



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

High-precision control optimization and validation of n-order Mach- Zehnder meshes for complex-values optical computing

TESI DI LAUREA MAGISTRALE IN
ELECTRONICS ENGINEERING - INGEGNERIA ELETTRONICA

Author: **Edoardo Mauri**

Student ID: 246888

Advisor: Prof. Marco Sampietro

Co-advisors: Alessandro di Tria, Francesco Zanetto

Academic Year: 2024-25

Abstract

Developments in the information technology field have always been strictly linked to the evolution of the associated electronics circuits and systems, but this trend, along with Moore's law, is nearing its end.

Applications such as Artificial Intelligence (AI) and data centers are growing in both dimension and complexity, pushing electronics to its physical limits. Increasing the frequency and size of a digital electronic circuit results in a more than proportional increase in power consumption, following the relation $P \approx fCV^2$. Moreover, copper wires are gradually being substituted with faster optical data lines in many applications where data integrity is mandatory. For these reasons, an alternative to standard electronic systems must be found and a cooperation of photonics and digital electronics seems to be a step in the right direction.

Photonics, by means of interferometer meshes, can mitigate all of these problems. Data transmission, at least for the photonic domain path, has a negligible power loss, independent of the data rate, giving us the possibility to operate at much higher frequencies and for longer distances without the risk of losing the signal integrity because all the transmission is done at the speed of light. Moreover, thanks to the linear behavior of most optical components, it is possible to design photonic structures that perform mathematical operations, namely Matrix-Vector Multiplication (MVM) which is the fundamental operation performed in neural networks training.

However, the photonic alternative is not without challenges. Process non-idealities and environmental contributions strongly affect optics; thermal drifts or crosstalk change the behavior of optical components, resulting in an unreliable and unexpected result. The solution to these problems comes from the cooperation of both worlds. An optical mesh is used in combination with an electronic control layer in order to achieve, thanks to a specifically designed control algorithm, a precise and stable behavior. To achieve this, an FPGA-based control loop is used in our setup, allowing reprogrammability and flexibility for the control system to be used with larger mesh sizes as well. The sensing is done by means of transparent photodiodes, characterized by minimal insertion losses, needed

to measure power and phase at any point of the mesh without affecting the signal light, while the actuation is done by changing locally the optical fiber temperature with TiN heaters.

The focus of this thesis was to validate and optimize the control chain for a 3x3 optical mesh realized with Mach-Zehnder interferometers. The whole system has been characterized and validated at a wavelength $\lambda = 1550nm$, confirming the accuracy reached with a previous implementation of a 2x2 mesh, with a setting error as low as 2%. This demonstrates the possibility to extend our proposed control system to larger meshes, thus paving the way for electro-optical computing.

Keywords: MZI, Optical computing, FPGA, transparent photodiode, Matrix vector multiplication, Control

Abstract in lingua italiana

Lo sviluppo della tecnologia dell'informazione è sempre stato strettamente legato all'evoluzione dell'elettronica associata, ma questa tendenza, insieme alla legge di Moore, sta giungendo al termine. Applicazioni come l'intelligenza artificiale e i data center stanno crescendo sia in dimensioni che in complessità, spingendo l'elettronica ai suoi limiti fisici. L'aumento della frequenza e delle dimensioni di un circuito elettronico digitale comporta un incremento più che proporzionale del consumo di potenza, secondo la relazione $P \approx fCV^2$; inoltre, esistono molte applicazioni in cui le linee dati ottiche sono indispensabili per la salvaguardia dell'integrità dei dati. Per questa ragione, è necessario trovare un'alternativa, e una cooperazione tra la fotonica in silicio e l'elettronica digitale sembra essere un passo nella giusta direzione.

La fotonica, per mezzo di reti di interferometri, può risolvere o evitare quasi tutti questi problemi. La trasmissione dei dati, almeno per il percorso nel dominio ottico, ha una perdita di potenza trascurabile e indipendente dalla velocità di trasmissione, dandoci la possibilità di operare a frequenze molto più elevate e su distanze maggiori senza il rischio di perdere l'integrità del segnale, poiché tutta la trasmissione avviene alla velocità della luce.

Tuttavia, l'alternativa fotonica non è priva di sfide. Le non-idealità di processo e i contributi ambientali influenzano l'ottica allo stesso modo dell'elettronica, o anche di più; le derivate termiche o il crosstalk modificano il comportamento rispetto a quello previsto, portando a un risultato inaffidabile e inaspettato. La soluzione a questi problemi deriva dalla cooperazione di entrambi i mondi. Una rete ottica viene utilizzata in combinazione con una catena di controllo elettronica per ottenere, grazie a un algoritmo di controllo appositamente progettato, un comportamento preciso e stabile. Per raggiungere questo obiettivo, nel nostro setup viene utilizzato un anello di controllo basato su FPGA, che consente riprogrammabilità e flessibilità affinché il sistema di controllo possa essere utilizzato anche con reti di dimensioni maggiori. Il rilevamento del segnale ottico (sensing) viene effettuato tramite fotodiodi trasparenti, caratterizzati da perdite di inserzione minime, necessari per misurare potenza e fase in qualsiasi punto della rete senza influenzare il segnale luminoso, mentre l'attuazione avviene modificando localmente la temperatura

della fibra ottica con riscaldatori in TiN (nitruro di titanio).

L'obiettivo di questa tesi è stato validare e ottimizzare la catena di controllo per una rete ottica 3x3 realizzata con interferometri di Mach-Zehnder. L'intero sistema è stato quindi caratterizzato e validato utilizzando un laser alla lunghezza d'onda $\lambda = 1550nm$, confermando l'accuratezza raggiunta con una precedente implementazione di una rete 2x2, con un errore di settaggio in alcuni casi inferiore al 2%, e dimostrando la possibilità di estendere questo controllo a reti di dimensioni maggiori, aprendo così la strada al calcolo elettro-ottico.

Parole chiave: MZI, Calcolo ottico, FPGA, fotodiodi trasparenti, Moltiplicazione vettore matrice, Controllo

Contents

Abstract	i
Abstract in lingua italiana	iii
Contents	v
Introduction	1
1 Fundamental structures	3
1.1 Silicon for optics and electronics	3
1.1.1 Waveguide and light propagation	5
1.2 Actuators	7
1.3 Light detectors	9
1.3.1 Transparent detectors	10
1.4 Interferometers and couplers	13
1.4.1 Couplers	13
1.4.2 -3dB Coupler	14
1.4.3 Mach-Zehnder Interferometer	15
2 Basics on Optical Computing	17
2.1 Classification of photonic architectures	17
2.2 Photonic Coherent Architectures	19
2.3 Mesh composition	22
2.3.1 Reck architecture	22
2.3.2 Clements architecture	24
3 Matrix setting and control	27
3.1 MZI setting	28
3.1.1 Setting point evaluation	29
3.2 3x3 MZI implementation	34

3.2.1	3x3 Mesh transfer function	36
3.2.2	Input generator	38
4	Electronic control system	43
4.1	Mantis PCB	44
4.2	Helios 3 motherboard	45
4.2.1	Acquisition module	46
4.2.2	Photodiodes readout	48
4.2.3	Actuation module	50
4.3	FPGA Algorithm Implementation	51
4.3.1	Low-Pass Chain	51
4.3.2	Lock-in chain	54
4.3.3	Integral control	55
5	System characterization and experimental validation	59
5.1	TiN heaters electrical behavior	60
5.1.1	100 μm Heaters	62
5.1.2	60 μm Heaters	64
5.2	Transparent photodiodes non linearity	67
5.3	Phase measurement and matrix setting	70
5.3.1	MZI phase measurement	71
5.3.2	Matrix setting and precision	75
5.3.3	Manual input vector setting	76
5.3.4	Error map	78
6	Conclusions	87
	Bibliography	89
	A Appendix A	93
	List of Figures	101
	List of Tables	109

Introduction

The continuous expansion of modern computational models demands hardware architectures capable of performing complex linear algebra with unprecedented speed and minimal energy overhead [1]. While conventional complementary metal-oxide-semiconductor (CMOS) platforms face insurmountable thermal dissipation and bandwidth bottlenecks, Silicon Photonics provides a disruptive architectural alternative [2]. By manipulating optical phases and amplitudes within programmable networks of Mach-Zehnder Interferometers (MZIs), operations such as Matrix-Vector Multiplications (MVMs) can be executed entirely in the optical domain [3, 4]. This paradigm reduces computational latency to the optical time-of-flight and minimizes active power dissipation during data routing. However, the physical realization of complex-valued optical computing is limited by the susceptibility of photonic integrated circuits to fabrication tolerances, environmental drift, and thermal cross-talk. Static open-loop calibration methods, such as the construction of complex look-up tables [5], fundamentally fail to dynamically counteract these non-idealities. To overcome this, recent foundational research [6] introduced a calibration-free localized feedback control strategy that successfully stabilized a single 2×2 MZI using transparent detectors and thermo-optic actuators. Starting from that premise, the primary objective of this thesis is to scale, optimize, and experimentally validate this distributed electronic control architecture onto a higher-order 3×3 MZI mesh based on the Reck topology [7]. By leveraging a Field Programmable Gate Array (FPGA) to manage in-line power monitoring via transparent photodiodes based on Surface-State Absorption (SSA), and localized actuation via Titanium Nitride (TiN) micro-heaters, the proposed system forces unitary transformations without requiring resource-intensive global training.

1 | Fundamental structures

In this chapter, for a better understanding of the whole system, an introduction to the main components and structures adopted in optical computing will be done.

1.1. Silicon for optics and electronics

Silicon is the backbone of today's electronics, thanks to the decade-long evolution of the CMOS technological process. In more recent years, silicon has also been investigated for photonics applications, thanks to its excellent optical properties and the expertise stemming from the CMOS process, which led to the development of a new *Silicon Photonics* process [8]. This duality of silicon is the main reason why an hybrid system seems to be the perfect choice to exploit the best of both worlds but to use silicon, a proper understanding of its optical properties is mandatory.

The intrinsic silicon band gap is $E_g \simeq 1.12eV$ at ambient temperature, with a subtle dependence on temperature given by the Varshni equation [9], $E_g(T) = E_g(0) + \frac{\alpha T^2}{T + \beta}$, with $E_g(0)$ the energy gap at a temperature of $0K$, T the temperature and α and β material-specific fitting parameters to be determined experimentally.

For a proper transmission of light we need to use light frequencies with a photon energy low enough to avoid absorption, and in silicon this can be found for wavelengths above $1100nm$, where the material is *transparent*. Therefore, the most common wavelengths used in optics for single-mode transmission are $1310nm$ and $1550nm$, both corresponding to absorption minima in the infrared range.

At $1550nm$ we have the lowest possible attenuation, which is optimal for long distance transmissions, and thus it will be the wavelength of interest in our case.

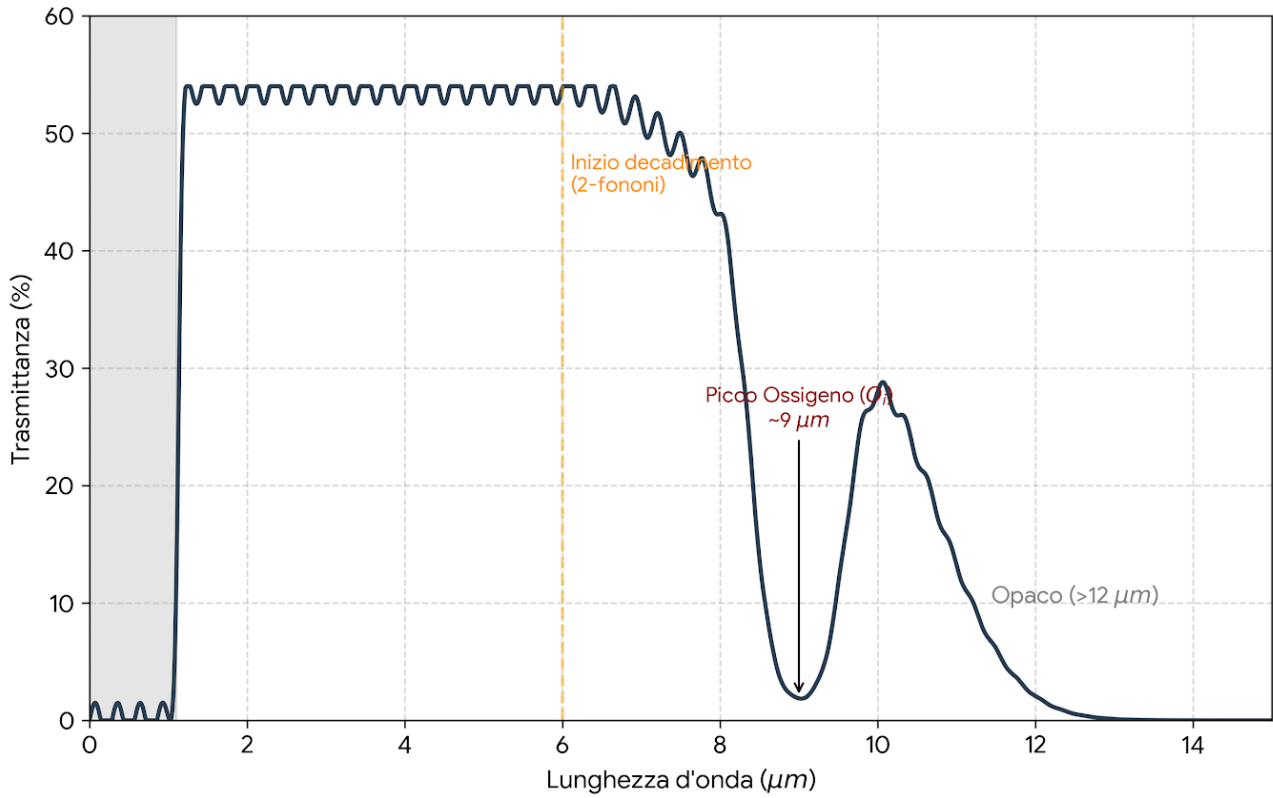


Figure 1.1: Silicon transmittance, 1550nm falls perfectly inside transparency window

What the transmittance plot in Figure 1.1 does not show is the dependence of the optical properties of silicon on temperature [10] (Figure 1.2).

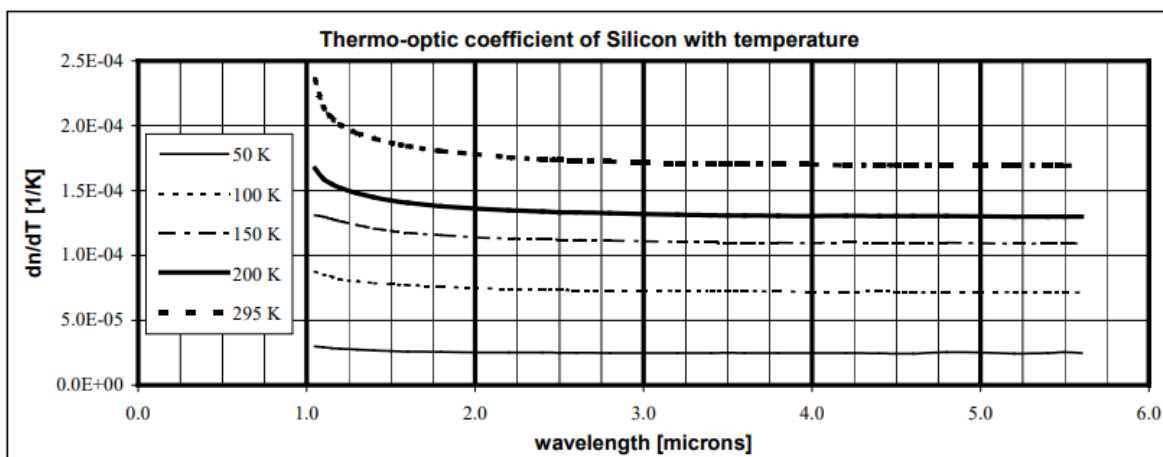


Figure 3: Measured thermo-optic coefficient (dn/dT) of silicon as a function of wavelength for selected temperatures

Figure 1.2: Silicon thermo-optical coefficient as a function of temperature

Here, silicon's thermo-optical coefficient is represented as a function of both wavelength and temperature, showing an increase of the coefficient with temperature. This means that the operation of photonic devices strongly depends on the local temperature of the circuit and can be affected by drifts, resulting in the necessity to have an active control that stabilizes their operation.

1.1.1. Waveguide and light propagation

The waveguide (WG) is the basic structure used to confine and direction light inside of a photonic structure, representing the photonic alternative to electrical wires.

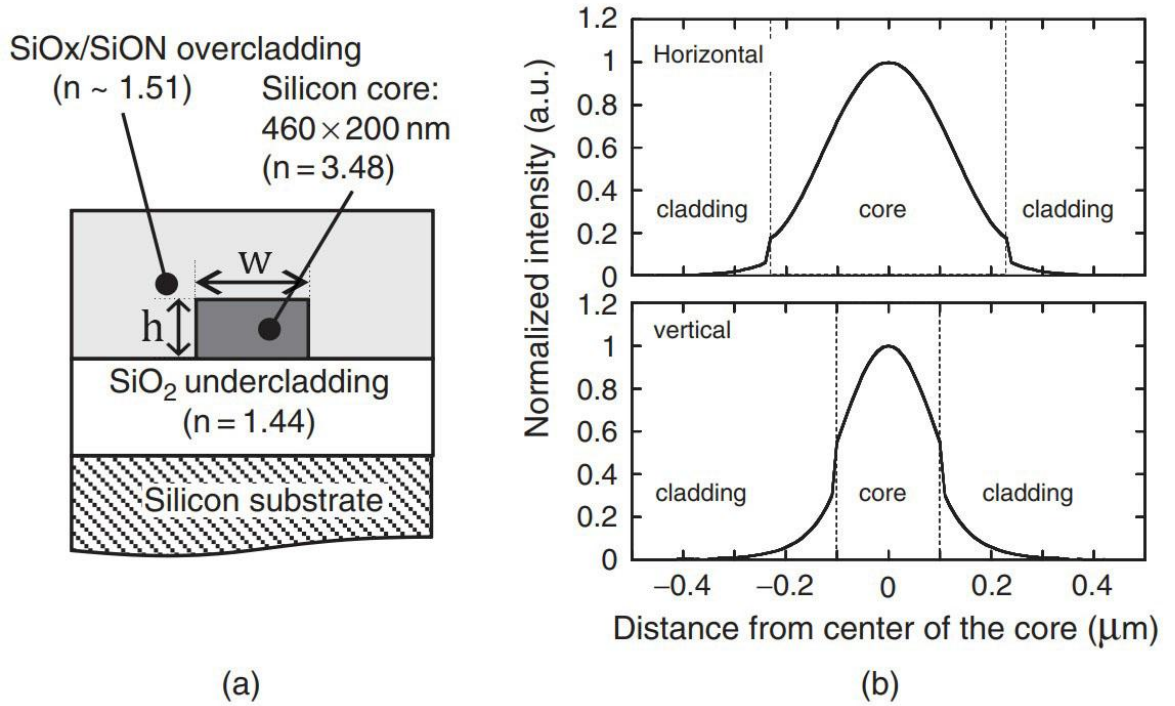


Figure 1.3: a)Example of a buried waveguide b)Representation of light intensity inside of the waveguide for a different size of the core. For the largest core the light is better confined and the losses due to tunneling in the cladding are limited.

For their realization, silicon and silicon oxide are the ideal candidates due to the large difference between their refractive indices, $n_{Si} \simeq 3.45$ while $n_{SiO_2} \simeq 1.45$. Thanks to this, by using *Si* for the guide core and *SiO₂* for the cladding, thanks to the 40% difference between the two refractive indices, a total internal reflection is achieved up to an incident angle of 60°.

Considering a circular infinite waveguide, an approximation of the minimal diameter capable of carrying the wavelength of our interest can be done starting from the V-number equation.

$$d = \frac{\lambda V}{2\pi} \cdot \frac{1}{\sqrt{n_{core}^2 - n_{cladding}^2}}$$

Where $n_{core} = n_{Si} = 3.45$ and $n_{cladding} = n_{SiO_2} = 1.45$ are the two refractive indices, V is the $V - number$ of the waveguide, which is lower than 2.405 for single-mode fibers, and λ is the wavelength of interest that is $1550nm$ in our case. Considering a single mode regime, we get a minimum core size of

$$d \cong 189.5nm$$

The minimum required size sets the physical limit of photonics in terms of area occupation, which is not further scalable. Beyond that, a too thin core is generally non preferable, as can be seen in Figure 1.3b. By comparing two waveguides with the same size, for a smaller core we will have a higher power loss due to tunneling in the cladding.

Once defined the properties of the material we are using to carry our light, we need a mathematical representation of the light traveling in the waveguide, in particular of its associated electrical field:

$$E_{in} = E_{out}e^{j\beta L}e^{-\alpha L}$$

the two exponential terms represent an amplitude decay and a phase shift, respectively. α is a material-dependent coefficient variable along the electromagnetic spectrum, resulting in the importance of choosing the appropriate wavelength wisely.

The phase term is of primary importance in optical computing, and represents the cardinal principle thanks to which we can perform vector-matrix operations. As its relation shows: $\beta = \frac{2\pi}{\lambda}n_{eff}$, the dependence is strictly bonded to the refraction index of the material, and can therefore be modified in a controlled manner by means of actuators. Given the amplitude-phase notation, a complex representation will be often used for transfer matrices.

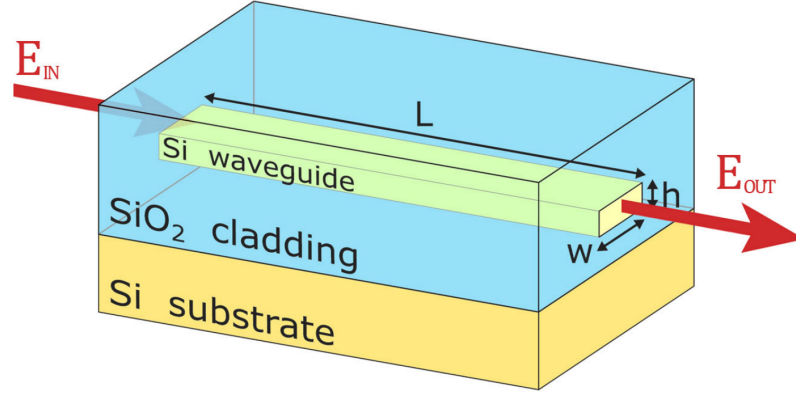


Figure 1.4: Example of a buried silicon waveguide

1.2. Actuators

Optical actuators are devices that enable the local control of the refractive index in waveguides or optical materials. As mentioned in 1.1.1, a controlled change in the refractive index is crucial to electrically tune the behavior of the optical device of interest, opening up the possibility of realizing tunable filters, modulators and programmable photonic circuits where, independently from the physical phenomena inducing the change, the phase variation produced over a length L is: $\Delta\varphi = \frac{2\pi}{\lambda}\Delta n_{eff}L$. To produce this effect, the most used actuators are:

- **Electro-optic** actuators [11]: these actuators use the Pockels or Kerr effect, where the refractive index shows a dependence on the applied electric field. Kerr effect produces a refractive index variation $\Delta n \approx E^2$ and can occur in any dielectric material, while Pockels effect only concerns non-centrosymmetric materials with a produced shift directly proportional to the electric field. This actuation is usually exploited where speed is the main goal, with frequencies in the tens of GHz range reachable with materials like lithium niobate. In addition to this, they show low power consumption, with almost no heat produced, reducing to the lowest the possibility of thermal cross-talks. However, high insertion losses are usually a concern with this technology and the footprint is in the mm range, preventing their use in scalable technologies.

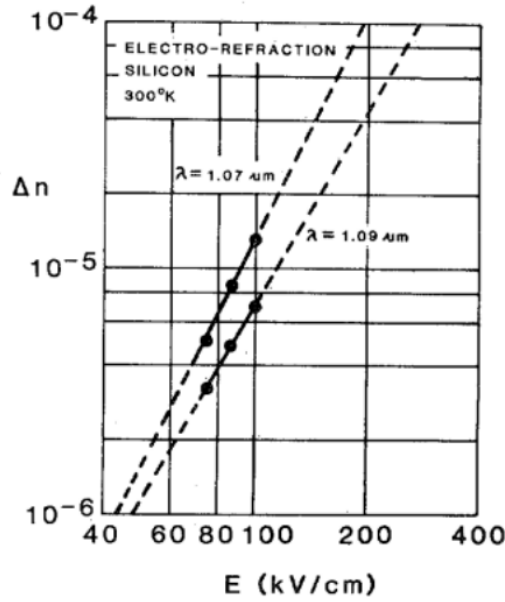


Figure 1.5: Dependence of the refractive index as a function of the applied electric field at two wavelengths

- **Thermo-optic** actuators [12]: they exploit the dependence of the refractive index on temperature, as shown in Figure 1.2, with a resulting phase shift $\Delta\varphi \approx \frac{dn}{dT} \Delta T$. They are basically small resistances, usually of the order of $100 \mu\text{m}$, which locally heat the photosensitive material to reach the desired phase shift, with a power consumption of the order of $\approx 20 \frac{\text{mW}}{\pi}$. Because of the resistive principle of the actuator, the produced heat is proportional to power, and thus the square of the voltage, with a phase variation following this trend. When using these actuators, thermal cross-talk must be taken into account, especially in dense integrated meshes where tens of these heaters are used closely together. A way to limit their interaction is to design the structure including trenches in between them and on the edges of the heaters, placing a physical limit to the diffusion of heat and speeding up the heater's heating. Their use is usually avoided in the case of bulky structures, because the required heating time will result in a too slow response time.
- **Opto-mechanical** actuators [13]: they rely on microelectromechanical system (MEMS) tuning mechanisms that implement a physical stretch of the optical sensitive material. These devices apparently combine the positive aspects of both worlds, with a footprint in the order of $\approx 2000 \mu\text{m}^2$ which makes them comparable in size to thermo-optical actuators, and a tuning time of about $\approx 2.5 \mu\text{s}$ that enables possible operations in the order of hundreds of MHz. Moreover, with these sizes and relatively small driving voltages, a phase shift of more than 3π is achievable at 1550nm ,

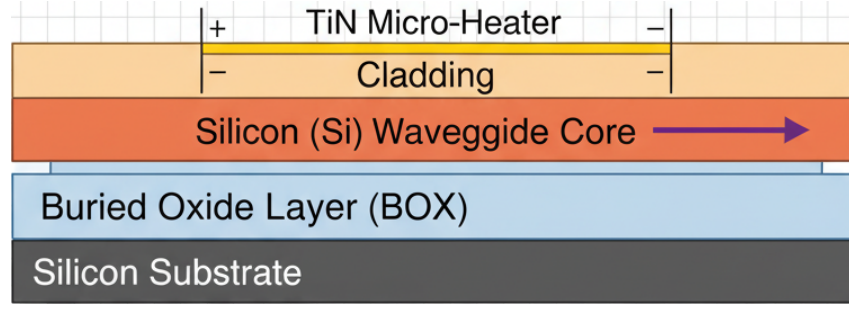


Figure 1.6: Example of a buried silicon waveguide with a titanium nitride heater on top

opening the possibility to a possible use in photonic integrated circuits.

1.3. Light detectors

Light detection is fundamental in photonic circuits in many ways. Photodetectors are used to monitor the optical power reaching different points of the optical circuit, and can be used to check its correct operation and to possibly allow the operation of a control system.

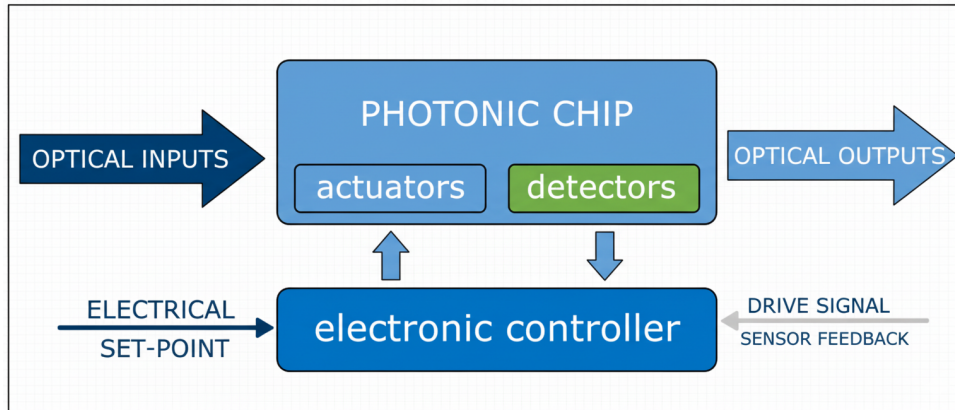


Figure 1.7: Control chain schematic of a PIC

Traditional photodetectors measure the light power by absorbing photons and converting them into electron-hole pairs. In silicon at the $1550nm$ wavelength, this is usually achieved using germanium photodiodes, which have the correct band-gap to absorb light at that wavelength [14].

These devices base their functioning on the absorption of the incoming light; a p-n junction, or better a p-i-n junction to improve device's efficiency, is generally used to fulfill this task. Energy from the absorbed photons is then used to promote electrons in the conduction band, generating a free electron-hole pair which represents the equivalent elec-

trical signal associated to a single photon absorption.

Further improvements can be done, for example by multiplying electron-hole pairs by means of local positive feedback as in avalanche photodiodes [15], but the generic responsivity of this kind of devices can be expressed as:

$$R(\lambda) = \frac{\eta q}{h\nu}$$

Where η is the quantum efficiency of the material, representing the percentage of incident photons successfully converted, q is the elementary charge, ν is the frequency of the incoming light and h is the Planck constant.

It is straightforward that, without tapping the guide and extracting only a portion of the light, most of the optical signal is absorbed, resulting in a destructive light measurement, suitable only at the end of our acquisition chain at the output of the mesh.

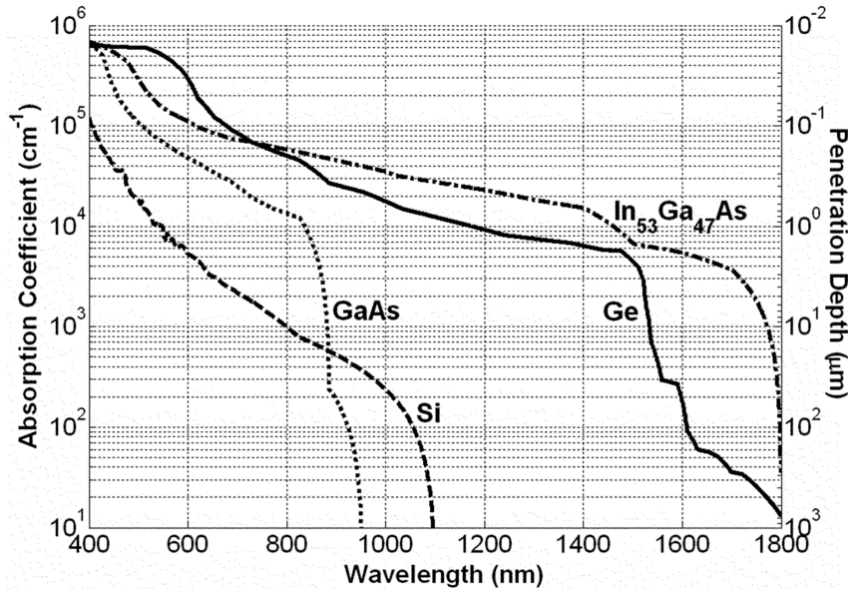


Figure 1.8: Absorption coefficient for different materials as a function of the wavelength. Around the wavelength of interest (1550nm) *Si* shows the worst absorption together with *GaAs*, while *Ge* and *In₅₃Ga₄₇As* shows a better behavior.

1.3.1. Transparent detectors

The only way to measure light without affecting its integrity is by means of transparent photodiodes. These devices, differently from standard photodetectors, do not rely on direct photon absorption, but on an intrinsic phenomenon occurring in silicon waveguides even for wavelengths where the material is transparent, which is *Surface state absorption (SSA)* [16]. SSA is the absorption of light at the boundary of the waveguide, where the

crystalline structure of the silicon core is interrupted by the oxide cladding, creating trap sites with intermediate energy levels.

Due to these levels, wavelengths that usually are not absorbed by the material (the ones in the IR spectrum, for example), will have enough energy to promote electrons to the trap sites and from the trap sites to the conduction band, producing an electrical signal proportional to the optical power. The SSA mechanism unlocks the possibility to sense the light inline with the waveguide in any point of the photonic circuit, at the cost of a lower responsivity with respect to standard absorbing photodiodes.

A brief review of SSA-based transparent sensors will be provided.

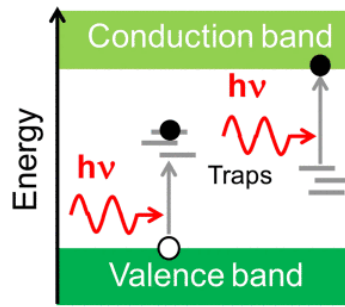


Figure 1.9: SSA two-step absorption process

- **CLIPP sensors:** One way to sense variations in the optical signal in the guide is by measuring variations in the guide's resistance by means of a *CLIPP* sensor (*Contact-Less Integrated Photonic Probe*) [17]. In this sensor, the cladding of the waveguide under test is contacted by means of two electrodes, creating a capacitive coupling between the guide and the electrodes. A high-frequency signal is then used to polarize this capacitive coupling, resulting in an output current proportional to the conductance of the waveguide and therefore also to its variations.
- **Transparent p-i-n photodiodes:** When instead selective doping is available, another approach is to use the waveguide as the intrinsic zone of a p-i-n photodiode [6]. The guide is widened in the vicinity of the photodiode, in order to reduce the electrodes interaction with the waveguides core, and two highly-doped silicon areas are placed transversely to the guide to create a diode-like structure. The structure is then biased, and the free charges in the guide generated by the SSA mechanism are collected by the two electrodes, with a resulting electrical signal proportional to the flowing light. It is important to notice that although silicon shows a dependence on the refractive index, as shown in Figure 1.5, in this case we are applying a constant electric field to the junction, which will lead to an offset that can be easily

eliminated in post-processing.

- **Transparent photoconductors:** Transparent photoconductors are a higher-gain alternative to transparent photodiodes [18]. Their structure is similar to that of p-i-n photodiodes, with two highly-doped regions of the same type placed laterally with respect to the guide, designed to ensure ohmic electric contacts, and a slightly doped waveguide among them. Their functioning is based on the different travel times of holes and electrons. The electron, characterized by higher mobility, reaches on average the positive electrode while the hole is still traveling in the slightly-doped core. To maintain an overall charge neutrality, another electron is then injected from the other electrode and this process proceeds until the hole reaches the negative electrode. This process, also known as the *recirculating current mechanism*, extends the generated current duration, increasing the overall number of carriers resulting from a single photon absorption.

To avoid optical losses, doping the guide is generally not a good option, because it increases insertion losses deriving from the free carriers in the waveguide. On the other hand, an intrinsic guide shows a much higher resistance, which makes it difficult to detect small variations due to the SSA mechanism, and the performance of the sensor is limited by the presence of oxide-trapped charges. However, it has been demonstrated that, by applying a voltage at the chip substrate, the effect of these charges can be compensated and the conductivity tuned, bringing the waveguide back to similar or even better performances than the doped one.

The process just described has linked for a long time the Quantum Efficiency (QE) or Gain (G) in photoconductive devices to the expression $G = \frac{\tau}{\tau_{tr}}$ where τ is the carrier recombination lifetime and τ_{tr} is the carrier transit time, but it has been demonstrated that intrinsic devices are gain-limited to unity and that high gain is inherently a secondary photoconductivity phenomenon driven by trapping/accumulation effects, rather than being a simple consequence of the $\frac{\tau}{\tau_{tr}}$ ratio [19].

1.4. Interferometers and couplers

1.4.1. Couplers

Optical couplers are devices specifically designed to split or combine different optical signals in optical systems. Their use ranges from signal monitoring, realizing power taps to collect part of the signal from the main waveguide, power distribution and optical multiplexing. Couplers generally can be divided into two main classes, *directional* and *non-directional* couplers. Directional couplers allow light transfer in only one direction, marking a clear distinction between inputs and outputs, while non-directional couplers transmit light in both forward and reflected directions, resulting in an overall light readout generally less precise and more complex.

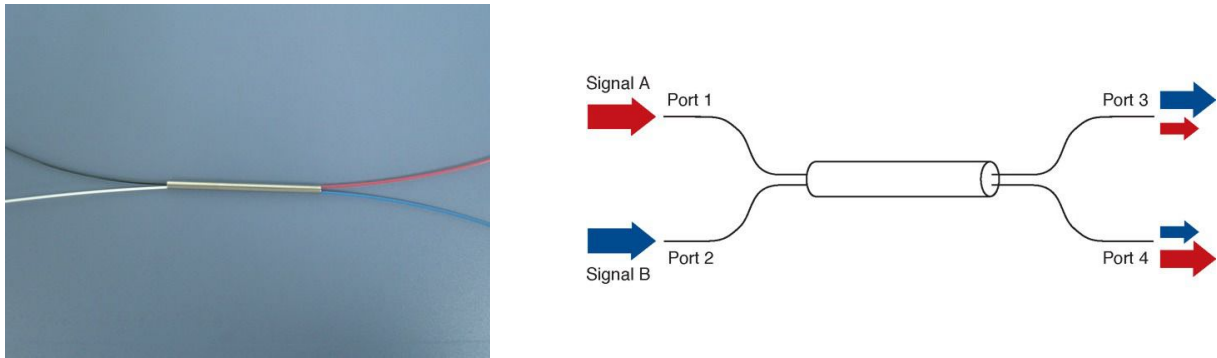


Figure 1.10: 2x2 directional coupler realized with optical fibers

Regardless of directionality, some common coupler structures are as follows:

- **Y couplers:** also called tap couplers, their function is to split a single light beam in two, with a precisely controllable power split between the output branches. A particular kind of Y coupler is the **T coupler**. The advantage of this coupler with respect to the Y one is the possibility to cascade many of them to connect multiple terminals on a network.
- **Star couplers:** these couplers include multiple inputs and outputs, not necessarily in the same number, uniformly distributing the input power among the outputs. They can be classified in *directional star couplers*, where the light collected from the inputs is mixed and then distributed between the outputs, and *nondirectional star couplers*, where all the channels are inputs and outputs at the same time, and light going into one port emerges from all the ports, including the input one.
- **1XN couplers:** structurally similar to star couplers, they split input light uniformly among N outputs. Tree couplers have been extensively used to split and mix optical

signals in CATV, LANs and all other kinds of optical communication systems.

