

**Título del Proyecto de Investigación
al que corresponde el Reporte Técnico:**

Diseño de un sistema embebido en FPGA para su aplicación en sensores electro-ópticos.

Tipo de financiamiento

Sin financiamiento

Fecha de Inicio: 20/11/2021

Fecha de Término: 30/07/2023

Tipo de Reporte

Parcial

Final

Autor (es) del reporte técnico:

Abimael Jiménez Pérez
Ángel Saucedá Carvajal
José Mireles Jr. García
J. Antonio Muñoz Gómez
Rafael E. Gonzalez Landaeta

Resumen del reporte técnico en español (máximo 250 palabras)

La demodulación homodina es una técnica ampliamente utilizada en la detección de señales de sensores interferométricos. El proceso de demodulación se desarrolla normalmente utilizando circuitos analógicos. Sin embargo, para mejorar el rendimiento del demodulador, se debe emplear un sistema digital. En este proyecto, desarrollamos un sensor de fibra óptica combinando: (a) un interferómetro Michelson, (b) un dispositivo MEMS (sistema micro-electro-mecánico) y (c) un interrogador basado en una matriz de compuertas programable en campo (FPGA). El procesamiento de señales se implementó en un sistema embebido basado en FPGA. El algoritmo de demodulación homodina se implementó con módulos de hardware desarrollados en el lenguaje de descripción de hardware (HDL) para proporcionar un sistema digital portátil, de bajo costo y escalable. El presente estudio demuestra con éxito el desarrollo y validación de un interrogador basado en FPGA capaz de procesar interferogramas a través de un esquema de demodulación homodina. Los resultados experimentales demuestran la medición correcta del desplazamiento de la masa de prueba de un MEMS y los pocos recursos de hardware utilizados. Las mediciones de desplazamiento coincidieron con las obtenidas con un sistema de caracterización certificado. Dado que el sistema se puede reconfigurar fácilmente, se puede desarrollar una metodología de medición similar utilizando otros esquemas de demodulación y sensores de fibra óptica.

Resumen del reporte técnico en inglés (máximo 250 palabras):

Homodyne demodulation is a convenient technique for signal detection in interferometric sensors. The demodulation process is typically developed using analog circuits. However, to improve the performance of the demodulator, a digital system must be employed. In this project, we developed an optical fiber sensor by combining: (a) a Michelson interferometer, (b) a micro-electro-mechanical system (MEMS) device, and (c) a field-programmable gate array (FPGA)-based interrogator. Signal processing was integrated into the FPGA-embedded system. The homodyne demodulation algorithm was implemented with hardware modules developed in the hardware description language

(HDL) to provide a portable, low-cost, and scalable digital system. The present study successfully demonstrates the development and validation of an FPGA-based interrogator capable of processing interferograms through a homodyne demodulation scheme. The experimental results reveal proper displacement measurements of the proof-mass MEMS and the low amount of hardware resources used. The displacement measurements obtained from the system matched those obtained from a certified characterization system. As the system can be easily reconfigured to the required measured signal, a similar measurement methodology can be developed using other demodulation schemes and optical fiber sensors.

Palabras clave:

Demodulación homodina; sensor de fibra óptica; interferómetro Michelson; sistema embebido; sistema micro-electro-mecánico (MEMS); arreglo de compuertas programables de campo (FPGA)

Usuarios potenciales (del proyecto de investigación)

Industria regional y nacional que requieran un sistema de medición de desplazamiento o análisis de vibración.

Reconocimientos

Agradecimientos al Consejo Nacional de Humanidades, Ciencias y Tecnologías (Conahcyt) a través del apoyo del sistema nacional de investigadores, los colaboradores y estudiantes que apoyaron en la realización de este proyecto.

1. Introducción

La unión de las técnicas de detección óptica y el procesamiento digital de señales ha generado sistemas de medición en la industria automotriz, biomédica y de semiconductores. Los sensores interferométricos ofrecen una alta sensibilidad y una resolución fina (tanto espacial como temporal). Por otro lado, los dispositivos MEMS (sistemas micro-electro-mecánicos) tienen aplicaciones en la medición de diferentes magnitudes físicas (desplazamiento, vibración, presión, temperatura, etc.) y, al utilizarlos con sistemas ópticos (MOEMS) se genera una alternativa de medición innovadora, no destructiva e inmune a la interferencia electromagnética. Sin embargo, la señal de interés aparece modulada, como una señal radio de FM, lo que implica un proceso de demodulación para obtener la señal de interés o mesurando. Esto implica el uso de hardware y software especial como tarjetas de adquisición de datos, equipo de cómputo y software con licencia. En este proyecto se propone la creación de un sistema embebido

basado en un FPGA capaz de sustituir dichos componentes, creando un sistema reconfigurable, portátil y escalable para diferentes algoritmos de demodulación y/o sensores MOEMS con potenciales aplicaciones de medición en la industria.

2. Planteamiento

Existen distintos sistemas eletro-ópticos basados en dispositivos MEMS que realizan la medición de magnitudes físicas. Las señales obtenidas de estos sistemas deben ser procesadas para obtener las mediciones y/o parámetros necesarios. El cuerpo académico de microelectrónica está desarrollando sistemas eletro-ópticos basados en dispositivos MEMS. En 2015 Carlos Ramos desarrolló una tesis de maestría en la UACJ donde se diseñó un sistema digital para el procesamiento de interferogramas [1]. El procesamiento se implementó mediante soft-core en el procesador Nios II, embebido en el FPGA Quatus II de Altera. El procesamiento de los interferogramas se realizó satisfactoriamente; sin embargo, el tiempo de procesamiento fue muy lento, haciendo no viable su implementación en una aplicación real.

3. Objetivos (general y específicos)

Diseñar un sistema embebido portable en lenguaje de descripción de hardware que integre diferentes periféricos con la finalidad de que sea implementado en el procesamiento de interferogramas.

Objetivos específicos

- Diseñar una arquitectura portable de un sistema embebido en un FPGA con periféricos accesibles mediante una arquitectura de memoria mapeada con la finalidad de agregar varios periféricos.
- Diseñar los módulos aceleradores de hardware necesarios para cada una de las etapas del procesamiento de interferogramas.
- Integrar los módulos en la arquitectura del sistema embebido para que sean utilizados como periféricos.
- Desarrollar el software (drivers y aplicación) de cada módulo para que tengan interacción con el hardware del sistema embebido.
- Evaluar el desempeño del sistema embebido mediante simulación y su implementación en un FPGA.

4. Antecedentes

En los últimos años, varios trabajos han sido un desafío para realizar la medición de desplazamiento y vibración utilizando interferómetros ópticos [1-4]. El principio básico

de un sensor interferométrico de desplazamiento/vibración es que un estímulo aplicado provoca un cambio de fase entre dos haces de luz, que se combinan y se convierten en una señal eléctrica. Esta forma de onda tiene una relación no lineal con el cambio de fase; por lo tanto, es necesario aplicar técnicas de procesamiento de señales relativamente complejas para obtener una señal de salida que sea proporcional al estímulo. Las técnicas comunes incluyen pseudoheterodino [5], homodino pasivo y activo [6] y heterodino sintético [7]. Esta última técnica tiene la capacidad de medir desplazamientos con alta resolución porque tiene una alta inmunidad al ruido [8, 2].

