A 10 Gb/s/wire Inverter-Based Simultaneous Bi-Directional Transceiver in 45nm CMOS

Griffin Beams, Jason Bentley

***Abstract*— The slowing down of Moore's Law has led to new methods of constructing integrated circuits, notably by combining multiple chiplets on a single package. However, the primary challenge in this approach is the communication between chiplets. Various techniques are being explored to increase this communication speed, including non-return-to-zero (NRZ), Pulse Amplitude Modulation-4-level (PAM4), and simultaneous bi-directional signaling (SBD). While NRZ and PAM4 have significant limitations, SBD presents an effective solution, particularly when using a hybrid circuit design, involving a replica driver, voltage adder, and a transimpedance amplifier. The implementation of this design promises efficient communication between chiplets without a considerable increase in power and area overhead.**

I. Introduction

The need for faster computing stops for no one. For over 50 years this increased need was satisfied by what is known as Moore's law, the doubling of transistors on an integrated circuit every two years. In recent years Moore's law has been slowing down, reducing the number of transistors that can fit onto an integrated circuit and limiting how much the performance can be increased. This has given rise to a new method of constructing integrated circuits to gain back this performance increase. Combining multiple chiplets on a single package to have a greater area overall for transistors. This makes up for the scalability loss with the decrease of Moore's law. While this solution is effective, it introduces various new issues that need to be handled, especially as bandwidth speeds increase. The primary issue with using multiple chiplets is communication between them.

The biggest issues with communication are creating systems that can communicate fast enough for required bandwidth and limited wire space. The issue with creating systems that can communicate fast enough is that speed increases from more advanced transistor devices stopped around 10nm. This means other avenues will be needed to get the needed speeds. Along with this is the limited number of pins and wires available for said communication. As these packages are small, there is only so much space that can be used for communication, so space needs to be used efficiently.

There are various methods to get the speed increase needed for modern speed requirements. These are non-return-to-zero (NRZ), Pulse amplitude modulation-4-level (PAM4), and simultaneous bi-directional signaling (SBD). All of these have their own benefits and drawbacks that need to be carefully considered. This is especially the case with the context of being used for communications between chiplets.

The first method to get increased speed is NRZ. This involves going directly from the current voltage to the opposite one without returning to zero voltage. For example, if the transceiver was going from a 1 to a 0, It would go from a positive voltage directly to a negative voltage without stopping at zero voltage. This gives a significant speed up compared to return-to-zero, but not enough for modern performance needs. A particular issue is that to get future proof bandwidth, the infrastructure and power requirements are far too great to be practical. This leads to the need for an effective bandwidth doubler.

An effective bandwidth doubler being used more and more is PAM4. This works by having four voltage levels to represent two bits at a time, rather than the one bit that can be represented by NRZ. This doubles the amount of information that can be translated at the same time. This allows PAM4 systems to achieve high speeds that would be impractical for an NRZ system. The issue comes with the increased infrastructure requirements to make PAM4 work. When the signal is split between four different states instead of 2, it increases the requirements of the receiver to distinguish between the different levels and the transceiver to maintain the levels with reduced margins. This requires more advanced systems and greater area requirements to make PAM4 operate properly. In the context for communication between chiplets, this is a great cost where space is limited leading to the bandwidth doubler that this paper is focused on.

Simultaneous Bi-Directional signaling is a method to send two signals on the same wire at the same time. This effectively doubles that bandwidth and as will be discussed, the infrastructure to do so is more efficient than PAM4. An additional benefit is the greater wire and pin efficiency that the SBD has. This is because if 2 chiplets can send signals on 1 wire, they can communicate with each other on one wire instead of 2.

II. Background

The core problem that needs to be solved to enable simultaneous bi-directional signaling is being able to isolate the incoming signal. Since signals are sent both ways on the wire, the outgoing signal will be constructively and destructively interfering with the incoming signal. This becomes an issue when the receiver only wants the incoming signal as the system already knows what it is sending out. There are a few different methods to isolate the incoming signal, but they all work on the same principle: finding a way to subtract the outgoing signal to isolate the incoming signal.

