



**UNIVERSIDAD DE CONCEPCIÓN
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA**



**CONTROL DE FASE DE
INTERFERÓMETRO EN FIBRA ÓPTICA
MULTI-NÚCLEO UTILIZANDO FPGA**

POR

Mónica Eloísa Venegas Rosales

Memoria de Título presentada a la Facultad de Ingeniería de la Universidad de Concepción para optar al título profesional de Ingeniera Civil Electrónica

Profesores Guía
Miguel Figueroa Toro
Gabriel Saaverda Mondaca

Enero 2023
Concepción (Chile)

© 2023 Mónica Eloísa Venegas Rosales

© 2023 Mónica Eloísa Venegas Rosales

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento.

“Not a single one of us here today has done it alone. We are each a patchwork quilt of those who have loved us...”

Taylor Swift

Agradecimientos

Agradezco a mis padres Graciela y Edgardo por su amor y apoyo incondicional, gracias también por siempre incentivar mi aprendizaje y libertad, a mi hermano Raúl por su compañía y amistad forjada aún más en esta etapa. A Gloria, gracias por cuidarme, y enseñarme para enseñarte. Las amigas de la vida que también son familia, Victoria, Vale y Liza, gracias por todo su cariño.

A los amigos que me regaló la universidad, Feña, Rodrigo, Angelo, Neny, Miguel, Berna y Katty, por tantos almuerzos, tardes de estudio y no estudio, los paseos a la playa y las onces. Con ustedes fue todo más bonito.

Gracias a mis profesores guía, Gabriel y Miguel, por su infinita paciencia, disposición y apoyo en la realización de este trabajo.

A los integrantes de VLSI 2022, Javier, Yaimé, Sofi, Wladi, Caro, Bárbaro, Jocsan y Felipe. Nunca pensé encontrar un ambiente tan agradable en un laboratorio y menos a una persona que alegraría tanto mi vida.

Finalmente, gracias a todos los seres del gran azul que me acompañaron a llegar hasta acá, Seba, los recuerdos contigo estarán siempre presentes .

Resumen

Las aplicaciones del interferómetro llegan a campos como la mecánica, biología, astronomía y comunicaciones, estos dispositivos son altamente sensibles a las alteraciones físicas del medio, las cuales generan el poco deseado ruido de fase. En el presente trabajo se aborda el desarrollo y validación de un algoritmo de control para un interferómetro Mach-Zehnder de cuatro brazos, el que se basa en fibra óptica multi-núcleo y está emplazado en la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Concepción. El algoritmo de control busca estabilizar las fluctuaciones en la intensidad de salida requeridas por el sistema de una manera rápida, sencilla y eficiente.

Este trabajo se comenzó con el estudio de la planta a controlar y la toma de muestras de los parámetros relevantes para comprender su funcionamiento, para luego pasar a la revisión de algoritmos de optimización multiparámetro y su debido uso en el abordaje de la problemática a solucionar.

Una vez adaptado el algoritmo de Powell para esta implementación, se probó su funcionamiento al momento de controlar el sistema mediante simulación en software y en hardware, para luego contrastar dichos resultados. Esto mostró una similitud relevante entre ambos procedimientos, logrando validar el funcionamiento del algoritmo presentado en este trabajo.

Finalmente, se presenta el uso de recursos de la implementación en hardware de este controlador, simulado sobre una tarjeta de desarrollo Zybo Z7-20, la cual contiene un FPGA Zynq™-7000 de Xilinx.

Abstract

Interferometer applications include mechanics, biology, astronomy, and communications. These devices are susceptible to physical environmental changes, which generate unwanted phase noise. This paper deals with developing and validating a control algorithm for a four-arm Mach-Zehnder interferometer based on multi-core optical fiber located at the Faculty of Physical and Mathematical Sciences of the University of Concepcion. The control algorithm seeks to stabilize the fluctuations in the output intensity required by the system in a fast, simple, and efficient way.

This work began with studying the plant to be controlled and taking samples of relevant parameters to understand its operation. Then move on to reviewing multi-parameter optimization algorithms and their due use in addressing the problem to be solved.

Once Powell's algorithm was adapted for this implementation, its operation was tested when controlling the system using software and hardware simulation to contrast these results later. The simulations showed a relevant similarity between both procedures, managing to validate the operation of the algorithm presented in this work.

Finally, the use of resources for the hardware implementation of this controller is presented and simulated on a Zybo Z7-20 development board, which contains a Xilinx Zynq™-7000 FPGA.

Índice general

1. Introducción	1
1.1. Introducción general	1
1.2. Objetivos	2
1.2.1. Objetivo general	2
1.2.2. Objetivos específicos	2
1.3. Alcances y limitaciones	3
1.4. Temario	3
2. Marco teórico	4
2.1. Interferómetro Mach-Zehnder	4
2.1.1. Ruido de fase	5
2.1.2. Óptica matricial	6
2.2. Algoritmos de optimización multiparámetro sin derivadas	7
2.2.1. Algoritmo de Powell	7
3. Sistema interferométrico implementado en Universidad de Concepción	11
3.1. Componentes del sistema	12
3.1.1. Beam splitter	12
3.1.2. Controlador manual de polarización	12
3.1.3. Modulador de fase	13
3.1.4. Fotodetector	14
3.2. Interferómetro de 4 caminos	14

3.2.1.	Beam splitter de 4×4	15
3.2.2.	Matriz de fase	16
3.2.3.	Sistema completo	16
3.3.	Simulación del sistema	18
4.	Control ideal	21
4.1.	Programa a probar	21
4.1.1.	Maximización unidimensional	22
4.1.2.	Implementación en software	22
4.2.	Adaptación del algoritmo de Powell	23
4.3.	Simulación lazo de control en software	24
5.	Arquitectura del controlador	27
5.1.	Arquitectura general	27
5.2.	Arquitectura algoritmo de control	28
5.2.1.	Powell	28
5.2.2.	Maximiza	29
5.2.3.	Mover	30
5.3.	Hardware	30
5.3.1.	Zybo Z7-20	31
5.3.2.	Pmod AD1	32
5.3.3.	Pmod DA2	32
6.	Resultados	34
6.1.	Equivalencia de magnitudes	34
6.2.	Simulación hardware	35
6.2.1.	Testbench	35
6.2.2.	Entrega de datos	36
6.2.3.	Uso de recursos	36
7.	Conclusiones y trabajo futuro	38

Índice de figuras

2.1. Interferómetro Mach-Zehnder de dos brazos	5
2.2. Aplicación método de Powell en 2 dimensiones	10
3.1. Diagrama MZI. BS: beam splitter, ϕ^N : efecto del ruido de fase, ϕ^C : acción de control, I_{out} : intensidad de salida. Lo resaltado en naranja corresponde al lazo de control a implementar, donde I es la entrada al controlador y S_{c1} , S_{c2} y S_{c3} las tres señales de salida.	11
3.2. BS de 4 brazos	12
3.3. Controlador manual de polarización.	12
3.4. Voltaje de entrada v/s potencia de salida modulador[2].	13
3.5. Fotodetector para fibra óptica.	14
3.6. Interferómetro real. (1) El láser ingresa por la parte derecha para luego pasar por el BS1 y los polarizadores manuales. (2) Conexión de señales de entrada a los moduladores. (3) Luego de pasar por los segundos polarizadores y el BS2, la salida es conectada a los fotodetectores.	15
3.7. Ruido random walk entre 0 y 2π que simula el ruido de fase del sistema.	19
3.8. IMZ de 4 brazos reaccionando al ruido del sistema.	19
4.1. Random walk entre 0 y 2π	24
4.2. Sistema completo con ruido.	25
4.3. Comparación brazo 0 con y sin control.	25
4.4. Salidas del sistema con control incluido.	26
5.1. Arquitectura general del sistema.	28

5.2. Máquina de estados módulo Powell.	29
5.3. Máquina de estados módulo Maximiza.	30
5.4. Tarjeta de desarrollo Zybo Z7-20 [1].	31
5.5. Diagrama de reloj conversor análogo-digital AD7476A [9].	32
5.6. Diagrama de reloj conversor digital-análogo DAC121S101 [11].	33
6.1. Resultados de simulación.	37

Capítulo 1

Introducción

1.1. Introducción general

El interferómetro es un dispositivo óptico que utiliza la superposición de ondas coherentes en las cuales se experimentan cambios de fase para la medición de propiedades físicas, sus aplicaciones se han diversificado llegando a campos como la mecánica, biología, astronomía y comunicaciones, en particular en el tratamiento de información cuántica encriptada.

Por otro lado, estos dispositivos son altamente sensibles a las alteraciones físicas del medio, las cuales generan el poco deseado ruido de fase. Para un correcto desarrollo de las aplicaciones antes descritas es necesario apaciguar los efectos del ruido tanto de forma pasiva, con elementos aislantes, como activa, lo que lleva al desarrollo de controladores de fase a incluirse en estos sistemas.

En el presente trabajo se aborda el desarrollo de un algoritmo de control a implementar sobre un interferómetro Mach-Zehnder basado en fibra óptica multi-núcleo de cuatro brazos emplazado en la Universidad de Concepción, el cual busca de una manera rápida, sencilla y eficiente estabilizar las fluctuaciones en la intensidad de salida requerida en dicho sistema.

El flujo de trabajo comienza con la revisión de algoritmos de optimización multiparámetro propuestos, siendo el elegido para este proyecto el algoritmo de Powell. Con esto en mente,

se genera una simulación de la planta en MATLAB a partir de las ecuaciones que modelan el sistema, luego se pasa a probar el algoritmo de Powell en el software MATLAB, con funciones de prueba y finalmente con la correspondiente función del sistema.

Después de esta primera fase se entra a la adaptación del algoritmo de Powell para su implementación sobre un FPGA. La última parte de este trabajo consiste en generar el código en *Verilog* que implementa el algoritmo planteado para luego contrastar los resultados de simulación post-síntesis con lo obtenido de la simulación en software. Esto, sumado al análisis del uso de recursos de la implementación en el FPGA componen los resultados finales del proyecto.

1.2. Objetivos

1.2.1. Objetivo general

Desarrollar plano de control de fase para sistema interferométrico basado en fibra óptica multi-núcleo emplazado en la facultad de ciencias físicas y matemáticas de la Universidad de Concepción.

1.2.2. Objetivos específicos

- Selección de algoritmo de control para garantizar la estabilidad del sistema en toda la zona de operación.
- Evaluación del control con la planta simulada en MATLAB.
- Implementación del algoritmo de control de fase en una FPGA.
- Presentación del desarrollo y resultados de implementación en informe final.

1.3. Alcances y limitaciones

- La propuesta se realiza con base en la planta desarrollada en la Universidad de Concepción.
- La velocidad del controlador está limitada por el hardware a utilizar.
- Los resultados de la implementación del controlador se validan mediante la comparación con los resultados en software.

