SOI Photonic Electro-Optic Modulators: A Brief Study on Design, Fabrication and Measurements of Photonic EOMs

Neal Elharidy - 99272304

Department of Biomedical Engineering
The University of British Columbia - Vancouver, Canada

Abstract - This study intends on being an exercise in learning, and an investigation of the fundamentals of photonic interferometers, with a focus mainly on MZI based designs.

Two chips were designed and fabricated, a primary Electro-Optic tuneable interferometer fabricated at UBC, and a secondary simple MZI fabricated at Applied Nanotools Inc.

This report aims to document and analyze the two chip's design process, device parameters, fabrication processes, characterization of devices and finally a comparison between the fabricated devices and their initial simulations.

I. Introduction

Most recognize that traditional CMOS chips have reached or are rapidly reaching their physical and technological limits in terms of miniaturization and power efficiency leading to the exploration of alternative technologies. One such promising technology is Photonic Integrated Circuits (PICs).

Photonic Integrated Circuits are microchips that use light (photons) instead of electrons to perform a variety of functions. These functions can include generating, detecting, processing, and transporting light signals [1].

In the realm of high-speed, high-throughput data applications, silicon phonetics emerges as a game changer, overcoming the limitations faced by traditional CMOS technologies. Devices like Photonic Artificial Intelligence Chips (PAIC) utilize light beam propagation in silicon waveguides, embodying traits such as high parallelism configuration, swift calculation speed, and minimal latency [2].

Although PICs can a have a vast network of devices in order to create complex chips such as PAICs, in this study however we will explore imbalanced Mach-Zehnder Interferometers (MZIs) in order to grasp the fundamentals of PICs.

We created two variants that were separated into two chips. A primary chip with Electro-Optic Modulators (EOMs), and a secondary chip featuring non-modulated MZIs.

This study will primarily focus on the primary chip with EOMs. We aim to compare the projected efficiency from our simulations against the experimental data. In doing so, we hope to address any unexpected results and shortcomings in the design.

Furthermore, we will explore potential improvements that could be made to the design parameters of the EOMs. This could provide valuable insights for future iterations and enhancements of the chip design.

II. Theory & Modelling

A. Imbalanced Mach-Zehnder Interferometers

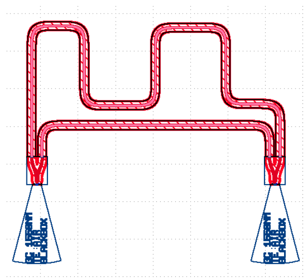


***Figure 1 -** Simple Imbalanced MZI*

Imbalanced MZIs are simple devices as seen above in Figure 1. Light is guided through a grating coupler and then split into two beams of equal intensity by a Y-splitter. The light then travel through identical waveguides of non-equal length. As the light travels through the unequal waveguides it is recombined at a Y-Combiner. Since the light travels unequal distances one branch will be phase shifted relative to the other. The degree of shift determines whether the light combines constructively or destructively. This phenomena is described in the MZI transfer function as seen below [1]:

$$\frac{I_o}{I_i} = \frac{1}{2}[1 + cos(\beta \Delta L)] \qquad (1)$$

$$\beta = \frac{2\pi n_{eff}}{\lambda} \qquad (2)$$

Where ΔL is the difference in length between the two branches of the MZI, and β is the propagation constant of light. n_{eff} is the effective index of the waveguide, and λ is the wavelength of the light which in this study was set to 1550 nm.

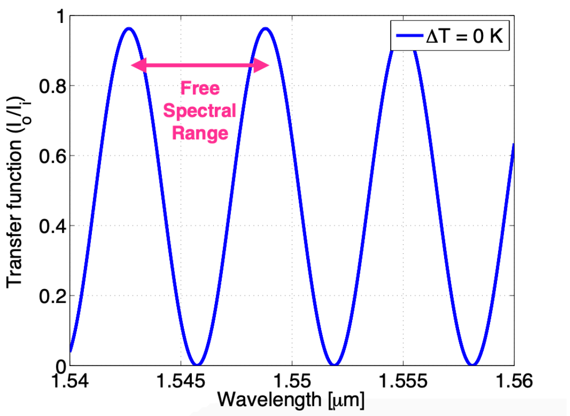


Figure 2 - *Optical Transmission Spectrum of MZI [1]*

As seen above if the transfer function of an MZI is graphed against the wavelength then we obtain an optical transmission spectrum [1]. The range between two peaks on the optical transmission spectrum is defined as the free spectral range (FSR) which is calculated with the following equations [1]:

$$FSR = \Delta\lambda = \frac{\lambda^2}{\Delta L n_g} \qquad (3)$$

$$n_g = n_{eff} - \lambda \frac{dn_{eff}}{d\lambda} \qquad (4)$$

Where n_g is the group index of the waveguides.

B. Electro-Optic Polymer Modulation (EOP)

EOP modulation works via the Pockels electro-optic effect, a phenomena where the refractive index of optical media changes in response to the application of a transverse electric field [3]. This can be seen in Equation 5 below that defining modulation sensitivity (the effective mode index change versus applied voltage) [3]:

$$S_p = \frac{\delta n_{eff}}{\delta V_{in}} = \frac{1}{2} n^3 r_{33} \frac{\Gamma}{d} \qquad (5)$$

Where n is the EOP material refractive index inside the slot, r_{33} is the electro-optic tensor coefficient of the material, Γ is the field overlap integral between electrical and optical fields , and d is the separation between the two electrodes through which the modulating signal is applied [3].

Poling the EOP material (HLD) is required to align the chromophore molecules towards the electric field as seen below in figure 3. This is done to yield a high r_{33} value since the tensor coefficient is approximately linear to the poling field applied as seen in figure 4. (Poling curve in appendix.)

