



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH



STUDY OF ASIC SERIAL MEMORY INTERFACE DESIGN

A Degree Thesis

**Submitted to the Faculty of the
Escola Tècnica d'Enginyeria de Telecomunicació de
Barcelona**

Universitat Politècnica de Catalunya

by

Enric Erra Bes

**In partial fulfilment
of the requirements for the degree in
Telecommunications Technologies and Services
Engineering**

Advisors: Francesc Moll and Diego Mateo

Barcelona, August 2020

Abstract

This degree thesis is about the design of the transmitter of a high-speed serial link interface to communicate an ASIC with a FPGA board. Using this serial interface, the ASIC will be able to access via FPGA the DDR3 RAM on the board. The target data rate is 8 Gbps, which has been achieved.

The system has 8 differential data inputs with a 1 Gbps data rate. This parallel data is converted to an 8 Gbps serial differential signal, and is amplified and filtered with a 3 tap FIR to equalize the channel. The output differential impedance is adapted to prevent reflections.

In future designs the signal amplitude and rise/fall time could be improved in order to enhance this transmitter.

Resum

Aquest treball de fi de grau tracta sobre el disseny del transmissor d'una interfície sèrie d'alta velocitat per comunicar un ASIC amb una placa FPGA. Utilitzant aquesta interfície, l'ASIC podrà accedir mitjançant FPGA a la RAM DDR3 que hi ha a la placa. L'objectiu de velocitat de transmissió de dades és de 8 Gbps, cosa que s'ha aconseguit.

El sistema compta amb 8 entrades de dades diferencials amb una velocitat de transmissió de dades d'1 Gbps. Aquestes dades paral·leles es converteixen en un senyal sèrie diferencial de 8 Gbps, i s'amplifica i es filtra amb un filtre FIR de 3 coeficients per equalitzar el canal. La impedància diferencial de sortida és adaptada per evitar reflexions.

En futurs dissenys es pot millorar l'amplitud del senyal i el temps de pujada/baixada per millorar aquest transmissor.

Resumen

Este trabajo de fin de grado trata sobre el diseño del transmisor de una interfaz serie de alta velocidad para comunicar un ASIC con una placa FPGA. Utilizando esta interfaz, el ASIC podrá acceder mediante FPGA en la RAM DDR3 que hay en la placa. El objetivo de velocidad de transmisión de datos es de 8 Gbps, que se ha conseguido.

El sistema cuenta con 8 entradas de datos diferenciales con una velocidad de transmisión de datos de 1 Gbps. Estos datos en paralelo se convierten en una señal serie diferencial de 8 Gbps, y se amplifica y se filtra con un filtro FIR de 3 coeficientes para ecualizar el canal. La impedancia diferencial de salida es adaptada para evitar reflexiones.

En futuros diseños se puede mejorar la amplitud de la señal y el tiempo de subida/bajada para mejorar este transmisor.



For my family and Ari.

Acknowledgements

I would like to express my deepest appreciation to my advisors, Francesc Moll and Diego Mateo, they have been a great support for this project and their constant guidance and advices were key to this thesis.

I also wish to acknowledge the very valuable help provided by Enrique Barajas in some critical moments of this thesis.

Revision history and approval record

Revision	Date	Purpose
0	01/06/2020	Document creation
1	05/08/2020	Document revision
2	15/08/2020	Document revision

DOCUMENT DISTRIBUTION LIST

Name	e-mail
Enric Erra Bes	enric.erra@estudiantat.upc.edu
Francesc Moll Echeto	francesc.moll@upc.edu
Diego Mateo Peña	diego.mateo@upc.edu

Written by:		Reviewed and approved by:	
Date	14/08/2020	Date	15/08/2020
Name	Enric Erra	Name	Francesc Moll
Position	Project Author	Position	Project Supervisor

Table of contents

Abstract	1
Resum	2
Resumen	3
Acknowledgements	5
Revision history and approval record	6
Table of contents	7
List of Figures	8
List of Tables	9
1. Introduction	10
1.1. Project background	10
1.2. Project overview	10
1.2.1. Eye diagram	10
1.2.2. Equalization	12
1.2.3. System description	15
1.3. State of the art of the technology used or applied in this thesis:	16
1.4. Milestones and Gantt diagram	18
1.4.1. Milestones	18
1.4.2. Gantt diagram	18
1.4.3. Deviations from the initial plan and incidences	18
2. Parts of the transmitter circuit	20
2.1. Serializer	20
2.1.1. Multiplexer	20
2.2. Pre-Amplifier	23
2.3. FIR filter	24
2.3.1. Differential inverter	24
2.3.2. Delay (Z^{-1})	26
2.3.3. XOR	28
2.3.4. Driver	30
3. Methodology / project development:	32
4. Results	33
5. Budget	36
6. Conclusions and future development:	37
Bibliography:	38
Glossary	39

List of Figures

Figure 1. Time-domain signal to be analysed using eye diagram.....	11
Figure 2. Eye diagram of the previous waveform	11
Figure 3. Example of TX FIR effects on transmitted pulse	12
Figure 4. RX FIR block diagram.....	13
Figure 5. RX CTLE schematic	13
Figure 6. RX DFE block diagram	14
Figure 7. Eye diagram without equalization Figure 8. Eye diagram with equalization	14
Figure 9. Complete serial link block diagram	15
Figure 10. Simple block diagram of the transmitter	15
Figure 11. Serializer chronogram.....	15
Figure 12. Detailed block diagram of the transmitter	16
Figure 13. Gantt diagram.....	18
Figure 14. Multiplexer schematic	20
Figure 15. Multiplexer wave diagram	20
Figure 16. Multiplexer Cadence implementation	21
Figure 17. Multiplexer simulation	21
Figure 18. Serializer Cadence implementation.....	22
Figure 19. Pre-Amplifier schematic.....	23
Figure 20. Pre-amplifier Cadence implementation	23
Figure 21. Pre-amplifier simulation	24
Figure 22. Differential inverter schematic.....	24
Figure 23. Differential inverter Cadence implementation.....	25
Figure 24. Differential inverter simulation.....	25
Figure 25. Delay schematic	26
Figure 26. Delay Cadence implementation	27
Figure 27. Delay simulation	27
Figure 28. XOR schematic.....	28
Figure 29. XOR Cadence implementation.....	28
Figure 30. XOR simulation: positive coefficient.....	29
Figure 31. XOR simulation: negative coefficient.....	29
Figure 32. Driver schematic.....	30

Figure 33. Driver Cadence implementation.....	30
Figure 34. Driver simulation	31
Figure 35. Full system schematic.....	33
Figure 36. Nominal eye diagram at the output of the transmitter	34
Figure 37. Worst case eye diagram at the output of the transmitter	34
Figure 38. Best case eye diagram at the output of the transmitter	35

List of Tables

Table 1. Aurora Transmitter Specifications ^[3]	17
Table 2. CEI-6G Transmitter Specifications ^[5]	17
Table 3. Power consumption	36
Table 4. Budget	36

1. Introduction

1.1. Project background

This project is inserted in a long-term activity to design advanced processor chips at the Barcelona Supercomputing Center. In particular, this thesis aims to design the transmitter of a high-speed serial interface to communicate an ASIC with a DDR3 RAM via a Xilinx FPGA KC705 board. The design is based on reference^[10].

The maximum bitrate of a DDR3 RAM on a KC705 board is around 100 Gbps. The current solution does not take advantage of that speed, using only 1.6 Gbps. The desired bitrate in this project is 8 Gbps in a serial link, and 4 serial links can be implemented in this board, leaving an overall bitrate target of 32 Gbps, increasing an order of magnitude from the current implementation.

In order to achieve this high speed, the design must use Current Mode Logic (CML) to achieve fast transition (rise/fall) times and the data must be differential to be more robust to interferences.

The technology used is 22FDX from Global Foundries. It employs 22 nm Fully-Depleted Silicon-On-Insulator (FD-SOI), which excels at delivering outstanding performance at ultra-low power, operating at 0.4 Volt and being power efficient. It also offers the possibility of body-biasing, which allows a trade-off between performance and power.

1.2. Project overview

1.2.1. Eye diagram

Due to distortions and ISI (Intersymbol Interference), not every sequential combination of bits has perfect transitions. The eye diagram is a superposition of successive waveforms of the same period in order to examine the worst case. A large signal is split in equal periods of 1 UI (Unit Interval), and plotted all together. The UI is the time between possible transitions in the signal. For an 8 Gbps signal, the UI is 125 ps. In the eye diagram we have all the possible binary logic-level combinations: 000, 001, 010, 011, 100, 101, 110, 111. With this method we can measure the vertical eye opening, the deterministic jitter or horizontal eye opening among other measurements. This method will be used to determine the quality of the resulting signal.

