

CONTENTS

I	Introduction	2
II	Channel Impulse Response	2
III	Pulse Response	3
IV	Cursor Magnitude	3
V	Worst Case Data Pattern	4
VI	Eye Diagram	5
VI-A	Random Eye Diagram	5
VI-B	Worst Case Eye Diagram	5
VII	3-Tap TX Finite Response Filter	6
VIII	3-tap DFE	8
IX	Differential Channel	9
X	Serializer	10
X-A	Half-Rate Clocking	11
X-B	Five Latch 2:1 MUX	12
XI	Voltage-Mode Driver	13
XI-A	Filter Design Choices and Equalization	13
XI-B	Impedance Matching	13
XI-C	De-emphasis	15
XII	Pre-driver	16
XIII	TX Simulation and Performance	16
XIII-A	TX-Side Schematics and Output	16
XIII-B	TX Eye Diagram	17
XIII-C	TX Power Consumption Breakdown	18
XIII-D	TX Performance and Specifications Summary	19
XIV	Channel Receiver Schematics and Components	19
XV	Quarter-rate Slicer	20
XV-A	RX Clock-rate	20
XV-B	Strong Arm Latch	20
XV-C	Track-and-Hold Switch	22
XV-D	Set Reset (SR) Latch	22
XV-E	Synchronizer	23
XV-F	Resistor Bank Impedance Matching	24
XVI	Clock Buffers	25
XVII	RX Simulation and Performance	26
XVII-A	RX Eye Diagram	26
XVII-B	Worst-case Sequence RX Output Recovery	27
XVII-C	PRBS Checker	28
XVII-D	RX Power Consumption Breakdown	29
XVII-E	RX Performance and Specifications Summary	29
XVIII	Conclusion	29
References		30

Design, Analysis and Simulation of an I/O Link

ELEC 403 Final Report

Isabelle André
Student ID: 12521589

Abstract

This project consists in designing, analyzing, and simulating an I/O link at the system level, and the TX and RX at the circuit level using a provided channel s-parameter file.

I. INTRODUCTION

This report describes the analysis of a differential channel's s-parameters and the design of a 4:1 serialized transmitter and receiver with 3 tap FIR equalization at 10 Gb/s using Cadence and GPKD 45nm.

In part 1, we generate the channel's impulse response and find its pulse response at 10 Gb/s. The magnitude of the 5 most important cursors are outlined. The worst case data patterns at 10 Gb/s are plotted as well as the eye diagram for a random sequence of bits and the worst case bit pattern. Finally, the Equalizer coefficients for a 3-tap TX FIR filter and DFE are calculated.

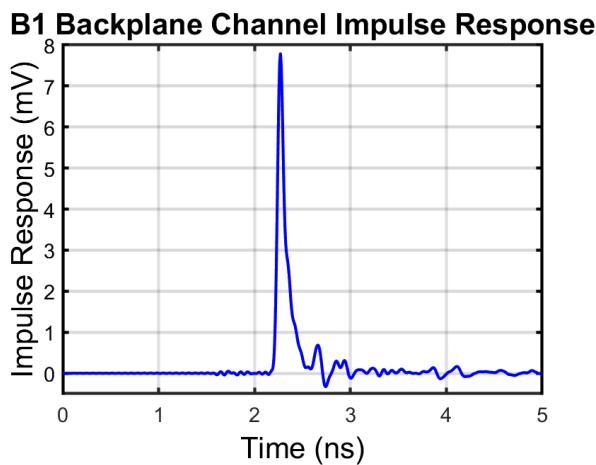
In part 2, a test bench for a differential channel was created using the provided channel to observe the cursor and post-cursor. Then a 4:1 serializer with a half clock-rate architecture. Finally, a voltage-mode driver was designed driven by a pre-driver to plot an eye diagram for PRBS7 data pattern captured over 10,000 UIs at TX output.

In the final part of the project, inputted bit sequences are analyzed and processed at the RX side of the channel. A quarter rate slicer is designed from a Strong Arm latch and an SR latch then inputted into a synchronizer. The RX output impedance is tuned to meet the termination specifications, and the worst case eye is plotted at the RX side of the channel. A generated PRBS-7 bit input sequence is checked at the output of the channel.

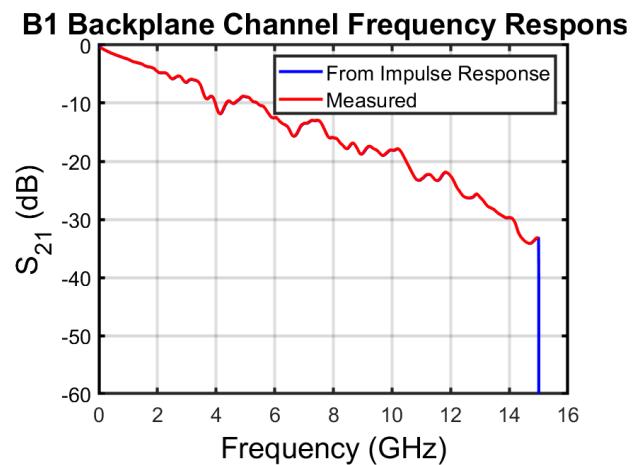
II. CHANNEL IMPULSE RESPONSE

The channel's 4-port single-ended S-parameters are first converted to 2-port mixed-mode S-parameters. We perform Inverse Fast Fourier Transform (IFFT) on measured data in the frequency domain to produce the channel's impulse response.

The impulse response of the differential channel is calculated and plotted from the provided sp4 parameters using Matlab as shown in Figure 1a. We can sanity check the response by doing a Fast Fourier Transform (FFT) as shown in Figure 1b on the impulse response and comparing it to the measured data.



(a) Channel Impulse Response at 10 Gb/s



(b) Channel Frequency Response at 10 Gb/s

Fig. 1: Backplane Channel Response at 10 Gb/s

III. PULSE RESPONSE

A pulse is generated and normalized to create a convolution with the differential channel's impulse response. The pulse is constructed to produce a 10 Gb/s channel response, therefore the unit interval (UI) is set to 100 ps. The pulse response is sampled at 1 ps.

The differential channel pulse response $p(t)$ is defined by the convolution in the time domain as

$$p(t) = \phi(t) * h(t) \quad (1)$$

where $\phi(t)$ is pulse sent through the channel and $h(t)$ is the impulse response of the channel. The result of the convolution can be seen in Figure 2, where a 1 UI pulse is inputted into the channel, resulting in the following pulse response.

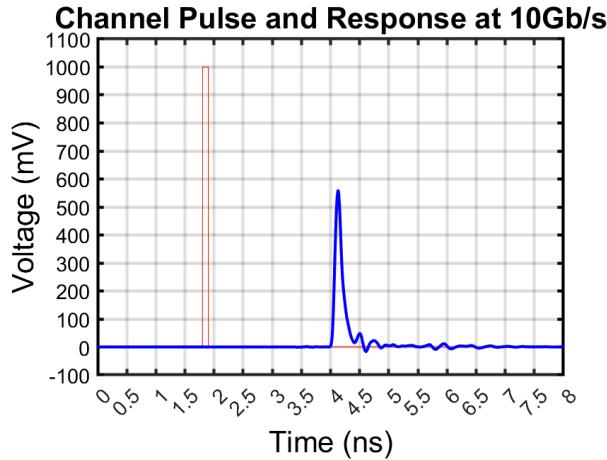


Fig. 2: Channel Pulse Response at 10 Gb/s

IV. CURSOR MAGNITUDE

Using the generated differential channel pulse response, the cursor, pre-cursors, and post-cursor can be determined. The cursor is positioned at the maximum point of the channel pulse response. From this point in time, the response is sampled forward and back at the unit interval of 100 ps. The sampled pulse response is shown in Figure 3.

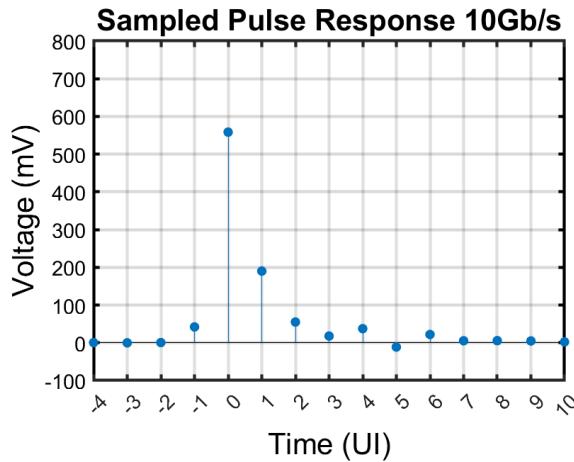


Fig. 3: Sampled Pulse Response at UI

The 5 cursors with the highest magnitude in the 10 Gb/s pulse response are shown in Table 1.

Unit Interval (UI)	Voltage (mV)
-1	42.044
0	558.479
1	189.957
2	55.016
3	37.427

TABLE I: Magnitude of The 5 Most Important Cursors

V. WORST CASE DATA PATTERN

Next, the worst case 0 and 1 bit patterns are determined by analyzing a sampled pulse response when sending a 0 or 1 bit on the 10 Gb/s differential channel. The worst case bit 1 is a summation of a 1 pulse with all negative Intersymbol Interference ISI^- , while the worst case bit 0 is a summation of a 0 pulse with all positive ISI^+ . A series of 100 bits are generated for 39 pre-cursors, the cursor, and 60 post-cursors from a mirrored channel pulse response.

To find the worst case 0, all cursors with an amplitude greater than 0 mV are set to 1 while all cursors less than 0 mV are set to 0. The cursor is set to 0. The equation for the worst case 0 response is

$$V_{WC0} = \sum ISI^+ \quad (2)$$

To find the worst case 1, all cursors with an amplitude greater than 0 mV are set to 0 while all cursors less than 0 mV are set to 1. The cursor is set to 1. The equation for the worst case 1 response is

$$V_{WC1} = cursor + \sum ISI^- \quad (3)$$

These samples are reshaped as pulses as shown in Figure 4a and 4b.

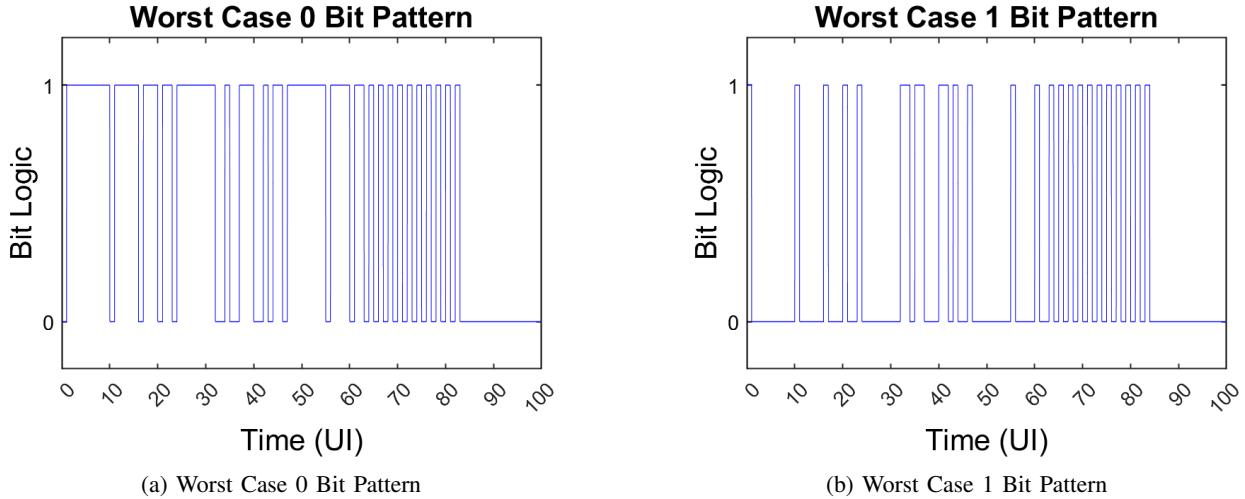


Fig. 4: Worst Case Bit Patterns

The 100 bit worst case 0 and 1 bit patterns are sent through the differential channel using a convolution with the channel's impulse response as shown in Equation 1 where the 0 and 1 bit patterns act as the input $\phi(t)$. The channel's pulse response for the worst case 0 and 1 bit patterns are shown in Figure 5a and 5b with the cursor location labeled.

