Design of a PAM-4 Driver Circuit for MZI Optical Moduators using eSim and Skywater 130 PDK

Subhash C. Arya
Lightwave Communication System Laboratory, Department of Electronics and Communication Engineering, NEHU, Shillong
aryasubh@yahoo.co.in

***Abstract*— This paper presents the design of a driver circuit for generating a four-level Pulse Amplitude Modulation (PAM-4) format in Mach-Zehnder Interferometer (MZI) optical modulators. Higher-level modulation formats are inevitable for enhancing spectral efficiency and achieving high data rates in silicon-on-insulator (SOI) optical modulators, meeting the demands of AI-driven data centers. The design and verification of the proposed driver circuit are conducted using open-source simulation software, eSim, with the Skywater 130 PDK. These methodologies and results aim to advance India's semiconductor mission (ISM) by accelerating efforts in indigenous chip design.**

Keywords—PAM-4, Mach-Zehnder Modulator, eSim, Skywater 130, CMOS

I. Introduction

High-speed driver circuits are essential for achieving efficient switching in silicon photonics-based optical modulators. As data communication demands surge, increasing spectral efficiency and enabling high data rate modulation on a single optical carrier have become essential to support next-generation AI-driven data centers and optical computing. Advanced modulation formats, such as Quadrature Amplitude Modulation (QAM), Quadrature Phase Shift Keying (QPSK), and Pulse Amplitude Modulation (PAM), offer potential solutions to meet these demands. In this study, we propose a driver circuit that generates a four-level PAM (PAM-4) signal to drive a Mach-Zehnder Interferometer (MZI) optical modulator, leveraging open-source eSim and Skywater 130 PDK for design validation.

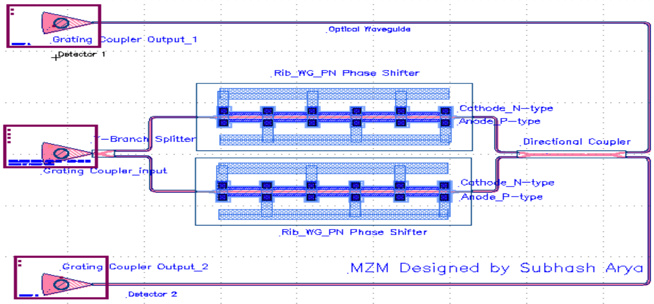


Fig. 1. GDS-II Layout of MZM using SIEPIC-pdk and tool

Figure 1 illustrates the GDS-II layout of the Mach-Zehnder Modulator (MZM), designed using the SiEPIC PDK and tools on the KLayout open-source platform. In this layout, the PAM-4 driver circuit is connected to the PN junction of the optical modulator. The four-level PAM-4 signal, generated from binary information bits, modulates the optical carrier at a wavelength of 1550 nm, selected due to its minimal optical loss, making it suitable for long-haul optical transmission. This design demonstrates the integration of multi-level signaling with silicon photonic modulators for efficient data transmission.

II. Modulator Driver Block Diagram

Figure 2 presents the block diagram of the proposed PAM-4 driver circuit designed to drive the Mach-Zehnder Modulator (MZM). The design employs a 2-bit synchronous counter that generates four binary states: 00, 01, 10, and 11. These states are converted into Gray code (00, 01, 11, 10) to ensure that only one-bit changes between adjacent states, thereby minimizing transition errors and enhancing signal stability.

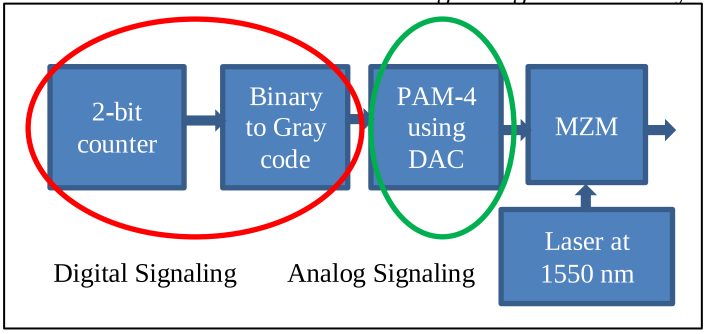


Fig. 2. Block diagram of designing PAM-4 driver

The Gray-coded output is then fed into a 2-bit Digital-to-Analog Converter (DAC) circuit, which produces a four-level PAM-4 signal. This signal drives the PN junction in the MZM, operating under reverse bias to achieve effective modulation of the 1550 nm optical carrier.