1.4. Temario

- Capítulo 2: se comienza con una revisión de los fundamentos teóricos que sustentan el proyecto, siendo los más relevantes el funcionamiento del MZI y el algoritmo de control a implementar.
- Capítulo 3: se describe el sistema interferométrico sobre el cual se trabaja y cómo se incluye el plano de control sobre éste.
- Capítulo 4: se explica el algoritmo de control elegido con un funcionamiento ideal.
- Capítulo 5: se explica la implementación del plano de control adaptado.
- Capítulo 6: presenta los resultados obtenidos tanto en software como en hardware en las diversas simulaciones realizadas.
- Capítulo 7: Conclusiones y futuros trabajos derivados del proyecto.

Capítulo 2

Marco teórico

En esta sección se introducen y dan a conocer los términos más relevantes de este proyecto, para contextualizar todo el trabajo posterior, comenzando por el Interferómetro Mach-Zehnder, su funcionamiento y problemáticas, para luego pasar al algoritmo de optimización a implementar.

2.1. Interferómetro Mach-Zehnder

El fenómeno físico de interferencia corresponde al producto de dos o más ondas que se superponen, formando una resultante de mayor, igual o menor amplitud. En el caso de interferencia óptica hablamos de haces de luz coherentes, los cuales pueden ser transmitidos por el aire o por otros medios, como es el caso de la fibra óptica. Existen diversas configuraciones ópticas para construir un sistema que produzca interferencia, algunos ejemplos son los interferómetros de Michelson, Sagnac y Mach-Zehnder. La configuración de este último se detalla a continuación.

El funcionamiento del MZI consiste en la división de un haz de luz coherente cuyas partes toman caminos distintos, para luego ser recombinados en la salida de la configuración [7]. Si los haces divididos mantienen la condición de coherencia, al ser recombinados producirán una interferencia de salida dependiente de la fase, producto de la diferencia de camino. La

fase o diferencia de camino, puede ser variada con elementos activos, como un modulador de fase controlado por voltaje, en cuyo caso la fase tendrá una dependencia en la tensión aplicada al elemento activo [8]. El fenómeno de interferencia se puede expresar como:

$$I(t, V) \propto 1 + \cos [\Delta\phi(t) + \Delta\varphi(V)], \quad (2.1)$$

donde $\Delta\phi(t)$ corresponde a las fluctuaciones de la fase por el medio (o ruido de fase) y $\Delta\varphi(V)$ al desfase causado por el modulador de fase controlado por voltaje [10] y producida por la señal de control.

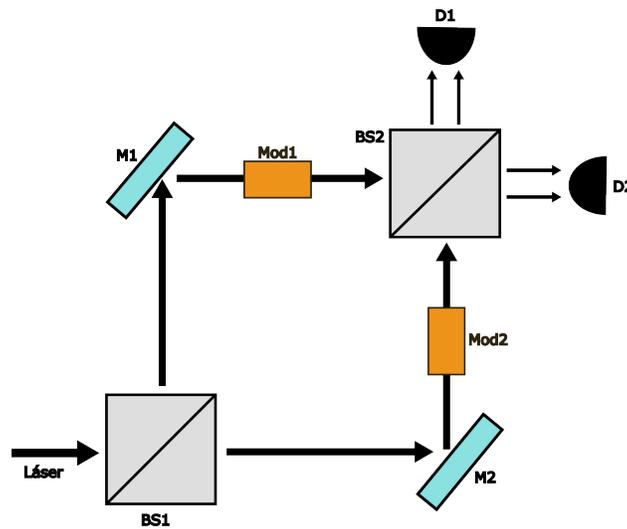


Figura 2.1: Interferómetro Mach-Zehnder de dos brazos

Una representación básica del MZI de dos brazos se observa en la Figura 2.1. Se tiene un haz de luz dividido por un primer beam splitter (BS), luego ambos haces resultantes son reflejados por espejos (M) y modificados a través de los moduladores de fase (Mod) para ser recombinados posteriormente en un segundo BS, finalmente las señales resultantes son captadas por fotodetectores (D).

2.1.1. Ruido de fase

Un problema común en las aplicaciones de ingeniería corresponde al ruido en las señales con las que se desea trabajar, para el caso particular del MZI se experimenta el ruido de

fase, debido al retraso temporal en las ondas propagadas. Este fenómeno se produce por diversas perturbaciones del medio, como alteraciones mecánicas o acústicas en el entorno y es necesario contrarrestarlas para mejorar el desempeño y la precisión del interferómetro.

Una forma de disminuir el ruido de fase es a través de protección pasiva, con un entorno aislante, pero muchas veces esto no es suficiente y surge la necesidad de incorporar un controlador externo en el sistema para maximizar la intensidad en la salida [6].

Como menciona Cariñe [8], es posible controlar el desfase de un interferómetro alterando la diferencia entre caminos. Esto puede realizarse con un elemento piezoeléctrico sobre un espejo en espacio libre, con un stretcher de fibra al utilizar fibra óptica o agregando un modulador de fase al sistema. En este proyecto se considera la tercera opción, incorporando moduladores de fase controlados por voltaje a los brazos del interferómetro.

2.1.2. Óptica matricial

M. Born y E. Wolf [4] explican la óptica matricial como una técnica para trazar rayos paraxiales. En la aproximación paraxial, la posición y el ángulo en los planos de entrada y salida de un sistema óptico están relacionados por dos ecuaciones algebraicas lineales. Como resultado, el sistema óptico se describe mediante una matriz de 2×2 denominada matriz de transferencia de rayos.

La conveniencia de utilizar métodos de matriz radica en el hecho de que la matriz de transferencia de rayos de una cascada de componentes (o sistemas) ópticos es un producto de las matrices de transferencia de rayos de los componentes individuales. Por tanto, la óptica matricial proporciona un mecanismo formal para describir sistemas ópticos complejos en la aproximación paraxial. Luego, los componentes utilizados y sistemas implementados de aquí en adelante serán explicados mediante sus matrices de transferencia.

2.2. Algoritmos de optimización multiparámetro sin derivadas

De los puntos anteriores se desprende que el problema a solucionar es controlar las fluctuaciones de fase del MZI de cuatro brazos, donde la intensidad de salida de cada brazo es dependiente de las fases de todo el sistema. Luego, basta con optimizar uno de los cuatro brazos para tener el sistema controlado. Considerando que el controlador se desea implementar en una FPGA se deben tener en cuenta las limitaciones aritméticas que este tipo de procesamiento presenta, lo cual motiva la búsqueda de métodos de optimización multidimensionales sin derivadas. Algunos de estos métodos son: perturbar y observar[12] , gradiente conjugado, métodos métricos variables[15] y método de Powell[13], el cual se desea trabajar en este proyecto y se presenta a continuación.

2.2.1. Algoritmo de Powell

El algoritmo de Powell es un tipo de optimización multivariable que se desprende del método básico de búsqueda de patrones. Es un método de búsqueda directa ampliamente utilizado y puede demostrarse que es un método de direcciones conjugadas[13]. El algoritmo consiste en realizar sucesivas optimizaciones de una dimensión para luego encontrar un vector de dirección que apunte hacia el punto donde la función a optimizar sea máxima o mínima según sea requerido.

Se comienza con una cantidad de direcciones igual a la cantidad de variables de la función, y estos deben ser ortogonales entre sí. Por tanto, un buen grupo inicial corresponde a vectores que componen la matriz identidad. Esto es, para un caso de 3 dimensiones, por ejemplo,

$$\mathbf{U} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

con,

$$\mathbf{u}'_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{u}'_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \mathbf{u}'_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

En cada dirección se busca el punto máximo, y desde ahí sigue la búsqueda en la siguiente dirección. Una vez que se termina de maximizar en las direcciones iniciales, se procede a buscar una nueva, la cual es el resultado de conectar el punto inicial con el punto alcanzado en la última optimización.

Con esto se actualiza la matriz de direcciones, eliminando la primera fila e incluyendo la nueva dirección, luego de hacer un corrimiento de filas, es decir:

$$\mathbf{U}_n = \begin{pmatrix} u_2 \\ u_3 \\ a \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a_1 & a_2 & a_3 \end{pmatrix}$$

y comienza nuevamente la búsqueda en cada dirección, este proceso se repite sucesivamente hasta llegar a un error mínimo establecido entre la posición actual y la deseada.

Sean $f(x_1, \dots, x_N)$ la función a optimizar, con N cantidad de variables; \mathbf{U} la matriz de direcciones de búsqueda, compuesta por los vectores unitarios $\mathbf{u}_1, \dots, \mathbf{u}_N$, y \mathbf{P}_0 los puntos iniciales, el Algoritmo 2.2.1 muestra este proceso. El cual se puede separar en tres etapas fundamentales:

- **Inicialización (línea 1):** Se fijan algunos parámetros iniciales, como la función a optimizar ($f(\mathbf{x})$), el punto de partida (\mathbf{P}_0), la matriz de direcciones iniciales (\mathbf{U}) y el error máximo aceptable (ϵ).
- **Direcciones ortogonales (líneas 3 a 5):** comienza la optimización en cada una de las direcciones iniciales. En caso de tener error menor al impuesto, se detiene la búsqueda en esta parte.

Algoritmo 1 Procedimiento básico de Powell[14]

- 1: Guarde la posición inicial como \mathbf{P}_0 .
 - 2: **mientras** $error > \epsilon$ **hacer**
 - 3: **para** $i \leftarrow 0, N - 1$ **hacer**
 - 4: Mueva \mathbf{P}_i al máximo a lo largo de la dirección \mathbf{u}_i y llame a este punto \mathbf{P}_{i+1} .
 - 5: **fin para**
 - 6: **para** $i \leftarrow 0, N - 2$ **hacer**
 - 7: $\mathbf{u}_i \leftarrow \mathbf{u}_{i+1}$.
 - 8: **fin para**
 - 9: $\mathbf{u}_{N-1} \leftarrow \mathbf{P}_N - \mathbf{P}_0$.
 - 10: Mueva \mathbf{P}_N al máximo a lo largo de la dirección \mathbf{u}_{N-1} y llame a este punto \mathbf{P}_0 .
 - 11: **fin mientras**
-

- **Actualiza matriz de direcciones (líneas 6 a 9):** luego de optimizar cada en cada una de las direcciones antes mencionadas, se realiza el corrimiento de vectores para actualizar la matriz y luego se restan el punto actual y el inicial, para obtener la nueva dirección de búsqueda y se agrega la nueva fila a la matriz de direcciones.

En la Figura 2.2 se aprecia el paso a paso del algoritmo de Powell en dos dimensiones. Siendo los primeros dos trazos correspondientes a las direcciones iniciales (ejes x e y respectivamente), luego se repite la búsqueda en estos mismos ejes y finalmente se genera la búsqueda en la dirección resultante (diagonal).

