**CAPITULO III.**

**Método.…………………………………………………………………………… 23**

* 1. Sujetos ………………………………………………………………………. 24
  2. Procedimiento ...……………………………………………………………..24
     1. Recolectar datos …………………………………………………......... 25
     2. Pre procesar datos ……………………………………………….......... 26
     3. Explorar datos …………………………………………………….......... 27
     4. Aplicar el algoritmo ……………………………………………………...28
     5. Evaluar el algoritmo ……………………………………………………. 28
     6. Utilizar el modelo …………………………………………………......... 28
  3. Materiales y equipo ………………………………………………………… 29

\subsection{Identificación del problema e investigación}

Se determina la problemática, en base al análisis del entorno actual de la computación, al cual se le buscará solución con el proyecto que se presenta en este trabajo. El problema que se aborda es la necesidad de encontrar alternativas a las tendencias actuales para el procesamiento de señales con mejores prestaciones de cómputo intensivo y con menor consumo de potencia, específicamente para la operación de la convolución.

Además, se recopila información acerca de trabajos similares en bases de datos de artículos científicos y en libros técnicos. Esta información sirve como punto de referencia para comparar el rendimiento de la arquitectura propuesta en este trabajo, además se analizan las áreas de oportunidades de estos trabajos para proponer una solución más óptima al problema planteado. En esta etapa se investiga cuál es la situación actual de la tecnología de cómputo de alto rendimiento la cual se describe en el estado del arte.

\subsection{Determinación de los requerimientos y alcances del proyecto}

Se determinan los requerimientos que se buscan satisfacer con el coprocesador desarrollado en este trabajo. Además, se indican cuáles son los alcances y delimitaciones del proyecto. Este trabajo se limitará a diseñar la arquitectura de un coprocesador capaz de resolver la operación de la convolución de dos señales. Se implementará en un FPGA y se verificará a través de un entorno de simulación con la ayuda de MATLAB, donde se realizarán pruebas de rendimiento con respecto al tiempo de procesamiento. Las limitaciones del proyecto son el tiempo disponible, dinero, recursos y licencias.

\subsection{Programación en MATLAB de algoritmos de convolución}

Se programan diferentes algoritmos existentes para resolver la convolución utilizando MATLAB. El objetivo de esta etapa es comparar el rendimiento en tiempo de ejecución y uso de recursos de memoria de los diferentes algoritmos y analizar las ventajas y desventajas de cada uno de ellos.

\subsection{Desarrollo de la arquitectura mediante herramientas de software}

Se desarrolla la arquitectura en un ambiente de software, haciendo uso de programas de software para descripción y simulación de la arquitectura, una metodología de diseño y un lenguaje de descripción de hardware.

\subsubsection{Diseño de la arquitectura}

Mediante el uso de la metodología Top-Down y en base a las arquitecturas de la actualidad se desarrolla una arquitectura capaz de realizar la operación de la convolución con un menor consumo de potencia y con una velocidad de procesamiento comparable con las arquitecturas modernas.

\subsubsection{Desarrollo de arquitectura con Verilog}

Con la ayuda de un lenguaje de descripción de hardware, en este caso Verilog, se describe la arquitectura digital en Xilinx ISE 14.7. Se describen los diferentes bloques que forman parte de la arquitectura, posteriormente se unen estos elementos con un top level para formar un sistema capaz de convolucionar dos señales.

\subsubsection{Pruebas - validación funcional}

Se simula la arquitectura mediante el uso de test benchs. Los resultados se comparan con resultados obtenidos con la función predeterminada de MATLAB llamada conv() la cual realiza la operación de la convolución de dos señales. En caso de no obtener los resultados esperados, se vuelve a diseñar la arquitectura.

\subsubsection{Síntesis}

Se realiza la síntesis lógica del código, la cual consiste en convertir la descripción de hardware especificada mediante Verilog en una implementación de diseño en término de compuertas lógicas, la cual es un archivo de flujo de bits (.bit).

\subsection{Implementación a nivel de hardware}

En esta etapa se programa el FPGA. Además, se realizan pruebas a la arquitectura propuesta para comprobar que se cumplen los requerimientos planteados al inicio del proyecto y para verificar que la operación de la convolución se realiza de manera correcta.

\subsubsection{Implementación en FPGA}

Una vez que se cuenta con la arquitectura sintetizada, se programa el archivo de flujo de bits en el FPGA utilizando el software Digilent Adept. Este archivo le indica al FPGA cómo se tiene que configurar para realizar la operación de la convolución.