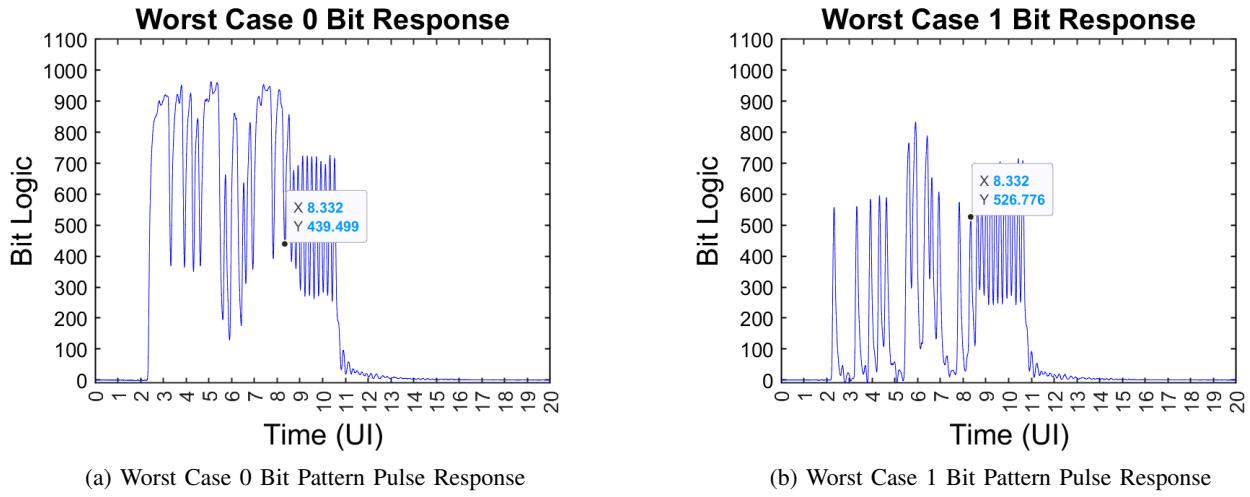


Fig. 5: Worst Case Bit Pattern Pulse Response

The worst case 0 and 1 are located at the cursor on each pulse response plot. The worst case 0 bit response is 439.499 mV and the worst case 1 bit response is 526.776 mV.

VI. EYE DIAGRAM

An eye diagram is plotted for a random sequence of bits as comparison for the eye diagram of the worst case 0 and 1 bit responses. The width and height of each eye diagrams are compared over a period of 200 ps.

A. Random Eye Diagram

A random eye diagram is plotted using 1000 randomly generated 0 and 1 bits inputted into the differential channel.

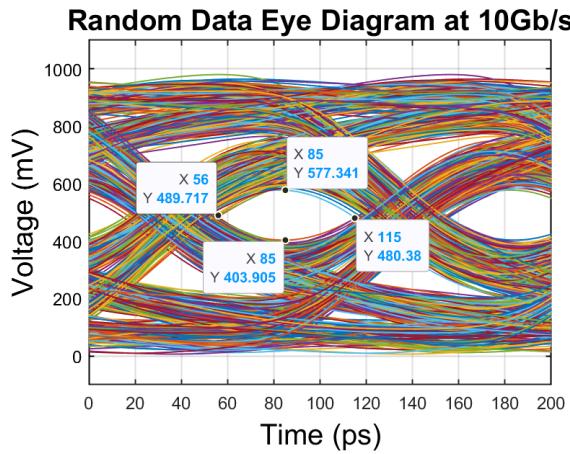


Fig. 6: Random 1000 Bit Data Eye Diagram

The eye height and width are as shown in Table 2.

Eye Width (ps)	Eye Height (mV)
59	173.436

TABLE II: Random Data Eye Width and Height

B. Worst Case Eye Diagram

The worst case eye diagram can be plotted using the worst case 0 and 1 pulse response previously generated in Section 5. The combined eye diagram for the worst case 0 and 1 responses is plotted in Figure 7.

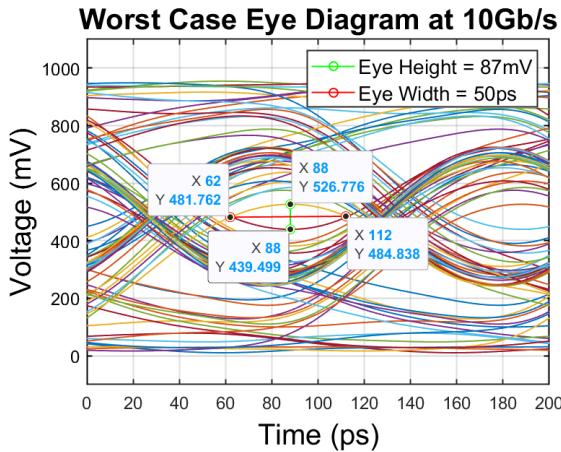


Fig. 7: Worst Case 0 and 1 Eye Diagram

The eye height and width are measured on Figure 7 and shown in Table 3 below.

Eye Width (ps)	Eye Height (mV)
50	87

TABLE III: Worst Case 0 and 1 Eye Width and Height

When comparing the 1000 bit random data eye diagram to the worst case 0 and 1 eye diagram, the eye height has considerably decreased, resulting in a much smaller eye opening.

VII. 3-TAP TX FINITE RESPONSE FILTER

The equalized output samples for an N-tap FIR equalizer is defined as

$$Y_{des}[j] = \sum_{k=-\infty}^{\infty} p[k] \cdot w[j-k] \quad (4)$$

Where the 3-tap FIR equalizer coefficients can be represented as

$$W(z) = w[0] + w[1]z^{-1} + w[2]z^{-2} \quad (5)$$

The Minimum Mean-Square Error (MMSE) method is used to generate 3 taps for a TX-Finite Response Filter using a similar equation to Equation 4 in matrix form:

$$Y_{des} = PW \quad (6)$$

where Y_{des} is the desired output of the channel, P is the channel's pulse response without equalization, and W is the 3-tap equalizer coefficient vector.

A pre-cursor tap of 1 and a TX equalization resolution of 3 bits is set for this section.

The Y_{des} matrix is a matrix of length $y = k + l - 1$ where k is the length of the pulse response, and l is the number of equalizer tap. In this case, $k = 41$ and $l = 3$. The Y_{des} matrix is therefore a 1 column matrix with 43 rows. This matrix consists of an array of zeros and a single 1 at the position of the cursor to represent the desired output of the imputed pulse. Taking the most significant cursors, 1 pre-cursor and 10 post-cursors are taken in the Y_{des} matrix.

Therefore, for a 43x1 matrix:

$$\begin{aligned} Y_{des}[0:10] &= 0 \\ Y_{des}[11] &= 1 \\ Y_{des}[12:42] &= 0 \end{aligned}$$

The P matrix is constructed as a matrix of $k + l - 1$ rows and l columns, where $k = 41$ samples and $l = 3$ taps.

$$P = \begin{bmatrix} p[0] & 0 & 0 \\ p[1] & p[0] & 0 \\ p[2] & p[1] & p[0] \\ \dots & \dots & \dots \\ p[k-1] & p[k-2] & p[k-3] \\ 0 & p[k-1] & p[k-2] \\ 0 & 0 & p[k-1] \end{bmatrix} \quad (7)$$

Using the known Y_{des} and P matrices, the W matrix is constructed as a 3x1 vector and can be solved as follows:

$$W = \begin{bmatrix} w[0] \\ w[1] \\ w[2] \end{bmatrix} \quad (8)$$

$$W = (P^T P)^{-1} \cdot P^T \cdot Y_{des} \quad (9)$$

The 3 tap coefficients generated using Matlab after solving the above operation are shown in Table 4.

w[0]	w[1]	w[2]
0	0.7143	-0.2857

TABLE IV: 3-Tap TX-FIR Coefficients

The equalized coefficients are then reshaped into an input pulse to be sent through the channel as shown in Figure 8.

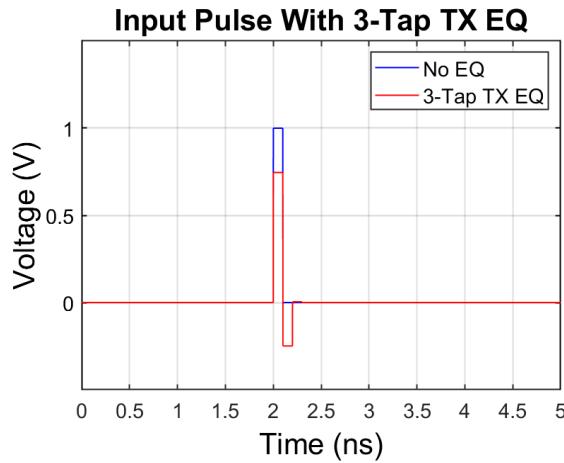


Fig. 8: 3-Tap Coefficient Input Pulse Comparison

Finally, the input pulse is convolved with the unequalized channel pulse response. The output is an equalized pulse response with attenuated cursors as shown in the pulse response and sampled response plots Figure 9a and 9b.

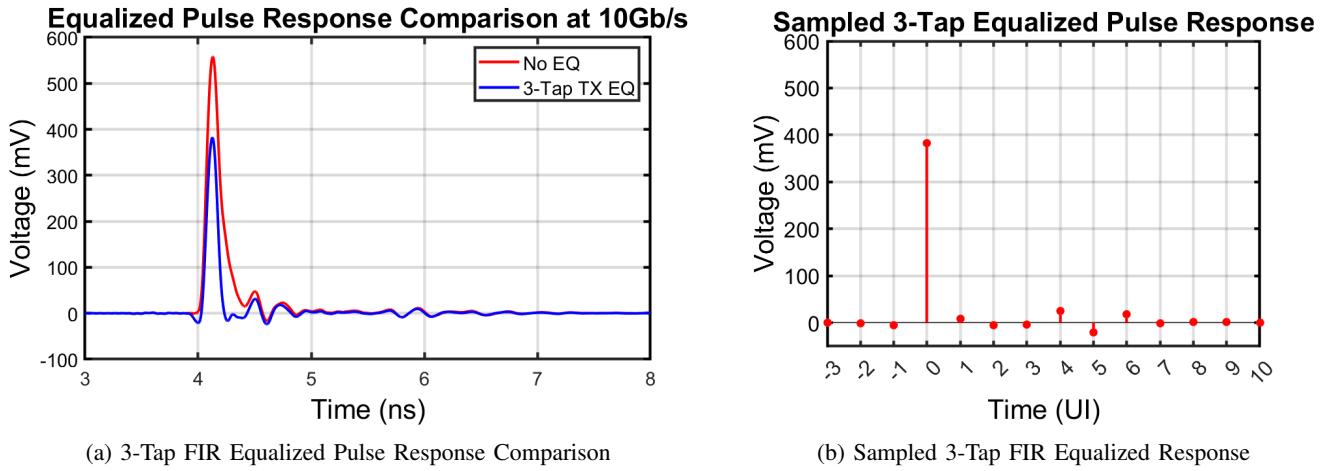


Fig. 9: 3-Tap TX-FIR EQ Response

The 5 cursors with the highest magnitude in the sampled pulse response and their UI location are shown in Table 5.

Unit Interval (UI)	Voltage (mV)
0	382.696
1	9.316
4	25.6651
5	-19.81
6	18.171

TABLE V: Magnitude of The 5 Most Important Cursors

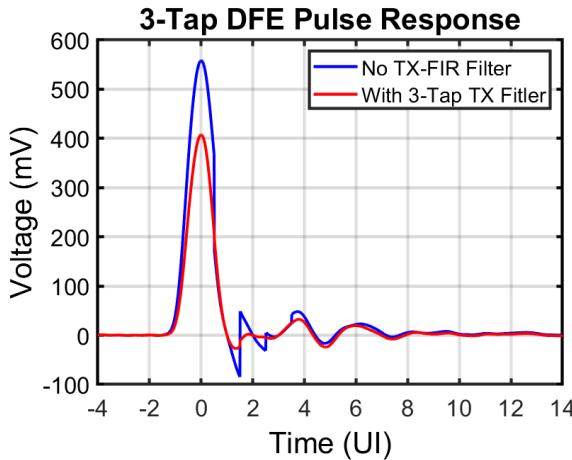
VIII. 3-TAP DFE

A Decision Feedback Equalizer (DFE) is implemented at the receiver of the differential channel. This prevents noise from being enhanced. For an 3-Tap DFE filter, the tap coefficients are the 3 first post-cursors of the unequalized pulse response. As a DFE filter can only eliminate post-cursor ISI at the slicer input, the DFE filter is often used in conjunction with a 3-Tap DFE filter. In this case, the DFE filter is applied to the channel after the pulse response is processed by the TX-FIR filter. The 3-tap coefficients calculated using Matlab for the DFE filter with and without a 3-tap TX-FIR filter are shown in Table 6.

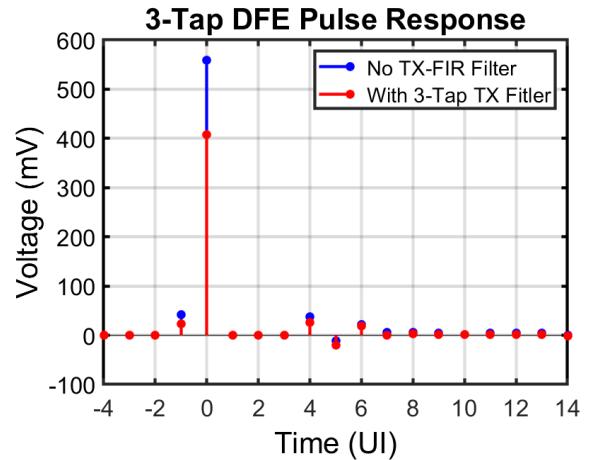
Filters	w[0]	w[1]	w[2]
No TX-FIR	0.095	0.0275	0.0089
With TX-FIR	0.0045	-0.0014	-0.0007

TABLE VI: 3-Tap DFE Filter Coefficients

Finally, the DFE filter is applied once with a TX-FIR filter, and once without to compare the resulting channel pulse response. Both cases are plotted in Figure 10a and 10b for their pulse response and sampled response.