III. Simulation Results and Discussion

Digital Signaling
The digital signal part of the circuit consists of a 2-bit asynchronous counter (for simplicity). The asynchronous counter has been implemented using two JK-flip flops on the eSim schematic editor that has been shown in Figure 3.

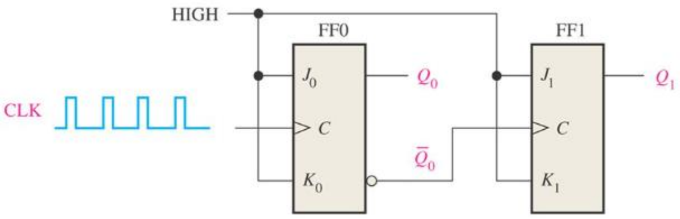


Fig. 3. Two bit Asynchronous Counter using JK Flip-flops (Theoretical)

This counter consists of two positive edge-triggered JK flip-flops, labeled FF0 and FF1, each with inputs J, K, and outputs Q and Qbar. The J and K inputs for both flip-flops are connected to a high logic level (logic "1"), placing each flip-flop in toggle mode. In this mode, the output Q toggles state with each positive clock edge. The primary clock signal is applied only to the first flip-flop, FF0. On every rising edge of this clock pulse, FF0 toggles its output, switching Q0 from low to high or high to low. The second flip-flop, FF1, does not receive the same clock pulse directly. Instead, its clock input is driven by the Q0bar output of FF0. This configuration is key to the asynchronous behaviour of the circuit. The use of Q0bar to clock FF1 ensures that FF1 toggles only when Q0 transitions from high to low, effectively introducing a delayed clock pulse to FF1. Thus, FF1 changes state only on the positive edge of Q0bar. Since FF0 toggles on each primary clock pulse and FF1 is triggered by FF0's complementary output, FF1 effectively operates at half the frequency of FF0. This behaviour achieves a divide-by-two frequency for each subsequent flip-flop, as each only toggles after its preceding stage completes a full cycle.

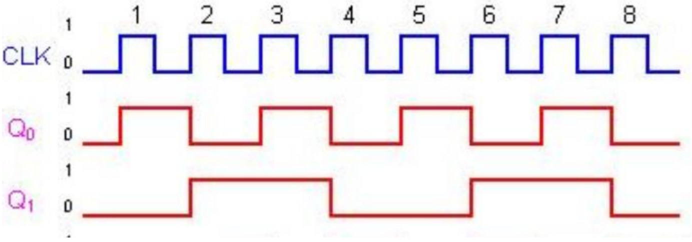


Fig.4. Input Clock pulse and outputs Q0 and Q1 as a timing diagram for the 2-bit asynchronous counter

Counting Sequence:

The two flip-flops produce a binary counting sequence in response to the primary clock. The states proceed as follows:
1.Clock Pulse 0: Q1Q0=00
2.Clock Pulse 1: Q1Q0=01 (FF0 toggles on rising edge of primary clock)
3.Clock Pulse 2: Q1Q0=10 (FF0 toggles; FF1 toggles due to positive edge of Q0bar)
4.Clock Pulse 3: Q1Q0=11 (FF0 toggles)
5.Clock Pulse 4: Q1Q0=00 (Counter resets to initial state)
This sequence repeats every four pulses, creating a "mod-4 counter."

Timing and Propagation Delay:

In an asynchronous counter, each flip-flop reacts to changes in its clock input after a short propagation delay. In this configuration, FF1 is not directly clocked by the primary clock but by the Q0bar output of FF0, which causes a ripple effect. The asynchronous nature introduces slight timing delays with each additional flip-flop stage, as the output of one stage must propagate to the next. This is acceptable for applications with moderate speed requirements but can result in accumulated delay for high-speed applications

2-bit Binary to 2-bit Gray Code Converter

The figure shows the 2-bit binary to gray code converter circuit. In this one wire connection for G1, as it's directly connected to B1. One XOR gate to calculate G0 by performing an XOR operation on B1 and B0