- **Wavelength-division multiplexing couplers:** WDM consists in the use of a single fiber to combine, transmit and then separate again optical signals with different wavelengths. This technique is often used in telecommunications or when we need to interrogate multiple optical sensors working at different wavelengths with the same fiber, with an overall increase in the data rate. They are similar to *T couplers*, with the difference that in this case we have both a power and a wavelength split.

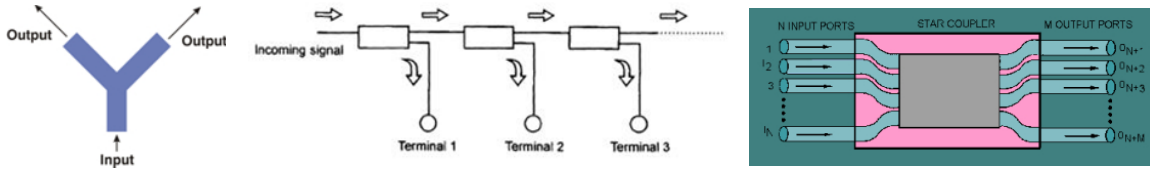


Figure 1.11: On the left a Y coupler, in the middle an example of a cascade of T couplers and on the right a representation of a directional star coupler

1.4.2. -3dB Coupler

Among all the possible couplers, directional couplers play a crucial role in light manipulation, opening up the possibility of creating more complex structures, such as interferometers, as we shall see later. As illustrated in Figure 1.12, the coupler has four terminals, which belong in pairs to the same waveguide, with the two paths completely decoupled apart from a portion of length L .

The main parameter that characterizes the guide is the coupling ratio K , whose value

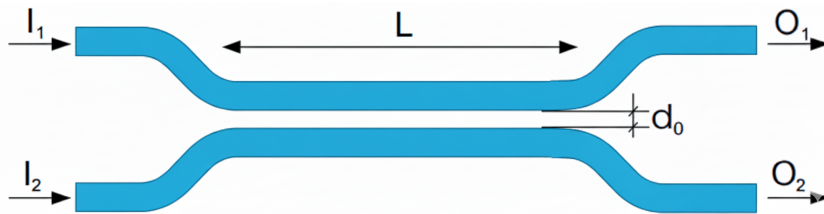


Figure 1.12: Directional coupler realized with two waveguides

depends on the interaction length L and on the technological parameters of the waveguides, such as the distance between them.

Its expression is the following:

$$K = \frac{P_2}{P_1 + P_2} \quad (1.1)$$

With P_1 and P_2 the power coming out of each branch. A common notation to indicate the power coming out from the same waveguide is P_{bar} , while the power crossing the coupler and exiting from the other branch is called P_{cross}

A common case is with the coupler specifically designed to reach a $K = 0.5$, and thus an equal power split between the two branches. The overall coupler transfer function obtained in this condition is the following:

$$T_{coupler} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix} \quad (1.2)$$

Ideally, the light crosses the coupler without losses, and the overall power is preserved between input and output, making it suitable for large photonic meshes.

1.4.3. Mach-Zehnder Interferometer

The Mach-Zehnder interferometer (MZI) is an easy filter to realize in integrated optics by cascading two -3dB couplers. The optical path connecting the couplers is usually adopted to apply a phase difference to one of the two branches, changing the interaction between the two light beams at the second coupler and so changing the overall transfer function.

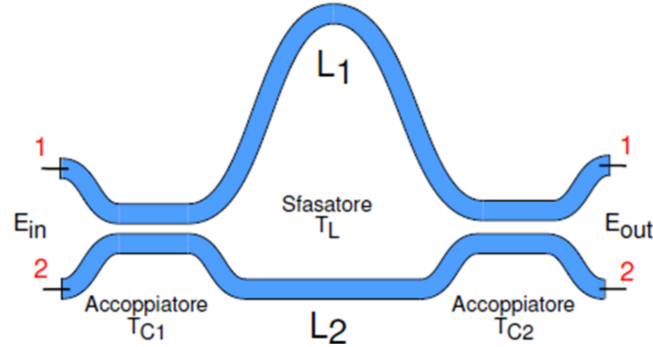


Figure 1.13: MZI with two paths of different length L_1 and L_2

Generally speaking, the MZI transfer function can be changed by including an input phase shift φ and an inner phase shift θ .

As already said before, θ models the difference in the equivalent optical path between the two couplers, a difference that can be realized by physically making one branch longer than the other or by actively changing the refractive index of one of them. On the other hand, the phase φ can represent the input phase difference of the two entering beams, or instead an active phase shifter used to create a defined phase difference between the inputs.

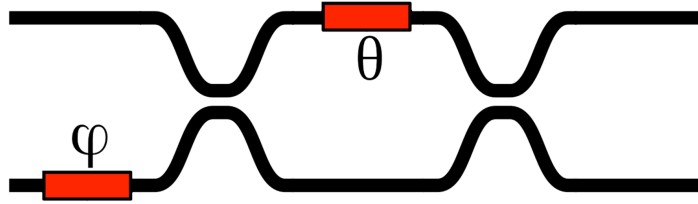


Figure 1.14: MZI with the inner phase θ and the input phase φ

The equation resulting from the cascade of the two couplers with expression (1.2), and the phase contributions represented by two diagonal matrices, is the following:

$$T_{MZI} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix} \begin{bmatrix} e^{j\theta} & 0 \\ 0 & 1 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & e^{j\varphi} \end{bmatrix} \quad (1.3)$$

Performing the product between matrices, the transfer function of the MZI with complex exponential notation is:

$$T_{MZI} = \frac{j}{2} \begin{bmatrix} -j(e^{j\theta} - 1) & e^{j\varphi}(e^{j\theta} + 1) \\ (e^{j\theta} + 1) & je^{j\varphi}(e^{j\theta} - 1) \end{bmatrix} \quad (1.4)$$

Knowing that $\frac{j(e^{j\theta}-1)}{2} = -\sin(\frac{\theta}{2})e^{j\frac{\theta}{2}}$ and $\frac{e^{j\theta}+1}{2} = \cos(\frac{\theta}{2})e^{j\frac{\theta}{2}}$, an easier sinusoidal expression can be found:

$$T_{MZI} = je^{j\frac{\theta}{2}} \begin{bmatrix} \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2})e^{j\varphi} \\ \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2})e^{j\varphi} \end{bmatrix} \quad (1.5)$$

It can be proven that this matrix preserves the modulus of a generic vector, thus achieving a unitary transformation.

2 | Basics on Optical Computing

Light is probably one of the main competitors to electronics when high-speed and long distance communications are needed [20]. Among all areas, one of the most challenging but indispensable tools in data processing is matrix-vector multiplication (MVM), whose applications range from image processing, for tasks like recognition and elaboration, to AI and numerical calculus. MVM is as indispensable as it is complex and computationally expensive, and standard electronics is facing difficulties in keeping energy and space expenditure limited. In recent years light has presented itself as one of the best competitors to electronics for data processing, with photonic accelerators that owe their recent development to undeniable advantages like speed, low power consumption and the ability to effectively implement parallel computing.

In this chapter a brief overview on the available optical architectures will be presented, with a particular focus on the spatial coherent structures based on Mach-Zehnder interferometers.

2.1. Classification of photonic architectures

Depending on the purpose of use, many optical alternatives can be suitable. The general idea is to use the amplitude and phase of the optical beams to represent complex-valued data in order to implement hardware accelerators for specific tasks.

- **Mach-Zehnder Interferometer mesh [3]:** This is one of the most common architectures for matrix multiplication, capable of realizing any unitary matrix, and consequently the transformations implemented by them. The MZI combines the light coming from two optical waveguides by controlling the phase difference between the two paths, effectively changing the interferometry.
- **Micro-Ring resonator architectures [21]:** They are based on micro-ring-shaped resonators placed next to a linear waveguide, when the guide is crossed by a specific wavelength, the light enters in resonance with the ring and is trapped. Changing the resonance point with a heater we can select the wavelength to be trapped or

how many light collect from the guide, applying a "weight" to the light. By using many of them, and with the possibility to properly tune the micro-rings, we can achieve matrix-vector multiplications. Comparing them to MZI, we can achieve a much higher computing density thanks to the smaller size of the micro-rings, along with an inferior power consumption. However, a high thermal and manufacturing precision is mandatory to achieve precise control of these structures.

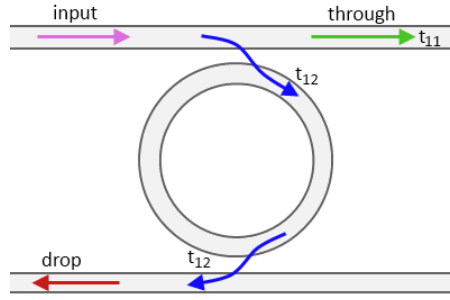


Figure 2.1: Example of a micro-ring resonator with the above through guide and the bottom drop guide

- **Wavelength Division Multiplexing (WDM) structures [21]:** This structure bases its advantage in the parallelization of computation by means of a simultaneous use of multiple wavelengths, with a photonic computing core than can be based both on MZI or micro-rings, increasing the overall throughput in a way proportional to the number of wavelengths used. To exploit this technique, an optical multiplexer, before the computing mesh, and a demultiplexer at the end are needed to convoy the different wavelengths in the same channel and then divide them to gather the information from each of them. The choice of material for the optical guide is also of primary importance, as it must have very low absorption at all wavelengths of interest.
- **Free-space optics computing [22]:** They rely on free space propagation of light through some mediums like optical films or lenses in order to achieve specific operations. The first examples date back to 1960, when the basic principles and architectures performing specific mathematical operations were outlined [23]. A simple architecture can be realized by superimposing two optical planes P_1 and P_2 , with their own 2-D spatial transmissions $T_1(x, y)$ and $T_2(x, y)$, to achieve an overall transfer function $I(x, y) = I_0 T_1(x, y) T_2(x, y)$. It is then straightforward to see that by doing this we can perform element-wise multiplications.
- **Time-domain based architectures [24]:** Time Domain Photonic Computing is

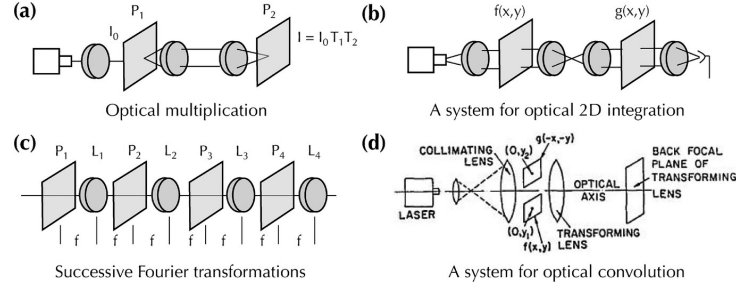


Figure 2.2: (a) Multiplication by means of two optical planes (b) Bi-dimensional optical integration (c) Successive Fourier transformations (d) Optical convolution in two dimensions

an emerging computation paradigm that leverages the temporal properties of light pulses to encode and process information. This approach uses the shape, duration or timing of light pulses to model a mathematical problem. The advantages of using this approach is to convert multi-dimensional data, like an image or a video, into a one-dimensional data flow, exploiting the characterizing aspects of a signal over time to represent the different dimensions. The advantage of this methodology lies in the ability to compute a large amount of data, such as the processing of a video or of a large image data set, without the need to have too many channels in parallel and therefore reducing the space needed to build a photonic chip.

One of the most promising applications is the *Reservoir Computer*, a type of recurrent neural network where a loop is realized with an optical fiber and a delay line. The light recirculating in the loop interferes with a new version of itself, creating a more complex output. This structure has proven to perform much better than electronic architectures for tasks such as speech recognition and prediction of chaotic time-series [25].

2.2. Photonic Coherent Architectures

Among all photonic architectures, coherent ones rely on single wavelength sources, like lasers [26]. Coherent architectures have their main advantage in the possibility to use for the calculation purposes not only the power, which is the exploitable property in incoherent beams, but also the phase of the incoming light. This opens up the possibility of realizing complex transformations, allowing for a higher number of possible implementable operations. Coherent structures also open up the possibility to use multiple wavelengths in parallel in a controlled manner through WDM, increasing the overall achievable transfer bandwidth.

Most photonic circuits are implemented for a specific task and, for this reason, are called

ASPIC (Application Specific Photonic Integrated Circuits). Instead, a reprogrammable implementation which changes at run-time how light interacts and recombines along the optical path, can implement many different linear functions.

The basic 2x2 block takes the light from two input waveguides and returns an output given by a linear combination of the two optical beams. One of the most common implementations for 2x2 gates is the MZI. This structure, whose transfer function has already been illustrated in the previous chapter (3.1), displays an adjustable output split and phase difference depending on two programmable actuators.

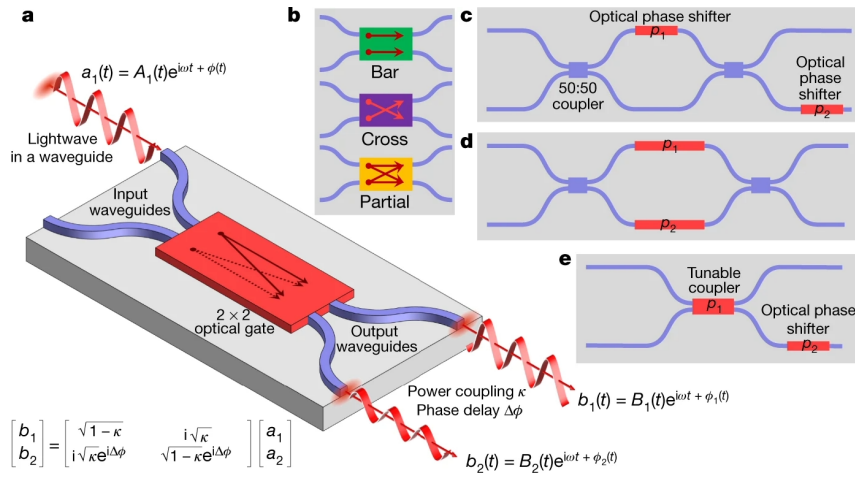


Figure 2.3: (a) 2x2 Optical gate mixing the two optical waves a_1 and a_2 (b) Possible configurations of the light through the coupler (c,d,e) Two degrees of freedom p_1 and p_2 to control output phase and amplitude, tuning the coupling ratio or the phase shift.[27]

We will focus on MZI meshes, based on 2x2 basic blocks which implement an overall transfer function determined by how the MZIs are arranged. An easily expandable and configurable control system is then mandatory to correctly set the working points along different meshes, without drastic changes in the overall control scheme. In this thesis, an integral control is implemented to correctly set a 3x3 matrix implemented with 3 MZIs, laying the foundation for a subsequent expansion to higher order matrices keeping the complexity comparable.

It can be seen that, theoretically, if the elements are lossless or with negligible losses, the obtained transfer function of the single MZI, and therefore also of the overall matrix, is unitary. In general, a complex square matrix U is unitary when its inverse is equal to its transposed conjugate. Calling the inverse U^{-1} , the conjugate transposed U^* and I the identity matrix, the following relation holds:

$$U^*U = UU^* = U^{-1}U = UU^{-1} = I \quad (2.1)$$

Unitary transformations find application in many scientific fields, from telecommunications [28] to physics [29], because of their intrinsic property to preserve the norm of a vector and the scalar product between vectors, thus applying a rotation in the complex plane without altering the energetic properties of the vector itself.

When the matrix of interest is not unitary, *singular value decomposition (SVD)* comes into play. SVD is the decomposition of a complex matrix into three new matrices, two of which are unitary and the third one is diagonal. Considering a general A matrix, it can be written through SVD as:

$$A = V\Sigma U^* \quad (2.2)$$

Where V and U^* , which is the transposed conjugate of U , are unitary matrices, Σ is a diagonal matrix and A is the matrix we want to decompose. Σ and U are obtained by computing the eigenvalues and eigenvectors of A^*A . The singular values of A^*A , that are defined as $\sigma_i = \sqrt{\lambda_i}$, where λ_i are the eigenvalues, constitute the diagonal of the matrix Σ . V is instead obtained by using the eigenvectors of A^*A as columns (right-singular vectors) of the matrix, ordered in the same way as the singular values in Σ . The columns of V (left-singular vectors), can be obtained by inverting equation (2.2), or directly by computing the eigenvectors of AA^* and arranging them according to the same criteria as before as columns of the matrix.

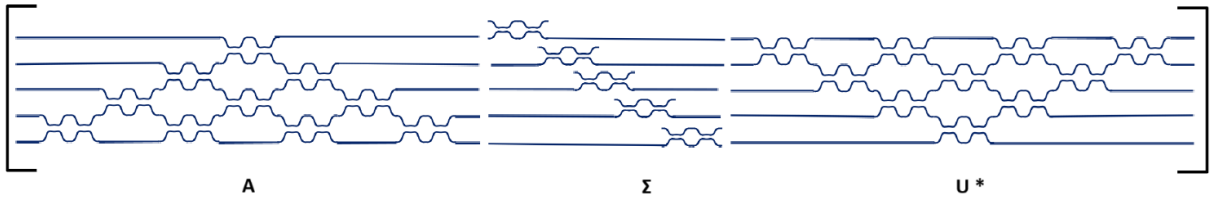


Figure 2.4: Optical implementation of the SVD, with A and U^* meshes implementing unitary matrices, and the diagonal matrix Σ implemented by the structure in the middle.

To optically implement SVD, we use two MZI meshes implementing A and U^* , while for the diagonal matrix Σ , MZIs are introduced along the optical path to remove a portion of the flowing light, thus allowing the implementation of non-unitary diagonal matrices. It is important to be noticed that, in order to improve symmetry as much as possible, the two unitary matrices are implemented as an upper and a lower triangular mesh, in this way considering a linear path from entrance to exit the light passes through the same number of MZI.

2.3. Mesh composition

To implement an arbitrary $N \times N$ unitary matrix, as previously anticipated, a mesh made of 2×2 building blocks has to be realized. Up to now, the design rules proposed by *Reck et al.* and by *Clements et al.* represent the two cornerstones of reconfigurable coherent photonic computing.

2.3.1. Reck architecture

As addressed by Michael Reck and Anton Zeilinger in 1994 [7], any $N \times N$ unitary matrix describing a lossless transformation between N inputs and N outputs, can be built by a sequence of simpler two-dimensional transformations. The 2×2 transformation can be implemented with a variable reflectivity splitter or by using a MZI with 50:50 symmetric beam splitters and phase shifters. The algorithm works recursively, similar to Gaussian elimination. It applies these 2×2 transformations sequentially to "zero out" the off-diagonal elements of the matrix, reducing the effective dimension of the problem from N to $N-1$, then to $N-2$, and so on. The physical implementation of this algorithm naturally results in a triangular array of beam splitters. With this recursive implementation, the number of MZIs needed is $\frac{N(N-1)}{2}$, showing a trend increasingly similar to a quadratic function, the more the order of the matrix increases.

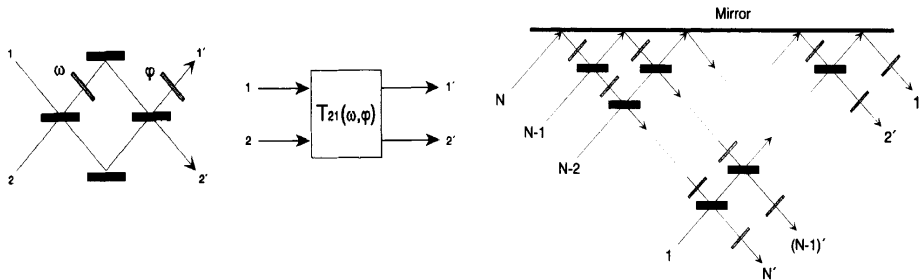


Figure 2.5: On the left a MZI realized with two 50:50 beam splitters, two mirrors and two phase shifters to properly tune the transfer function. In the middle a block representation of a MZI. On the right a triangular array of beam splitters implementing an $N \times N$ unitary matrix as an optical multiport.[7]

Once the order of the transfer function is defined, the complete unitary matrix is simply the composition of smaller unitary matrices following the design rules mentioned above. An example will be presented using a 4×4 matrix implemented by a triangular mesh. As can be seen, the matrix was divided into smaller sections U_n with $n \in (1, \dots, 4)$, so that each block is the cascade (and so the product) of single MZI transfer functions extended

to order N .

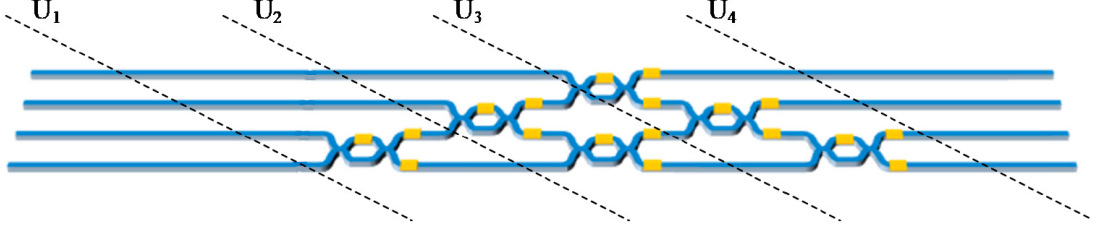


Figure 2.6: A 4×4 triangular mesh divided in such a way to result the product of extended 2×2 matrices

By calling $R_{i,j}$ the extension of the two-dimensional matrix to a generic dimension N , and U_n the unitary matrix at the n_{th} stage, it is possible to build the overall transfer function as follows:

$$U_1 = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}$$

$$U_2 = R_{1,1}U_1 = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & R_{1,1} & \\ & & & 1 \end{bmatrix} U_1$$

$$U_3 = R_{2,1}R_{2,2}U_2 = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & R_{2,1} & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & R_{2,2} & & \\ & & 1 & \\ & & & 1 \end{bmatrix} U_2$$

$$U_4 = R_{3,1}R_{3,2}R_{3,3}U_3 = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & R_{3,1} & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & R_{3,2} & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} R_{3,3} & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} U_3$$

Substituting then U_3 and U_2 in the previous equations, we finally get:

$$U_4 = R_{3,1}R_{3,2}R_{3,3}R_{2,1}R_{2,2}R_{1,1} \quad (2.3)$$

As can be easily deduced, the general expression for an N order transformation will be:

$$U(N) = R_{N-1,1}R_{N-1,2} \dots R_{N-1,N-1} \dots R_{2,1}R_{2,2}R_{1,1} \quad (2.4)$$

2.3.2. Clements architecture

This architecture [30] was proposed in 2016 to outperform that by Reck et al. in the design of universal multiport interferometers. A new nulling algorithm was introduced, with the objective to decompose a unitary matrix U in the product of $T_{m,n}$ matrices following the explicit algorithm:

$$U = D(\Pi_{(m,n) \in S} T_{m,n}) \quad (2.5)$$

Where S defines a specific ordered sequence of two-mode transformations, and where D is a diagonal matrix with complex elements with modulus equal to one on the diagonal. The algorithm follows a diagonal zeroing scheme, starting from the lower left corner up to the main diagonal of the matrix. Every zeroed element corresponds to a beam splitter or an MZI in the matrix. This new implementation corresponds no more to a triangular shape, but to a more compact square structure that takes up about half the space of the Reck counterpart. This change in the structure results in an overall improved compactness, with an optical depth (number of beam splitters or interferometers crossed by the light along its path) reduced from $2N-3$ to N , marking a decisive improvement over previous architectures for large matrices. In addition to this, the RMS error introduced by components non-ideality is more or less the same, but with the advantage to have it uniformly distributed among all the output ports, making the circuit more predictable and reliable.

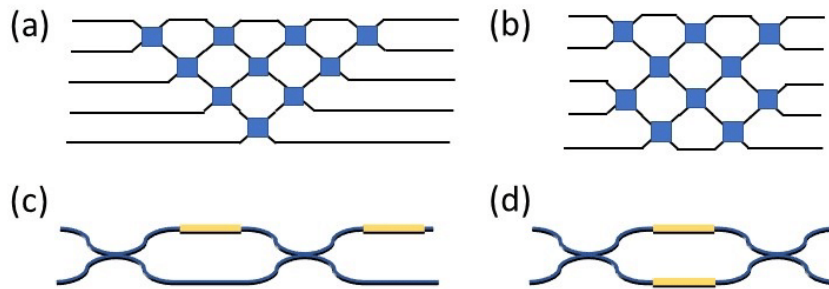


Figure 2.7: (a)Reck scheme with the blue dots representing tunable beam and phase shifters (b)Clements implementation of the same structure (c)Asymmetric beam splitter with one internal and one external shifter (d)MZI with two internal phase shifters for improved symmetry and compactness

Overall, the Clements architecture shows a lower area occupation, with a size of about 50 percent compared to that of Reck, and at the same time maintains a very high fidelity even with high internal losses and high order matrices. However, due to its asymmetric

structure, the fidelity of Reck design rapidly drops as component losses and circuit size increases. In terms of noise tolerance, the two designs are very similar, with fidelity that decreases proportionally to the number of noisy elements, which is the same in both designs ($\frac{N(N-1)}{2}$)[31].

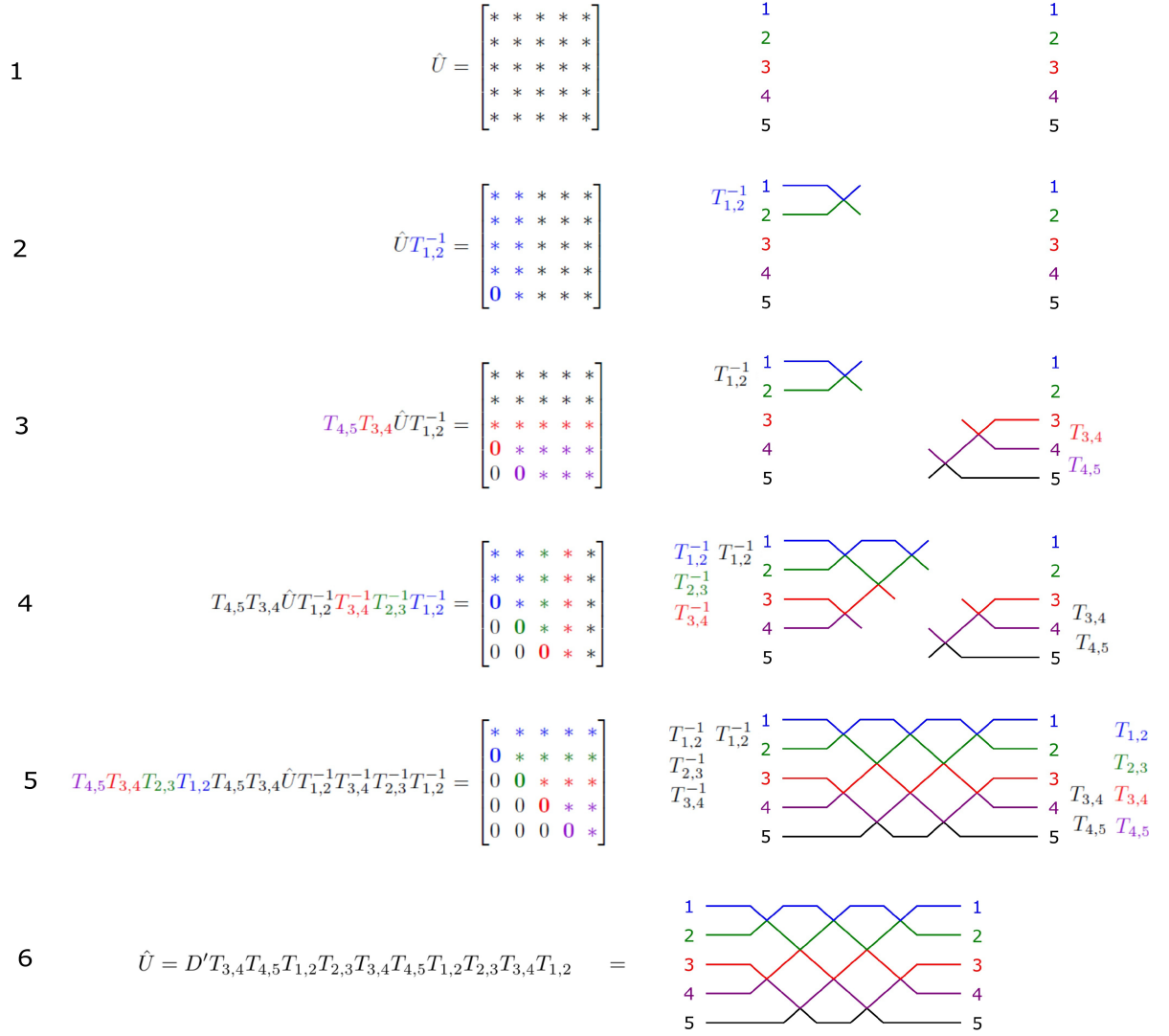


Figure 2.8: Incremental steps (1-6) to achieve the overall diagonal zeroing illustrated by Clements et al. showing how the mesh is populated following the cancellation of the matrix values.[30]

Apart from the above-mentioned general-purpose designs for unitary matrices, *Fast architectures* are structures specifically designed to compute fast Fourier transforms and Sylvester's transforms. These highly specialized structures are based on an optical implementation of the fast Fourier decomposition by Barak et al.[32]. It has been proven that these highly specialized structures are the best option when the application allows their

use[31], offering better performance both in terms of internal losses and in terms of noise.

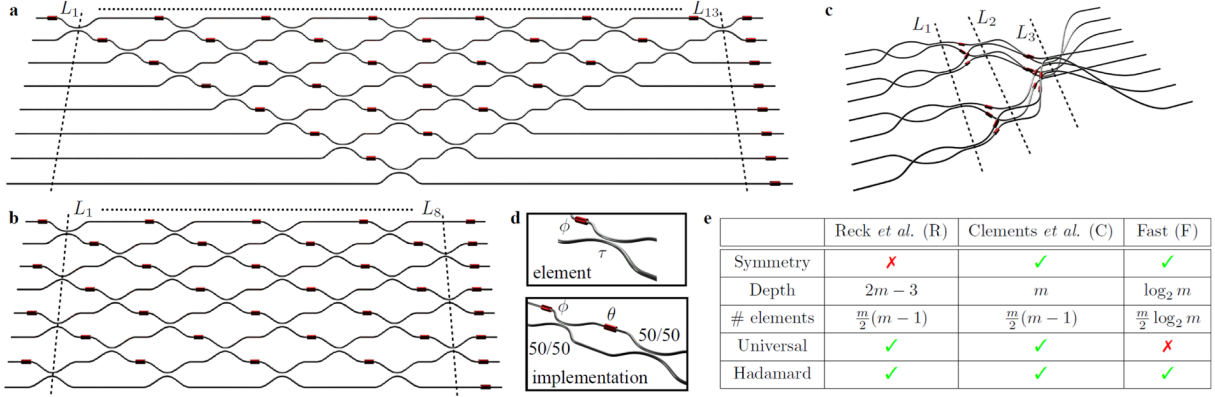


Figure 2.9: (a) 8×8 matrix implementation with Reck architecture (b) 8×8 matrix implementation with Clements architecture (c) Fast architecture implementation with 8 inputs, designed to solve specific problems like Fast Fourier Transform (FFT), Quantum Fourier transform (QFT) and Sylvester's transformations. (d) Unit cells common in the three layouts (e) Comparison table among the three architectures [31]

3 | Matrix setting and control

An optical matrix implementation seems to be promising, but a way to program it with precision and to reprogram it whenever we need must be implemented. An electro-optic cooperation is mandatory to achieve the task. A reprogrammable optical mesh designed following one of the previously mentioned architectures (2.3), or similar architectures suitable for the task, is adopted as the core of the structure where the actual calculation takes place. To set the desired coefficients of the overall obtained transfer function we use actuators (1.2), being them electro-optical, MEMS or thermo-optical, in order to set the desired phase shifts along the mesh.

Up to now the process was straightforward, once chosen the desired transfer function, the needed phase shifts can be obtained by matching the expected output with the obtained ones, calibrating the optical chip. To keep the results reliable and comparable over time, the optical chip needs to be thermostatically stabilized, reducing as far as possible temperature variations which would invalidate the calibration. This long calibration process has to be done for each optical chip, because of optical non-uniformities and process variations among them, and this requires lot of time, a large and reliable thermal stabilizer, and can lead to precision losses over time.

Another calibration strategy for interferometer-based photonic meshes was proposed by David A. B. Miller [33]. Instead of setting each working point by means of calculations, which can lead to the wrong matrix because of mesh non-idealities and would require increasingly higher computational power as the size increases. The method proceeds with a progressive recursive training, feeding the mesh the expected output and input vectors and relying on local feedback loops to make the different points of the matrix converge where we want. The bi-directionality of the process wouldn't be a problem for MZI meshes, that can be crossed by light in both directions, the problem lies instead in the necessity to shine and read an optical signal back both at the input and at the output, which would place stringent limits on how the setup can be done.

The approach chosen in this thesis was to use an automated process, with multiple local feedback to sense and control various crucial points along the mesh, thus acting to converge the mesh working points towards the desired matrix. As will be deeply explained

later, this approach is particularly focused on scalability, keeping the base structure of the feedbacks unchanged with an increase of system order, because of the "local nature" of the feedback itself, thus increasing the number of feedback but keeping the same complexity. The feedbacks are made possible by transparent photodetectors placed along the optical path to sense light in a lossless way. The power datas are then collected and processed by an FPGA-based electronic chain, which acts on heaters to properly set the correct phase shifts [6]. This chapter will be focused on an overview of the setting and control of the components along the optical path, starting with 2×2 basic block, implemented by a single MZI, and then extending the concept to a larger 3×3 matrix implemented by 3 MZI.

3.1. MZI setting

To correctly control the system, a proper knowledge of the building blocks of the mesh is mandatory, thus a deep analysis of the Mach Zehnder interferometer as a function of its two run-time controllable parameters will be performed.

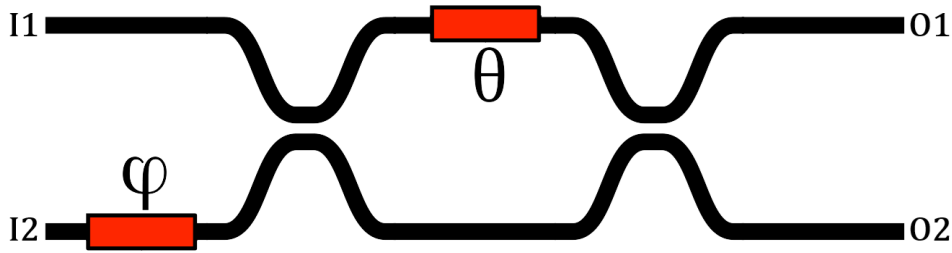


Figure 3.1: Mach Zehnder interferometer implementation, with the two phase-shifter actuators modeled by a φ phase, compensating the input branch of the interferometer, and a θ shifter in between the two couplers.

$$T_{MZI} = j e^{j\frac{\theta}{2}} \begin{bmatrix} \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) e^{j\varphi} \\ \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) e^{j\varphi} \end{bmatrix}$$

Given then 2 generic inputs In_1 and In_2 , the two resulting outputs Out_1 and Out_2 will be:

$$T_{MZI} \times \begin{bmatrix} In_1 \\ In_2 \end{bmatrix} = \begin{bmatrix} Out_1 \\ Out_2 \end{bmatrix} \quad (3.1)$$

$$\begin{bmatrix} Out_1 \\ Out_2 \end{bmatrix} = j e^{j\frac{\theta}{2}} \begin{bmatrix} \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) e^{j\varphi} \\ \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) e^{j\varphi} \end{bmatrix} \times \begin{bmatrix} In_1 \\ In_2 \end{bmatrix} \quad (3.2)$$

The T_{MZI} matrix can also be written in the exponential form as previously shown (1.4.3), the use of a form or the other will be useful for calculations in different scenarios. Being complex numbers, is important to evaluate both phase and amplitude to find the correct values for each position of the matrix and avoid ambiguity, but also to understand how different signal interacts when we combine multiple MZIs together to form a mesh.

