A Self-Optimizing 4-Channel 30 Gbaud/s PAM-4 Packaged Silicon Photonics Subsystem with Binary Driving Signals

Nathan C. Abrams[1], David M. Calhoun[1], Christine P. Chen[1], Keren Bergman[1]

[1]Dept. of Electrical Engineering, Columbia University, New York, NY 10027 nca2123@columbia.edu

Abstract *We demonstrate four channel PAM-4 modulation in a silicon-photonic platform driven by push-pull binary electrical signals. The subsystem self-optimizes through a gradient descent algorithm across the full range of starting voltages.*

Introduction

Silicon photonic (SiP) devices offer the potential for improving traditional electronic-interconnected systems in terms of energy consumption, footprint, and bandwidth[1,2]. The bandwidth density of a silicon photonic system can further be improved by employing advanced modulation formats, such as quadrature amplitude modulation (QAM), quadrature phase shift keying (QPSK), and pulse amplitude modulation (PAM). PAM-4 has been demonstrated in silicon modulators, and is an attractive modulation format due to a 2x bandwidth density advantage compared to on-off keying (OOK) while still allowing for direct detection[2]. Optical PAM-4 can be generated by driving a device with a 4-level electrical pattern generator, or by driving in push-pull mode with two binary electrical patterns[3].

A significant concern with any modulation format is actively tuning and maintaining optimal signal quality. Systems that can automatically self-optimize signal quality offer the benefit of ease-of-use and repeatability, with the potential for scaled integration into larger systems. Previous work has explored optimizing OOK modulation with bit-error-rate (BER) monitoring[4], but the complexity of self-tuning increases as the modulation formats become more advanced.

This work investigates the increased bandwidth of wavelength division multiplexing (WDM) with PAM-4. The result is a subsystem that achieves binary drive PAM-4 and self-optimizes using a gradient descent algorithm executed on an external, software-programmable system.

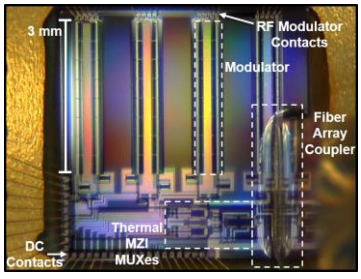

Figure 1: Image of wirebonded four channel MZM transmitter device

Device

The device consists of four concurrent Mach-Zehnder modulators (MZM), spatially-multiplexed using two stages of Mach-Zehnder interferometers (MZI). Each arm of the four MZMs contains a pn junction operated in depletion mode with fully independent differential drive and a thermal control. The device is electrically packaged in a high-speed chip carrier with gold wire bonds and optically packaged using a 5-port optical grating coupler epoxy-bonded at surface normal. The packaged device is mounted on a printed circuit board with connected breakouts for control of both high-speed and DC-biased electrical contacts.

Experimental setup

Fig. 2 depicts the experimental setup. Two tunable lasers are polarized and injected into two MZMs. The WDM output is amplified using an erbium-doped fiber amplifier (EDFA) followed by an optical grating filter, and finally sampled on a digital serial analyzer (DSA).

Two electrical OOK data streams are driven on both arms of a single MZM using a PPG with 2^{31}-1 data, and are DC-biased using DC voltage supplies. The signals are phase-aligned using electrical delays to ensure push-pull operation, and are driven with different voltage amplitudes to create the PAM-4 modulation. One driving voltage is delayed with respect to the other to ensure the two arms are not correlated. Static voltages for the MZI and MZM bias heaters are