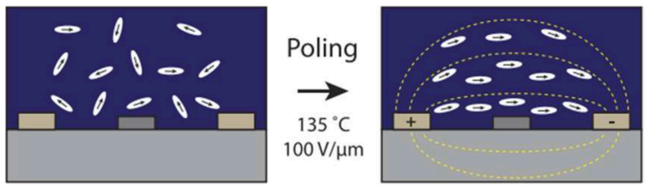


Figure 3 - *Poling of chromophores in*

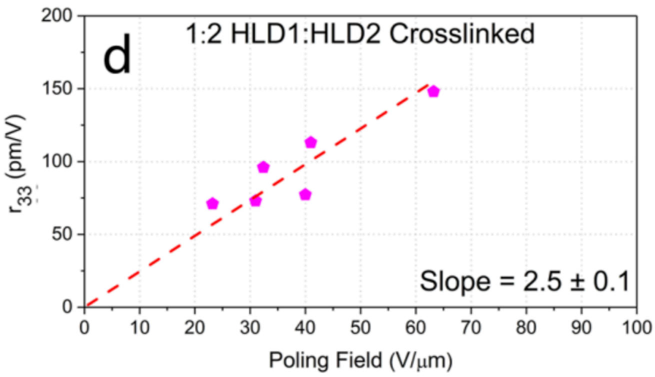


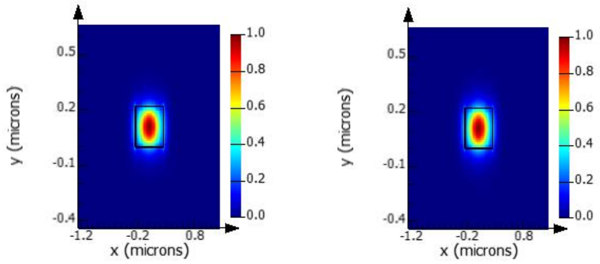
Figure 4 - *R33 vs Poling field applied to HLD*

The modulation sensitivity can then be used to find a measure of efficiency for modulators via Equation 6 as seen below [3]:

$$V_\pi L = \frac{\lambda}{2S_p} \qquad (6)$$

Where V_π is the required voltage to introduce a π phase shift, L is the RF electrode length, λ is the wavelength of operation, and finally S_p is the aforementioned modulation sensitivity.

C. Simulation Results



(A) ***(B)***

Figure 5 - Mode profiles for air cladding (A) and HLD cladding (B)

As seen above simulations were performed using Lumerical for air cladding which was given an index of 1, and HLD1/HLD2 cladding which was given an index of 1.6.

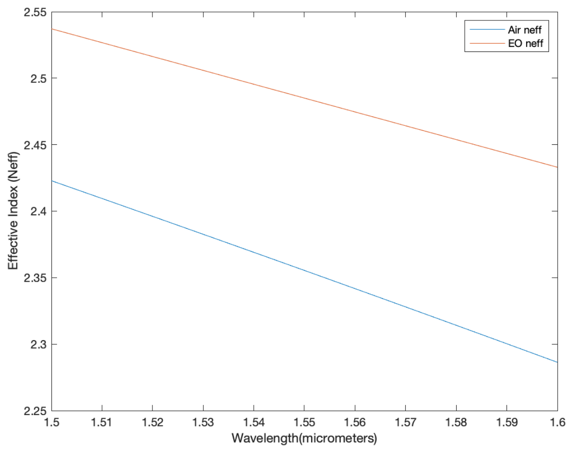


Figure 6 - *Effective Index vs Wavelength for air and EOP.*

As seen above a sweep was also performed from 1500nm to 1600nm for both air and HLD cladding, and the results effective indices were shown above in figure 6 and group indices below in figure 7.

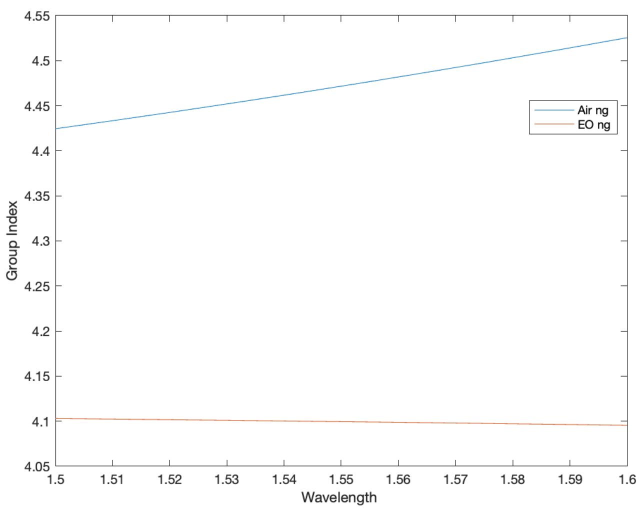


Figure 7 - *Group Index vs Wavelength for air and EOP*

III. Design

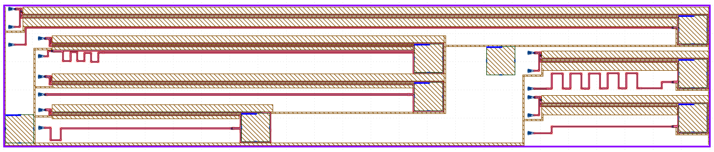


Figure 8- *Top view of chip design*

A. Design Objective

The main figure of merit of the primary chip design is reducing V_π as to increase the efficiency of the modulator as much as possible. Rearranging Equation 6 for easier comprehension of variables of interest yields:

$$V_\pi = \frac{\lambda d}{n^3 r_{33} \Gamma L} \qquad (7)$$

Thus it was concluded that minimizing V_π would require minimizing d (the separation between the two electrodes) , and maximizing L (the electrodes length).

B. Design Methodology & Experiment Design

Following identification of the design objectives, it was decided that the distance between electrodes would be varied amongst 3 groups (30, 28, & 27 μm) and the length of the electrodes would be varied amongst 4 groups (960, 1325, 2556, & 4637 μm). At first these variations would have resulted in 12 unique devices, however due to the size restriction of the chip's floor plan 6 devices were modelled with multiple variations added to specific devices, and redundancies added incase of manufacturing defects. After the designs were complete they were supposed to be tested with K-layout and Lumerical Mode, unfortunately due to issue with licensing my Lumerical refuse to run any longer, my issue was posted on piazza but no response was given by the instructor on how to fix the licence error.

MZI #	d (μm)	L (μm)
1	30	4637
2	28	2556
3	28	2556
4	30	960

MZI #	d (μm)	L (μm)
5	30	960
6	27	1325

Table 1 - Variations of MZI Within Primary Chip

The main hypothesis was that the larger the variable L and smaller the variable d the lower the V_π and thus the higher the efficiency of the MZI.

Therefore the first MZI aimed to obtain the maximum length allowable within the floor plan's dimensions, and then MZI 2 & 3 were made to be around half that length, as a large reduction was assumed to have drastic changes to the overall efficiency of the MZI. MZI 6 was then around half the length of MZIs 2 &3 to gather how halving the length twice would cause the efficiency to decrease and whether the decrease would be perfectly linear or otherwise.

An error on my part in design is not varying the distance between electrodes between copies of the same MZI, it would have been wiser to differentiate the d variable within each group in order to isolate the variable and analyze its independent effect. This would be one of my main recommendations for future endeavours.

C. Mask Layout

As seen on Figure 8, the floor plan for the chip was 5000 μm x 1000 μm and fit 6 variations of MZIs. MZI 1, 2, & 3 are all three grating couplers MZI devices while 4, 5, & 6 are all two grating couplers MZI devices. This was done to allow more MZI devices to fit within the floor plan.