(a) 3-Tap DFE Equalized Pulse Response Comparison



(b) Sampled 3-Tap DFE Equalized Response

Fig. 10: 3-Tap DFE Equalized Response

The 5 cursors with the highest magnitude in the DFE sampled pulse response are shown in Table 7. It can be observed that only the post-cursor ISI are eliminated in the sampled response when only filtered by DFE.

Unit Interval (UI)	No TX-FIR Filter Voltage (mV)	3-Tap TX-FIR Filter Voltage (mV)
-1	42.044	23.219
0	558.479	407.727
4	26.169	20.169
5	-11.316	-19.700
6	21.997	19.511

TABLE VII: Magnitude of The 5 Most Important Cursors

IX. DIFFERENTIAL CHANNEL

Using the provided ELEC403 Cadence library and differential channel cell, the testbench shown in Figure 11 was created. The differential channel produces an impulse response as plotted from the same provided sp4 parameters as Mini-Report 1.

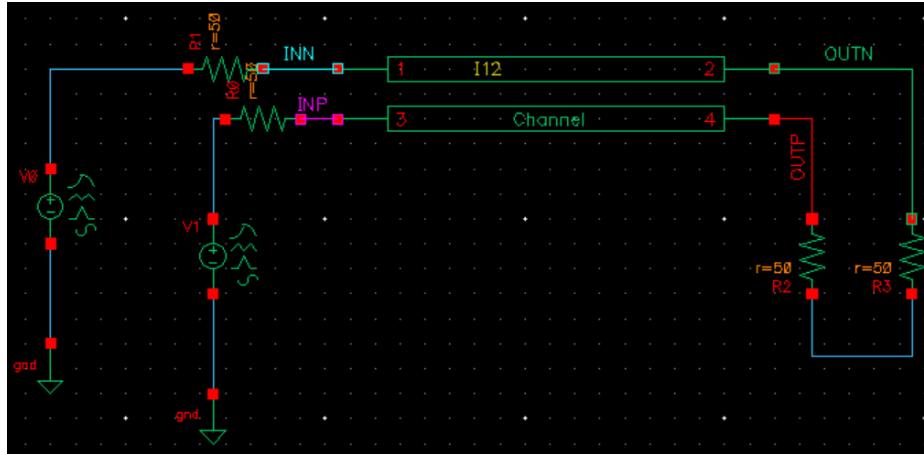


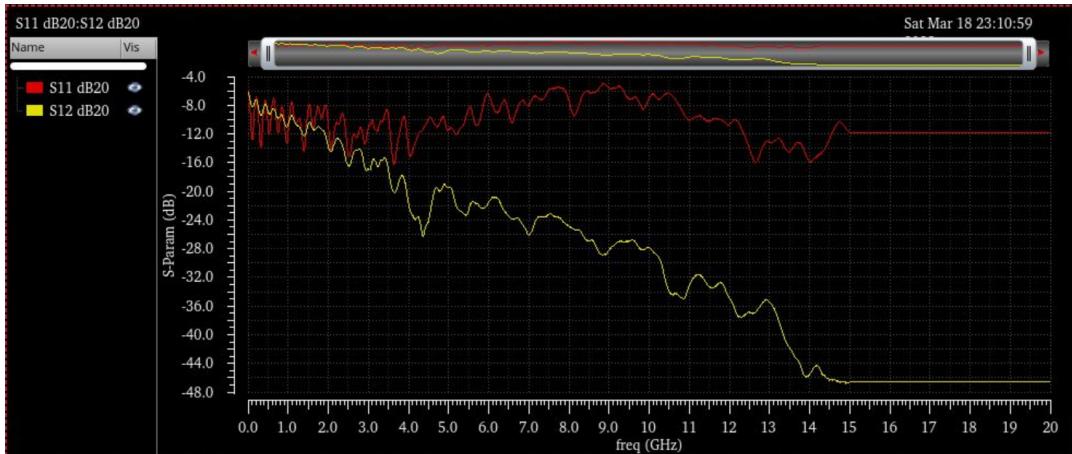
Fig. 11: Differential Channel Testbench

Next, we input two pulse inverse of one another through each channel, terminated by 50Ω each to observe the differential pulse response in its frequency and time domain. The channel's frequency response and s-parameters behaviour can be observed in Figure 12a. The pulse response in time domain and its cursors are shown in Figure 12b.

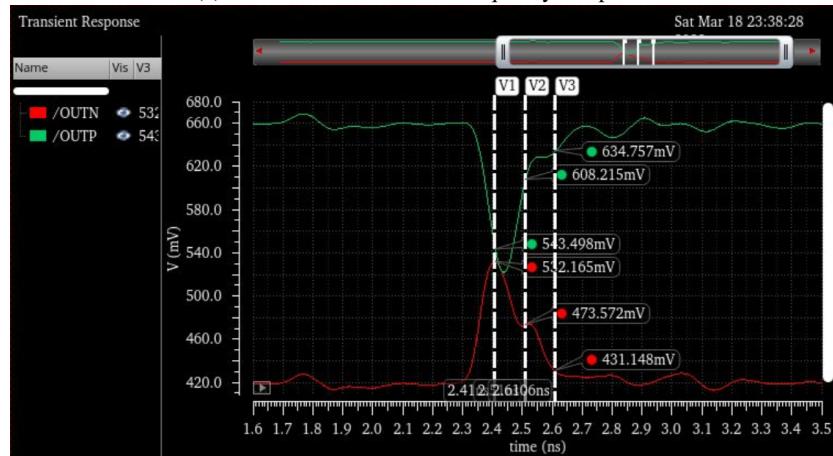
The magnitude three most important cursors are shown in Table 1.

Time (ns)	Voltage (mV)
2.41	532
2.51	473
2.61	431

TABLE VIII: Magnitude of The 3 Most Important Cursors



(a) S-Parameters Channel Frequency Response



(b) Differential Channel Pulse Response and Cursors

Fig. 12: Backplane Channel Response at 10 Gb/s

X. SERIALIZER

Next, a 4:1 serializer is designed using a half-clock rate architecture and a 5 Latch 2:1 MUX topology as shown in Figure 13.

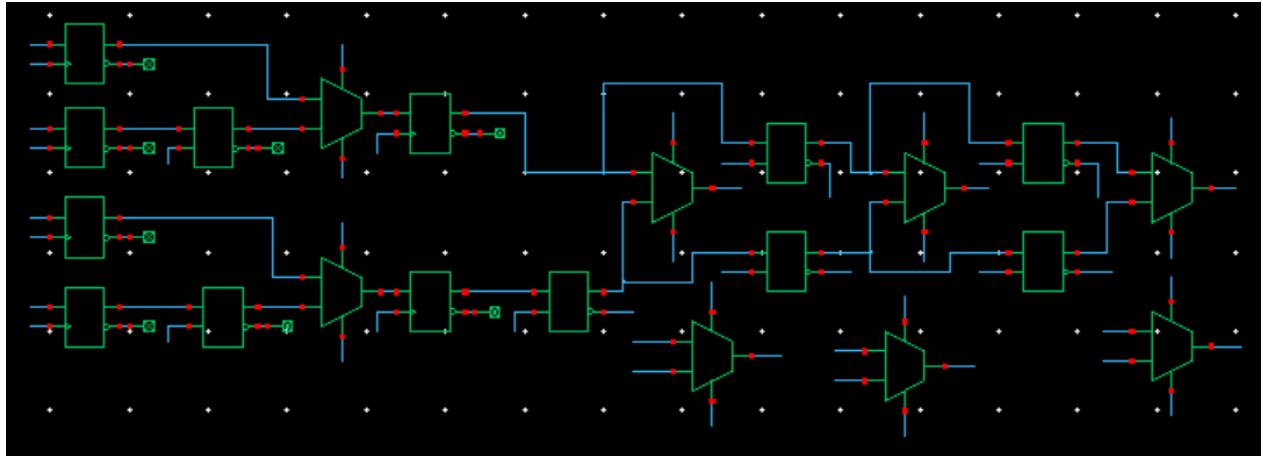


Fig. 13: 4:1 Serializer

This serializer outputs 6 signals, including a pre-cursor, cursor, and post-cursor pulse and their inverted signal, each delayed by one half-clock cycle. The delayed cursor signals are simulated using a flip-flop at the input of each additional output MUX. A testbench was created for the serializer with four simulated inputs to be serialized, set as follows:

$$\begin{aligned} IN0 &= 1V \\ IN0 &= 1V \\ IN0 &= 1V \\ IN0 &= 0V \end{aligned}$$

As shown in Figure 14, after the first 2 cycles of data propagation, the inputs are serialized with each cursor output delayed by a half-clock cycle, by approximately 100ps.

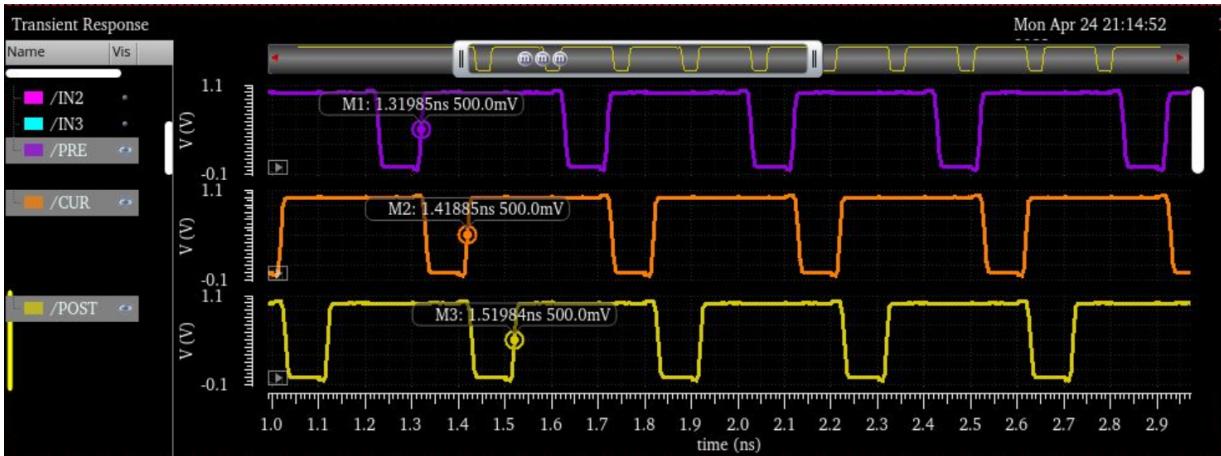


Fig. 14: 4:1 Serializer Pre-cursor, Cursor, and Post-cursor Outputs

A. Half-Rate Clocking

Clock generation and distribution in the serializer can limit the data rate on the transmission side for CMOS processes. In full-rate clocking architectures, the clock is sampling data only at a specific location, and the output is well defined and stable. The clock period is equal to the UI. However, the minimum UI at transmission is limited by TX clocking with a timing overhead of approximately 3 FO4, limiting how fast data can be transmitted.

Half-rate clocking allows the increase of maximum achievable data rate, multiple phase of a slower clock can be used to multiplex or serialize the data in parallel. The half-clock rate architecture uses two clock phases separated by 180 degrees as select signals to a MUX. The odd and even data are sampled at the rising edge of the half clock at a 50% duty cycle. The UI is equal to half a clock period.

$$UI = \frac{T_{ck}}{2} \rightarrow T_{ck} = 2UI \quad (10)$$

$$f_{ck} = \frac{f_b}{2} \rightarrow f_{Nyq} = \frac{f_b}{2} \quad (11)$$

Therefore the clock frequency is half of the bitrate, at 5 Gb/s and is said to be running at half rate as shown in Figure 15.

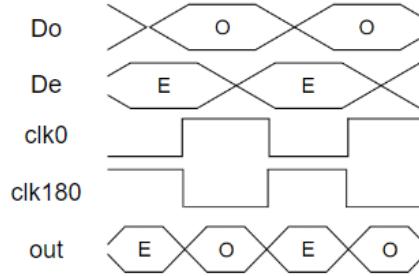


Fig. 15: Half-rate Clock Sampling

The final re-timing flip-flop is eliminated, therefore it is important for the opposite clock signals to be non-overlapping and have a duty cycle of exactly 50%, eliminating duty cycle distortion.

B. Five Latch 2:1 MUX

It is important to choose a good 2:1 MUX topology to implement for a highspeed circuit to reduce glitches and prevent duty cycle distortion to occur in the serializer. A 2:1 MUX with a select and select bar signal can be defined as $Y = SD_1 + \bar{S}D_0$. A 2:1 MUX with inverted output was designed by shorting the outputs of two tristate inverters, then inverting the output as shown in Figure 16.