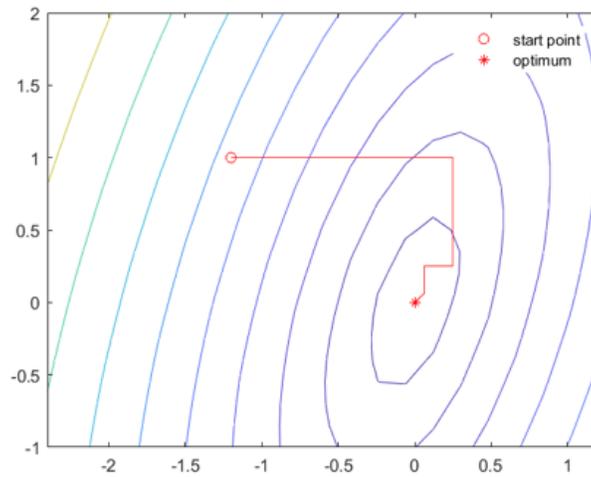


Figura 2.2: Aplicación método de Powell en 2 dimensiones

Capítulo 3

Sistema interferométrico implementado en Universidad de Concepción

En el laboratorio 203 de la Facultad de Física de la Universidad de Concepción se encuentra emplazado el sistema a controlar, el cual consiste en un interferómetro Mach-Zehnder de 4 brazos basado en fibra óptica multi-núcleo (MCF). Dicha planta consta de los siguientes componentes y se puede observar su esquema en la Figura 3.1.

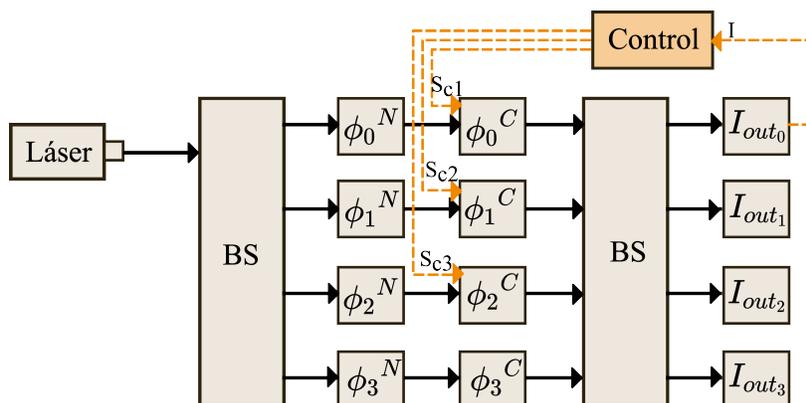


Figura 3.1: Diagrama MZI. BS: beam splitter, ϕ^N : efecto del ruido de fase, ϕ^C : acción de control, I_{out} : intensidad de salida. Lo resaltado en naranja corresponde al lazo de control a implementar, donde I es la entrada al controlador y S_{c1} , S_{c2} y S_{c3} las tres señales de salida.

3.1. Componentes del sistema

3.1.1. Beam splitter

Dispositivo óptico que divide la luz en dos o más haces dependiendo de su configuración. Es parte esencial de diversos sistemas ópticos, entre ellos, los interferómetros. Para este caso particular se tienen dos BS que dividen los haces de luz en cuatro brazos y están fabricados en MCF. En la Sección 3.2.1 se puede revisar su expresión matricial.

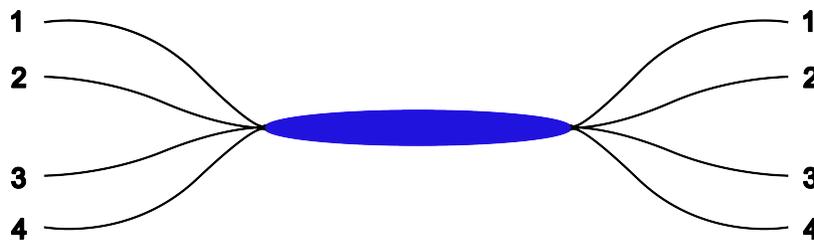


Figura 3.2: BS de 4 brazos

3.1.2. Controlador manual de polarización

El controlador manual de polarización, o CPM por su nombre en inglés, es un dispositivo capaz de controlar y/o cambiar el estado de polarización de un haz de entrada. Esto lo logra realizando torsiones a una fibra óptica que recorre por el interior del dispositivo, como muestra la Figura 3.3, con ellos se busca maximizar la intensidad de salida en el sistema.

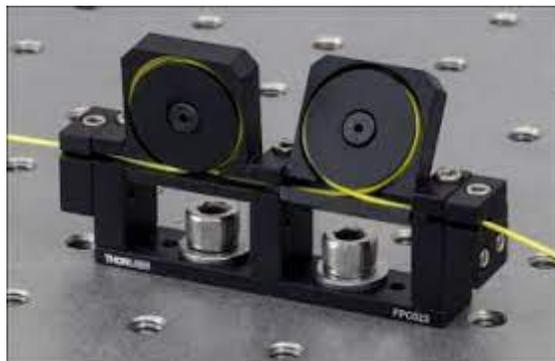


Figura 3.3: Controlador manual de polarización.

3.1.3. Modulador de fase

Este aparato opto-electrónico consiste en un modulador en el cual, al ingresar un voltaje, se genera un campo magnético que altera el índice de refracción del material utilizado. La tensión necesaria para provocar un desfase de 180° corresponde a V_π , como se aprecia en la Figura 3.4. Este es uno de los parámetros más relevantes al momento de escoger un modulador de fase, lo ideal es que dicho voltaje sea bajo, para consumir la menor potencia eléctrica durante la modulación. Sin embargo, esto provoca que el modulador sea más lento, por lo que se deben sopesar estas dos características al momento de decidir cuál modulador utilizar [16].

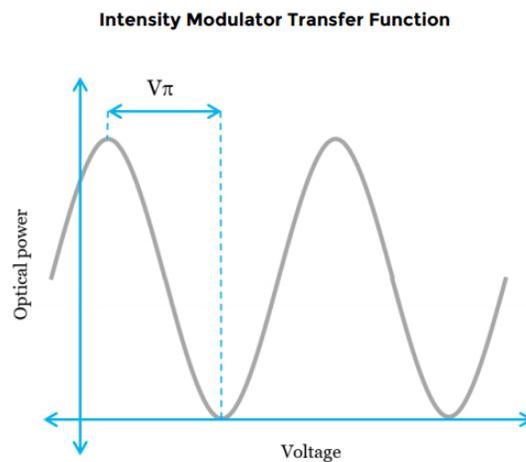


Figura 3.4: Voltaje de entrada v/s potencia de salida modulador[2].

3.1.4. Fotodetector

Un fotodetector es un dispositivo electrónico que convierte los fotones recibidos en corriente eléctrica, esto sucede cuando la energía del fotón incidente libera un electrón desde la banda de valencia. En particular, los dispositivos utilizados en esta implementación reciben luz a través de fibra óptica. En la Figura 3.5 se observa el modelo de fotodetector utilizado en esta implementación, el cual tiene una señal de salida que va entre los 0 y 5 V.



Figura 3.5: Fotodetector para fibra óptica.

3.2. Interferómetro de 4 caminos

Al igual que el modelo de dos brazos visto en la Figura 2.1, en el interferómetro de cuatro caminos se incluyen dos BS, uno al inicio y otro al final. Considerando en este caso una configuración donde los haces viajen por fibra óptica en lugar del aire, se suprime el uso de los espejos y se utiliza MCF para cada uno de los brazos. Por último, los moduladores se siguen manteniendo. En la Figura 3.6 se observa la disposición real del MZI de 4 brazos con los componentes antes mencionados y su forma esquemática se presenta en la Figura 3.1.

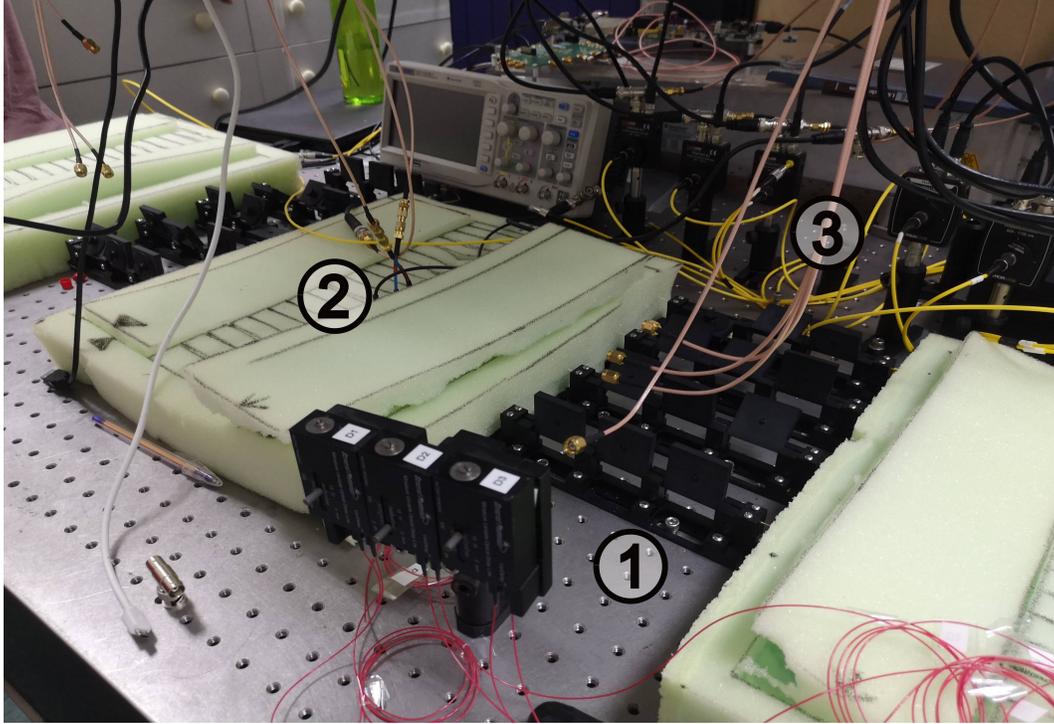


Figura 3.6: Interferómetro real. (1) El láser ingresa por la parte derecha para luego pasar por el BS1 y los polarizadores manuales. (2) Conexión de señales de entrada a los moduladores. (3) Luego de pasar por los segundos polarizadores y el BS2, la salida es conectada a los fotodetectores.

Aplicando la óptica matricial a este sistema, se puede modelar su comportamiento como se muestra a continuación, siendo M_{BS} el beam splitter, $M_{\phi\tau}$ la matriz de fase y ϕ_i^n , con ϕ_i^c ruido de fase y control de fase respectivamente.

3.2.1. Beam splitter de 4×4

Para el caso particular de un BS en fibra óptica multinúcleo de 4×4 , se tiene la representación matricial

$$M_{BS_{4 \times 4}} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & e^{j\phi} & -1 & -e^{j\phi} \\ 1 & -1 & 1 & -1 \\ 1 & -e^{j\phi} & -1 & e^{j\phi} \end{pmatrix}, \quad (3.1)$$

cuyo desarrollo se puede observar en el artículo [5] para más información. Ahora, para un BS que no presenta cambios de fase por la arquitectura del sistema, se puede asumir que $\phi = 0$. De esta forma, se presenta la matriz de transferencia asociada a un BS multipuerto de 4×4 dada por

$$M_{BS_{4 \times 4}} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}. \quad (3.2)$$

3.2.2. Matriz de fase

La matriz de fase es una matriz diagonal que resume las fluctuaciones de fase de cada brazo del interferómetro. Se tiene una ganancia de transmisión τ , manipulable a través de los controladores manuales de polarización; el ruido de fase ϕ_i^n , producido por las perturbaciones del sistema y, finalmente, el control de fase ϕ_i^c que viene dado por los moduladores de fase. Esta matriz está definida como:

$$M_{\phi\tau} = \begin{pmatrix} \tau e^{j(\phi_0^n + \phi_i^c)} & 0 & 0 & 0 \\ 0 & \tau e^{j(\phi_1^n + \phi_i^c)} & 0 & 0 \\ 0 & 0 & \tau e^{j(\phi_i^2 + \phi_i^c)} & 0 \\ 0 & 0 & 0 & \tau e^{j(\phi_i^3 + \phi_i^c)} \end{pmatrix}. \quad (3.3)$$

Considerando que la ganancia de transmisión τ puede ser manipulada manualmente en los ajustes del sistema para ser unitaria, se considerará de ahora en adelante igual a uno. Esto se logra maximizando la señal de salida al ajustar los controladores manuales de polarización.