La implementación digital de los algoritmos de demodulación requiere varias operaciones con un alto consumo de recursos de hardware. Afortunadamente, los dispositivos FPGA contienen bloques lógicos masivos, un gran almacenamiento de memoria de acceso aleatorio (RAM) y unidades de procesamiento digital de señales (DSP), proporcionan una plataforma potencialmente poderosa para la implementación de estos esquemas [9].

Con el fin de validar las principales ventajas del sistema embebido propuesto frente a otros sistemas basados en FPGA, en la Tabla 1 se muestra una comparación de las principales características de cada diseño como tipo de interferómetro, dispositivo FPGA, recursos de hardware, entre otros. En [10] se presenta un sistema de medidas de desplazamiento basado en un microcontrolador y un FPGA. No se reporta el uso de núcleos de propiedad intelectual (IP), LUT (LookUp Table) o módulos DSP. Dado que el procesamiento se realiza mediante dispositivos separados, la velocidad de procesamiento se ve comprometida. Los sistemas propuestos en [11,12] se basan en módulos IP. El inconveniente de los módulos IP de los proveedores de FPGA es que generalmente los adaptan a sus propias plataformas propietarias. No son portátiles y con frecuencia se entregan como cajas negras. Por lo tanto, un sistema debe rediseñarse si se implementa en un FPGA de un fabricante diferente. El diseño de hardware de [13] se realizó mediante lenguaje de descripción de hardware (HDL), pero no se reporta la arquitectura de hardware. Los diseños de hardware de [4,14,15] se construyeron en el sistema embebido comercial Red Pitaya STEMLab 125-14, basado en un FPGA Xilinx Zynq 7010. Las desventajas son que los diseños están limitados a las capacidades de la arquitectura del sistema y el código no es portable.

Tabla 1. Comparación de diferentes sistemas basados en FPGA para el procesamiento de interferogramas

Referencia	Tipo de interferómetro	FPGA	LUT/DSP	ADC
[10]	Heterodino	Intel EPM7128SLC84-15	* / *	10 bits
[11]	Heterodino	Spartan 3E XC3S1600E	* / *	14 bits

[12]	*	Stratix III EP3SE110F1152I3	8867 (11%) / 47 (6%)	*
[13]	Laser feedback	Virtex6 XC6VLX75T	5154 (7%) / 3 (1%)	*
[14]	Homodino	Zynq7010 (SoC Red Pitaya)	* / *	14 bits
[15]	*	Zynq7010 (SoC Red Pitaya)	9289 (53 %) / *	14 bits
[4]	Homodino	Zynq7010 (SoC Red Pitaya)	* / *	14 bits

* No reportado

5. Marco teórico

5.1. Microprocesador

Un microprocesador es un circuito integrado que contiene una unidad de procesamiento central (CPU), y una unidad de memoria. Se dice que un microprocesador es de sistema abierto ya que puede ser utilizado para diversos propósitos, pero necesita ser conectado a diferentes periféricos de entrada y salida para tener un correcto funcionamiento [18, 19, 20].

5.2. Tipos de Microprocesadores

Un microprocesador tipo CISC (complex instruction set computer) es un tipo de procesador con gran cantidad de instrucciones. Algunas de éstas muy sofisticadas y poderosas, pero con la desventaja de tener un mayor tamaño y tiempo de ejecución. Por otra parte, un microprocesador tipo RISC (reduced instruction set computer) es un tipo de procesador con un repertorio de instrucciones reducido y simple, lo que lo hace más flexible al sólo contener las instrucciones más indispensables y suelen ejecutarse en un ciclo de reloj [21].

5.3. Arquitectura Harvard

Esta arquitectura dispone de dos memorias independientes, una memoria RAM de datos y una memoria de programa (flash), las cuales se conectan mediante dos buses separados. Estas son independientes y pueden ser de diferente tamaño, permitiendo que el CPU pueda acceder de forma simultánea e independiente a la memoria de datos o del programa, reduciendo el tiempo de ejecución [19, 20, 22]. Las principales características de esta arquitectura son:

- El tamaño de las instrucciones no está relacionado con el de los datos, pudiendo ser optimizado para cualquier instrucción, ocupando una sola posición en memoria.

- El tiempo de acceso se reduce gracias a que el acceso a los datos e instrucciones puede ser simultáneo.

5.4. Sistema embebido

Un sistema embebido se define como cualquier sistema computarizado que realice ciertas tareas específicas [23, 24]. Los sistemas embebidos pueden ser implementados en distintos dispositivos como un microcontrolador o un FPGA. En general los sistemas embebidos se implementan mediante hardware ya definido, utilizando como unidad de procesamiento un microcontrolador. Aunque este tipo de sistemas embebidos sea el más común por la gran velocidad de operación y su fácil programación, limita las posibilidades de agregar nuevo hardware al sistema. Debido que una modificación del hardware puede requerir de un nuevo microcontrolador que controle o contenga el nuevo hardware. Una buena alternativa para evitar este tipo de problemas en un sistema embebido que requiere cambios recurrentes en hardware es el uso de una arquitectura FPGA [22].

5.5. Funcionamiento de un FPGA.

Un FPGA es un dispositivo lógico programable, el cual soporta la implementación de sistemas lógicos relativamente grandes. Éste está compuesto de bloques lógicos y bloques de Entrada / Salida (E/S). Cada bloque lógico en un FPGA funciona como un pequeño sistema con un número de E/S limitado. Aunque los bloques de un FPGA no sean los mismos, el más utilizado es la LUT, el cual contiene celdas de almacenamiento que sirven para implementar una función lógica pequeña [23, 24, 25, 26]. Para comprender el funcionamiento de los FPGAs, es necesario hablar de los recursos básicos de estos dispositivos:

- Bloques Lógicos Configurables (CLB). Los CLB contienen los recursos necesarios para implementar la lógica combinatorial y registros. Los CLBs son creados a partir de rebanadas que contienen un par de generadores de funciones de 4 entradas y dos elementos de almacenamiento (flip flop D).
- Bloques de Entrada/Salida (IOB): los IOB pueden ser configurados para ajustarse a un amplio rango de estándares de señales.
- Programmable Routing: en conjunto con los CLB existen las Matrices de Ruteo General (GRM). Los GRM son un tipo de matriz de interruptores que se utilizan para interconectar distintos CLBs.