A common method to do this subtraction is to feed the combined signal into a differential amplifier as well as a replica of the outgoing signal. This will amplify the difference between the two signals, which is the incoming signal. The issue with this system though is the need for a highspeed differential amplifier and a dummy RC for the replica. The high-speed differential amplifier is a significant design undertaking that can also consume a significant amount of space and power overhead. A dummy RC is also required to match the replica with the combined signal to ensure the incoming signal alone is isolated. These issues combined make this solution lacking and a better solution is wanted.

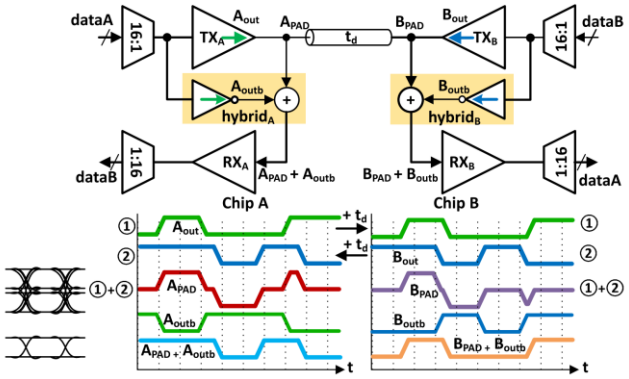

Fig. 1. Simultaneous bi-directional signaling. [2]

Fig. 1 illustrates the design this paper will be focusing on, a hybrid design that uses a replica driver to produce the inverse signal of the outgoing signal, then using essentially a voltage adder being fed into a transimpedance amplifier is isolate the incoming signal. The replica used to produce the inverse of the outgoing signal is a similar design to the transceiver, just with an added inverter to convert the signal. This makes the replica relatively simple to design and will take up relatively less room on an integrated circuit. The transimpedance amplifier (TIA), like the replica, is relatively simple to design with the most common design being an inverter with a feedback resistor. This also takes up significantly less area than a differential amplifier. These changes also eliminate the need for an accurate dummy RC to slow the replica down appropriately. This reduces the amount of calibration needed which can simplify this system to a product.

In addition to the replica and transimpedance amplifier, this design and any other SBD design is the need for clock forwarding architecture. This is vital to ensure that both chiplets are operating at the correct clock to send and receive data at the correct times relative to each other. Clock forwarding also has the advantage of automatically introducing the needed delay to match the delay seen across the link. This greatly simplifies the process as it is automatic and clock forwarding can also be set to have feedback to ensure the chiplets are always working on the correct clocks to function.

Overall, the design of a hybrid circuit has significant advantages that allow SBD signaling to be more viable which has great promise for communication between chiplets on the same package. By using a replica circuit and a voltage adder to isolate the incoming signal, then feeding that into a transimpedance amplifier this is simpler and has a smaller area and power overhead than other solutions. These components are also relatively simple to design compared to other topologies, namely designing a highspeed differential amplifier. SBD also requires the use of clock forwarding to function reliably, but this has the advantage of making clock delay matching an automatic process. In total this design promises to be an effective bandwidth doubler for communication between chiplets that will not consume a preclusive amount power and area to be viable.

III. Design Parameters

In the Simultaneous Bidirectional (SBD) transceiver (Tx/Rx) topology, efficient cancellation of the outgoing signal from the resultant combined incoming and outgoing signal at the node V_{pad} is critical. This process facilitates the isolation of the incoming signal, which can subsequently be routed into a receiver. Achieving this isolation necessitates careful selection of transistor sizes and resistor values. Fig. 2 illustrates the model for one side of the SBD topology. Achieving this isolation necessitates careful selection of transistor sizes and resistor values. In order to nullify the outgoing signal from the incoming one, the device sizes and resistor values need to be defined such that the hybrid current, I_{hyb}, originating from V_{pad} is afforded a return path to ground via the resistive path $R_{rep} + R_{h2}$.This configuration ensures that the voltage at node V_{rx} is nullified, thereby canceling the signal contribution from the transmitter. As illustrated in Fig. 2(c) The residual current flowing through V_{rx} constitutes the signal contribution from the incoming signal.