3.2.3. Sistema completo

Como se explica en la Sección 2.1.2, una representación matricial del MZI de cuatro caminos viene dada por la multiplicación de los componentes que constituyen el sistema, lo cual se

muestra a continuación:

$$\begin{pmatrix} O_0 \\ O_1 \\ O_2 \\ O_3 \end{pmatrix} = BSM_{\phi\tau} BS \begin{pmatrix} E_0 \\ E_1 \\ E_2 \\ E_3 \end{pmatrix} \quad (3.4)$$

$$= \frac{1}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} e^{j(\phi_0^n + \phi_0^c)} & 0 & 0 & 0 \\ 0 & e^{j(\phi_1^n + \phi_1^c)} & 0 & 0 \\ 0 & 0 & e^{j(\phi_2^n + \phi_2^c)} & 0 \\ 0 & 0 & 0 & e^{j(\phi_3^n + \phi_3^c)} \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} E_0 \\ E_1 \\ E_2 \\ E_3 \end{pmatrix} \quad (3.5)$$

Ahora, siendo la entrada al sistema máxima en uno de los brazos y nula en los demás, se tiene:

$$= \frac{1}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} e^{j(\phi_0^n + \phi_0^c)} & 0 & 0 & 0 \\ 0 & e^{j(\phi_1^n + \phi_1^c)} & 0 & 0 \\ 0 & 0 & e^{j(\phi_2^n + \phi_2^c)} & 0 \\ 0 & 0 & 0 & e^{j(\phi_3^n + \phi_3^c)} \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.6)$$

$$\begin{pmatrix} O_0 \\ O_1 \\ O_2 \\ O_3 \end{pmatrix} = \frac{1}{4} \begin{pmatrix} e^{j(\phi_0^n + \phi_0^c)} + e^{j(\phi_1^n + \phi_1^c)} + e^{j(\phi_2^n + \phi_2^c)} + e^{j(\phi_3^n + \phi_3^c)} \\ e^{j(\phi_0^n + \phi_0^c)} + e^{j(\phi_1^n + \phi_1^c)} - e^{j(\phi_2^n + \phi_2^c)} - e^{j(\phi_3^n + \phi_3^c)} \\ e^{j(\phi_0^n + \phi_0^c)} - e^{j(\phi_1^n + \phi_1^c)} + e^{j(\phi_2^n + \phi_2^c)} - e^{j(\phi_3^n + \phi_3^c)} \\ e^{j(\phi_0^n + \phi_0^c)} - e^{j(\phi_1^n + \phi_1^c)} - e^{j(\phi_2^n + \phi_2^c)} + e^{j(\phi_3^n + \phi_3^c)} \end{pmatrix}, \quad (3.7)$$

Luego, las intensidades de salida vienen dadas por:

$$I_i = |O_i|^2 = \bar{O}_i O_i, \quad (3.8)$$

Considerando la fase ϕ_i como la suma de la fase controlada y el ruido de fase $\phi_i^n + \phi_i^c$, las salidas del sistema corresponden a las siguientes expresiones:

$$I_0 = \frac{1}{16} [1 + e^{j(\phi_0 - \phi_1)} + e^{j(\phi_0 - \phi_2)} + e^{j(\phi_0 - \phi_3)} + e^{-j(\phi_0 + \phi_1)} + 1 + e^{j(\phi_1 - \phi_2)} + e^{j(\phi_1 - \phi_3)} + e^{-j(\phi_0 - \phi_2)} + e^{-j(\phi_1 - \phi_2)} + 1 + e^{j(\phi_2 - \phi_3)} + e^{-j(\phi_0 - \phi_3)} + e^{-j(\phi_1 - \phi_3)} + e^{-j(\phi_2 - \phi_3)} + 1] \quad (3.9)$$

$$I_1 = \frac{1}{16} [1 + e^{j(\phi_0 - \phi_1)} - e^{j(\phi_0 - \phi_2)} - e^{j(\phi_0 - \phi_3)} + e^{-j(\phi_0 + \phi_1)} + 1 - e^{j(\phi_1 - \phi_2)} - e^{j(\phi_1 - \phi_3)} - e^{-j(\phi_0 - \phi_2)} - e^{-j(\phi_1 - \phi_2)} + 1 + e^{j(\phi_2 - \phi_3)} - e^{-j(\phi_0 - \phi_3)} - e^{-j(\phi_1 - \phi_3)} + e^{-j(\phi_2 - \phi_3)} + 1] \quad (3.10)$$

$$I_2 = \frac{1}{16} [1 - e^{j(\phi_0 - \phi_1)} + e^{j(\phi_0 - \phi_2)} - e^{j(\phi_0 - \phi_3)} - e^{-j(\phi_0 + \phi_1)} + 1 - e^{j(\phi_1 - \phi_2)} + e^{j(\phi_1 - \phi_3)} + e^{-j(\phi_0 - \phi_2)} - e^{-j(\phi_1 - \phi_2)} + 1 - e^{j(\phi_2 - \phi_3)} - e^{-j(\phi_0 - \phi_3)} + e^{-j(\phi_1 - \phi_3)} - e^{-j(\phi_2 - \phi_3)} + 1] \quad (3.11)$$

$$I_3 = \frac{1}{16} [1 - e^{j(\phi_0 - \phi_1)} - e^{j(\phi_0 - \phi_2)} + e^{j(\phi_0 - \phi_3)} - e^{-j(\phi_0 + \phi_1)} + 1 + e^{j(\phi_1 - \phi_2)} - e^{j(\phi_1 - \phi_3)} - e^{-j(\phi_0 - \phi_2)} + e^{-j(\phi_1 - \phi_2)} + 1 - e^{j(\phi_2 - \phi_3)} + e^{-j(\phi_0 - \phi_3)} - e^{-j(\phi_1 - \phi_3)} - e^{-j(\phi_2 - \phi_3)} + 1]. \quad (3.12)$$

Finalmente, por identidades trigonométricas se tiene que $e^{j\theta} + e^{-j\theta} = 2 \cdot \cos\theta$, con lo cual las expresiones previas se reducen a la siguiente forma:

$$I_0 = \frac{1}{8} [2 + \cos(\phi_0 - \phi_1) + \cos(\phi_0 - \phi_2) + \cos(\phi_0 - \phi_3) + \cos(\phi_1 - \phi_2) + \cos(\phi_1 - \phi_3) + \cos(\phi_2 - \phi_3)] \quad (3.13)$$

$$I_1 = \frac{1}{8} [2 + \cos(\phi_0 - \phi_1) - \cos(\phi_0 - \phi_2) - \cos(\phi_0 - \phi_3) - \cos(\phi_1 - \phi_2) - \cos(\phi_1 - \phi_3) + \cos(\phi_2 - \phi_3)] \quad (3.14)$$

$$I_2 = \frac{1}{8} [2 - \cos(\phi_0 - \phi_1) + \cos(\phi_0 - \phi_2) - \cos(\phi_0 - \phi_3) - \cos(\phi_1 - \phi_2) + \cos(\phi_1 - \phi_3) - \cos(\phi_2 - \phi_3)] \quad (3.15)$$

$$I_3 = \frac{1}{8} [2 - \cos(\phi_0 - \phi_1) - \cos(\phi_0 - \phi_2) + \cos(\phi_0 - \phi_3) + \cos(\phi_1 - \phi_2) - \cos(\phi_1 - \phi_3) - \cos(\phi_2 - \phi_3)], \quad (3.16)$$

Siendo estas las ecuaciones que modelan el sistema a controlar.

3.3. Simulación del sistema

El sistema de ecuaciones obtenido en la sección anterior se puede modelar y simular en software para observar el comportamiento del sistema final y cuáles son las variables que

alteran su comportamiento.

En la Figura 3.7 se observa el comportamiento de un ruido de fase fluctuante y que se amplifica en el tiempo, específicamente del tipo *random walk* acotado entre 0 y 2π que es el dominio de la fase.

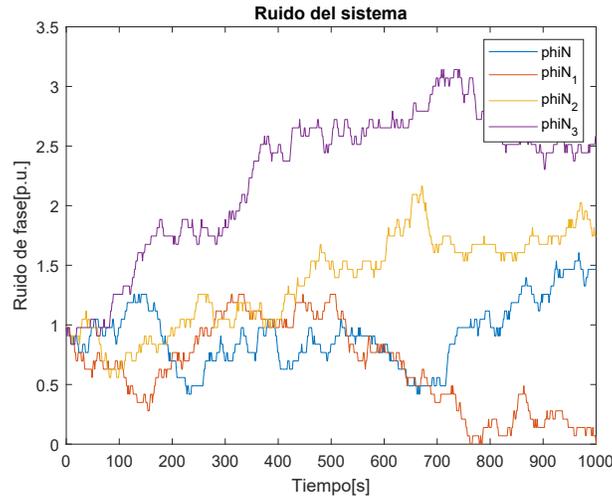


Figura 3.7: Ruido random walk entre 0 y 2π que simula el ruido de fase del sistema.

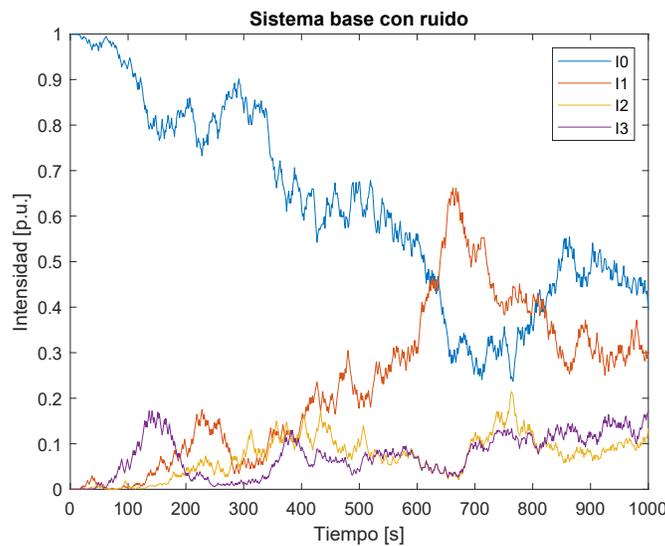


Figura 3.8: IMZ de 4 brazos reaccionando al ruido del sistema.