5.6. Lenguaje de Descripción de Hardware

Para programar un FPGA se utilizan lenguajes HDL. Un lenguaje de descripción de hardware es un lenguaje de computadora diseñado para realizar la descripción del hardware de un sistema. Es decir, un sistema descrito mediante una descripción HDL caracteriza el funcionamiento, comportamiento y/o estructura [24]. Un lenguaje HDL es parecido a un lenguaje orientado a software (como C, C++, java, etc.) pero con la diferencia de que mientras uno describe un comportamiento interno (software) y el otro lo describe un comportamiento externo (hardware). Además, los HDL tienen la necesidad de describir la concurrencia entre los distintos componentes de hardware. Es importante mencionar que los HDL incluyen notaciones explícitas para expresiones de tiempo y concurrencias [24, 27, 28].

5.7. VHDL

Existen, principalmente, dos lenguajes de descripción de hardware (HDL) Verilog y VHDL. Aunque ninguno es superior, el lenguaje de nuestro interés es el VHDL. VHDL es un acrónimo que viene de VHSIC (very high speed integrated circuits) y de HDL [25]. Principalmente, el lenguaje VHDL tiene algunas características clave para diferenciarlo de un lenguaje de software. En primer lugar, VHDL necesita describir la concurrencia en los diferentes bloques de Hardware, los cuales son descritos mediante procesos. En segundo lugar, las conexiones entre bloques se realizan mediante señales que representan una conexión física. Y para concluir, VHDL necesita modular tiempo de manera que sepa qué componente deberá funcionar primero [27, 29].

5.8. MicroBlaze

Es un microprocesador modificable de 32/64 bits basado en un procesador tipo RISC y una arquitectura Harvard. Microblaze está especializado para sistemas embebidos gracias a su amplio set de instrucciones optimizadas para este tipo de sistemas embebidos. Debido a que MicroBlaze se implementa mediante soft-core puede ser configurado de distintos modos [30, 31, 32]. Es decir, el microprocesador es descrito mediante un código HDL y es implementado en un FPGA dando como resultado un procesador configurable [33].

5.9 Interferómetro.

Un interferómetro consiste en una fuente de luz que emite haces de luz que se propagan en algún medio (espacio libre, fibra óptica o cualquier otro medio que sirva de guía para las ondas luminosas) dependiendo de la aplicación. Al pasar el haz de luz por un divisor de haz, éste se separa en dos nuevos haces que toman diferentes trayectorias antes de volver a unirse.

En la figura 1 se observa que uno de los haces recorre una trayectoria hasta ser reflejado en una superficie de referencia. El camino recorrido por este haz se considera conocido. El otro haz, llamado haz de prueba, recorre una trayectoria distinta y es reflejado en otra superficie que puede estar en movimiento o en la que cambia alguna de sus propiedades físicas, como temperatura, índice de refracción, espesor, etc.

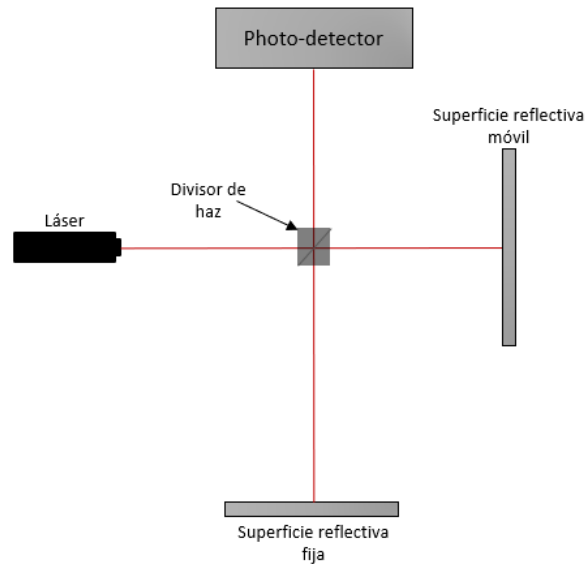


Figura 1. Arreglo esquemático de un interferómetro tipo Michelson.

Los cambios en la trayectoria del haz de prueba provocan un cambio en la fase relativa entre los dos haces, debido a la diferencia de camino óptico. Una vez reflejados ambos haces, estos vuelven a atravesar el divisor de haz y se combinan dando lugar al fenómeno de interferencia. Posteriormente son detectados por un fotodetector y convertidos en una imagen digital o en el caso de un interferómetro de fibra óptica, en una señal eléctrica dependiente del tiempo (t) [34].

Los cambios en la trayectoria del haz de prueba provocan un cambio en la fase relativa entre los dos haces, debido a la diferencia de camino óptico. Una vez reflejados ambos haces, estos vuelven a atravesar el divisor de haz y se combinan dando lugar al fenómeno de interferencia. Posteriormente son detectados por un fotodetector y convertidos en una imagen digital o en el caso de un interferómetro de fibra óptica, en una señal eléctrica dependiente del tiempo (t) [34].

La señal eléctrica obtenida de un interferómetro de fibra óptica tipo Michelson está descrita por la ecuación

$$I(t) = A + V \cos\left(\frac{4\pi}{\lambda} OPD(t)\right) \quad (1)$$

donde A y V son parámetros que dependen de valores del interferómetro como la intensidad de la fuente y, los coeficientes de acoplamiento y reflexión de la fibra óptica, entre otros. λ es la longitud de onda de la fuente de luz y la función $OPD(t)$ (optical path difference), también llamada fase óptica da cuenta de la diferencia de camino óptico inducida por la variación del mesurando. En ésta se encuentra la información de interés y el procedimiento para extraerla, recibe el nombre de demodulación. Se reconoce que la información de interés se encuentra dentro del argumento de un coseno y; por lo tanto, al cambiar su valor, cambia la frecuencia del coseno; como en los sistemas de comunicación de frecuencia modulada.

5.10 Demodulación

Aunque existen varios esquemas de demodulación o extracción de $OPD(t)$, en el caso de los sensores interferométricos, uno de los algoritmos más utilizados es el que se basa en la detección homodina. Éste consiste en cambiar la frecuencia de uno de los haces de luz, introduciendo una modulación de frecuencia conocida, proporcional a $C \cos \omega_0 t$. Este término equivale a la portadora de un sistema de comunicaciones de FM. Por lo tanto, la información es obtenida mediante una demodulación de FM [35]. En este caso, la foto-intensidad detectada de la ecuación 1, puede ser expresada por:

$$I(t) = A + V \cos(C \cos(\omega_0 t) + OPD(t)) \quad (2)$$

donde C representa la amplitud máxima de la fase arbitraria y ω_0 es la frecuencia de esta modulación.

Existen varias formas de introducir experimentalmente la fase arbitraria $C \cos(\omega_0 t)$. Una de ellas es modulando la frecuencia del láser, lo cual sería una técnica sintético-homodina [34] o mediante un estirador de fibra óptica. Cualquiera que sea el caso, la ecuación 3 puede ser reescrita como:

$$I(t) = A + V \cos(C \cos(\omega_0 t) + \varphi(t)) \quad (3)$$

donde $\varphi(t) = OPD(t) +$ variaciones indeseables en la fase de interés, representa no solo la fase óptica debida al mesurando, sino también a contribuciones indeseables por perturbaciones ambientales como temperatura, presión, vibraciones mecánicas, etc.