Pruebas

Se realizan pruebas al hardware mediante una interfaz de comunicación serial la cual conecta al FPGA con una PC. Se envian dos señales al FPGA, las cuales se convolucionan en la arquitectura digital, y este da como resultado una tercera señal la cual se envía a la PC. El resultado se compara con resultados obtenidos bajo un ambiente conocido en MATLAB y sirve para comprobar que el sistema digital está funcionando de manera correcta. Además, se realizan pruebas de velocidad de procesamiento y consumo de potencia para comparar con diferentes arquitecturas.

# lll. MÉTODO

A continuación, se describe la ruta metodológica que se tomó para conseguir resolver el problema planteado.

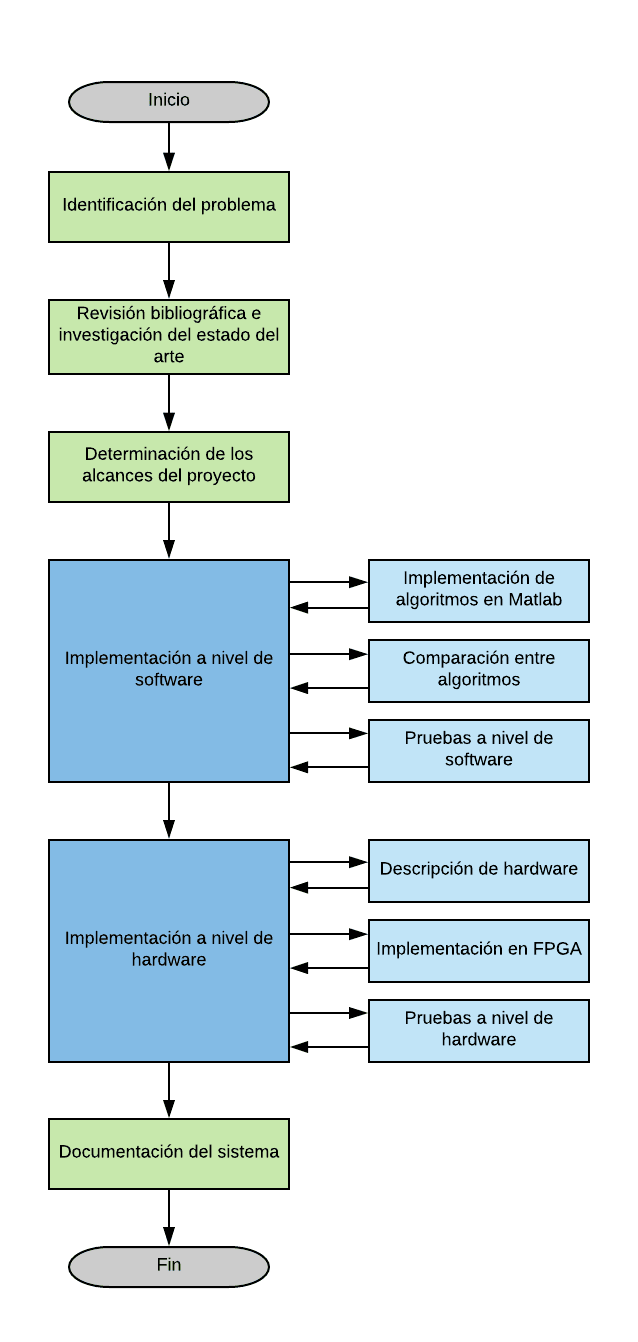
En este capítulo se describe la ruta metodológica que se tomó para diseñar y desarrollar el coprocesador para la convolución. Primero, se indica el sujeto de estudio. Después, se muestra un diagrama de flujo con las etapas del proyecto, así como la descripción de cada una. Por último, se enlistan los materiales y herramientas utilizados.

**3.1 Sujeto**

Investigadores e ingenieros del area de desarrollo de hardware y cómputo de alto rendimiento.

**3.2 Procedimiento**

1. Inicio
2. Identificación del problema
3. Delimitación del alcance del proyecto
4. Diseño mediante herramientas de software
   1. Programar algoritmo en lenguaje de programación a alto nivel
   2. Diseño de la arquitectura capaz de resolver la convolución en hardware utilizando una metodología de diseño (ya ses buttom up o la otra)
   3. Descripción de hardware mediante un HDL, en este caso será verilog
   4. Sintetizar el código y ver que construya bien
   5. Pruebas con las herramientas de desarrollo - hacer test benchs
5. Implementación a nivel hardware
   1. Programar el FPGA
   2. Pruebas a nivel de hardware
6. Documentación del proyecto
7. Fin



***Figura 1: Diagrama de flujo de método a seguir. - CHECAR EL OTRO DIAGRAMA DE FLUJO QUE YA TENGO en ../tescrito/img/procedimiento.jpeg***

En la figura 4 se muestran los pasos que se siguieron para desarrollar el coprocesador para la convolución. A continuación, se describre cada una de las etapas del proyecto.