3.1.1. Setting point evaluation

We can start computing the output power of a MZI, starting from the general transfer function and considering a generic input vector with an ideally unknown phase shift between the two inputs that we will call λ :

$$\begin{bmatrix} Out_1 \\ Out_2 \end{bmatrix} = j e^{j\frac{\theta}{2}} \begin{bmatrix} \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2})e^{j\varphi} \\ \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2})e^{j\varphi} \end{bmatrix} \times \begin{bmatrix} In_1 \\ In_2 e^{j\lambda} \end{bmatrix}$$

$$Out_1 = j e^{j\frac{\theta}{2}} \left(\sin(\frac{\theta}{2}) In_1 + e^{j\varphi} \cos(\frac{\theta}{2}) In_2 e^{j\lambda} \right) \quad (3.3)$$

$$Out_2 = j e^{j\frac{\theta}{2}} \left(\cos(\frac{\theta}{2}) In_1 - e^{j\varphi} \sin(\frac{\theta}{2}) In_2 e^{j\lambda} \right) \quad (3.4)$$

The output power is then proportional to the amplitude squared of the output values:

$$|O_1|^2 = |j e^{j\frac{\theta}{2}} \left(\sin(\frac{\theta}{2}) In_1 + e^{j(\varphi+\lambda)} \cos(\frac{\theta}{2}) In_2 \right)|^2$$

$$|O_2|^2 = |j e^{j\frac{\theta}{2}} \left(\cos(\frac{\theta}{2}) In_1 - e^{j(\varphi+\lambda)} \sin(\frac{\theta}{2}) In_2 \right)|^2$$

The overall calculation will be performed only for one of the 2 outputs, it has to be noticed that the common phase term can be neglected for the amplitude computation:

$$\begin{aligned} |O_1^2| &= \left| \sin(\frac{\theta}{2}) In_1 + e^{j(\varphi+\lambda)} \cos(\frac{\theta}{2}) In_2 \right|^2 \\ &= \left| \sin(\frac{\theta}{2}) In_1 + [\cos(\lambda + \theta) + j \sin(\lambda + \varphi)] \cos(\frac{\theta}{2}) In_2 \right|^2 \\ &= \left| \sin(\frac{\theta}{2}) In_1 + \cos(\lambda + \varphi) \cos(\frac{\theta}{2}) In_2 + j \sin(\lambda + \varphi) \cos(\frac{\theta}{2}) In_2 \right|^2 \\ &= \sin^2(\frac{\theta}{2}) In_1^2 + \cos^2(\frac{\theta}{2}) In_2^2 + 2 \sin(\theta) \cos(\lambda + \varphi) In_1 In_2 \end{aligned} \quad (3.5)$$

The other output computation follows the same approach, with an overall result that differs from the previous one only for the sign of the double product.

$$PO_1 \propto |O_1|^2 = \sin^2\left(\frac{\theta}{2}\right)I_{in1}^2 + \cos^2\left(\frac{\theta}{2}\right)I_{in2}^2 + \sin(\theta)\cos(\varphi)I_{in1}I_{in2} \quad (3.6)$$

$$PO_2 \propto |O_2|^2 = \cos^2\left(\frac{\theta}{2}\right)I_{in1}^2 + \sin^2\left(\frac{\theta}{2}\right)I_{in2}^2 - \sin(\theta)\cos(\varphi)I_{in1}I_{in2} \quad (3.7)$$

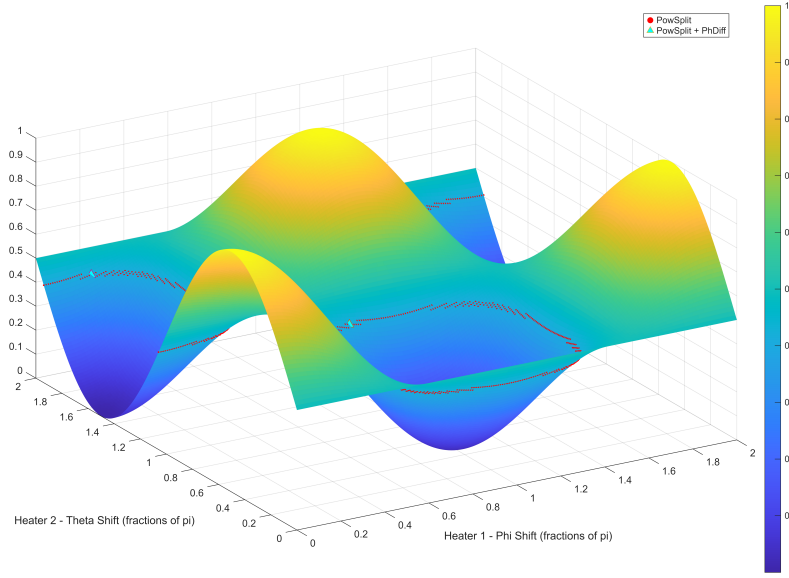


Figure 3.2: 3D map of MZI output power split. The red lines shows all the points with the desired power split, while the blue triangles mark the points where we have both the desired phase shift and power split.

It can be observed how the overall output power doesn't change once fixed θ and φ , the power is only redistributed between the two outputs of the interferometer.

Proceeding then to set the desired working points means choosing a determined combination of phase and power because of the complex nature of the transfer function. Power information is easy to be retrieved, by directly measuring it on-chip along the optical path by means of lossless transparent photodiodes. For phase instead we would have to rely on specific structures, whose insertion in the mesh cannot be done at will and without problems in any point, and usually are present at the end of the mesh. Will be shown later how MZIs configured in a specific way can be used as phase meters, despite the fact that they cannot be used simultaneously for optical computing and phase measurement purposes. Consider now to want to set a certain working point. The information available along the optical path is usually only the one obtainable from the transparent photodetectors, which is the power of the light crossing the sensors. We can then choose a specific

power split for the MZI and set the feedback to converge at that point, but a problem arises. Only using power as a variable, our system has almost infinite different point of convergence, but also adding the phase information, and considering the points for which both power and phase correspond, we get two possible solutions for each 2π by 2π square.

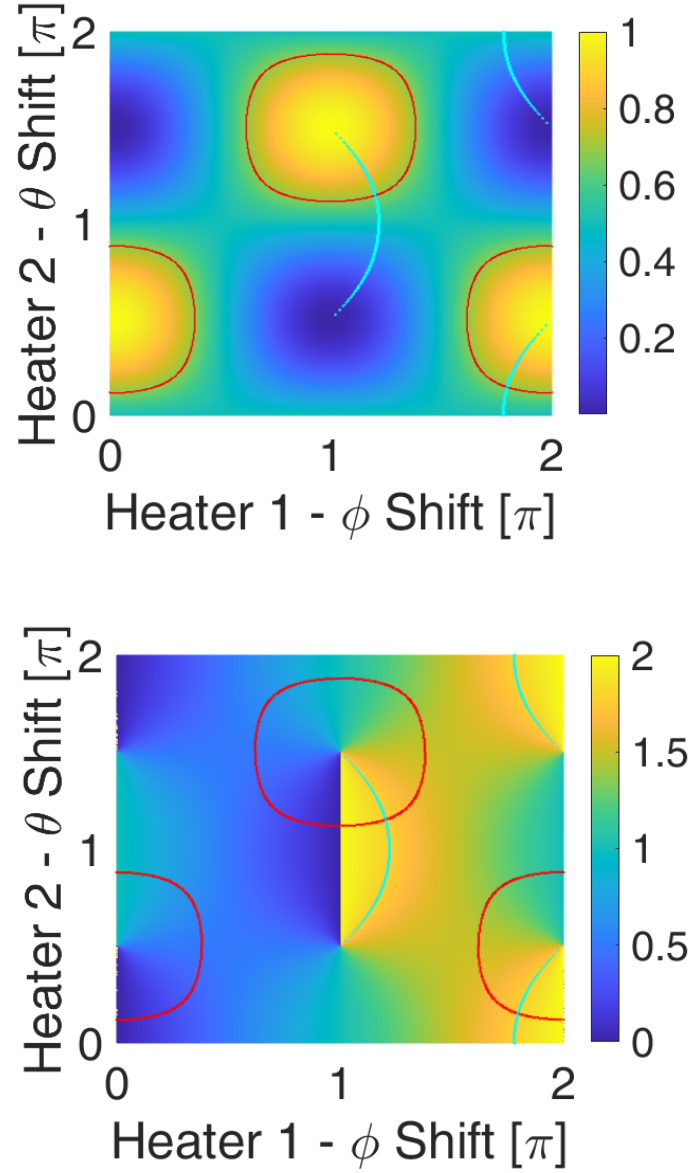


Figure 3.3: On top, overview of the power map along one period for both heaters. On the bottom the same representation but for the phase difference between the two output branches. The red line represents the points meeting the power of interest, while the blue line indicate the points with the searched phase. It can be seen that even using both information we still get two possible points to which our feedback can converge in a single period.

Other than phase and power, what can help us to distinguish between the different points is the local derivative in a neighborhood of points of interest. This means dividing the problem along the 2 phase-variables, moving from a three-dimensional problem to a two-dimensional one, and use the derivative sign along the 2 axis for the θ and the φ variable around the working points of interest, to differentiate them. We are then considering the MZI as the combination of two couplers in cascade, thus removing the need to control the MZI in its entirety and controlling each coupler by itself, checking that after each stage power split and derivative are the expected ones[6].

Computing the power split after the first coupler, the following result is obtained:

$$\begin{bmatrix} Out_1 \\ Out_2 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & e^{j\varphi} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 e^{j\xi} \end{bmatrix}$$

$$\begin{bmatrix} Out_1 \\ Out_2 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} I_1 & j e^{j(\varphi+\xi)} I_2 \\ j I_1 & e^{j(\varphi+\xi)} I_2 \end{bmatrix}$$

Lets now consider the power split on the second output for reference, the second output results in:

$$Out_2 = \frac{1}{\sqrt{2}} (j I_1 + e^{j(\varphi+\xi)} I_2)$$

And by then computing the squared modulus, to get an equation proportional to the power, we get:

$$\begin{aligned} P_2 \propto |Out_2|^2 &= \frac{1}{2} [I_2^2 \cos^2(\varphi + \xi) + I_1^2 + \sin^2(\varphi + \xi) I_2^2 + 2 I_1 I_2 \sin(\varphi + \xi)] \\ &= \frac{1}{2} [I_1^2 + I_2^2 + 2 I_1 I_2 \sin(\varphi + \xi)] \end{aligned} \quad (3.8)$$

Now, by considering $P_{tot} \propto Out_{1MZI}^2 + Out_{2MZI}^2$, where Out_{1MZI} and Out_{2MZI} are the outputs of the MZI after the 2 couplers, we can normalize the previously calculated P_2 [3.1.1] with respect to the overall power $P_{norm2} = \frac{Out_2^2}{P_{tot}}$. By now assuming power conservation across the device, given the unitary nature of the interferometer, we get $Out_{1MZI}^2 + Out_{2MZI}^2 = In_1^2 + In_2^2$ thus resulting in:

$$\begin{aligned} P_{norm2} &= \frac{Out_2^2}{P_{tot}} \\ &= \frac{\frac{1}{2} [I_1^2 + I_2^2 + 2 I_1 I_2 \sin(\varphi + \xi)]}{\frac{1}{2} [In_1^2 + In_2^2]} \\ &= \frac{1}{2} + \frac{I_1 I_2}{I_1^2 + I_2^2} \sin(\varphi + \xi) \end{aligned} \quad (3.9)$$

By now fixing the input I_1 and I_2 values, the power behavior is that of a sinusoid shifted upwards by an offset of $1/2$ and repeats periodically every 2π , and being a sinusoid, for each period we get two different points with the same power split. As previously anticipated, now we have two alternatives to get the correct set point, by computing the phase between the two branches or by considering the derivative in the two possible solutions. As previously mentioned, phase is generally hard to be retrieved along the optical path, so we will proceed by exploiting the information on the derivative.

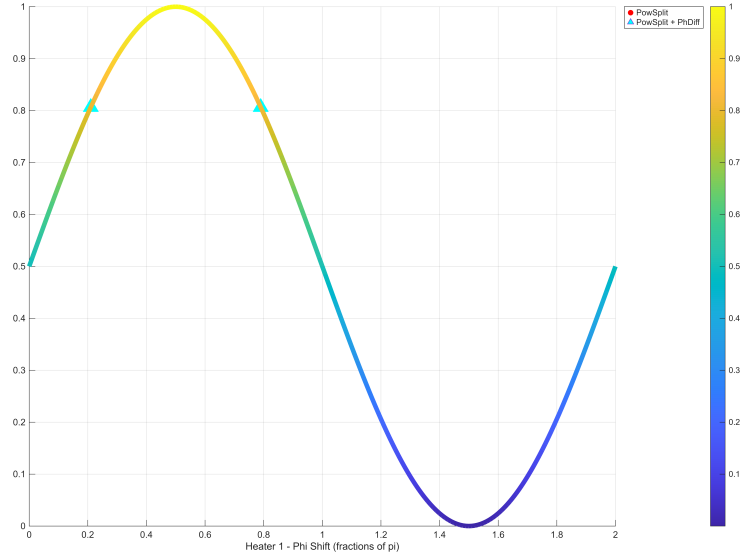


Figure 3.4: On the left a phase simulation of a MZI output phase difference, on the output power split. Both simulations were performed along 2 periods.

Let's consider the example depicted in the picture, for the first coupler and with a 0.8 power split among the outputs that is obtainable for two ϕ values, 0.2π or 0.8π . Computing the derivative we get:

$$\frac{dP_{norm2}(\phi)}{d\phi} = \frac{I_1 I_2}{I_1^2 + I_2^2} \cos(\phi + \xi) \quad (3.10)$$

Assuming for example $\xi = 0$, and calling the constant (once fixed the inputs) relationship $\frac{I_1 I_2}{I_1^2 + I_2^2} = C$. In the two cases we get $\frac{dP_{norm2}(0.2)}{d\phi} = 0.809C$ and $\frac{dP_{norm2}(0.8)}{d\phi} = -0.809C$ with a derivative, as expected, equal but opposite between the two points.

The system, overall, from a desired MZI transfer function splits it in the two couplers, choosing the correct working point for each stage along with the phase shift to be implemented on each heater. The derivative is then computed in a neighborhood of the corresponding setpoint for both the phase axis, and the sign of the control loop along with it, to allow convergence only at the desired points.

3.2. 3x3 MZI implementation

Up to now we analyzed how to properly implement a desired working point for a single MZI, which corresponds to a 2×2 matrix, by exploiting the power split for each coupler and the derivative sign for the two phase shifters. To further investigate control with larger and more scientifically useful meshes, a 3×3 mesh, implemented with Reck architecture, will be realized and analyzed. The generic number of MZIs needed to implement a given matrix is $\frac{N(N-1)}{2}$ in this architecture, thus resulting in the necessity to control 3 different interferometers.

The base structure of the control is kept almost equal for the single MZI, splitting the interferometer in the two couplers and controlling each working point separately. In this case the precision with which each device is set influences the maximum achievable precision along the mesh, with a result that we expect to be more discordant the further the optical path we travel. The feedback must therefore be able to perfectly control each device and to know in real time the state of each point of the mesh.

Considering only the matrix structure, 9 transparent photodiodes and 6 phase shifters are placed along the mesh. They are placed along the mesh following the same structure adopted for the 2×2 , this because of the local nature of the feedback. Each coupler is controlled by its own, and thus being independent of power losses along the path because the control sets the phase shifts accounting for the signal reaching each stage. It is important to note that the control complexity remains unchanged as the MZI increases, since the same structure is replicated multiple times, promoting the scalability of the adopted system.

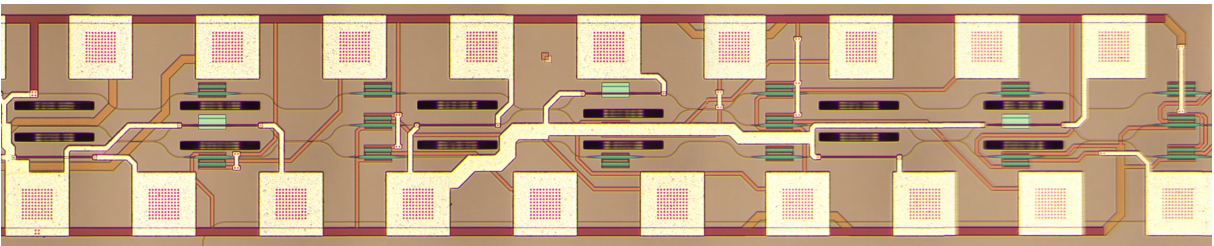


Figure 3.5: Picture of the actual implementation of the 3×3 structure. Can be seen the actuators (heaters), i.e. the purple structures along the optical guides, and the transparent photodiodes along with the dummy photodiodes recognizable where the guide widens and takes on a bluish color.

Because of the necessity to recreate the 3 inputs, a specific structure has been realized to allow a proper and feasible test of the mesh. This structure is programmable and we can

use it to choose arbitrarily the power split among the 3 inputs of the 3×3 matrix.

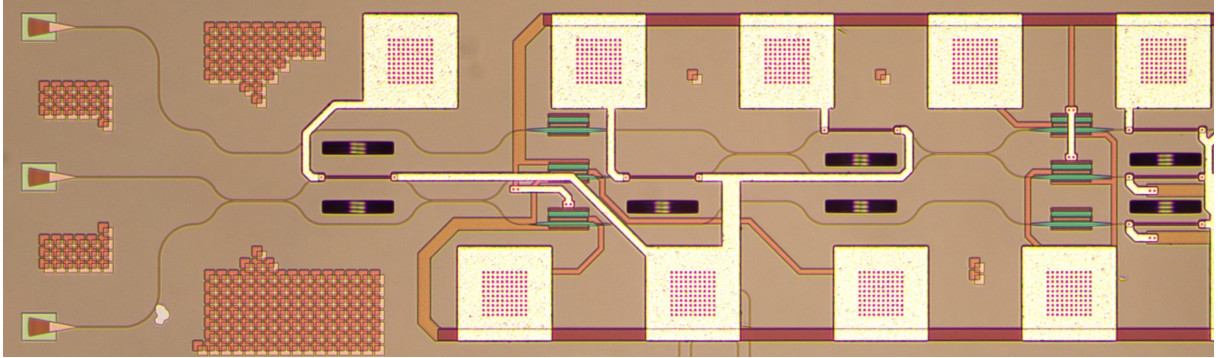


Figure 3.6: Input vector generator. By means of it we can generate any 3×1 vector in power starting from a single input

To allow than a measurement both in power and in phase at the output of our matrix, a 3-inputs phase meter was inserted in order to retrieve all the information possible, and obtain a full complex description of the output vector. It is a diagonal Mach-Zehnder structure that relies on germanium photodiodes to detect the light at the end of the optical path.

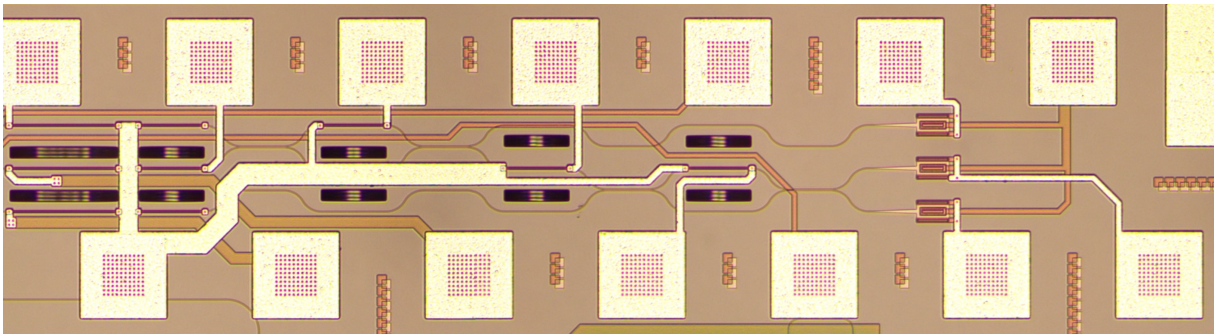


Figure 3.7: Diagonal MZIs structure with output photodiodes made out of germanium, designed for coherent light phase measurements.

3.2.1. 3x3 Mesh transfer function

Starting from the 2×2 transfer function, the 3×3 is obtained by a proper combination of them following a triangular structure accordingly to Reck architecture.

$$T_{3 \times 3} = T_{2 \times 2 \text{ low}} T_{2 \times 2 \text{ high}} T_{2 \times 2 \text{ low}}$$

Where $T_{2 \times 2 \text{ low}}$ is the generic MZI matrix extended to a 3×3 dimension considering a unitary transfer for the element in position $(1; 1)$ in the input vector

$$T_{2 \times 2 \text{ low}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{e^{j\theta}-1}{2} & j e^{j\varphi} \frac{e^{j\theta}+1}{2} \\ 0 & j \frac{e^{j\theta}+1}{2} & e^{j\varphi} \frac{1-e^{j\theta}}{2} \end{bmatrix}$$

And $T_{2 \times 2 \text{ high}}$ is, as before, an extension of the standard T_{MZI} but instead it multiplies by 1 the element in position $(3; 1)$

$$T_{2 \times 2 \text{ high}} = \begin{bmatrix} \frac{e^{j\theta}-1}{2} & j e^{j\varphi} \frac{e^{j\theta}+1}{2} & 0 \\ j \frac{e^{j\theta}+1}{2} & e^{j\varphi} \frac{1-e^{j\theta}}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

By then putting them together, the overall transfer function results in:

$$T_{3 \times 3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{e^{j\theta_1}-1}{2} & j e^{j\varphi_1} \frac{e^{j\theta_1}+1}{2} \\ 0 & j \frac{e^{j\theta_1}+1}{2} & e^{j\varphi_1} \frac{1-e^{j\theta_1}}{2} \end{bmatrix} \begin{bmatrix} \frac{e^{j\theta_2}-1}{2} & j e^{j\varphi_2} \frac{e^{j\theta_2}+1}{2} & 0 \\ j \frac{e^{j\theta_2}+1}{2} & e^{j\varphi_2} \frac{1-e^{j\theta_2}}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{e^{j\theta_3}-1}{2} & j e^{j\varphi_3} \frac{e^{j\theta_3}+1}{2} \\ 0 & j \frac{e^{j\theta_3}+1}{2} & e^{j\varphi_3} \frac{1-e^{j\theta_3}}{2} \end{bmatrix} \quad (3.11)$$

It can be clearly seen that simulating an outcome is not as simple as it would be for a single MZI, we have a total of 6 variables and a visual representation of all possible outcomes is not feasible. In this thesis we won't address the problem of linking a 3×3 matrix with the 3 base 2×2 matrices, procedure feasible through nulling algorithms, we will instead set a desired phase to the heaters, and we will check the accuracy of the matrix resulting by this setting with respect to the theoretical one we would have with the same phases.

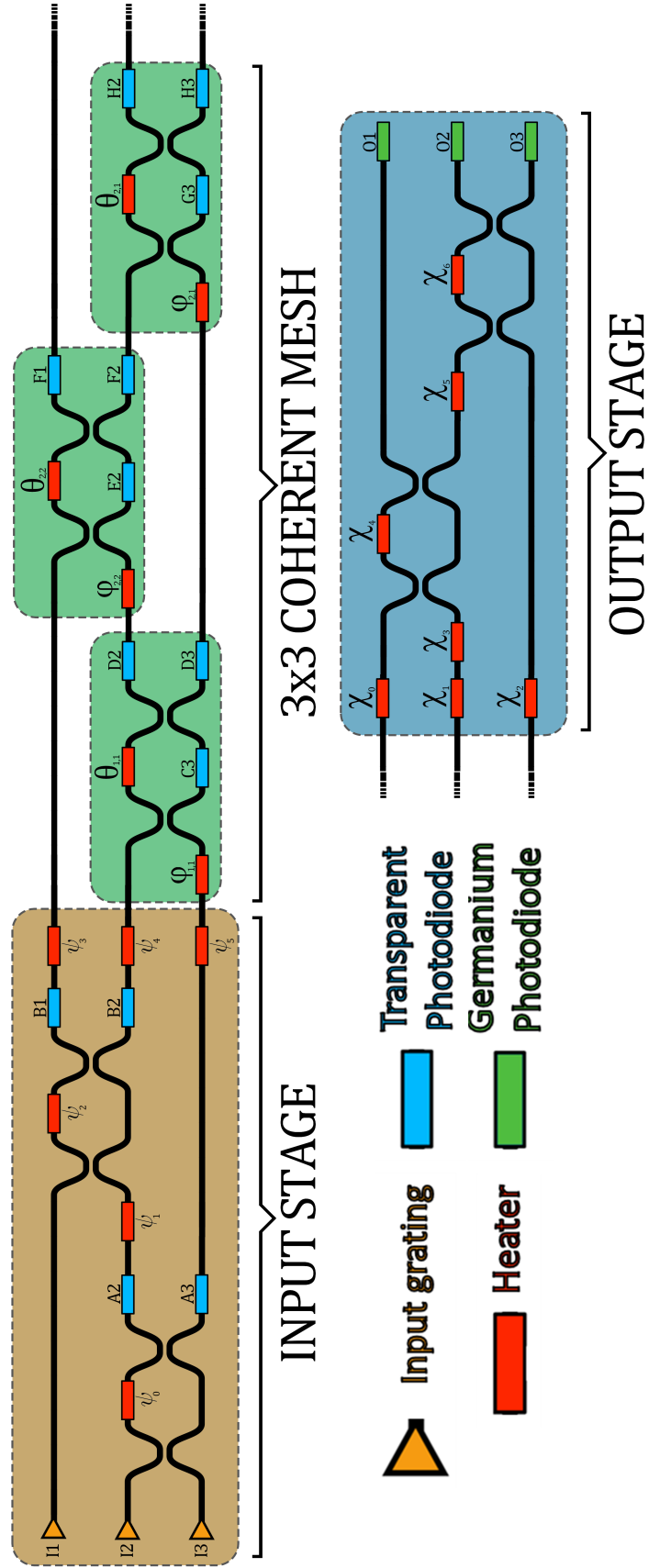


Figure 3.8: Schematic representation of the 3x3 mesh (green) along with the input generator (brown) and the phase meter (blue).

3.2.2. Input generator

As anticipated, to properly test the behavior of the mesh we must feed the matrix with a sufficient amount of inputs, and in the meantime be sure that the input vector is the expected one. Using 3 different input fibers, with perfectly controlled light beams and optically coupled with the grating couplers, is a challenging task.

The solution adopted is therefore to use a single coherent light beam provided by an external source, and then by means of a triangular mesh redistribute it between the 3 entrances.

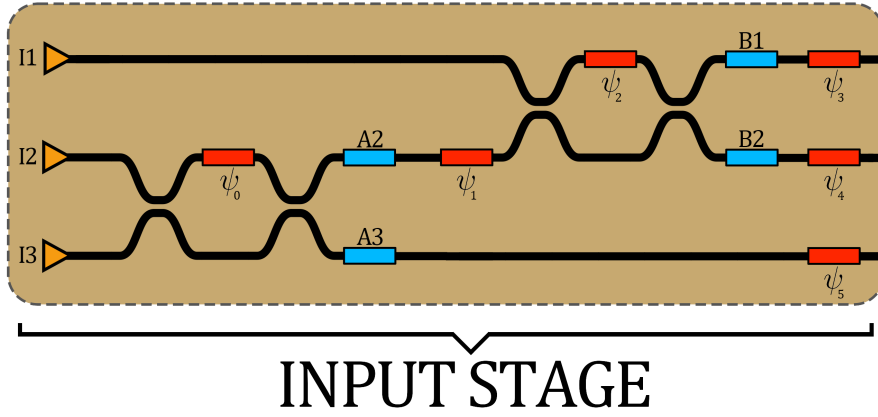


Figure 3.9: Input generator base on a diagonal structure with two MZIs. The transparent photodiodes are the blue squares along the optical path, while the heaters are the red ones.

A feedback scheme, similar to the one adopted for the matrix itself, is used to recreate an input. An input vector is chosen by software and the heaters are thus properly driven to achieve the desired power split. The control is coarser than the one present in the matrix, given the absence of transparent photodiodes between the two MZI couplers, but it still manages to set a power split very precisely. Similarly to the 3×3 matrix, the transfer function can be easily computed by combining an upper and a lower extension of the 2×2 basic transfer function. The complete transfer function $T_{In} = T_{2 \times 2 \text{ high}} T_{2 \times 2 \text{ low}}$ will be computed, accounting for the missing φ_1 actuator in the first MZI:

$$\begin{aligned}
T_{In} &= \begin{bmatrix} \frac{e^{j\theta_2}-1}{2} & j e^{j\varphi_2} \frac{e^{j\theta_2}+1}{2} & 0 \\ j \frac{e^{j\theta_2}+1}{2} & e^{j\varphi_2} \frac{1-e^{j\theta_2}}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{e^{j\theta_1}-1}{2} & j \frac{e^{j\theta_1}+1}{2} \\ 0 & j \frac{e^{j\theta_1}+1}{2} & \frac{1-e^{j\theta_1}}{2} \end{bmatrix} \\
&= \begin{bmatrix} \frac{e^{j\theta_2}-1}{2} & \frac{j}{4} e^{j\varphi_2} (e^{j\theta_2}+1)(e^{j\theta_1}-1) & -\frac{1}{4} e^{j\varphi_2} (e^{j\theta_2}+1)(e^{j\theta_1}+1) \\ j \frac{e^{j\theta_2}+1}{2} & \frac{1}{4} e^{j\varphi_2} (1-e^{j\theta_2})(e^{j\theta_1}-1) & \frac{j}{4} e^{j\varphi_2} (1-e^{j\theta_2})(1+e^{j\theta_1}) \\ 0 & \frac{j}{2} (e^{j\theta_1}+1) & \frac{1}{2} (1-e^{j\theta_1}) \end{bmatrix}
\end{aligned}$$

In our setup, we will generally enter from the bottom grating, with a corresponding vector $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, and we will use only θ_1 and θ_2 actuators. Simplifying then φ_2 as well and performing the required calculations, the equations describing the input of the matrix are the following ones:

$$\begin{bmatrix} In_1 \\ In_2 \\ In_3 \end{bmatrix} = \begin{bmatrix} -\frac{1}{4}(e^{j\theta_1}+1)(e^{j\theta_2}+1) \\ -\frac{j}{4}(e^{j\theta_1}+1)(e^{j\theta_2}-1) \\ \frac{1}{2}(1-e^{j\theta_1}) \end{bmatrix} \quad (3.12)$$

As expected the third terms is θ_1 dependent only, while the first two inputs being the outputs of the second MZI are dependent on both θ_1 and θ_2 .

To have an expectation of the possible results, a three-dimensional simulation of the powers achieved by the *Inputs generator* can be processed.

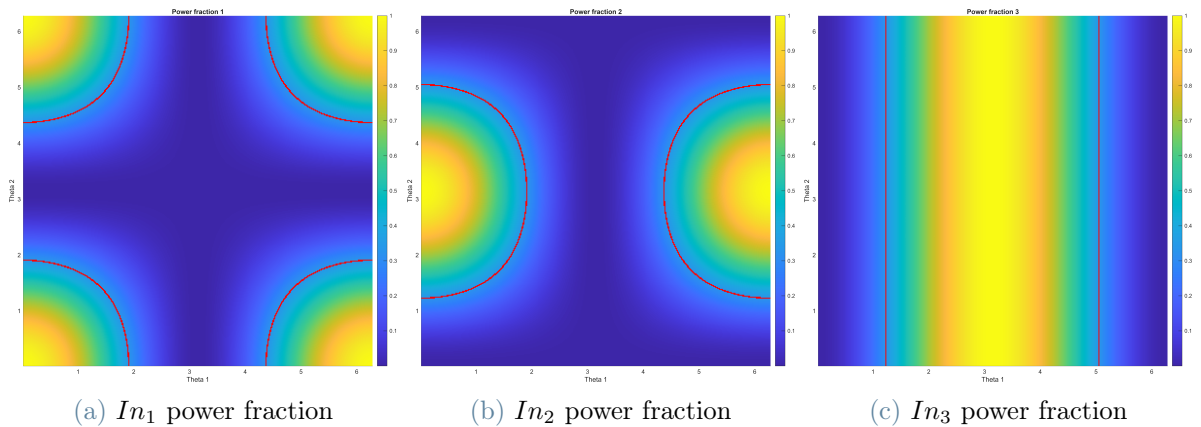


Figure 3.10: Simulation of the three input powers normalized with respect to the overall power available. The red lines in the 3 images show the phase combinations allowed to get a power split of $\frac{1}{3}$ for each one of the inputs.

As shown in the previous image, the equipotence curves takes a different shape depending on the chosen input. Choosing an input means finding the appropriate θ_1 and θ_2 pairs where the equipotence lines corresponding to the desired power splits are overlapped.

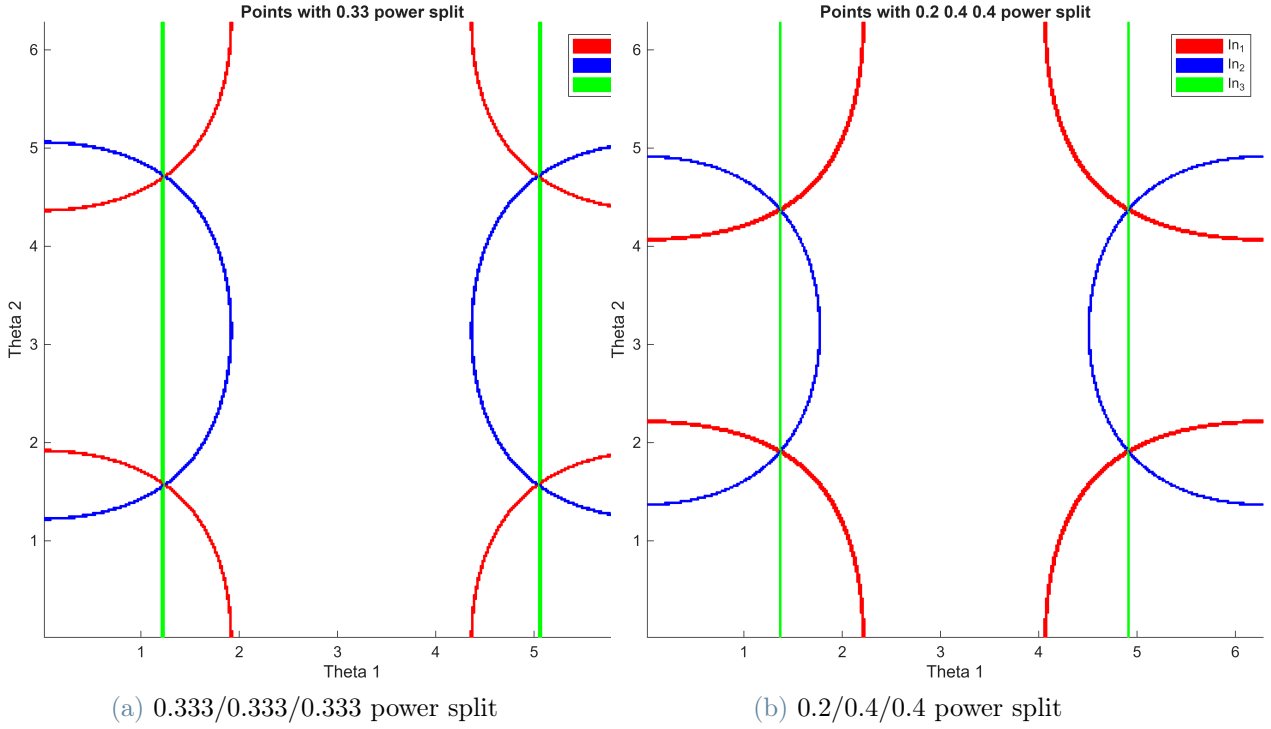


Figure 3.11: Two different power combinations for the input are shown in the plots. It can be clearly seen that 4 different solutions can be found, one for each $\frac{\pi}{2}$ by $\frac{\pi}{2}$ subplane.

Over a 2π by 2π plane, four possible solutions can be found. Even though power information is the same in the four points, what is still missing is phase information.

Up to now we haven't accounted for phase setting in the input generation process, this because while to set power we rely on the signal from transparent photodiodes, which is directly proportional to the light power itself, retrieving phase information is more complex. Once chosen a desired amplitude for the input vector, the phases can be selected among those permitted by these four points.

Phase by itself is meaningless if we don't choose a reference against which to measure it. For this reason we will measure phases in couples, so that by knowing one, obtaining the others is immediate. While theoretically each of the 3 inputs can be used as reference, experimentally we will use MZI_1 as a phase meter given its direct positioning after the second and third input. In_2 will be then chosen as reference, and a comparison between the simulated phase map and the experimental one will be performed. In this

way, the consistency of the simulation with the experimental results can be evaluated, thus accounting for any non-idealities or offsets.

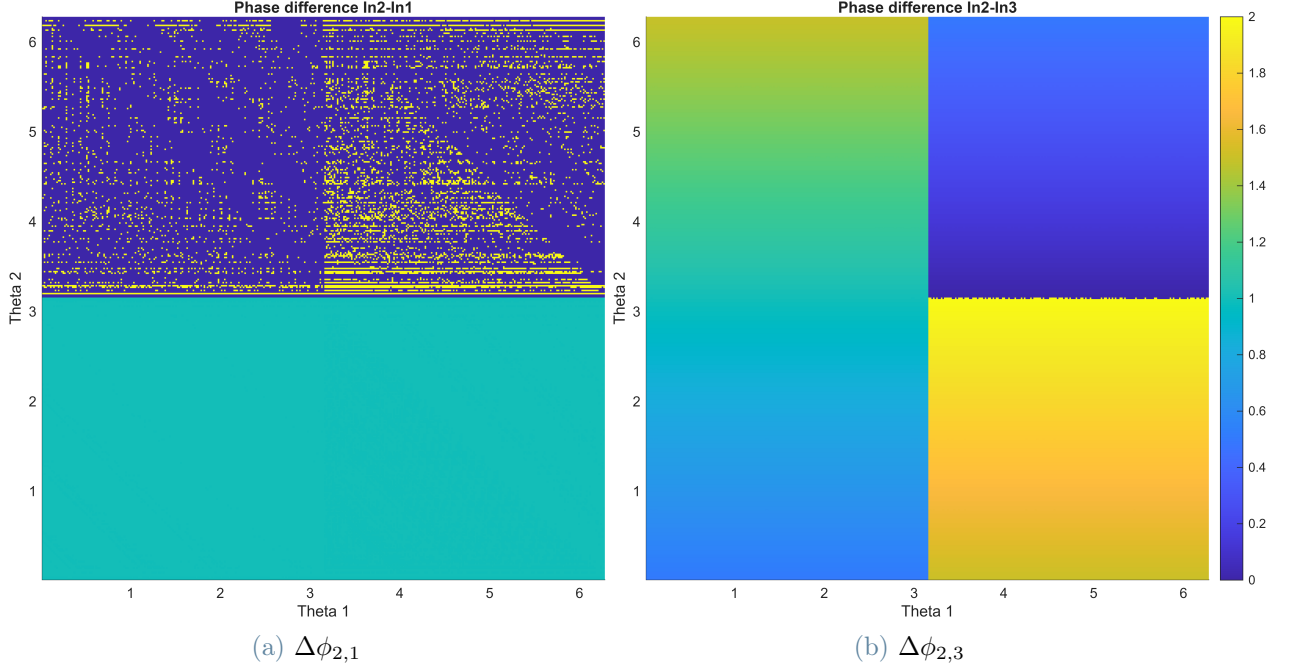


Figure 3.12: Map of the phase difference between In_2 and both In_1 (a) and In_3 (b) as a function of θ_1 and θ_2 . It can be appreciated from graph (a) how the phase between the first two inputs is 0 or π as a function of θ_2 , thus making it easier to be modeled.