El algoritmo para recuperar la señal de interés $\varphi(t)$ se representa en la Figura 2. Básicamente consiste en mezclar el interferograma obtenido experimentalmente con dos osciladores locales a frecuencias angulares ω_0 y $2\omega_0$. Posteriormente, se eliminan mediante filtraje las armónicas por encima de la máxima frecuencia de interés, obteniéndose:

$$S_1(t) = AVJ_1(C) \sin(\varphi(t)) \quad (4)$$

$$S_2(t) = AVJ_2(C) \cos(\varphi(t)) \quad (5)$$

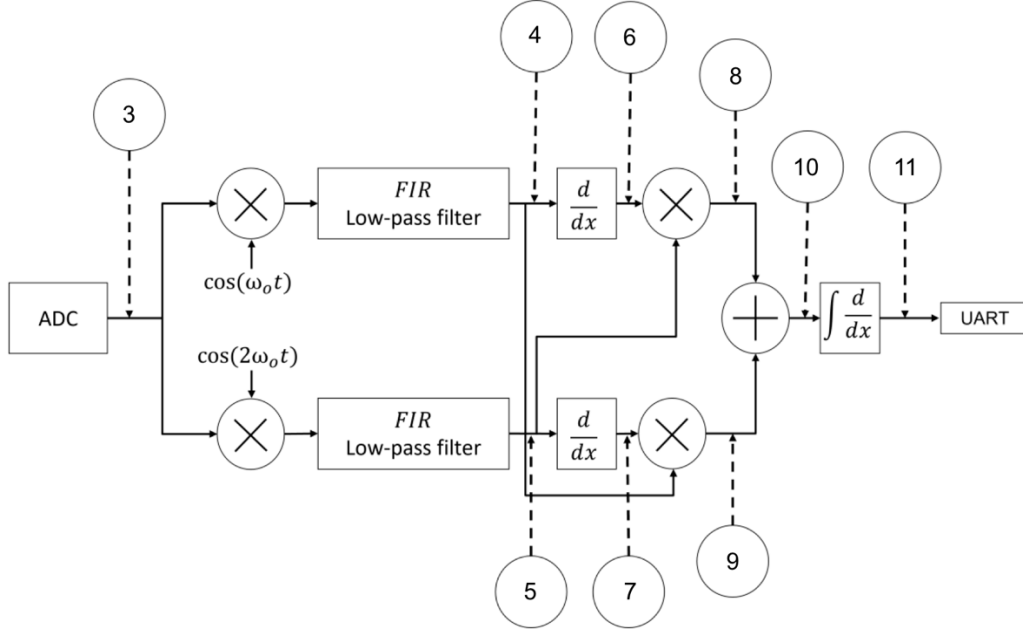


Figura 2. Algoritmo de demodulación homodino para recuperar la señal de interés $\varphi(t)$.

para las trayectorias superior e inferior, respectivamente (ver figura 2). $J_1(C)$ y $J_2(C)$ representan los valores de las funciones de Bessel evaluadas en C de primera clase de orden 1 y 2, respectivamente. El valor de C se escoge de tal manera que $J_1(C) = J_2(C)$; por lo tanto, el primer valor de C que satisface esta condición es $C \cong 2.37 \text{ rad}$. Posteriormente ambas señales $S_1(t)$ y $S_2(t)$, se derivan respecto a t , obteniendo:

$$S_3(t) = AVJ_1(C)\dot{\varphi}(t) \cos(\varphi(t)) \quad (6)$$

$$S_4(t) = -AVJ_2(C)\dot{\varphi}(t) \sin(\varphi(t)) \quad (7)$$

donde $\dot{\varphi}(t) = \frac{d\varphi}{dt}$. Al multiplicar $S_1(t)$ por $S_4(t)$ y $S_2(t)$ por $S_3(t)$ se obtiene:

$$S_5(t) = -(AV)^2 J_1(C)J_2(C)\dot{\varphi}(t) \sin^2(\varphi(t)) \quad (8)$$

$$S_6(t) = (AV)^2 J_1(C)J_2(C)\dot{\varphi}(t) \cos^2(\varphi(t)) \quad (9)$$

Por último, sustrayendo $S_5(t)$ de $S_6(t)$, tenemos:

$$S_0(t) = AV^2 J_1(C)J_2(C)\dot{\varphi}(t) \quad (10)$$

La ecuación 10 contiene la derivada de la señal de interés; por lo tanto, al integrar respecto a t se recupera la información de interés.

$$\int S_0(t) dt = \varphi(t) \quad (11)$$

6. Metodología

El sistema embebido propuesto utilizará el procesador soft-core Microblaze en modo de Microcontroller Preset, el cual al ser implementado en un FPGA Artix-7 funciona a 303 DMIPs (dhrystone millions of instructions per second) y trabaja a una frecuencia máxima de 226MHz. Una vez que se tiene el diseño del microcontrolador, se implementará un sistema que accederá a los periféricos mediante una arquitectura de memoria mapeada. Para agregar un módulo de hardware se tendrá que conectar a uno de los registros de función especial (RFE) libres para que funcione como vía de comunicación entre el microcontrolador y el módulo (ver Figura 3). El microcontrolador dejará valores específicos en un RFE, al cual accederá como si fuera a almacenar datos en RAM y el periférico leerá e interpretará los datos guardados en ese RFE.

Es importante mencionar que existe una limitante en el número de periféricos que se puede agregar al sistema. Por ejemplo, si en un sistema embebido se tiene solo 64 registros inutilizados, entonces solo se tendrá la posibilidad de implementar 64 periféricos (ver Figura 3a). Para evitarlo, se utilizará un sistema que encapsula un número específico de periféricos implementados (ver Figura 3b). De manera que, si el subsistema encapsula 64 periféricos e implementamos 64 sistemas de encapsulamiento tendremos un total de 4096 periféricos.

El sistema de encapsulamiento de periféricos está basado en la arquitectura propuesta en [23], el cual es un sistema que decodifica los datos de tres RFE y selecciona el módulo de hardware que se utilizará. El sistema de encapsulamiento consiste en un decodificador para habilitar y deshabilitar los distintos módulos, un multiplexor que dirige la salida del módulo seleccionado a la salida del sistema, registros FIFO (first input first output) para guardar los datos de entrada de los módulos, y los módulos encapsulados de hardware.

