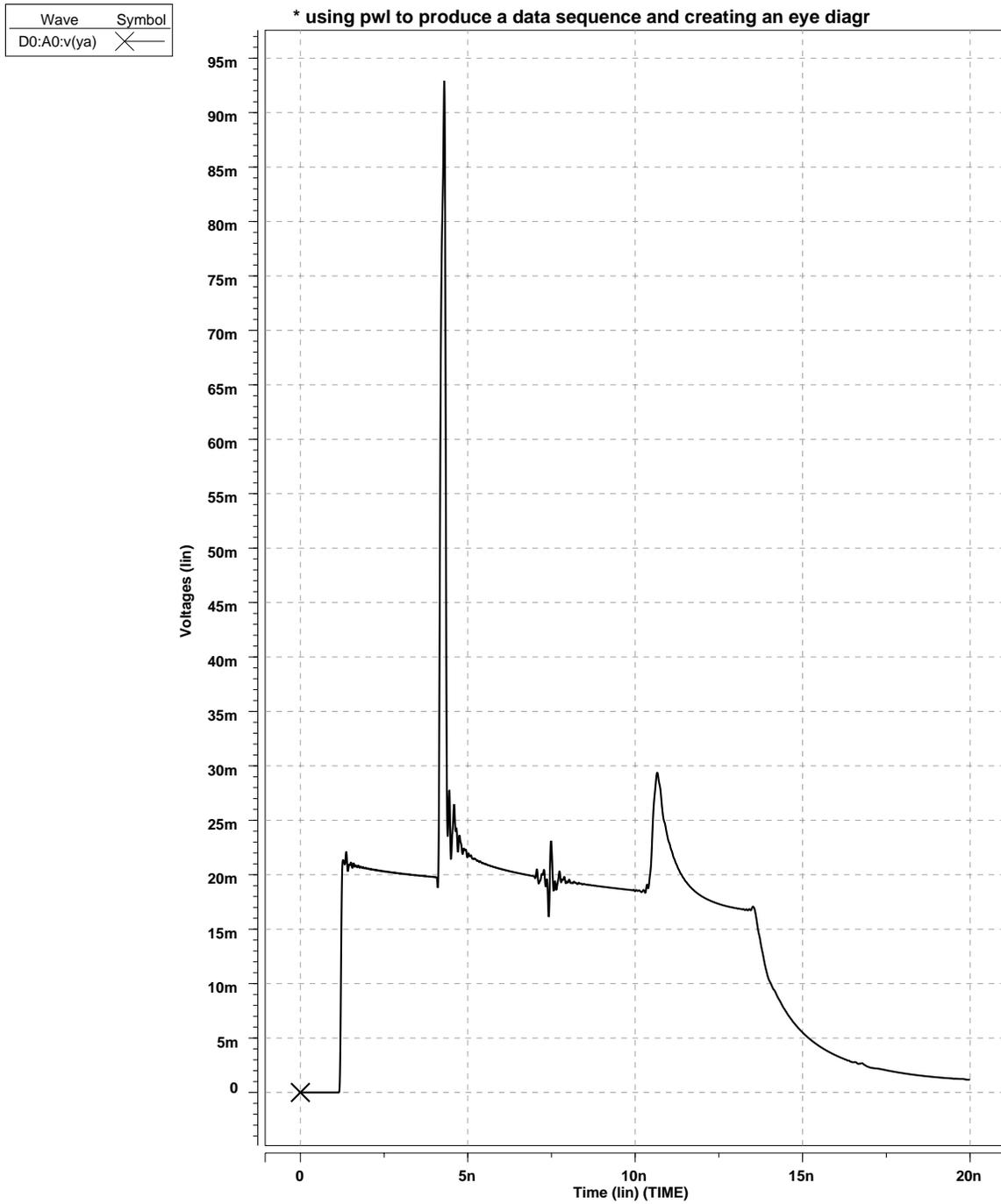


Figure 23 TDT

7. Chip package and Cost Analysis

7.1 Power Analysis

It is important to estimate power consumption to decide the package and cost. Since it is hard estimate the power in the various blocks in the link, we used the numbers from the

paper. The 4Gb/s transceiver described by M.J. Lee et. al dissipates 90mW. Since we are running at a higher bitrate but lower Vdd, the power will be different. To get the worst case power, we ignored the effect of reducing Vdd. Assuming power is proportional to the bitrate, estimated power is $90 \cdot 6/4 = 135\text{mW}$ per transceiver.