The chip also required connecting the electric rails on the chip's exterior with the electrodes of the EOP Modulators. This was done by connecting the left and right hand side borders of the floor plan, with two main rails that branch to form the necessary connections. Both main rails have test pads directly attached to them or in direct contact with the rail. All the test pads on the chip were made to be 200 μm x 200 μm

The metal electrodes flanking the waveguides were designed to extend as far as possible with the waveguide while minimizing the length of unmodulated sections of the waveguide in order to minimize the aforementioned modulator efficiency figure of merit V_π.

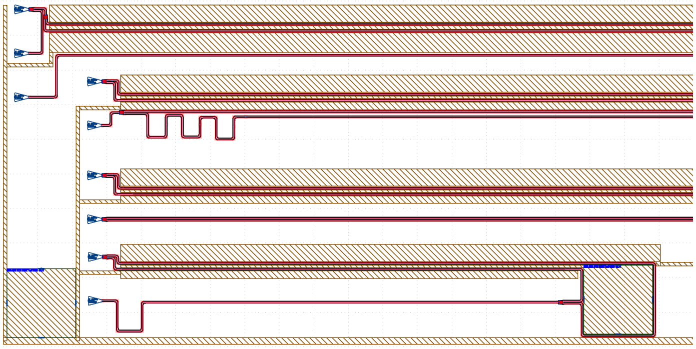


Figure 9- *Left Metal Rail Network*

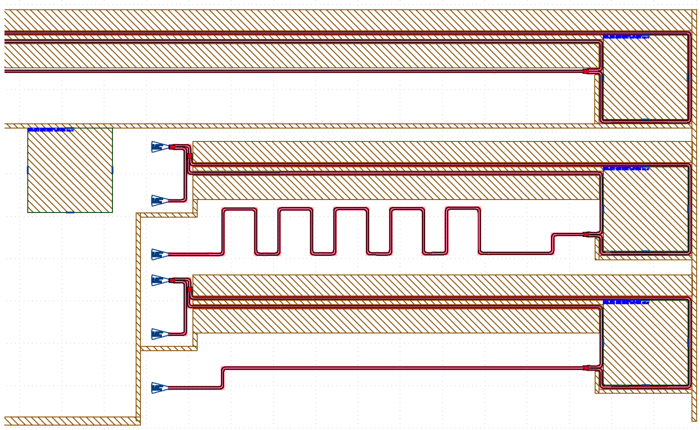


Figure 10 - *Right Metal Rail Newtork*

Other than the electric connections requirements, the manufacturing and testing process necessitated that the grating couplers be 127 μm apart and facing the right, as that is the configuration of the stage and fibre array automated tester used at the Brimacombe facilities.

IV. Experiments

A. Fabrication

As previously mentioned the primary chip with EOP Modulated MZI were fabricated at UBC's Advanced Materials & Process Engineering Laboratory Brimacombe facilities by a talented group of individuals including but not limited to Sheri Jahan and Iman Taghavi, This section will follow in great detail the steps taken by the team to fabricate the chip.

Starting with a 220nm thick Silicon on Insulator 8-inch wafer with crystal Miller Indices of 1-0-0 (90°) which can be easily cleaved using a diamond pen due to orientation of the crystal planes. The wafer was cleaved into 15 x 15 mm chips with a diamond pen as seen in process below in Figure 11.

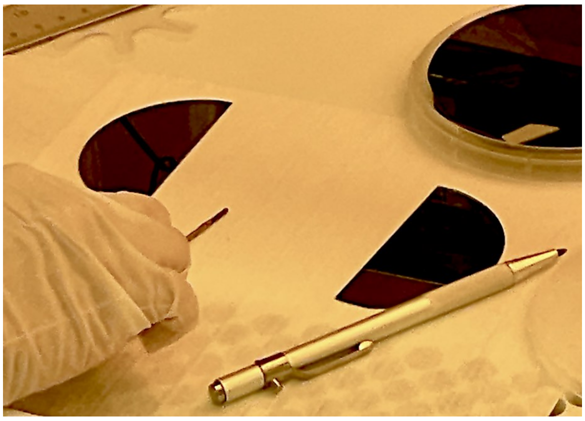


Figure 11 - Cleaving Wafer into chips

The chips are then cleaned to remove any residues on the surfaces. This was done by placing the chips in acetone, and then ultrasonically cleaned for 2 mins in water then 2 mins in IPA. The chips were then dried with a Nitrogen gas. The chips are then placed in an oxygen plasma chamber for 30 secs with 20 sccm of oxygen plasma flow and chamber pressure of ~288mT. The chips were then dried by baking at 180°C for 10 mins. This process not only ensures that no water residue remains on the chip but also that the chip has a slightly hydrophilic top layer due to the O2 plasma exposure which helps the photoresist spread evenly on the chip.

A positive resist (ZEP 520 A7) was then deposited ensuring no bubbles form on the surface. The resist was then spread for 3 secs at 500 rpm with a 200 rpm/s ramp, and then spun onto the chip surfaces at 2000 rpm ramped at 1000 rpm/s for 35 secs. This process yielded a measured 260 nm photoresist layer on the chip. The chip is then baked at 180°C for 60 secs to evaporate most of the solvent in the photoresist.

The chips are then patterned using Electron Beam Lithography. First the chips are loaded into a holding chamber where a vacuum is pulled to match the vacuum of the main E-Beam column. The chips are then loaded into the main chamber where they were exposed at 100keV, with a shot pitch of 4nm with a base dose of 213 $\mu C/cm$. The E-Beam lithographer adjusts for both electron repulsion scattering (forward scattering), and backward scattering from reflected electrons in the wafer. This allows the E-beam to actively expect and correct for the scattering allowing it to be extremely accurate in its exposure.

After the E-Beam Lithography ended, the resist was then stripped using ZED N50 for 60 secs. Then the chips are rinsed in isopropanol, and dried using Nitrogen gas.

The chips then needed etching, and they were sent to be plasma etched. First the chips were placed on the etcher's carrier plate ensuring there is no gap between the chip and the plate. The plate and the chip were then loaded into the plasma etcher. After the chip was etched we removed the plate and separate the chip from it. The chip was then cleaned of residue in ethyl acetate.

The chip was then place in a UV bath for 10 mins to denature the links in the photoresist. The chip was then submerged in photoresist remover for 15 mins at 80°C and then allowed to cool for 5 mins. This was followed with an IPA rinse and drying with N_2 Gas. The chips were then inspected post etch for defects and approved for the next round of lithography.