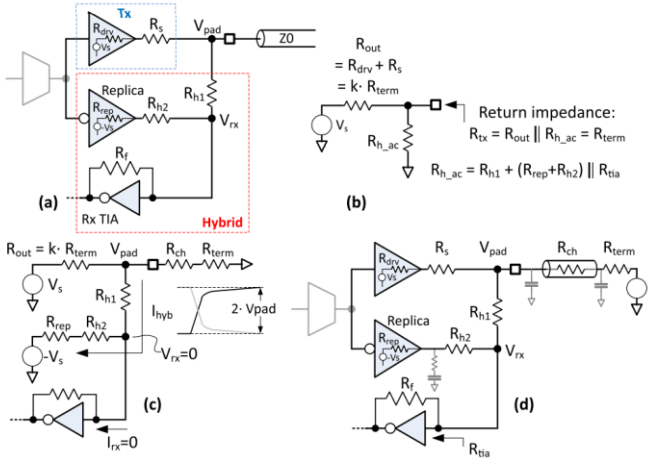

Fig. 2. Modeling of SBD hybrid. (a) Main driver and hybrid diagram. (b) Modeling for driver impedance calculation. (c) Modeling for V_{pad} calculation. (d) Modeling for V_{rx} calculation. [2]

From Fig. 2(c), due to $V_{rx} = 0$ we can assume this is a virtual ground. From this we can then write that

$$I_{hyb} = \frac{V_{pad}}{R_{h1}}. \tag{1}$$

However, to find the value of V_{pad} we must perform a KCL analysis at this node. Doing so we get that

$$V_{pad} = \frac{V_s R_{h1}(R_{ch}+R_{term})}{R_{h1}[R_{ch}+(1+k)R_{term}]+kR_{term}(R_{ch}+R_{term})}. \tag{2}$$

Combining (1) and (2) we can then write I_{hyb} as

$$I_{hyb} = \frac{V_{\text{pad}}}{R_{h1}} = \frac{V_s(R_{ch}+R_{term})}{R_{h1}[R_{ch}+(1+k)R_{term}]+kR_{term}(R_{ch}+R_{term})}. \quad (3)$$

From (3) we can see that the node V_{pad} acts as a current divider where a portion of the current is passed through R_{h1} as the current I_{hyb} while the remaining current is passed through the channel to be received by a downstream Rx. To calculate the values of $R_{h2} + R_{rep}$, which will create a return path to ground for I_{hyb}, we can first write

$$-I_{hyb} = \frac{-V_s+V_{rx}}{R_{rep}+R_{h2}}. \quad (4)$$

Then, combining (3) and (4), after some trivial algebraic manipulation we get

$$R_{rep} + R_{h2} = R_{h1}\left[1 + kR_{term}\left(\frac{1}{R_{ch}+R_{term}} + \frac{1}{R_{h1}}\right)\right]. \quad (5)$$

Additionally, we need to derive an equation which will allow us to find the k term. Referring to Fig. 2(b) we can write

$$R_{term} = kR_{term}||\left[R_{h1} + \left(R_{rep} + R_{h2}\right)||R_{tia}\right]. \quad (6)$$

After some significant algebraic manipulation and canceling out R_{term} we can write

$$1 = \frac{kR_{tia}\left(R_{h1} + R_{h2} + R_{rep}\right) + R_{out}R_{h1}\left(R_{Rep} + R_{h2}\right)}{R_{tia}\left(R_{out} + R_{h1} + R_{h2} + R_{rep}\right) + \left(R_{out} + R_{h1}\right)\left(R_{rep} + R_{h2}\right)}, \quad (7)$$

where $R_{out} = kR_{term}$.