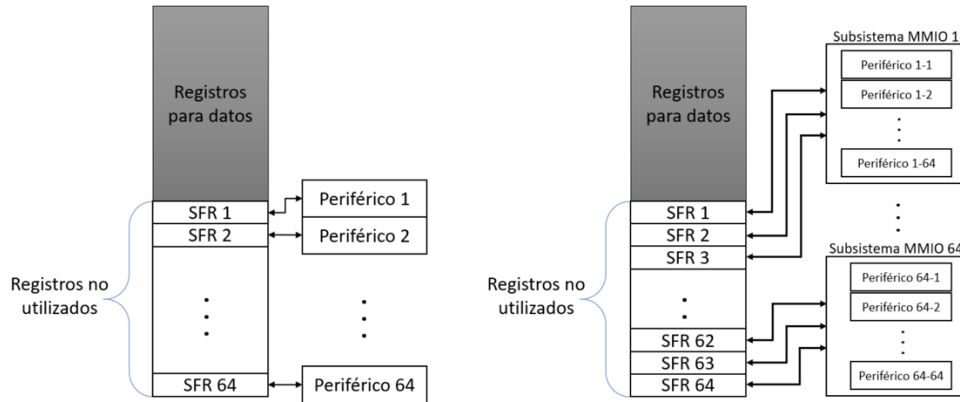


Figura 3. Arquitectura de sistema embebido a) Conexión de periféricos a SFR. b) Conexión de periféricos mediante un sistema de memoria mapeada.

Además, en este proyecto se utilizará el subsistema Vanilla propuesto en [23] (ver Figura 4), el cual utiliza los módulos básicos: Timer, UART (universal asynchronous receiver-transmitter), GPO (general purpose output), GPI (general purpose input), ADC (analog-digital converter) como sistema mínimo de funcionamiento.

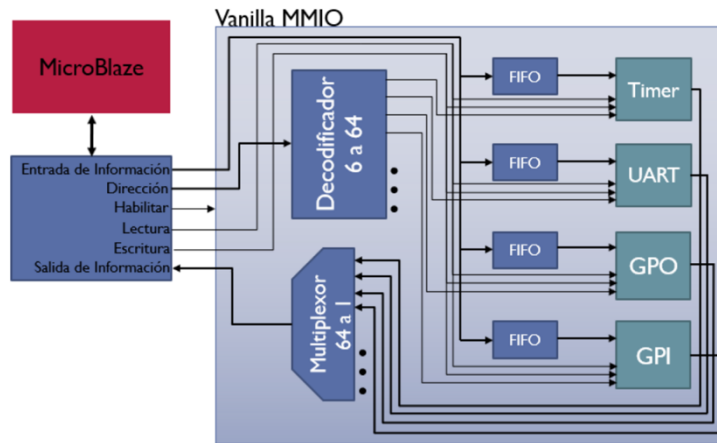


Figura 4. Arquitectura del subsistema Vanilla [23].

Al subsistema Vanilla se le agregarán los módulos de hardware necesarios para el procesamiento de interferogramas. Los módulos deben ser compatibles con los espacios del sistema de memoria mapeada. Para lograr esa compatibilidad es necesario que los bloques tengan las siguientes entradas y salidas:

- Una señal de 32 bits para leer los datos que entraran a un módulo de hardware.
- Una señal de 32 bits para escribir los datos que saldrán de un módulo de hardware.

- Una señal de 5 bits con la cual se activará el módulo de hardware.
- Un bit que habilitará la escritura.
- Un bit que habilitará la lectura.
- Un bit que habilitará el módulo.

Una vez que ya se ha agregado el módulo de hardware en el sistema, es necesario crear el driver que realizará la interfaz entre el hardware y el software. El driver de un módulo como mínimo debe declarar apuntadores con valores previamente definidos con la dirección del sistema. Aunque no sea totalmente necesario que existan drivers para interactuar con el hardware, permiten que el diseño de hardware y de software sea portable ya que el cambiar alguno de los dos solo implicará un cambio de drivers.

7. Instituciones, organismos o empresas de los sectores social, público o productivo participantes (Si aplica)

Universidad de Guadalajara – Centro Universitario de la Costa Sur (CUCSUR).

8. Resultados

En el presente estudio, se validó un sistema digital basado en FPGA con un esquema de demodulación homodina midiendo el movimiento de la masa de prueba de un MEMS, tomando como referencia la vibración generada con un sistema de caracterización certificado. El sistema consta de un controlador Kistler 5022, un acelerómetro de referencia 8076K y un shaker B&K 4809. El sistema embebido se probó introduciendo el interferograma de entrada con $A = V = 1$ $I(t) = 1 + \cos[2.6 \cos(2\pi 1000t) + \varphi(t)]$. Esta señal tiene una frecuencia portadora de $\omega_0 = 1000$ Hz y una amplitud de $C = 2.6$ rad. El shaker genera señales sinusoidales para excitar la masa de prueba. Por lo tanto, $\varphi(t) = (4\pi/\lambda)D \sin(\omega t) + \text{variaciones indeseables de la fase}$, donde D y ω representan la amplitud máxima y la frecuencia angular de las vibraciones del mensurando, respectivamente. Se utilizaron dos señales moduladas $\varphi(t)$ con una amplitud de 0.75 rad y frecuencias angulares de $\omega = 2\pi 200$ y $\omega = 2\pi 100$. Sin embargo, el sistema se probó en el rango de frecuencia de 50 a 500 Hz, que es un rango útil de este sensor MEMS de desplazamiento. La señal de salida del sistema se obtuvo utilizando el módulo receptor-transmisor asíncrono universal (UART). Este establece un canal de comunicación en serie con un ordenador a través de un puerto en serie, y su interfaz utiliza búfers FIFO.

La Figura 5 muestra la curva del oscilador $\cos \omega_0$ tal como se implementó en hardware, utilizando un módulo LUT de 512 palabras.

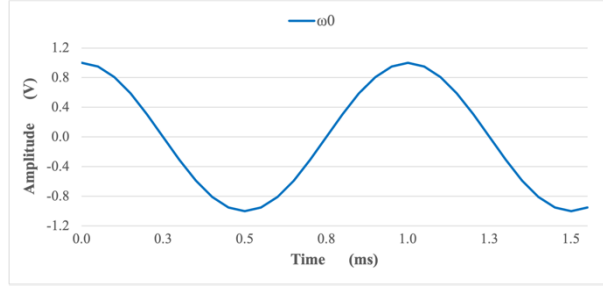


Figura 5. Valores de $\cos \omega_0$ guardados en la LUT para la implementación en VHDL de los módulos osciladores.

La Figura 6 muestra la respuesta FFT del filtro pasa-bajas tal como se implementa en el FPGA. El filtro dispone de los parámetros adecuados para eliminar los armónicos por encima de las frecuencias máximas de interés de $\varphi(t)$, $\omega = 2\pi 200$ y $\omega = 2\pi 100$.

La Fig. 7 muestra las señales de entrada y salida del sistema embebido para una señal modulada de 200 Hz. La Fig. 7a muestra el interferograma de entrada $I(t)$. Además, en la Figura 7b se muestra una comparación entre la señal de salida y la señal modulada $\varphi(t) = 0.75 \cos(2\pi 200t)$. Para cambiar las unidades de radianes a metros, ambas señales se escalan a $\frac{\lambda}{4\pi}$.