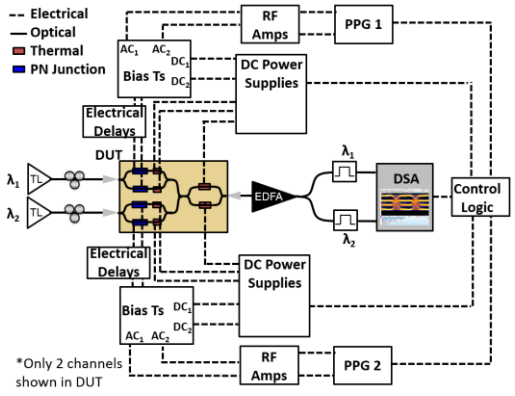

Figure 2: Experimental layout depicting the control subsystem for two channels of the device

```
i = 0
while (i < number_iterations) do
    initialize_probe_directions()
    apply_volt(v_top[i], v_bot[i])
    hist_data = histogram(trace_data)
    peaks = find_peaks(hist_data)
    obj[i] = calculate_obj(peaks)
    apply_volt(v_top[i] + v_probe, v_bot[i] + v_probe)
    hist_data = histogram(trace_data)
    peaks = find_peaks(hist_data)
    obj_temp = calculate_obj(peaks)
    v_top[i+1], v_bot[i+1] = update_volts(v_top[i], v_bot[i], obj[i], obj_temp)
    i = i + 1
```

Figure 3: Simplified psuedo code

provided via DC voltage supplies. A control layer is implemented using GPIB interfaces and a Python script orchestrates the control of the PPGs, DSA, and DC voltage sources.

Gradient descent optimization algorithm

The self-optimizing algorithm is based on a two dimensional gradient descent using real-time data from the DSA. The data is processed by applying a scaling metric, after which the control logic separates and histograms 8000 data points into 50 bins. The PAM-4 signal levels should correspond to the four largest peaks in the histogram. An objective function is calculated according to the following:

$$\Theta = \frac{1}{(L_4 - L_3)^{1.5}} + \frac{1}{(L_3 - L_2)^{1.3}} + \frac{1}{(L_2 - L_1)^{1.0}} \quad (1)$$

The various levels of the PAM-4 signal in terms of optical power are represented by the variable L, with L_4 corresponding to the level with the largest optical power. These exponent values produced the most symmetrical levels across several heuristic trials. Bias voltages with PAM-4 levels that are close together will result in a relatively large objective function. This approach allows us to evenly space the PAM-4 levels as we minimize the objective function.

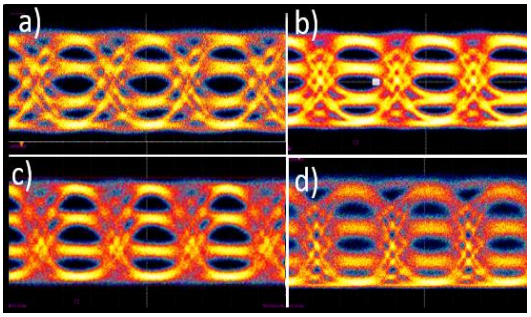

Figure 4: PAM-4 with each of the four modulators at 7.5 Gbaud/s. a) Modulator 1 b) Modulator 2 c) Modulator 3 d) Modulator 4

Each iteration of the gradient descent involves multiple steps, depicted in Fig. 3. First, the updated voltage is applied and the objective function is calculated. Then, the voltage is probed in one dimension, which means that either the top or bottom pn junction is either increased or decreased by 0.1 volts. The objective function is again calculated, and the driving voltage is adjusted as shown in Eq. (2). The process is repeated for the other pn junction. For each iteration, the order of the modulators and whether the probe voltage is positive or negative is randomly selected. This randomization allows the objective function to decrease faster, as the algorithm appeared more effective when it was probing in the direction of descending gradient.

$$V = V - \eta \nabla \Theta(V) \quad (2)$$

The voltages for the next iteration are calculated by subtracting the gradient, scaled by a step size η, from the current voltages. The step size was ±0.05, with sign determined by the probing voltage polarity. The gradient descent converged faster when each voltage dimension was probed and updated separately.