IV. Sizing the SBD

A. Calculating Resistor Values

From the relationships derived in III we can see that we can make arbitrary choices for any resistance variable and then tune the other resistor values such that these mathematical relationships hold. For this design we chose $R_{h1} = 190\Omega$, $R_{ch} = 21.5\Omega$, and $R_f = 1200\Omega$ according to the values presented in Y. Nishi, et al. [2]. Additionally, we also chose $R_{rep} = 250\Omega$ for ease of designing the circuit. First, we must find the input resistance of the TIA, R_{tia}. To do this, we create a test bench in Cadence and simulate our TIA design with a 10Gb/s bitstream using a 1A current source and run an ac simulation. We then measure $V_{tia} = (V_{in} - V_{out})$ which will be the resistance value of the TIA seen by the current source (since $R_{tia} = V_{tia}/1A$). Using this method, we find that our TIA design has an input resistance of $R_{tia} \approx 4000\Omega$. From this result we can then approximate (7) as

$$1 = \frac{k\left(R_{h1} + R_{h2} + R_{rep}\right)}{\left(kR_{term} + R_{h1} + R_{h2} + R_{rep}\right)}. \quad (8)$$

From Y. Nishi, et al. [2] we know that the characteristic impedance of the channel is 40Ω so we set our target termination value, $R_{term} = 40\Omega$. Therefore from (8) we have that

$$k40 + 440 + R_{h2} = k(440 + R_{h2}). \quad (9)$$

As suggested by Y. Nishi, et al. [2] we then iteratively solve for R_{h2} and k using an excel spreadsheet. From this method we extract the values of $k = 1.049$ and $R_{h2} = 115\Omega$. From this we then choose $R_s + R_{drv} = kR_{term}$ such that $R_s = R_{drv}$. Then to further confirm these values we ensure that (3) is equal to (4) or comes very close. Doing this calculation, we find that (3) $\approx 2.7398mA$ and (4) $\approx 2.7397mA$. From these results only $100nA$ of residual current from the outgoing signal will be seen by the TIA. This accounts for approximately 0.004% of $I_{hyb} = 2.7398mA$ (from (3)) at the input of the TIA. Table I summarizes the chosen and calculated equivalent resistor values.

TABLE I
Equivalent Resistor Values For The SBD

Resistor	Ω
R_s	21
R_{drv}	21
R_{h1}	190
R_{h2}	115
R_{rep}	250
R_{ch}	21.5
R_{term}	40
R_{tia}	4000

B. Device Sizing

Sizing the devices was done using an iterative approach. Given the equivalent resistance values in Table 1 we had to ensure that the output resistances, r_o, of both PMOS and NMOS for all CMOS inverters were equal, and when the CMOS inverters were summed in parallel for each hybrid and driver the total output resistance would equal the output resistances found un IV.A.

The Driver of the SBD presented in this report consists of 22 CMOS inverters in parallel. Each inverter likewise was a minimum size inverter on the input functioning as a driver. This is illustrated in Fig. 3. Using an iterative approach, we found that a PMOS size of 3.11 μm and an NMOS size of $2.34\mu m$ yielded an output resistance of 462Ω. When 22 of these blocks are combined in parallel to form the driver, we see an output resistance of 21Ω.

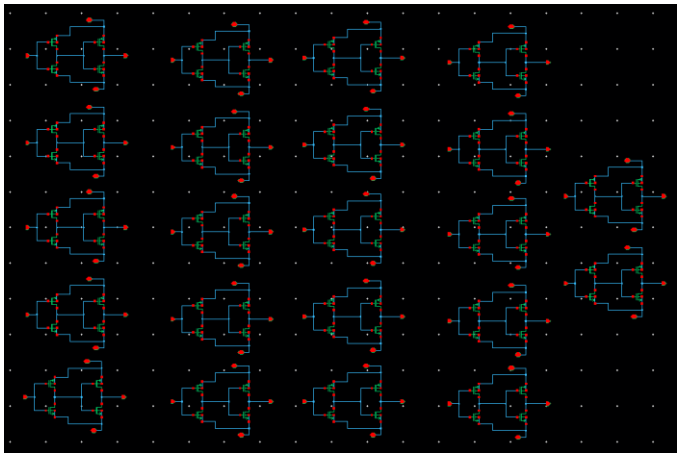

Fig. 3. *The internal structure of the Tx Driver. In each individual circuit the CMOS invt on the left is the driving circuit for the CMOS invt on the right.*

Likewise, for the replica driver, to achieve a 250Ω output resistance we found that 8 CMOS inverters in parallel, each with an output resistance of 2000Ω will realize this requirement, as illustrated in Fig. 4. Through an iterative process we choose the PMOS to be $805nm$ and the NMOS to be $575nm$.