**3.2 Identificación del problema**

Se determina la problemática en base al análisis del entorno actual de la computación al cual se le buscará solución con el proyecto que se presenta en este trabajo. El problema que se aborda es la necesidad de encontrar alternativas a las tendencias actuales para el procesamiento de señales con mejores prestaciones de cómputo intensivo y con menor consumo de potencia.

**3.2.1.**

**Identificación del problema e investigación**

Se determina la problemática, en base al análisis del entorno actual de la computación, al

cual se le buscará solución con el proyecto que se presenta en este trabajo. El problema

que se aborda es la necesidad de encontrar alternativas a las tendencias actuales para

el procesamiento de señales con mejores prestaciones de cómputo intensivo y con menor

consumo de potencia, especı́ficamente para la operación de la convolución.

Además, se recopila información acerca de trabajos similares en bases de datos de artı́cu-

los cientı́ficos y en libros técnicos. Esta información sirve como punto de referencia para

comparar el rendimiento de la arquitectura propuesta en este trabajo, además se analizan

las áreas de oportunidades de estos trabajos para proponer una solución más óptima al

problema planteado. En esta etapa se investiga cuál es la situación actual de la tecnologı́a

de cómputo de alto rendimiento la cual se describe en el estado del arte.

**3.3 Revisión bibliográfica e investigación del estado del arte**

Se recopila información en base de datos de artículos científicos y en libros técnicos trabajos similares al que se está proponiendo en este documento para poder comparar las similitudes y qué se puede mejorar con respecto a cómo se han realizado al momento. Esta etapa consiste en la investigación del estado del arte, es decir, cuál es la situación actual de esta tecnología.

**3.3 Determinación de los alcances del proyecto**

Se determina cuáles son los alcances del proyecto, delimitaciones, es decir hasta dónde se va a investigar y desarrollar. El diseño del procesador se limitará a implementar la arquitectura para un coprocesador para la convolución. Se implementará en un FPGA y verificará a través de un entorno de simulación con la ayuda de Matlab. Además, que se plantea el por qué no se puede realizar más trabajo. En este proyecto las limitaciones son el tiempo disponible, dinero, recursos y licencias.

**3.2.2.**

**Determinación de los requerimientos y alcances del proyecto**

Se determinan los requerimientos que se buscan satisfacer con el coprocesador desarrolla-

do en este trabajo. Además, se indican cuáles son los alcances y delimitaciones del proyecto.

Este trabajo se limitará a diseñar la arquitectura de un coprocesador capaz de resolver la

operación de la convolución de dos señales. Se implementará en un FPGA y se verificará a

través de un entorno de simulación con la ayuda de MATLAB, donde se realizarán pruebas

de rendimiento con respecto al tiempo de procesamiento. Las limitaciones del proyecto son

el tiempo disponible, dinero, recursos y licencias.

**3.2.3.Programación en MATLAB de algoritmos de convolución**

Se programan diferentes algoritmos existentes para resolver la convolución utilizando MATLAB.

El objetivo de esta etapa es comparar el rendimiento en tiempo de ejecución y uso de recur-

sos de memoria de los diferentes algoritmos y analizar las ventajas y desventajas de cada

uno de ellos.

**3.4 Implementación a nivel de software**

3.2.4.

Desarrollo de la arquitectura mediante herramientas de software

Se desarrolla la arquitectura en un ambiente de software, haciendo uso de programas de

software para descripción y simulación de la arquitectura, una metodologı́a de diseño y un

lenguaje de descripción de hardware.

Diseño de la arquitectura

Mediante el uso de la metodologı́a Top-Down y en base a las arquitecturas de la actua-

lidad se desarrolla una arquitectura capaz de realizar la operación de la convolución con

un menor consumo de potencia y con una velocidad de procesamiento comparable con las

arquitecturas modernas.

Desarrollo de arquitectura con Verilog

Con la ayuda de un lenguaje de descripción de hardware, en este caso Verilog, se describe la

arquitectura digital en Xilinx ISE 14.7. Se describen los diferentes bloques que forman parte

de la arquitectura, posteriormente se unen estos elementos con un top level para formar un

sistema capaz de convolucionar dos señales.

Pruebas - validación funcional

Se simula la arquitectura mediante el uso de test benchs. Los resultados se comparan con

resultados obtenidos con la función predeterminada de MATLAB llamada conv() la cual rea-

liza la operación de la convolución de dos señales.