If the control system is unable to find four prominent peaks, then this gradient descent

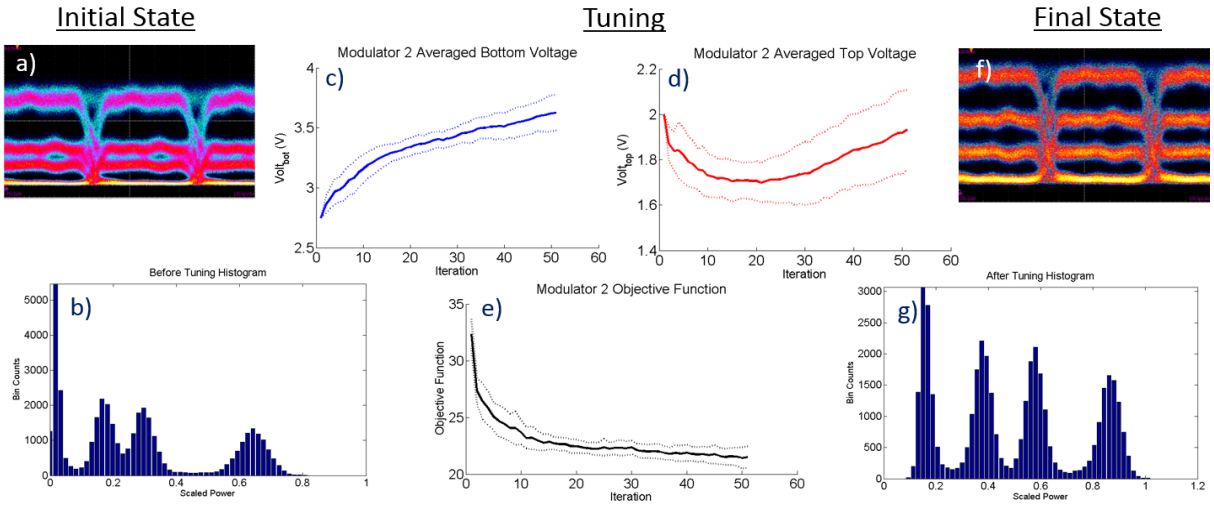

Figure 5: 20 repeated trials for suboptimal starting position tuning for modulator 2. a) The waveform before tuning b) A histogram before tuning c) The averaged bottom pn voltage with standard deviation d) The averaged top pn voltage with standard deviation e) The averaged objective function with standard deviation f) The waveform after tuning g) A histogram after tuning

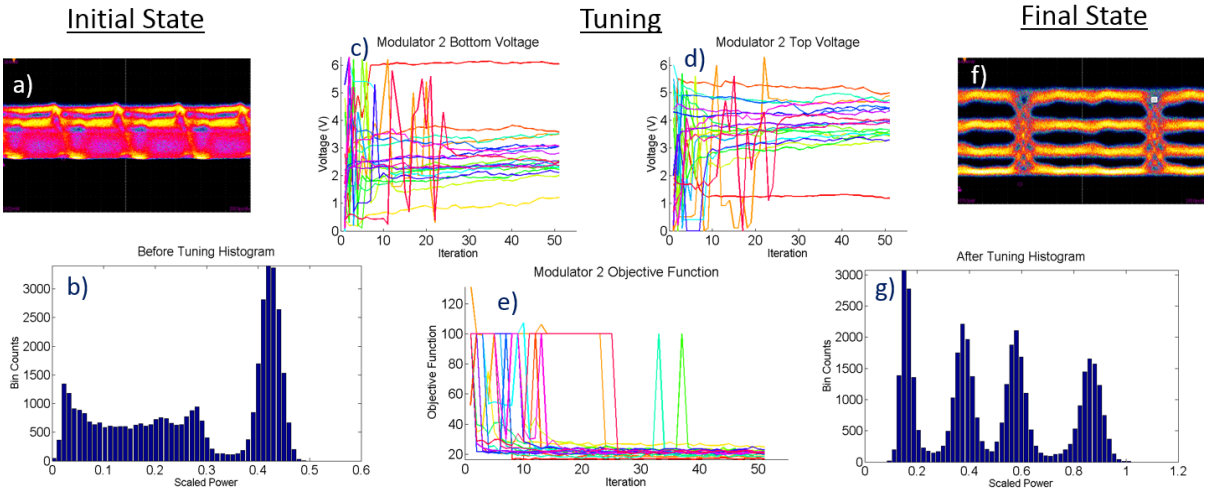

Figure 6: 20 repeated trials for unidentifiable starting position tuning for modulator 2. a) A waveform before tuning b) A histogram before tuning c) The bottom pn voltage d) The top pn voltage e) The objective function f) The waveform after tuning g) A histogram after tuning