En respuesta a esto, en la Figura 3.8 se observa como el sistema comienza a disminuir su salida máxima de intensidad y las salidas de todos los brazos se ven alteradas, sin embargo,

la sumatoria de las señales de salida es constante e igual a 1, ya que no se toman en cuenta las posibles pérdidas presentes en un sistema real de gran longitud.

Capítulo 4

Control ideal

En este capítulo se aplica el algoritmo de Powell mostrado en la Sección 2.2.1 como optimizador de un sistema, el cual se prueba a través de la implementación en software. Esto muestra el paso a paso del algoritmo y como afecta la decisión del algoritmo interno de optimización unidimensional a utilizar. El programa implementado y sus resultados se explican a cabalidad a continuación.

4.1. Programa a probar

Giovani Tonel[3] presentó una implementación del algoritmo de Powell para el software MATLAB, el cual se adapta en este trabajo para probar la respuesta al modelo del MZI de 2 y 4 brazos.

El programa recibe como datos de entrada los que se muestran a continuación:

- **Función a maximizar:** ecuación descriptiva de uno de los brazos del sistema (3.13).
- **Método de optimización:** coggins o golden section.
- **Puntos óptimos a los que se debe llegar:** intensidades de salida.
- **Error máximo:** diferencia entre la intensidad óptima y la actual.

- **Punto inicial:** fases de partida.

Y como datos de salida:

- Iteraciones realizadas.
- Puntos alcanzados.
- Salida de la función (se incluye a modo de validación para los resultados).

4.1.1. Maximización unidimensional

El algoritmo de Powell utiliza un parámetro denominado α en cada una de las iteraciones se debe encontrar dicho valor. Para obtener este parámetro existen diferentes modalidades, en este caso el algoritmo ideal fue probado con una ecuación de tercer orden y aplicando dos métodos distintos para encontrar α , los cuales son *Coggins* y *Golden section*.

4.1.2. Implementación en software

Se realizan pruebas con el algoritmo descrito a las funciones del modelo de IMZ de 1 y 4 brazos, probando los dos métodos para encontrar α . Las pruebas 1, 3 y 5 en ambas tablas utilizan el método de *Coggins*, mientras que las pruebas 2,4 y 6 usan la optimización *Golden section*. Es relevante mencionar que las señales de entrada en todos los casos corresponden a números entre 0 y 2π , lo que queda dentro del rango de las fases para el modelo.

Tabla 4.1: Resultados de la aplicación del algoritmo de Powell al maximizar la función de un brazo en MZI de 2 brazos.

Prueba 1 <i>ns</i> = 34			Prueba 2 <i>ns</i> = 105			Prueba 3 <i>ns</i> = 41			Prueba 4 <i>ns</i> = 87			Prueba 5 <i>ns</i> = 41			Prueba 6 <i>ns</i> = 87		
x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o
0.1	0.2	1	0.1	0,1726	1	1	2	1	1	2	1	5	6.2	1	5	6,1999	1
0.2	0.2	0	0.2	0.1726	$1,7383e - 10$	2	2	0	2	2	0	6.2	6.2	0	6.2	6,1999	$1,1932e - 10$

Tabla 4.2: Resultados de la aplicación del algoritmo de Powell al maximizar la función de un brazo en MZI de 4 brazos.

Prueba 1 $ns = 170$			Prueba 2 $ns = 363$			Prueba 3 $ns = 207$			Prueba 4 $ns = 430$			Prueba 5 $ns = 220$			Prueba 6 $ns = 445$		
x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o	x_0	x_n	I_o
0.1	0.3176	1	0.1	0.3176	1	1	3.1737	1	1	3.1736	1	5	4.2731	1	5	4.2732	1
0.2	0.3176	$1.3878e - 17$	0.2	0.3177	$3.5188e - 11$	2	3.1737	0	2	3.1736	$2.3731e - 11$	6.2	4.2732	$8.1781e - 11$	6.2	4.2732	$3.3494e - 12$
0.3	0.3176	$-1.3878e - 17$	0.3	0.3176	$1.5917e - 11$	3	3.1737	0	3	3.1736	$6.9016e - 12$	3	4.2731	$1.1441e - 10$	3	4.2732	$1.0136e - 10$
0.4	0.3176	0	0.4	0.3176	$1.3860e - 10$	4	3.1737	0	4	3.1736	$1.0058e - 10$	4.5	4.2731	$3.5244e - 11$	4.5	4.2732	$1.2756e - 11$

Estos resultados confirman que la optimización *Golden section* es más precisa en los resultados, pero realizando muchas más iteraciones, en cambio *Coggins* tiene resultados igualmente aceptables en menos tiempo de ejecución.

4.2. Adaptación del algoritmo de Powell

Teniendo en cuenta el algoritmo original presentado en el Capítulo 2 se puede proceder a la explicación del algoritmo a implementar y el por qué de esta adaptación.

Para implementar el algoritmo de Powell original es necesario conocer la función a optimizar, el punto inicial (X_0), que en este caso corresponde al desfase de cada brazo, y el resultado final al que se quiere llegar, es decir, las intensidades de salida óptimas para el sistema. Lamentablemente, en esta implementación no se puede saber a priori cuáles son las fases exactas en las cuales se encuentra el sistema, ya que con los fotodetectores se obtiene solo la intensidad de salida y con el modelo del sistema se debiesen inferir las fases, pero esto puede no ser exacto, debido a la periodicidad de las funciones trigonométricas, tal como se muestra en las Ecuaciones (3.13), (3.14), (3.15) y (3.16). Luego, se decide extraer los pasos principales del algoritmo, pero realizando la optimización de cada dimensión (brazo) maximizando la salida a través de la variación de fase de cada uno de los brazos por separado, como si fuesen direcciones ortogonales, para luego mover todos en conjunto en una nueva ‘dirección’.

Algoritmo 2 Adaptación Powell

- 1: Maximiza intensidad 1 variando brazo 2.
 - 2: Maximiza intensidad 1 variando brazo 3.
 - 3: Maximiza intensidad 1 variando brazo 4.
 - 4: Mueve brazos 2, 3 y 4.
-

4.3. Simulación lazo de control en software

En MATLAB se modela el interferómetro Mach-Zehnder y su respectivo lazo de control. Según lo presentado en la Sección 3.2.3, la simulación en software se realiza utilizando el modelo básico del MZI de 4 brazos, con una entrada total de 1 y ruido del tipo *random walk* acotado entre 0 y 2π . Luego, esto pasa por el sistema de control *Powell adaptado*, los resultados obtenidos son los siguientes.

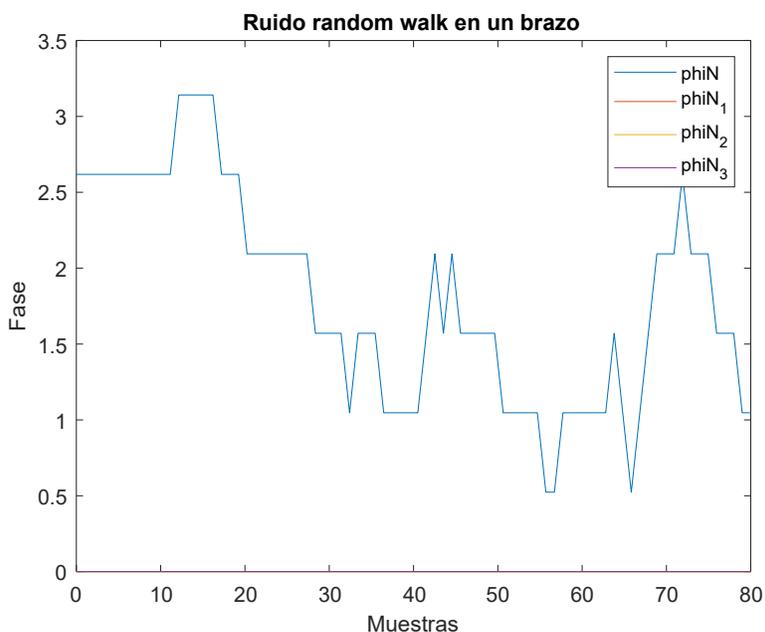


Figura 4.1: Random walk entre 0 y 2π .

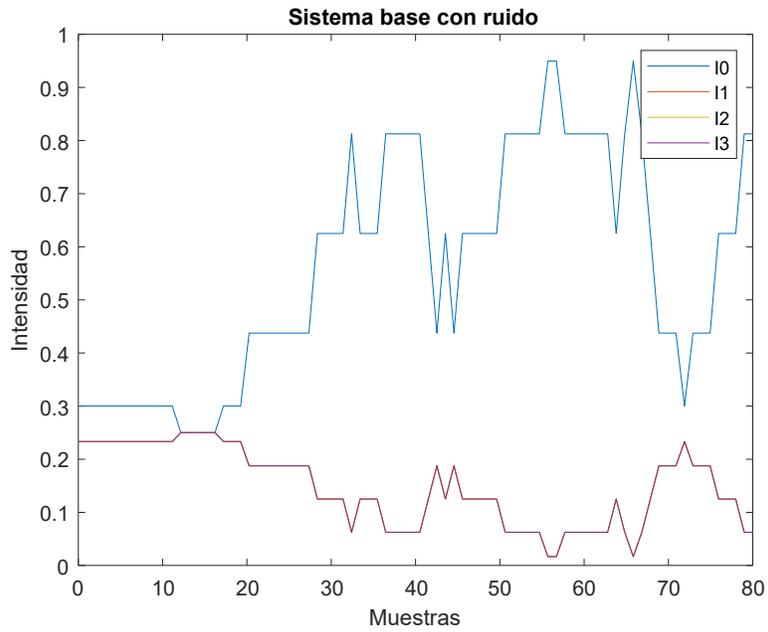


Figura 4.2: Sistema completo con ruido.

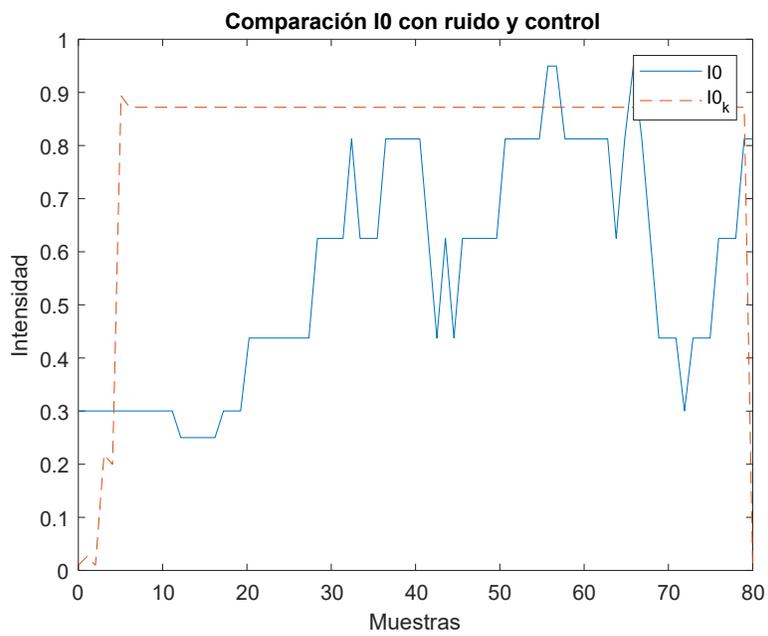


Figura 4.3: Comparación brazo 0 con y sin control.