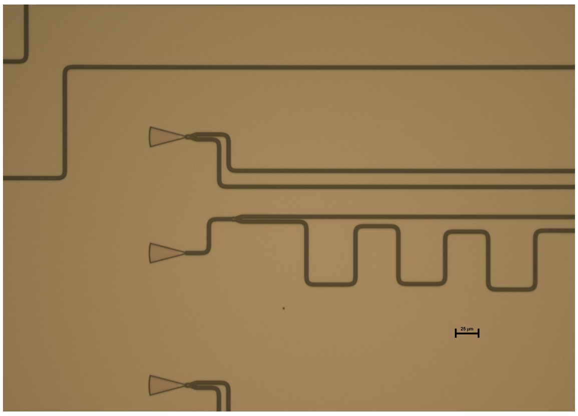


Figure 12 - *Post Etch Primary Chip*

Post inspection the chip was cleaned in IPA and acetone for a 60s each respectively. The chip was dried with N_2 Gas. The chip is then placed in the oxygen plasma chamber for 30 secs as we need to reduce the surfaces hydrophobic properties left over from solvent residue, it should be noted that if the chip is placed for longer the surface will become too hydrophilic and cause undercuts in the final result. The chips were then pre baked for lithography at 110°C for 5 mins.

The second round of photolithography uses a double exposure technique that essentially turns a positive photoresist into a negative photoresist. This was done via photolithography rather than E-Beam as the higher resolution is not needed for the relatively larger metal segments in the primary chip. AZ5214 is is first spun at 4000 rpm with a 1000 rpm/s ramp for 40 secs. The chip is then baked at 90°C for 60 secs, and then exposed with a 375 nm wavelength laser. The chip was then baked again at 110°C for 30 secs, and then re-exposed at 375 nm again making the final negative image. The chips are then rinsed for 40secs with Azmif 300 and then water for 40 secs.

The chips then are set for metal evaporation deposition, where first a 5nm titanium layer is deposited to help with the adhesion of the second layer of 100nm of gold, as gold doesn't stick well to Silicon. The chips are then placed in a ultrasound heated bath of aggressive solvent for cleaning.

The chips were then coated with electro-optical polymer NLM HLD according to the manufacturers instructions as seen on Reference 4. The chips are first cleaned with a N_2 gas, acetone, IPA, Di water, and oxygen plasma. Then a 2:1 ratio of HLD2 : HLD1 was dissolved into 1,1,2 - trichloroethane to make a 9% by weight solution of chromophore in the solvent. The solution was then placed in a sonicator for 15-30 mins to mix thoroughly. The polymer was then spin coated onto the chip at 500 RPM for 5 secs, then 850 RPM for 30 secs, and finally 1500 RPM for 30 secs. The devices were then poled and thermally cross linked according to the manufacturers instructions as written in the appendix below. The chips were then cleaned of excess polymer with acetone, and then the metal pads wire bonded to the pads.

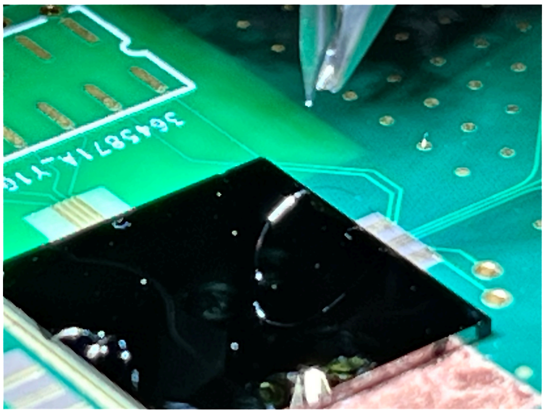


Figure 13 - *Wire bonded chip to PCB*

B. Testing Methodology and Results

For testing the devices were inspected via laser/photonic stage and fibre array system. The fibre array was originally angled at 8° however the compounded loss in signal post EOP coating and metal deposition required an adjustment of the angle of approach to ~20° during active testing. Unfortunately, my device was not successfully measured during active testing, thus the active data was taken from a colleague's device "timbitss_polymod2" [5]. The data obtained was for $\pm 0 - 10V$ modulation voltages as seen below in figures 14 and 15.

Measurements of my devices prior to metallization was successful and in the Appendix there is a graph of MZI 3 functioning as passive device. And further discussion and graphs exploring my design.

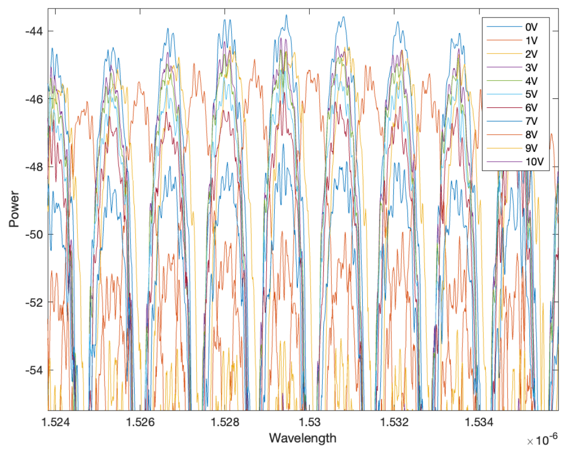


Figure 14 - *Power vs Wavelength w/ Positive Voltage applied*

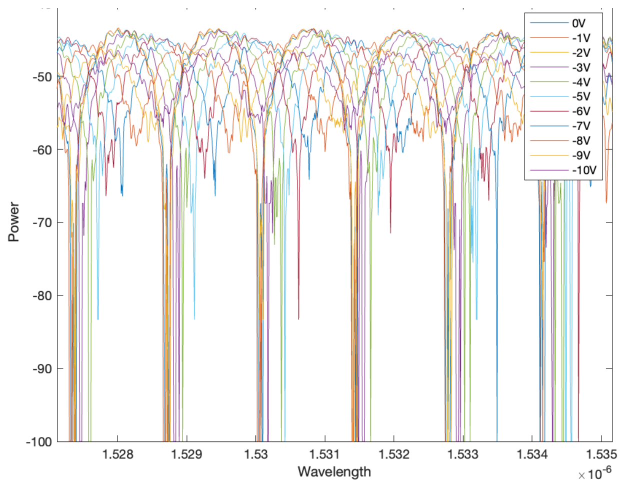


Figure 15 - *Power vs Wavelength w/ negative voltage*

V. Analysis

The negative voltage graphs were selected to calculate the FSR as the positive modulation voltage data had no tangible or consistent shift as seen in figure-14 and hence was unusable.

After smoothing and identifying the peaks, the average FSR was found to be 1.3568 nm. The phase shift was plotted at every voltage from -2 to -10V, and a best fit line was used to determine V_π which was found to be 6.569V.