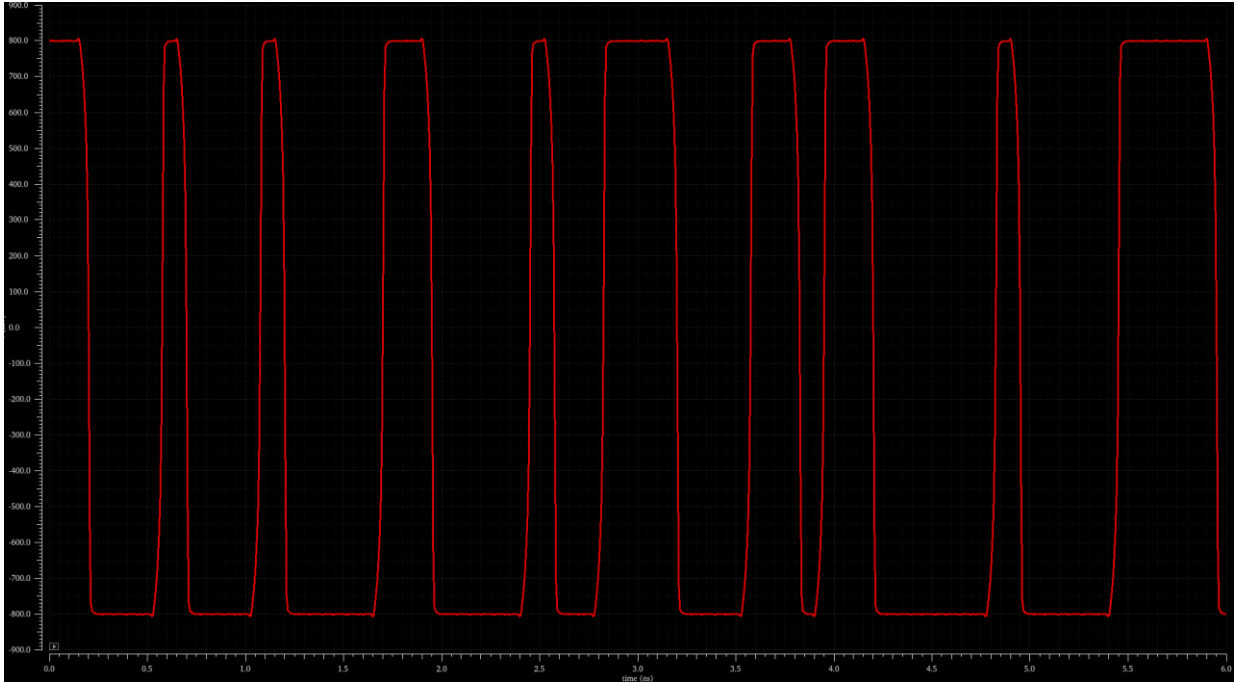


Figure 1. Time-domain signal to be analysed using eye diagram

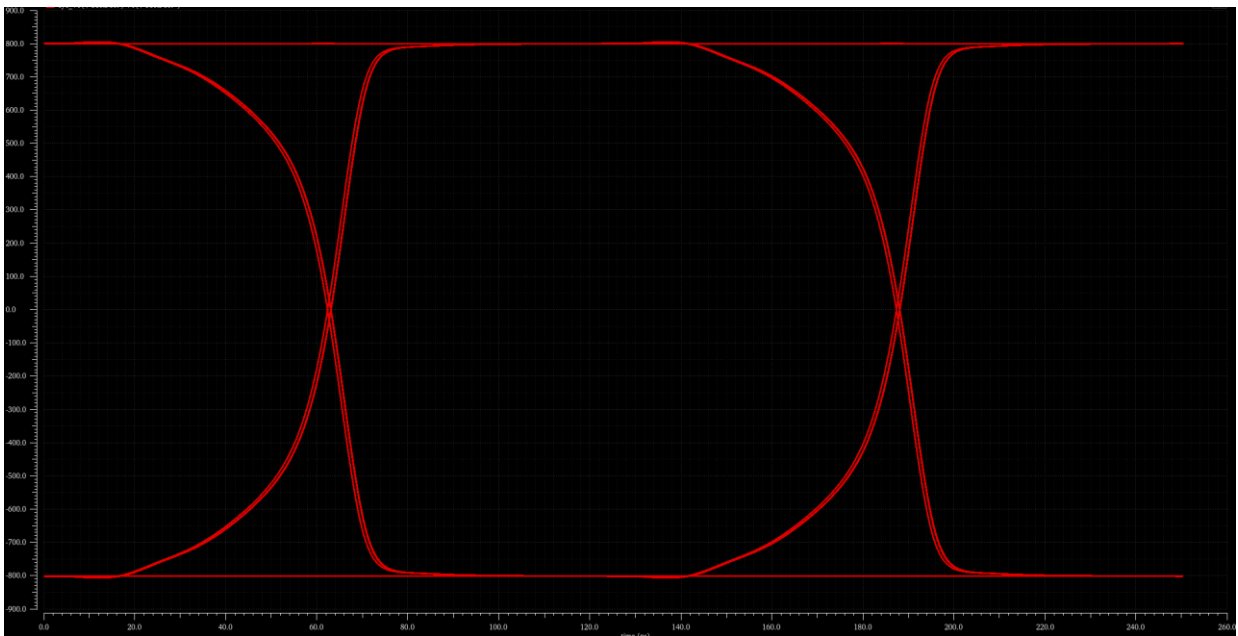


Figure 2. Eye diagram of the previous waveform

1.2.2. Equalization

The channels in high-speed serial links are not ideal. They produce ISI, attenuation, and other effects that worsen the signal quality. In order to minimize these effects, equalization techniques are applied. Using equalization properly can greatly improve the signal quality.

There are 4 main equalization techniques in high-speed serial links:

1.2.2.1. TX FIR

A transmitter finite impulse response filter pre-distorts the transmitted pulse in order to invert the channel distortion.

Advantages:

- Easy to implement
- Can cancel ISI in pre-cursors and post-cursors
- Does not amplify noise

Disadvantages:

- Attenuates low frequency content

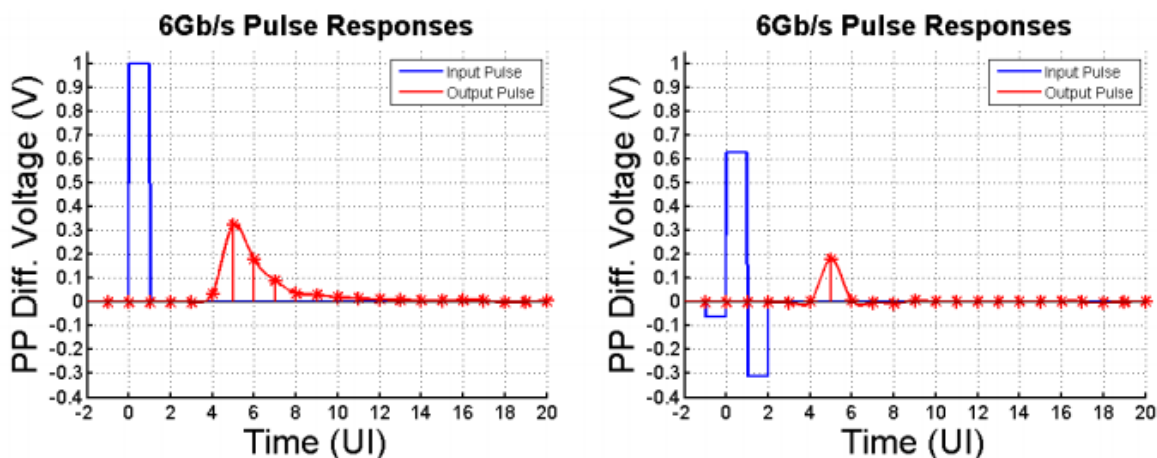


Figure 3. Example of TX FIR effects on transmitted pulse

Source: S.Palermo^[2]

1.2.2.2. RX FIR

A receiver finite impulse response filter distorts the signal received in order to invert the channel distortion.

Advantages:

- Can amplify high frequencies instead of attenuating low frequency content
- Can cancel ISI in pre-cursors and post-cursors

Disadvantages:

- Amplifies noise and crosstalk
- Needs analog delays

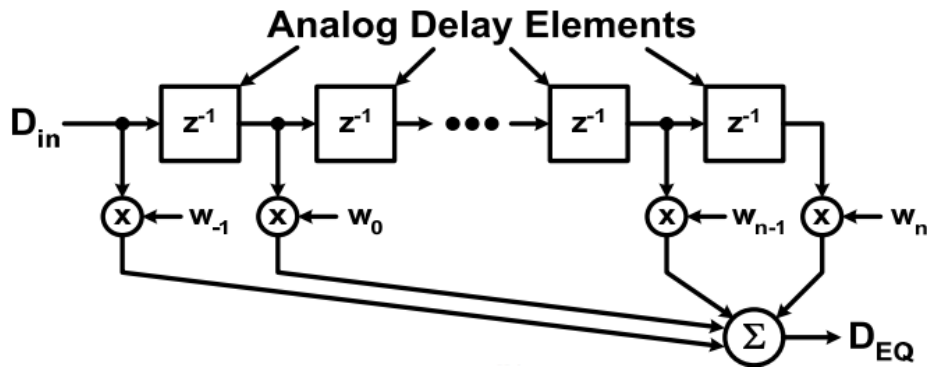


Figure 4. RX FIR block diagram

Source: S.Palermo^[2]

1.2.2.3. RX CTLE

The receiver continuous time linear equalizer aims to counteract the effects of the channel's transfer function.

Advantages:

- Provides gain and equalization with low power and area overhead
- Can cancel pre-cursor and long-tail ISI

Disadvantages:

- Generally limited to first order compensation
- Amplifies noise and crosstalk
- PVT sensitivity
- Hard to tune

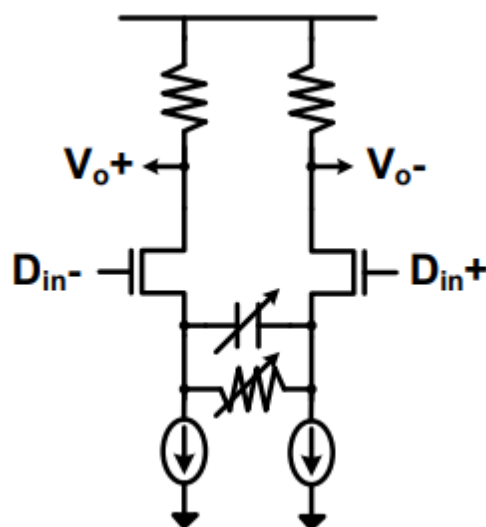


Figure 5. RX CTLE schematic

Source: S.Palermo^[2]

1.2.2.4. RX DFE

The receiver decision feedback equalizer aims to negate the effects of the post-cursors through feedback FIR filter and accurate sampling.

Advantages:

- Does not amplify noise and crosstalk

Disadvantages:

- Cannot cancel pre-cursor ISI
- Critical feedback timing path
- Timing of ISI subtraction complicates the CDR (clock and data recovery) phase detection

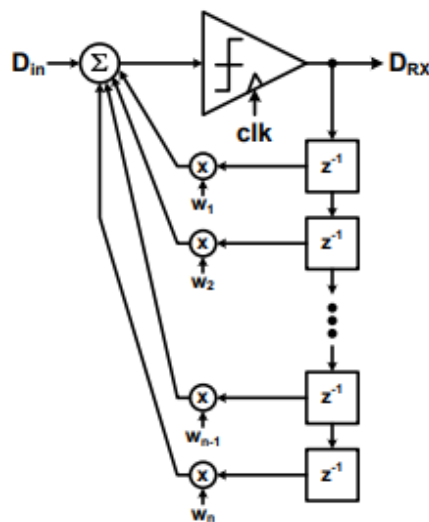


Figure 6. RX DFE block diagram

Source: S.Palermo^[2]

One of these techniques will be implemented in this project: the FIR filter at the transmitter^{[1][2]}. This is the only technique in this list available for the transmitter part of the serial link. We used 3 taps to be able to handle one pre-cursor and one post-cursor, while trying not to raise too much the parasitic capacitance at the output (each tap adds up capacitance, and more capacitance means a lower maximum frequency).

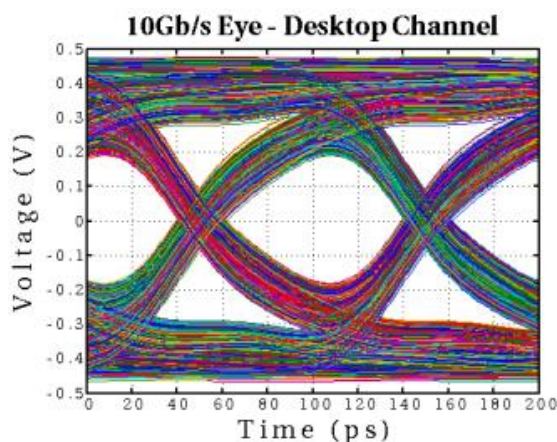


Figure 7. Eye diagram without equalization

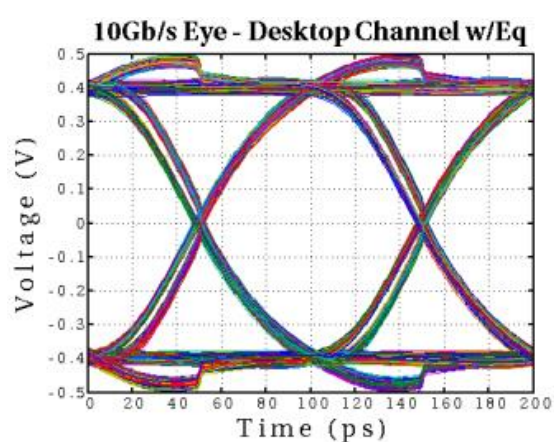


Figure 8. Eye diagram with equalization

Sources: S.Palermo^[2]

1.2.3. System description

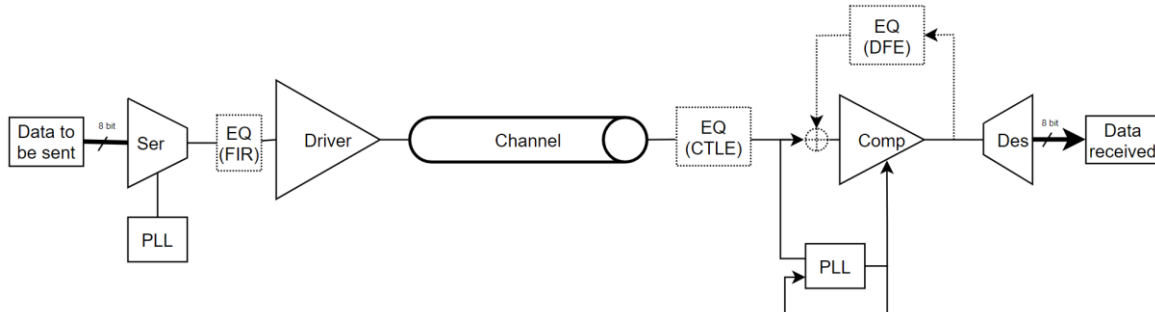


Figure 9. Complete serial link block diagram

A serial link includes two different stages: the transmitter and the receiver. They are separated by the channel, the physical medium where the signal propagates. This thesis is focused on the transmitter part, which includes the Serializer, FIR filter equalization, and a driver. The other blocks (the CTLE, DFE, Comparator, De-serializer and the PLL for the clock signal generation) are out of scope for this project.