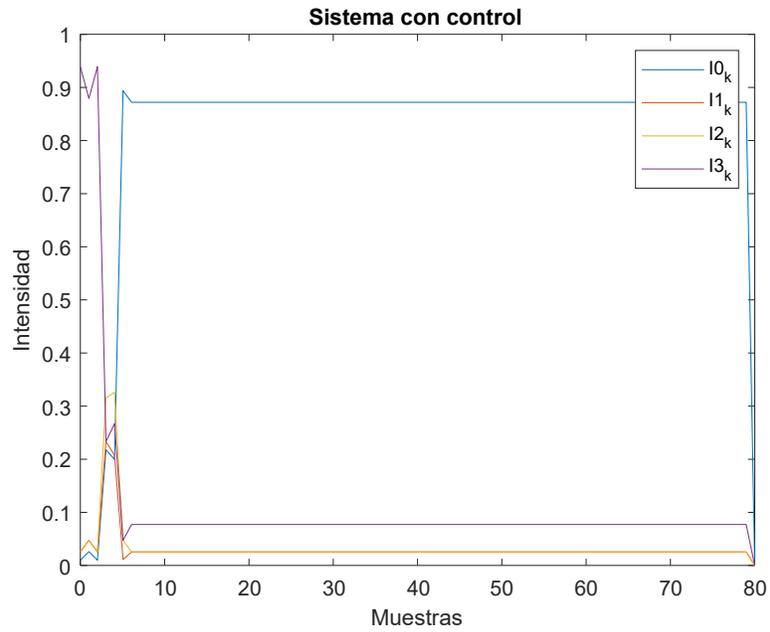


Figura 4.4: Salidas del sistema con control incluido.

A pesar de las variaciones en la entrada, el algoritmo logra maximizar y estabilizar la salida deseada, correspondiente a la intensidad de salida del brazo 0, lo que además ocurre relativamente rápido y sin sobrepasos peligrosos para los actuadores.

Capítulo 5

Arquitectura del controlador

En este capítulo se describe la implementación en hardware del controlador previamente explicado, para ello se generan tres módulos principales en lenguaje de descripción de hardware *Verilog*, uno que comunica el control con la entrada y salida de datos llamada *Control de fase* y otro que engloba al algoritmo, llamado *Powell*, en el cual se instancian los otros dos módulos restantes, *Maximiza* y *Mueve*. A continuación se especifica el funcionamiento de cada uno.

Además de esto, se instancia repetidamente un módulo *One shot* para generar las señales de habilitación entre módulos.

5.1. Arquitectura general

La arquitectura general del sistema se desarrolló en hardware, teniendo datos recibidos y enviados a través de los puertos PMOD. Como se muestra en la Figura 5.1, se tiene una primera parte que controla la obtención de datos, luego la parte central del proceso correspondiente al algoritmo de control de fase y finalmente el módulo encargado de entregar la información al sistema.

Las señales de entrada y salida del sistema corresponden a la señal entregada por los fotodetectores y el voltaje de accionamiento a los moduladores respectivamente.

- **Entradas:** una señal de entre 0 y 5V dada por el fotodetector que receptiona la salida del primer brazo del interferómetro. Luego de pasar por un adaptador, esta señal queda en el rango de 0 a 3.3V.
- **Salidas:** 3 señales de entre 0 y 3.3V que son amplificadas a la salida para activar los moduladores de fase.

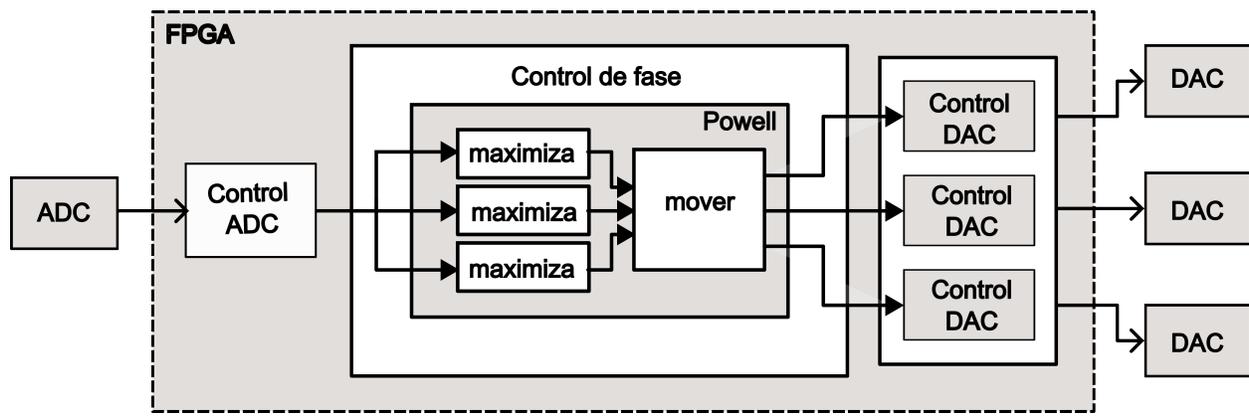


Figura 5.1: Arquitectura general del sistema.

5.2. Arquitectura algoritmo de control

5.2.1. Powell

En este módulo se define el paso a paso del algoritmo de Powell adaptado [2] y se instancian los módulos internos que realizan las acciones de maximizar y mover.

La máquina de Moore mostrada en la Figura 5.2 está compuesta por cinco estados, donde el primero corresponde a la inicialización de las variables y los tres siguientes al control de los brazos del interferómetro, lo que se desencadena al dar la señal de alto que activa el módulo *maximiza*. Por último, el estado 4 activa el módulo *mover* el cual es encargado de llevar a cabo el último movimiento de cada brazo.

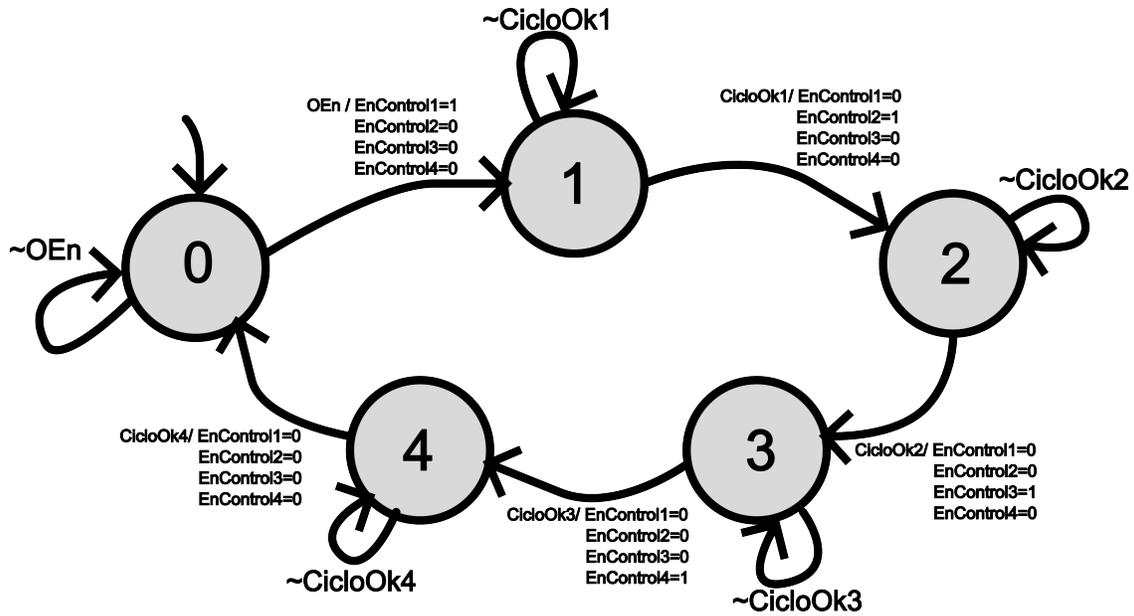


Figura 5.2: Máquina de estados módulo Powell.

A continuación se desglosan los módulos mencionados y se explica el funcionamiento interno de cada uno.

5.2.2. Maximiza

Este módulo busca maximizar la intensidad de salida de uno de los brazos modificando la fase de un solo brazo. Su funcionamiento se basa en la comparación entre la intensidad actual y una anterior, dependiendo de este resultado se suma o resta un delta de fase al modulador que está siendo controlado hasta llegar al punto máximo. Siendo M definida por:

$$M = \begin{cases} 1 & \text{si } I_{ant} \leq I_{act} \\ 0 & \text{si } I_{ant} > I_{act} \end{cases} \quad (5.1)$$

Se ingresa la intensidad de salida, la cual es guardada en un registro para comparar su estado actual con el inmediatamente anterior, con esto se determina la variación de intensidad hasta encontrar el punto máximo, como se muestra en la máquina de estados de la Figura 5.3. Para generar cambios en dicha entrada se busca cambiar la señal de control en un delta fijo las veces que sea necesario, ya sea aumentando o disminuyendo la señal en tal cantidad. El

Punto será máximo cuando $M = 1$ e inmediatamente después $M = 0$. Se incluye, además, un contador que acumula la cantidad de veces que se suma un paso y disminuye en caso contrario.

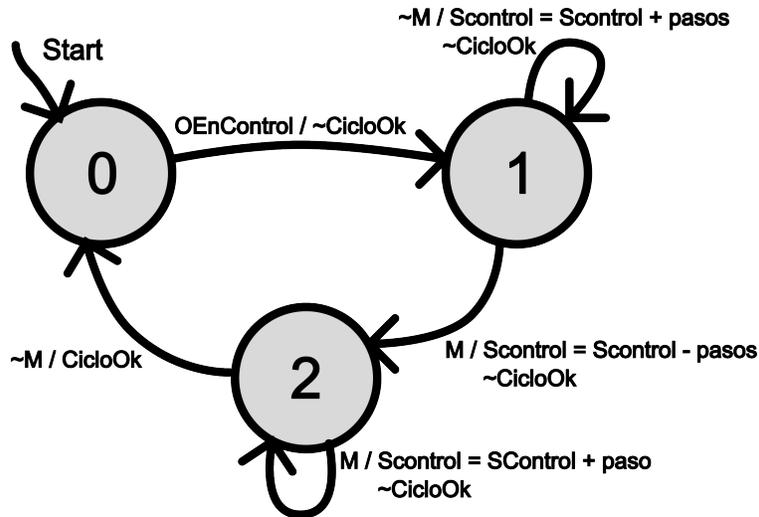


Figura 5.3: Máquina de estados módulo Maximiza.

5.2.3. Mover

Este módulo es el encargado de modificar la señal de salida para los tres moduladores al mismo tiempo. Las señales generadas son proporcionales a la cantidad de movimientos previos que se realizó en cada uno, ya que se utiliza el contador antes mencionado, este resultado se suma a la salida anterior de cada uno

$$S = S + pasos \cdot contador. \quad (5.2)$$

5.3. Hardware

En esta sección se detalla el hardware utilizado para el desarrollo y validación de la arquitectura descrita en el presente capítulo. Incluyendo la tarjeta de desarrollo, los conversores análogo-digital (ADC) para la adquisición de los datos de salida de los fotodetectores y

los conversores digital-análogo (DAC) utilizados para entregar las señales de entrada a los moduladores de fase.