Phase differences $\Delta\phi_{2,1}$ and $\Delta\phi_{2,3}$ were computed, showing very different behavior between the two graphs. $\Delta\phi_{2,1}$ shows a θ_2 only dependent trend, with a phase map that shows a phase difference of π between the first and the second half period. This is easily explained by looking at the path performed by light, it exits from the top branch of the first MZI of the input generator and enters in the bottom input of the second MZI of the structure. Its outputs, corresponding to In_1 and In_2 , will then share the phase term proportional to $\frac{\theta_1}{2}$ as long as the one proportional to $\frac{\theta_2}{2}$, thus making these terms irrelevant in terms of differential phase and keeping only the π shift. $\Delta\phi_{2,3}$ phase map is not as straightforward. While In_3 is only θ_1 dependent, In_2 as previously stated depends both on θ_1 and θ_2 . The overall phase difference results therefore in two ramps, one for each semi-period of θ_1 , with a $\frac{\theta_2}{2}$ slope, separated from each other by a π phase offset.

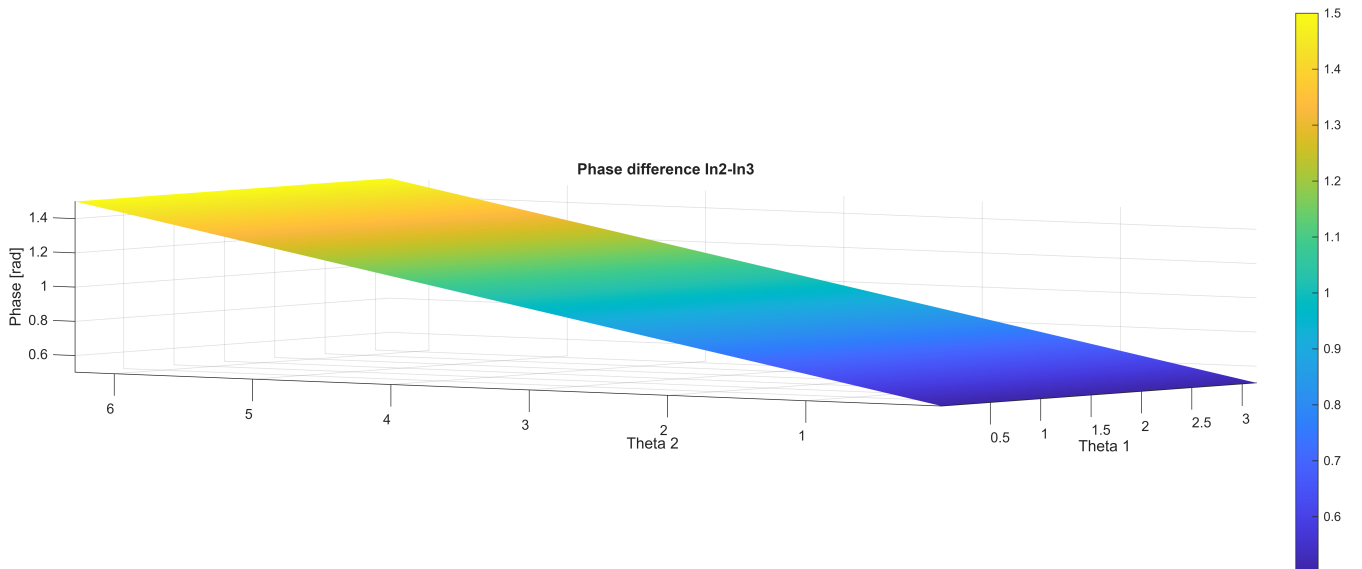


Figure 3.13: Lateral view of $\Delta\phi_{2,3}$, with only half a period simulation along θ_1 in order to make the $\frac{\theta_2}{2}$ slope of the phase map more clear.

4 | Electronic control system

In this chapter a brief introduction of the electronic system developed to read the sensors on the photonic chip, and to set the phase actuators accordingly to the required setpoints, will be given, showing the most structurally important components.

The platform used to achieve the control is an FPGA-based structure, adopted for its versatility and reprogrammability, which allows its use on different chips and for different purposes given its modularity. A **C#** user interface is then implemented to ensure a proper communication between the user and the data collected and elaborated by the FPGA. The system is composed of a motherboard, where the FPGA control unit is located, through which the necessary modules can be connected. Among these modules, an acquisition and an actuation one are responsible for sending and receiving signals from the chip, which is placed on an external PCB connected to the motherboard via a ribbon cable.

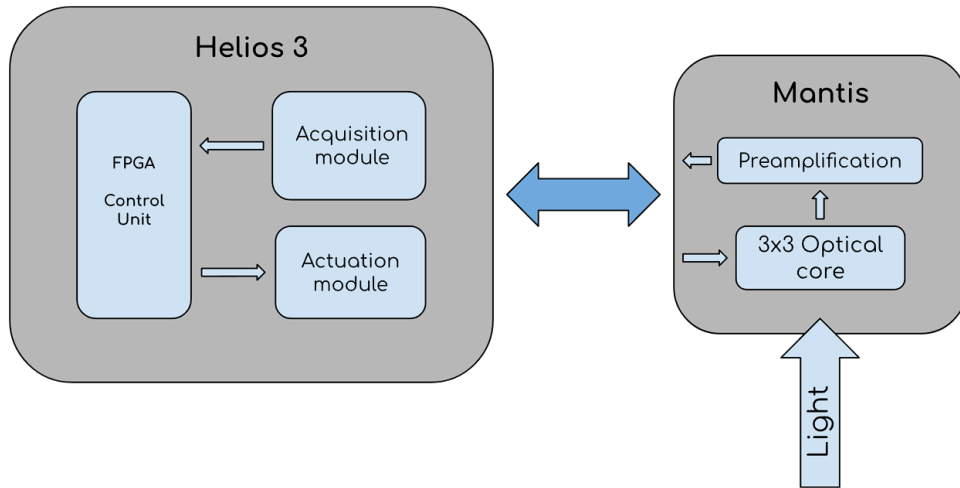


Figure 4.1: Block scheme description of the overall system

4.1. Mantis PCB

As previously stated, the optical chip is placed on top of a PCB specifically designed to optimize the vertical coupling of the optical fiber, and in the meantime it has enough space to accommodate all the components necessary for the preamplification of the on-chip photodiodes. The PCB accommodates all the components used by both 3x3 and 2x2 meshes, along with two different motherboard connection interfaces, to communicate separately with the two structures. The chip itself is then positioned and glued by means of an electrical conductive paste on top of a conductive substrate polarizable via an SMA connector.

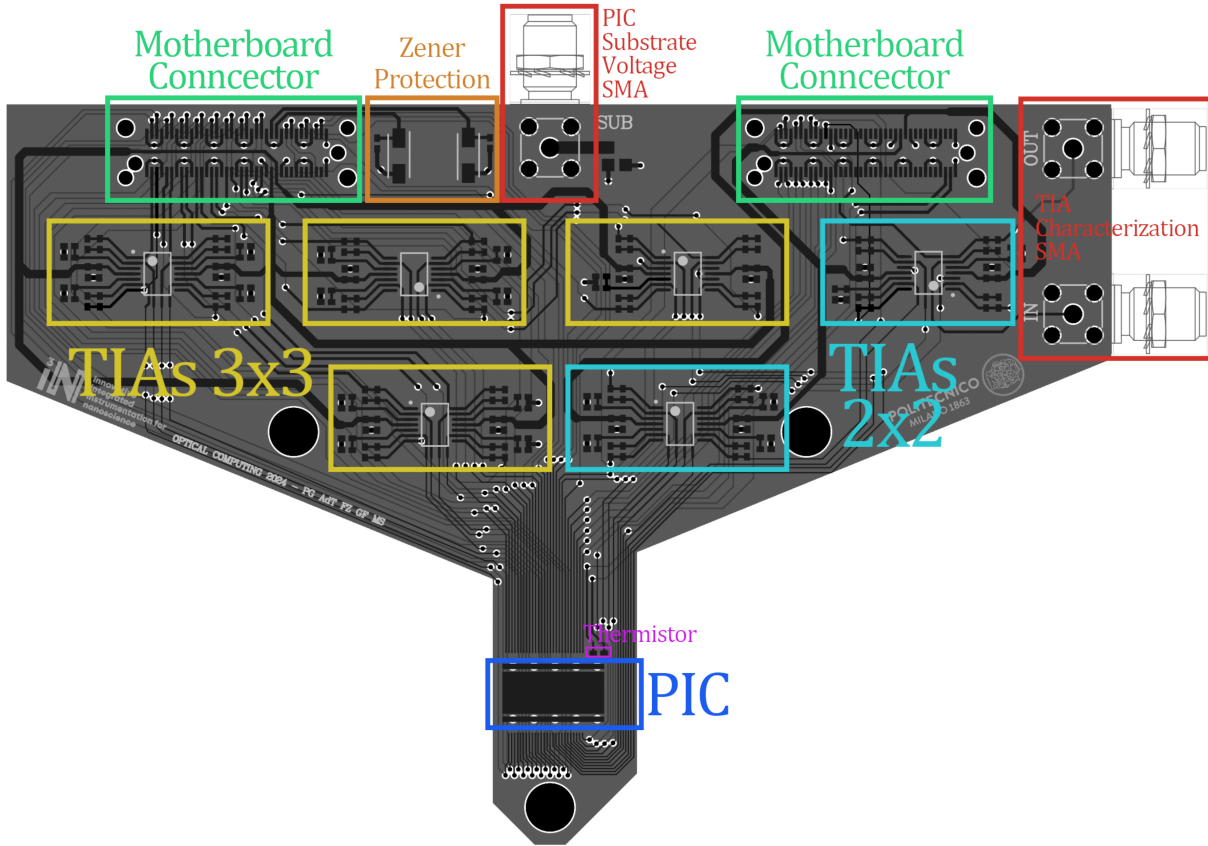


Figure 4.2: Top view of mantis PCB. It can be seen that the "two halves" of the board have the components necessary for the 2x2 and 3x3 mesh to work, along with two separate connectors adopted by the two meshes to communicate with the motherboard.

The description of the main PCB components will be listed below:

- **TIAs:** As previously stated, 6 amplifiers in Trans-Impedance (TIA) configuration will be used, 4 for the 3x3 mesh and 2 for the 2x2 mesh. Each TIA has 4 channels, and thus 4 photodiodes are preamplified by each one of them, with the feedback

sized according to the type of photodiode we are preamplifying. To preamplify transparent photodiodes, whose photocurrent reaches tens of pA , a $10M\Omega$ resistor is chosen, along with a $1pF$ capacitor, implementing a pole at $\approx 15.9kHz$. This bandwidth is enough for the system to detect eventual thermal changes (tens of Hz) but also unwanted vibrations from the setup, so that both are properly compensated by the feedback. For Germanium photodiodes we use instead $10k\Omega$ resistors, given the much higher responsivity for germanium photodetectors.

- **Substrate SMA connector:** To properly bias the substrate, an SMA connector is adopted. In our setup the voltage fed through the SMA will be generated by the actuation module itself. It was experimentally verified that biasing the substrate to $-9.89V$ helps both in terms of responsivity and gain uniformity for the transparent photodetectors along the chip.
- **Motherboard connectors:** A 74 pins ribbon cable is used to transfer and receive data from the Mantis to the motherboard, together with the amplifiers power supplies. Two separate connectors are present on the Mantis in order to decouple the actuation and the readout of the $2x2$ and the $3x3$ meshes.

4.2. Helios 3 motherboard

The Helios 3 motherboard is the backbone of the entire system, constituting the base platform around which the overall control chain is built. It is connected to the Mantis by means of 74-pins ribbon cables, used to carry on-chip photodetectors signals to the acquisition chain, as well as bringing the actuation signals and power supplies to the Mantis. Three of these connectors are present on the motherboard, connecting to PCIe connectors in pairs.

As we can see in the Figure 4.3, the motherboard hosts 6 PCIe interfaces to allow the installation of actuation and acquisition modules depending on the system under control, two of which, PCIe connectors *M1* and *M2*, are connected to SMA modules to facilitate debugging tasks. In our case these two slots will be occupied by respectively, an actuation and an acquisition module, responsible for communication between mantis and chip. An overview of these modules will be given in the next chapters. The "brain" of the motherboard relies on an FPGA-based board, the *XEM7310-A200 Opal Kelly FPGA Development Board*, featuring an *AMD-Xilinx Artix-7 XC7A200T-1* with a type-C USB interface to communicate with the development software. The board relies on a 19V DC supply, with a suitable power module generating all the necessary power lines starting from the supplied one.

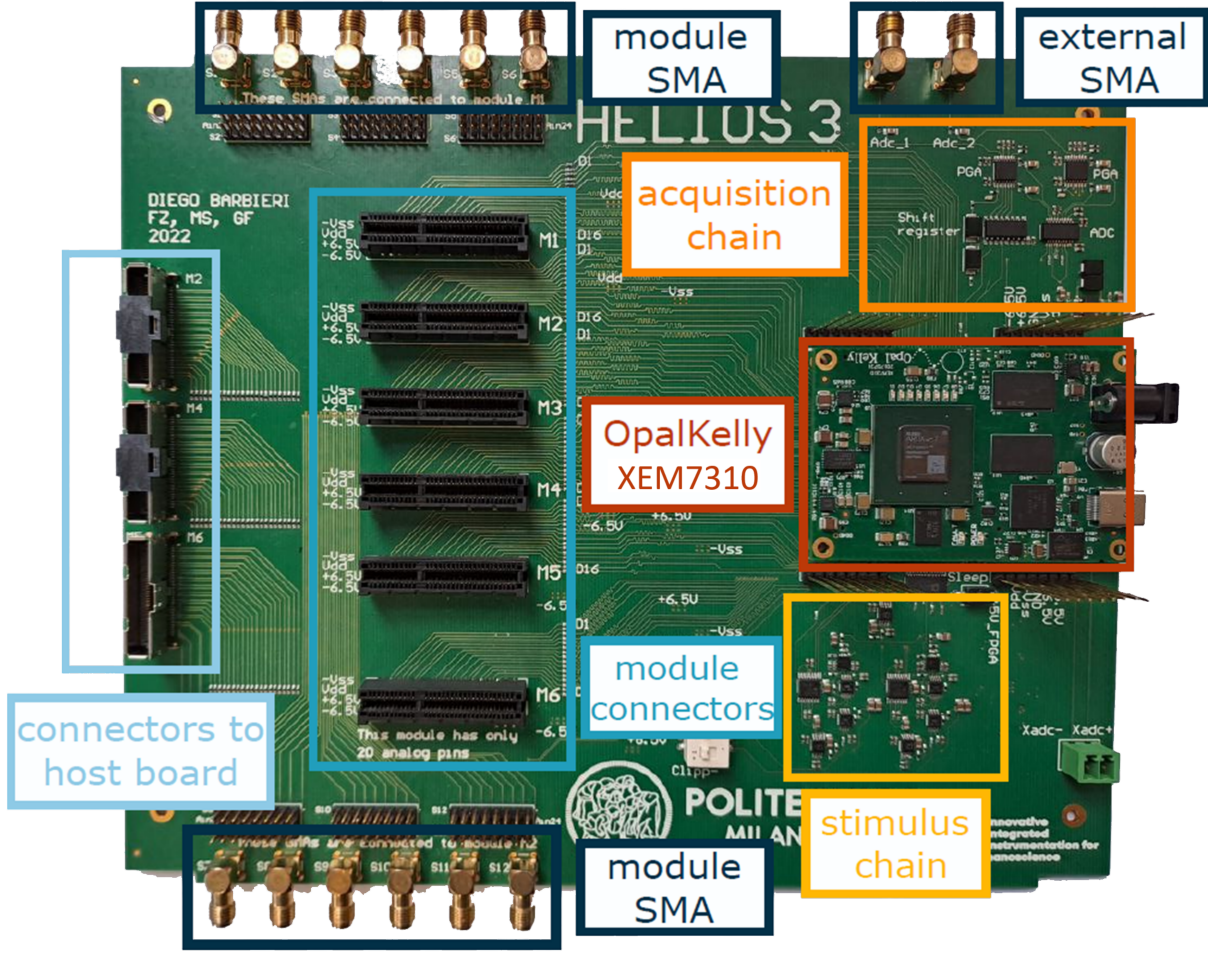


Figure 4.3: Top view of Helios 3 motherboard with a brief description of the main components.

4.2.1. Acquisition module

The acquisition module is responsible of collecting analog data from the on-chip photodiodes, and convert them in a digital version understandable by the the FPGA. Overall, 16 channels per module are available, allowing us to read all the sensors in the 3x3 mesh with a single module. The signal arrives preamplified by the TIAs for both germanium and transparent photodiodes, to increase the photocurrent signal as much as possible at the beginning of the electronic chain, but a further amplification is achievable by means of *PGAs* (*Programmable Gain Amplifiers*). Each PGA has a fixed gain of 10, which is then multiplied by one of the programmable gains obtainable with the 3 configuration bits. Eight two-channels THS7002 PGAs by Texas Instruments are present for each module, to cover all the available channels.

To correctly program all the PGAs in a single time stamp, all gain bits are sent in paral-

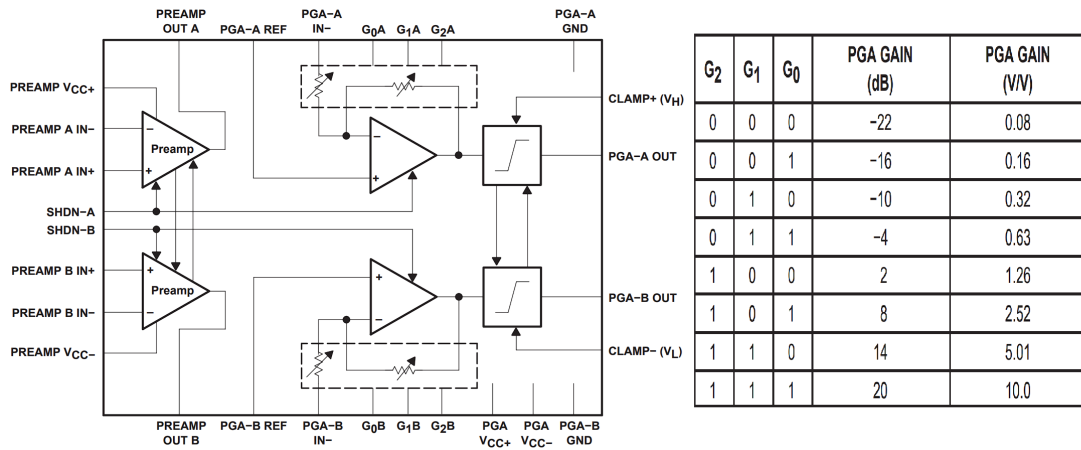


Figure 4.4: Schematics of THS7002 PGAs by Texas Instruments. Preamp gain is fixed to 10, while by choosing $G_{0,1,2}$ bits we can select the appropriate programmable gain.

lel. This is achieved by connecting six 8-bits serial-in parallel-out shift registers in a daisy chain configuration, and feeding them serially with the bit configurations provided by the FPGA. Finally, the digital data conversion is performed by 2, 8-channels, *Analog to Digital Converters (ADCs) SAR AD7606C-18* by Analog Devices with an 18 bit resolution, converting data coming from the PGAs and sending its digital version to the FPGA. The ADC also features an integrated anti-aliasing filter, with a bandwidth selectable between $25kHz$ and $220kHz$, both well above TIAs bandwidth of $\approx 15.9kHz$.

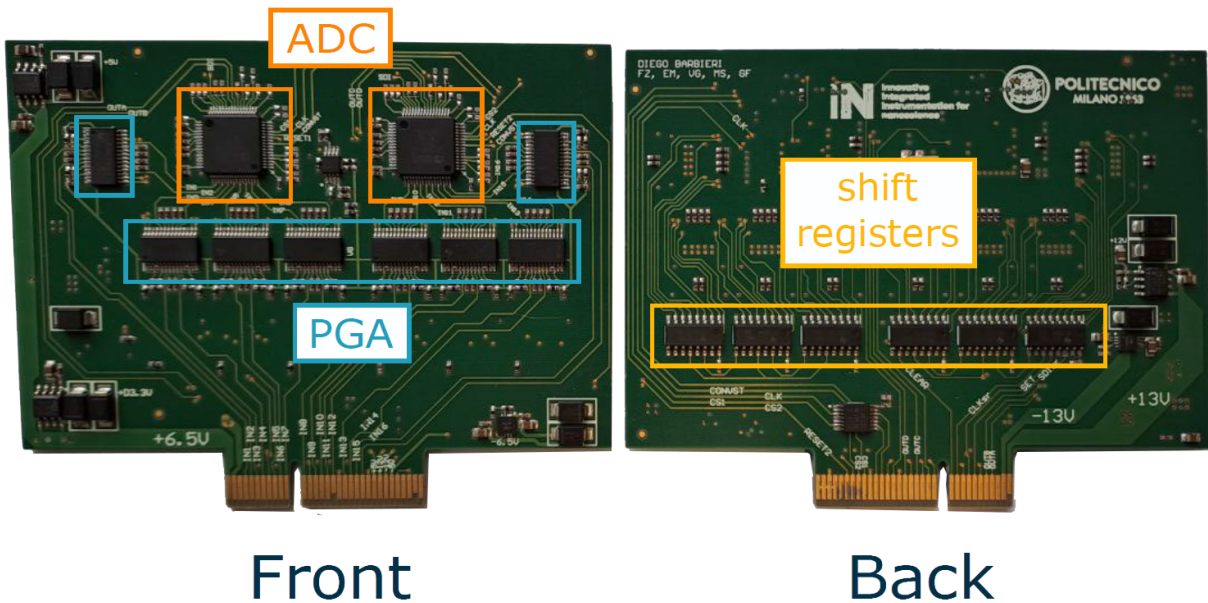


Figure 4.5: Front and back picture of acquisition module.

4.2.2. Photodiodes readout

To readout the signal along the optical path without actually affecting the signal amplitude, transparent photodiodes exploiting *Surface State Absorption (SSA)* are positioned along the optical path (section 1.3.1). Because of the physics behind their functioning, lossless inline photodetectors show a way lower responsivity compared to standard Germanium or InGaAs photodetectors working at 1550nm [19, 34].

In particular, the sensitivity we can expect from transparent photodiodes is in the order of:

$$R_{PD} \cong 20 \frac{nA}{mW}$$

Because of their low sensitivity, we must pay particular attention to noise and exploit all the available signal. A differential configuration with dummies was chosen for this reason, in order to ideally subtract all the dark current, and injecting only the photocurrent in the virtual ground of the TIA.

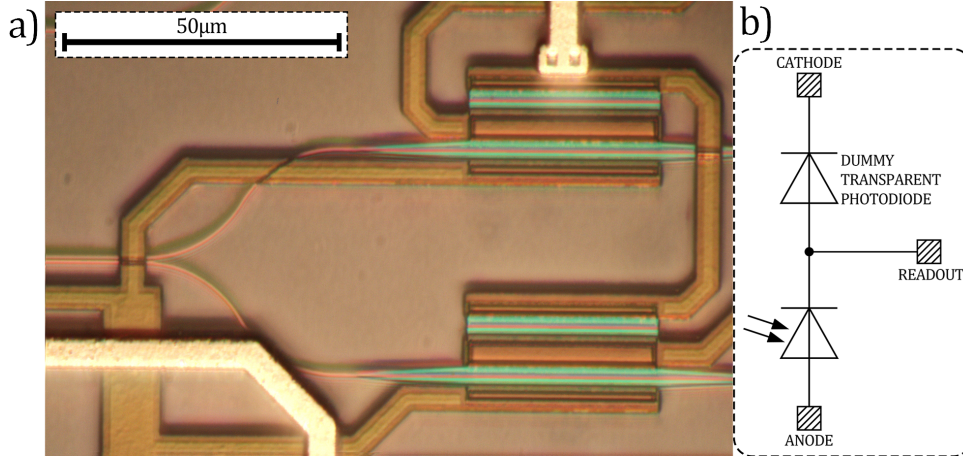


Figure 4.6: a) Physical implementation of transparent photodiodes, placed along the optical guide, along with the respective dummies b) Schematical representation of the two photodiodes, with TIA's virtual ground in between the two photodiodes.

In our experimental setup, we will use a laser source with a power in between $7.5dBm$ and $13dBm$, and therefore, also considering the losses of the optical fiber up to the insertion point, which were estimated to be in the order of $3.5dB$, we can expect a photocurrent ranging from $51nA$ to $182nA$, ≈ 5 orders of magnitude lower than the one we would obtain with a germanium sensor. A proper preamplification is then obtained by using a $10M\Omega$ resistor in the amplifier's feedback, with a resulting output voltage between $50mV$ and $180mV$, further amplifiable by the following PGAs as reported in the previous paragraph

(4.2.1).

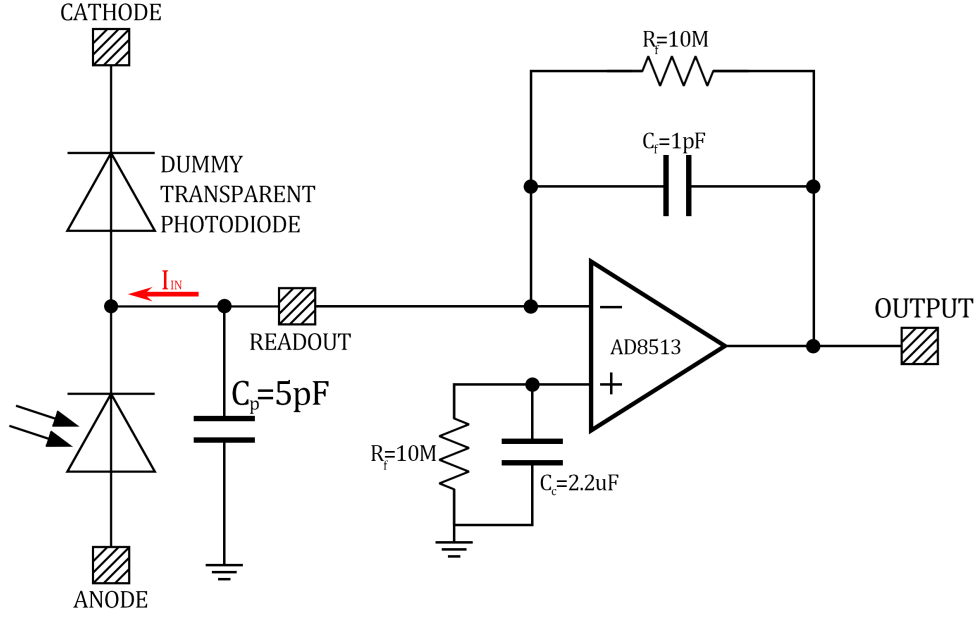


Figure 4.7: Schematics of photodiodes readout chain on the Mantis. The dummy photodiode has its anode connected to the virtual ground to subtract the dark current. As can be seen, the same resistance is present at the "+" and "-" terminal of the amplifier to compensate for bias currents coming out of the amplifier.

The amplifier used for the role is an *AD8513 JFET Operational Amplifier* from Analog Devices, characterized by a low current noise characteristic of JFETs and a typical voltage noise of $8 \frac{nV}{\sqrt{Hz}}$ at $1kHz$, making it perfect for amplifying high impedance current sources such as photodiodes. Feedback resistance, together with feedback capacitance, defines the amplifier's bandwidth with a $-3dB$ dominant pole at $15.9kHz$, introducing also the main source of noise, with a current noise of:

$$i_{R_F} = \sqrt{\frac{4kT}{R_F}} \cong 40.7 \frac{fA}{\sqrt{Hz}}.$$

A similar readout will be implemented also for Germanium photodiodes implemented at the end of the mesh for phase readings. The overall lower signal arriving at the end of the mesh, around 80% of the input one, together with an higher responsivity for germanium photodiodes of $R_{Ge} \cong 0.7 \frac{A}{W}$, makes the feedback resistor in this case chosen to be $10k\Omega$.

4.2.3. Actuation module

The actuation module is the final step closing the control loop. The signal, preamplified by TIAs, acquired and digitized by the acquisition module and elaborated by the FPGA, which retrieves both the DC component of the signal and the derivative, to achieve the required function described by the control algorithm, is then sent to the actuation module that properly drives the phase shifters, and biases the photodiodes.

The module has the capability to drive up to 24 loads, with a maximum channel current of $50mA$, obtained thanks to six 4-channels *AD8513 JFET Operational Amplifier* used as current drivers.

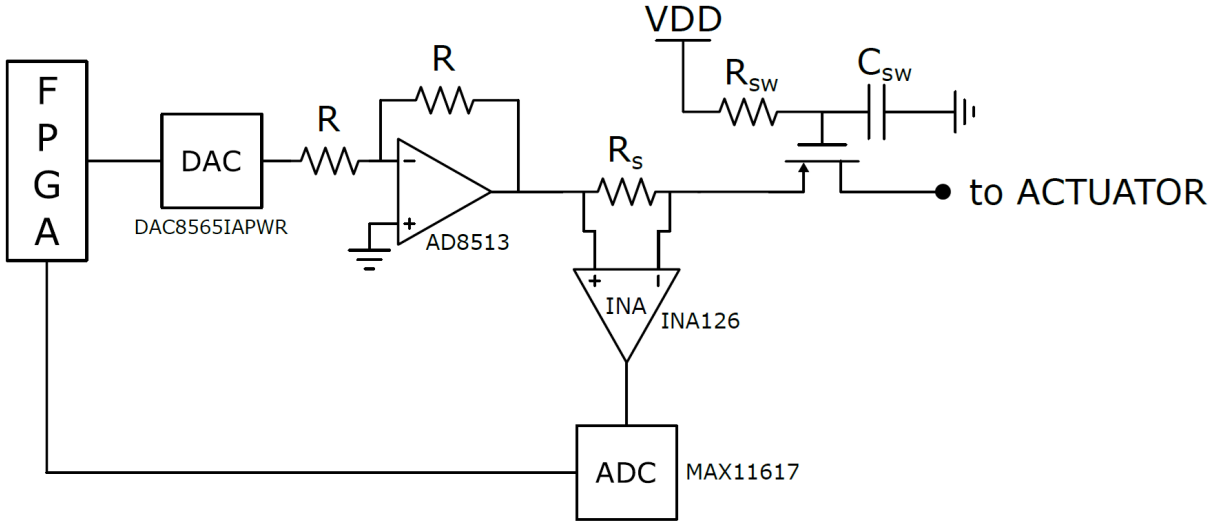


Figure 4.8: Actuation module schematic, showing signal path from the FPGA to the actuator. The digital signal from the FPGA is then amplified and feed into a shunt resistor, to read the real-time flowing current. A transistor used as a switch decouples then the actuator from the driving electronics at start-up.

As shown in schematic 4.9, the digital output generated by the FPGA is converted to an analog signal by a *Digital-to-Analog Converter (DAC)* *DAC8565IAPWR* by *Texas Instruments*. It is a 4-channel 16 bit converter working up to $50MHz$ with an output dynamics between 0 and $5V$. It communicates with the FPGA by using a 3-wire SPI interface.

At the output of the current driver, a small shunt resistance R_s is readen by an INA to return a real time measurement of the current flowing in the actuator. The INA gain was chosen to be 10 to fit the input signal dynamics of a following ADC bringing the information back to the FPGA by means of a 2-wire I2C interface. The ADC in question has 12 bits, enough for a sufficiently precise approximation of the current.

A protection transistor is then placed right before the actuator to protect the heaters from unwanted voltage spikes at start-up.

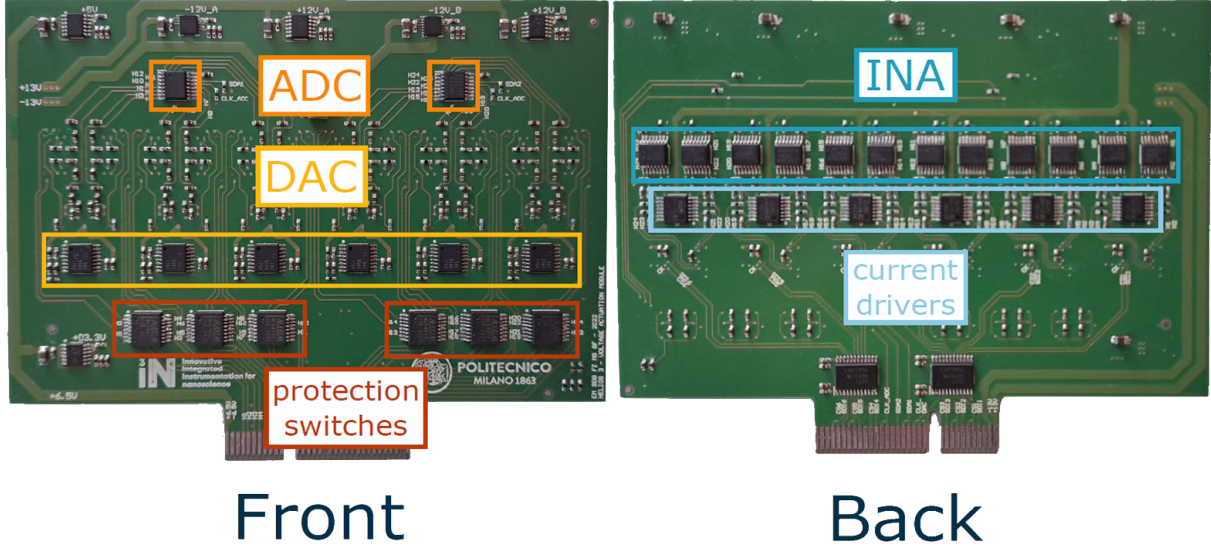


Figure 4.9: Actuation module schematic, showing signal path from the FPGA to the actuator.

4.3. FPGA Algorithm Implementation

The core of the control chain lies in the signal processing by the FPGA, which extracts the DC value and the derivative from the digitized signal coming from the ADC. This information is then used by the integral control to reach the desired working point in the two possible modes: *PowSplit* and *Max/Min*.

4.3.1. Low-Pass Chain

A digital low pass chain is implemented to extract the DC value from the incoming signal. It is based on a digitally implemented low pass filter (LPF), modeled via the Z-transform, tunable by means of an α parameter selectable by software, and followed by a FIFO adopted to store a decimated version of the acquired data.

The decimation is performed by means of tunable *Cascaded Integrator Comb (CIC)* filters, positioned between the LPF and the output register and implementing also an anti aliasing function. Its use of only shift registers and adders makes it particularly suitable for low power applications like FPGAs.

To obtain the transfer function in the Z-domain, we can start from the general expression in the S-domain:

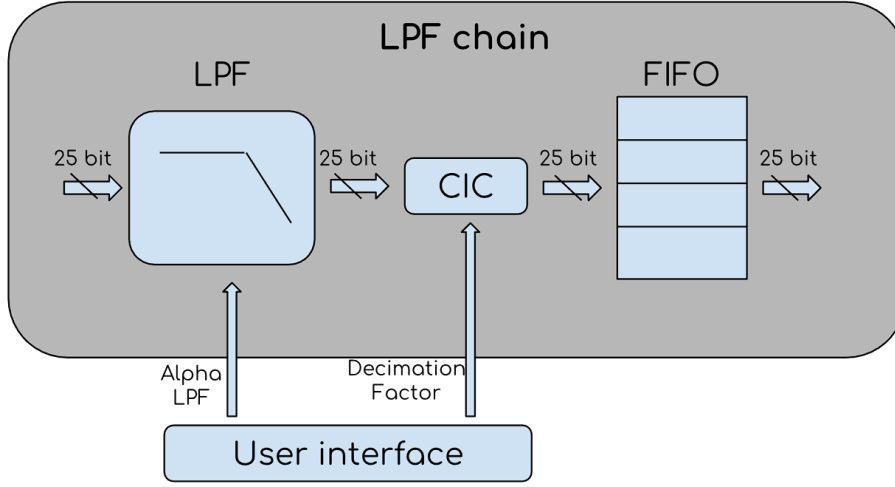


Figure 4.10: LPF chain for the DC value extraction

$$H_{LPF}(S) = \frac{\mu}{1 + S\tau}$$

Now applying the Tustin bilinear transformation, where f_s is the sampling frequency:

$$S = 2f_s \frac{1 - Z^{-1}}{1 + Z^{-1}} \quad (4.1)$$

we get:

$$H_{LPF}(Z) = \mu' \frac{1 + Z^{-1}}{1 - \alpha Z^{-1}} \quad (4.2)$$

Where:

$$\mu' = \frac{\mu}{1 + 2f_s\tau}$$

$$\alpha = \frac{2f_s\tau - 1}{2f_s\tau + 1}$$

The frequency of the pole deduced through these parameters is the further:

$$f_p = \frac{f_s}{\pi} \frac{1 - \alpha}{1 + \alpha} \quad (4.3)$$

From equation (4.1) we can deduce that at low frequency $Z^{-1} = 1$. By knowing this, the low frequency amplitude is:

$$H_{LPF}(0Hz) = \mu' \frac{2}{1 - \alpha} \quad (4.4)$$

We notice that both expression 4.3 and 4.4 contains α , meaning that a variation in the frequency of the pole results in a low frequency gain change, and therefore also in a DC

loop gain amplitude variation. A decoupling of the two parameters has to be performed, and this is achieved by normalizing the DC transfer gain, and thus dividing by $\frac{2}{1-\alpha}$.

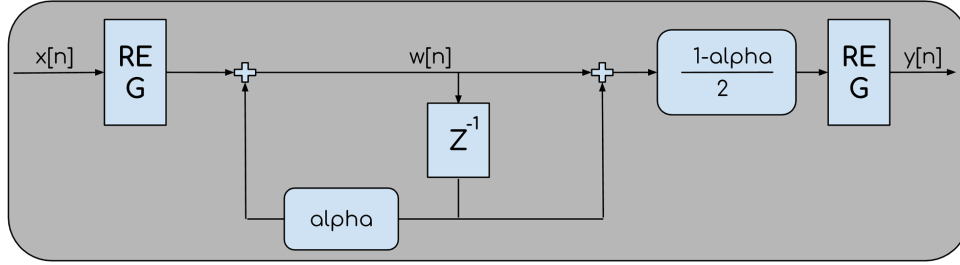


Figure 4.11: Normalized *Low Pass Filter* scheme

The previous scheme implements the following transfer function:

$$H_{LPF}(Z) = \frac{1 + Z^{-1}}{1 - \alpha Z^{-1}} \frac{1 - \alpha}{2} \quad (4.5)$$

By now decoupling the $-3dB$ pole and the DC gain, the desired LPF frequency can be chosen without worrying about variations in the DC loop gain.