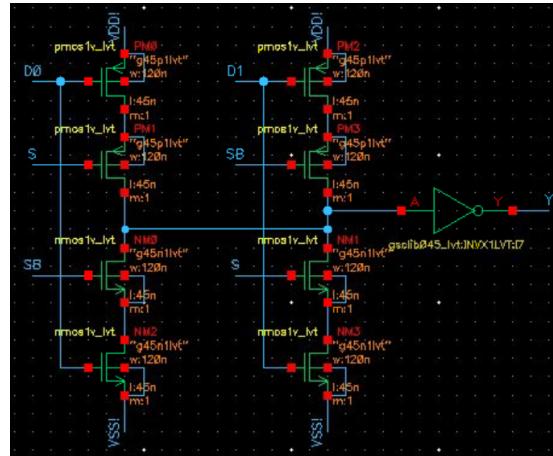


Fig. 16: 2:1 Inverted Multiplexer

With a half-clock architecture, a MUX is prone to glitches if the UI and data are not perfectly aligned. Therefore we use 5 latches, or a 2 flip-flop and 1 latch MUX topology as shown in Figure 17.

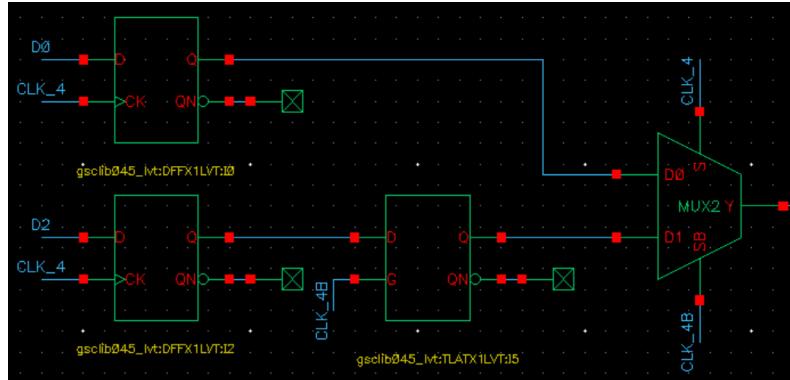


Fig. 17: 5 Latch offset Multiplexer

The data odd and even inputs are re-timed using a flip-flop on each input, then a latch on the odd input to prevent glitches or pulse width distortion.

XI. VOLTAGE-MODE DRIVER

Next, a high-swing voltage-mode (VM) driver is designed with an impedance tuning range of 30% of the 50Ω channel termination resistance. The required tuning impedance ranges from $35\text{-}65\Omega$.

A. Filter Design Choices and Equalization

Cursors are de-emphasized using a 3 tap TX-FIR filter with 4 bits of resolution in the voltage-mode driver. While the 3-Tap DFE provided a better channel performance and eliminated a prominent pre-cursor in the pulse response as shown in Figure 10.3 a) from Part 1, this level of improvement is not enough to justify the added complexity that its implementation would bring into this project. The added feedback loop would require strict timing constraints that must be met requiring precise tuning. Furthermore, the tap coefficients calculated for a 3-Tap DFE Filter are magnitudes smaller than the coefficients for the 3-Tap TX-FIR as shown in Table IV from Section VII. Therefore, a simpler filter was chosen to be used. The 3-Tap TX-FIR filter showed adequate equalization for cursor and post-cursors. 3 taps were chosen as 2 taps would likely not provide enough equalization to eliminate the most important cursors, and 4 taps would add much more complexity with an additional coefficient to implement, as well as an additional segment for de-emphasis in the voltage-mode driver.

Previously in Part 1 of the project, a TX equalization resolution of 3 bits were used. Once the tap coefficients calculated, it was noticed that one of the coefficients were set to zero as shown in Table IV, suggesting that a higher bit resolution would be required. Therefore, we choose 4 bits of resolution to implement our voltage-mode driver with de-emphasis. The new 3-tap coefficients with 4 bits of resolution are calculated using Matlab and shown in Table 9.

w[0]	w[1]	w[2]
-0.0667	0.7333	-0.2000
1/15	11/15	3/15

TABLE IX: 4 Bit Resolution 3 Tap RX-FIR Filter Coefficients in mV

Each tap coefficient is converted into a fraction and add up to 1. The fraction determines the sizing ratio of a segment compared to a standard inverter. In this case, the denominator is 15, therefore the filter equation can be re-written as follows:

$$y = -\frac{1}{15}d[n+1] + \frac{11}{15}d[n] - \frac{3}{15}d[n-1] \quad (12)$$

A negative tap coefficient signifies that the segment should be driven by an inverted input, while a positive tap should be driven by the unchanged original cursor.

B. Impedance Matching

The NMOS and PMOS in each VM Driver segment are sized such that the impedance of each transistor when turned ON are equivalent to the channel series termination $R_U = R_D = R_T$ as shown in Figure 18.

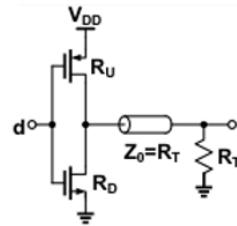


Fig. 18: Segment Impedance Matching and Sizing

To control the VM driver's output impedance, we split the driver into N slices where K slices are enabled, such that

$$R_T = (NR_N + NR_P)/K \quad (13)$$

where R_T is the termination resistance, and R_N and R_P are the NMOS and PMOS side resistors respectively, as shown in Figure 19. These resistors are selected such that the output resistance seen remains constant when either the driver slice's NMOS or PMOS is ON.

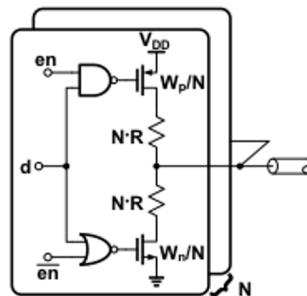


Fig. 19: VM Driver Split into N Segments

To match the tuning range, we choose each VM driver slice to have a 350Ω output impedance, resulting in $N = 10$ slices in total. Table 10 shows the expected impedance per slice calculated by measuring the parallel output impedance per slice enabled.

Slices	1	2	3	4	5	6	7	8	9	10
Impedance (Ω)	350	175	116.6	87.5	70	58.3	50	43.7	38.8	35

 TABLE X: Impedance Per VM Driver Slice for 10 Slices at 350Ω Each

A Z-Parameter sweep was conducted to match a single slice's output impedance as close as possible to the chosen 350Ω resistance as shown in Figure 20. When the NMOS is ON, the output impedance is 348.5Ω . When the PMOS is ON, the output impedance is 355Ω .

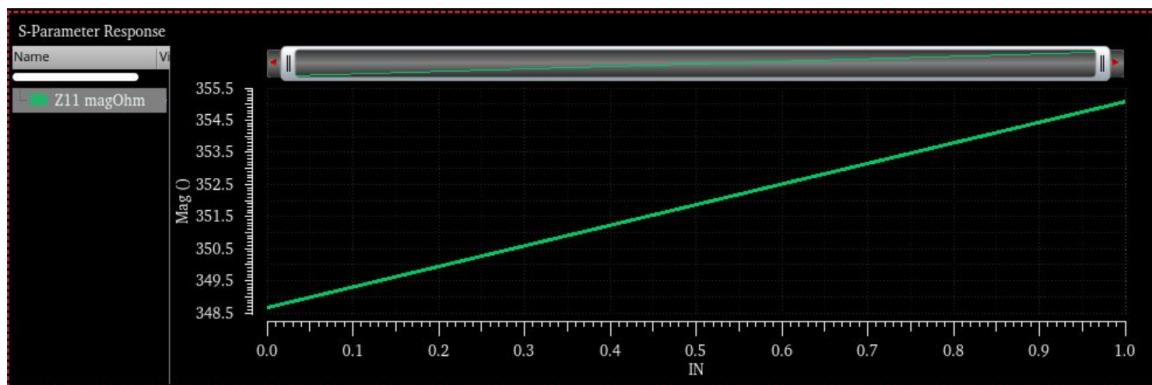


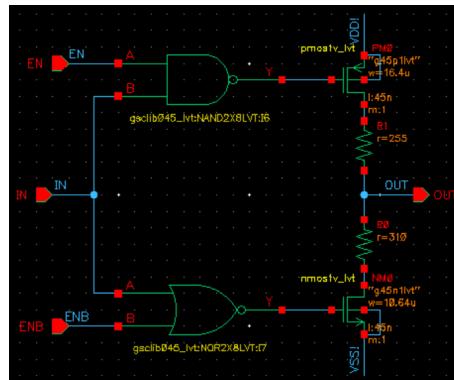
Fig. 20: Z-Parameter Sweep Impedance Matching

Using a parameter sweep, the transistor widths and R_P and R_N values minimizing this impedance range are shown in Table 11.

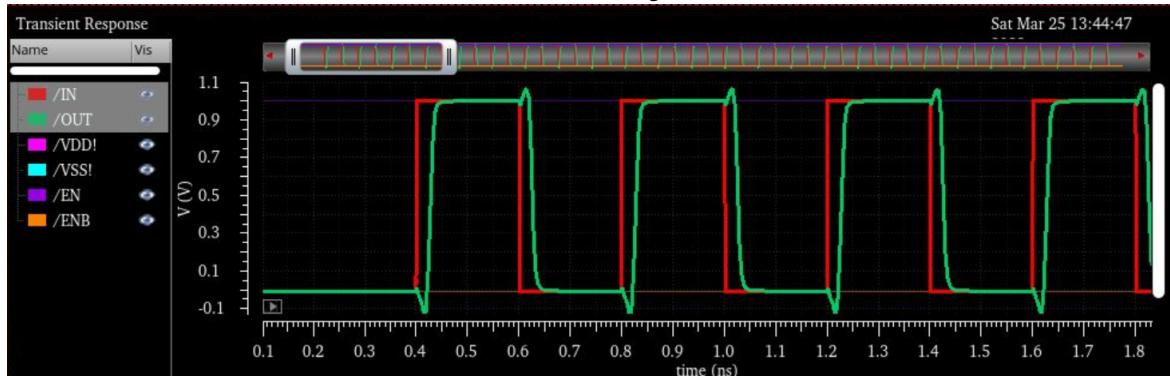
$W_P (\mu)$	$W_N (\mu)$	$R_P (\Omega)$	$R_N (\Omega)$
16.4	10.64	255	310

TABLE XI: Slice PMOS and NMOS Impedance Matched Width and Resistance

Figure 21a shows a VM driver segment designed with a NAND and NOR gate with sized transistors and resistances after impedance matching. Figure 21b shows the segment's symmetric rise and fall time outputs when inputting a pulse, showing that impedance is well matched.



(a) VM Driver Segment



(b) VM Driver Segment Rise and Fall Time

Fig. 21: Impedance Matched VM Driver Segment

C. De-emphasis

Next, we create a VM driver slice by connecting 3 segments in series for each cursor to attenuate as shown in Figure 22.

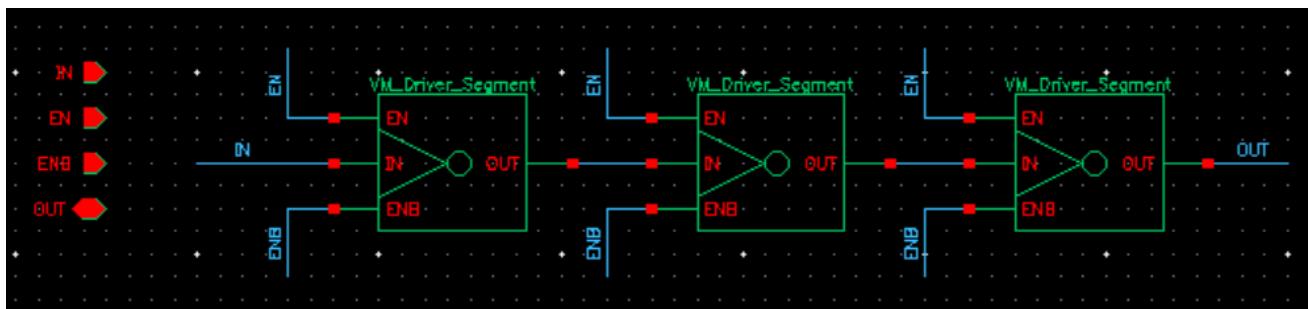


Fig. 22: VM Driver Slice

A transient simulation is created with a simple data pattern enabling a slice and inputting 3 delayed pulses as pre-cursor, cursor, and post-cursor. Figure 23 clearly shows 3 different voltage levels for attenuation of the cursors.

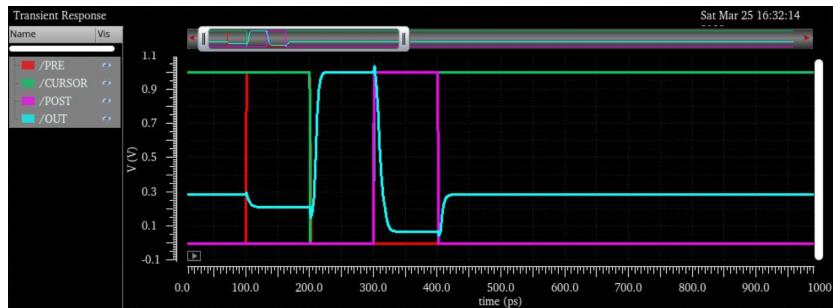


Fig. 23: VM Driver Slice Cursor Voltage Levels

Finally, we create our full VM driver shorting all slice outputs together. Another parameter sweep is run to observe the output impedance of the VM driver. Figure 24 shows the output impedance range as different number of slices are enabled or disabled, with 7 slices being the nearest to the termination resistance of 50Ω .