Sı́ntesis

Se realiza la sı́ntesis lógica del código, la cual consiste en convertir la descripción de hard-

ware especificada mediante Verilog en una implementación de diseño en término de com-

puertas lógicas, la cual es un archivo de flujo de bits (.bit).

Se desarrolla el software del proyecto.

**3.4.1 Implementación de algoritmos en Matlab**

Se programan diferentes algoritmos para realizar la convolución en Matlab para su posterior análisis. Entre ellos están el algoritmo desde la entrada y el algoritmo desde la salida. En este trabajo se trata con la convolución discreta.

**3.4.2 Comparación entre algoritmos**

Se realiza una comparación entre los algoritmos implementados con la ayuda de las funciones predeterminadas con las que cuenta Matlab.

**3.4.3 Pruebas a nivel de software**

Se realizan diferentes pruebas a los diferentes algoritmos, como medir el tiempo de ejecución de cada uno de ellos. Además, que se utiliza la función de Matlab que sirve para paralelizar las operaciones y se observa el comportamiento de cada uno de los algoritmos. Para comprobar que la implementación de los algoritmos es correcta se compara con la función predeterminada de Matlab llamada conv().

**3.5 Implementación a nivel de hardware**

Se desarrolla el hardware del proyecto.

**3.5.1 Descripción de hardware**

Con la ayuda de un lenguaje de descripción de hardware, en este caso Verilog, se describe la arquitectura digital. Se describen los diferentes bloques que forman parte de la arquitectura, posteriormente se unen estos elementos con un *top level.* Por último, se simula la arquitectura y se comprueba que está libre de errores.

**3.5.2 Implementación en FPGA**

Una vez que se cuenta con la arquitectura se sintetiza, se seleccionan los pines de entrada – salida y se programa el FPGA.

**3.5.3 Pruebas a nivel de hardware**

Se realizan pruebas al hardware mediante una interfaz de comunicación serial la cual conecta al FPGA con una PC. Se alimenta al FPGA con dos señales, las cuales se convolucionan en la arquitectura digital, y el FPGA da como resultado una tercera señal. Este resultado se compara con resultados obtenidos bajo un ambiente conocido en Matlab y sirve para comprobar que el sistema digital está funcionando de manera adecuada. Se realizan pruebas de velocidad de procesamiento y consumo de potencia.

3.2.5.Implementación a nivel de hardware

En esta etapa se programa el FPGA. Además, se realizan pruebas a la arquitectura pro-

puesta para comprobar que se cumplen los requerimientos planteados al inicio del proyecto

15y para verificar que la operación de la convolución se realiza de manera correcta.

Implementación en FPGA

Una vez que se cuenta con la arquitectura sintetizada, se programa el archivo de flujo de

bits en el FPGA utilizando el software Digilent Adept. Este archivo le indica al FPGA cómo

se tiene que configurar para realizar la operación de la convolución.

Pruebas

Se realizan pruebas al hardware mediante una interfaz de comunicación serial la cual co-

necta al FPGA con una PC. Se envian dos señales al FPGA, las cuales se convolucionan en

la arquitectura digital, y este da como resultado una tercera señal la cual se envı́a a la PC.

El resultado se compara con resultados obtenidos bajo un ambiente conocido en MATLAB y

sirve para comprobar que el sistema digital está funcionando de manera correcta. Además,

se realizan pruebas de velocidad de procesamiento y consumo de potencia para comparar

con diferentes arquitecturas.

**3.6 Documentación**

Se realiza un trabajo escrito el cual contiene los aspectos más relevantes del proyecto. Se da una introducción al proyecto, después se pone en contexto al lector, para al final presentar el trabajo, los resultados y el análisis de los mismos.

Por último, se realiza un trabajo escrito en el cual se documentan los aspectos más relevantes del proyecto. Primero, se da una introducción al proyecto y se pone en contexto al lector, después se presenta el trabajo realizado así como se describe a detalle la forma en que se realizó cada etapa, posteriormente, se presentan los resultados obtenidos así como un análisis de los mismos, haciendo una comparación con las arquitecturas que ya están disponibles en el mercado, por último, se concluye y se proponen mejoras para trabajos futuros.

**3.3 Materiales y Herramientas**

- Laptop

- GCC - GNU Compiler Collection

- Xilinx Vivado

- Laptop Dell Inspiron 13-5378

- MathWorks MATLAB R2015a

- Xilinx ISE 14.7

- 16Digilent Adept

- agregar los que faltan (FPGA, OS(?))