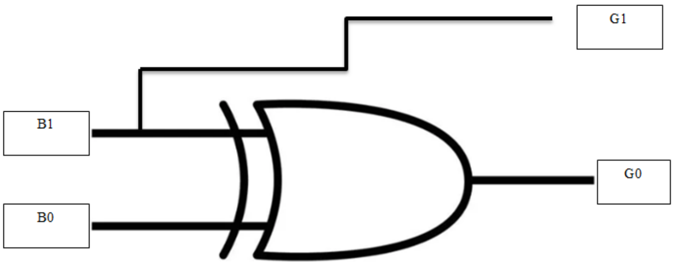


Fig. 5. A 2-bit binary to gray code converter circuit

eSim Platform Overview

eSim is an open-source Electronic Design Automation (EDA) tool developed by FOSSEE (Free/Libre and Open Source Software for Education) in collaboration with IIT Bombay. It provides an affordable and accessible alternative to proprietary tools for circuit design, simulation, and analysis. eSim combines KiCad (for schematic and PCB design) and Ngspice (for simulation), making it a comprehensive solution for electronics design. With eSim, users can create circuit schematics, perform simulations, and even design printed circuit boards (PCBs) without incurring licensing fees, which is highly beneficial for educational institutions and research labs.

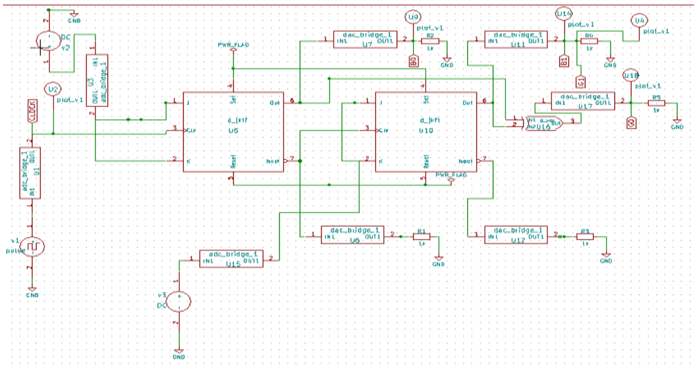


Fig. 6. A 2-bit Asynchronous Counter implemented using JK Flip-flops followed by the 2-bit Binary to Gray Code Circuit Schematic drawn on eSim

Circuit Description Using eSim

The schematic of the 2-bit asynchronous counter followed by a 2-bit binary to Gray code converter is designed and simulated using eSim, showcasing the tool's capability to handle both digital logic circuits and conversion designs. Clock Signal: In eSim, a clock signal can be generated within the schematic as a digital pulse source. This clock signal is fed into the first JK flip-flop, driving the toggling mechanism for the asynchronous counter.
2-bit Asynchronous Counter: The counter in this schematic is implemented using two JK flip-flops, each designed with eSim's component library: First JK Flip-Flop: Receives the clock signal directly. This flip-flop toggles its output (Q0) with every clock pulse, producing the least significant bit (LSB), referred to as B0.
Second JK Flip-Flop: Configured to toggle on the falling edge of the first flip-flop's output (Q0), providing the most significant bit (MSB) output B1. This setup forms an

asynchronous (or ripple) counter, where the MSB changes half as frequently as the LSB, following the binary sequence.
Binary Output: In eSim, the outputs B0 and B1 from the two flip-flops represent the 2-bit binary count (00, 01, 10, 11). eSim allows visualization of these signals in the waveform viewer, helping users verify the counter's behavior.

2-bit Binary to Gray Code Converter:

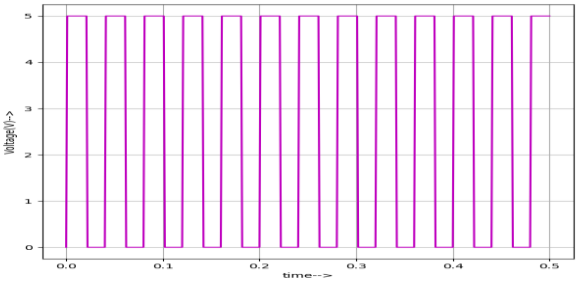
The binary counter outputs B0 and B1 are then connected to a simple binary-to-Gray code converter circuit:

Gray Code MSB (G1): This output directly mirrors B1, as G1 = B1.
Gray Code LSB (G0): This output is generated by applying an XOR operation between B0 and B1 using an XOR gate component from eSim's library. Thus, G0 = B0 XOR B1.
Gray Code Output: The resulting Gray code output, G1 G0, is generated based on the binary input and displayed in eSim's simulation viewer to confirm correct code conversion.