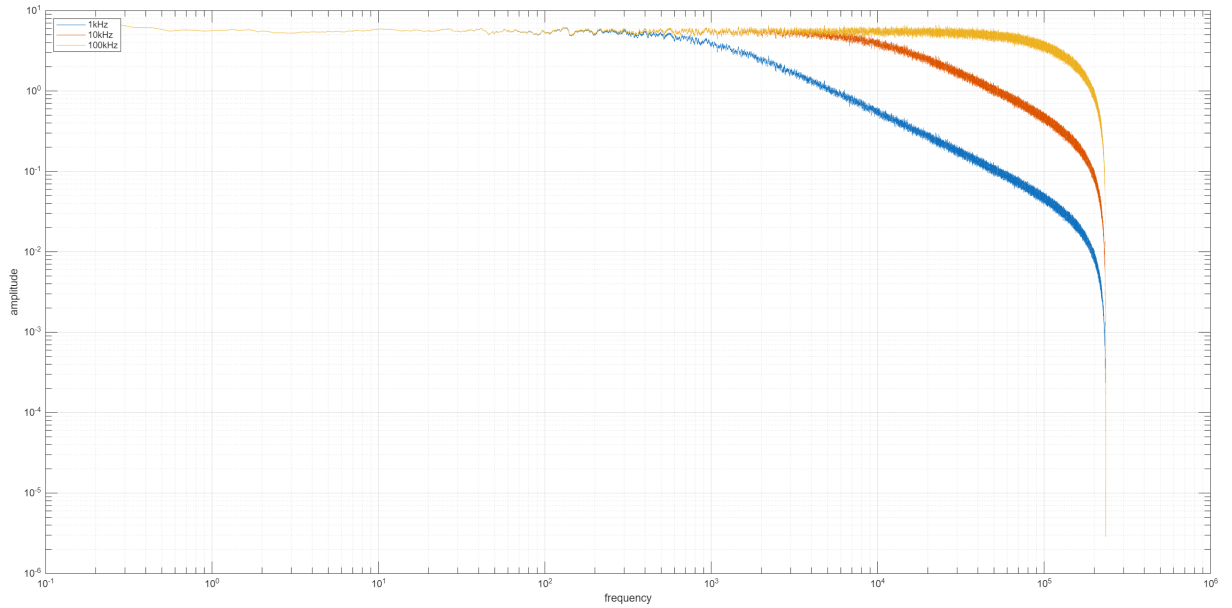


Figure 4.12: LPF simulation of the actual VHDL implementation with 3 different poles, taken with a decade of distance between them at: $1kHz$ $10kHz$ and $100kHz$, showing the same in-band amplitude. It is also visible how the filter works properly up to the Nyquist frequency, which in our case is $f_n \cong 234kHz$.

To properly simulate the filter's behavior, a white noise signal was generated on Matlab and then provided as input in a Vivado simulation, where the filter itself is implemented.

It is visible how, before the pole, the filter's gain is the same regardless of the pole position, thus showing an independence of the filter gain from the pole position.

Anti-transforming expression 4.5, a time domain expression of the previous equation can be computed:

$$y[n] = \frac{1 - \alpha}{2}(x[n] + x[n - 1]) + \alpha y[n - 1] \quad (4.6)$$

Following the same principle, the digital signal is also normalized by the PGA gain, to keep the loop gain as independent as possible from variables that could vary runtime.

4.3.2. Lock-in chain

The Lock-In chain was implemented to obtain the derivative of the incoming signal by means of dithering.

Dithering is a technique used to extract the first derivative of a signal by superimposing a modulated signal over the DC one. In our specific case this is obtained by adding a small sinusoidal voltage over the main one driving the heaters, resulting in an output power read by the photodetectors modulated around the operating point. Mathematically speaking, the result we will get is the following:

$$V_{ACT}(t) = V_{BIAS} + v_{dith} \sin(2\pi f_{dith} t)$$

With $V_{ACT}(t)$ the overall actuation voltage, and V_{BIAS} and v_{dith} respectively the bias of the heater and the superimposed modulation. The resulting power, if the dithering signal is small enough, can be approximated with a first-order Taylor series around V_{BIAS} as:

$$P_{OUT}(t) \approx P_{OUT}(V_{BIAS}) + \left. \frac{\partial P_{OUT}}{\partial V_{ACT}} \right|_{V_{BIAS}} (V_{ACT} - V_{BIAS}) \quad (4.7)$$

Noting that $V_{ACT} - V_{BIAS} = v_{dith} \sin(2\pi f_{dith} t)$, we see how a component proportional to the first derivative of the signal is modulated at the frequency f_{dith} [35].

Lock-In chain is thus implemented to get this high frequency component. A high-pass filter is adopted to cancel out all the DC components. A demodulation is then performed to bring the component at f_{dith} back to DC, and then low-pass filtered to reduce the remaining noise. To cancel out the remaining spurious harmonics, CIC filters are implemented also in this case, tuned to have their notches matching with the spurious harmonics frequencies.

The HPF in the *Lock-In chain* is digitally implemented following a scheme similar to the one depicted in figure 4.11.

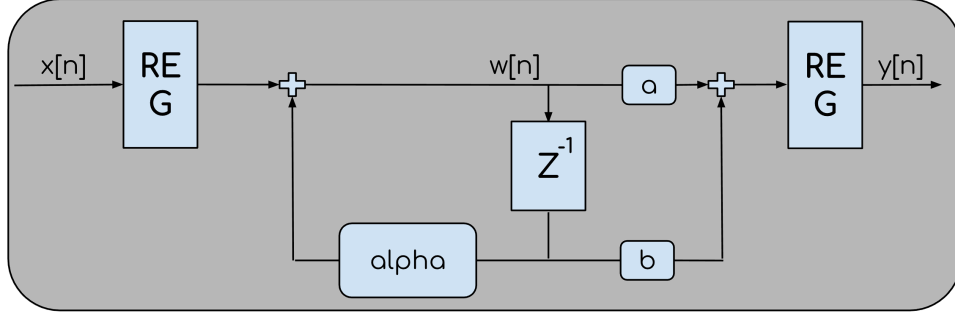


Figure 4.13: Generic digital filter scheme as a function of parameters a, b and α

The general equation in the S-domain of scheme 4.13 is the following:

$$H(S) = \frac{a + b + S \frac{T_s}{2} (a - b)}{1 - \alpha + S \frac{T_s}{2} (1 + \alpha)} \quad (4.8)$$

It can be noticed how to obtain a high-pass filter with a zero in the origin, the two coefficients a and b must be equal and opposite. By choosing $a = 1$ and $b = -1$ the following transfer function implementing a *High-Pass filter* in the Z -domain is obtained:

$$H(Z) = \frac{1 - Z^{-1}}{1 + \alpha Z^{-1}} \quad (4.9)$$

With a time response described by the following equation:

$$y(n) = x(n) - x(n-1) + \alpha y(n-1) \quad (4.10)$$

4.3.3. Integral control

The integral control is responsible for the convergence of the actuators towards the working points of interest. By means of it, the voltage needed to reach the desired phase shift is computed, with the possibility to choose between *PowerSplit mode* and *Maximum/Minimum setting mode*. The user can freely select the desired operation mode via software.

- **PowSplit setting mode:** the integral controller, using the DC value of the photodiodes coming from the Low-Pass filtering chain, forces a specific power split given by the user.

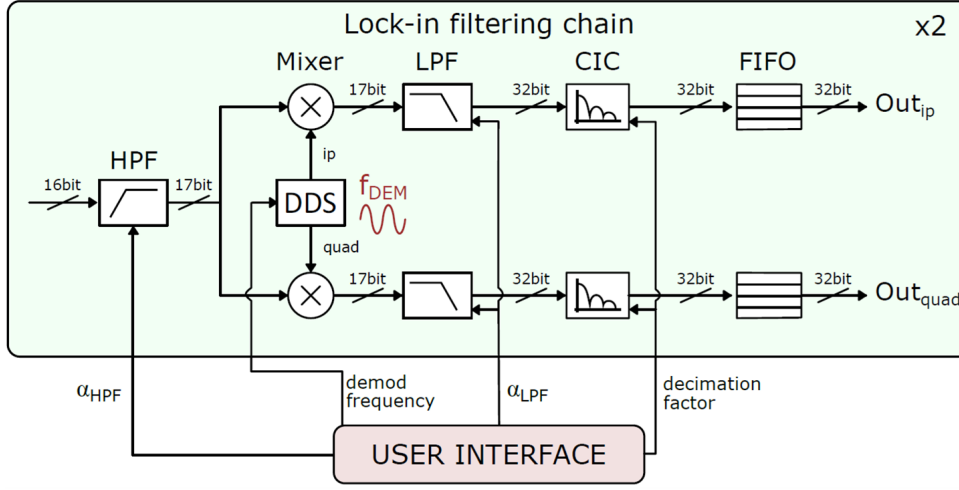


Figure 4.14: Lock in filtering chain showing both *in-phase* and *quadrature* paths.

- **Maximum-Minimum setting mode:** the integral controller works with the derivative of the signal coming from the Lock-in chain. The integrator then acts in order to minimize or maximize the derivative to reach a stationary point, and thus a minimum or a maximum for the signal, setting 0 as target value. The convergence towards one stationary point or the other is defined by the loop sign.

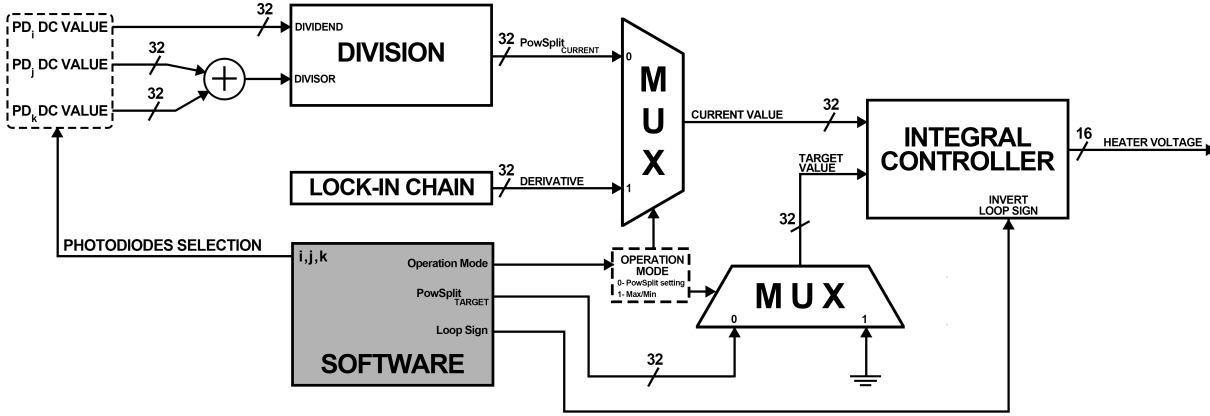


Figure 4.15: Schematic description of the integral acquisition chain. The operation mode is set with two different multiplexers. The first one is used to feed the integral control with desired current value, and the other to choose the correct target, calculated as a function of the operation mode.

The power split mode is the one actually used when performing optical computing operations. It takes three signals coming from the photodiodes and evaluates the local power split:

$$P_{split} = \frac{P_{Di}}{P_{Dtot}} = \frac{P_{Di}}{P_{Di} + P_{Dj}}$$

The power split is usually computed by evaluating the light that crosses the photodiode in one of the two branches, and normalize it by the overall light in the same section of the MZI. An exception is the power split concerning the portion in-between the two couplers, because of the presence of a single photodiode in the bottom branch, which is why the overall power is taken at the output of the MZI.

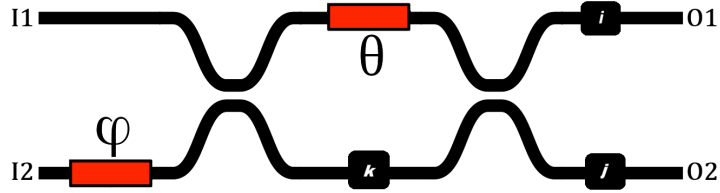


Figure 4.16: Positioning of the photodiodes inside of a MZI. Three photodiodes are present for each MZI, here represented by k , i and j .

The integral controller adopted is a trapezoidal integrator as shown in [36], with the following time implementation:

$$y[n] = k(x[n] + x[n - 1]) + y[n - 1]$$

Corresponding to a Z-domain transform:

$$H_{integrator}(Z) = \frac{1 + Z^{-1}}{1 - Z^{-1}}k \quad (4.11)$$

It can be seen how this corresponds to the same transfer function of the LPF shown in 4.5, with the α coefficient equal to 1 to bring the pole in the origin.

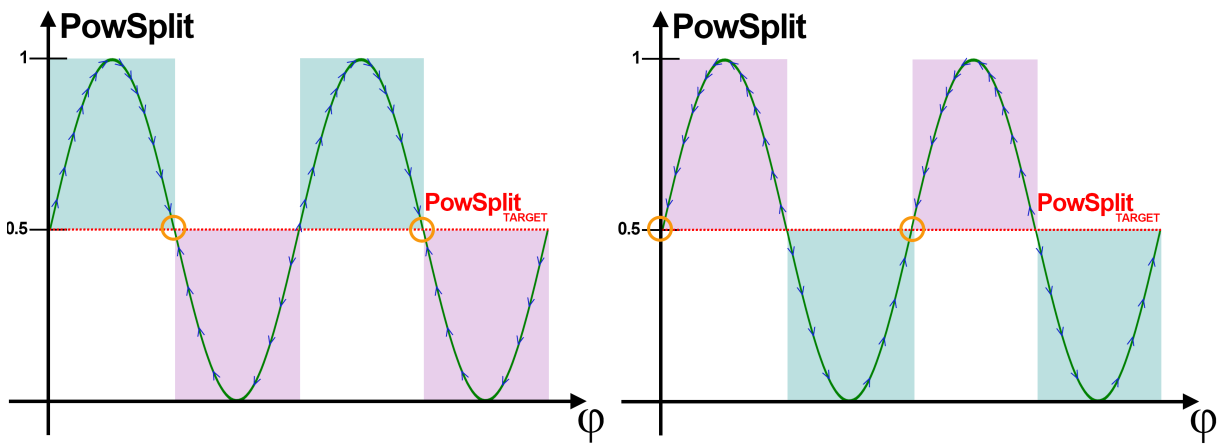


Figure 4.17: On the left we model the error at the input of the integrator as $S_{real}[n] - S_{target}[n] = S_{error}[n]$, this results in an increase of heaters voltage for measured values above the target ones, and in a decrease of heater voltage for the lower ones. On the right a loop with the opposite sign is shown, with $S_{target}[n] - S_{real}[n] = S_{error}[n]$, which results in a different direction of convergence and thus in different convergence points.

5 | System characterization and experimental validation

In this chapter, an overview of the actual behavior of the implemented system and the actual achievements we obtained during its validation are shown.

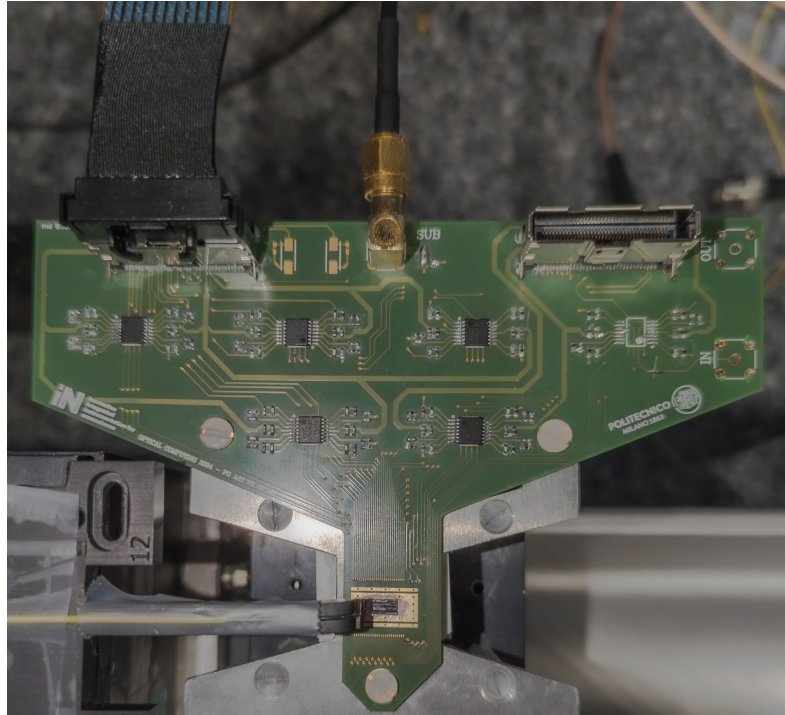


Figure 5.1: Top view of Mantis PCB mounted on the optical bench. The input optical fiber can be seen in the bottom, together with the optical chip, mounted on a 3D printed support to guarantee a 12° entrance angle for the light. On the top portion of the board instead, the ribbon cable for data transfer and the SMA connector to bias the substrate are connected.

The system under exam is a 3×3 matrix, realized with *Mach Zehnder interferometers* (MZI) following Reck architecture, thus requiring the implementation of three MZIs for the matrix structure 2.3.1. Each interferometer is realized with two phase actuators

(TiN heaters), and three transparent photodiodes, to precisely sense and control the interferometer's behavior at run-time. Light is fed to the mesh by means of an external laser source at $1550nm$ with a variable power from $7.5dBm$ to $13dBm$, carried with a single optical fiber and finally coupled to the structure with optical gratings. The single input light beam is then manipulated by an input vector generator, a specific structure constituted by a diagonal mesh of interferometers designed to obtain all the desired possibilities for what concerns the input vectors in terms of power split .

During the preliminary characterization of the system, particular attention will be dedicated on testing the electro-optical behavior of *TiN* (*Titanium Nitride*) heaters and *transparent photodiodes*, the two main elements allowing the control of the structure. *VMM* (*vector-matrix multiplication*) tasks were then performed, thus verifying the accuracy of the system.

5.1. TiN heaters electrical behavior

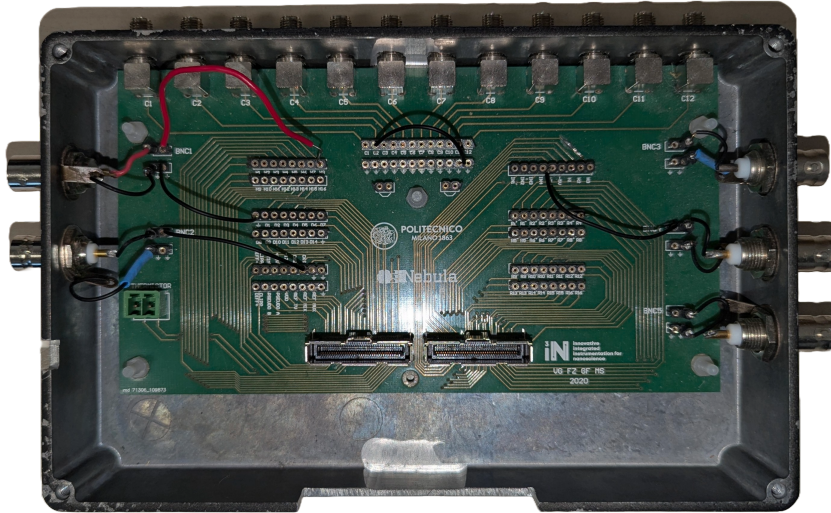


Figure 5.2: Interface board to create a communication between Mantis PCB and external instrumentation. The connection with the Mantis takes place by means of ribbon cables, while the interaction with external instrumentation is performed with BNC and SMA connectors.

An electrical characterization of the heaters was performed to understand the safe operating range of the devices. To properly perform it, the Mantis PCB, where the chip is positioned, is connected with a ribbon cable to an interface board. This interface board is routed to bring the signals from the ribbon connector to *turned-pin female headers*, to be then connected to *SMA*s (*SubMiniature version A coaxial connector*) or *BNC*s (*Bayonet*

Neill Concelman connector).

Each heater integrated in the chip is connected between a dedicated supply and ground. A *SMU* (*Source Measure Unit*) is then used to force a voltage between the two nodes just mentioned, and the resulting current is measured.

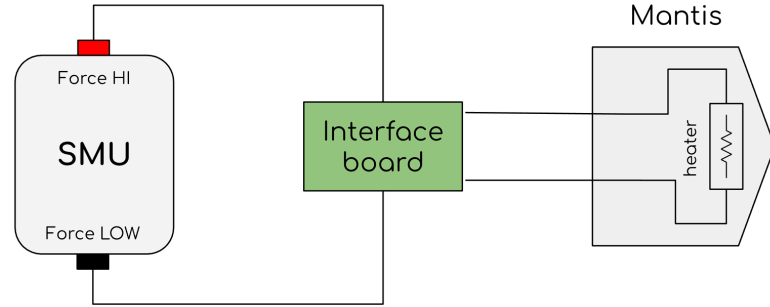


Figure 5.3: SMU interconnection diagram with the heater

The SMU used for this purpose was a Keithley 2450, with the main specifications summarized in the following table:

Range	Source Resolution	Measure Resolution
<i>Voltage Specifications</i>		
20 mV	500 nV	10 nV
200 mV	5 μ V	100 nV
2 V	50 μ V	1 μ V
20 V	500 μ V	10 μ V
200 V	5 mV	100 μ V
<i>Current Specifications</i>		
10 nA	500 fA	10 fA
100 nA	5 pA	100 fA
1 μ A	50 pA	1 pA
10 μ A	500 pA	10 pA
100 μ A	5 nA	100 pA
1 mA	50 nA	1 nA
10 mA	500 nA	10 nA
100 mA	5 μ A	100 nA
1 A	50 μ A	1 μ A

Table 5.1: Keithley 2450 SMU Source and Measure Specifications

Within the chip, heaters of two different sizes and thus different resistances were implemented, namely with length $60\mu\text{m}$ and $100\mu\text{m}$, resulting in the need to make two distinct characterizations. A sweep was performed for them both, along with a time stress-test, to

see both the linearity of the components and their resistance over time, to operate them safely during optical computing tasks.

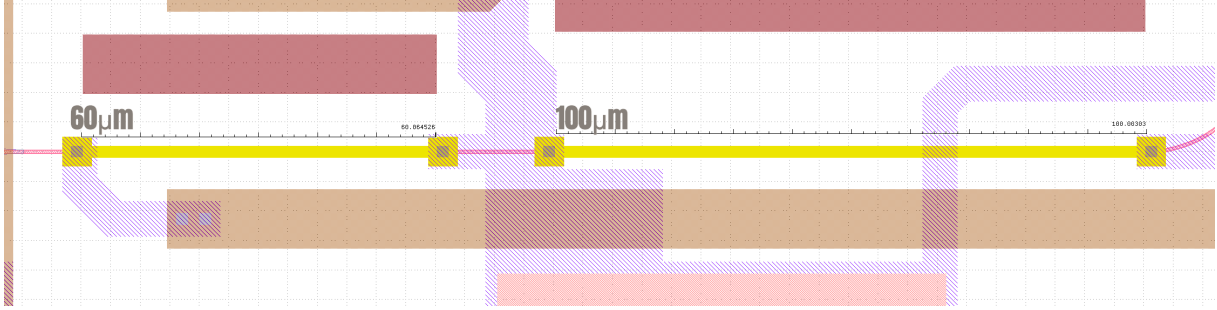


Figure 5.4: Heaters with different size were used for the input mesh and the computational stage, respectively $60\mu m$ and $100\mu m$

5.1.1. $100\mu m$ Heaters

A first characterization of the longest $100\mu m$ heaters was initially performed, the main purpose was to experimentally verify the resistance of the phase shifters of the matrix, their linearity with respect to increasing voltages and the maximum current the components can sustain.

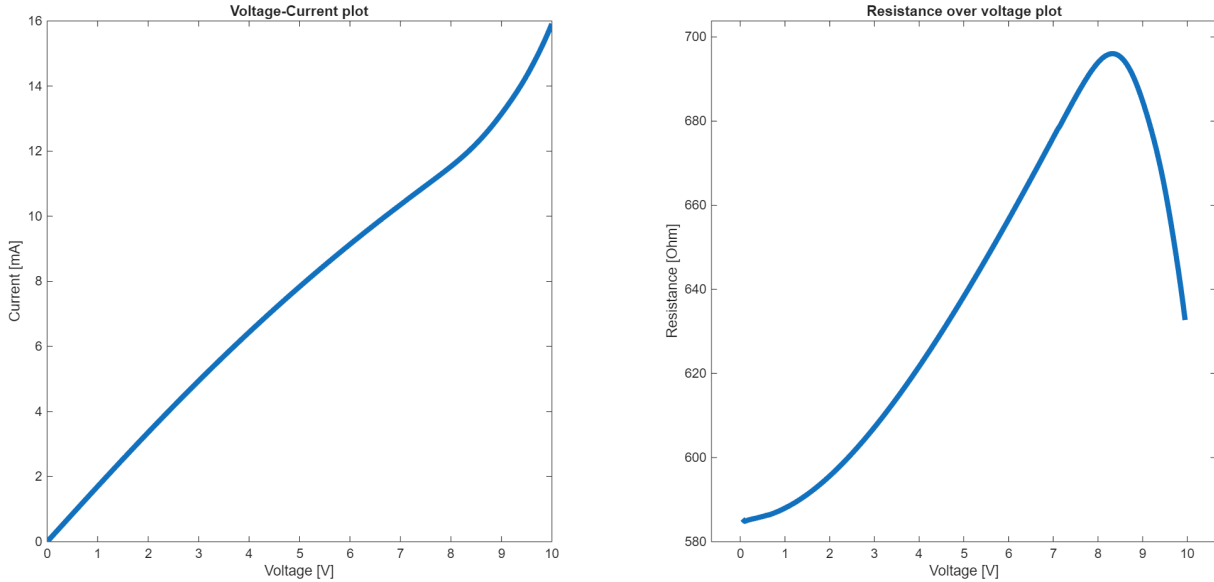


Figure 5.5: On the left, the current-voltage plot for the $100\mu m$ heaters between 0 and 10V. On the right the corresponding resistance. An increasing resistance is measured up to 8.35V, showing the PTC behavior of TiN.

Figure 5.6

A sweep between 0 and 10V was performed, with 49.7mV steps spaced 182.6ms apart to guarantee a precise description of $I-V$ the curve. Up to $\approx 8.35V$ the heater shows a coherent trend, with an overall resistance variation of $\approx 110\Omega$, corresponding to a 19% variation of the initial resistance. Up to 8.35V TiN has a metallic behavior as expected, showing a *PTC* (*Positive Temperature Coefficient*) and thus an increase in the resistance because of self heating. Above this voltage, a sudden decrease in the resistance value suggests a variation in the physical structure of the heaters, for this reason a further analysis at higher voltages was performed.

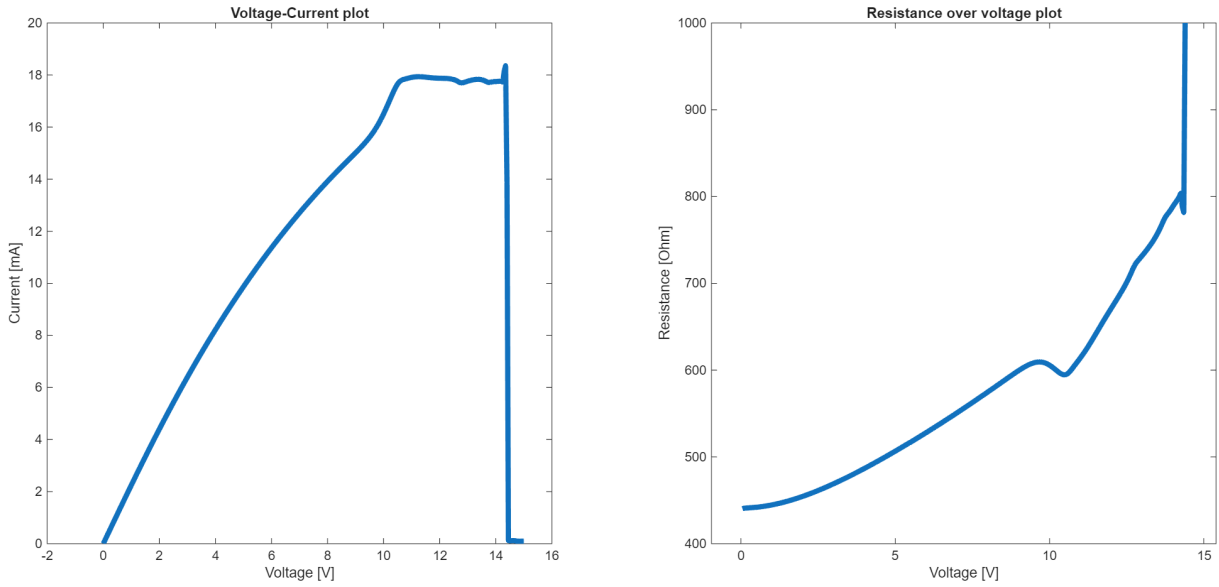


Figure 5.7: On the left, the current-voltage plot for the $100\mu m$ heaters between 0 and 15V. On the right the corresponding resistance. The much lower resistance along all the sweep clearly shows the wear of the component.

As expected, the change in resistance due to the previous measurement can be seen from the beginning of the curve, with a resistance starting at 440Ω , more than 140Ω lower with respect to the previous measurement.

At 7.5V and 8V, voltages at the boundary of the safe operating range for the resistor (at $\approx 8.35V$ an inversion in the resistance trend can be seen), the behavior of the resistor does not change significantly over the 5 minutes of the simulation. Nevertheless, the decrease of resistance over time, along with a lower starting resistance evaluated during the second measurement at 8V, suggests a possible wear in the resistance (figure 5.9).

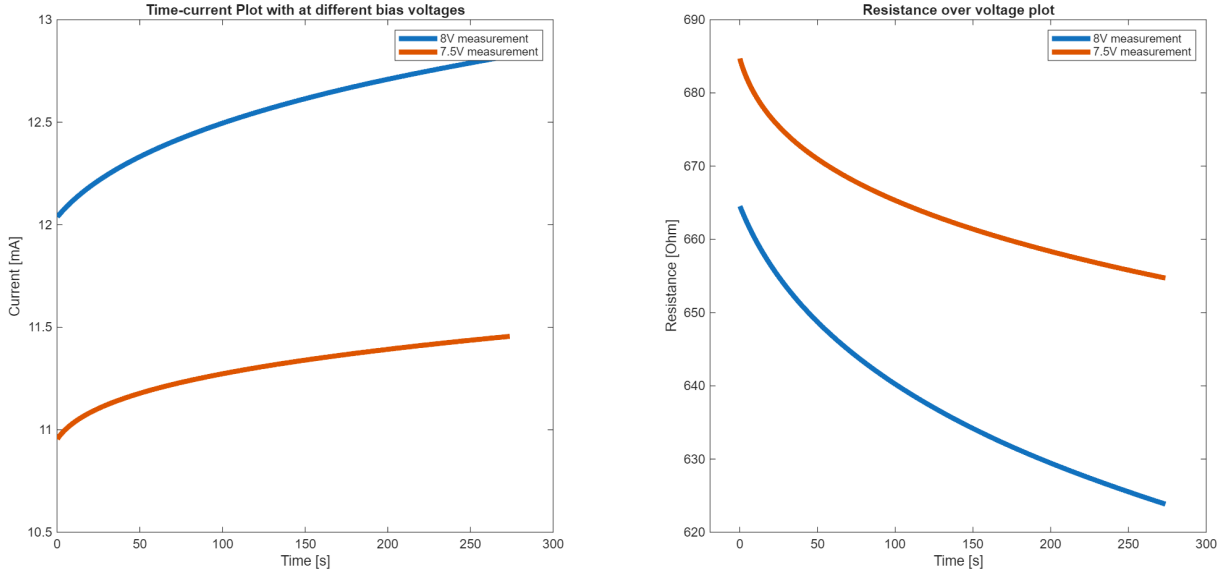


Figure 5.8: Plot over time, at fixed voltages, of the resistance and the current in the component. The voltages chosen are at the boundaries of the safe operation for the heater.

5.1.2. $60\mu\text{m}$ Heaters

The same measurement was then performed for the shorter $60\mu\text{m}$ heaters, with voltages further increasing to investigate the resulting wear on the actuator.

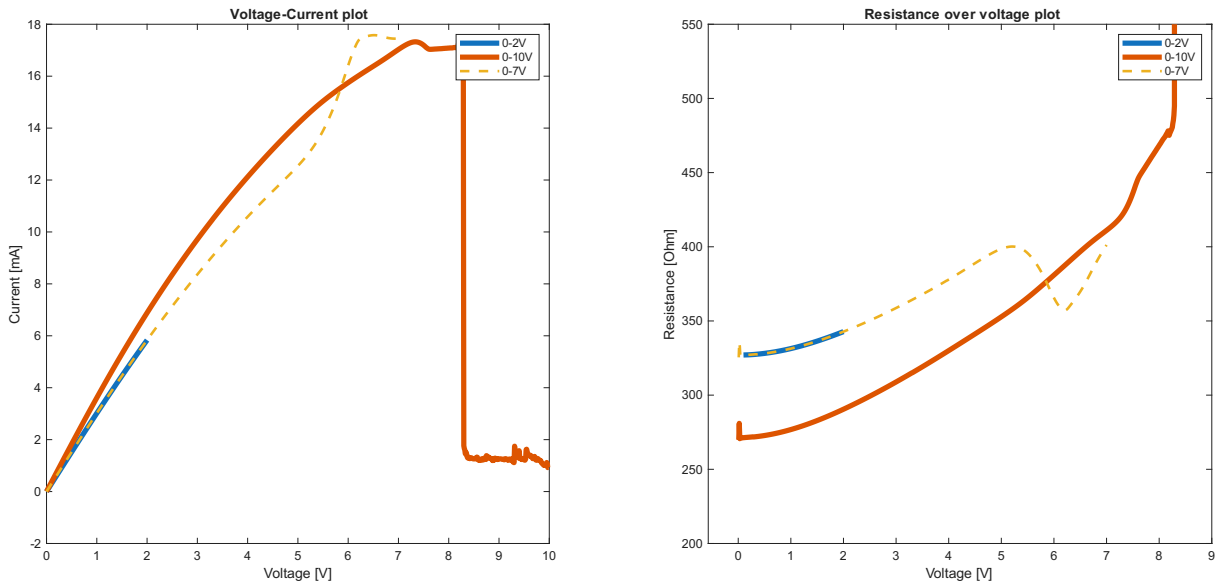


Figure 5.9: Results of the three sweeps at increasing voltages, showing no wear in the low range of heater dynamics as can be seen from the perfect overlap between blue and yellow line up to 2V.

As for the longer heaters, a PTC behavior is present up to $\approx 12mA$, followed by a change in the resistance plot concavity that marks a change in the resistive behavior of the component. The last sweep performed, the one between 0V and 10V, shows a 17.5% lower starting resistance, suggesting an irreversible change in the actuator structure.

A time-prolonged measurement was then performed, with voltages ranging from 4V to 10V.

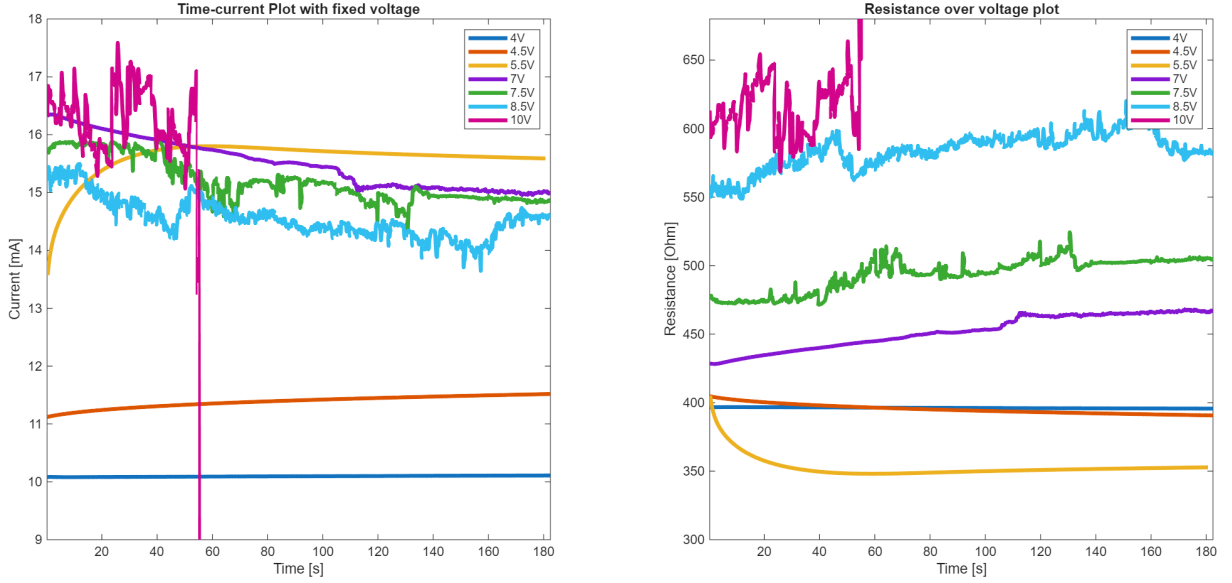


Figure 5.10: Measurement over time for $60\mu m$ heaters with incremental voltages. No significant change no significant change can be appreciated up to 4.5V.

As can be seen from Figure 5.10, up to 4.5V no significant variation in the resistance is shown, with the first relevant drop visible at 5.5V and a clearly altered behavior after this value. The maximum achievable current with both long and short heaters is about $18mA$.