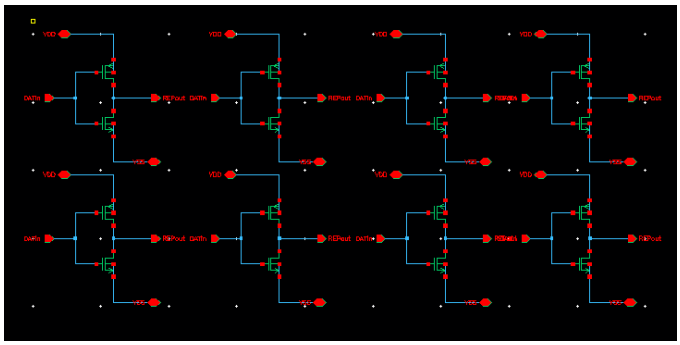

Fig. 4. *The internal structure of the hybrid replica driver.*

As illustrated in Fig. 5 the TIA is a single CMOS inverter. In Y. Nishi, et al. [2] the TIA devices were sized to 24 times their minimum size. Furthermore, through experimentation we found that an NMOS to PMOS device size ratio of 1.6 generally yielded devices with similar characteristics. Using the information presented in Y. Nishi, et al. and our own experimentation we initially choose the NMOS size to be $1\mu m$ and the PMOS to be $1.6\mu m$, and an R_f value of 1200Ω which was also given in Y. Nishi, et al. Once the circuit was complete, we then used an iterative approach to find the ideal sizes for R_f and the CMOS transistors. Ultimately, we chose $R_f = 1800\Omega$, while the NMOS and PMOS were both chosen to be $1\mu m$. The device sizes for all devices in this design are listed in Table II.

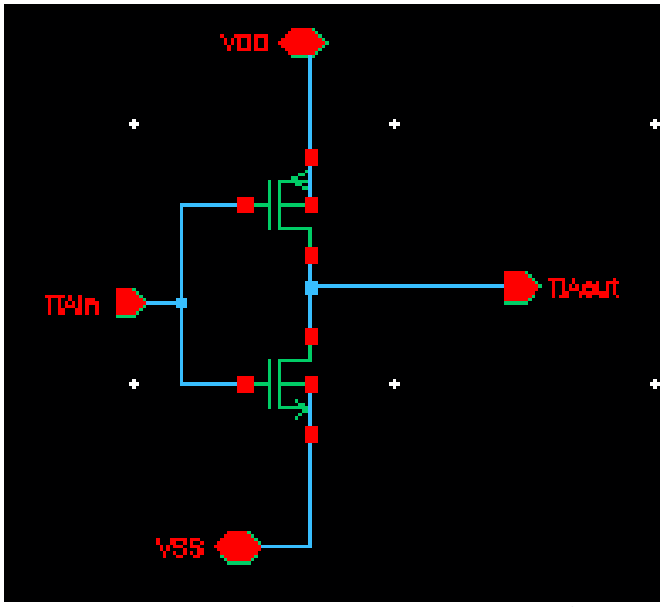

Fig. 5. *A simple CMOS inverter makes up the internal structure of the TIA*

TABLE II
TRANSISTOR DEVICE SIZES

Device	Width
Driver NMOS	$2.34\mu m$
Driver PMOS	$3.11\mu m$
Replica NMOS	$575nm$
Replica PMOS	$805nm$
TIA NMOS	$1\mu m$
TIA PMOS	$1\mu m$

C. *Other Parameters*

In addition to the transistor devices and resistors there are also two other elements in this circuit: the channel model and a dummy RC on the output of the replica. For the channel model we used the channel resistance provided in Y. Nishi, et al. of $R_s = 21.5\Omega$ [2]. In addition, the channel model also has a $100fF$ capacitor on each side of the channel to model the pad capacitances of the chip-to-chip interconnects. Likewise, the dummy RC on each of the replicas attempts to mimic the impedance of the channel and similarly uses a series RC of 21.5Ω and $100fF$ in parallel with the output of the replica driver.