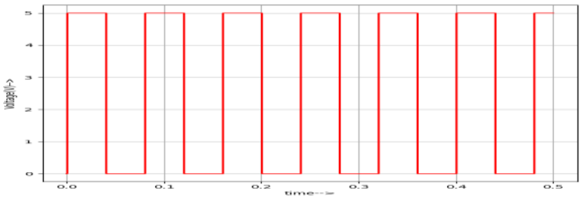
Benefits of Using eSim for This Circuit
Using eSim provides several advantages for this type of digital logic design:

Free and Open-Source: eSim is accessible to users without any cost, making it ideal for educational and research applications.
Integrated Simulation: By leveraging Ngspice within eSim, users can simulate the behavior of both analog and digital components, viewing waveforms to verify timing, transitions, and logic states.
Component Libraries: eSim includes an extensive library of components, such as JK flip-flops and logic gates, facilitating straightforward digital design.
Waveform Analysis: eSim's waveform viewer provides a clear visualization of signal transitions, which is essential for verifying the behavior of asynchronous counters and code converters.

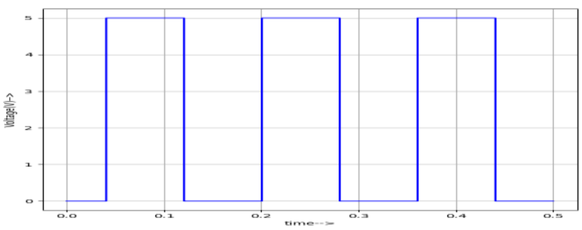
Timing Diagram of two bit asynchronous counter followed by the 2-bit binary to gray code converter timing signals.



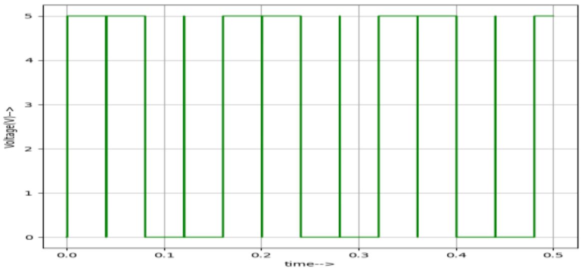
7 (a): Clock applied to first JK Flip-flop



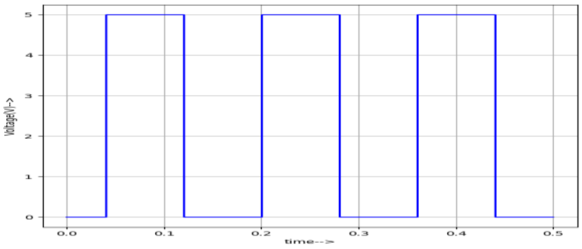
7 (b): LSB (Bo or Q0) is the output of 2-bit asynchronous counter from first JK Flip-flop



7. (c): MSB (B1or Q1) is the output of 2-bit asynchronous counter from second JK Flip-flop



7. (d): LSB (Go) is the output of 2-bit Binary to Gray Code Converter i.e. B0 XOR B1)



7. (e): MSB (G1) is the output of 2-bit Binary to Gray Code Converter i.e. G1 =B1)

The timing diagram shown in figures 7 (a) to 7(e) illustrate the operation of a two-bit asynchronous counter implemented with two JK flip-flops, followed by a 2-bit binary to Gray code converter.
Clock Signal (a): This is the input clock signal applied to the first JK flip-flop in the counter. It serves as the triggering signal that causes the JK flip-flops to toggle states and generate the binary count sequence. Binary Counter LSB (B0 or Q0) (b): This signal is the least significant bit (LSB) output of the 2-bit asynchronous counter, produced by the first JK flip-flop. It toggles on each clock pulse, representing the fast-changing bit in the binary counting sequence. Binary Counter MSB (B1 or Q1) (c): This signal is the most significant bit (MSB) output of the 2-bit asynchronous counter, generated by the second JK flip-flop. It toggles at half the frequency of B0, showing the slower-changing bit in the binary sequence. Gray Code Converter LSB (G0) (d): This is the least significant bit (LSB) output of the Gray code converter, obtained by performing an XOR operation on B0 and B1 (i.e., $G0=B0 \oplus B1$). This signal represents the LSB of the converted Gray code output, where each bit transition changes only one bit compared to the previous state. Gray Code Converter MSB (G1) (e): This is the most significant bit (MSB) output of the Gray code converter, which is simply equal to B1 (i.e., G1=B1). This signal represents the MSB of the

Gray code output, following the same toggling behaviour as B1 but now in the Gray code format.