The power consumed in the core logic is hard to estimate. We will assume that the same amount of power will be dissipated in the core logic. The resulting total power is:

Line card chip: $135 \cdot 32 \cdot 2 = 8.6\text{W}$

Switch card chip: $135 \cdot 256 \cdot 2 = 69\text{W}$

7.2 Number of power pins

It is hard to estimate because we don't have enough information about the power consumption and current profile of the core logic. We will assume that we will use one power or ground for every 4 signals (2 diff pairs).

Line card: $64/4 = 16$

Switching card: $512/4 = 128$

7.3 Total number of pin

Total # of pins = Signal + power + clk + control + testing + slack

Line card (1chip) = $64 + 16 + 2 + 16 + 5 + 1 = 104$

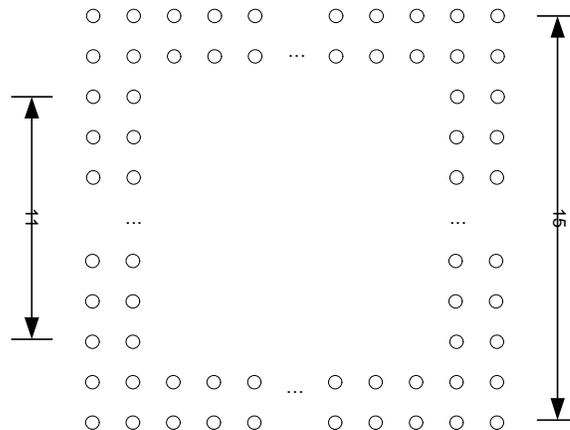


Figure 24 Line Card Chip pin layout

Switch card = $512 + 128 + 2 + 16 + 5 + 19 = 672$

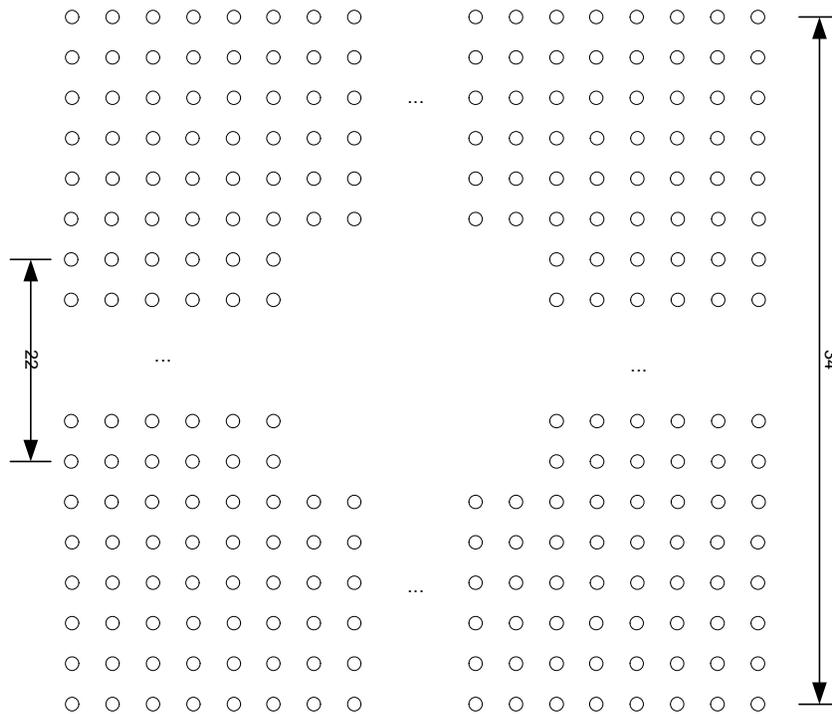


Figure 25 Switch Card chip pin layout

7.4 Size of the chip

The size is restricted by the 1mm pitch BGA pattern.

For the line card, the total number of pins is less than 36*4 (which is the minimum number of pins required for a six row layout). As we do not have enough pins to populate size rows, we use only 2 rows as seen in Figure 24. As the pin array is 15x15, the size of the chip is approximately 2cm x 2cm.

For the switch card, the pin layout is 34x34 as seen in Figure 25. The estimated size is 3.9cm x 3.9cm.

7.5 Estimate of Cost

Part	Cost	Quantity	Total Part Cost
Line Card, Switching Card	\$20	18	\$360
Line Card chip	\$40.4	16	\$646.4
Switching card chip	\$97.2	2	\$194.4
Oscillator	\$5	18	\$90
Line card power	\$172	16	\$2752
Switch card power	\$1380	2	\$2760
System Total			\$6802.8