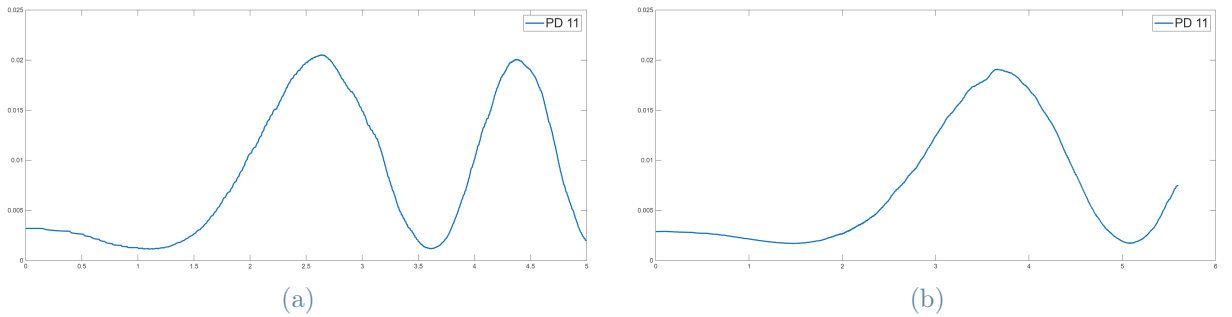


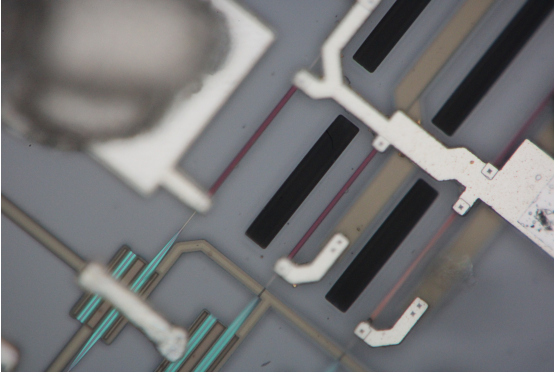
Figure 5.11: a) 0V – 5V sweep of a $60\mu m$ heater b) 0V – 5.6V sweep of a $100\mu m$ heater with incident light.

The behavior was then checked optically, to see if the required phase shift is achievable

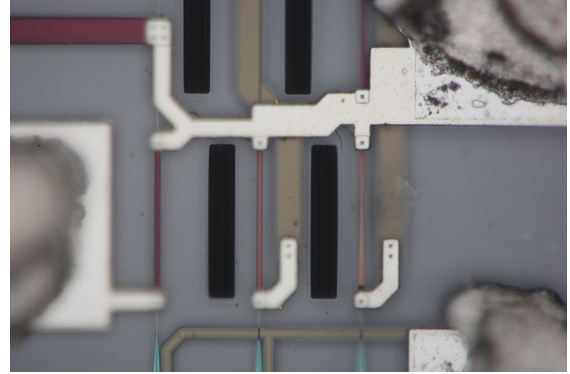
in a safe voltage range, and accounting that, for the control loop to properly work, a 3π phase shift is needed. Light was fed into the system, and separately the two different sized heaters were swept to achieve the sinusoidal shape in equation 3.9. The sweep was performed linearly in power, because of the linear correlation between the electrical power supplied to the heater and the resulting phase shift. The measured optical signal (fig. 5.12) was then used to link the phase shift applied to the corresponding voltage as shown in the following table.

Heater size	Power per π	Voltage per 2π	Voltage per 3π
$60\ \mu\text{m}$	$\approx 16\ \text{mW}$	$\approx 3.42\ \text{V}$	$\approx 4.3\ \text{V}$
$100\ \mu\text{m}$	$\approx 18.5\ \text{mW}$	$\approx 4.87\ \text{V}$	$\approx 6.05\ \text{V}$

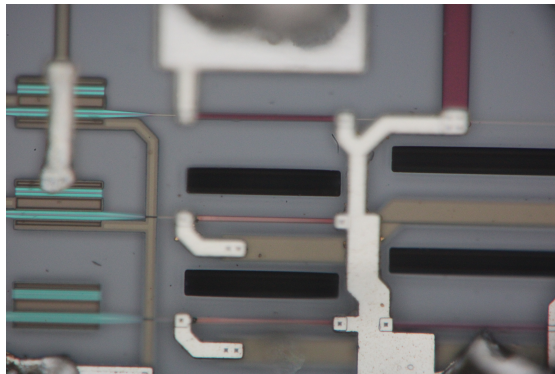
Table 5.2: Voltage-Phase description of the heaters, with all values well within the proper working range and thus ensuring predictable behavior.



(a) Bottom heater broken, middle and top heater untouched.



(b) Bottom heater broken, top heater untouched and middle heater after a 5.5V stress test.



(c) Bottom and middle heater broken, top heater untouched.

Figure 5.12: We can see how from a) to c) the colour of the middle heater changes, showing a physical change in the material.

5.2. Transparent photodiodes non linearity

Photodiodes non linearity has to be accounted when using them. This effect is particularly evident for higher input powers, where the SSA (surface state absorption) effect is more relevant.

To show and evaluate its effect, the same amount of light power was fed at the inputs of a MZI. Was then performed a sweep of the input phase shifter, resulting in a change in the power split after the first coupler following the equation:

$$PowSplit_{low} = \frac{1}{2} + \frac{I_1 I_2}{I_1^2 + I_2^2} \sin(\varphi) = \frac{1}{2} + \frac{I_1 I_2}{P_{TOT}} \sin(\varphi)$$

Where φ is the phase shift we are applying. This power shift is referred to the lower branch of the MZI after the first coupler.

Taking as reference figure 4.16, the measurement in-between the two couplers will take place by means of PD_k , while the overall output power is the result of $PD_j + PD_i$.

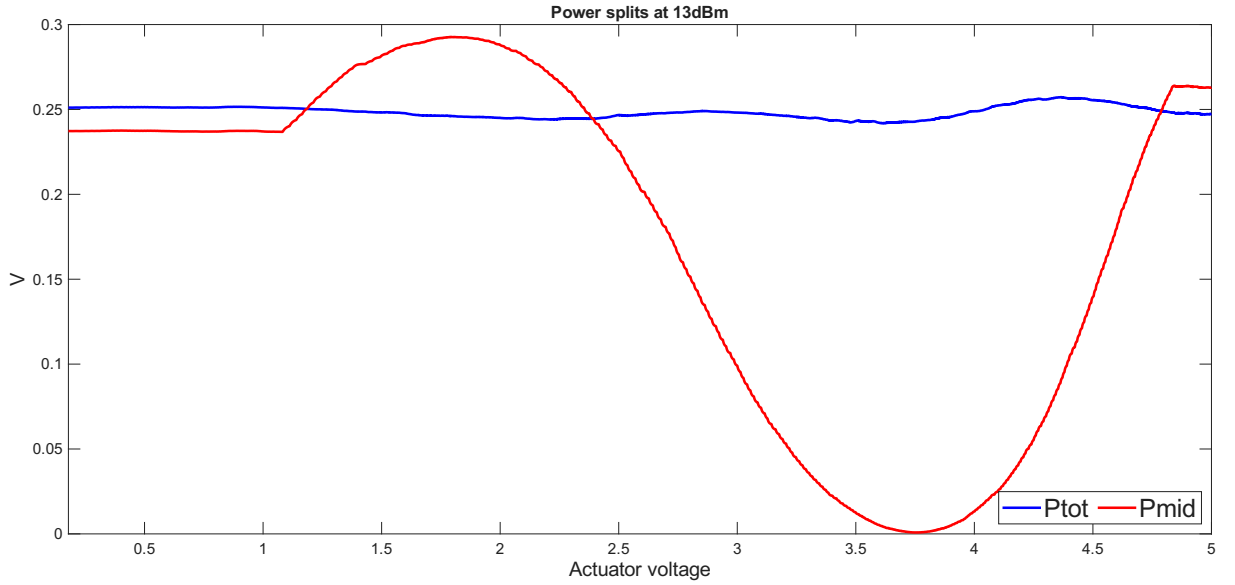


Figure 5.13: Internal power split measurement with respect to the overall power at 13dBm. It can be seen how the maximum signal read by the single internal photodiode at its maximums seems $\approx 17.26\%$ larger than the overall output signal.

The non linear responsivity is evident looking at the internal photodiode when the maximum available optical signal of 13dBm is fed to the system. The internal photodiode sees the sinusoidal shape induced by the input heater sweep, and therefore will experience

more nonlinearity due to high signal levels, making it appear as if $\approx 17\%$ of the light entering in the MZI is flowing through it. The two output ones instead sees a signal swing around half dynamics, reducing the effect of non linearity which is only visible in the small oscillation of the blue line, which shows a variation of approximately 3% from the average output power.

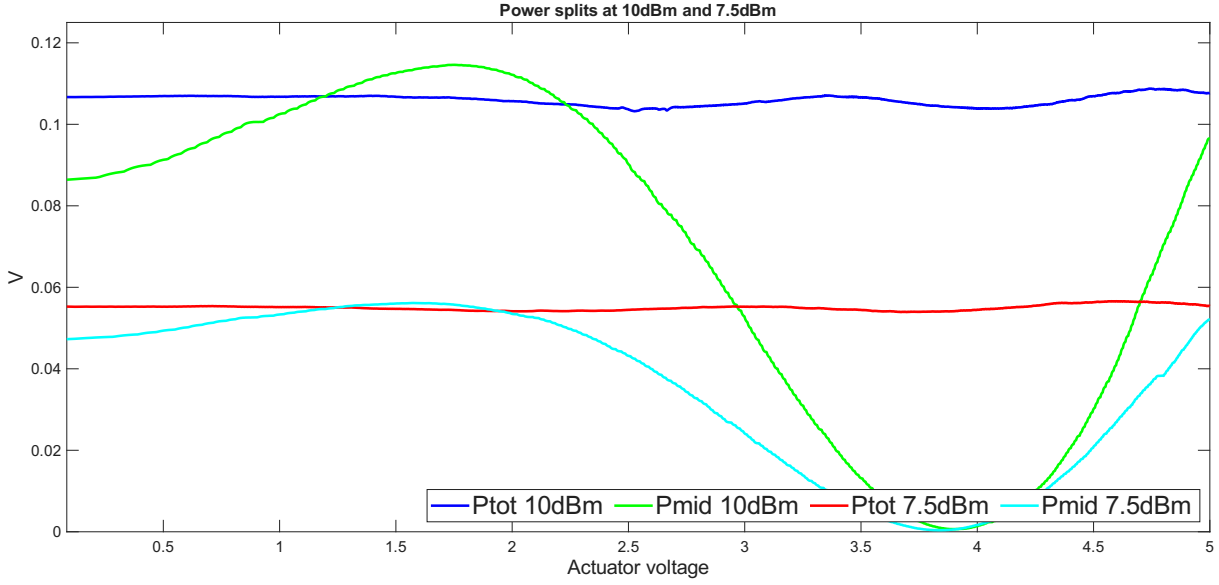


Figure 5.14: Internal power split measurement with respect to the overall power at 10dBm and 7.5dBm. The overshoot visible in the internal photodiode is $\approx 8.86\%$ in this case for 10dBm and $\approx 2.6\%$ for 7.5dBm.

Referring now to figure 5.14, reducing input power by 3dB the maximum signal flowing through PD_k is lower by a factor 2 in power, and thus resulting in a smaller non linearity. In this case we find the maximum value measured by PD_k to be 8.86% larger than the total average power in the device. Also the oscillation of the total power shows a smaller variation, which is about the 1.5% in this case.

The last measurement was performed at 7.5dBm, the lowest possible by our laser generator. In this condition the maximum measured power is limited to 2.6% more than the average total power, with an output power oscillation of $\approx 1\%$. The progressive reduction in the overshoot, following the reduction of the input power, suggests the use of a lower input powers for better results and for a more reliable photodiodes readout.

To evaluate now how the input power change affects the setting of the power splits along the mesh, an arbitrary matrix was chosen and set with the control at 7.5dBm. The control was then paused at the point of interest, and the input power was changed to 13dBm, 10dBm and finally back to 7.5dBm to evaluate the drift of the various points in time.

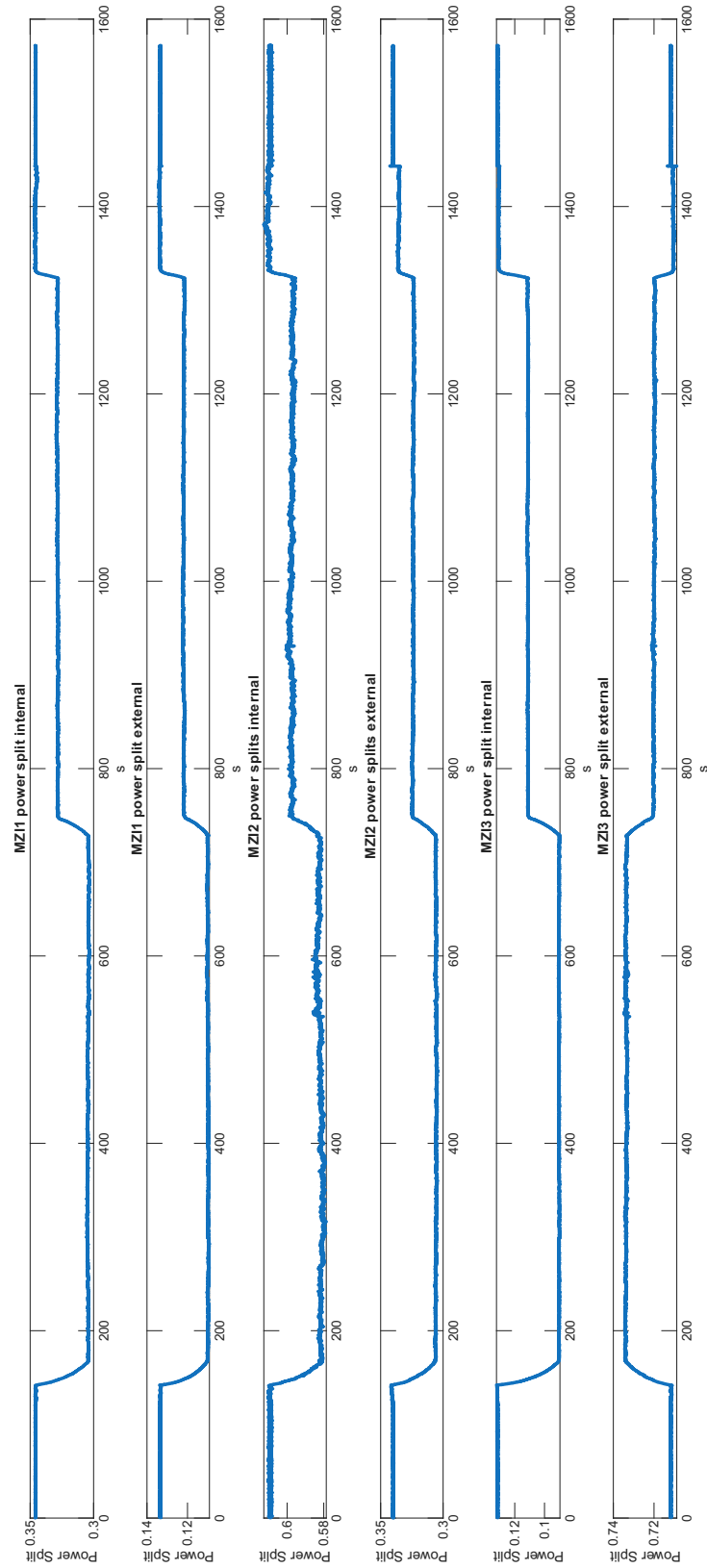


Figure 5.15: Power split measurement over 26 minutes for three different input powers: $13dBm$, $10dBm$ and $7.5dBm$. The represented timescale is in seconds.

The variations were then compared with the optical signal present in the photodiodes during the measurement, to control if the largest variations were associated with the highest signals also in this case:

Photodiode	13dBm	10dBm	7.5dBm after time
<i>Voltages for each photodiode</i>			
PD9	37.79 mV	17.8 mV	10 mV
PD6	13.6 mV	6.5 mV	4 mV
PD3	110.66 mV	47 mV	25 mV
PD12	38 mV	16.7 mV	10 mV
PD11	17.8 mV	9 mV	5.4 mV
PD5	40.4 mV	18.8 mV	10.5 mV
PD8	10.6 mV	5.8 mV	3.7 mV
PD13	87 mV	37 mV	20 mV
PD7	31.6 mV	14.5 mV	8 mV
<i>% variation from 7.5dBm</i>			
PD9	3.7707	1.78	0.1
PD6	3.5219	1.625	0.3
PD3	4.3979	1.88	0.2
PD12	3.4390	1.67	0.7
PD11	3.2439	1.67	1.6
PD5	3.7999	1.79	0.8
PD8	2.8720	1.56	0.5
PD13	4.3198	1.85	0
PD7	3.8749	1.81	0.6

Table 5.3: Power setting variation between different powers

It can be appreciated how we have a larger percentage variation where the signal is higher, following what just shown in figure 5.13 and 5.14.

5.3. Phase measurement and matrix setting

Once analyzed the physical properties of the devices adopted for the control of the matrix, Vector-Matrix Multiplication (VMM) tasks were performed taking into account the know-how just acquired, to verify the operational behavior of the system under real operating conditions.

As seen in 3.2.2, to properly set the matrix we should know with the highest possible precision the input vector, both in phase and in power. Power is easily retrieved from

the input photodiodes, the same exploited by the control to set the desired power splits with the input generator. Phase instead is not as straightforward, a rough estimate can be deduced by looking at the reached power splits and linking them to the theoretical transfer function of the MZI. This, although, doesn't account for non-ideality, inevitable in real implementations. An in-line phase measurement is therefore necessary.

5.3.1. MZI phase measurement

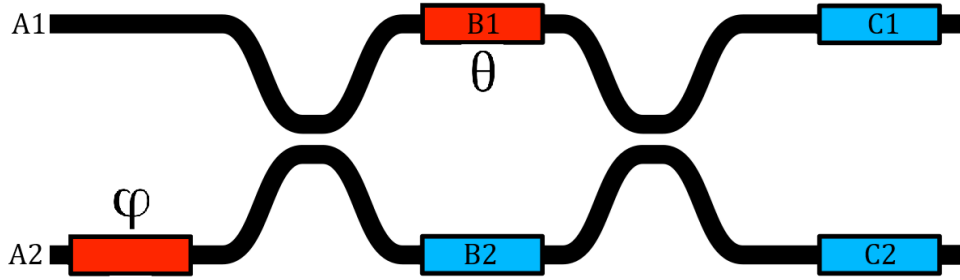


Figure 5.16: MZI schematics, showing the positioning of heaters (red) and photodiodes (blue) along the device.

To properly describe the phase at the input, a measurement of its values along all the possible input power splits is performed using a MZI as a coherent phase detector [6].

By calling χ_c the differential phase between the two coherent beams arriving in A_1 and A_2 , we can express the differential phase as:

$$\chi_c = \arctan\left(\frac{P_{B1} - P_{B2}}{P_{C1} - P_{C2}}\right)$$

Given that in our case P_{B1} isn't retrievable, because we have a photodiode only in the bottom branch while in the upper one we have an heater, we will express it as a function of the overall power and of the other photodiode as follows:

$$P_{B1} = P_{TOT} - P_{B2} = P_{C1} + P_{C2} - P_{B2}$$

The resulting equation is the following:

$$\chi_c = \arctan\left(\frac{P_{C1} + P_{C2} - 2P_{B2}}{P_{C1} - P_{C2}}\right) \quad (5.1)$$

This is true only under particular conditions, in fact, for the device to properly work, θ must be equal to $\pi/2$.

The phase shift is obtained by feeding port A_1 and leveraging the fact that a 50/50 output power split corresponds to $\theta = \pi/2 + m\pi$. The *PowSplit* control mode is then used to lock the system at this balanced condition, with the convergence to the desired working point ensured by choosing the correct derivative.

A preliminary phase measurement to evaluate the precision of this configuration was performed adopting the structure depicted in figure 3.9, the same one adopted to generate the inputs, to perform the phase shift under exam, and MZI_1 , the first of the optical core, as a phase meter.

Two phase measurements were performed, one for each ψ_1 half period, and along a full period in the ψ_2 direction. Thanks to this we can check how well our phase meter follows the $\frac{\psi_2}{2}$ ramp, both for low (first half of the period) and high (second half of the period) actuation voltages, reporting also the measured signal amplitude with respect to the theoretical one.

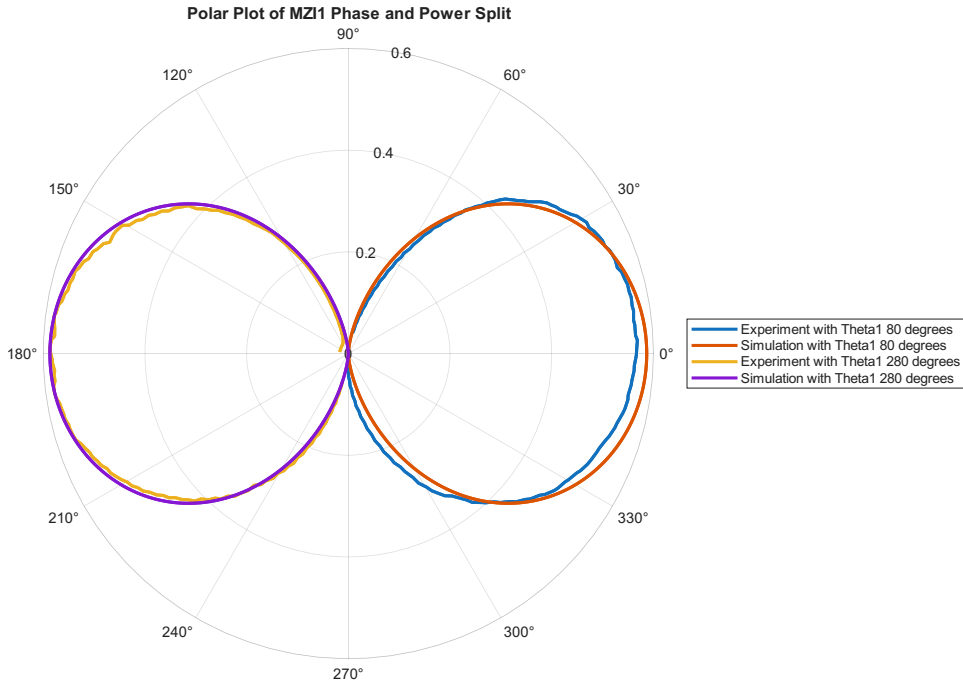


Figure 5.17: Polar plot showing a comparison between the simulated result, represented by the orange and the purple curves, and the experimental results.

The result is shown in figure 5.17, where the largest error between real and experimental measurement is present at the high and low power points of the curve. The largest error is estimated to be $\approx 4.5\%$ in amplitude for the right curve.

Evaluating instead only the capability of our meter to measure the phase, after removing

eventual offsets that are due to the physical implementation of the device, the resulting phase measurement is the one depicted in picture 5.18, showing an RMS error along all the measurement of 3.4%.

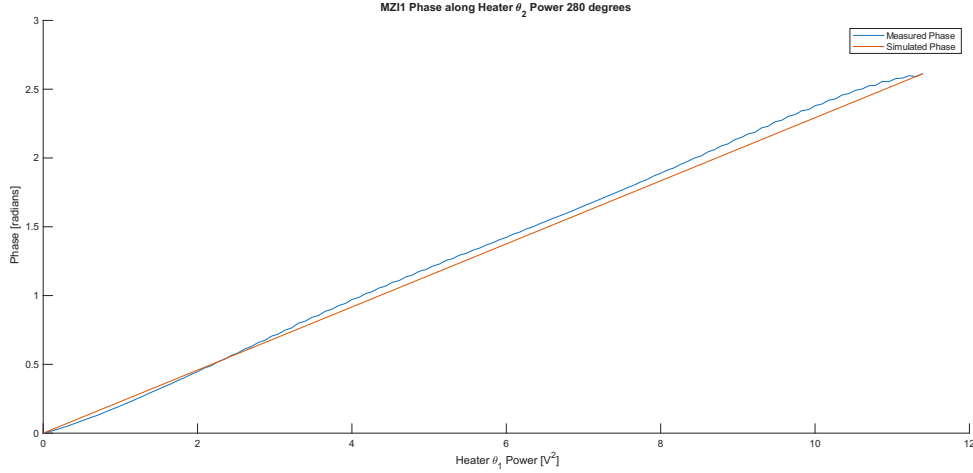


Figure 5.18: Phase measurement section along ψ_2 compared to the simulated measurement.

Now that we have discussed the phase-metering capabilities of our setup, we must use them to describe our input with the highest possible precision available.

Looking at figure 3.8, we see how the first MZI of the 3×3 matrix is positioned directly after In_2 and In_3 , for this reason we will use it as input phase meter. Once measured $\chi_{2,3}$, $\chi_{1,2}$ is obtained by assuming is to be 0 or π as resulting from simulation 3.12a. Unfortunately, because of the disposition of the MZIs inside of the mesh, a direct measurement of $\chi_{1,2}$ was not possible, thus resulting in some level of uncertainty on the input vector.

To include all the possible phases for our input vector, the phase measurement was performed by sweeping ψ_0 and ψ_2 from 0V to 3.6V while reading the phase with MZI_1 , and thus covering more than one period in phase for both, as visible from table 5.2 for $60\mu m$ heaters. This results in a three dimensional phase map, thanks to which we can link every possible combination in power at the input to the corresponding phase, and this by looking at the voltage of the heaters for that specific combination. The result is reported in figure 5.3.4.

A simulation of the expected phase is presented in figure 3.12b, where a change of π is visible between the first and the second half period along θ_1 , which in our case corresponds to the phase shift induced by ψ_0 . Along the other direction instead, we see a ramp proportional to $\frac{\theta_2}{2}$, and thus to the phase shift induced by ψ_2 .

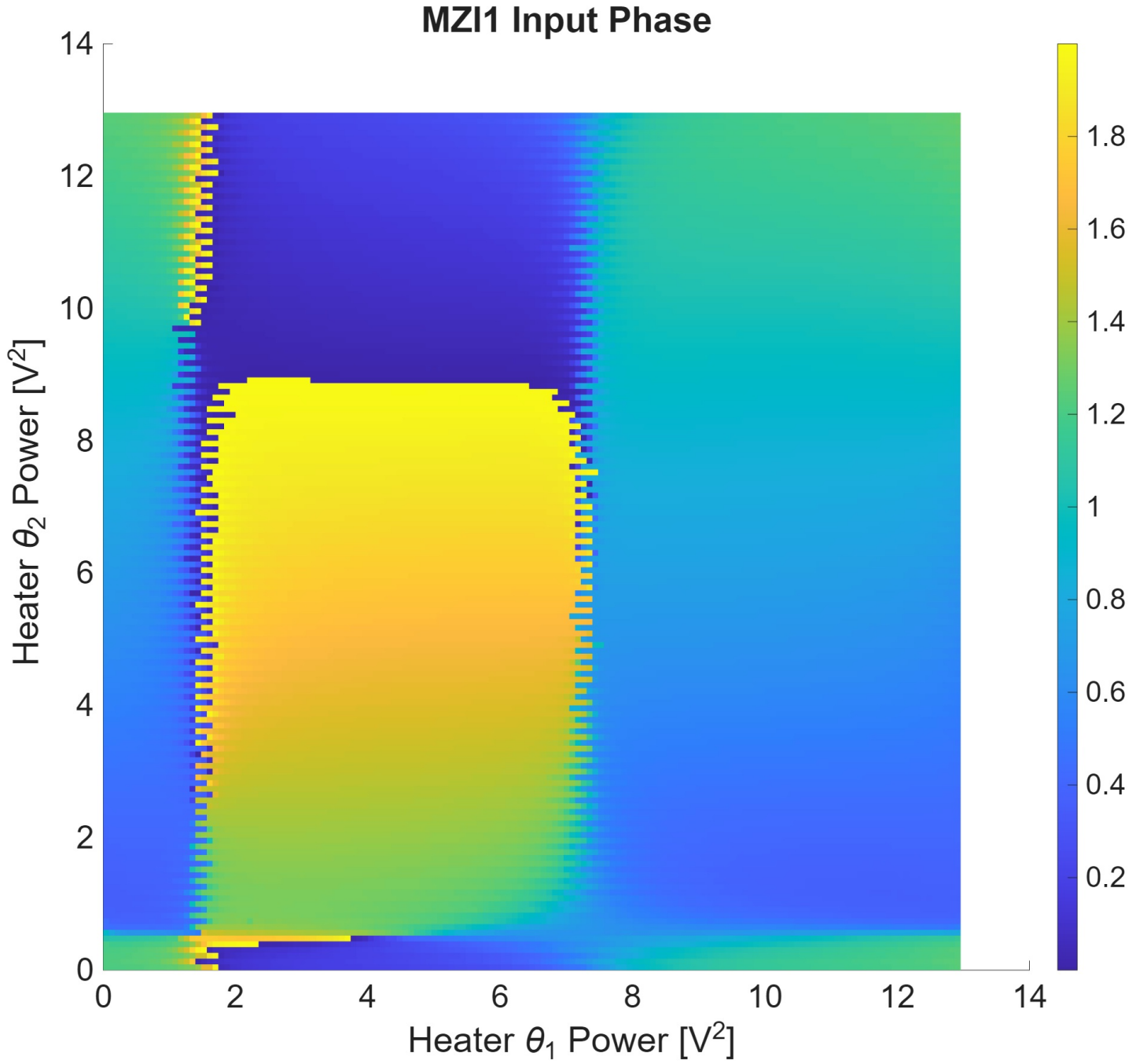


Figure 5.19: Three dimensional representation of MZI_1 input phase. The two axis show the squared voltage applied to the heaters of the input generator. It is present an offset in the origin along both directions with respect to the simulation results due to non-ideality of the physical implementation.

5.3.2. Matrix setting and precision

To validate optical computing tasks, an arbitrary matrix was set with a generic input vector used as reference. Once convergence is achieved for each one of the 6 integrators associated with matrix phase shifters, the system is paused and each heater is frozen with the voltage reached when the control was on.

The voltage applied on ψ_0 and ψ_2 , used to set the input vector during matrix setting, are then exploited to get the reference phase at which our system has been set by looking at Figure 5.3.4. When setting the matrix we didn't account for input phase, and thus we didn't compensate it, because by looking only at the differential phase variations with respect to an initial reference, we avoid any offsets due to the single measurement.

In our case, the input vector generator was configured to set $IN = \begin{bmatrix} 0.3333 & 0.3333 & 0.3333 \end{bmatrix}^T$ as initial power split, where all the powers are normalized with respect to the overall power entering in the mesh. This vector can be represented by four different voltage combinations in the 2π by 2π phase shift range, as presented in figure 3.11. The one chosen for these measurements, and imposed through an appropriate selection of derivative, is the one found in the right bottom π by π sub-plane of the phase map (see Figure 3.11 for reference).

The matrix implemented for validation is the following:

MZI	Phi	Theta
<i>Phase setting for each point</i>		
MZI_1	1.1π	0.3π
MZI_2	0.25π	1.3π
MZI_3	0.8π	1.4π

Table 5.4: Phase setting for each point of the matrix. φ heaters are the ones before the first coupler, while θ heaters are the ones between the two couplers.

The validation procedure is performed following two main methodologies. The first one consists in setting the matrix following the methodology reported before, and checking the accuracy of the *VMM* by manually changing different inputs and comparing experimental and expected results. The second one follows a methodology similar to the one used when measuring the phase map. The matrix is set in the same way as before, it is therefore kept on hold and a sweep of heater ψ_0 and ψ_2 is performed. We get in this way six different three-dimensional representations, one for each setting point, that are then compared by using the phase map and the power data extrapolated from it, with the expected results.

5.3.3. Manual input vector setting

The first measurement is performed with a maximum light reaching our photodiodes in the order of $1dBm$, corresponding to the optimum optical power in terms of precision and linearity.

The matrix in table 5.4 is the one chosen to be set, while at the input we have $IN = \begin{bmatrix} 0.3333 & 0.3333 & 0.3333 \end{bmatrix}^T$ as base vector. With this configuration the voltages applied to the input generator heaters are:

$$\psi_0(V) = 3.22V \Rightarrow 10.34V^2$$

$$\psi_2(V) = 1.75V \Rightarrow 3.06V^2$$

Based on the phase map shown in Figure 5.3.4, the phase of the input vector was evaluated as $\chi_0 = 0.487\pi$ rad. This value was then subtracted from the original phase map to obtain a calibrated distribution, effectively using χ_0 as the new reference level.

	MZI1 Pow split int	MZI1 Pow split ext	MZI2 Pow split int	MZI2 Pow split ext	MZI3 Pow split int	MZI3 Pow split ext
<i>Vector₀(0.3333/0.3333/0.3333)</i>						
Real	0.34668	0.15482	0.60361	0.2949	0.12314	0.71567
Target	0.34669	0.15483	0.60363	0.295	0.12314	0.71567
Error(%)	0.001	0.001	0.01	0	0	0
<i>Vector₁(0.25/0.25/0.5)</i>						
Real	0.3567	0.2669	0.5336	0.1365	0.0839	0.7188
Target	0.3643	0.2724	0.5647	0.1086	0.0886	0.7274
Error(%)	0.76	0.55	3.11	-2.79	0.47	0.86
<i>Vector₂(0.45/0.2/0.35)</i>						
Real	0.2883	0.2849	0.6978	0.2752	0.0939	0.7698
Target	0.2851	0.2762	0.6721	0.2928	0.0988	0.7805
Error(%)	-0.32	-0.87	-2.57	1.76	0.49	1.07
<i>Vector₃(0.45/0.4/0.15)</i>						
Real	0.3511	0.0576	0.7074	0.589	0.2012	0.7021
Target	0.3399	0.0422	0.6439	0.624	0.1895	0.6965
Error(%)	-1.12	-1.54	-6.35	3.5	-1.17	-0.56
<i>Vector₄(0.2/0.55/0.25)</i>						
Real	0.4434	0.0458	0.4938	0.4378	0.2336	0.5659
Target	0.4541	0.0321	0.4922	0.4584	0.222	0.5629
Error(%)	1.07	-1.37	-0.16	2.06	-1.16	-0.3
<i>Vector₅(0.35/0.15/0.5)</i>						
Real	0.3025	0.4032	0.6548	0.1473	0.0781	0.7639
Target	0.3183	0.3907	0.6507	0.1403	0.0853	0.7763
Error(%)	1.58	-1.25	-0.41	-0.7	0.72	1.24
<i>Vector₆(0.1/0.5/0.4)</i>						
Real	0.5094	0.092	0.2871	0.2023	0.1856	0.498
Target	0.5295	0.0889	0.2887	0.1556	0.1656	0.5133
Error(%)	2.01	-0.31	0.16	-4.67	-2	1.53
<i>Vector₇(0.6/0.25/0.15)</i>						
Real	0.2894	0.1276	0.7465	0.5535	0.1412	0.769
Target	0.2666	0.1079	0.6656	0.5872	0.1309	0.7779
Error(%)	-2.28	-1.97	-8.09	3.37	-1.03	0.89
<i>Vector₈(0.8/0.1/0.1)</i>						
Real	0.2348	0.2572	0.7751	0.6045	0.1256	0.7842
Target	0.195	0.2106	0.6581	0.6281	0.1139	0.8141
Error(%)	-3.98	-4.66	-11.7	2.36	-1.17	2.99
RMS ERROR(%)	1.96	2.02	5.69	2.88	1.12	1.41

Table 5.5: Power split results along the mesh measured with a maximum signal of $1dBm$ readen by the photodiodes.

To check the fidelity of the matrix with respect to the theoretical calculations, the system is paused. The integrators are turned off, and the convergence voltage previously found is kept on the heaters to fix the calculated matrix. Eight different input power splits are then provided to the matrix, along with the phase difference between them and χ_0 , in order to have a precise input description both in phase and power.

For each input vector, a theoretical calculation of the expected power splits is performed by the control software, in order to have a reference against which to compare the experimental results. The two results are then compared, and the RMS percentage error is calculated for each point as reported in figure 5.5.

	MZI1 Pow split int	MZI1 Pow split ext	MZI2 Pow split int	MZI2 Pow split ext	MZI3 Pow split int	MZI3 Pow split ext
<i>Vector</i> ₀ (0.3333/0.3333/0.3333)						
Real	0.34729	0.16471	0.60098	0.27533	0.11960	0.71725
Target	0.34731	0.16473	0.60101	0.27536	0.11961	0.71727
Error(%)	0.002	0.002	0.003	0.003	0.001	0.002
<i>Vector</i> ₁ (0.25/0.25/0.5)						
Real	0.3684	0.278	0.525	0.1347	0.0843	0.7324
Target	0.3649	0.2844	0.5636	0.0954	0.0865	0.7289
Error(%)	-0.35	0.64	3.86	-3.93	0.22	-0.35
<i>Vector</i> ₂ (0.45/0.2/0.35)						
Real	0.2974	0.2923	0.7067	0.2667	0.1005	0.7955
Target	0.2939	0.2849	0.6619	0.2748	0.0956	0.7778
Error(%)	-0.35	-0.74	-4.48	0.81	-0.49	-1.77
<i>Vector</i> ₃ (0.45/0.4/0.15)						
Real	0.3302	0.0589	0.7139	0.5662	0.1941	0.6882
Target	0.3322	0.0485	0.651	0.6062	0.1865	0.7019
Error(%)	0.2	-1.04	-6.29	4	-0.76	1.37
<i>Vector</i> ₄ (0.2/0.55/0.25)						
Real	0.4269	0.0503	0.4773	0.4021	0.2049	0.5376
Target	0.4848	0.0349	0.4250	0.4473	0.2081	0.5458
Error(%)	5.79	-1.54	-5.23	4.52	0.32	0.82
<i>Vector</i> ₅ (0.35/0.15/0.5)						
Real	0.3143	0.4177	0.6504	0.1473	0.0858	0.8025
Target	0.3107	0.4043	0.6666	0.1326	0.0874	0.7802
Error(%)	-0.36	-1.34	1.62	-1.47	0.16	-2.23
<i>Vector</i> ₆ (0.1/0.5/0.4)						
Real	0.5061	0.1016	0.2841	0.1919	0.1528	0.4754
Target	0.5393	0.0966	0.2649	0.1481	0.1583	0.5112
Error(%)	3.32	-0.5	-1.92	-4.38	0.55	3.58
<i>Vector</i> ₇ (0.6/0.25/0.15)						
Real	0.276	0.1317	0.7724	0.543	0.1479	0.7801
Target	0.2471	0.1233	0.6836	0.5732	0.1364	0.7887
Error(%)	-2.89	-0.84	-8.88	3.02	-1.15	0.86
<i>Vector</i> ₈ (0.8/0.1/0.1)						
Real	0.2181	0.2608	0.8216	0.61	0.1347	0.8389
Target	0.1788	0.2293	0.6859	0.6352	0.0992	0.7947
Error(%)	-3.93	-3.15	-13.57	2.52	-3.55	-4.42
RMS ERROR(%)	2.93	1.46	6.81	3.34	1.38	2.34

Table 5.6: Power split results along the mesh and RMS error between theoretical and experimental power splits at $\approx 3.76dBm$.