5.3.1. Zybo Z7-20

Es una tarjeta de desarrollo diseñada por Digilent y construida alrededor de un FPGA Zynq™-7000 de Xilinx. Variados periféricos multimedia y de conectividad hacen de esta placa una muy buena opción de trabajo. En particular, para este proyecto se utilizarán los conectores Pmod para añadir los conversores señalados en los siguientes puntos. A continuación se detallan algunas características relevantes de esta placa.

- SoC Zynq-7000.
- 1 GB de memoria DDR3L with 32-bit bus @ 1066 MHz.
- USB-UART bridge.
- 6 puertos Pmod.
- 4 switch, 5 LEDs y 6 botones para uso de usuario.

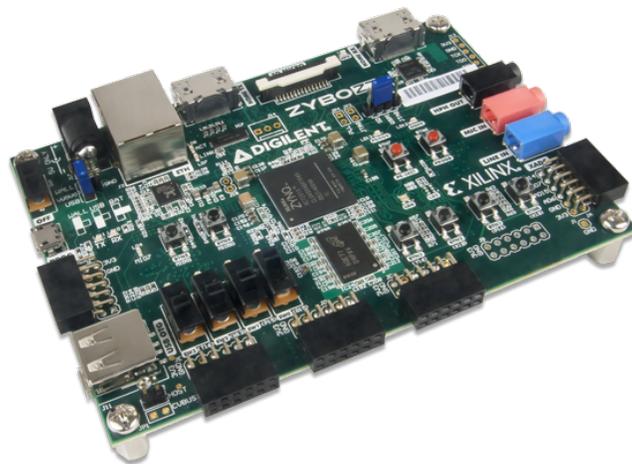


Figura 5.4: Tarjeta de desarrollo Zybo Z7-20 [1].

5.3.2. Pmod AD1

Compuesto por dos conversores análogo/digital ADCS7476MSPS de 12 bits y un μs de tiempo de conversión, además de dos filtros anti-aliasing, uno para cada entrada, con frecuencia de corte en 500kHz. Permite entradas análogas de 0 a V_{cc} , que conectado a la tarjeta de desarrollo de Zybo corresponde a 3.3V.

Los conversores antes mencionados son de aproximaciones sucesivas y funcionan a una frecuencia máxima de operación de 20MHz.

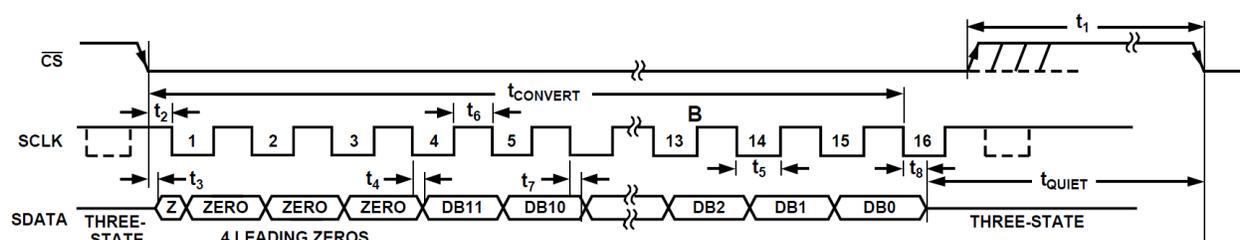


Figura 5.5: Diagrama de reloj conversor análogo-digital AD7476A [9].

Siguiendo el diagrama de reloj de la Figura 5.5 se puede desprender el protocolo a seguir para el procesamiento de la data ingresada a través de este periférico, todo esto se controla en el módulo *Control ADC* que se detalla a continuación. Un dato relevante corresponde al tiempo de silencio necesario antes de comenzar el procesamiento de una nueva señal, el cual es de 50ns como mínimo.

5.3.3. Pmod DA2

Esta tarjeta posee dos conversores digital/análogo DAC121S101 de 12 bits y 8 μs de tiempo de estabilización. Las salidas análogas van de 0 a V_{cc} , que igual que en el caso anterior corresponde a 3.3V. Frecuencia máxima de transmisión serial es de 30MHz.

Las salidas de 12 bits están compuestas por dos bits de términos superfluos, 2 de modo de operación, siendo '00' correspondiente al modo normal, finalmente, los últimos 8 bits son de datos.

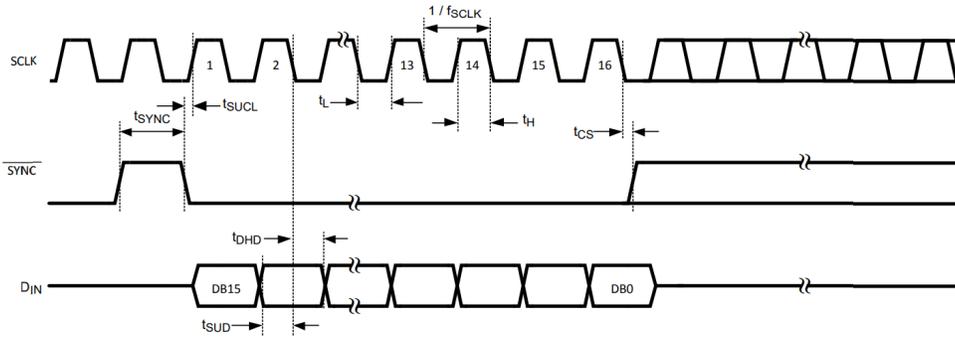


Figura 5.6: Diagrama de reloj conversor digital-análogo DAC121S101 [11].

Al igual que con el periférico anterior, de las indicaciones entregadas por el fabricante se puede extraer el protocolo a seguir para generar la conversión correctamente, en el diagrama de la Figura 5.6 se observa lo anteriormente descrito.

Capítulo 6

Resultados

La arquitectura mostrada en el Capítulo 4 se implementó en el FPGA utilizando el lenguaje de descripción de hardware *Verilog*.

En este capítulo se muestran los resultados de simulación post-síntesis obtenidos de la implementación en hardware, mostrando el uso de recursos lógicos. La validación del control en hardware se realiza mediante la comparación de los datos obtenidos en la simulación en software del sistema. Esto es, extraer las señales de la salida del sistema (Intensidad de salida) y las señales de control entrantes al interferómetro (CS: control signals), dichas señales se preprocesan (re-escalamiento) para alcanzar los valores obtenidos por los adaptadores y luego iniciar las pruebas en hardware.

La metodología de prueba consiste en realizar el control en lazo cerrado en software (MATLAB), luego se abre el lazo, se extraen las variables relevantes y se contrastan resultados con los obtenidos del sistema de control en lazo abierto implementado en hardware.

6.1. Equivalencia de magnitudes

- **Salida del sistema / entrada al controlador:** esta magnitud corresponde a un voltaje entre 0 y 1 en simulación del sistema, pero es detectado por el fotodetector en

un rango de 0 a 5 V y luego de pasar por un adaptador de voltaje y un ADC entra al sistema de control (FPGA) como una señal de 12 bits correspondiente a valores entre 0 y 4095.

- **Entrada al modulador / salida del controlador:** considerando que el DAC tiene una resolución de 8 bits de datos, se utiliza este rango completo para definir la señal de salida del controlador (rango de 0 a 254), esto a su vez pasa por un adaptador que genera voltajes entre 0 y 5V para entrar al modulador controlado por voltaje. Finalmente, esto corresponde a un desfase entre 0 y pi como salida del modulador.

Luego, el valor adecuado para el '*paso*' se puede considerar como un 10% del rango total de entrada al modulador, esto sería 20 truncando el decimal.

6.2. Simulación hardware

6.2.1. Testbench

La generación de señales de prueba para el controlador en hardware debe considerar la salida del sistema en lazo cerrado, para incorporar el efecto de la señal de entrada al controlador sobre el sistema (esto ya que no se co-simula el modelo del sistema). Luego, el método para esto consiste en extraer un archivo de texto de la simulación en software con las salidas del sistema (I_0) a diferentes perturbaciones y otro con las salidas del controlador (entradas al modulador).

Se crea un módulo de lectura de archivo de texto plano (.txt) que además los entrega, según una frecuencia adecuada, al sistema de control implementado. La salida del sistema se comparará con el archivo generado de software igualmente, para finalmente evaluar el desempeño de la implementación en hardware y su similitud a las respuestas entregadas por software.

6.2.2. Entrega de datos

Los datos de salida del sistema en hardware corresponden a valores entre 0 y 255 que activarán el modulador controlado por voltaje. En la Figura 6.1 se muestran las señales de control para cada brazo del interferómetro, considerando un paso igual a 20 a sumar o restar en cada iteración.

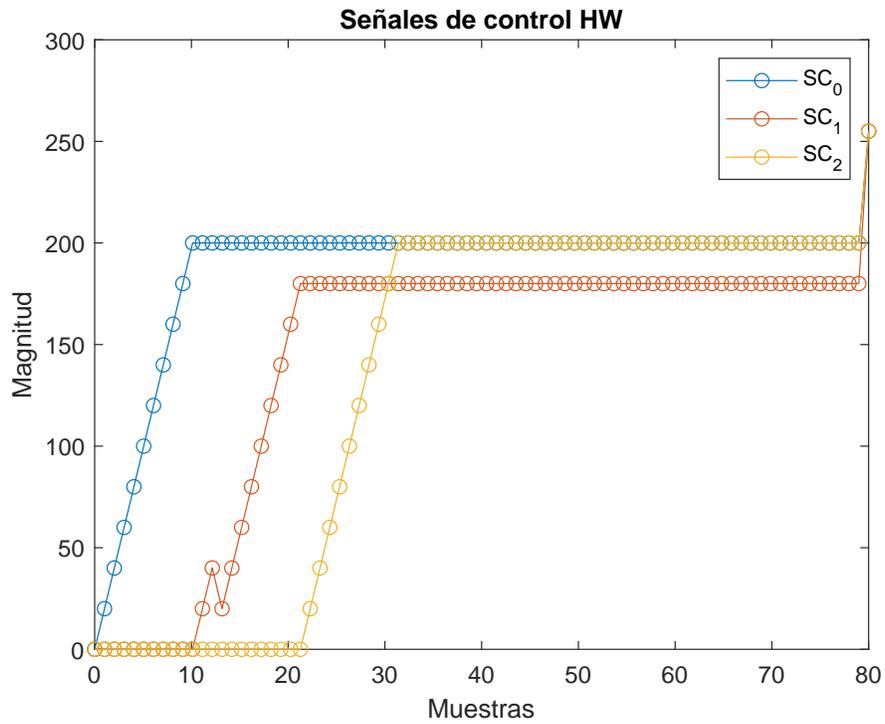
Se observa la transición del algoritmo de un brazo a otro como es requerido, cuando el contador llega a 10 o se alcanza un *peak* en la intensidad de entrada. Otro punto a destacar es el hecho de que se ven bajas en la magnitud en ciertos momentos, esto se debe a que la intensidad baja y para intentar cambiar su dirección la señal de entrada al modulador igual debe bajar. Por último, el salto de los tres brazos que se observa al final corresponde al módulo *mueve* del algoritmo, el cual se explica en la Sección 5.2.3.