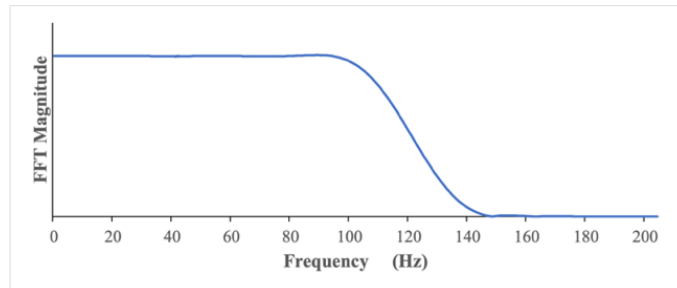


Figura 6. Respuesta FFT response del filtro FIR pasa-bajas diseñado en Matlab e implementado en el FPGA.

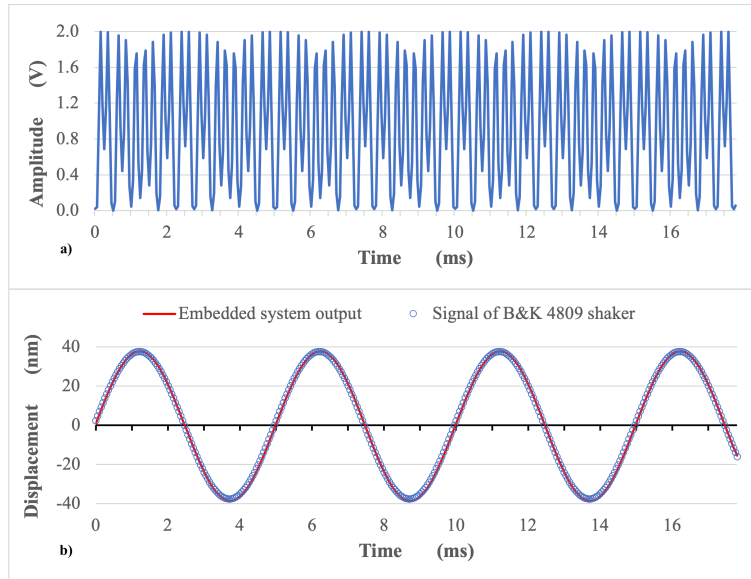


Figura 7. a) Interferograma de entrada para una señal modulada de 200 Hz. b) Comparación entre la señal modulada y la señal de salida del sistema embebido, escalado a $\frac{\lambda}{4\pi}$.

La Fig. 8 muestra una comparación entre la señal de salida del sistema embebido y la señal modulada $\varphi(t) = 0.75 \cos(2\pi 100t)$. La Fig. 9 muestra la FFT de las respuestas de desplazamiento correspondientes a las Figs. 7b y 8. Las Figs. 9a y 9b muestran la comparación del espectro entre la salida del sistema embebido y la señal del shaker cuando la masa de prueba se excita con señales sinusoidales de 200 y 100 Hz, respectivamente.

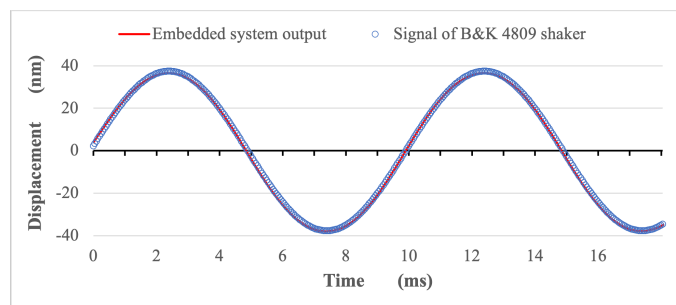


Figura 8. Comparación entre la señal modulada de $\omega = 2\pi 100$ y la señal de salida del sistema embebido, escalado a $\frac{\lambda}{4\pi}$.

La Fig. 5 muestra que los módulos de los osciladores con las formas de onda coseno, guardados en LUTs, se pueden reconfigurar fácilmente en el sistema; en lugar del cálculo de las funciones trigonométricas. El mismo método utilizado en la Figura 5 se utilizó en la implementación del oscilador $\cos 2\omega_0$.

La señal de interés se recuperó correctamente utilizando la técnica de demodulación homodina, como se muestra en las Figs. 7b y 8. Es decir, describieron la respuesta de la masa de prueba del MEMS a la señal senoidal generada por el shaker B&K 4809.

El orden del filtro de pasa-bajas fue $N = 141$. Sin embargo, para $N > 141$, el estimador del demodulador presenta un comportamiento de la señal que está en fase y los valores máximo y mínimo se estiman correctamente. En la Figura 9 se muestra que el sistema propuesto detecta las frecuencias de interés de 200, 100 Hz y sus armónicos.

En la Tabla 1 se presentan los resultados obtenidos tras la aplicación de la demodulación homodina a las señales de interferogramas adquiridas experimentalmente. El error cuadrático medio (RMS) se calculó comparando el desplazamiento de salida del sistema embebido con el desplazamiento sinusoidal de referencia generado por el shaker B&K 4809.

Nos gustaría destacar que el diseño e implementación de los módulos para el prototipo del sistema embebido consume muy pocos recursos del FPGA Artix-7: 6149/63400 (10%) LUTs, 32.5/135 (24%) bloques de RAM y 9/240 (4%) módulos DSP. Este bajo consumo de recursos de hardware da como resultado un sistema en FPGA eficiente y portátil para un demodulador homodino.

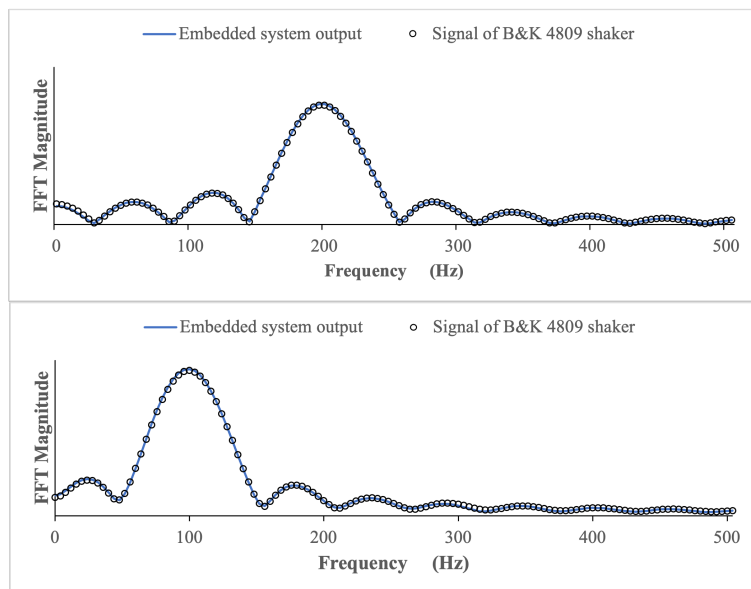


Figura 9. Comparación de FFT entre la señal de salida del sistema embebido y la señal del shaker, Figs. 7b y 8^a, respectivamente. Para a) $\omega = 2\pi 200$ y b) $\omega = 2\pi 100$.