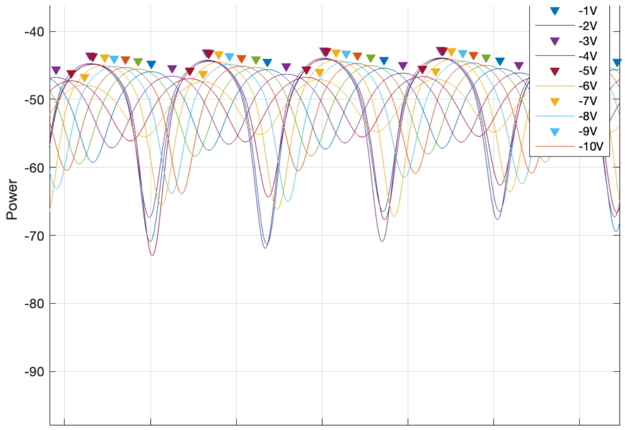


Figure 16 - *Smoothed and Peak identified Power vs wavelength*

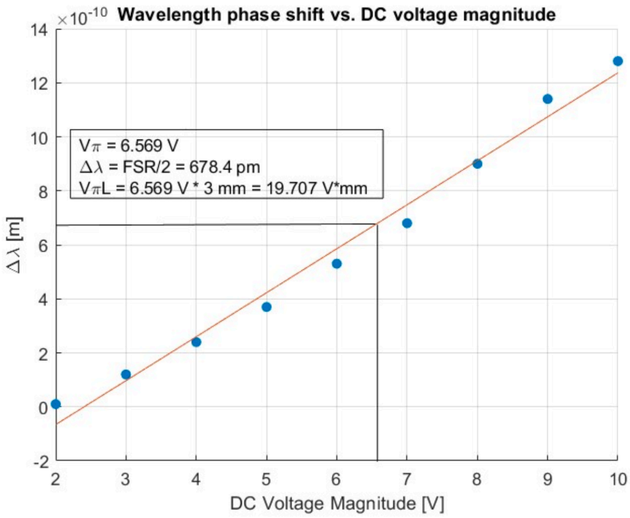


Figure 17 - *Wavelength phase shift vs Volts applied [5].*

Using Eq 5 and 6 we can obtain an r_{33} of 767 pm/V from the measured V_π.

VI. Discussion

The found V_π of 6.569 is too good alluding to either a mistake in calculation or an error in testing. Another possible reason for the low V_π is that only the negative voltage curves were used as the positive voltage curves showed no shift and no higher voltages were tested during our lab session. Furthermore due to the failing of manufacturing of my chip design, no real comparison can be made with found data and simulated data of my design.

In reviewing past work done by students in 2022, the vast majority of calculations of V_π yielded much less efficient devices, ranging from 40-100 V. It seems that a recurring issue in running these experiments is obtaining functional MZIs that can be used to compare simulated data vs experimental data reliably and with some form of statistical significance. This makes the largest limitation of future endeavours being able to design a MZI EOP modulator and then being able to reliably test the device for every student within the course, an endeavour that is quite difficult and time consuming.

VII. Conclusion

During this study, two chips were designed and manufactured, a primary modulating chip and a secondary non modulating chip. Unfortunately my design was not successfully manufactured for the primary chip and a colleagues design and active data were used. The found V_π was 6.569 V which is a relatively low voltage required for a π shift, thus it was concluded that some error in calculation or measurement was done throughout the study. I would recommend for any future endeavours to consolidate designs with a team of colleagues rather than attempt to manufacture a chip for every student, this will allow for more redundancies to be made and much easier analysis of chips between groups and between years.

Acknowledgment

I would like to thank Iman Taghavi for aiding in performing measurements, Sheri Jahan Chowdhury for aiding in the fabrications, and Lukas Chrostowski for being our instructor.

References

1. Chrostowski, L., & Hochberg, M. (2019). Silicon Photonics Design. Cambridge University Press.

2. Pei, L., Xi, Z., Bai, B., Wang, J., Zuo, X., Ning, T., Zheng, J., & Li, J. (2021). Key technologies of photonic artificial intelligence chip structure and algorithm. Applied Sciences, 11(12), 5719. https://doi.org/10.3390/app11125719

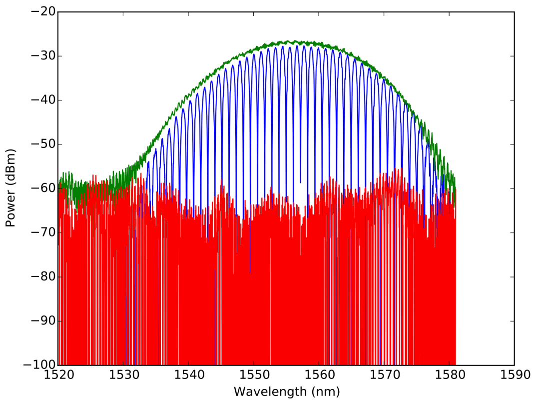
3. Taghavi, I., Moridsadat, M., Tofini, A., Raza, S., Jaeger, N. A., Chrostowski, L., Shastri, B. J., & Shekhar, S. (2022). Polymer modulators in Silicon Photonics: Review and Projections. Nanophotonics, 11(17), 3855–3871. https://doi.org/10.1515/nanoph-2022-0141

4. Nonlinear Materials Corporation. (n.d.). HLD1/HLD2 Recommended Handling, Solvent Casting, and Poling. Nonlinear Materials Corporation.

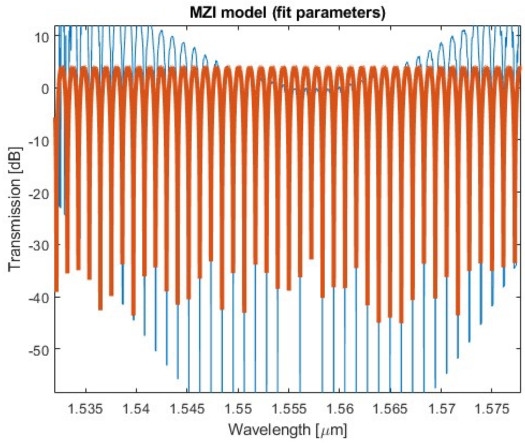
5. Nguyen, T. (n.d.). (rep.) (2023). Design, Fabrication, and Testing of Silicon Photonic MZI-Based Polymer Electro-Optic Modulators.

VIII. Appendix

A) Miscellaneous graphs



MZI 3 Power vs wavelength, pre metal deposition, functioning passively.