V. RESULTS

The results of the design presented in this report were obtained through Cadence simulations at a speed of 10Gb/s. To simulate the bit-stream data we used two random pulse generators, each with a different seed to ensure that we are not generating the same pseudo-random stream. The pulse generators were set to a V_s of $1V$ with a $100ps$ period and a $10ps$ rise and fall time. These bit-streams were then directly fed into the Driver and Replica of our design. Fig. 6 shows a schematic of our final design. The output waveform compared to the input data is illustrated in Fig. 7 where the top waveform is the input data on the right side of the circuit and

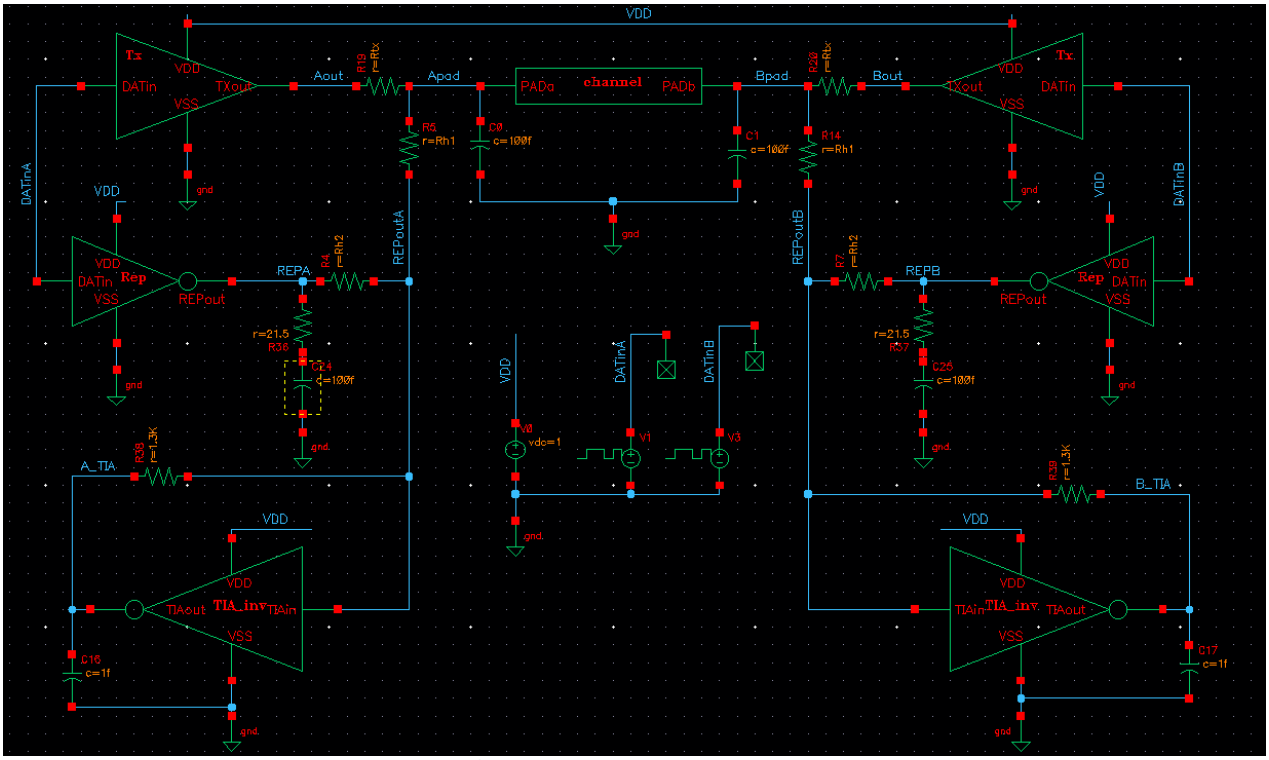

Fig. 6 *The top level block-diagram/schematic for the design presented in this report.*