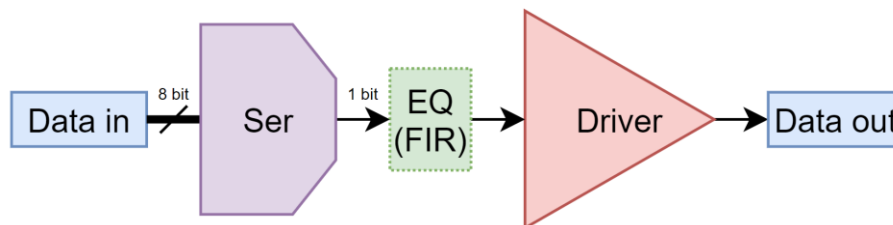


Figure 10. Simple block diagram of the transmitter

The data to be sent is stored in an 8-bit register, and is refreshed every 1 ns. The system starts serializing (purple blocks in figure 12) the eight signals to a single signal with a bit-rate of 8 Gbps, but the serializer does not have a wide signal swing at the output and has glitches caused by the clock transistor's commutation.

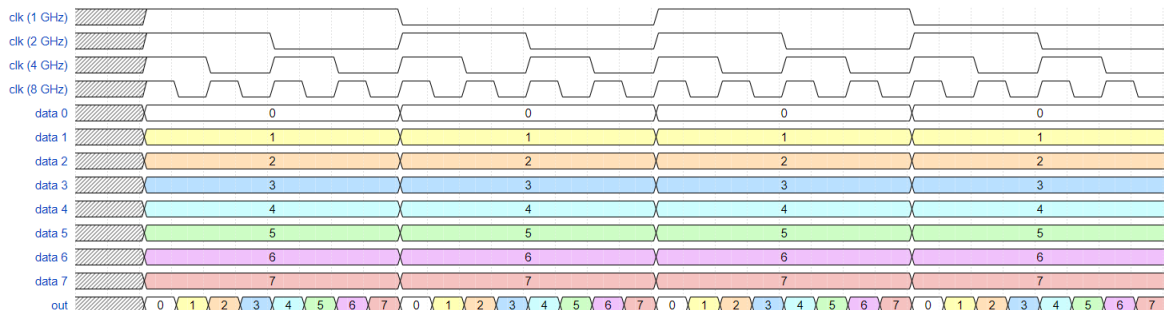


Figure 11. Serializer chronogram

To reset the high (V_{dd}, 0.8 V) and low voltage (GND) levels a differential inverter is ideal, but the differential inverter needs a pre-amplifier in order to have enough input signal swing for the inverter to correctly distinguish each signal level.

After the pre-amplifier (yellow block in figure 12) we can use the differential inverter and the signal is sent to a XOR and a delay.

Here begins the implementation of a 3-tap FIR filter (green blocks in figure 12) designed to equalize the channel.

The other input of the XOR is the sign of the first coefficient of the FIR filter. After the XOR, another inverter is connected to reset the signal and it is sent to the driver (red blocks in figure 12).

Following the delay, we will have the same XOR-driver structure repeated 3 times, but with a different delay each time.

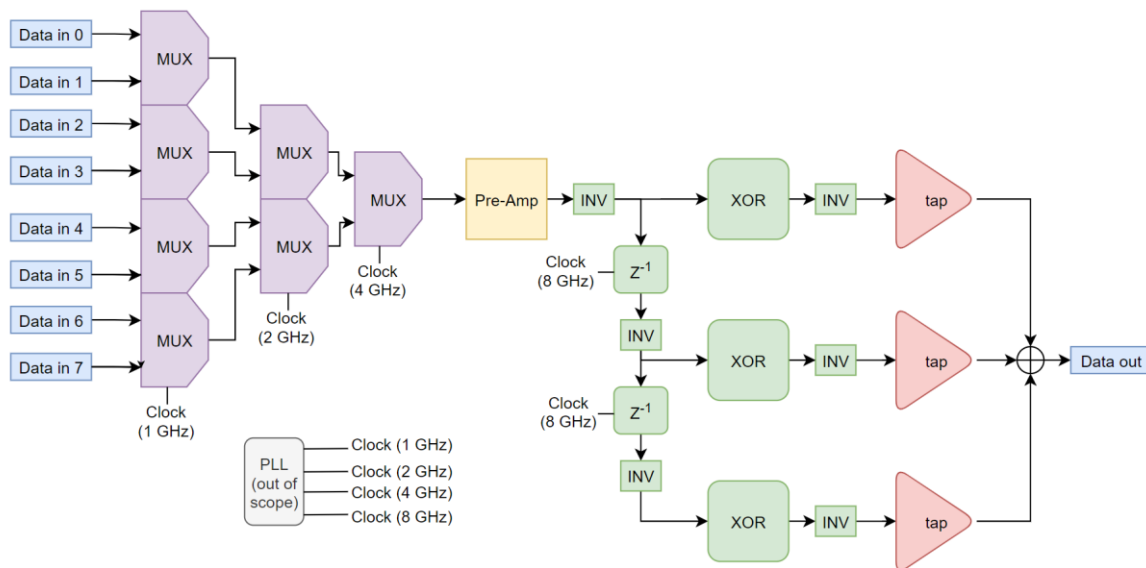


Figure 12. Detailed block diagram of the transmitter

1.3. State of the art of the technology used or applied in this thesis:

Every year bigger data transfer rates are needed by the industry. The trend is to go from parallel to serial (examples: “Parallel IDE” was replaced by “SATA”, and “PCI (parallel)” was replaced by “PCI Express (serial)”^[2]).

In this thesis we focused on a processor-to-memory interface, but high-speed serial links can be applied to a lot of systems, including processor-to-peripheral, processor-to-processor, storage, networks, etc.

For chip-to-chip serial links communication exist some standards, including Aurora^[3] or Interlaken^[4], which include electrical specifications of transmitter and receiver circuits. For example, Interlaken includes CEI-6G or CEI-11G^[5], and future work improving this project could satisfy one of them.

The most used technology to achieve fast speed is Current Mode Logic, which focuses on current steering through resistors instead of short-circuiting the output to ground or power supply. We used differential signalling because it is easier to implement and is more robust

to interferences than single ended signalling, at the cost of more transmission lines and input/output pins^[1].

Nowadays high-speed serial links are in the 10+ Gbps range, using modern equalization and clock and data recovery techniques. These equalization techniques include a FIR filter at the transmitter, and we implemented it. The rest of techniques are beyond the scope of this thesis.

Table 6-3: Transmitter AC Timing Specifications - 3.125 Gbps Baud Rate

Characteristic	Symbol	Range		Unit	Notes
		Min	Max		
Differential output voltage	V _{DIFF}	800	1600	mV	Peak-to-peak differential
Rise/fall time	T _{RF}	30		ps	At driver output
Deterministic jitter	J _D		0.17	UI	
Total jitter	J _T		0.35	UI	
Output skew	S _O		15	ps	Skew at a transmitter output between the two signals comprising a differential pair
Multiple output skew	S _{MO}		1000	ps	Skew at the transmitter output between lanes of a multi-lane channel
Unit interval	UI	320	320	ps	+/- 100 ppm

Note: AC coupling is required to guarantee interoperability between vendors.

Table 1. Aurora Transmitter Specifications^[3]

Table 6-1. CEI-6G-SR Transmitter Output Electrical Specifications

Characteristic	Symbol	Condition	MIN.	TYP.	MAX.	UNIT
Baud Rate	T_Baud	See 6.4.1.2	4.976		6.375	Gsym/s
Output Differential voltage (into floating load Rload=100Ω)	T_Vdiff	See 6.4.1.3	400		750	mVppd
Differential Resistance	T_Rd	See 6.4.1.5	80	100	120	Ω
Recommended output rise and fall times (20% to 80%)	T_tr, T_tf	See 6.4.1.4	30			ps
Differential Output Return Loss (100MHz to 0.75*T_Baud)	T_SDD22	See 6.4.1.5			-8	dB
Differential Output Return Loss (0.75*T_Baud to T_Baud)						
Common Mode Return Loss (100MHz to 0.75 *T_Baud)	T_SCC22	See 6.4.1.5			-6	dB
Transmitter Common Mode Noise	T_Ncm				5% of T_Vdiff	mVppd

Table 2. CEI-6G Transmitter Specifications^[5]

1.4. Milestones and Gantt diagram

1.4.1. Milestones

1. Research about GTX interface and channel model
2. Serializer design
3. Review of possible FIR implementations
4. FIR design
5. Driver design
6. Integration
7. Testing

1.4.2. Gantt diagram

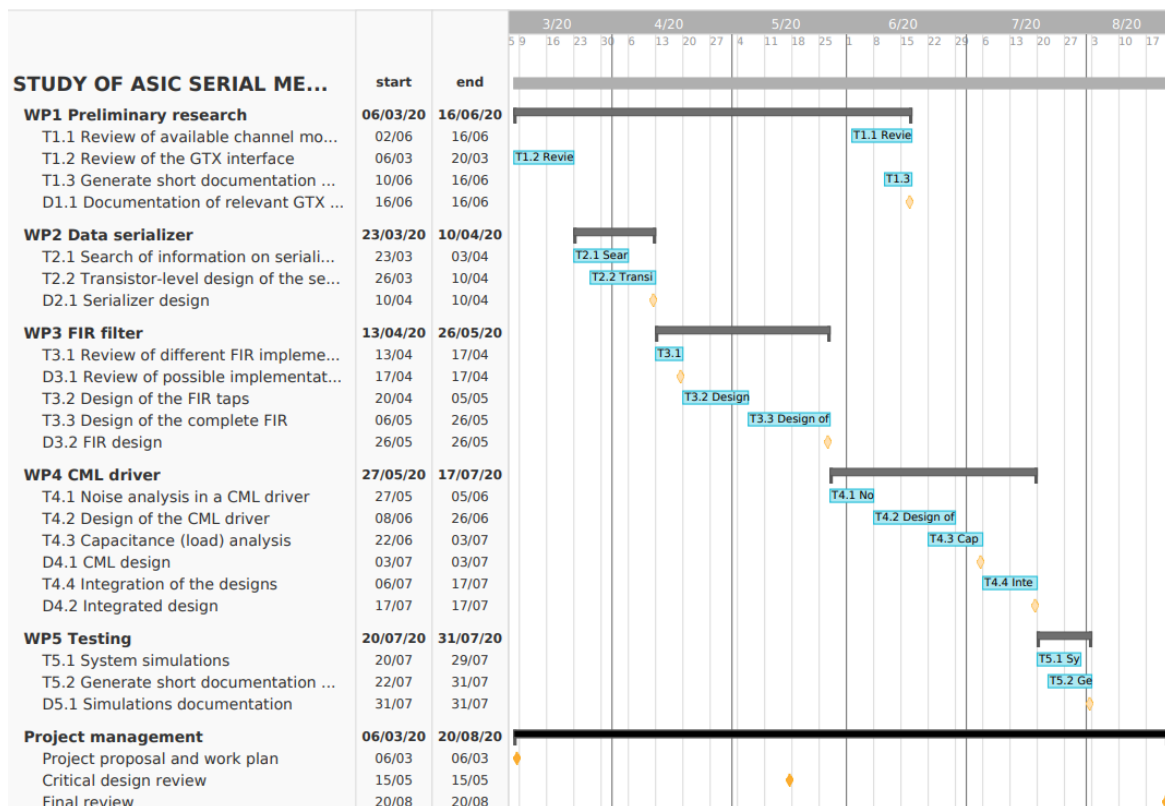


Figure 13. Gantt diagram

1.4.3. Deviations from the initial plan and incidences

Due to the COVID-19 global pandemic, in March 2020 the emergency state was declared in Spain. The university had to close, forbidding the presential work, effectively blocking the access to the computers used to develop this project. Since the main CAD software used was Cadence, and it cannot be installed without an expensive license, remote

connection had to be established to continue with this thesis, and several days were lost at the beginning of the schedule for this reason.

Technical difficulties have been found when trying to install and use software for the channel model acquisition, heavily affecting the first work package.

The main modifications are delays in the early tasks and milestones, but the thesis has been successfully finished in time.

2. Parts of the transmitter circuit

All simulations shown in this document will be single ended for simplicity

2.1. Serializer

As stated before, this block converts 8 signals at 1 Gbps to 1 signal at 8 Gbps. It consists of multiplexers connected forming 3 layers of 4, 2, and 1 multiplexer. Keep in mind that each signal is differential.