To evaluate the influence of nonlinearities on precision, subsequent measurements were conducted using increased optical power levels, going from 1 *dBm* to ≈ 4 *dbm* at the photodiodes. As illustrated in Table 5.6, aside from the external power split of *MZI*₁, which exhibited a marginally lower RMS error across the eight inputs, the experimental results showed more pronounced deviations from theoretical values than the low-power results in Table 5.5. This trend aligns with the analysis in Section 5.2, corroborating the

link between a change in photodiode responsivity and diminished measurement accuracy.

As visible in both tables, the RMS error corresponding to the internal power split of MZI_2 is the highest. This can be explained by the approximate knowledge of phase difference between In_1 and In_2 . While χ_{23} was determined through experimental measurement, χ_{12} relied solely on simulation. The physical arrangement of the MZIs relative to the input generator effectively precludes a direct measurement of this specific phase relationship.

For the RMS error evaluation, only the results coming from vector changes one to eight are considered, excluding the measurement performed with active control. This to shows more clearly how precise was the system to reach the desired point.

Parameter	RMS %Error (1dBm)	RMS %Error (3.76dBm)	Δ
MZI1 Pow split int	1.96	2.93	0.97
MZI1 Pow split ext	2.02	1.46	-0.56
MZI2 Pow split int	5.69	6.81	1.12
MZI2 Pow split ext	2.89	3.34	0.45
MZI3 Pow split int	1.12	1.38	0.26
MZI3 Pow split ext	1.41	2.34	0.93

Table 5.7: Comparison and variation of RMS errors between the two configurations

5.3.4. Error map

For the second validation procedure, the matrix is set up in the same way as presented in the previous section, with the light power adopted to achieve the results visible in table 5.5. The matrix is then fixed in the same way as before, and a sweep of ψ_0 and ψ_2 between 0V and 3.6V is performed. These voltages are enough to reach $\approx 2.2\pi$, and were specifically chosen to be the same as those used for phase measurement so that a direct comparison could be made.

The sweep was performed linearly in power, to have a linear proportionality between actuation voltage and phase shift. The overall measurement, consisting of all the non-redundant obtainable combinations between the 150 equidistant points in power for each heater, lasts about 40 minutes. A sufficiently large time interval was thus left between one point and the next, ensuring an adequate convergence for each of the 22500 points and avoiding undesired transient effects.

While doing the sweep, all the traces from mesh photodiodes were saved, to be then used to evaluate signal amplitude and the power splits of interest along the matrix. Post processing is then done on these data, both the traces from input sweep and from the phase

measurement reported in figure are imported in a Matlab script. An input vector, with both amplitude and phase description, is generated for each point of the 150×150 matrix, starting from the phase measurement data, and accounting for the "baseline phase" corresponding to the vector used to set the matrix used for the sweep. These vectors are then fed to a theoretical simulator of the 3×3 matrix, which evaluates the expected power splits starting from these experimentally evaluated inputs. In the meantime, the traces from input sweep are used to evaluate the same power splits, resulting now from the direct measurements along the mesh. The two results, the theoretical one starting from the experimental phase measurement and the one coming from the matrix setting, are compared point to point to evaluate the discrepancies.

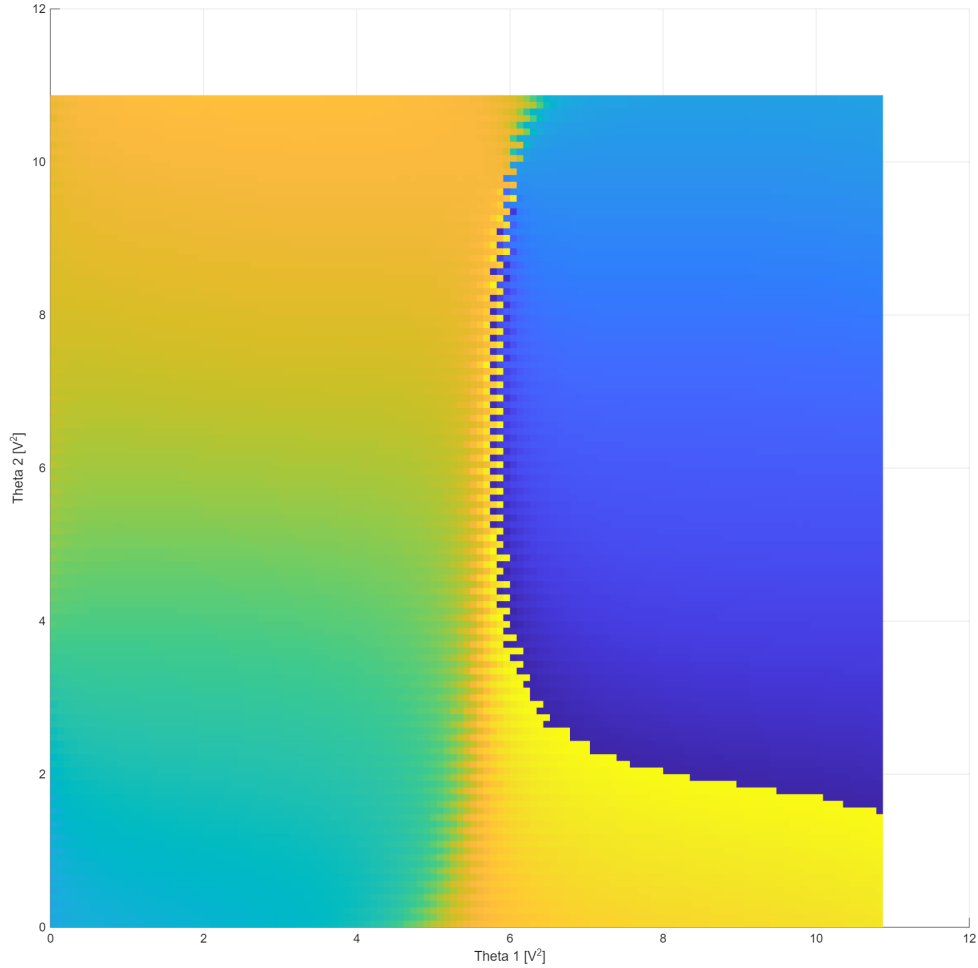


Figure 5.20: χ_{23} phase map after subtracting the base phase difference of $\chi_{23}(0) = 0.498968\pi$, corresponding to the one associated with input vector $IN = \begin{bmatrix} 0.3333 & 0.3333 & 0.3333 \end{bmatrix}^T$ used to set the matrix. The phase measurement was cropped to show a 2π by 2π map.

The results are the following, with the power split nomenclature referring to scheme 3.8:

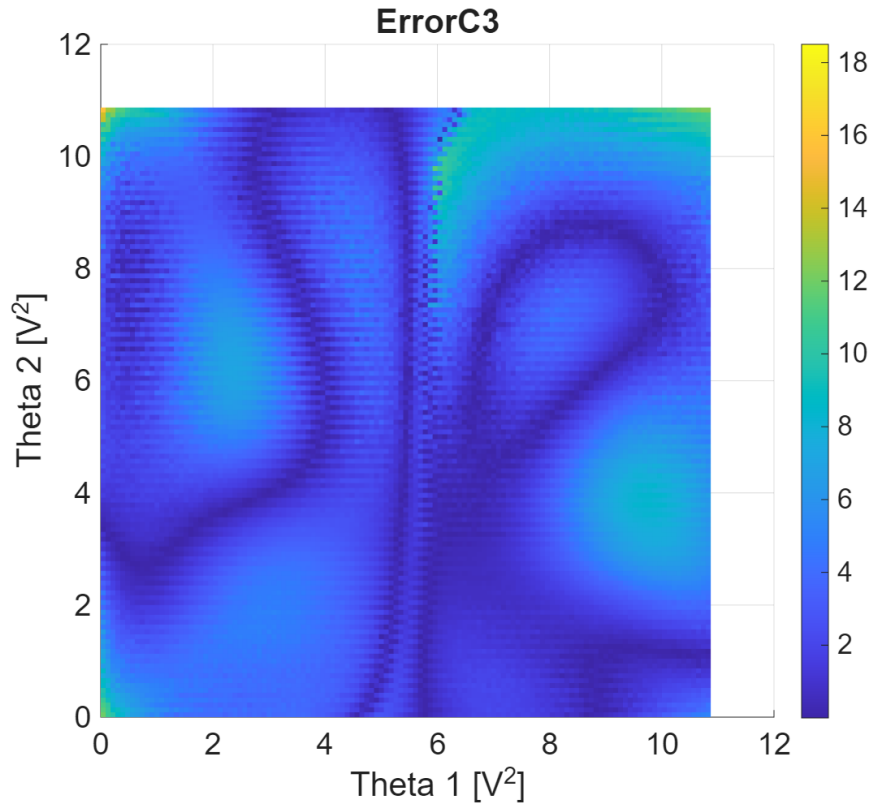


Figure 5.21: Percentage error for the internal power split of MZI_1

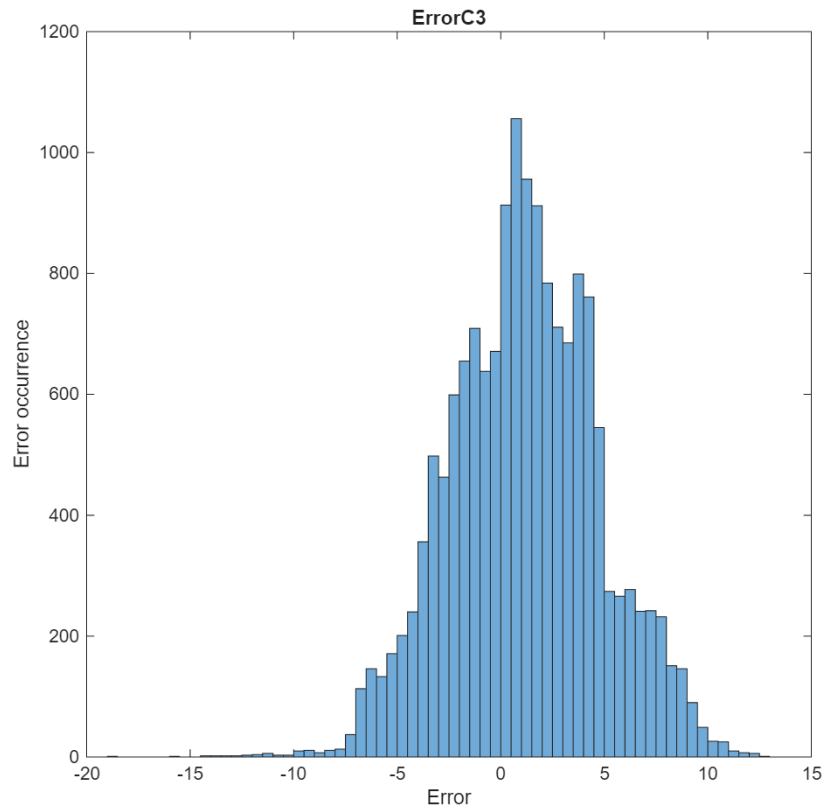


Figure 5.22: Histogram showing percentage error distribution for the internal power split of MZI_1

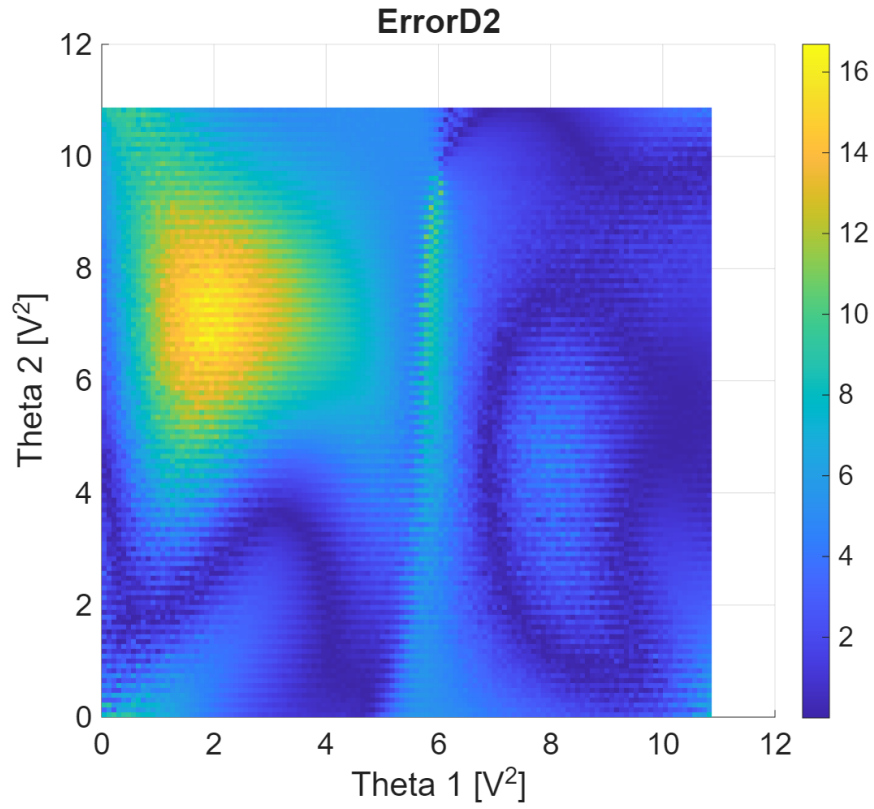


Figure 5.23: Percentage error for the output power split of MZI_1

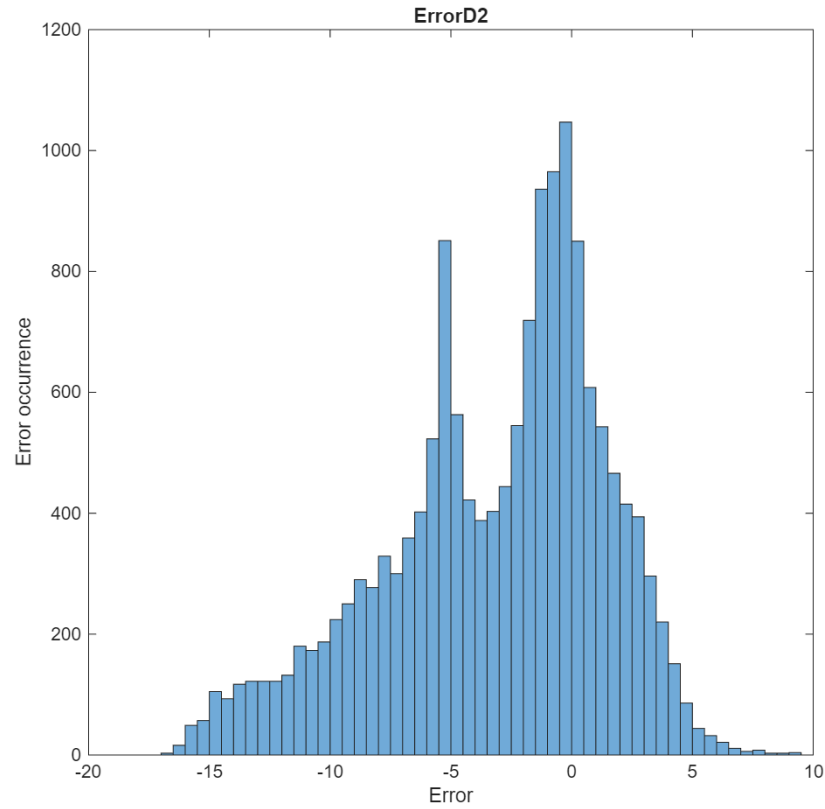


Figure 5.24: Histogram showing percentage error distribution for the output power split of MZI_1

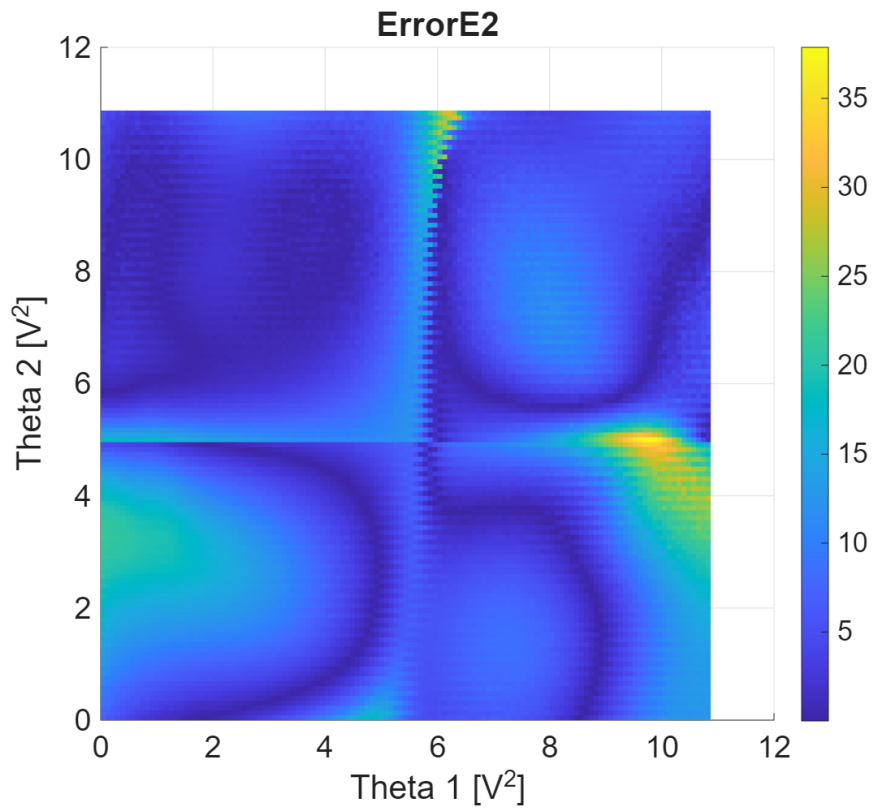


Figure 5.25: Percentage error for the internal power split of MZI_2

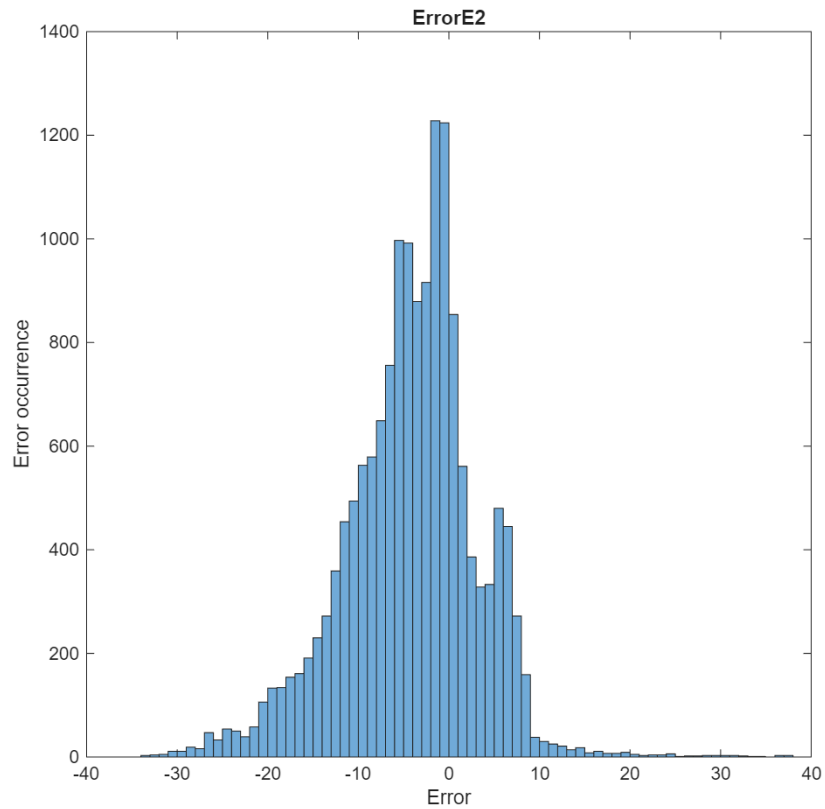


Figure 5.26: Histogram showing percentage error distribution for the internal power split of MZI_2

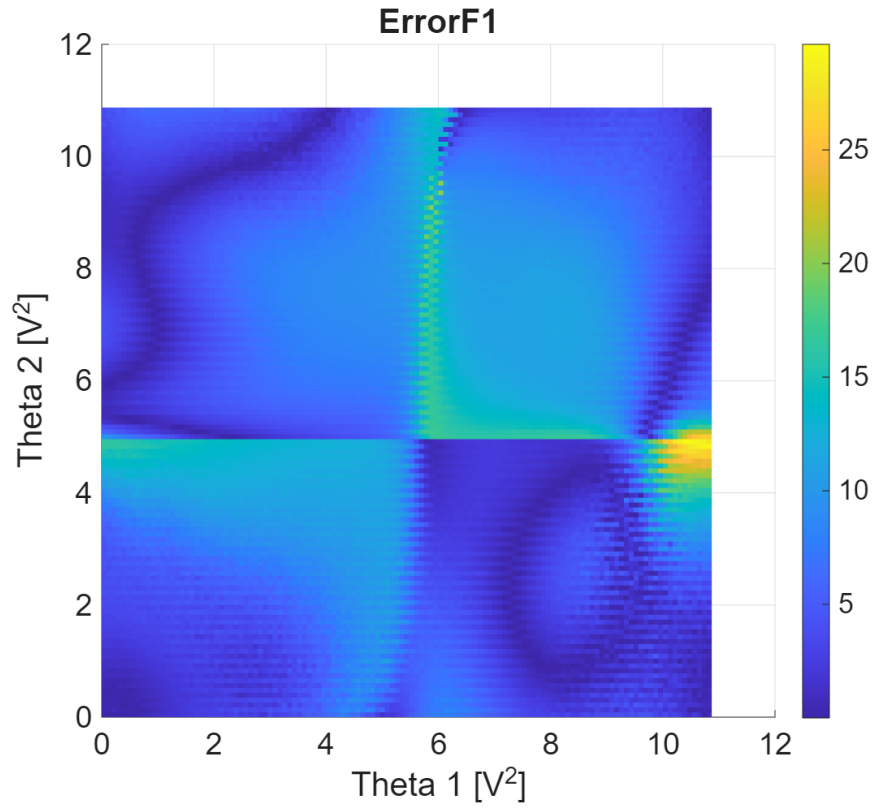


Figure 5.27: Percentage error for the output power split of MZI_2

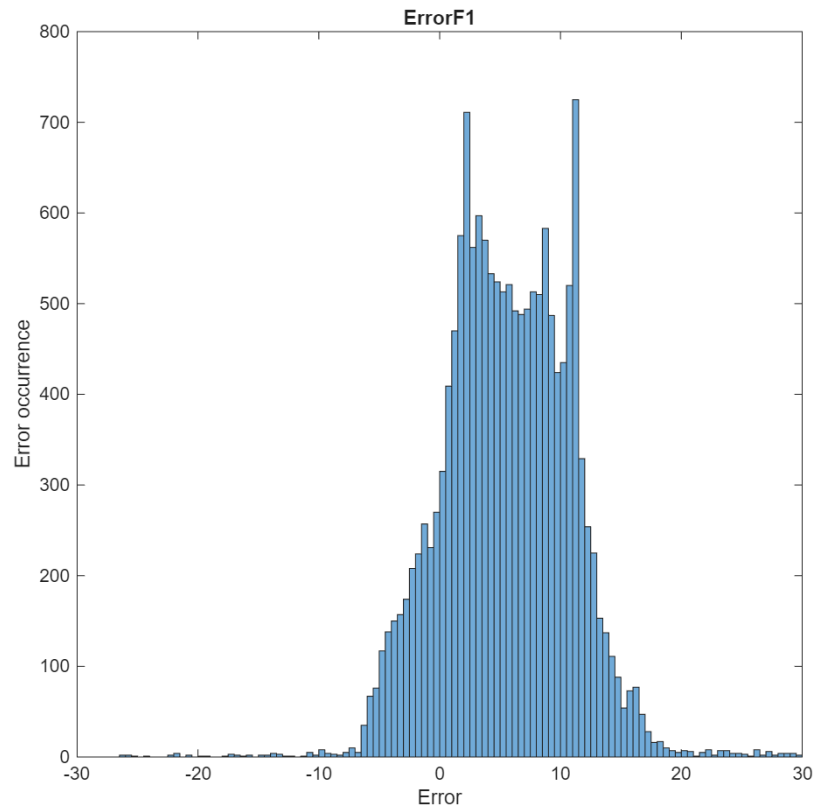


Figure 5.28: Histogram showing percentage error distribution for the output power split of MZI_2

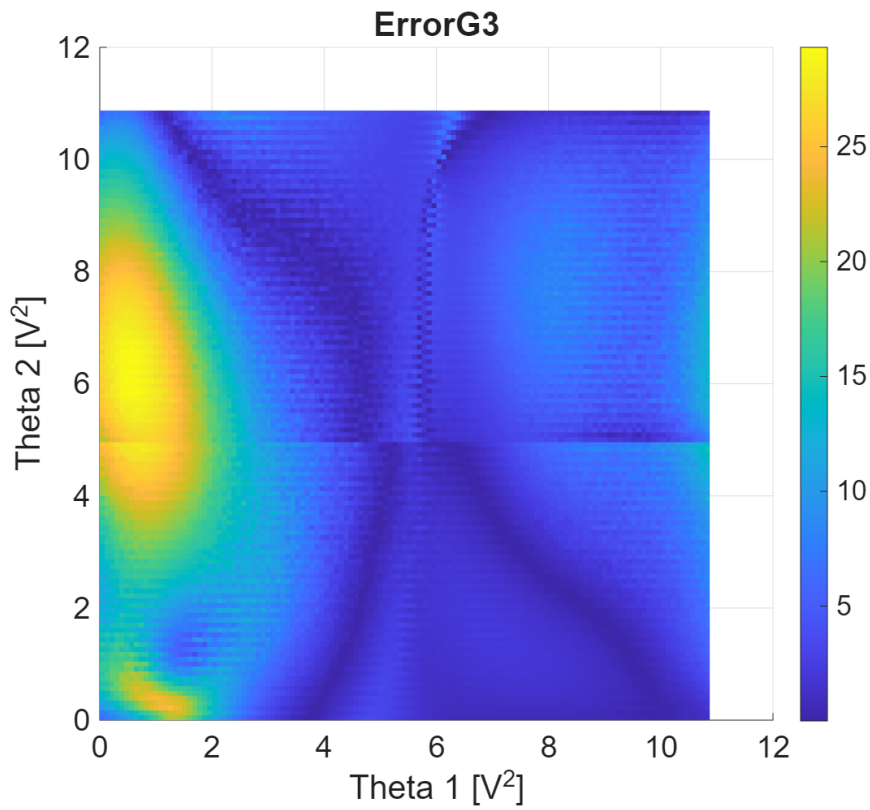


Figure 5.29: Percentage error for the internal power split of MZI_3

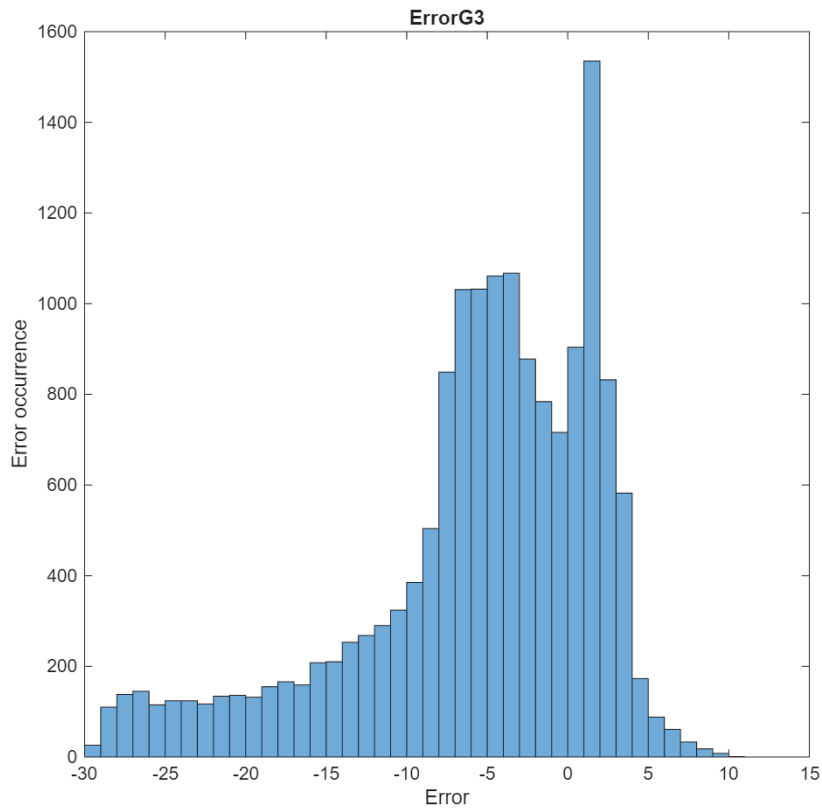


Figure 5.30: Histogram showing percentage error distribution for the internal power split of MZI_3

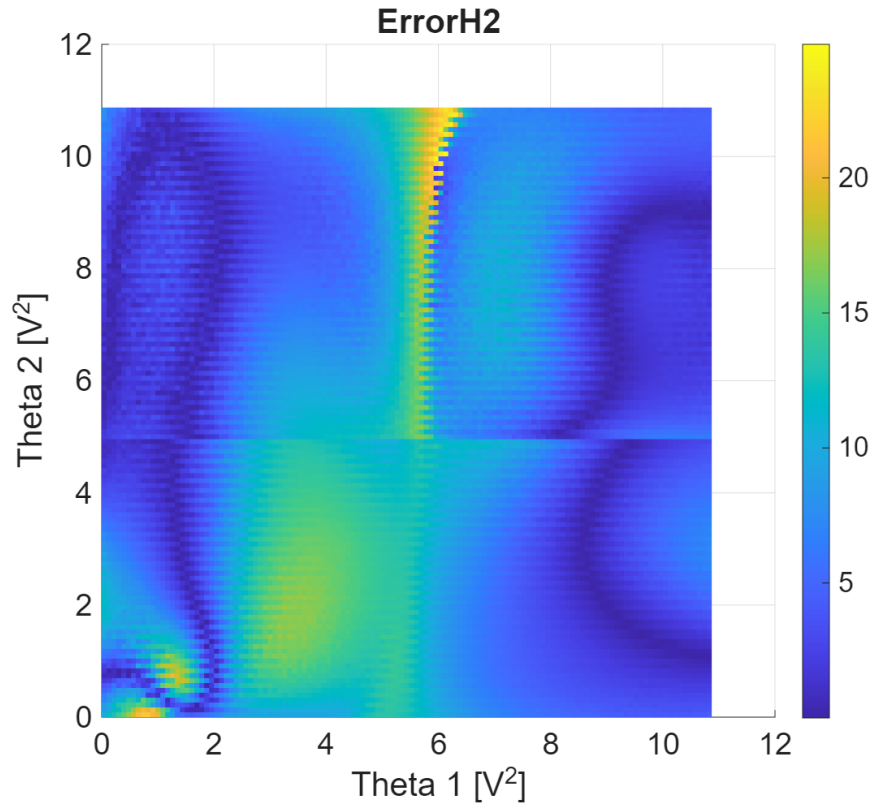


Figure 5.31: Percentage error for the output power split of MZI_3

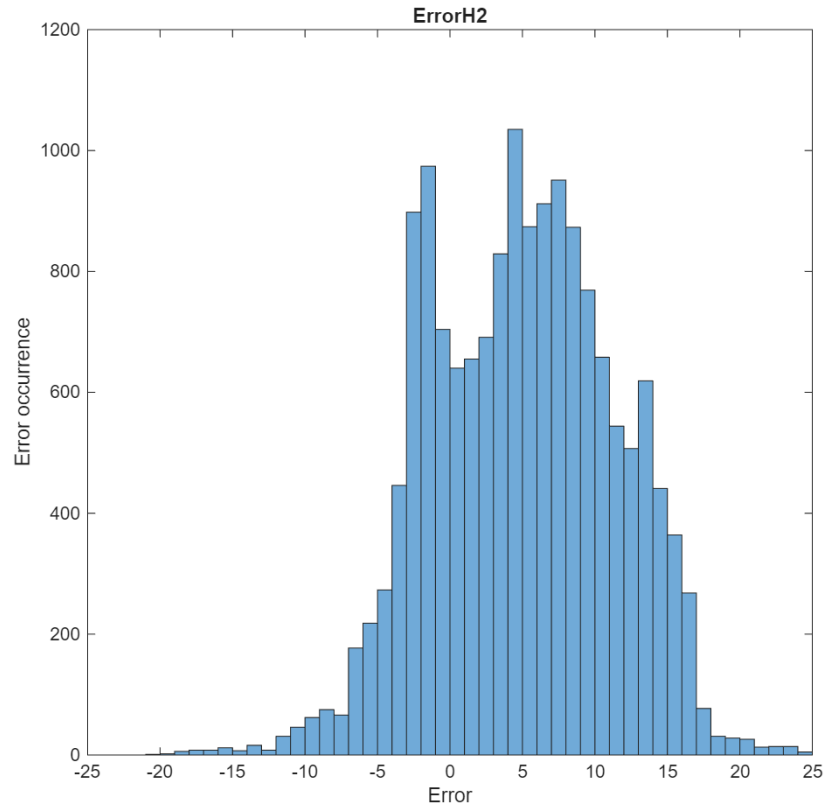


Figure 5.32: Histogram showing percentage error distribution for the output power split of MZI_3

Looking at figure 5.33, MZI_1 and MZI_2 behaves similarly to what we have already observed in table 5.5 and 5.6, with the first one showing the best accuracy. This is probably due to the fact that the inputs of MZI_1 are the best described both in power and phase, while the ones of MZI_2 have a phase only theoretically guessed. The resulting average error over the map is higher than the one previously presented during the first validation process, this was expected given the duration of the sweep and the eventual drifts and thermal cross-talks affecting the mesh.

For what concerns MZI_3 , the error is higher than expected from previous measurements. Its degraded accuracy is likely due to errors from previous stages. The non perfect phase description affecting measurements 5.26 and 5.28 is visible also for MZI_3 , and recognizable by the clear error lines that divide the four π by π quadrants. The "edges" of these four phase quadrants being the most error-prone. At the same time, the error peak visible at the output of MZI_1 (5.24), shows a similar correspondence in the internal power split of MZI_3 (5.30).

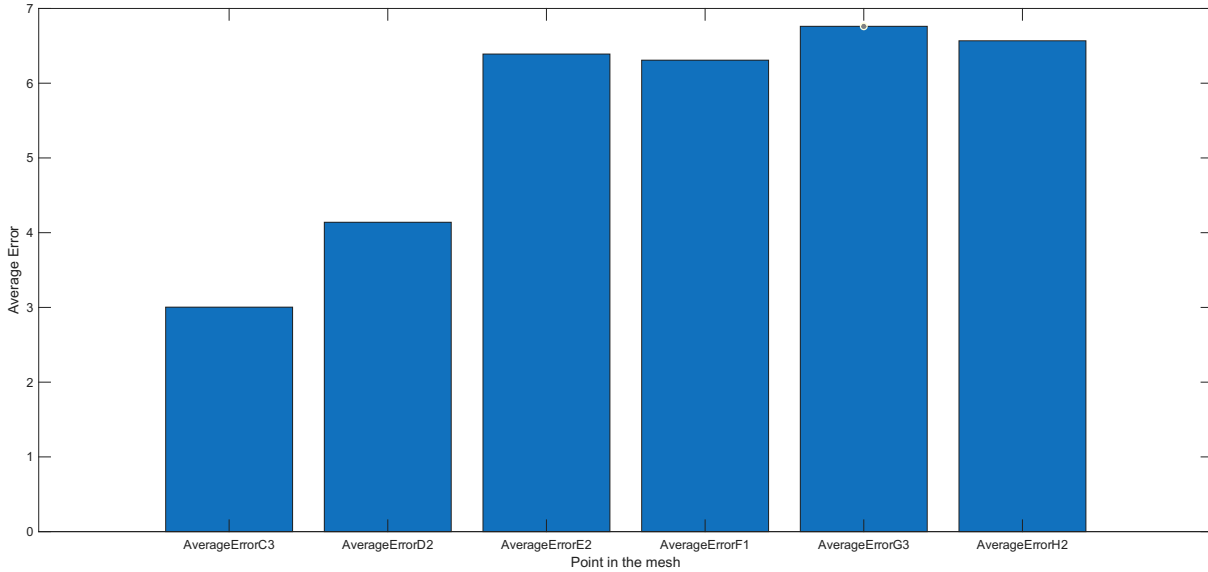


Figure 5.33: Histogram showing the average of the error modules for each power split of the matrix.

The validation algorithm implemented on Matlab can be found in Appendix A.

6 | Conclusions

In this thesis, the optimization, and experimental validation of a high-precision electronic control system for a 3×3 Mach-Zehnder Interferometer (MZI) mesh were systematically investigated.

Driven by the computational bottlenecks of modern digital electronics and the scaling limits of Moore's Law, photonic integrated circuits (PICs) present an apparently perfect solution for high-throughput, energy-efficient Vector-Matrix Multiplications (VMM). However, the inherent susceptibility of optical components to thermal cross-talk, process variations, and environmental drift necessitates robust, active stabilization. This work successfully demonstrated a highly scalable, FPGA-based closed-loop control strategy applied to a 3×3 coherent optical mesh, realized with Reck architecture and operating at a 1550 nm wavelength, proving localized feedback mechanisms to accurately define the transfer of each MZI.