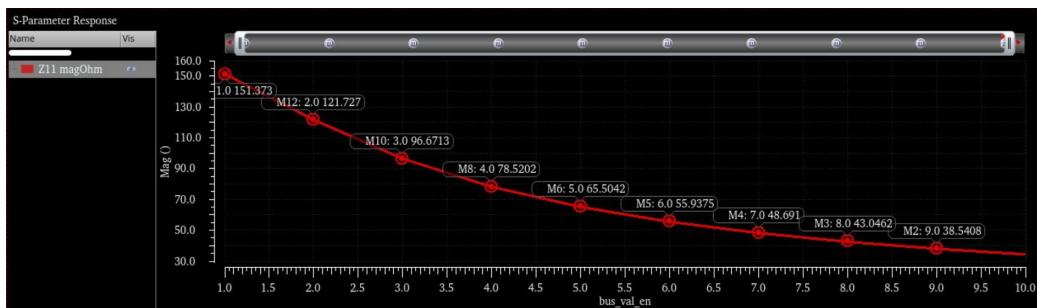


Fig. 24: VM Driver Slice Enabled Output Impedance

XII. PRE-DRIVER

The VM driver inputs by a pre-driver as shown in Figure 25. The pre-driver consists of 6 chains of two inverters sized as multiples of FO4 to drive each pre-cursor, cursor, and post-cursor VM driver input, as well as their inverted signal.

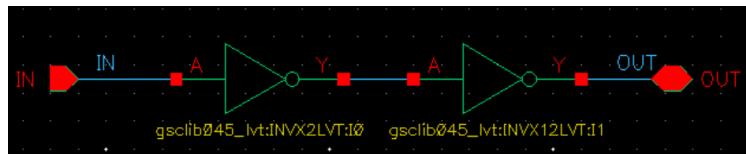


Fig. 25: Two Inverter Pre-driver For Cursor Input

XIII. TX SIMULATION AND PERFORMANCE

A. TX-Side Schematics and Output

All components are linked together, including the 4:1 serializer, pre-driver for each voltage-mode driver input, and one voltage-mode driver for each of the differential channel inputs. The output of the differential channel is terminated by two 50Ω resistors in series for a total of 100Ω . The serializer inputs are linked to a PRBS source to generate a random data pattern. Figure 26 shows the voltage mode driver and channel set up and the voltage sources acting as bit inputs to the 4:1 serializer, with each of its outputs connected to a pre-driver.

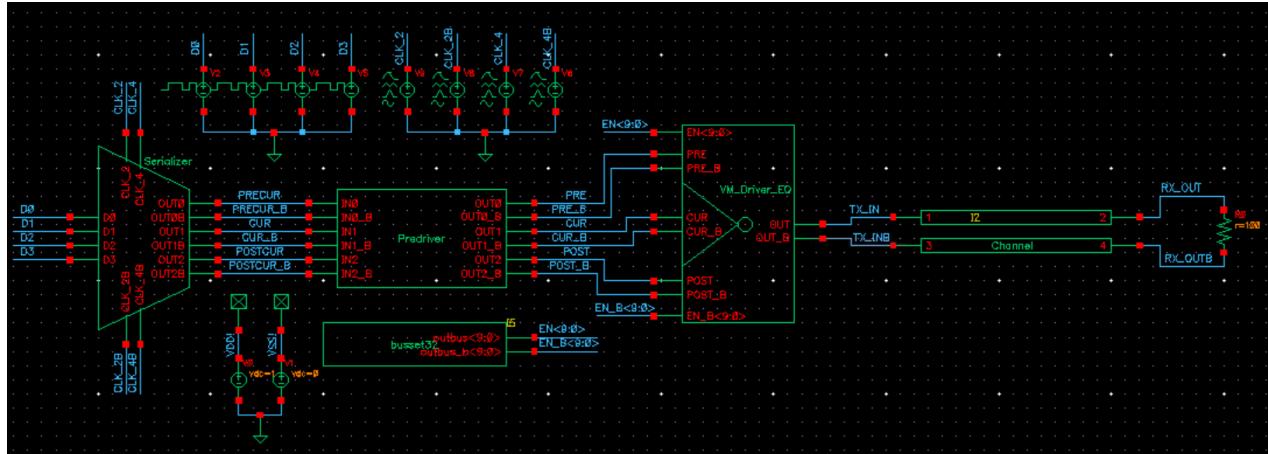


Fig. 26: TX Simulation Differential Channel Testbench

A transient waveform is run showing the alternating data transition at the TX side for 6 UI. Figure 27 shows the differential signal at the channel's output.

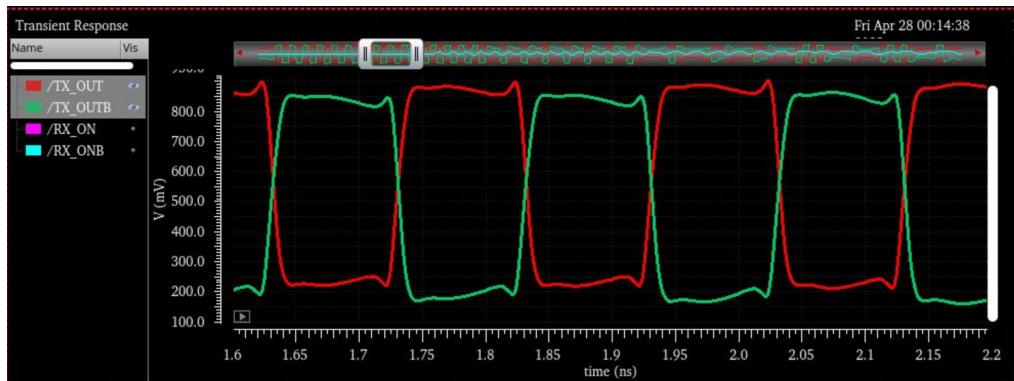


Fig. 27: Transient Simulation

B. TX Eye Diagram

First, a transient simulation is run for $1\mu s$ (10,000 UI) to generate an inputted PRBS-32 data pattern eye diagram. Figure 28 shows the eye diagram generated at the TX side output for 10,000 UI (approximately 1us). Table 12 shows the eye's approximate height and width.

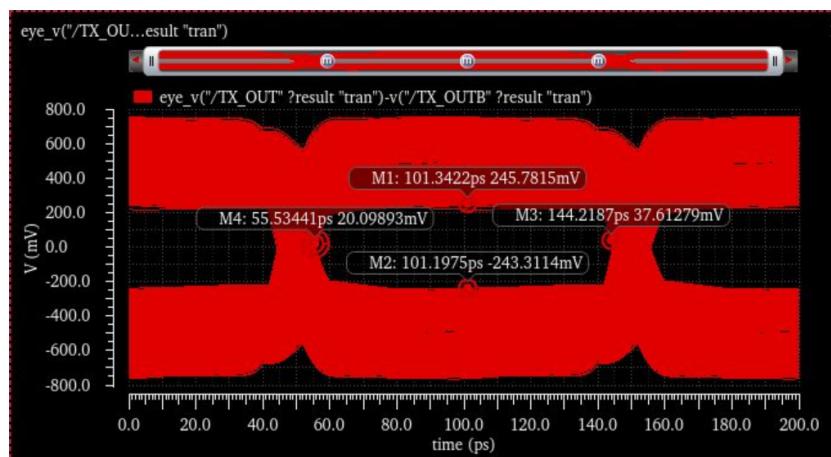


Fig. 28: PRBS 10,000 UI Eye Diagram

Height (mV)	Width (ps)
489.0929	88.6843

TABLE XII: PRBS 10,000 UI Eye Height and Width at TX Output

Next, a transient simulation is run for the worst-case bit input. This bit pattern was generated in Section 5 using Matlab. This sequence represents the worst-case input that would sum a 1 pulse with all negative ISI^- or sum all 0 pulses with all positive ISI^+ , as represented in Figure 4.

Figure 29 shows the worst case eye diagram plotted from the worst-case bit input shown in Table 13. As the bits are to be inputted into a 4:1 serializer, the worst-case input are split across 4 data streams inputted in tandem as follows.

TABLE XIII: Worst-case Bit Input Streams

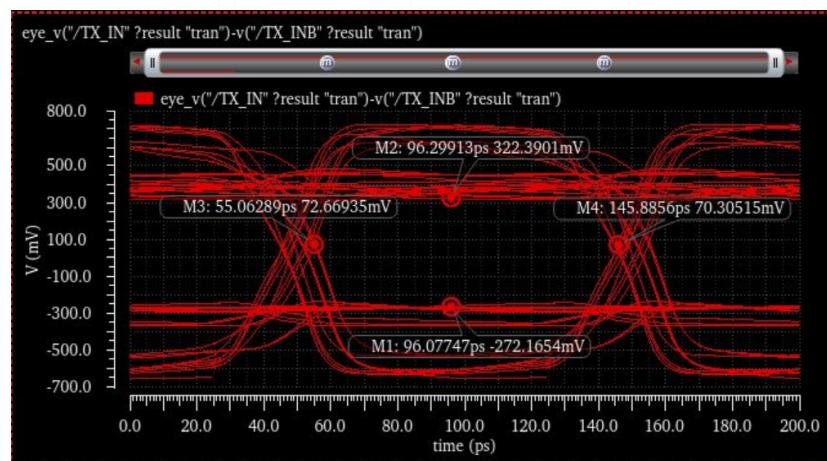


Fig. 29: Worst-case Eye Diagram

Table 14 shows the worst-case eye's approximate height and width.

Height (mV)	Width (ps)
484.5555	84.8227

TABLE XIV: Worst-case Eye Height and Width at TX Output

C. TX Power Consumption Breakdown

Knowing that for the instantaneous power consumption in Watts of a component:

$$P_{VDD} = I_{VDD}(t)V_{VDD}(t) \quad (14)$$

and that the average power is

$$P_{avg} = \frac{E}{T} = \frac{1}{T} \int_0^T P(t) dt \quad (15)$$

where T is time, and E is energy in Joules defined in energy/bit as

$$E = \int_0^T P(t)dt = \frac{1}{10Gb/s}P_{avg} \quad (16)$$

an approximate power and energy breakdown for individual components can be calculated. A 10ns transient simulation is run for each component including the serializer, pre-driver, voltage-mode driver and overall TX transmitter to determine the average current drawn at $V_{DD} = 1V$ and $V_{SS} = 0V$. As $V_{DD} = 1V$, the average power drawn will be equal to the average current drawn. Table 15 shows the power and energy breakdown for each component calculated using the above formulas and the current draw transient simulation.

Component	Average Power (mW)	Energy per Bit (pJ/bit)
Serializer	0.302	0.0302
Pre-driver (6 units)	8	0.8
Voltage-mode Driver (2 units)	22.4	2.24
TX Transmitter	34.8	3.48

TABLE XV: Power and Energy Component Breakdown

D. TX Performance and Specifications Summary

A summary of the full TX transmitter specifications met are included in Table 16, and a summary of the transmitter's performance is included in Table 17.

Description	Specification
Data Rate	10Gb/s
DC Sources	Single Current Source, Multiple Voltage Sources
DC Voltage Source(s)	Multiple
VDD	1V
PDK	45nm
Max W or L	700 μ m
Synchronizer Output	20 fF Capacitor
NMOS Body	GND
Tuning Range	30% of ideal termination (30-65 Ω)

TABLE XVI: TX Specifications

Metric	Performance
Worst-case Eye Height	484.5555 mV
Worst-case Eye Width	84.8227 ps
TX Power Consumption	34.8 mW
TX Energy per Bit Consumption	3.48 pJ/bit

TABLE XVII: TX Performance

XIV. CHANNEL RECEIVER SCHEMATICS AND COMPONENTS

On the receiver side of the channel, several blocks were designed to process and check the output of the channel, including a quarter-rate slicer, a synchronizer, a resistor bank for impedance matching, and clock buffers. A PRBS-7 Checker was attached to the output of the synchronizer to confirm the correctness of a PRBS-7 generated input bit stream. Figure 30 and Figure 31 show the quarter-rate slicer and synchronizer with buffers, the clock buffers, resistor bank, and 4 PRBS-7 Checkers.