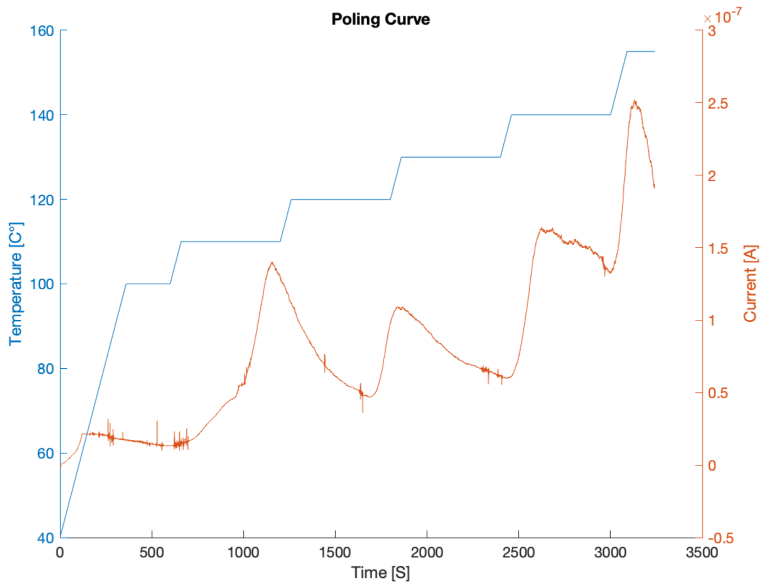


Partially successful Autocorrelation curve fitting attempt for chip-0

<u>Poling/Crosslinking Procedure</u>

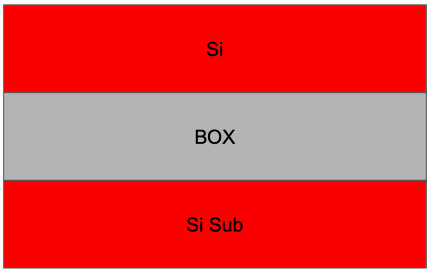
- The crosslinking reaction will take place during the heat ramp listed below. After crosslinking, the chromophore becomes insoluble and can no longer be washed away with solvent

1. Place device on poling stage and affix electrical connections
2. Start nitrogen purge (2-3 L/min).
3. Near room temperature step up voltage to desired poling field over ~1 minute
4. Heat to 95 ˚C at rate of ~10 ˚C/min. A current increase should be observed (unless a charge barrier layer is used)
5. Hold temperature for 10 min.
6. Increase to 105 ˚C and hold for 10 min.
7. Increase to 115 ˚C and hold for 10 min.
8. Increase to 125 ˚C and hold for 10 min.
9. Increase to 135 ˚C and hold for 10 min.
10. Increase to 145 ˚C and hold for 10 min.
11. Increase to 155 ˚C and hold for 10 min.
12. Let sample cool to ~35 ˚C or lower
13. Turn off poling field.



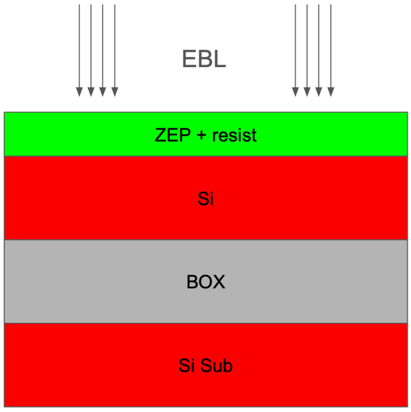
Poling curve obtained from poling HLD.

B) Cross section diagrams of manufacturing:

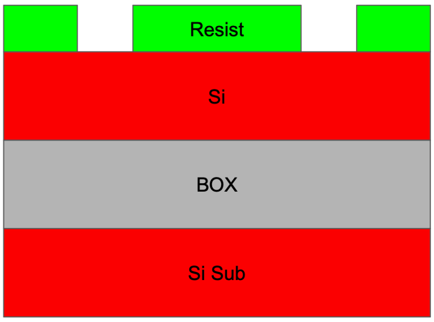


a) Standard SIO wafer

b) Zep + photoresist deposition

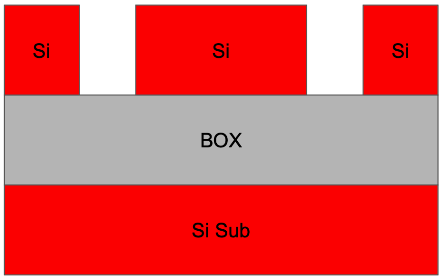


c) Electron beam lithography

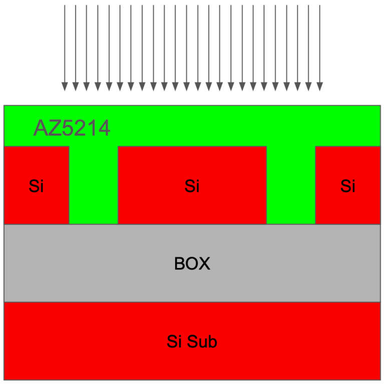


d) Zep Resist post development

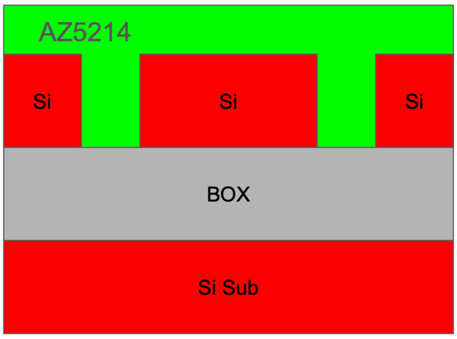
e) Plasma etch



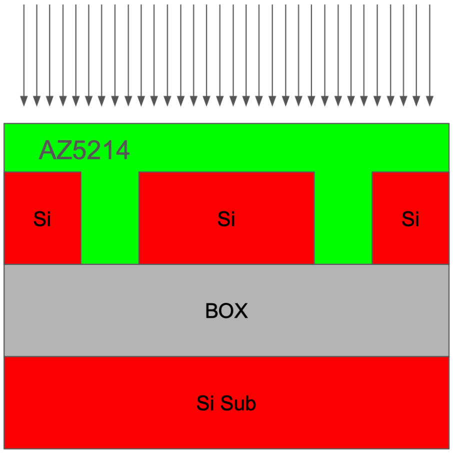
f) Strip Resist



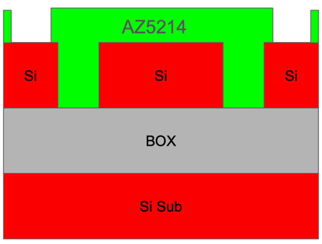
g) Az5214 positive resist deposition and first exposure



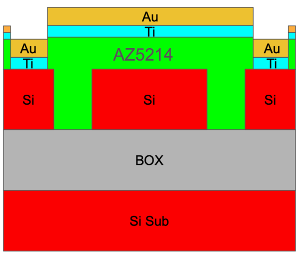
g) Az5214 Bake step in image reversal.