Analog Signaling

In the circuit described, the Gray-coded output from the 2-bit binary to Gray code converter is fed into a 2-bit Digital-to-Analog Converter (DAC). The purpose of the DAC here is to convert the discrete Gray code levels (00, 01, 11, 10) into corresponding analog voltage levels. With a 2-bit DAC, four distinct voltage levels are generated, corresponding to the four Gray code inputs. Generating PAM-4 Signal: PAM-4 (Pulse Amplitude Modulation with 4 levels) is a signaling scheme that uses four distinct amplitude levels to represent data, allowing for 2 bits of information per symbol. The four-level output of the DAC creates a PAM-4 signal, where each level represents a unique combination of Gray code inputs, generating the distinct amplitudes required for PAM-4 transmission. This PAM-4 signal serves as the input to the MZI optical modulator driver. The PAM-4 signal's varying amplitude modulates the intensity of light in the MZI, effectively encoding the data for optical transmission.

Design implementation of 2-bit Digital to Analog Converter (DAC) circuit using Sky 130 pdk.

An n-bit Digital to Analog Converter (DAC) transforms an n-bit digital input into an analog voltage that is proportional to a reference voltage. The potentiometric DAC operates on the principle of a voltage divider. In an N-bit DAC, the analog voltage range, represented by the reference voltage (Vref, which is 3.3 V in this case), is divided into 2N2N discrete voltage levels. This division is accomplished using a series of 2N2N identical resistors, with taps placed across each resistor. The switches that select these tap points are configured based on the n-bit digital input where n=2 for present case.

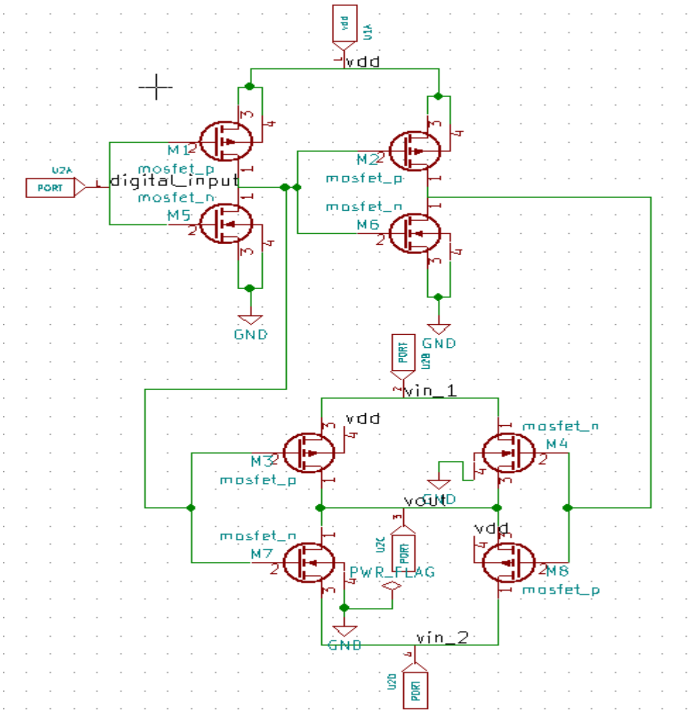


Fig. 8. An schematic of switch implemented on eSim using sky130 pdk.

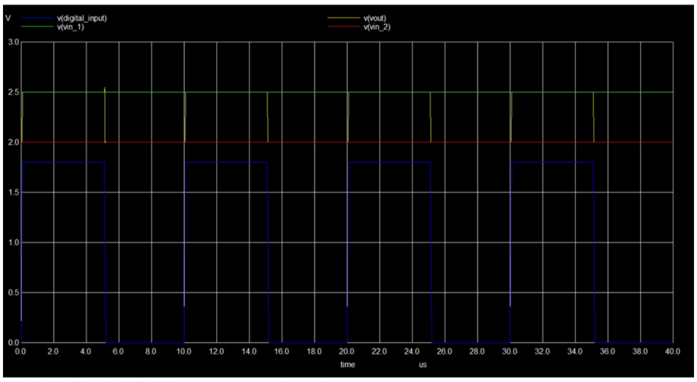


Fig. 9. Output of an electronic switch implemented on eSim using sky130 PDK, showing the relationship between the input bit stream, reference voltages, and the output voltage.

Explanation of Various Curves:

Blue Curve (v(digital_input)): This represents the digital input bit stream. The voltage alternates between low (0V) and high (approximately 2.5V) at regular intervals.