Table 1. Error máximo y RMS para la técnica de demodulación homodina.

Amplitud de vibración (nm)	Frecuencia de entrada (Hz)	Error RMS (nm)	Error máximo (nm)
37.8	200	1.35	2.48
37.8	100	1.62	2.90

9. Productos generados

Artículos de investigación:

1. Jiménez, A.; Saucedo, Á.; Muñoz, A.; Duarte, J.; Mireles, J., Jr. FPGA-Based Hardware Implementation of Homodyne Demodulation for Optical Fiber Sensors. *Photonics* 2023, 10, 258. <https://doi.org/10.3390/photonics10030258>
2. Mireles, J., Jr.; Saucedo, Á.; Jiménez, A.; Ramos, M.; Gonzalez-Landaeta, R. Design and Development of a MOEMS Accelerometer Using SOI Technology. *Micromachines* 2023, 14, 231. <https://doi.org/10.3390/mi14010231>

Formación de recursos humanos:

Tesis de maestría:

1. Diseño de un sistema embebido basado en FPGA para el procesamiento de interferograma. Alumno: José Ángel Duarte Espino, matrícula: 217037.

Tesis de licenciatura:

1. Diseño en hardware de un filtro pasa-bajas para su integración en un sensor de fibra óptica. Alumno: Ricardo Trejo Castro, matrícula: 150392.

10. Conclusiones

Este trabajo demuestra con éxito el desarrollo y validación de un interrogador basado en FPGA capaz de procesar interferogramas a través de un esquema de demodulación homodina, cuyo objetivo principal es la interrogación de sensores de fibra óptica. Los módulos de hardware se desarrollaron desde cero en VHDL. Este método permite un bajo consumo de recursos del FPGA: en promedio, consumimos solo el 13% de los recursos en la plataforma de hardware. Además, el protocolo de bus y los controladores de dispositivos no estaban vinculados a ninguna plataforma comercial específica. Por lo tanto, el sistema embebido es portátil y el hardware se puede sintetizar para un diseño VLSI de un circuito integrado o reutilizarse en diferentes dispositivos FPGA y tarjetas de desarrollo.

11. Mecanismos de transferencia. (Si aplica)

12. Contribución e impacto del proyecto

La contribución de este proyecto tiene un impacto positivo en el desarrollo de tecnología en la industria de semiconductores para desarrollar sistemas de medición de desplazamiento, vibración o presión. Las ventajas del sistema propuesto son: inmunidad a interferencia electromagnética, técnica de medición sin contacto y alta resolución.

13. Impacto económico, social y/o ambiental en la región

El impacto económico se enfoca en el desarrollo de un sistema digital portátil de muy bajo costo para el procesamiento de interferogramas. Este sistema puede ser utilizado en la industria de la región para resolver problemas de medición de vibración o desplazamiento.

14. Referencias (bibliografía)

- [1] T. V. Kazieva, K. L. Gubskiy, A. P. Kuznetsov, and V. N. Reshetov, "3D push-pull heterodyne interferometer for SPM metrology," 2019, *Appl. Opt.*, 58, pp 4000 – 4006.
- [2] Thanh Dong Nguyen, Masato Higuchi, Thanh Tung Vu, Dong Wei, and Masato Aketagawa, "10-pm-order mechanical displacement measurements using heterodyne interferometry," *Appl. Opt.*, 2020, 59, pp 8478 - 8485.
- [3] L. Yu, G. Molnar, C. A. Werner, C. Weichert, R. Koenig, H.-U. Danzebrink, J. Tan, and J. Fluegge, "Single beam 3DoF homodyne interferometer," *Meas. Sci. Technol.*, 2020, 31, (8).
- [4] Straube, G., Fischer Calderón, J.S., Ortlepp, I. et al. "A Heterodyne Interferometer with Separated Beam Paths for High-Precision Displacement and Angular Measurements," *Nanomanuf Metrol*, 2021, 4, pp 200 - 207.
- [5] D. A. Jackson, A. D. Kersey, M. Corke, and J. D. Jones, "Pseudoheterodyne detection scheme for optical interferometers," *Electron. Lett.*, 1982, 18, (25-26), pp 1081 - 1083.
- [6] A. Dandridge, A. B. Tveten, and T. G. Giallorenzi, "Homodyne demodulation scheme for fiber optic sensors using phase generated carrier," *IEEE J. Quantum Electron.*, 1982, 18, (10), pp 1647 - 1653.
- [7] J. H. Cole, B. A. Danver and J. A. Bucaro, "Synthetic-Heterodyne Interferometric Demodulation," *IEEE Transactions on Microwave Theory and Techniques*, 1982, 30, (4), pp 540 - 543.
- [8] S. Rahm, C. Loth, A. Dabas, and S. Elouragini, "Improvement of heterodyne detection with optical amplifier and pulsed coherent Doppler Lidar," *J. Mod. Opt.*, 1994, 41, (11), pp 2145 - 2151.
- [9] Xilinx, "Specifications Versal Architecture," 2022, https://www.xilinx.com/support/documentation/data_sheets/ds950-versal-overview.pdf.
- [10] Li Zhang, Shulian Zhang, Zhou Lufei, Yusaku Fujii, "Nanometer-resolution displacement measurement system based on weak feedback effect of dual-frequency laser," *Physics Procedia*, 2009 2, (1), pp 19 – 26.
- [11] Vera-Salas, L. A., Moreno-Tapia, S. V., Garcia-Perez, A., Romero-Troncoso, R. D. J., Osornio-Rios, R. A., Serroukh, I., and Cabal-Yepez, E., "FPGA-based smart sensor for online displacement measurements using a heterodyne interferometer," *Sensors*, 2011, 11, (8), pp 7710 - 7723.
- [12] Cui, K., Li, S., Ren, Z., & Zhu, R. "A highly compact and efficient interrogation controller based on FPGA for fiber-optic sensor array using interferometric TDM. *IEEE Sensors Journal*, 2017, 17 (11), pp 3490 - 3496.
- [13] Ehtesham, A., Zabit, U., Bernal, O. D., Raja, G., and Bosch, T. "Analysis and implementation of a direct phase unwrapping method for displacement measurement using self-mixing interferometry," *IEEE Sensors Journal*, 2017, 17, (22), pp 7425 - 7432.
- [14] Liping Yan, Zhouqiang Chen, Benyong Chen, Jiandong Xie, Shihua Zhang, Yingtian Lou, and Enzheng Zhang, "Precision PGC demodulation for homodyne interferometer modulated with a combined sinusoidal and triangular signal," *Opt. Express*, 2018, 26, (4), pp 4818 - 4831.
- [15] J. Elaskar, M. A. Luda, L. Tozzetti, J. Codnia and C. J. Oton, "FPGA-Based High-Speed Optical Fiber Sensor Based on Multitone-Mixing Interferometry," *IEEE Transactions on Instrumentation and Measurement*, 2022, 71, pp 1-11.
- [16] Jiménez, A.; Saucedo, Á.; Muñoz, A.; Duarte, J.; Mireles, J., Jr. FPGA-Based Hardware Implementation of Homodyne Demodulation for Optical Fiber Sensors. *Photonics* 2023, 10, 258. <https://doi.org/10.3390/photonics10030258>.