2.1.1. Multiplexer

As the serializer is made by multiplexers, we need to understand how it works.

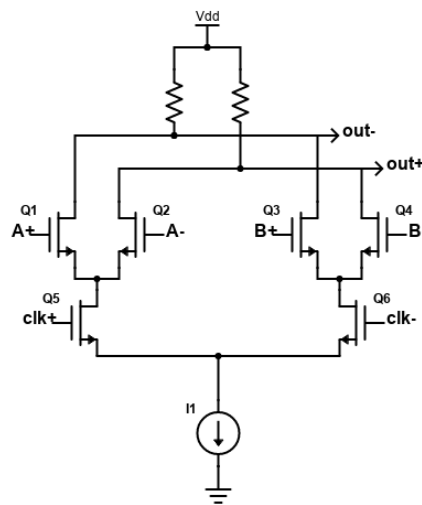


Figure 14. Multiplexer schematic

When the clock signal is high (clk+ high, Q5 ON, and clk- low, Q6 OFF) the A data will be copied to the output using the resistors. When the clock switches (clk+ low, Q5 OFF, and clk- high, Q6 ON), the B data will be copied to the output.

Note that figure 15 is simplified, considering single-ended signals. In the real schematic, each signal will have its complementary signal.

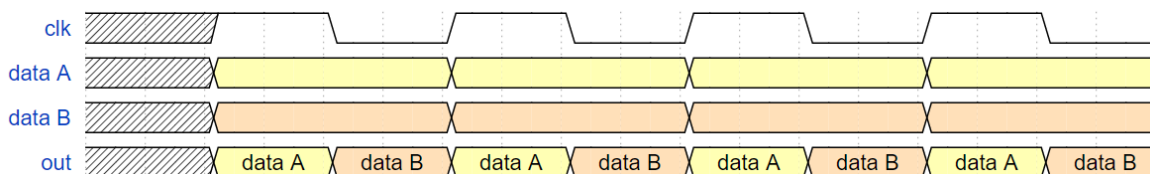


Figure 15. Multiplexer wave diagram

Using the block diagram from figure 12, with a different clock for each multiplexer stage, the signal is correctly serialized. The clock's frequency must be the signal data rate at the

input of the multiplexer, so for the first stage (of 4 multiplexers) the clock signal is 1 GHz, for the second stage (2 multiplexers) is 2 GHz and the third stage (1 multiplexer) is 4 GHz.

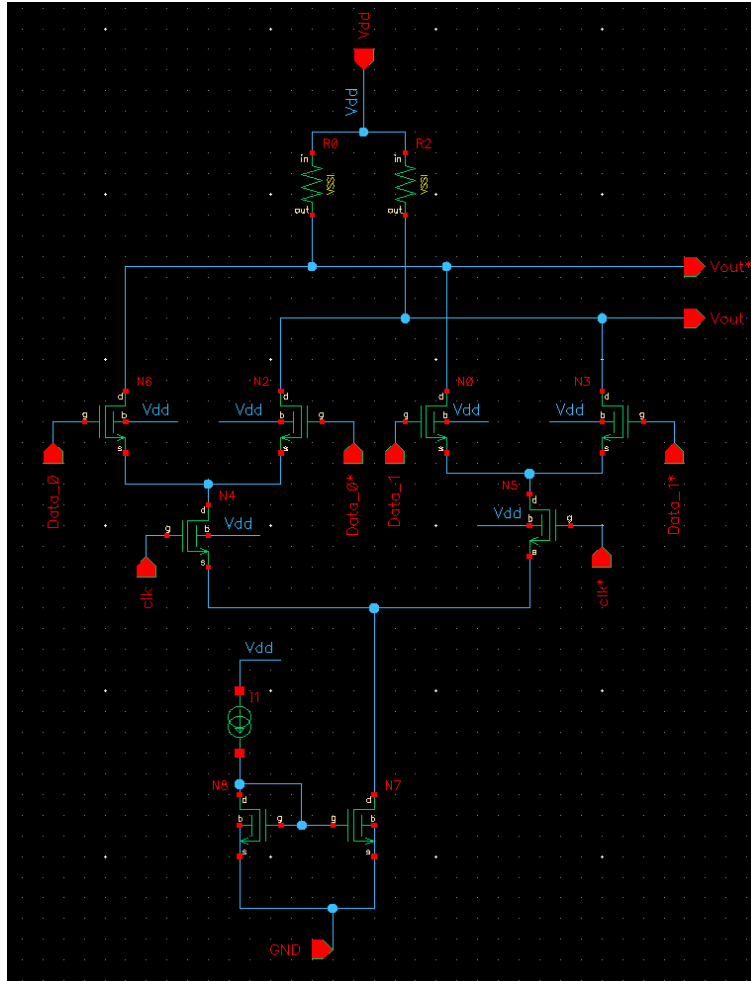


Figure 16. Multiplexer Cadence implementation

Implementing this design in cadence as in figure 17, we can show real simulations:

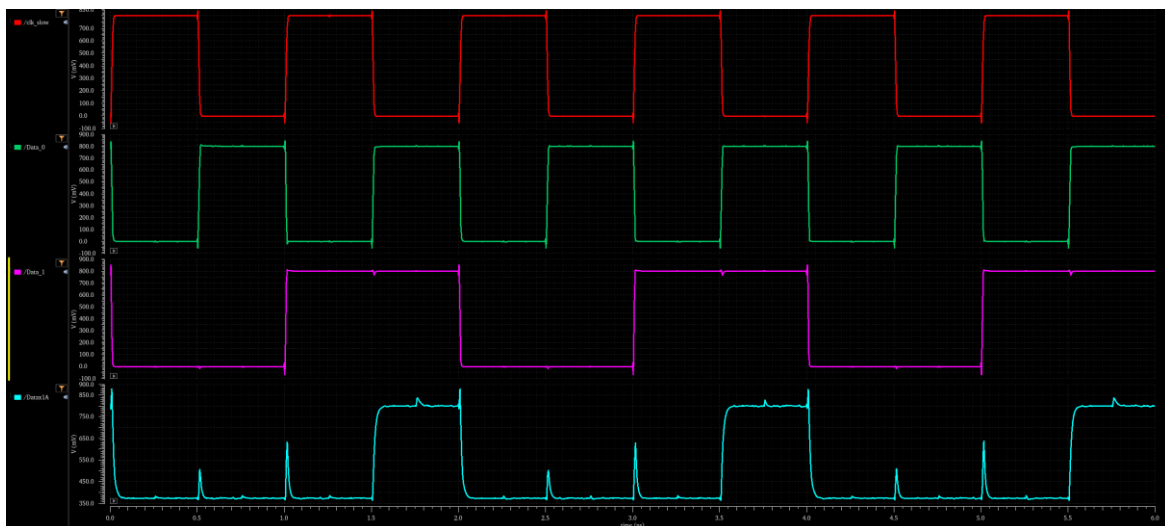


Figure 17. Multiplexer simulation

In figure 17:

- Red: clock
- Green: A signal
- Pink: B signal
- Blue: output signal

When the clock signal is high, the signal A is copied to the output, and when the clock signal is low, the signal B is copied to the output. Some glitches appear in the output due to the clock transistor's transitions.

The serializer is completed (figure 18) by using 7 multiplexers as shown in the block diagram in figure 12.

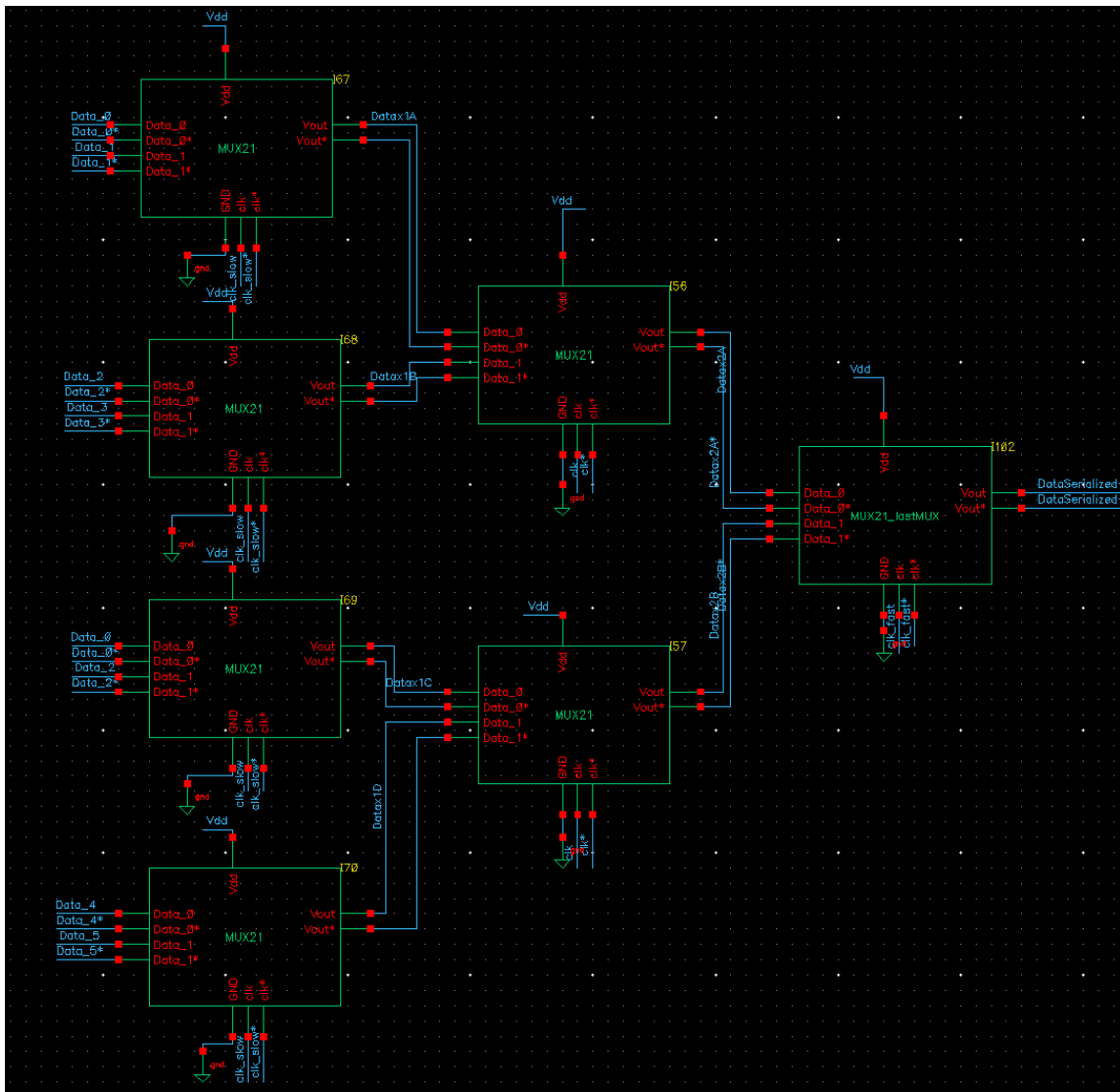


Figure 18. Serializer Cadence implementation

2.2. Pre-Amplifier

In order to have enough voltage difference between the differential signals we have to use an amplifier after the serializer, before the filter. Also, this block reduces the glitches appeared in the input signal due to multiplexer's clock transitions, because this block is asynchronous. The schematic is shown on figure 19.

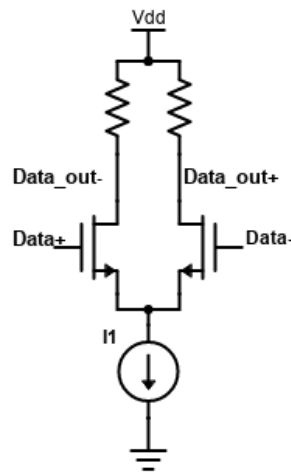


Figure 19. Pre-Amplifier schematic

The circuit is a simple differential CML inverter. The amplification is achieved using a bigger current than in the multiplexers (assuming the same resistors). Cadence implementation in Figure 20.