the bottom waveform is the output data which our design retrieves after the TIA on the left side of the circuit. We notice an expected delay between the input data and the retrieved output data. Additionally, we also observe some significant noise on the output data. The source of this noise can be attributed to the $100nA$ leakage current from I_{hyb} seen by the TIA, as previously discussed IV. A. Additionally, we observe approximately a $200mV$ swing around a bias voltage of $500mV$.

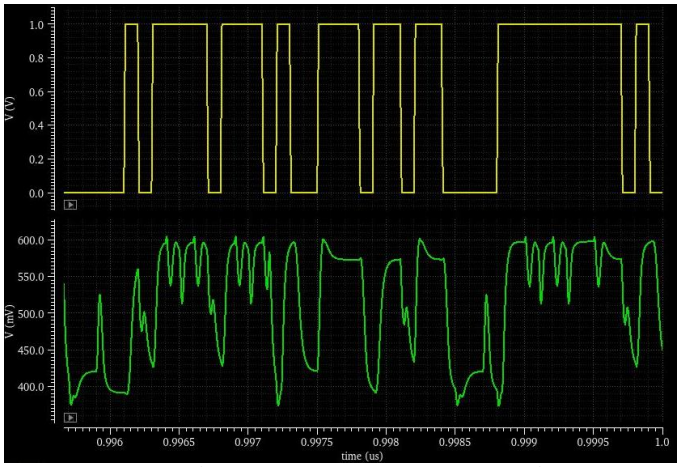

Fig. 7. *The top wave form shows the input data on incoming from the right side of the circuit (DATinB). Bottom waveform shows this data extracted from the combination of incoming and outgoing data on the bottom left side of the circuit (A_TIA).*

To gain a more accurate understanding of this design, Fig. 8 presents the eye-diagram for our SBD. From this diagram we observe an eye opening of $100mV$ and a width of $36ps$. Therefore, while the eye opening is comparable to that of the results in Y. Nishi, et al., the eye width shows that there is significant jitter in our design. It should be noted that the design in Y. Nishi, et al. implements significant timing and delay circuitry to mitigate jitter and increase the eye width, whereas our design purely focuses on implementing the driver and replica portion of the architecture.

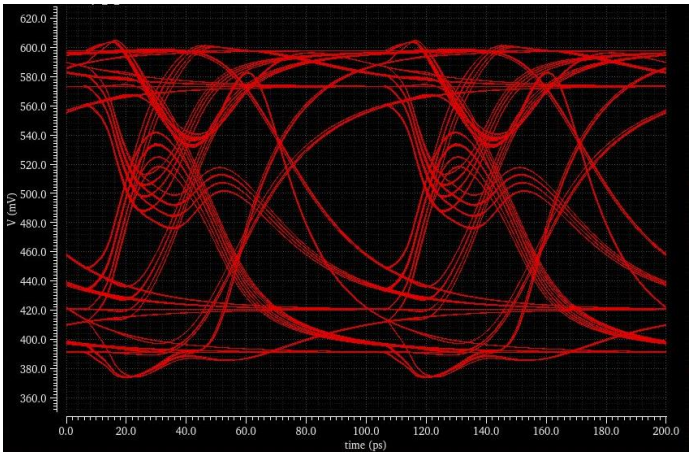

Fig. 8. *The eye diagram for the design presented in this report. At 100ps an eye opening of 100mV and eye width of 36ps is observed.*

Table III presents our results compared to Y. Nishi et al, and other works.