- [17] Mireles, J., Jr.; Saucedo, Á.; Jiménez, A.; Ramos, M.; Gonzalez-Landaeta, R. Design and Development of a MOEMS Accelerometer Using SOI Technology. *Micromachines* 2023, 14, 231. <https://doi.org/10.3390/mi14010231>.
- [18] D. A. Protopapas, *Microcomputer hardware design*, Englewood Cliffs, N.J.: Prentice Hall, 1988.
- [19] R. B. Reese, *Microprocessors: from assembly language to C using the PIC18Fxx2*, Hingham, Mass: Da Vinci Engineering Press, 2005.
- [20] J. S. Parab, *Exploring C for microcontrollers: a hands on approach*, London: Springer, 2007.
- [21] P. Quiroga, *Arquitectura de computadoras*, Buenos Aires: Alfaomega, 2010.
- [22] C. GU, *Building Embedded Systems*, San Diego: Apress, 2016.
- [23] P. P. Chu, *FPGA prototyping by VHDL examples*, Hoboken, NJ, USA: Wiley, 2017.
- [24] F. Vahid, *Digital design, with RTL design, VHDL, and Verilog*, Hoboken, NJ: Wiley, 2011.
- [25] J. P. Deschamps, G. J. Antoine Bioul y G. D. Sutter, *Synthesis of Arithmetic Circuits: FPGA, ASIC, and Embedded Systems*, Hoboken, New Jersey: Wiley, 2006.
- [26] S. Brown y Z. Vranesic, *Fundamentals of Digital Logic with VHDL design*, New York: McGraw-Hill, 2009.
- [27] P. Marwedel, *Embedded Systems Design*, Dordrecht: Springer, 2006.
- [28] N. Botros, *HDL with Digital Design VHDL and Verilog*, Dulles, Virginia: Mercury Learning and Information, 2015.
- [29] J. F. Wakerly, *Digital design: principles and practices*, New Jersey: Prentice Hall, 2001.
- [30] Xilinx, «Xilinx,» Xilinx, [En línea]. Available: <https://www.xilinx.com/products/design-tools/microblaze.html#processorDetails>. [Último acceso: 25 febrero 2023].
- [31] Suneeta, R. Srinivasan y RamSagar, «SoC implementation of three phase BLDC motor using Microblaze soft IP core,» de 2017 International Conference on Computer, Communications and Electronics (Comptelix), Jaipur, India, 2017.
- [32] B. Muralikrishna y K. Gnana Deepika, «Input/output peripheral devices control through serial communication using MicroBlaze processor,» de 2012 International Conference on Devices, Circuits and Systems (ICDCS), Coimbatore, 2012.
- [33] P. P. Chu, *Embedded SoPC Design with Nios II Processor and VHDL Examples*, Hoboken, N.J: Wiley, 2011.
- [34] Connelly, M. J. (2002). Digital synthetic-heterodyne interferometric demodulation. *Journal of Optics A: Pure and Applied Optics*, 4(6), S400.
- [35] Rodríguez Antonio, A. (2015). Diseño y construcción de un circuito demodulador de fase óptica para su utilización en la extracción de señales de biosensores interferométricos. *Licenciatura en Ingeniería Biomédica*.

Anexos

1.1 Taxonomía de los Roles de Colaborador (con las actividades logradas)

Roles	Definición de los roles	Nombre de él(la) investigador(a)	Figura	Grado de contribución	Actividades logradas durante el proyecto	Tiempo promedio semanal (en horas) dedicado al proyecto
Responsabilidad de la dirección del proyecto	Coordinar la planificación y ejecución de la actividad de investigación. Organiza los roles de cada colaborador, tiene la habilidad de identificar potenciales de cada individuo para generar una sinergia de equipo colaborativo.	Abimael Jiménez Pérez	Director del proyecto	Principal	Diseño de la arquitectura del sistema embebido en HDL.	8 hrs.
Desarrollo o diseño de la metodología	Contribuir con el diseño de la metodología, modelos a implementar y el sustento teórico, empírico y científico para la aplicabilidad de los instrumentos en la ejecución del proyecto.	Ángel Saucedo Carvajal	Diseñador de la metodología	Principal	Diseño de la configuración electro-óptica del sistema y el análisis del algoritmo de demodulación heterodina.	8 hrs.
Elaboración del análisis formal de la investigación	Aplicar métodos estadísticos, matemáticos, computacionales, teóricos u otras técnicas formales para analizar o sintetizar los datos del estudio. Verifica los resultados preliminares de cada etapa del análisis, los experimentos	J. Antonio Muñoz Gómez	Analista de datos	Principal	Análisis de los resultados del algoritmo demodulación en el sistema embebido y el desarrollo de software y aplicación.	6 hrs.

	implementados y otros productos comprometidos en el proyecto.					
Recopilación/recolección de datos e información	Ejecuta las estrategias propuestas en acciones encaminadas a obtener la información, haciendo la recopilación de datos y la inclusión de la evidencia en el proceso.	José Mireles Jr. García	Analista de datos	Principal	Integración de reportes técnicos, análisis de datos e información de los resultados obtenidos.	5 hrs.
Elaboración del análisis formal de la investigación	Aplicar métodos estadísticos, matemáticos, computacionales, teóricos u otras técnicas formales para analizar o sintetizar los datos del estudio. Verifica los resultados preliminares de cada etapa del análisis, los experimentos implementados y otros productos comprometidos en el proyecto.	Rafael Eliecer González Landaeta	Analista de datos	Principal	Análisis de resultados de los sistemas electrónicos analógicos utilizados y su interacción con la respuesta del sistema embebido diseñado.	5 hrs.

1.1.1 Estudiantes participantes en el proyecto

Nombre de estudiante(s)	Matrícula	Tiempo promedio semanal (en horas) dedicado al proyecto	Actividades logradas en la ejecución del proyecto
José Ángel Duarte Espino	217037	10	Desarrollo de módulos de hardware para integrarse en la arquitectura de memoria mapeada del sistema embebido.

Ricardo Trejo Castro	177397	6	Pruebas del ADC, muestreo, resolución y frecuencia de la señal interferométrica.
Jesús Enrique Valenzuela de la Cruz	148715	4	Desarrollo de drivers y aplicaciones de cada uno de los módulos de hardware desarrollados.