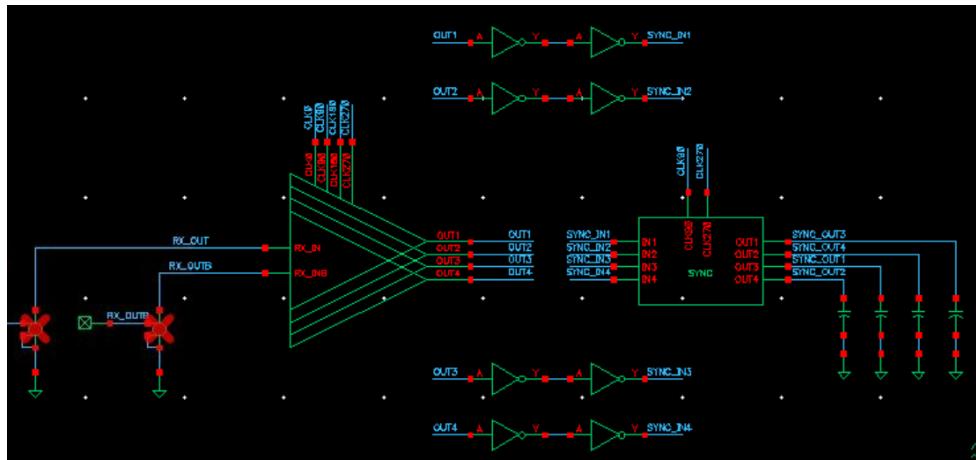


Fig. 30: RX Schematics Connections

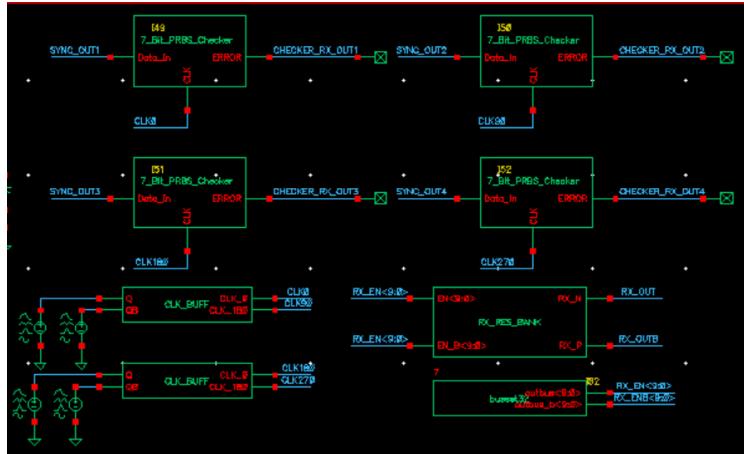


Fig. 31: RX Schematics Connections

XV. QUARTER-RATE SLICER

A. RX Clock-rate

To drive the quarter-rate slicer, a quarter-rate clock was chosen as the bandwidth for Track-and-Hold switches must be minimum 3 times the frequency. This also allows more timing room for slight inaccuracies and the 4 different Strong Arm latch states.

B. Strong Arm Latch

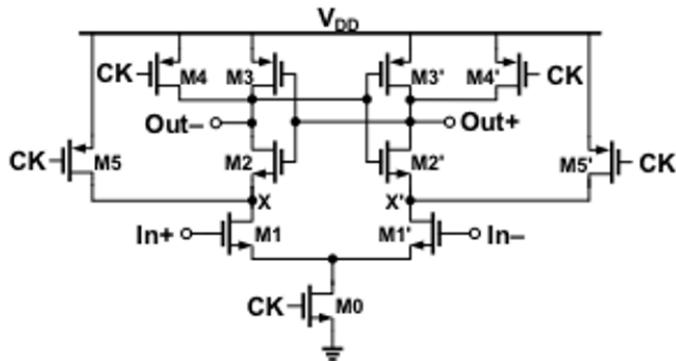


Fig. 32: RX Schematics Connections

A Strong Arm latch is a regenerative amplifier as shown in Figure 32 (provided by ELEC 403 slideset 8). It samples the continuous input at clock edges and resolves the differential to a binary 0 or 1. The operation of a Strong Arm latch includes 4 phases:

- Reset:** During the reset phase, $CLK = 0$, and nodes X , X' , $OUT+$, and $OUT-$ are all pre-charged to VDD .
- Sampling:** During the sampling phase, the $M1$ pair discharges nodes X , X' to $VDD - VTN2$. The $M2$ pair begins discharging nodes $OUT+$ and $OUT-$.
- Regeneration:** During the regeneration phase, the cross-coupled inverters amplify and attenuate $OUT+$ and $OUT-$ by positive-feedback. The PMOS turn on at time 1 until a decision at time 2.
- Decision:** One of the two NMOS and PMOS in each pair transition to linear region. The differential output can now be translated to a binary 1 or 0.

These phases are summarized and represented by Figure 33's timing diagram over one clock cycle.

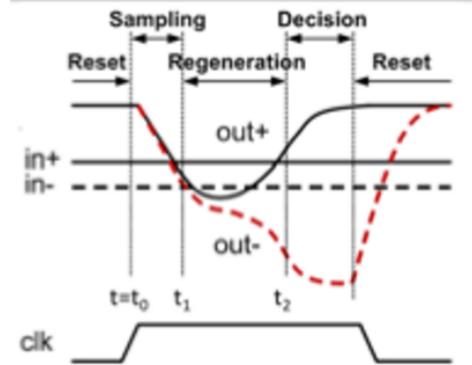


Fig. 33: Strong Arm Latch 4 Phases

The NMOS and PMOS sizing for the Strong Arm latch are as detailed in the thesis "Desgin and and Measurement of StrongARM Comparators" as a basis, then tuned as needed to meet the timing requirements. Figure 34 shows the designed Strong Arm latch, and Table 18 shows the sizing for the transistors. Sizings are adjusted as needed to obtain an acceptable driving strength. Track-and-hold switches are also often used in conjunction with Strong Arm latches, and will be explored in the following subsection.

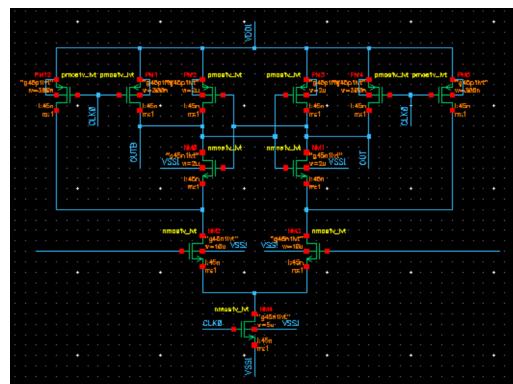


Fig. 34: Strong Arm Latch

Transistor	Sizing (μm)
M5, M5', M4, M4'	0.3
M3, M3', M2, M2'	2
M1, M1'	10
M0	5

TABLE XVIII: Strong Arm Latch Transistor Sizing

Figure 35 shows a PRBS input with an amplitude of 100 mV to simulate the conditions of the input signal from the RX side of the channel. The 4 phases of the SAL are visible and clocked to a single clock, without a track-and-hold.

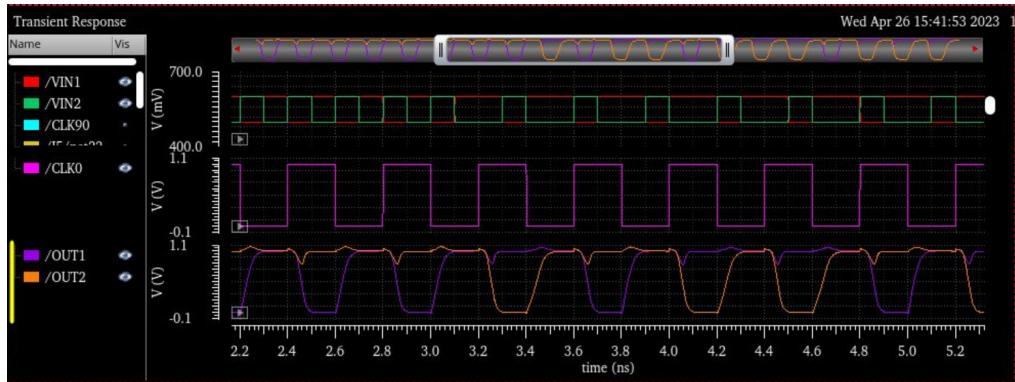


Fig. 35: Strong Arm Latch

C. Track-and-Hold Switch

Next, the track-and-hold switch is added on each side of the Strong Arm latch as shown in Figure 36. There are multiple ways to design a track-and-hold switch. We opt for a transmission gate design with two opposite clocks. The addition of a T/H switch in each slice reduces the loading, and helps setting up the timing aperture for the comparator. Jitter impact is also reduced. The transistors in the T/H switches are kept at a relatively low size similarly to M5 and M5' to avoid affecting the 4 timing of the 4 phases.

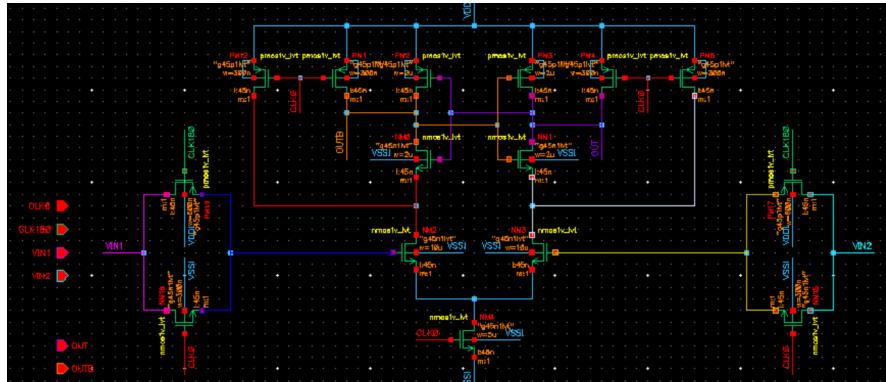


Fig. 36: Strong Arm Latch with Track and Hold Switch

D. Set Reset (SR) Latch

A Set Reset (SR) latch is used to hold the output data during the precharge stage of the SAL. Figure 37 shows the chosen circuit topology, consisting of two cross coupled NAND gates.

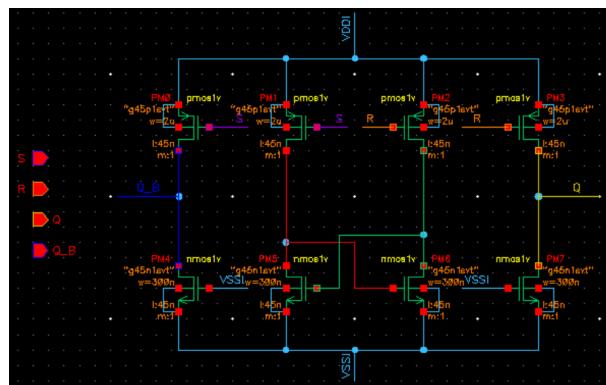


Fig. 37: SR Latch

The actions taken by the SR latch can be summarized by Table 19. The output of an SR latch must never both reach zero at the same time. Two 1 inputs would result in the previous stage being held.

S	R	Action
0	0	Not Allowed
0	1	$OUT = 1$ $OUT_B = 0$
1	0	$OUT = 0$ $OUT_B = 1$
1	1	No Change

TABLE XIX: SR Latch States

The inputs are buffered first as the SR latch is too weak to be driven without. A testbench was designed for a single SR latch to observe the actions as a response to the inputs. A combination of inputs are inputted into a testbench, yielding the waveform shown in Figure 38. This waveform supports Table 17 and shows the latch's behaviour when $S = 1$ and $R = 0$, then $S = 0$ and $R = 1$, and finally when both inputs are 1. A slight bump in voltage is seen in this stage, but the voltage level remains consistent.

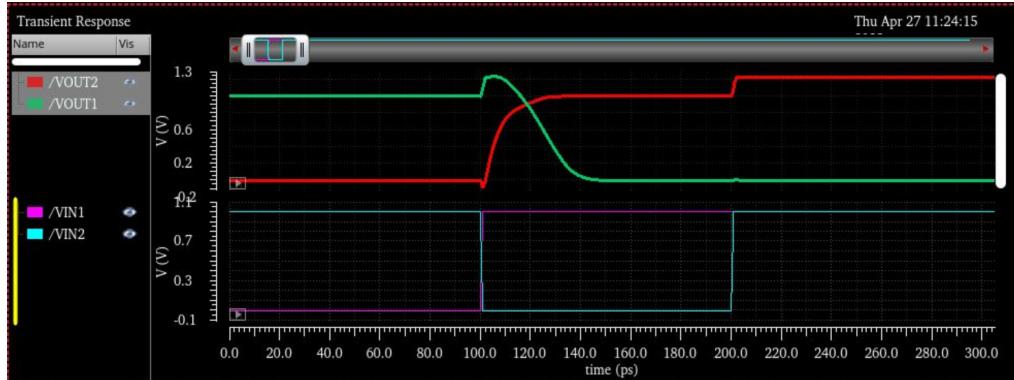


Fig. 38: SR Latch

Figure 39 shows the Strong Arm with T/H switch outputs as well as the SR Latch output. Once again, all 4 distinct phases of the Strong Arm latch are visible, and it is noticed that the SR latch output holds the data during the SAL precharge phase as expected.