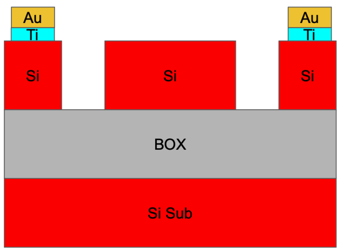
g) Az5214 second exposure in image reversal.



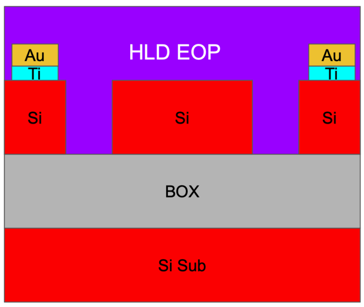
h) Az5214 developed as negative resist.



i) Titanium and Gold deposition.



j) Lift off

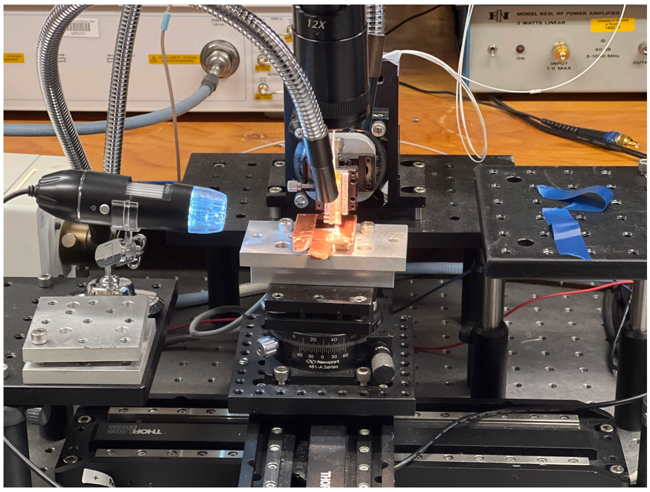


k) EOP polymer application, poling and cross linking.

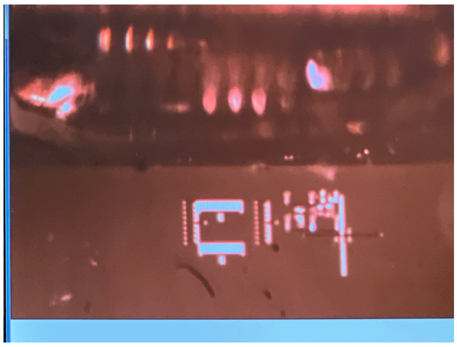
C) Photos from lab

Plasma etcher

Plasma etcher settings



Testing assembly



Aligning fibre array with chip's grating coupler.