The proper functioning of the proposed control scheme is made possible by transparent photodiodes, placed along the optical path, to perform a non invasive measurement of light, and Titanium Nitride (TiN) micro-heaters for localized phase modulation. By dividing each MZI into two independently controllable directional couplers, the implemented control algorithm avoids the computational complexity associated with global mesh training. Relying on local integral feedback loops implemented within the FPGA, the system controls the power split and the derivative in six different points along the mesh, describing a transfer function uniquely.

Rigorous electro-optical characterizations of the active elements were performed to establish the safe operational boundaries and to map intrinsic non-idealities. The TiN phase-shifters exhibited a clear Positive Temperature Coefficient (PTC) behavior, maintaining structural integrity and linear phase-power translation within specific dynamic ranges. Experimental boundaries were then defined, showing safe operation up to 7.5 V for 100 μm heaters and ≈ 4.5 V for 60 μm heaters. Exceeding these voltages resulted in an irreversible structural degradation of the actuators over time.

Furthermore, an exhaustive analysis of the transparent photodiodes revealed a power-dependent, non-linear responsivity resulting from the Surface State Absorption (SSA)

mechanism. From the measurements was evident how high incident optical powers (13 dBm at the source, and thus ≈ 6 dBm at the sensor) resulted in a non-linear signal overshoots $\approx 17.26\%$ higher than the average total power. Conversely, reducing the optical injection by roughly a factor 4 effectively mitigated these gain-limiting effects, restricting the reading discrepancy to a about 2.6% and thus ensuring higher readout reliability.

The accuracy of the overall system was ultimately validated through the execution of *VMM operations*, performed by setting an arbitrary matrix and then feeding arbitrarily chosen inputs to the optical core. By exploiting the first MZI of the matrix as a coherent phase detector, it was possible to describe the chosen input vectors both in power and phase. Operating then at an optimal optical power regime, the automated control successfully drove the interferometers to their target states with remarkable precision. The matrix was then fixed to the convergence values just reached, and multiple inputs were changed while the feedback was paused. A comparison between the theoretically expected power splits and the empirically measured output vectors, yielded to Root Mean Square (RMS) setting errors predominantly in the 1% to 3% range for fully phase-characterized nodes, specifically MZI_1 . Slightly higher discrepancies were observed in intermediate sections—such as MZI_2 , which yielded to an RMS error of $\approx 5.69\%$ because of its simulated-only input phase description. These findings underscore the critical necessity of comprehensive inline phase-metering for each component of the input vector to achieve absolute determinism.

In conclusion, this research successfully validates the premise that complex-valued optical computing matrices can be stabilized and programmed through localized, electro-optical feedback loops without relying on resource-intensive global optimization methods. Future developments include further validation of the output vector by means of the already implemented phase meter at the end of the structure [3.7,3.8], thanks to which it is also possible to verify the correspondence between real and expected phases. Higher order optical meshes should be also implemented, to further test and verify the scalability of the control algorithm.

Ultimately, the methodologies and hardware architectures examined in this thesis pave the way for scalable, reprogrammable photonic processors capable of satisfying the stringent speed and energy efficiency demands of next-generation artificial intelligence and high-performance computing tasks.

Bibliography

- [1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [2] J. Cheng, H. Zhou, and J. Dong, “Photonic matrix computing: from fundamentals to applications,” *Nanomaterials*, vol. 11, no. 7, p. 1683, 2021.
- [3] S. Bandyopadhyay, A. Sludds, S. Krastanov, R. Hamerly, N. Harris, D. Bunandar, M. Streshinsky, M. Hochberg, and D. Englund, “Single-chip photonic deep neural network with forward-only training,” *Nature Photonics*, vol. 18, no. 12, pp. 1335–1343, 2024.
- [4] S. Pai, Z. Sun, T. W. Hughes, T. Park, B. Bartlett, I. A. D. Williamson, M. Minkov, M. Milanizadeh, N. Abebe, F. Morichetti, A. Melloni, S. Fan, O. Solgaard, and D. A. B. Miller, “Experimentally realized in situ backpropagation for deep learning in photonic neural networks,” *Science*, vol. 380, no. 6643, pp. 398–404, 2023.
- [5] S. Bandyopadhyay, R. Hamerly, and D. Englund, “Hardware error correction for programmable photonics,” *Optica*, vol. 8, no. 10, pp. 1247–1255, 2021.
- [6] A. di Tria, G. Cavicchioli, P. Giannoccaro, F. Morichetti, A. Melloni, G. Ferrari, M. Sampietro, and F. Zanetto, “Automatically setting arbitrary magnitude and phase of mach-zehnder interferometers for scalable optical computing,” 2025.
- [7] M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, “Experimental realization of any discrete unitary operator,” *Physical review letters*, vol. 73, no. 1, p. 58, 1994.
- [8] D. Thomson, A. Zilkie, J. E. Bowers, T. Komljenovic, G. T. Reed, L. Vivien, D. Marris-Morini, E. Cassan, L. Viot, J.-M. Fédéli, *et al.*, “Roadmap on silicon photonics,” *Journal of Optics*, vol. 18, no. 7, p. 073003, 2016.
- [9] H. Moriya, A. Kaneta, and S. Adachi, “Photorefectance study of crystalline silicon,” *Materials Science and Engineering: B*, vol. 76, no. 3, pp. 232–236, 2000.
- [10] H. Weakliem and D. Redfield, “Temperature dependence of the optical properties of silicon,” *Journal of Applied Physics*, vol. 50, no. 3, pp. 1491–1493, 1979.

- [11] B. Korenko, J. Jasenek, and J. Červeňová, “Pockels and kerr effect investigation in fibre bragg gratings,” *J. Electr. Eng.*, vol. 63, pp. 148–151, 2012.
- [12] J. Parra, J. Navarro-Arenas, and P. Sanchis, “Silicon thermo-optic phase shifters: a review of configurations and optimization strategies,” *Advanced Photonics Nexus*, vol. 3, no. 4, pp. 044001–044001, 2024.
- [13] W. Bogaerts, H. Sattari, P. Edinger, A. Y. Takabayashi, I. Zand, X. Wang, A. Ribeiro, M. Jezzini, C. Errando-Herranz, G. Talli, *et al.*, “Morphic: programmable photonic circuits enabled by silicon photonic mems,” in *Silicon Photonics XV*, vol. 11285, p. 1128503, SPIE, 2020.
- [14] S. J. Koester, J. D. Schaub, G. Dehlinger, and J. O. Chu, “Germanium-on-soi infrared detectors for integrated photonic applications,” *IEEE Journal of selected topics in quantum electronics*, vol. 12, no. 6, pp. 1489–1502, 2007.
- [15] J. C. Campbell, S. Demiguel, F. Ma, A. Beck, X. Guo, S. Wang, X. Zheng, X. Li, J. D. Beck, M. A. Kinch, *et al.*, “Recent advances in avalanche photodiodes,” *IEEE Journal of selected topics in quantum electronics*, vol. 10, no. 4, pp. 777–787, 2004.
- [16] F. Grillot, L. Vivien, S. Laval, and E. Cassan, “Propagation loss in single-mode ultrasmall square silicon-on-insulator optical waveguides,” *Journal of Lightwave Technology*, vol. 24, no. 2, p. 891, 2006.
- [17] F. Morichetti, S. Grillanda, M. Carminati, G. Ferrari, M. Sampietro, M. J. Strain, M. Sorel, and A. Melloni, “Non-invasive on-chip light observation by contactless waveguide conductivity monitoring,” *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 20, no. 4, pp. 292–301, 2014.
- [18] A. Perino, F. Zanetto, M. Petrini, F. Toso, F. Morichetti, A. Melloni, G. Ferrari, and M. Sampietro, “High-sensitivity transparent photoconductors in voltage-controlled silicon waveguides,” *Optics Letters*, vol. 47, no. 6, pp. 1327–1330, 2022.
- [19] N. Vrucinic and Y. Zhang, “Reexamination of gain theory for intrinsic photoconductive devices,” *Photonics*, vol. 12, p. 523, 05 2025.
- [20] N. Peserico, B. J. Shastri, and V. J. Sorger, “Integrated photonic tensor processing unit for a matrix multiply: a review,” *Journal of Lightwave Technology*, vol. 41, no. 12, pp. 3704–3716, 2023.
- [21] S. Ohno, R. Tang, K. Toprasertpong, S. Takagi, and M. Takenaka, “Si microring resonator crossbar array for on-chip inference and training of the optical neural network,” *Acs Photonics*, vol. 9, no. 8, pp. 2614–2622, 2022.

- [22] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, “All-optical machine learning using diffractive deep neural networks,” *Science*, vol. 361, no. 6406, pp. 1004–1008, 2018.
- [23] R. Chen and W. Gao, “Free-space optical computing systems,” *Annalen der Physik*, vol. 537, no. 6, p. 2400390, 2025.
- [24] T. Yamaguchi, K. Arai, T. Niiyama, A. Uchida, and S. Sunada, “Time-domain photonic image processor based on speckle projection and reservoir computing,” *Communications Physics*, vol. 6, no. 1, p. 250, 2023.
- [25] D. Brunner, B. Shastri, M. Qadasi, H. Ballani, S. Barbay, S. Biasi, P. Bienstman, S. Bilodeau, W. Bogaerts, F. Böhm, G. Brennan, S. Buckley, X. Cai, M. Strinati, B. Canakci, B. Charbonnier, M. Chemnitz, Y. Chen, S. Cheung, and W. Zhang, “Roadmap on neuromorphic photonics,” 01 2025.
- [26] B. Wu, H. Zhou, J. Dong, and X. Zhang, “Programmable integrated photonic coherent matrix: Principle, configuring, and applications,” *Applied Physics Reviews*, vol. 11, no. 1, 2024.
- [27] W. Bogaerts, D. Pérez, J. Capmany, D. A. B. Miller, J. Poon, D. Englund, F. Morichetti, and A. Melloni, “Programmable photonic circuits,” *Nature*, vol. 586, pp. 207–216, 10 2020.
- [28] S. Wagner, S. Sesia, and D. T. M. Slock, “Unitary beamforming under constant modulus constraint in mimo broadcast channels,” in *2009 IEEE 10th Workshop on Signal Processing Advances in Wireless Communications*, pp. 151–155, 2009.
- [29] S. B. Giddings, “Black holes, quantum information, and unitary evolution,” *Physical Review D*, vol. 85, June 2012.
- [30] W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walmsley, “Optimal design for universal multiport interferometers,” *Optica*, vol. 3, pp. 1460–1465, Dec 2016.
- [31] F. Flamini, N. Spagnolo, N. Viggianiello, A. Crespi, R. Osellame, and F. Sciarrino, “Benchmarking integrated linear-optical architectures for quantum information processing,” *Scientific Reports*, vol. 7, p. 15133, 11 2017.
- [32] R. Barak and Y. Ben-Aryeh, “Quantum fast fourier transform and quantum computation by linear optics,” *J. Opt. Soc. Am. B*, vol. 24, pp. 231–240, Feb 2007.

- [33] D. A. B. Miller, “Self-configuring universal linear optical component,” *Photon. Res.*, vol. 1, pp. 1–15, Jun 2013.
- [34] M. López, H. Hofer, K. D. Stock, J. C. Bermúdez, A. Schirmacher, F. Schneck, and S. Kück, “Spectral reflectance and responsivity of ge- and ingaas-photodiodes in the near-infrared: measurement and model,” *Appl. Opt.*, vol. 46, pp. 7337–7344, Oct 2007.
- [35] F. Zanetto, V. Grimaldi, F. Toso, E. Guglielmi, M. Milanizadeh, D. Aguiar, M. Moralis-Pegios, S. Pitris, T. Alexoudi, F. Morichetti, *et al.*, “Dithering-based real-time control of cascaded silicon photonic devices by means of non-invasive detectors,” *IET Optoelectronics*, vol. 15, no. 2, pp. 111–120, 2021.
- [36] R. G. Lyons, *Understanding Digital Signal Processing*. Pearson, 3rd ed., 2013.

A | Appendix A

```

1 clear
2 close all
3 clc
4
5 syms theta_1 theta_2 theta_3 phi_2 phi_1 phi_3;
6 syms phi_coupler theta_coupler;
7
8 %power at the output of each MZI during matrix sweep to check the power
   at
9 %each stage and to veify its not too low
10 load MZI_sweep_power_check.mat
11 %real measured power splits along the mesh
12 load sweep_data.mat
13 %datas from the phase measurements
14 load data.mat
15 %load data1.mat
16 load data2.mat
17 load data3.mat
18
19 error_counter = 0;
20 %averaging of the photodiodes datas to reduce input phase measurement
   error
21
22 %data+data2, averaging of phase measurements
23
24 PD3_average = (PD3+PD3_1)/2;
25 PD6_average = (PD6+PD6_1)/2;
26 PD9_average = (PD9+PD9_1)/2;
27 PD4_average = (PD4+PD4_1)/2;
28 PD1_average = (PD1+PD1_1)/2;
29 PD2_average = (PD2+PD2_1)/2;
30 PD10_average = (PD10+PD10_1)/2;
31
32 %evaluation of PD datas
33 phase = atan2(PD6_average+PD3_average-2*PD9_average, PD6_average -
   PD3_average)/pi;

```

```

34 phase(phase < 0) = phase(phase < 0) +2;
35
36 phase_diff = phase - 0.498968;
37 phase_diff(phase_diff < 0) = phase_diff(phase_diff < 0) +2;
38
39 figure
40 surf(ht8,ht10,phase,'EdgeColor','none')
41 xlabel('Theta 1')
42 ylabel('Theta 2')
43 axis square
44
45 figure
46 surf(ht8(1:126),ht10(1:126),phase_diff(18:143,21:146),'EdgeColor','none'
47 )
48 xlabel('Theta 1 [V^2]')
49 ylabel('Theta 2 [V^2]')
50 axis square
51
52 %input powers
53 In1_power = PD10_average./(PD1_average+PD2_average);
54 In2_power = PD4_average./(PD1_average+PD2_average);
55 In3_power = PD2_average./(PD1_average+PD2_average);
56
57 %input vector
58
59 phase_second_channel = phase_diff;
60 phase_second_channel(75:150,:) = phase_second_channel(75:150,:)+0.98;
61 phase_second_channel(1:7,:) = phase_second_channel(1:7,:)+0.98;
62
63 input_vector = [sqrt(In1_power).*exp(i*pi*(phase_second_channel)); sqrt(
64     In2_power).*exp(i*pi*(phase_diff));sqrt(In3_power).*exp(i*pi*0)];
65 %phases
66
67 phase_vector = [1.1, 0.3, 0.25, 1.3, 0.8, 1.4];
68 %MZ tranfer function
69
70 MZ1 = [1 0 0;0 (exp(i*theta_1)-1)/2 i*exp(i*phi_1)*(1+exp(i*theta_1))/2;
71     0 i*(exp(i*theta_1)+1)/2 exp(i*phi_1)*(1-exp(i*theta_1))/2];
72 MZ2 = [(exp(i*theta_2)-1)/2 i*exp(i*phi_2)*(1+exp(i*theta_2))/2 0; i*(
73     exp(i*theta_2)+1)/2 exp(i*phi_2)*(1-exp(i*theta_2))/2 0; 0 0 1];
74 MZ3 = [1 0 0;0 (exp(i*theta_3)-1)/2 i*exp(i*phi_3)*(1+exp(i*theta_3))/2;
75     0 i*(exp(i*theta_3)+1)/2 exp(i*phi_3)*(1-exp(i*theta_3))/2];
76 %First coupler trasnfer function positive

```

```

73 Coupler_1 = 1/sqrt(2)*[1 i; i 1]*[1 0; 0 exp(i*phi_coupler)];
74 %Second coupler transfer function positive
75
76 Coupler_2 = 1/sqrt(2)*[1 i; i 1]*[exp(i*theta_coupler) 0; 0 1];
77 %Negative MZ transfer function
78
79 % MZ2 = -i/2*[-2/i 0 0;0 i*(exp(-i*theta_1)-1) exp(-i*phi_1)*(1+exp(-i*
      theta_1)); 0 exp(-i*theta_1)+1 i*exp(-i*phi_1)*(1-exp(-i*theta_1))];
80 % MZ1 = -i/2*[i*(exp(-i*theta_2)-1) exp(-i*phi_2)*(1+exp(-i*theta_2)) 0;
      exp(-i*theta_2)+1 i*exp(-i*phi_2)*(1-exp(-i*theta_2)) 0; 0 0 -2/i];
81 % MZ3 = -i/2*[i*(exp(-i*theta_3)-1) exp(-i*phi_3)*(1+exp(-i*theta_3)) 0;
      exp(-i*theta_3)+1 i*exp(-i*phi_3)*(1-exp(-i*theta_3)) 0; 0 0 -2/i];
82 %First coupler trasnfer function negative
83 % Coupler_1 = 1/sqrt(2)*[1 -i; -i 1]*[1 0; 0 exp(-i*phi_coupler)];
84 %Second coupler transfer function negative
85 % Coupler_2 = 1/sqrt(2)*[1 -i; -i 1]*[exp(-i*theta_coupler) 0; 0 1];
86
87 %Building of the overall 3x3 structure
88
89 MZ_3_3 = MZ3*MZ2*MZ1;
90
91 %amplitudes along the mesh
92 %internal power split in MZ1
93
94 Coupler_1_1 = double(subs(Coupler_1, [phi_coupler],[phase_vector(1,1)*pi
      ]));
95 Coupler_1_1_out1 = Coupler_1_1(1,1).*input_vector(151:300,:) +
      Coupler_1_1(1,2).*input_vector(301:450,:);
96 Coupler_1_1_out2 = Coupler_1_1(2,1).*input_vector(151:300,:) +
      Coupler_1_1(2,2).*input_vector(301:450,:);
97 PowSplitC3 = (abs(Coupler_1_1_out2).^2)./(abs(Coupler_1_1_out1).^2+abs(
      Coupler_1_1_out2).^2);
98 %output split MZ1
99
100 Coupler_2_1 = double(subs(Coupler_2, [theta_coupler],[phase_vector(1,2)*
      pi]));
101 Coupler_2_1_out1 = Coupler_2_1(1,1).*Coupler_1_1_out1+Coupler_2_1(1,2).*
      Coupler_1_1_out2;
102 Coupler_2_1_out2 = Coupler_2_1(2,1).*Coupler_1_1_out1+Coupler_2_1(2,2).*
      Coupler_1_1_out2;
103 PowSplitD2 = (abs(Coupler_2_1_out1).^2)./(abs(Coupler_2_1_out1).^2+abs(
      Coupler_2_1_out2).^2);
104 %internal split MZ2
105

```

```

106 Coupler_1_2 = double(subs(Coupler_1, [phi_coupler],[phase_vector(1,3)*pi
    ]));
107 Coupler_1_2_out1 = Coupler_1_2(1,1).*input_vector(1:150,:) + Coupler_1_2
    (1,2).*Coupler_2_1_out1;
108 Coupler_1_2_out2 = Coupler_1_2(2,1).*input_vector(1:150,:) + Coupler_1_2
    (2,2).*Coupler_2_1_out1;
109 PowSplitE2 = (abs(Coupler_1_2_out2).^2)./(abs(Coupler_1_2_out2).^2+abs(
    Coupler_1_2_out1).^2);
110 %output split MZ2
111
112 Coupler_2_2 = double(subs(Coupler_2, [theta_coupler],[phase_vector(1,4)*
    pi]));
113 Coupler_2_2_out1 = Coupler_2_2(1,1).*Coupler_1_2_out1 + Coupler_2_2(1,2)
    .*Coupler_1_2_out2;
114 Coupler_2_2_out2 = Coupler_2_2(2,1).*Coupler_1_2_out1 + Coupler_2_2(2,2)
    .*Coupler_1_2_out2;
115 PowSplitF1 = (abs(Coupler_2_2_out1).^2)./(abs(Coupler_2_2_out1).^2+abs(
    Coupler_2_2_out2).^2);
116 %input split MZ3
117
118 Coupler_1_3 = double(subs(Coupler_1, [phi_coupler],[phase_vector(1,5)*pi
    ]));
119 Coupler_1_3_out1 = Coupler_1_3(1,1).*Coupler_2_2_out2 + Coupler_1_3(1,2)
    .*Coupler_2_1_out2;
120 Coupler_1_3_out2 = Coupler_1_3(2,1).*Coupler_2_2_out2 + Coupler_1_3(2,2)
    .*Coupler_2_1_out2;
121 PowSplitG3 = (abs(Coupler_1_3_out2).^2)./(abs(Coupler_1_3_out2).^2+abs(
    Coupler_1_3_out1).^2);
122 %output split MZ3
123
124 Coupler_2_3 = double(subs(Coupler_2, [theta_coupler],[phase_vector(1,6)*
    pi]));
125 Coupler_2_3_out1 = Coupler_2_3(1,1).*Coupler_1_3_out1 + Coupler_2_3(1,2)
    .*Coupler_1_3_out2;
126 Coupler_2_3_out2 = Coupler_2_3(2,1).*Coupler_1_3_out1 + Coupler_2_3(2,2)
    .*Coupler_1_3_out2;
127 PowSplitH2 = (abs(Coupler_2_3_out1).^2)./(abs(Coupler_2_3_out2).^2+abs(
    Coupler_2_3_out1).^2);
128
129
130 %%absolute errors
131
132 ErrorC3 = abs(PowSplitC3(18:143,21:146) - MZ1_PowSplit_Int
    (18:143,21:146)).*100;

```

```

133
134 ErrorD2 = abs(PowSplitD2(18:143,21:146) - MZ1_PowSplit_Ext
      (18:143,21:146)).*100;
135
136 ErrorE2 = abs(PowSplitE2(18:143,21:146) - MZ2_PowSplit_Int
      (18:143,21:146)).*100;
137
138 ErrorF1 = abs(PowSplitF1(18:143,21:146) - MZ2_PowSplit_Ext
      (18:143,21:146)).*100;
139
140 ErrorG3 = abs(PowSplitG3(18:143,21:146) - MZ3_PowSplit_Int
      (18:143,21:146)).*100;
141
142 ErrorH2 = abs(PowSplitH2(18:143,21:146) - MZ3_PowSplit_Ext
      (18:143,21:146)).*100;
143
144 %mean
145
146 AverageErrorC3 = mean(ErrorC3(ErrorC3<=52))
147
148 AverageErrorD2 = mean(ErrorD2(ErrorD2<=52))
149
150 AverageErrorE2 = mean(ErrorE2(ErrorE2<=52))
151
152 AverageErrorF1 = mean(ErrorF1(ErrorF1<=52))
153
154 AverageErrorG3 = mean(ErrorG3(ErrorG3<=52))
155
156 AverageErrorH2 = mean(ErrorH2(ErrorH2<=52))
157
158 %non-consideration of the error when input power is too low
159 PD3,6..._sweep are the photodiodes of the matrix set
160 for i = 1:126
161     for j = 1:126
162         if (PD9_sweep(i,j)) <= (PD1_sweep(i,j)+PD2_sweep(i,j))/50
163             if(i > 1 && j > 1 )
164                 ErrorC3(i,j) = ErrorC3(i-1,j-1);
165                 error_counter = error_counter + 1;
166             else
167                 ErrorC3(i,j) = 0;
168                 error_counter = error_counter + 1;
169             end
170         end
171         if (PD6_sweep(i,j)) <= (PD1_sweep(i,j)+PD2_sweep(i,j))/100

```

```

172         if(i > 1 && j > 1 )
173             ErrorD2(i,j) = ErrorD2(i-1,j-1);
174             error_counter = error_counter + 1;
175         else
176             ErrorD2(i,j) = 0;
177             error_counter = error_counter + 1;
178         end
179     end
180     if (PD11_sweep(i,j)) <= (PD1_sweep(i,j)+PD2_sweep(i,j))/100
181         if(i > 1 && j > 1 )
182             ErrorE2(i,j) = ErrorE2(i-1,j-1);
183             error_counter = error_counter + 1;
184         else
185             ErrorE2(i,j) = 0;
186             error_counter = error_counter + 1;
187         end
188     end
189     if (PD12_sweep(i,j)) <= (PD1_sweep(i,j)+PD2_sweep(i,j))/100
190         if(i > 1 && j > 1 )
191             ErrorF1(i,j) = ErrorF1(i-1,j-1);
192             error_counter = error_counter + 1;
193         else
194             ErrorF1(i,j) = 0;
195             error_counter = error_counter + 1;
196         end
197     end
198     if (PD13_sweep(i,j)) <= (PD1_sweep(i,j)+PD2_sweep(i,j))/100
199         if(i > 1 && j > 1 )
200             ErrorG3(i,j) = ErrorG3(i-1,j-1);
201             error_counter = error_counter + 1;
202         else
203             ErrorG3(i,j) = 0;
204             error_counter = error_counter + 1;
205         end
206     end
207     if (PD8_sweep(i,j)) <= (PD1_sweep(i,j)+PD2_sweep(i,j))/100
208         if(i > 1 && j > 1 )
209             ErrorH2(i,j) = ErrorH2(i-1,j-1);
210             error_counter = error_counter + 1;
211         else
212             ErrorH2(i,j) = 0;
213             error_counter = error_counter + 1;
214         end
215     end

```



```
216         end
217     end
218
219     %rms
220
221     RMSErrorC3 = rms(ErrorC3(ErrorC3<=52))
222
223     RMSErrorD2 = rms(ErrorD2(ErrorD2<=52))
224
225     RMSErrorE2 = rms(ErrorE2(ErrorE2<=52))
226
227     RMSErrorF1 = rms(ErrorF1(ErrorF1<=52))
228
229     RMSErrorG3 = rms(ErrorG3(ErrorG3<=52))
230
231     RMSErrorH2 = rms(ErrorH2(ErrorH2<=52))
```


List of Figures

1.1	Silicon transmittance, $1550nm$ falls perfectly inside transparency window .	4
1.2	Silicon thermo-optical coefficient as a function of temperature	4
1.3	a)Example of a buried waveguide b)Representation of light intensity inside of the waveguide for a different size of the core. For the largest core the light is better confined and the losses due to tunneling in the cladding are limited.	5
1.4	Example of a buried silicon waveguide	7
1.5	Dependence of the refractive index as a function of the applied electric field at two wavelengths	8
1.6	Example of a buried silicon waveguide with a titanium nitride heater on top	9
1.7	Control chain schematic of a PIC	9
1.8	Absorption coefficient for different materials as a function of the wavelength. Around the wavelength of interest ($1550nm$) <i>Si</i> shows the worst absorption together with <i>GaAs</i> , while <i>Ge</i> and <i>In₅₃Ga₄₇As</i> shows a better behavior.	10
1.9	<i>SSA</i> two-step absorption process	11
1.10	2x2 directional coupler realized with optical fibers	13
1.11	On the left a Y coupler, in the middle an example of a cascade of T couplers and on the right a representation of a directional star coupler	14
1.12	Directional coupler realized with two waveguides	14
1.13	MZI with two paths of different length L_1 and L_2	15
1.14	MZI with the inner phase θ and the input phase φ	16
2.1	Example of a micro-ring resonator with the above through guide and the bottom drop guide	18
2.2	(a)Multiplication by means of two optical planes (b)Bi-dimensional optical integration (c)Successive Fourier transformations (d)Optical convolution in two dimensions	19

2.3	(a)2x2 Optical gate mixing the two optical waves a_1 and a_2 (b)Possible configurations of the light through the coupler (c,d,e)Two degrees of freedom p_1 and p_2 to control output phase and amplitude, tuning the coupling ratio or the phase shift.[27]	20
2.4	Optical implementation of the SVD, with A and U^* meshes implementing unitary matrices, and the diagonal matrix Σ implemented by the structure in the middle.	21
2.5	On the left a MZI realized with two 50:50 beam splitters, two mirrors and two phase shifters to properly tune the transfer function In the middle a block representation of a MZI. On the right a triangular array of beam splitters implementing an $N \times N$ unitary matrix as an optical multiport.[7]	22
2.6	A 4×4 triangular mesh divided in such a way to result the product of extended 2×2 matrices	23
2.7	(a)Reck scheme with the blue dots representing tunable beam and phase shifters (b)Clements implementation of the same structure (c)Asymmetric beam splitter with one internal and one external shifter (d)MZI with two internal phase shifters for improved symmetry and compactness	24
2.8	Incremental steps (1-6) to achieve the overall diagonal zeroing illustrated by Clements et al. showing how the mesh is populated following the cancellation of the matrix values.[30]	25
2.9	(a) 8×8 matrix implementation with Reck architecture (b) 8×8 matrix implementation with Clements architecture (c) Fast architecture implementation with 8 inputs, designed designed to solve specific problems like Fast Fourier Transform (FFT), Quantum Fourier transform (QFT) and Sylvester's transformations. (d) Unit cells common in the three layouts (e) Comparison table among the three architectures [31]	26
3.1	Mach Zehnder interferometer implementation, with the two phase-shifter actuators modeled by a φ phase, compensating the input branch of the interferometer, and a θ shifter in between the two couplers.	28
3.2	3D map of MZI output power split. The red lines shows all the points with the desired power split, while the blue triangles mark the points where we have both the desired phase shift and power split.	30

3.3	On top, overview of the power map along one period for both heaters. On the bottom the same representation but for the phase difference between the two output branches. The red line represents the points meeting the power of interest, while the blue line indicate the points with the searched phase. It can be seen that even using both information we still get two possible points to which our feedback can converge in a single period. . . .	31
3.4	On the left a phase simulation of a MZI output phase difference, on the output power split. Both simulations were performed along 2 periods. . . .	33
3.5	Picture of the actual implementation of the 3×3 structure. Can be seen the actuators (heaters), i.e. the purple structures along the optical guides, and the transparent photodiodes along with the dummy photodiodes recognizable where the guide widens and takes on a bluish color.	34
3.6	Input vector generator. By means of it we can generate any 3×1 vector in power starting from a single input	35
3.7	Diagonal MZIs structure with output photodiodes made out of germanium, designed for coherent light phase measurements.	35
3.8	Schematic representation of the 3×3 mesh (green) along with the input generator (brown) and the phase meter (blue).	37
3.9	Input generator base on a diagonal structure with two MZIs. The transparent photodiodes are the blue squares along the optical path, while the heaters are the red ones.	38
3.10	Simulation of the three input powers normalized with respect to the overall power available. The red lines in the 3 images show the phase combinations allowed to get a power split of $\frac{1}{3}$ for each one of the inputs.	39
3.11	Two different power combinations for the input are shown in the plots. It can be clearly seen that 4 different solutions can be found, one for each $\frac{\pi}{2}$ by $\frac{\pi}{2}$ subplane.	40
3.12	Map of the phase difference between In_2 and both In_1 (a) and In_3 (b) as a function of θ_1 and θ_2 . It can be appreciated from graph (a) how the phase between the first two inputs is 0 or π as a function of θ_2 , thus making it easier to be modeled.	41
3.13	Lateral view of $\Delta\phi_{2,3}$, with only half a period simulation along θ_1 in order to make the $\frac{\theta_2}{2}$ slope of the phase map more clear.	42
4.1	Block scheme description of the overall system	43

4.2	Top view of mantis PCB. It can be seen that the "two halves" of the board have the components necessary for the 2x2 and 3x3 mesh to work, along with two separate connectors adopted by the two meshes to communicate with the motherboard.	44
4.3	Top view of Helios 3 motherboard with a brief description of the main components.	46
4.4	Schematics of THS7002 PGAs by Texas Instruments. Preamplifier gain is fixed to 10, while by choosing $G_{0,1,2}$ bits we can select the appropriate programmable gain.	47
4.5	Front and back picture of acquisition module.	47
4.6	a)Physical implementation of transparent photodiodes, placed along the optical guide, along with the respective dummies b)Schematic representation of the two photodiodes, with TIA's virtual ground in between the two photodiodes.	48
4.7	Schematics of photodiodes readout chain on the Mantis. The dummy photodiode has its anode connected to the virtual ground to subtract the dark current. As can be seen, the same resistance is present at the "+" and "-" terminal of the amplifier to compensate for bias currents coming out of the amplifier.	49
4.8	Actuation module schematic, showing signal path from the FPGA to the actuator. The digital signal from the FPGA is then amplified and feed into a shunt resistor, to read the real-time flowing current. A transistor used as a switch decouples then the actuator from the driving electronics at start-up.	50
4.9	Actuation module schematic, showing signal path from the FPGA to the actuator.	51
4.10	LPF chain for the DC value extraction	52
4.11	Normalized <i>Low Pass Filter</i> scheme	53
4.12	LPF simulation of the actual VHDL implementation with 3 different poles, taken with a decade of distance between them at: $1kHz$ $10kHz$ and $100kHz$, showing the same in-band amplitude. It is also visible how the filter works properly up to the Nyquist frequency, which in our case is $f_n \cong 234kHz$	53
4.13	Generic digital filter scheme as a function of parameters a, b and α	55
4.14	Lock in filtering chain showing both <i>in-phase</i> and <i>quadrature</i> paths.	56

4.15	Schematic description of the integral acquisition chain. The operation mode is set with two different multiplexers. The first one is used to feed the integral control with desired current value, and the other to choose the correct target, calculated as a function of the operation mode.	56
4.16	Positioning of the photodiodes inside of a MZI. Three photodiodes are present for each MZI, here represented by k, i and j	57
4.17	On the left we model the error at the input of the integrator as $S_{real}[n] - S_{target}[n] = S_{error}[n]$, this results in an increase of heaters voltage for measured values above the target ones, and in a decrease of heater voltage for the lower ones. On the right a loop with the opposite sign is shown, with $S_{target}[n] - S_{real}[n] = S_{error}[n]$, which results in a different direction of convergence and thus in different convergence points.	58
5.1	Top view of Mantis PCB mounted on the optical bench. The input optical fiber can be seen in the bottom, together with the optical chip, mounted on a 3D printed support to guarantee a 12° entrance angle for the light. On the top portion of the board instead, the ribbon cable for data transfer and the SMA connector to bias the substrate are connected.	59
5.2	Interface board to create a communication between Mantis PCB and external instrumentation. The connection with the Mantis takes place by means of ribbon cables, while the interaction with external instrumentation is performed with BNC and SMA connectors.	60
5.3	SMU interconnection diagram with the heater	61
5.4	Heaters with different size were used for the input mesh and the computational stage, respectively $60\mu m$ and $100\mu m$	62
5.5	On the left, the current-voltage plot for the $100\mu m$ heaters between 0 and 10V. On the right the corresponding resistance. An increasing resistance is measured up to 8.35V, showing the PTC behavior of TiN.	62
5.6		62
5.7	On the left, the current-voltage plot for the $100\mu m$ heaters between 0 and 15V. On the right the corresponding resistance. The much lower resistance along all the sweep clearly shows the wear of the component.	63
5.8	Plot over time, at fixed voltages, of the resistance and the current in the component. The voltages chosen are at the boundaries of the safe operation for the heater.	64

5.9	Results of the three sweeps at increasing voltages, showing no wear in the low range of heater dynamics as can be seen from the perfect overlap between blue and yellow line up to $2V$	64
5.10	Measurement over time for $60\mu m$ heaters with incremental voltages. No significant change no significant change can be appreciated up to $4.5V$. . .	65
5.11	a) $0V - 5V$ sweep of a $60\mu m$ heater b) $0V - 5.6V$ sweep of a $100\mu m$ heater with incident light.	65
5.12	We can see how from a) to c) the colour of the middle heater changes, showing a physical change in the material.	66
5.13	Internal power split measurement with respect to the overall power at $13dBm$. It can be seen how the maximum signal read by the single internal photodiode at its maximums seems $\approx 17.26\%$ larger than the overall output signal.	67
5.14	Internal power split measurement with respect to the overall power at $10dBm$ and $7.5dBm$. The overshoot visible in the internal photodiode is $\approx 8.86\%$ in this case for $10dBm$ and $\approx 2.6\%$ for $7.5dBm$	68
5.15	Power split measurement over 26 minutes for three different input powers: $13dBm$, $10dBm$ and $7.5dBm$. The represented timescale is in seconds. . . .	69
5.16	MZI schematics, showing the positioning of heaters (red) and photodiodes (blue) along the device.	71
5.17	Polar plot showing a comparison between the simulated result, represented by the orange and the purple curves, and the experimental results.	72
5.18	Phase measurement section along ψ_2 compared to the simulated measurement.	73
5.19	Three dimensional representation of MZI_1 input phase. The two axis show the squared voltage applied to the heaters of the input generator. It is present an offset in the origin along both directions with respect to the simulation results due to non-ideality of the physical implementation. . . .	74
5.20	χ_{23} phase map after subtracting the base phase difference of $\chi_{23}(0) = 0.498968\pi$, corresponding to the one associated with input vector $IN = \begin{bmatrix} 0.3333 & 0.3333 & 0.3333 \end{bmatrix}^T$ used to set the matrix. The phase measurement was cropped to show a 2π by 2π map.	79
5.21	Percentage error for the internal power split of MZI_1	80
5.22	Histogram showing percentage error distribution for the internal power split of MZI_1	80
5.23	Percentage error for the output power split of MZI_1	81

5.24	Histogram showing percentage error distribution for the output power split of MZI_1	81
5.25	Percentage error for the internal power split of MZI_2	82
5.26	Histogram showing percentage error distribution for the internal power split of MZI_2	82
5.27	Percentage error for the output power split of MZI_2	83
5.28	Histogram showing percentage error distribution for the output power split of MZI_2	83
5.29	Percentage error for the internal power split of MZI_3	84
5.30	Histogram showing percentage error distribution for the internal power split of MZI_3	84
5.31	Percentage error for the output power split of MZI_3	85
5.32	Histogram showing percentage error distribution for the output power split of MZI_3	85
5.33	Histogram showing the average of the error modules for each power split of the matrix.	86

List of Tables

5.1	Keithley 2450 SMU Source and Measure Specifications	61
5.2	Voltage-Phase description of the heaters, with all values well within the proper working range and thus ensuring predictable behavior.	66
5.3	Power setting variation between different powers	70
5.4	Phase setting for each point of the matrix. φ heaters are the ones before the first coupler, while θ heaters are the ones between the two couplers. . .	75
5.5	Power split results along the mesh measured with a maximum signal of $1dBm$ readen by the photodiodes.	76
5.6	Power split results along the mesh and RMS error between theoretical and experimental power splits at $\approx 3.76dBm$	77
5.7	Comparison and variation of RMS errors between the two configurations .	78