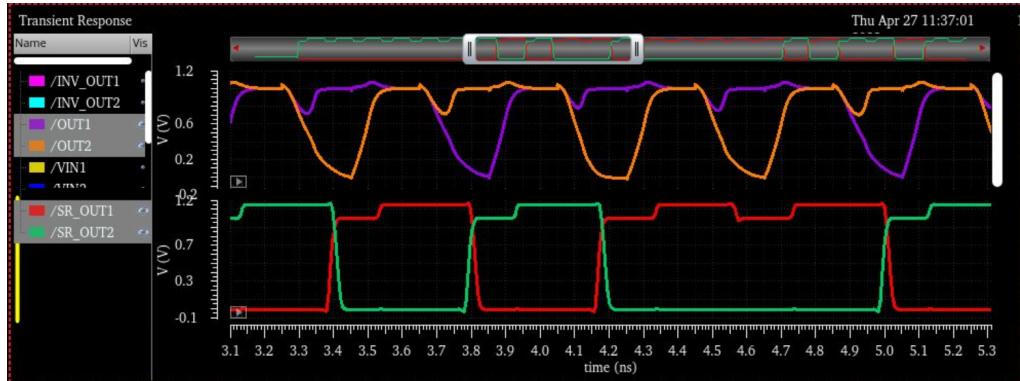


Fig. 39: Strong Arm and SR Latch Waveform

E. Synchronizer

A dual clock synchronizer is designed, clocked at phase shifts of 90 and 270 degrees in order to synchronize the outputs of the SR Latch to the 90 degree phase clock. The schematics for the synchronizer is shown in Figure 40. Once again, a buffer is added to the input of the synchronizer in order to be able to drive a 20fF load. Standard DFF latches are chosen as they are shown to meet timing requirements at quarter bitrate.

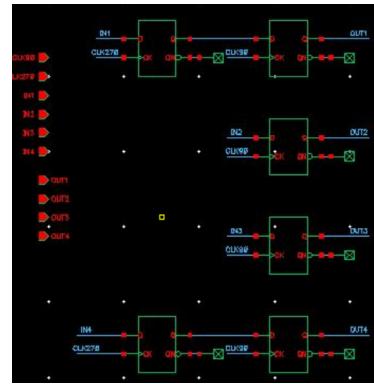


Fig. 40: Synchronizer

A testbench is created for the synchronizer, and a PRBS bit pattern is inputted to determine whether bits are correctly being synchronized to clock 90. The transient is shown in Figure 41. Displaying a single input and output, it is shown that the input signal is successfully synchronized to the 90 degree phase shifted clock.

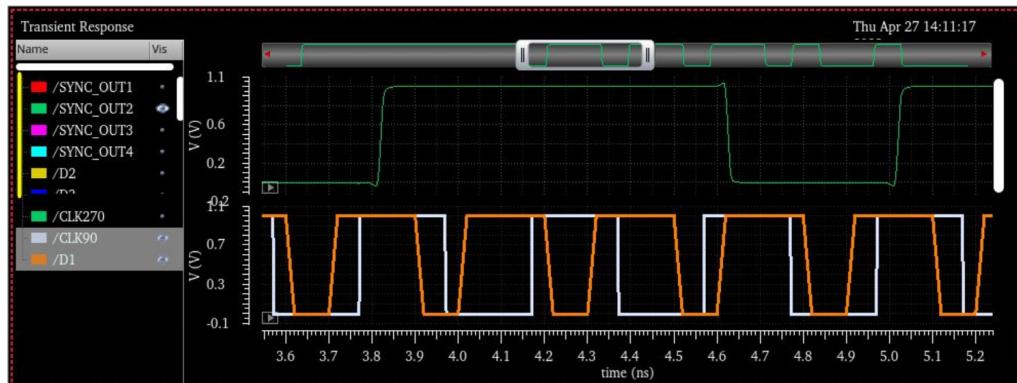


Fig. 41: Synchronizer Waveform

F. Resistor Bank Impedance Matching

To control the output impedance on the RX side, an impedance matching block is designed using resistors in parallel controlled by a transmission gate as shown in Figure 42.

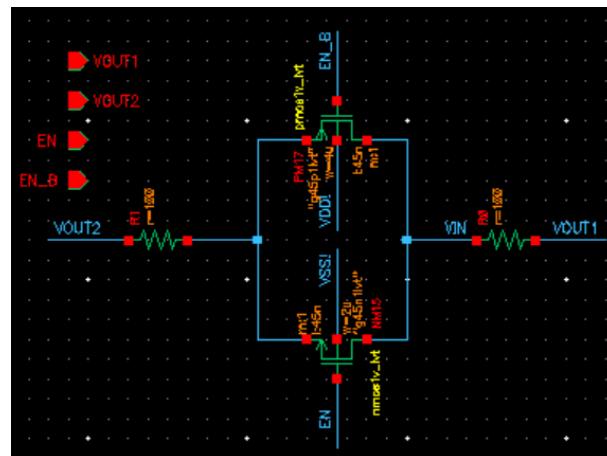


Fig. 42: RX Impedance Slice

For 30% tuning, the input impedance range is between $70\text{--}130 \Omega$. A similar slicing-based design as Section 11.B Figure 19 is used, demonstrating the activation of N Slices to match impedance. For simplicity, 10 slices are once again chosen. For 10

slices, we choose an impedance of 700Ω per slice. The resistor bank is split into 10 slices, where K slices are enabled, such that $RT = 700/K$

The slice enable signals are set by the provided *bussel32* component.

To control the VM driver's output impedance, we split the driver into N slices where K slices are enabled, such that

$$R_{IN} = 700\Omega/K \quad (17)$$

Table 20 shows the expected impedance per slice calculated by measuring the parallel output impedance per slice enabled.

Slices	1	2	3	4	5	6	7	8	9	10
Impedance (Ω)	700	350	233.3	175	140	116.7	100	87.5	77.8	70

TABLE XX: Impedance Per Resistor Bank 10 Slices at 700Ω Each

Finally, we create our full Resistor Bank by shorting all slice outputs together as shown in Figure 43.

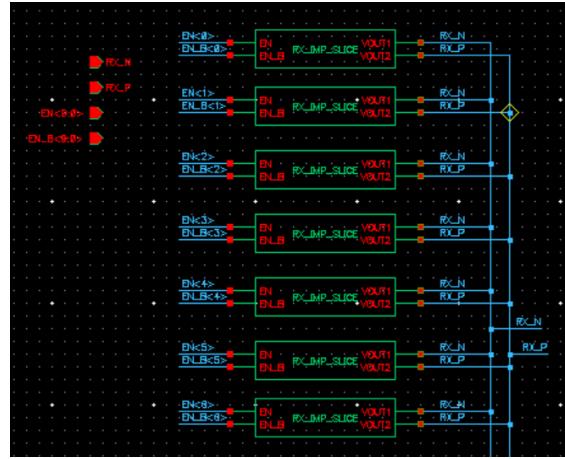


Fig. 43: RX Resistor Bank

Another parameter sweep is run to observe the output impedance of the Resistor Bank. Figure 44 shows the output impedance range as different number of slices are enabled or disabled. Note that the Z-parameters of this parameter sweep are wrong as time ran out, and is only meant as an approximate example of how to find the impedance range. We will assume that 8 slices is the nearest to the termination resistance of 100Ω .

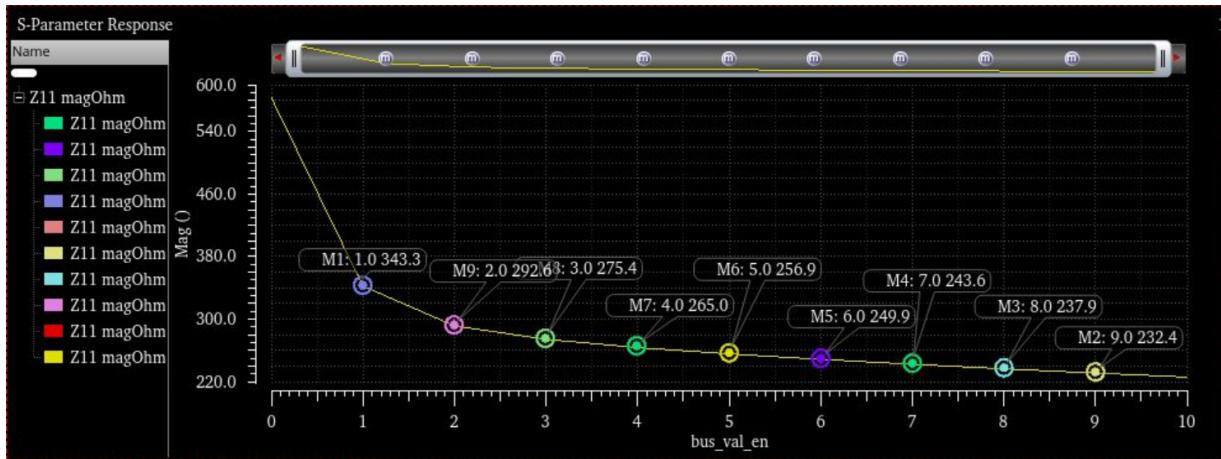


Fig. 44: Z-Parameter Sweep Impedance Matching

XVI. CLOCK BUFFERS

Clock buffers are designed in order to simulate the 90 degree phase shift for each of the 4 clocks. The clock buffer topology is shown in Figure 45. Each clock buffer takes a pulse input and input bar and outputs 180 degree out of phase non-ideal clock. The clock buffers were designed and sized such that project requirements were met and a load of 20 fF could be driven.

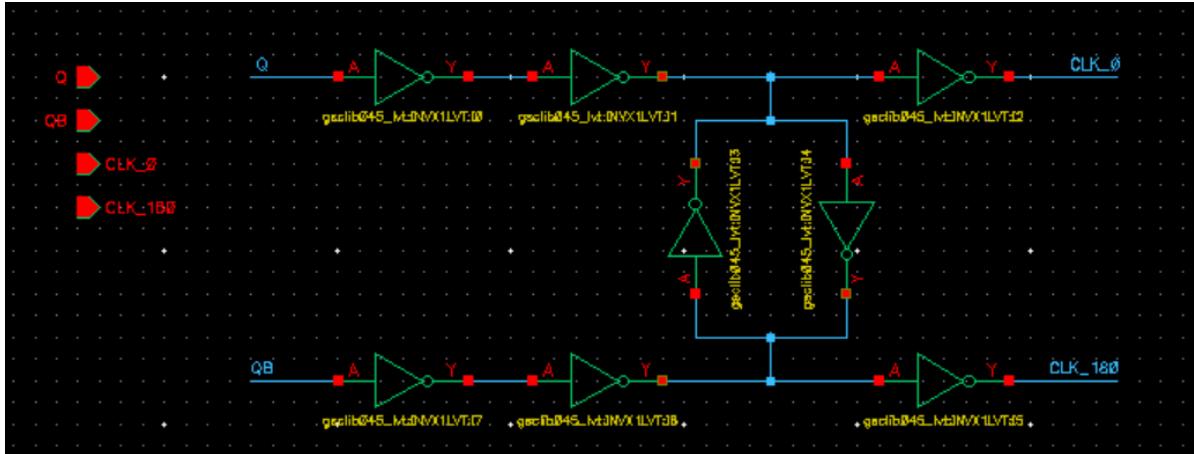


Fig. 45: Clock Buffer

XVII. RX SIMULATION AND PERFORMANCE

Finally, all of the above components are connected together into a single transceiver test bench. The receiver input is matched to the output of the channel, and the output of the synchronizers are connected to PRBS-7 Checkers. First, a transient waveform is run showing the alternating data transition at the RX side for 6 UI. Figure 46 shows the differential signal at the channel's output.

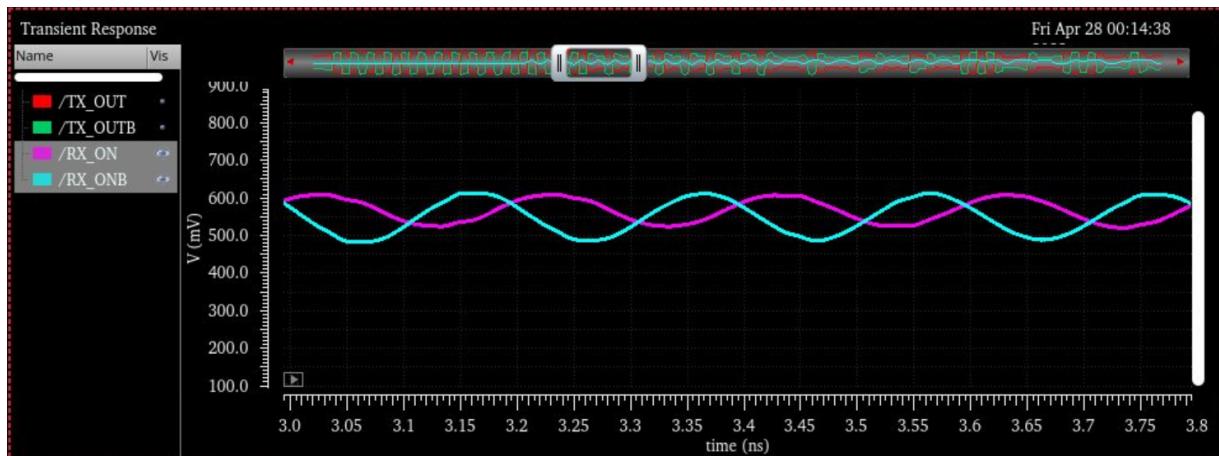


Fig. 46: RX Side Channel Output

A. RX Eye Diagram

Similarly to Section 13.B, an eye diagram is plotted using 10,000 randomly generated 0 and 1 bits inputted into the differential channel as shown in Figure 47.