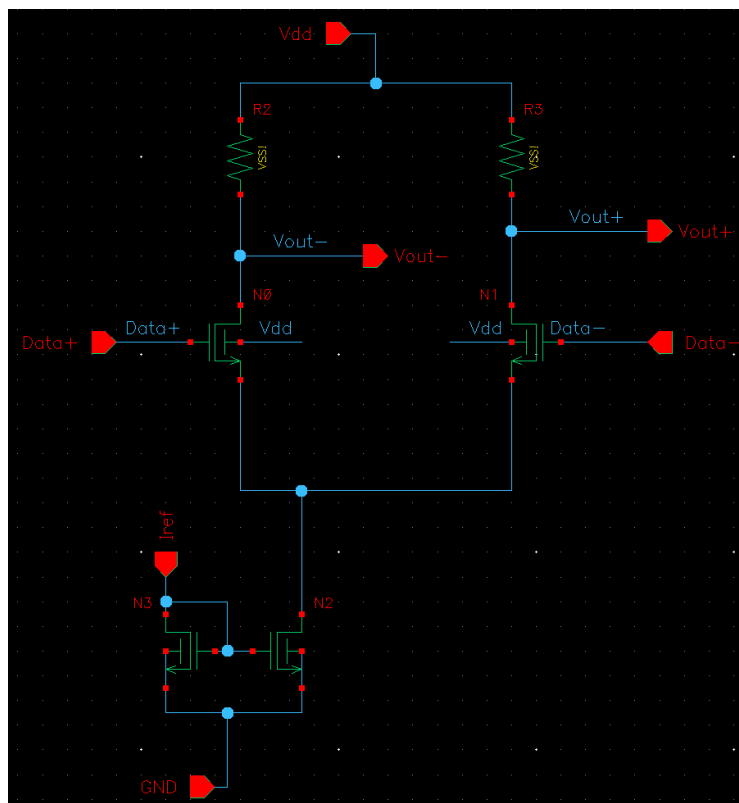


Figure 20. Pre-amplifier Cadence implementation

In figure 21 we can see the effects of this block: amplification, but also glitches attenuation. The high logic level is the same in both cases, 800 mV (Vdd), but the low logic level is different. In the first signal it is 230 mV, in the second is 170 mV. It may not seem much amplification, but when process variations were applied, this extra voltage margin was needed.

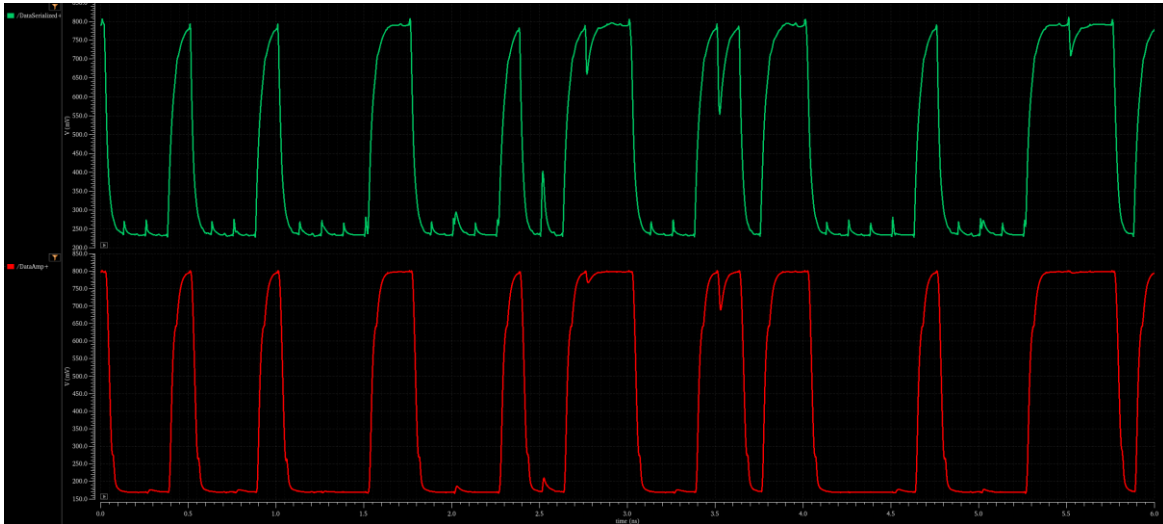


Figure 21. Pre-amplifier simulation

2.3. FIR filter

These next components are needed in order to have a FIR (Finite Impulse Response) filter: differential inverter for signal quality, a delay block to separate the coefficients and a XOR block to control the coefficient's sign.

2.3.1. Differential inverter

In order to reset the high logic value to Vdd and the low logic value to GND, we needed a circuit that short-circuits the output to either Vdd or GND. In addition, we needed a differential circuit to avoid mismatches in the output signals. We used the next circuit because it fitted the requirements:

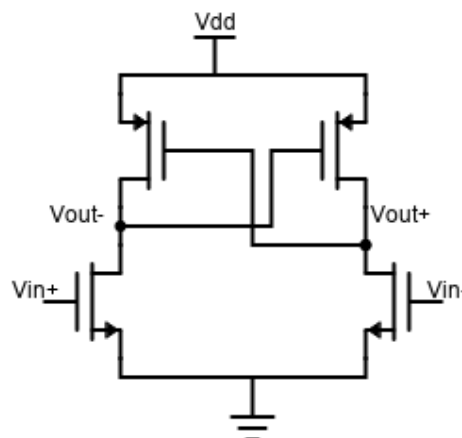


Figure 22. Differential inverter schematic

When an input is high, it lowers the value of the complementary output and forces the other output to be high. When the input is low, it does the opposite thing, resulting in a differential inverter, because both inputs cannot be the same.

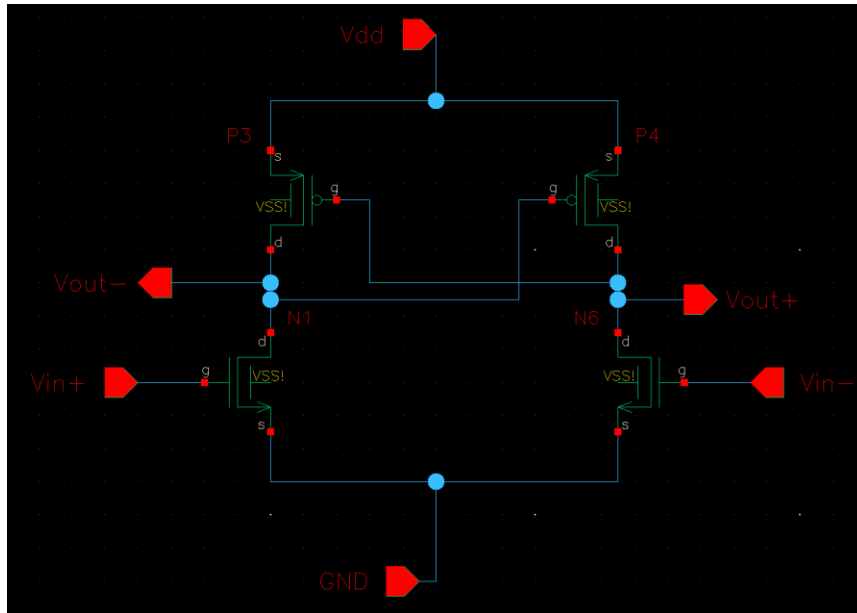


Figure 23. Differential inverter Cadence implementation

In figure 24 we can see the correct behaviour of the block: the signal gets a voltage range of almost 0 V to 800 mV (Vdd) and it gets cleaner.

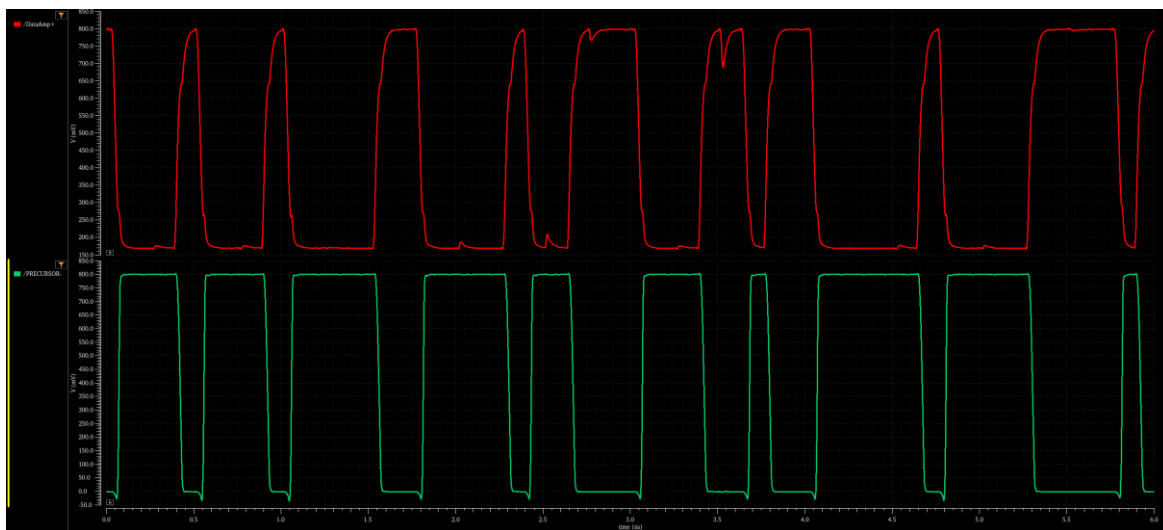


Figure 24. Differential inverter simulation

2.3.2. Delay (Z^{-1})

This block is a synchronous circuit. It delays 1 clock period the input signal, in our case 125 ps (for a clock frequency of 8 GHz). It is formed by cascading 2 D-latch. It works with 2 cycles, one when the clock signal is low, and another one when the clock signal is high.

At the first latch: when the clock is low, the input signal is copied to the middle node. When the clock gets high, the signal is fixed in the middle node.

At the second latch: when the clock is high, the middle node signal is copied to the output. When the clock is low, the signal gets fixed in the output node. This makes a rising edge flip flop.

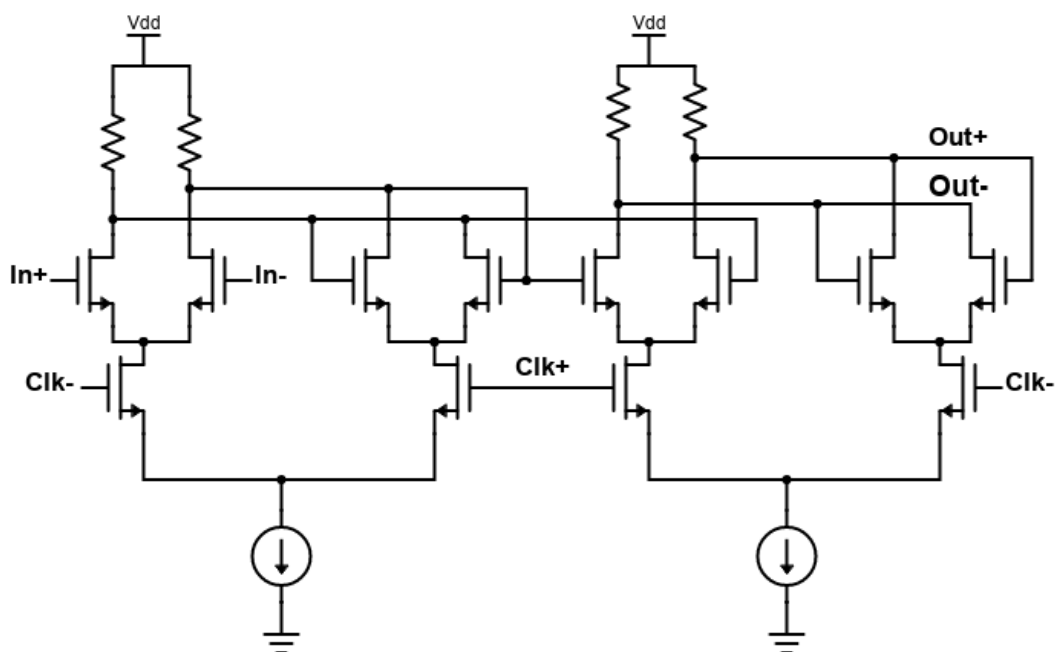


Figure 25. Delay schematic

Implementation of this design using Cadence in figure 26.