TABLE III
RESULTS OF VARIOUS WORK

	Our Work	Y.Nishi, et al. [2]	VLSI21 [3]	JSSC20 [4]	JSSC16 [5]
Technology	45nm	5nm	7nm	7nm	28nm
Supply[mV]	1000	750	800	800, 300	N/A
Data Gb/s	10	50.4	20	8	20
pJ/Bit	0.634	0.297	0.46	0.56	0.3

VI. Conclusion

In this paper we have designed and simulated a simultaneous bi-directional transceiver in cadence virtuoso that can operate at 10Gb/s. Our design used a hybrid topology as described in Y. Nishi's paper in which we used a replica driver and TIA to great effect to isolate the incoming signal to a receiver to allow SBD. After taking care to understand and match all resistors and divide sizes, we have a function SBD transceiver.

Overall, our design functions at 10Gb/s even if the signal is messy. We have an eye width of $36ps$ and an eye opening of $100mV$ which is enough to be used by an accurate clock system and comparator. Most of this slow down and signal jittering could be fixed by using a smaller device technology or by implementing the other architecture needed for this system, but these are outside the scope of this project. This design could also be slowed down and tweaked to work at a slower speed such as 5Gb/s where the signals would be far clearer and more reliable. This is because it appears 10Gb/s is around the limit that 45nm technology can handle in an SBD format. Ultimately, considering the size we are using compared to the paper, it appears to be functioning fairly well all things considered.

Given more time our design could be improved by additional testing and optimization to either get it functioning better at 10Gb/s, or by finding the sweet spot for speed for our design and technology size. While we did follow the equations to calculate device sizes and resistor values, often non idealities can make these calculated values off the mark. Without a greater intuition of how the devices in SBD topologies interact, brute force testing may be used in lieu of intuition to find the ideal values for device and resistor sizes. Fortunately, Cadence has parametric analysis which allows the user to set a range of values to test which cadence will automatically do. This is great for optimizing designs, but the issue comes with how long this can take. This is especially the case for our design where we have many resistors and transistors that need to be looked at for optimization. This means we simply ran out of time to do these long tests. Parametric analyses would be the next step we would take if we had more time.

In addition to what could be improved for the transceiver, there could also be work done on all the architecture surrounding the transceiver to make it function. Again, this is outside the scope of this project, but it will need to be done for implementation into physical products. Specifically, the clock forwarding architecture is vital for making the system function and has the added benefit of improving the performance of the transceiver by delay matching reducing the jitteriness of the signal seen in our results.

Overall, this project proves that a 10Gb/s SBD transceiver can be built using a $45nm$ technology. This speed is near the max the device sizes can handle, but it is possible. Using more advanced devices with smaller sizes and increased speed has proven to be an effective solution to improve the communication speeds between multiple chiplets on a single package, while being space and power efficient.

VII. References

[1] B. Dally, "Insights from Nvidia Research | Bill Dally," YouTube, https://www.youtube.com/watch?v=W6k2Q45nlA4 (accessed Jun. 1, 2023).

[2] Y. Nishi *et al.*, "A 0.297-PJ/bit 50.4-GB/s/wire inverter-based short-reach simultaneous bi-directional transceiver for die-to-die interface in 5-NM CMOS," *IEEE Journal of Solid-State Circuits*, vol. 58, no. 4, pp. 1062–1073, 2023. doi:10.1109/jssc.2022.3232024

[3] Y.-Y. Hsu, P.-C. Kuo, C.-L. Chuang, P.-H. Chang, H.-H. Shen, and C.-F. Chiang, "A 7 nm 0.46pJ/bit 20Gbps with BER 1E-25 die-todie link using minimum intrinsic auto alignment and noise-immunity encode," in Proc. Symp. VLSI Circuits, Jun. 2021, pp. 1–2.

[4] M.-S. Lin et al., "A 7-nm 4-GHz Arm1-core-based CoWoS1 chiplet design for high-performance computing," IEEE J. Solid-State Circuits, vol. 55, no. 4, pp. 956–966, Apr. 2020.

[5] B. Dehlaghi and A. C. Carusone, "A 20 Gb/s 0.3 pJ/b single-ended dieto-die transceiver in 28 nm-SOI CMOS," in Proc. IEEE Custom Integr. Circuits Conf. (CICC), Sep. 2015, pp. 1–4.