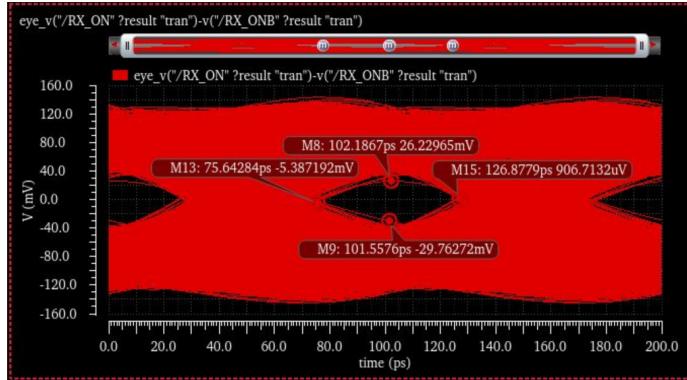


Fig. 47: Random 1000 Bit RX Data Eye Diagram

The eye height and width are as shown in Table 21.

Eye Width (ps)	Eye Height (mV)
51.2351	55.9923

TABLE XXI: Random RX Data Eye Width and Height

The worst case eye diagram can be plotted using the data stream used in Section 12. Figure 48 shows the worst-case eye diagram. Table 22 shows the eye height and width.

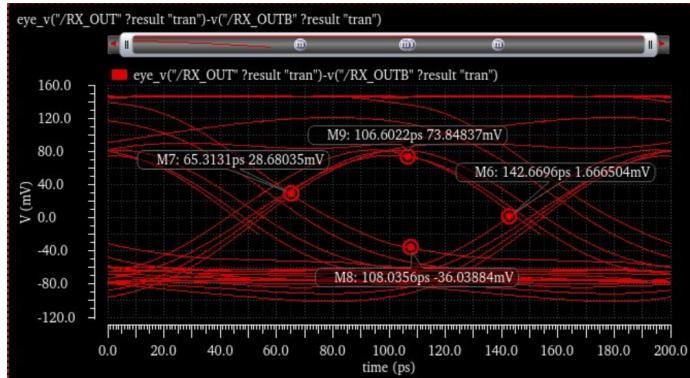


Fig. 48: Worst Case 0 and 1 RX Eye Diagram

Eye Width (ps)	Eye Height (mV)
77.3565	109.8872

TABLE XXII: Worst-case RX Data Eye Width and Height

B. Worst-case Sequence RX Output Recovery

The worst-case sequence as described in Section 12 and Table 13 is used as an input to the transceiver. The bit input to the channel and outputted by the synchronizer are shown to match, although shifted down due to clock synchronizing. The bits retain their original ordering as shown in Figure 49 and 50.

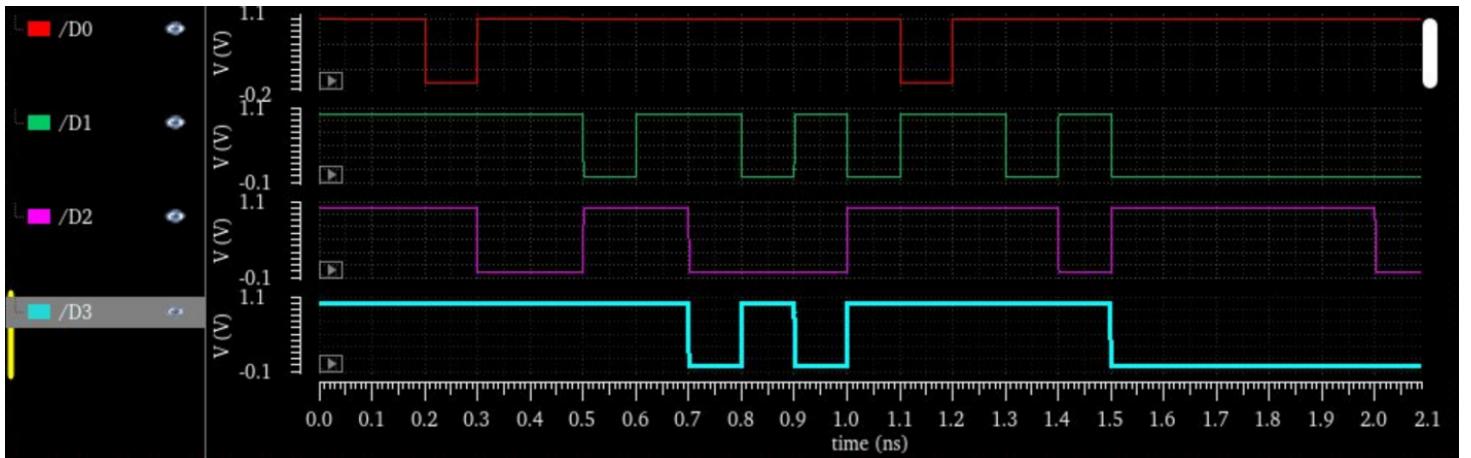


Fig. 49: Worst-case Data Input Stream

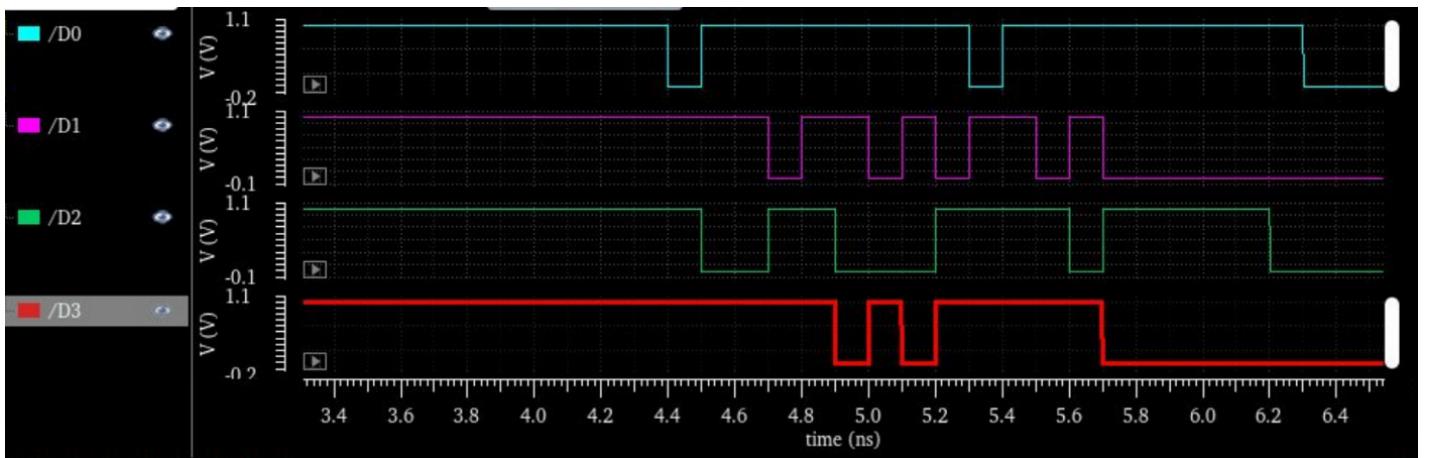


Fig. 50: Worst-case Data Output Stream

C. PRBS Checker

With PRBS-7 bit input, we use 4 PRBS-7 Checkers for each input stream to verify whether any errors occur across the datapath of the tranceiver. If a pulse is seen past 10ns, a bit error is occurring. As shown in Figure 51, the error signal does not go up after the first 10ns.

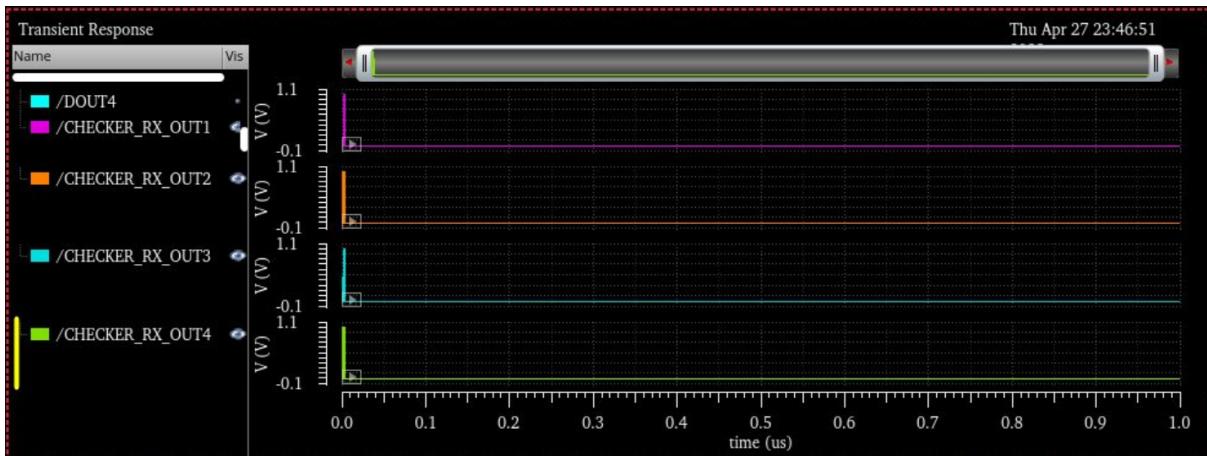


Fig. 51: PRBS-7 Error Checking Output

D. RX Power Consumption Breakdown

Knowing that for the instantaneous power consumption in Watts of a component:

$$P_{VDD} = I_{VDD}(t)V_{VDD}(t) \quad (18)$$

and that the average power is

$$P_{avg} = \frac{E}{T} = \frac{1}{T} \int_0^T P(t)dt \quad (19)$$

where T is time, and E is energy in Joules defined in energy/bit as

$$E = \int_0^T P(t)dt = \frac{1}{10Gb/s} P_{avg} \quad (20)$$

an approximate power and energy breakdown for individual components can be calculated. A 10ns transient simulation is run for each component including the Strong Arm Latch with T/H, SR Latch, Synchronizer, Resistor Bank, Clock Buffers, and overall RX receiver to determine the average current drawn at $V_{DD} = 1V$ and $V_{SS} = 0V$. As $V_{DD} = 1V$, the average power drawn will be equal to the average current drawn. Table 23 shows the power and energy breakdown for each component calculated using the above formulas and the current draw transient simulation.

Component	Average Power (mW)	Energy per Bit (pJ/bit)
SAL and T/H (4 units)	0.125	0.0125
SR Latch	0.401	0.0401
Synchronizer	0.292	0.0292
Resistor Bank	84E-6	8.4E-6
RX Receiver	0.810	0.0810

TABLE XXIII: RX Power and Energy Component Breakdown

E. RX Performance and Specifications Summary

A summary of the full RX receiver specifications met are included in Table 24, and a summary of the transmitter's performance is included in Table 25.

Description	Specification
Data Rate	10Gb/s
DC Sources	Single Current Source, Multiple Voltage Sources
DC Voltage Source(s)	Multiple
VDD	1V
PDK	45nm
Max W or L	700 μm
Synchronizer Ouptut	20 fF Capacitor
NMOS Body	GND
Tuning Range	30% of ideal termination (30-65 Ω)

TABLE XXIV: TX Specifications

Metric	Performance
Worst-case Eye Height	109.8872 mV
Worst-case Eye Width	77.3565 ps
TX Power Consumption	0.810 mW
TX Energy per Bit Consumption	0.0810 pJ/bit

TABLE XXV: TX Performance

XVIII. CONCLUSION

This project consisted in analyzing the pulse response generated from a differential channel's s-parameters and designing the TX and RX side of a 10 Gb/s high-speed link.

First, The worst case data patterns at 10 Gb/s were generated and plotted as well as the eye diagram for a random sequence of bits and worst case bit pattern. The effects of filtering using DFE with and without a TX-FIR filter were observed and compared. For each case, the equalizer coefficients were determined using various methods. The magnitude of the 5 most important cursors for each filtering method were outlined and compared.

Next, a 4:1 serializer, a pre-driver, a voltage-mode driver, and a differential channel using Cadence and GPDK 45nm. Voltage-mode driver slice impedance matching was conducted to match the differential channel's termination resistance. 3 Tap TX-FIR

equalization with 4 bits of resolution is implemented by tuning the voltage-mode driver segments to de-emphasize the pre-cursor, cursor, and post-cursor produced by the serializer. Finally an eye diagram is plotted for PRBS7 data pattern captured over 10,000 UIs at the channel's output.

Finally, a quarter-rate slicer constructed from 4 Strong Arm latches and 4 SR latches was designed with a track-and-hold and tuned for impedance. A synchronizer was designed and finally the output of the channel was compared with its input for the worst case bit sequence.

REFERENCES

- [1] P.Hanumolu, S.Palermo, and S.Shekhar, *ELEC403 Lecure Notes 2023, Slide Set 1-15*, [PowerPoint], 2023.
- [2] S.Shekhar *ELEC403 Matlab Code 2023*, [Online], 2023.
- [3] N.R Whitehead, *Design and Measurement of StrongARM Comparators*. Master of Science Thesis, Brigham Young University Provo, Utah, 2019. [Online]. Available: <https://scholarsarchive.byu.edu/etd/8715/>