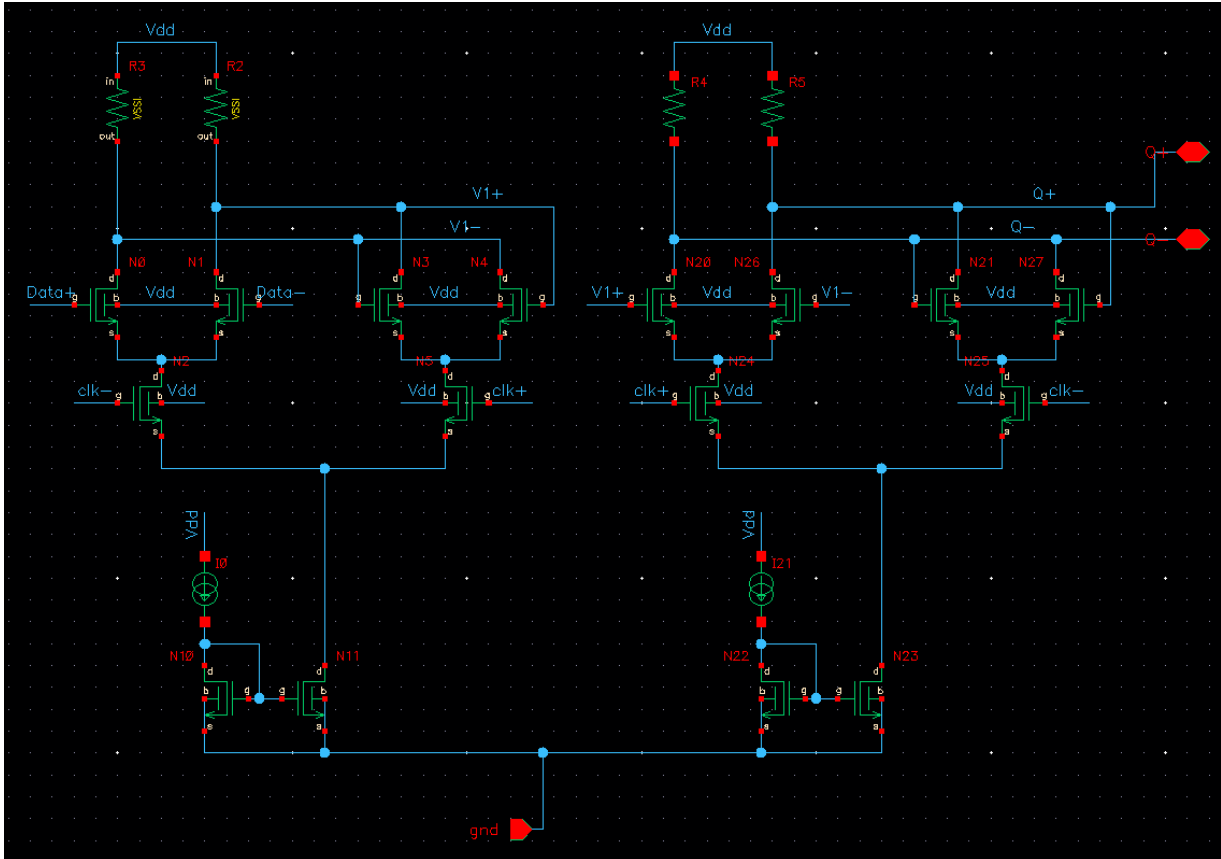


Figure 26. Delay Cadence implementation

This delay block does not give a clean signal in the output, so another differential inverter is implemented after it. We can see the 3 output signals in the simulation in figure 27.

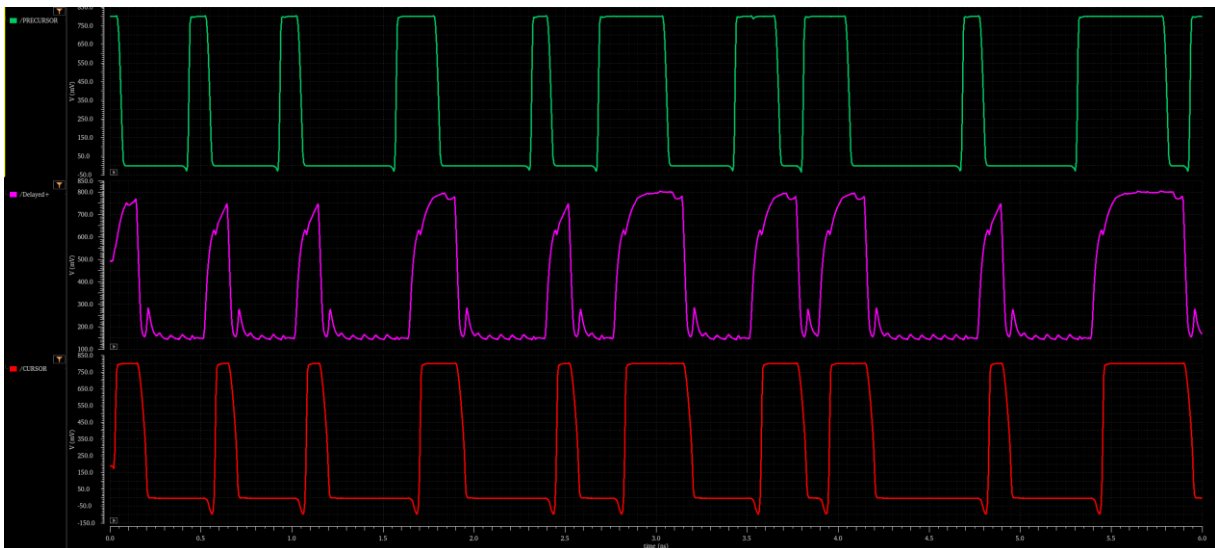


Figure 27. Delay simulation

In figure 27:

- Green: Before the delay
- Pink: After the delay, before the inverter
- Red: After the inverter

2.3.3. XOR

The XOR is only used to control the sign of the current to be added in each driver. One input will be the data stream and the other will be a control signal. This block is designed to change the signal if the control signal is low, and maintain it if the control signal is high.

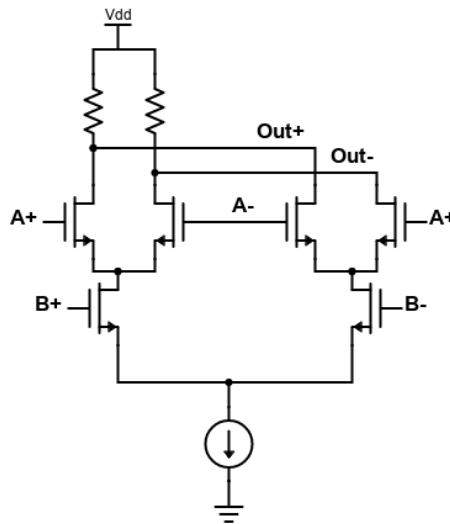


Figure 28. XOR schematic

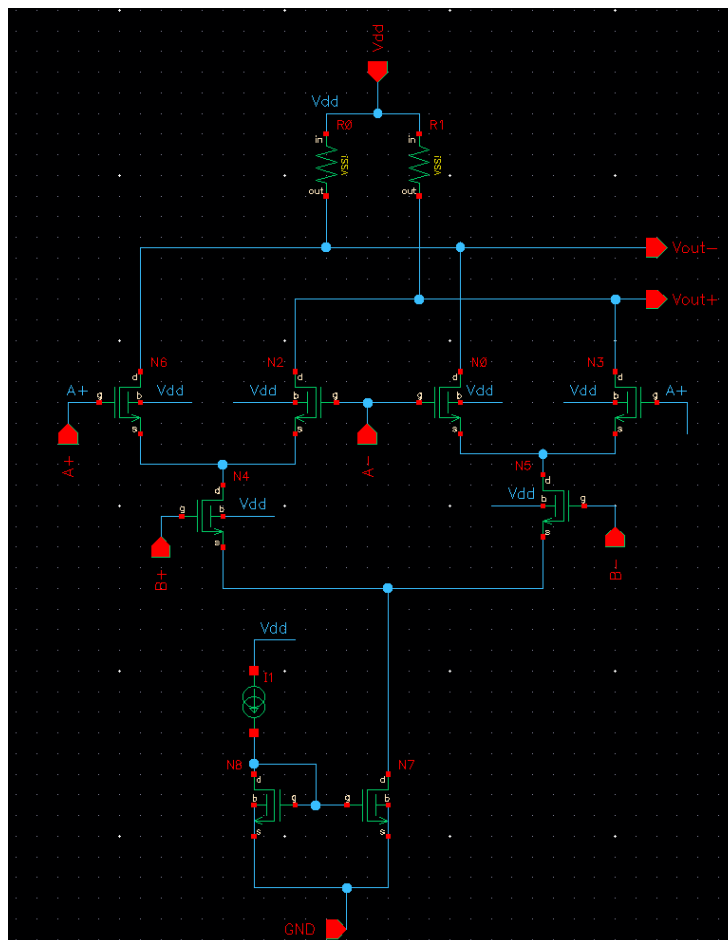


Figure 29. XOR Cadence implementation

A positive coefficient example is shown in figure 30 and a negative coefficient example is shown in figure 31:

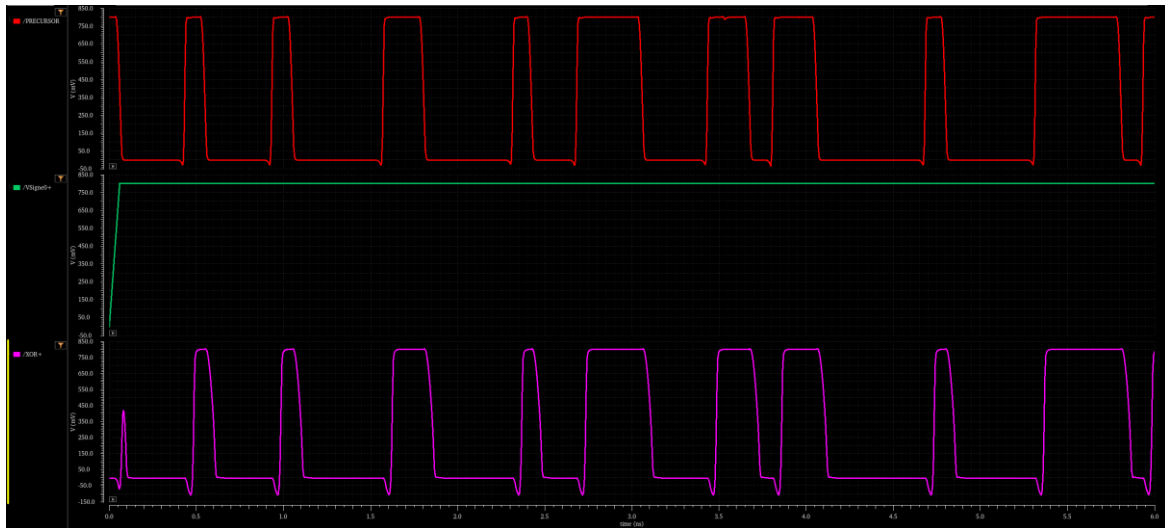


Figure 30. XOR simulation: positive coefficient

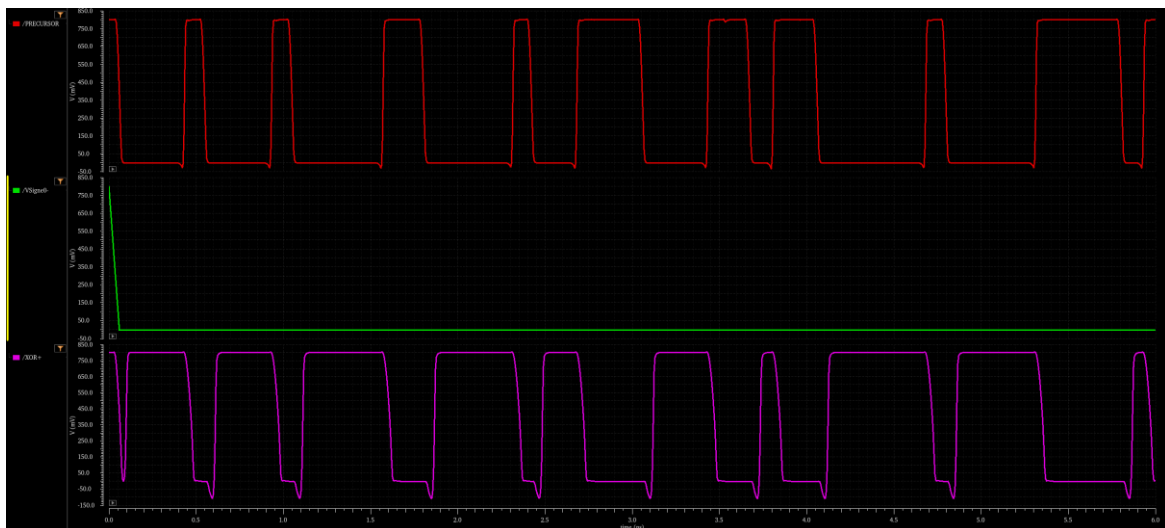


Figure 31. XOR simulation: negative coefficient

In figures 30 and 31:

- Red: before the XOR
- Gren: control signal
- Pink: after the XOR

2.3.4. Driver

The last step is to send the data through the channel. In our system there are 3 taps in parallel, so all of them will draw current from the same output resistors of $50\ \Omega$ and the channel. This resistance value is chosen to match the transmission line characteristic impedance at the output. The FIR filter coefficient of each tap is implemented using the current source of the tail node. If the coefficient is "1", the reference current must be $12\ \mu\text{A}$. Using 5 bits for the current source, we can use 32 possible coefficients for each tap. This block is the most consuming one by far.