D) Matlab scripts:

```
%%Voltage graphs

load("active data/0.mat")

% plot(wavelength,power(:,2))

findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)

hold on


load("active data/m1.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m2.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m3.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m4.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m5.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m6.mat")

% plot(wavelength,power(:,2))

findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m7.mat")

% plot(wavelength,power(:,2))

findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m8.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m9.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


load("active data/m10.mat")

% plot(wavelength,power(:,2))

%findpeaks(smoothdata(power(:,2),
'gaussian', 50), wavelength)


%legend('0V', '-1V', '-2V', '-3V', '-4V',
'-5V', '-6V', '-7V', '-8V', '-9V',
'-10V')


xlim([1.498e-6 1.559e-6])

xlabel("Wavelength")

ylabel("Power")


%% Neff

load Air_neff_vs_wavelength.mat;

plot(lum.x0,lum.y0)


hold on
```

```
load EO_neff_vs_wavelength.mat;

plot(lum.x0,lum.y0)


legend('Air neff','EO neff')

ylabel('Effective Index (Neff)')

xlabel('Wavelength(micrometers)')

%% ngroup

load Air_ng_vs_wavelength.mat;

plot(lum.x0, lum.y0)

hold on

load EO_ng_vs_wavelength.mat

plot(lum.x0, lum.y0)

ylabel('Group Index')

xlabel('Wavelength ')


legend('Air ng','EO ng')

%% Autocorrolation

%%

load("C:
\Users\benamin1.stu\Downloads\Neal_MZI3.m
at")

dL = 200;  % [micron] Path length
difference in the MZI

lambda=scandata.wavelength;

amplitude=scandata.power(:,1);

amplitude=transpose(amplitude);

a = find(lambda>1.52e-06);

lambda = lambda(a);

amplitude= amplitude(a)

plot(lambda, amplitude)

%% Curve fit data to a polynomial for
baseline correction

p=polyfit((lambda-mean(lambda))*1e6,
amplitude, 4);

amplitude_baseline=polyval(p,(lambda-
mean(lambda))*1e6);

% Perform baseline correction to flatten
the spectrum

% Use the curve polynomial, and subtract
from original data

amplitude_corrected = amplitude -
amplitude_baseline;

amplitude_corrected = amplitude_corrected
+ max(amplitude_baseline) -
max(amplitude);

amplitude_corrected =
amplitude_corrected(1,:);

%% data only within the wavelength range
of interest.

lambda_min = min(lambda);     % Can limit
the analysis to a range of wavelengths

lambda_max = max(lambda);   %  if the
data on the edges is noisy

lambda1=lambda_min:min(diff(lambda)):lamb
da_max;

amplitude=interp1(lambda,
amplitude_corrected, lambda1,'linear');

lambda=lambda1;

amplitude(find(amplitude==-inf))=-50;  %
check if there are -infinity data points

% plot baseline corrected spectrum

figure;

plot (lambda*1e6, amplitude);

xlabel ('Wavelength [\mum]');

ylabel ('Transmission [dB]');

axis tight

title ('Experimental data (baseline
corrected, wavelength range)');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Find ng from autocorrelation-based
frequency estimation of spectrum

% auto-correction

[r,lags]=xcorr(amplitude);

r=r(ge(lags,0));
```

```
lags=lags(ge(lags,0));

figure

plot(lags,r);

% estimate the frequency

d=diff(r);

start = find(gt(d,0)); start=start(1);

[peak_m, peak_i]=max(r(start:end));

peak_i=peak_i+start;  % location of the
1st peak in the autocorrelation

hold on;

plot(peak_i,0,'s','MarkerSize',20);

title ('Autocorrelation of spectrum')

xlabel('lag, sample number');

fsr = peak_i * mean(diff(lambda))

ng_av = mean(lambda)^2/(dL*1e-6)/fsr

% find starting point for curve fitting,
using the ng value

% lambda0 is in microns.

lambda0 = mean(lambda) * 1e6;

n1=2.4;

%modeNumber = n1_initial * dL / lambda0 -
0.5;

%n1 = (2*floor(modeNumber)+1)*lambda0/2/
dL;

n2 = (n1-ng_av)/lambda0;

nx_init = [n1 n2 0];

alpha_init = 1e-3;  % propagation loss
[micron^-1]

x0=[nx_init, alpha_init, 0];

% Define the MZI transfer function

% - as a Taylor expansion around the
central wavelength

% - Use units of [microns] – keeps the
variables closer to 1.

% - These make the curve fitting easier.

% use Matlab anonymous functions

% effective index:

neff = @(nx, lambda) ...

            (nx(1) + nx(2).*(lambda-
lambda0) + nx(3).*(lambda-lambda0).^2);

% neff([2.4, -1, 0], 1.56)  % test it.

% alpha = 1e-3;  % propagation loss
[micron^-1]

% complex propagation constant

beta = @(nx, alpha, lambda) ...

            (2*pi*neff(nx, lambda)./
lambda - 1i*alpha/
2*ones(1,length(lambda)) );

% beta([2.4, -1, 0], 1e-3, [1.56, 1.57])
% test it.

% MZI transfer function

T_MZI = @(X, lambda) ...

       (10*log10( 0.25*
abs(1+exp(-1i*beta(X(1:3), X(4),
lambda)*dL)).^2) +X(5) );

% T_MZI([2.4, -1, 0, 1e-3], [1.56, 1.57])
% test it.

figure;

plot (lambda*1e6, amplitude);

hold all;

plot (lambda0, -40,'s','MarkerSize',20);

plot(lambda*1e6, T_MZI(x0,
lambda*1e6),'--','LineWidth',3);

xlabel ('Wavelength [\mum]');

ylabel ('Transmission [dB]');

axis tight

title ('MZI model (initial parameters)');

% Autocorrelation again, to find the
shift between the fit function and
experimental data

[r,lags]=xcorr(amplitude, T_MZI(x0,
lambda*1e6));

r=r(ge(lags,0));

lags=lags(ge(lags,0));

[peak_m, peak_i]=max(r);
```

```
lambda_offset = peak_i(1) * mean(diff(lambda));

n_shift = lambda_offset*lambda0/fsr/dL;

x0(1)=x0(1)+n_shift;

figure;

plot (lambda*1e6, amplitude);

hold all;

plot (lambda0, -40,'s','MarkerSize',20);

plot(lambda*1e6, T_MZI(x0, lambda*1e6),'--','LineWidth',3);

xlabel ('Wavelength [\mum]');

ylabel ('Transmission [dB]');

axis tight

title ('MZI model (initial parameters, with shift)');

% Curve fit:

[xfit,resnorm] = lsqcurvefit(T_MZI,x0,lambda*1e6,amplitude);

xfit;

r=corrcoef(amplitude,T_MZI(xfit, lambda*1e6));

r2=r(1,2).^2;

figure;

plot (lambda*1e6, amplitude);

hold all;

plot(lambda*1e6, T_MZI(xfit, lambda*1e6),'LineWidth',3);

xlabel ('Wavelength [\mum]');

ylabel ('Transmission [dB]');

axis tight

title ('MZI model (fit parameters)');

%% Check if the fit is good.  If so, find ng

if (ge(r2,0.8))

 % plot ng curve

 figure;

 neff_fit = neff(xfit(1:3),lambda*1e6);

 dndlambda=diff(neff_fit)./diff(lambda);
dndlambda=[dndlambda, dndlambda(end)];

 ng=(neff_fit - lambda .* dndlambda);

 plot(lambda*1e6, ng, 'LineWidth',4);

 xlabel ('Wavelength [\mum]');

 ylabel ('Group index, n_g');

 axis tight

 title ('Group index (from MZI fit)');


 % waveguide parameters at lambda0

 ng0 = xfit(1) - lambda0*xfit(2);

end


%% FSR [5]

%% Load data

V0 = load("active data/m1.mat")

name = "m1"


%% Plot calibration loopback

lambda=V0.wavelength

amplitude=V0.power(:, 1)

figure(1);

plot (lambda*1e6, amplitude);

title ('Calibration loopback');

xlabel ('Wavelength [\mum]')

ylabel ('Insertion Loss [dBm]')

hold all;


%% Fit calibration loopback in wavelength of interest

% Fit the data with a polynomial

p=polyfit((lambda-mean(lambda))*1e6, amplitude, 5);

amplitude_LOOPBACK=polyval(p,(lambda-mean(lambda))*1e6);
```

```
plot (lambda*1e6, amplitude_LOOPBACK);

% find wavelength range with usable data,
in the loopback

loopback_IL = max(amplitude);

new_lambda_i=find(amplitude>loopback_IL-1
0);

lambda=lambda(new_lambda_i);

lambda_min = min(lambda);

lambda_max = max(lambda);

amplitude=amplitude(new_lambda_i);

% refit the loopback

LOOPBACK=polyfit((lambda-
mean(lambda))*1e6, amplitude, 4);

amplitude_LOOPBACK=polyval(LOOPBACK,
(lambda-mean(lambda))*1e6);

plot (lambda*1e6,
[amplitude_LOOPBACK],'r-','Linewidth',5);

axis tight;

%% Find peak

center_lambda = find(amplitude_LOOPBACK
== max(amplitude_LOOPBACK))

xline(lambda(center_lambda)*1e6)

%%

% MZI data:

lambda1=V0.wavelength

amplitude=V0.power(:, 2)

figure(2);

plot (lambda1*1e6, amplitude);

title ('MZI (raw data)');

xlabel ('Wavelength [\mum]')

ylabel ('Insertion Loss [dBm]')


%%


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% MZI data - calibrated

%

% data only within the bandwidth of
interest.

lambda=lambda_min:min(diff(lambda1)):lamb
da_max;

amplitude=interp1(lambda1, amplitude,
lambda,'linear');

amplitude(find(amplitude==-inf))=-50;

% calibrate data

amplitude_cal=amplitude-polyval(LOOPBACK,
(lambda-mean(lambda))*1e6);

figure(3);

plot (lambda, amplitude_cal);

hold on;

title ('MZI (calibrated with loopback)');

xlabel ('Wavelength []')

ylabel ('Insertion Loss [dB]')


%% Smooth data

amplitude_cal_smooth =
smoothdata(amplitude_cal, 'gaussian', 50)

plot(lambda*1e6, amplitude_cal_smooth);

legend(["No smoothing", "Smoothing"])


%% Find peaks

findpeaks(amplitude_cal_smooth,lambda);

[pks, locs] =
findpeaks(amplitude_cal_smooth,lambda);


%% Calculate average FSR

npks = length(pks)

for n = 1:npks-1

    fsr(n) = locs(n+1) - locs(n)

end

average_fsr = mean(fsr)
```

```
%% Save to file

save(sprintf("peaks\\%s.mat", name),
"average_fsr", "amplitude_cal_smooth",
"lambda", "pks", "locs")
```