6.2.3. Uso de recursos

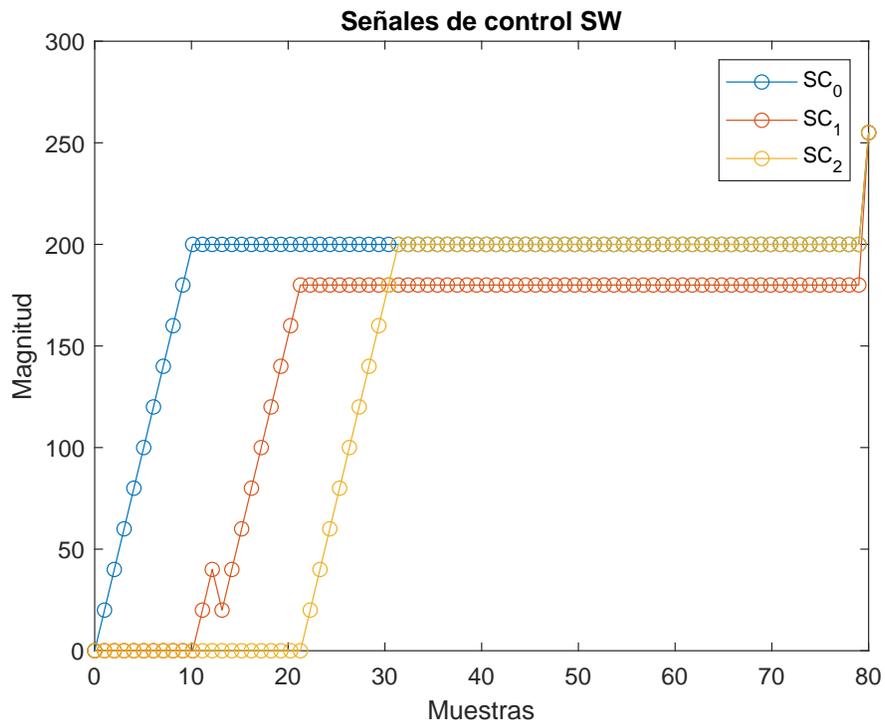
Como esta aplicación es secuencial, los recursos predominantes serán LUTs y FFs, esto se demuestra claramente en la tabla y gráfica de recursos. Por otro lado, los tres DSP utilizados corresponden a los multiplicadores del módulo *mover* que multiplica el paso por la cantidad señalada por el contador.

Tabla 6.1: Uso de recursos en FPGA.

Recurso	Estimación	Disponible	Utilización (%)
LUT	258	53200	0.48
LUTRAM	1	17400	0.01
FF	144	106400	0.14
DSP	3	220	1.36
BUFG	1	32	3.13



(a) Señales de control resultantes de simulación post-síntesis.



(b) Señales de control resultantes de simulación en hardware.

Figura 6.1: Resultados de simulación.

Capítulo 7

Conclusiones y trabajo futuro

En este trabajo se presentó una simulación del sistema de control para disminuir el ruido de fase de un interferómetro Mach-Zehnder de cuatro brazos, utilizando una versión adaptada del algoritmo de optimización multiparámetro de Powell. El sistema de control está constituido por tres lazos, donde cada uno actúa sobre un brazo del interferómetro mediante un modulador de fase controlado por voltaje. Dichos actuadores siguen el algoritmo establecido y comandado por la implementación presente en la tarjeta de desarrollo Zybo Z7-20.

De los métodos de optimización multiparámetro sin derivadas estudiados, se seleccionó el algoritmo de Powell como controlador de esta configuración. Luego de estudiar el funcionamiento de dicho algoritmo para la particularidad de la planta se decidió generar una adaptación de él, debido a la naturaleza de las ecuaciones que regían el modelo matemático del interferómetro Mach-Zehnder de cuatro brazos. Con este algoritmo se realizaron pruebas en el sistema simulado en el software Matlab llegando a resultados positivos donde se logró eliminar en gran medida el ruido de fase del sistema.

Para pasar a las pruebas de implementación del lazo de control se traspasó el algoritmo a lenguaje de descripción de hardware *Verilog*, con datos de entrada de 12 bits y datos de salida de 8 bits. Se utilizó una arquitectura compuesta por tres módulos principales. Finalmente, esta experiencia concluye con la comparación de la simulación post-síntesis y la simulación del controlador en software, la cual arroja resultados exitosos y se presume un

buen funcionamiento al momento de una posterior implementación en la planta real.

La continuación inmediata de este trabajo pasa directamente por realizar implementación del controlador en el sistema real, lo cual, por razones de tiempo, no pudo ser incorporado en este trabajo.

Por otro lado, se podría estudiar el uso de otro tipo de mecanismo de optimización, como podría ser la inclusión de algoritmos de *machine learning* para recalcular constantemente el punto de máxima intensidad de salida.

Glosario

- ADC: analog to digital converter.
- BS: beam-splitter.
- DAC: digital to analog converter.
- DEMUX: demultiplexor.
- FBS: fiber splitter.
- FPGA: field programmable gate array.
- HPC: high-pin count.
- M: mirror.
- MCBS: multi-core beam splitter.
- MMF: multiple mode fiber.
- MOD: Modulador de fase.
- MUX: multiplexor.
- MZI: Interferómetro Mach-Zehnder.
- PE: piezo-eléctrico.
- QKD: Quantum key distribution.
- SMF: single mode fiber

Bibliografía

- [1] Digilent. «Zybo Z7 Reference Manual.» (), dirección: <https://digilent.com/reference/programmable-logic/zybo-z7/reference-manual?redirect=1>.
- [2] ixblue, *Electro-Optical Modulators Operating Manual*, <https://www.ixblue.com/wp-content/uploads/2022/05/eom-operating-manualed050.pdf>, Retrieved October, 2022.
- [3] G. Tonel, *Unconstrained optimization using Powell*, <https://www.mathworks.com/matlabcentral/fileexchange/15072-unconstrained-optimization-using-powell>, MATLAB Central File Exchange, Retrieved September 30, 2022.
- [4] M. . Born y E. . Wolf, *Principles of Optics: 60th Anniversary Edition*, inglés americano, 7.^a ed. Cambridge University Press, 2020.
- [5] J. Cariñe, G. Cañas, P. Skrzypczyk et al., «Multi-core fiber integrated multi-port beam splitters for quantum information processing,» *Optica*, vol. 7, n.º 5, págs. 542-550, mayo de 2020. DOI: 10.1364/OPTICA.388912. dirección: <http://opg.optica.org/optica/abstract.cfm?URI=optica-7-5-542>.
- [6] Y. Xu, J. Lin, Y.-H. Li, H. Dai, S.-K. Liao y C.-Z. Peng, «Active Phase Stabilization for the Interferometer With 128 Actively Selectable Paths,» *IEEE Transactions on Nuclear Science*, vol. 66, n.º 7, págs. 1076-1080, 2019. DOI: 10.1109/TNS.2019.2921982.
- [7] L. Mach, L. Zehnder y C. Clark, «The Interferometers of Zehnder and Mach,» en feb. de 2017.
- [8] J. A. C. Catrileo, «Aplicación de electrónica embebida de bajo costo en el procesamiento de información cuántica,» Tesis doct., Universidad de Concepción, 2016.
- [9] *12-/10-/8-Bit ADCs in 6-Lead SC70*, AD7476A, Analog Devices, ene. de 2011.

- [10] G. B. Xavier y J. P. von der Weid, «Stable single-photon interference in a 1 km fiber-optic Mach–Zehnder interferometer with continuous phase adjustment,» *Optics Letters*, vol. 36, n.º 10, pág. 1764, 2011, ISSN: 1539-4794. DOI: 10 . 1364 / o l . 36 . 001764. dirección: <http://dx.doi.org/10.1364/OL.36.001764>.
- [11] *12-Bit MicroPower,RRO Digital to- Analog Converter*, DAC121S101, Texas Instrument, feb. de 2010.
- [12] L. Ruiz, J. Beristáin, I. Sosa y J. Hernández, «Estudio del algoritmo de seguimiento de punto de máxima potencia perturbar y observar,» *Revista de ingeniería eléctrica, electrónica y computación*, vol. 8, n.º 1, págs. 17-23, 2010.
- [13] S. . S. . Rao, *Engineering Optimization: Theory and Practice*, inglés americano, 4.^a ed. Wiley, 2009.
- [14] W. Press, S. Teukolsky, W. Vetterling y B. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 2007, ISBN: 9780521880688. dirección: <https://books.google.com.mx/books?id=1aA0dzK3FegC>.
- [15] Press, William H., *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, inglés americano, 3.^a ed. Cambridge University Press, 2007.
- [16] K.-P. Ho, «Phase-Modulated Optical Communications Systems, Springer,» en feb. de 205.

**UNIVERSIDAD DE CONCEPCION – FACULTAD DE INGENIERIA
RESUMEN DE MEMORIA DE TITULO**

Departamento : Departamento de Ingeniería eléctrica
Carrera : Ingeniería Civil electrónica
Nombre del memorista : Mónica Eloísa Venegas Rosales
Título de la memoria : Control de interferómetro en fibra óptica multi-núcleo utilizando FPGA
Fecha de la presentación oral : 23 de Enero de 2023

Profesor(es) Guía : Miguel Figueroa Toro y Gabriel Saavedra Mondaca
Profesor(es) Revisor(es) : Mario Medina Carrasco
Concepto :
Calificación :

Resumen (máximo 200 palabras)

Las aplicaciones del interferómetro llegan a campos como la mecánica, biología, astronomía y comunicaciones, estos dispositivos son altamente sensibles a las alteraciones físicas del medio, las cuales generan el poco deseado ruido de fase. En el presente trabajo se aborda el desarrollo y validación de un algoritmo de control para un interferómetro Mach-Zehnder de cuatro brazos, el que se basa en fibra óptica multi-núcleo y está emplazado en la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Concepción. El algoritmo de control busca estabilizar las fluctuaciones en la intensidad de salida requeridas por el sistema de una manera rápida, sencilla y eficiente.

Este trabajo se comenzó con el estudio de la planta a controlar y la toma de muestras de los parámetros relevantes para comprender su funcionamiento, para luego pasar a la revisión de algoritmos de optimización multiparámetro y su debido uso en el abordaje de la problemática a solucionar.

Una vez adaptado el algoritmo de Powell para esta implementación, se probó su funcionamiento al momento de controlar el sistema mediante simulación en software y en hardware, para luego contrastar dichos resultados. Esto mostró una similitud relevante entre ambos procedimientos, logrando validar el funcionamiento del algoritmo presentado en este trabajo.

Finalmente, se presenta el uso de recursos de la implementación en hardware de este controlador, simulado sobre una tarjeta de desarrollo Zybo Z7-20, la cual contiene un FPGA Zynq™-7000 de Xilinx.