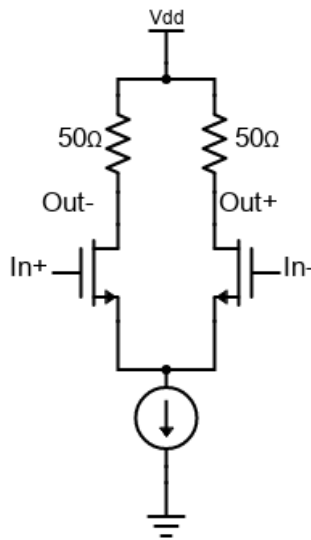


Figure 32. Driver schematic

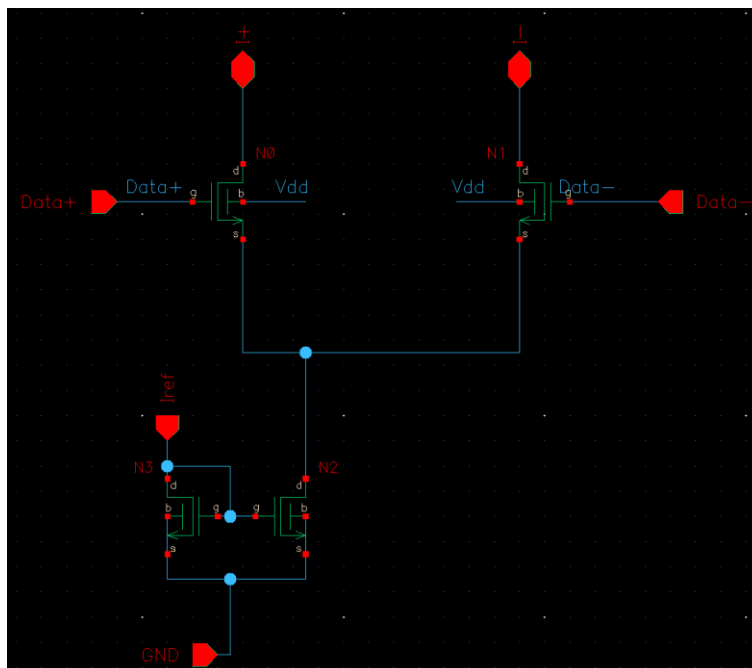


Figure 33. Driver Cadence implementation

As we can see in figure 34, the driver sends the signal correctly:

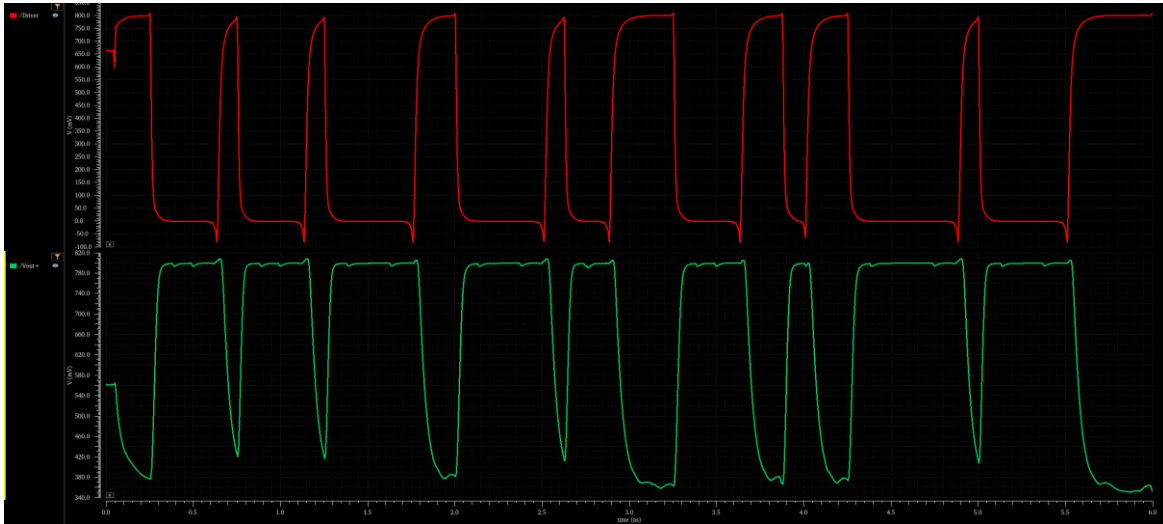


Figure 34. Driver simulation

3. Methodology / project development:

The main software used to design this system is Cadence Virtuoso, and Spectre is used for simulations. The research of related papers was made using specialized websites like IEEE Xplore^[6] or Research Gate^[7], and a lot of information was found in the project this system is based on^[10], as well on the official standards documentation or the manufacturer's documentation.

Each block was designed independently in Cadence Virtuoso, and then implemented step by step in the final schematic. Once everything was working in normal simulations, Monte Carlo simulations were used in order to find possible errors, and were solved one by one.

With the number of the failed Monte Carlo simulation and the seed for the pseudo-random generation, it was easy to get information about all the relevant waveforms and to check if the problem was solved. The method used was to follow the signal through all the blocks and detecting where the failure happened.

Some of the problems just needed more current in a specific block in order to be solved, but others required adjusting transistor widths to get a better common mode voltage in a specific node.

4. Results

The final Cadence Virtuoso design is the following:

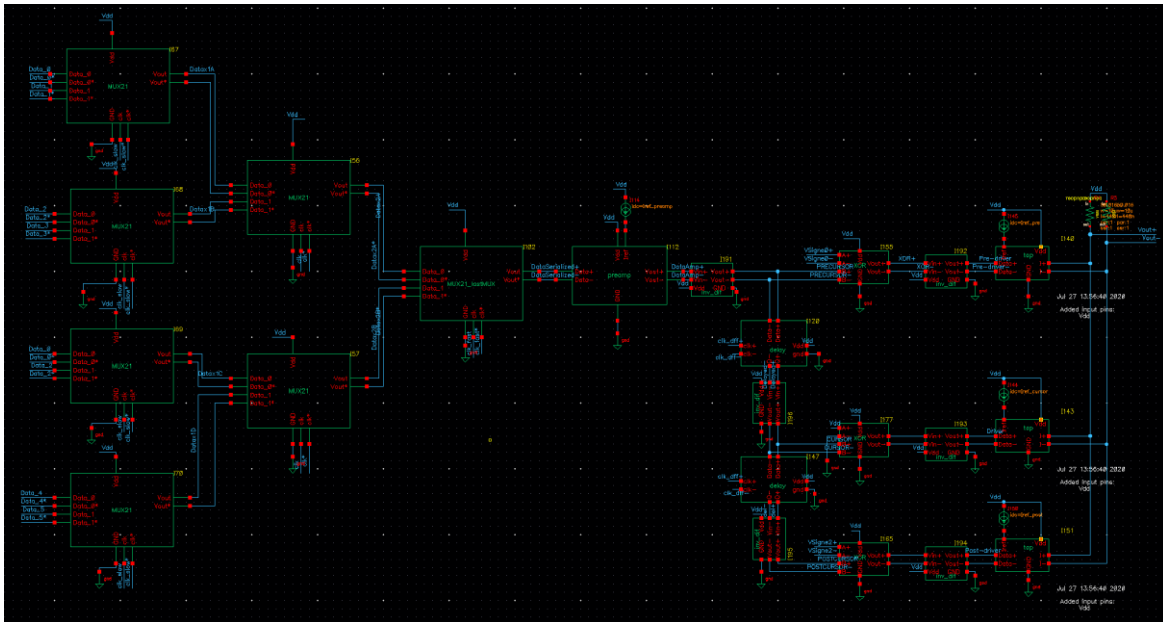


Figure 35. Full system schematic

The final system has been analysed with Monte Carlo simulations (200 points) to take into account process variations. Testing a sequence of 48 bits with bit-rate of 8 Gbps we achieved a nominal eye diagram opening of 398 mVpp (differential) when the output was loaded with a 100 Ω differential resistance (simulating the channel effect) and a 440 fF capacitor (simulating the output pad of the system).

The FIR coefficients for these measurements are set to [0, 1, 0] (no filtering). This was chosen for testing because no channel is measured in this setup, so there is no channel to equalize. Applying non zero coefficients for pre-cursor and post-cursor would distort the signal

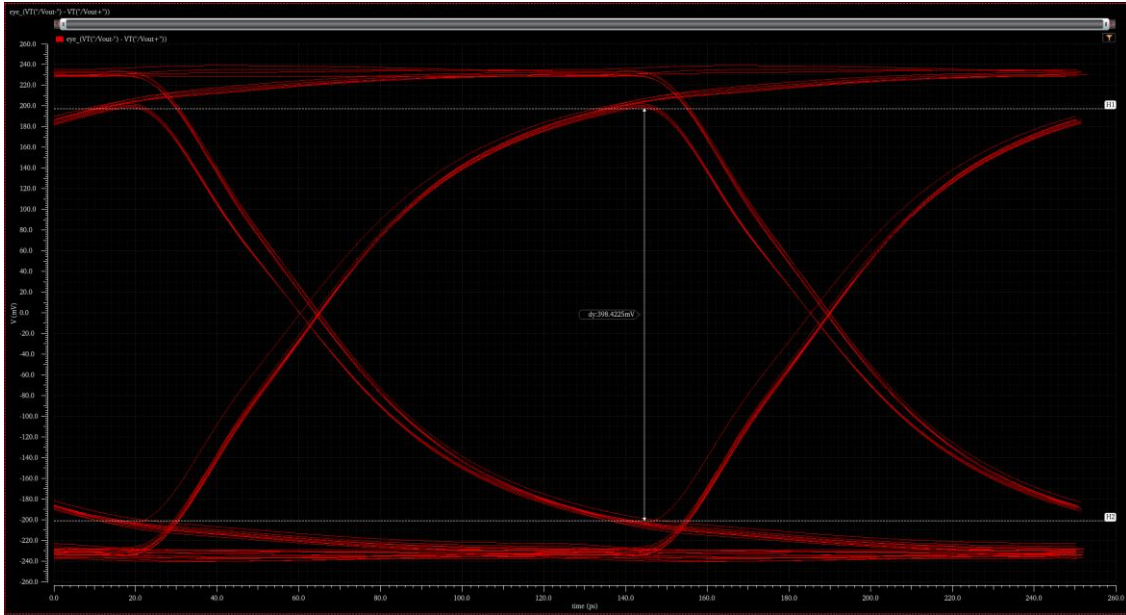


Figure 36. Nominal eye diagram at the output of the transmitter

The worst result of the Monte Carlo simulations was an eye opening of 242 mVpp:

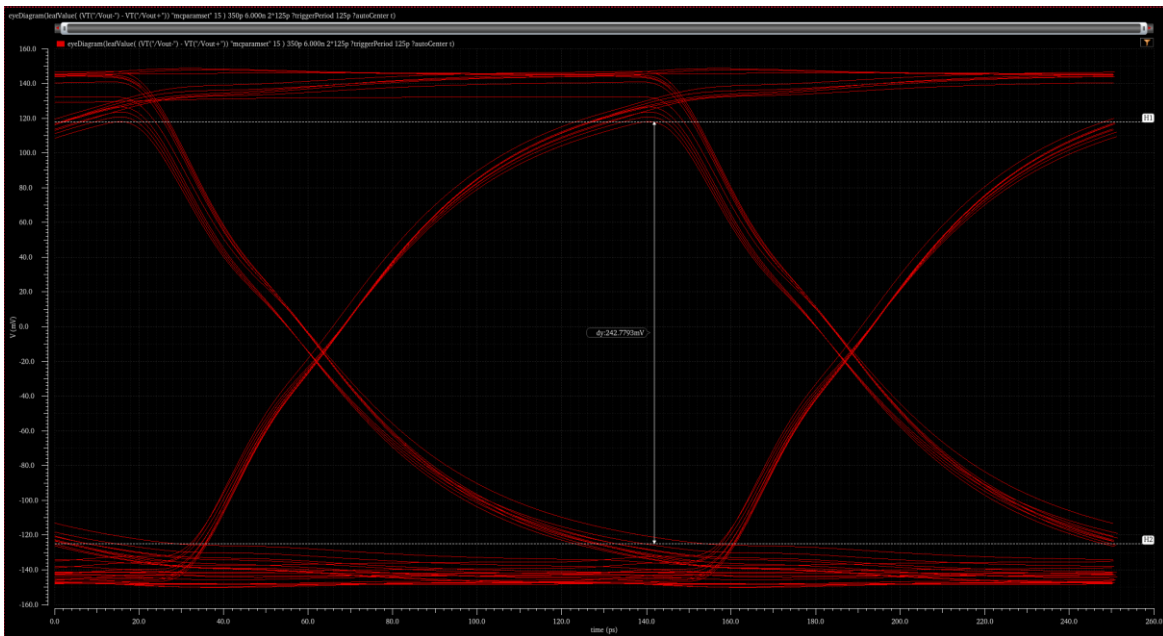


Figure 37. Worst case eye diagram at the output of the transmitter

The best result of the Monte Carlo simulations was an eye opening of 470 mVpp:

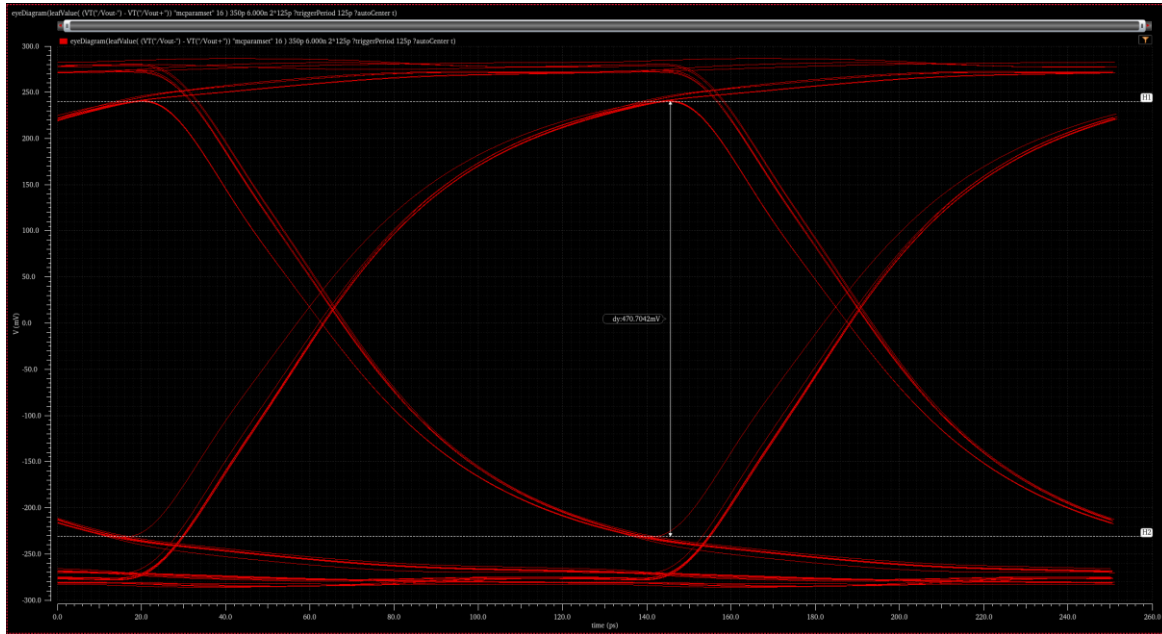


Figure 38. Best case eye diagram at the output of the transmitter

We achieved the 8 Gbps with an eye opening of 398 mV and Monte Carlo variations do not compromise the correct functioning of the system. However, the MC worst case has only 242 mV of eye opening.

Comparing these numbers with the CEI-6G specifications from Table 2, we do not meet that standard. CEI-6G needs between 400 mV and 750 mV, and this system's output differential voltage can be between 242 mV and 470 mV.

Comparing these numbers with Aurora specifications from Table 3, we do not meet that standard either. Aurora needs between 800 mV and 1600 mV. We would need an extra 558 mV of differential output voltage in order to fit in Aurora.

We have to take into account that neither CEI-6G or Aurora are meant for 8 Gbps. Aurora is designed to work at 1.25 - 3.125 Gbps^[3], being the maximum less than half the bit-rate of our system. CEI-6G is closer, with a maximum bit-rate of 6.375 Gbps.

At the output node we have a capacitance of 1.7 pF if we add the pad to the parasitic capacitances of the transistors.

The FIR filter coefficients can be adjusted by manipulating the current sources of the 3 drivers. If the coefficient must be negative, you can use the signal on the XOR block to do so.

The system needs 4 different clock frequencies: 1 GHz, 2 GHz and 4 GHz for the serializer, and an 8 GHz clock for the delay block at the filter.

The transmitter blocks receive reference currents of 5uA, and each driver's tap needs a 5-bit reference current up to 12 uA in order for the FIR filter to work properly. The total power consumption is 18.5 mW. Details in the next table:

Block	Consumption (mW)
6 x Multiplexer	0.202
Last Multiplexer	0.026
Pre-Amplifier	0.072
9 x Differential Inverter	0.232
Delay	0.146
XOR	0.171
Driver	17.7
Total	18.545

Table 3. Power consumption

5. Budget

As there is no prototype in this project, the costs are the engineer's work and the software licenses used, mainly.

The project started on 14th February and ended to 14th August, resulting in approximately 127 work days. Assuming 4h per day (508 h) and a salary of 9 €/h we get 4572 €.

The Cadence license should be added too. Cadence is not revealing their prices for their products, and it is not possible to know the cost of this license. However, we used an academic Cadence license and these prices are known.

The software used needs every year 600€ for "software only membership europractice" and 1890€ for "Cadence IC package (1-5 lic)". Since we used the software during 6 months, we will consider half the price.

Engineer work	4572 €
Software	1245€
Total	5772€

Table 4. Budget

6. Conclusions and future development:

Comparing the results with the CEI-6G (4.9 Gbps - 6.4 Gbps) or CEI-11G (9.9 Gbps – 11.2 Gbps) specifications, we do not satisfy any of them. Our data rate (8 Gbps) is in the middle of both standards and we cannot compare exactly, but our system could be improved. Aurora, in the other hand, differ a lot more in the bit-rate specifications, so CEI-6G/CEI-11G seem to be a better option.

For future development we could improve the vertical eye opening to meet one of these standards. Now we have a minimum opening of 242 mV and a maximum of 470 mV, meanwhile the CEI-6G standard has a minimum opening of 400 mV and maximum of 750 mV.

The rise and fall times should also be faster. This project achieved 50 ps of rise and fall times (20% to 80%), and the CEI-6G recommends a bit more than 30 ps. The main problem in order to satisfy this requirement is the output capacitance, which limits the maximum frequency of the output signal. A more complex digital FIR filter design could lower the driver transistor's width, and therefore the output capacitance associated with them (80% of the total output capacitance is related to the transistor capacitances and a 20% to the pad).

Bibliography:

- [1] E.Alon. "EE 290C: High-Speed Electrical Interface Circuit Design". *UC Berkeley*, 2011. [Online] Available: <http://www.infocobuild.com/education/audio-video-courses/electronics/ee290c-spring2011-berkeley.html> [Accessed: 13 August 2020].
- [2] S.Palermo. "ECEN 720: High-Speed Links Circuits and Systems". *Texas A&M University*, 2011. [Online] Available: <http://ece.tamu.edu/~spalermo/ecen720.html> [Accessed: 13 August 2020].
- [3] Aurora 8B/10B Protocol Specification, *Xilinx*, Oct 2014, v2.3. Available: https://www.xilinx.com/support/documentation/ip_documentation/aurora_8b10b_protocol_spec_sp002.pdf [Accessed: 13 August 2020].
- [4] Interlaken Protocol Definition, *Cortina Systems Inc and Cisco Systems Inc*, Oct 2008, v1.2. Available: http://interlakenalliance.com/wp-content/uploads/2019/12/Interlaken_Protocol_Definition_v1.2.pdf [Accessed: 13 August 2020].
- [5] IA Title: Common Electrical I/O (CEI) - Electrical and Jitter Interoperability agreements for 6G+ bps, 11G+ bps, 25G+ bps I/O and 56G+ bps, *OIF*, Dec 2017. Available: <https://www.oiforum.com/wp-content/uploads/2019/01/OIF-CEI-04.0.pdf> [Accessed: 13 August 2020].
- [6] IEEE Xplore, [Online]. Available: <https://ieeexplore.ieee.org/Xplore/home.jsp>
- [7] Research Gate, [Online]. Available: <https://www.researchgate.net/>
- [8] Kintex-7 FPGAs Data Sheet: DC and AC Switching Characteristics, *Xilinx*, June 2019. Available: https://www.xilinx.com/support/documentation/data_sheets/ds182_Kintex_7_Data_Sheet.pdf
- [9] Channel Operating Margin, COM, a new method to compliance check high speed serial links, May 2017. [Online] Available: <https://www.electronic.se/2017/05/05/channel-operating-margin-com-a-new-method-to-compliance-check-high-speed-serial-links/>
- [10] J.Wright. "Design of a Lightweight Serial Link Generator for Test Chips", *UC Berkeley*, 2017. Available: <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2017/EECS-2017-220.html>
- [11] Y.Luo, "A high speed serializer / deserializer design" , *University of New Hampshire*, 2010. Available: <https://scholars.unh.edu/cgi/viewcontent.cgi?article=1535&context=dissertation>
- [12] J.Schutt-Aine, "Signal integrity for high-speed design", *University of Illinois*, 2020. Available: <http://emlab.uiuc.edu/ece546/>
- [13] J.Zerbe, "Equalization and Clock Recovery for a 2.5–10-Gb/s 2-PAM/4-PAM Backplane Transceiver Cell", 2003. Available: <https://ieeexplore.ieee.org/abstract/document/1253859?section=abstract>
- [14] Europractice membership prices.
Available: <http://www.europractice.stfc.ac.uk/membership/membership.html>
- [15] Europractice software prices. Available: http://www.europractice.stfc.ac.uk/software/software_price.html

Glossary

TX: Transmitter

RX: Receiver

FIR: Finite Impulse Response

CTLE: Continuous time linear equalizer

DFE: Decision feedback equalizer

UI: Unit Interval

ISI: Inter Symbol Interference

ASIC: Application-Specific Integrated Circuit

FPGA: Field Programmable Gate Array

DDR3: Double Data Rate Type 3

RAM: Random Access Memory

CML: Current Mode Logic

XOR: Exclusive OR gate

Clk: Clock

COVID-19: Coronavirus disease of 2019

CAD: Computer Aided Design

IDE: Integrated Development Environment

SATA: Serial Advanced Technology Attachment

PCI: Peripheral Component Interconnect

CEI: Common Electrical I/O

IEEE: Institute of Electrical and Electronics Engineers