Green Curve (v(vin_1)): This represents one of the reference voltages, indicating the 'high' logic level. The voltage is constant at approximately 2.5V.

Red Curve (v(vin_2)): This represents the other reference voltage, indicating the 'low' logic level. The voltage is constant at approximately 2.0V.

Yellow Curve (v(vout)): This represents the output voltage of the switch circuit. When the input (blue curve) is low, the output (yellow curve) is high (approximately 2.5V). When the input is high, the output is low (approximately 2.0V).

Brief Description: The image shows the voltage waveforms of an electronic switch circuit simulated using eSim with the sky130 PDK. The digital input bit stream (blue curve) alternates between low and high states. The reference voltages (green and red curves) define the logic levels for 'high' and 'low'. The output voltage (yellow curve) of the switch circuit is high when the input is low and vice versa. This behavior is typical for an inverter or NOT gate, making it relevant for understanding basic digital logic circuits.

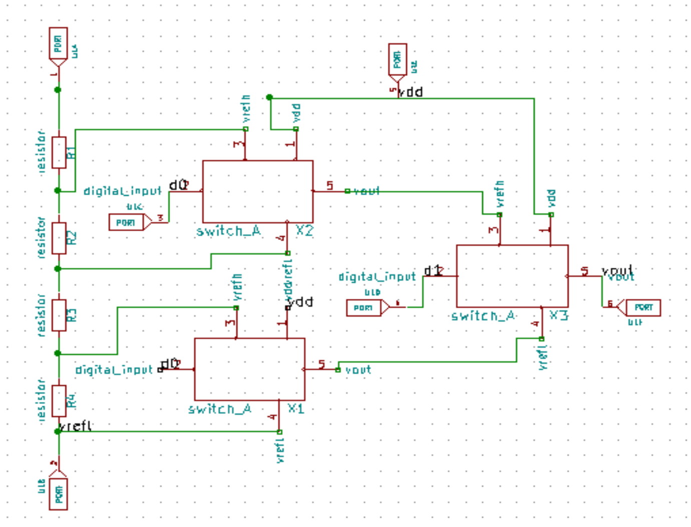


Fig. 10. Circuit schematic of 2-bit digital to analog converter using switch as a sub-circuit.

The figure 10 displays a schematic of a 2-bit Digital to Analog Converter (DAC) implemented using three switches, resistors, and digital input ports, designed with the sky130 PDK on the eSim platform. In this circuit, the two digital input bits, d0 and d1, control the switches, which connect to either Vdd (high voltage) or ground (low voltage). The switches are arranged in a ladder configuration, creating a voltage divider network. Depending on the digital input values (00, 01, 10, 11), the switches operate to connect the resistors to Vdd or ground. This arrangement of resistors causes different voltage levels at the output based on the combination of input bits, resulting in an analog output voltage (vout) that is a weighted sum of the digital inputs. This 2-bit DAC demonstrates how digital signals are converted into analog form, which is crucial for applications in digital signal processing and communication systems.

Figutre. 11 shows the expected waveform output. The non-return-to-zero (NRZ) signal will be generated using a counter, and then binary to Gray code will assign the reference voltages to DAC to give PAM-4 signal.

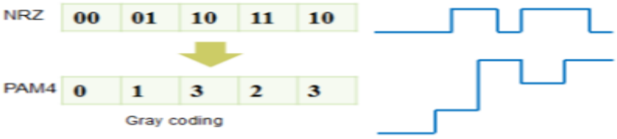


Fig. 11. The PAM-4 signal as per NRZ and Gray code (left its weight and right its waveshape at 4 voltage levels) [2]

Acknowledgement

I express my gratidute to Mr. Kunal Ghosh and Mr. Sameer S. Durgoji for their kind support for completion of this work. I also express my thanks to Mr. Sumanto Kar for extending his prompt support for troubleshooting as and when it was essential.

References

[1] R. Malngiang and S. C. Arya, "Mathematical Modeling of Carrier Depletion based P-N junction Optical Phase Shifter using Evolutionary Algorithm," 2019 2nd International Conference on Innovations in Electronics, Signal Processing and Communication (IESC), 2019, pp. 305-310, doi: 10.1109/IESPC.2019.8902382.

[2] N. Dikhaminjia et al., "High-speed serial link challenge using multi-level signaling," 2015 IEEE 24th Electrical Performance of Electronic Packaging and Systems (EPEPS), 2015, pp. 57-60, doi: 10.1109/EPEPS.2015.7347129.