cannot be performed. In the case where four prominent peaks are not found in one section of an iteration, the objective function is (symbolically) set to 100, and the voltage settings are restored to the previous state with four sufficient peaks. In the case where four prominent peaks are not found across all the steps of an iteration, the voltages are randomized to move to a different location in the voltage space on the next iteration.

Optimization data

The system will encounter one of two initial conditions: a signal with four prominent peaks (sub-optimal), or one without four prominent peaks (unidentifiable).The data for a sub-optimal starting position can be found in Fig. 5. The gradient descent algorithm runs for 50 iterations for each of the 20 trials. The objective function decreases sharply in the first 10 iterations. The standard deviation for both the final voltages is approximately 0.2 volts. The data from an unidentifiable starting position is shown in Fig. 6. By about iteration 15, nearly all of the 20 trials identify a voltage combination with four distinct peaks, and the objective function can be appropriately optimized. The two later spikes were cases when the algorithm was unable to identify four prominent peaks, and jumped to an objective function of 100 before restoring to the previous voltage state.

Simultaneous tuning of multiple modulators is less reliable than tuning a single modulator. With dual tuning, the heat from thermal tuning on one modulator will affect the thermal tuning of a different modulator. Additionally, the EDFA in the setup results in output power of the two wavelengths proportional to the input power of the two wavelengths. As a result, if tuning on one modulator results in a change in average power, then the ratio of the power of the two wavelengths will change, and so the output power of the non-adjusted channel will also be affected.

The tuning for these cases was done at 2 Gbaud/s. The voltage levels were still optimized when the frequency was increased to 7.5 Gbaud/s. The optimized PAM-4 modulation for all four channels is shown in Fig. 4. Error free operation (BER < 10^{-12}) with OOK was achieved on all channels with simultaneous modulation.

Conclusions

We demonstrate four channel PAM-4 modulation driven by binary electrical signals in a push-pull manner. A WDM signal is transmitted through a SiP transmitter chip that self-tunes to an optimal bias point via the gradient descent algorithm. This process allows for improved modulation along each data channel, a proof-of-concept that greatly optimizes signal recovery of higher-order data modulation formats.

Acknowledgements

The work of the authors is supported by the DoD NDSEG Program, NSF IGERT DGE-1069240 and the Intel/SRC PhD Fellowship. The authors thank G. Pomrenke, of AFOSR, for his support through FA9550-13-1-434 0027, FA9550-14-1-0198, FA9550-12-C-0079 and funding for OpSIS (FA9550-10-1-0439). The authors acknowledge Y. Liu and M. Hochberg from Coriant ATG for support of OpSIS devices.

References

[1] D. M. Calhoun et al., “Hardware-Software Integrated Silicon Photonics for Computing Systems,” Silicon Photonics III, 122, p. 157-189 (2016).

[2] C. Kachris. K. Bergman, I. Tomkos, “Optical Interconnects for Future Data Center Networks,” Springer Science & Business Media (2012)

[3] P. J. Winzer et al., “Advanced Optical Modulation Formats,” Proc. of the IEEE, 94(5), p. 952-985 (2001).

[4] A. Samani et al., “A Silicon Photonic PAM-4 Modulator Based on Dual-Parallel Mach-Zehnder Interferometers,” IEEE Photonics Journal, 8(1) (2016).

[5] W. A. Zortman et al., “Bit-Error Rate Monitoring for Active Wavelength Control of Resonant Modulators” IEEE Micro, 33(1), p. 42-52 (2012).