**Roberto Valenzuela - Convolucionador**

**Capítulo lll. Método**

3.1 Sujetos

3.2 Procedimiento

3.2.1 Identificación del problema

3.2.2 Revisión bibliográfica e investigación del estado del arte

3.2.3 Determinación de los alcances del proyecto

3.2.4 Diseño del modelo de referencia en alto nivel

3.2.4.1 Implementación de algoritmos en C

3.2.4.2 Pruebas a nivel de software

3.2.4.3 Diseño de arquitectura de alto nivel

3.2.5 Implementación a nivel de hardware

3.2.5.2 Descripción de hardware en Verilog

3.2.5.3 Pruebas tipo test bench con herramientas de XIlinx

3.2.6 Documentación

3.3 Materiales y equipo

**lll. Método**

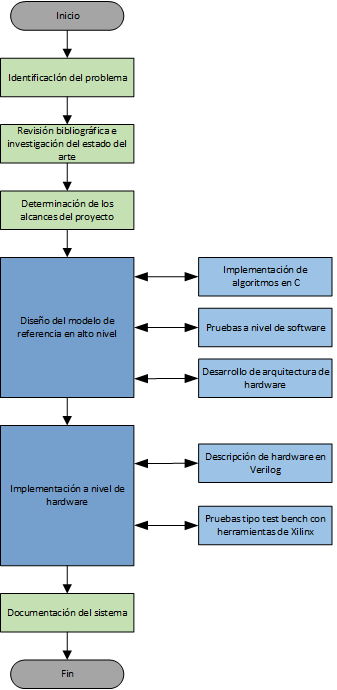
A continuación, se describe la ruta metodológica que se tomó para conseguir resolver el problema planteado en el capítulo 1. En este capítulo se describen los pasos que se tomaron para diseñar y desarrollar el coprocesador para la convolución. Primero, se indica el sujeto de estudio. Después, se muestra 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 Sujetos**

Este proyecto fue desarrollado por un alumno de la carrera ingeniería en electrónica asesorado por un maestro del área de ingeniería eléctrica y electrónica del Instituto Tecnológico de Sonora. El presente trabajo va dirigido a científicos, investigadores, estudiantes o ingenieros que deseen expandir sus conocimientos en el área de diseño digital y desarrollo de hardware, procesamiento de señales digitales a nivel de hardware o cómputo de alto rendimiento.

**3.2 Procedimiento**

En el diagrama de la figura X se muestran los pasos que se siguieron para diseñar y desarrollar el coprocesador para la convolución. A continuación, se describe cada una de las etapas del proyecto.

**

*Figura x. Diagrama de flujo de procedimiento a seguir*

**3.2.1 Identificación del problema**

Se determina la problemática en base al análisis del entorno actual de la computación a la cual se le buscará dar 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.

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

Se recopila información sobre trabajos similares propuesto en este documento 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 óptima al problema planteado. En esta etapa se investiga cuál es la situación actual de la tecnología de diseño digital para cómputo de alto rendimiento la cual se describe en el estado del arte.

**3.2.3 Determinación de los alcances del proyecto**

Se determina cuáles son los requerimientos del proyecto, es decir, los alcances, limitaciones y delimitaciones de la investigación y desarrollo del proyecto. Este trabajo se limitará a diseñar la arquitectura para un coprocesador para la convolución utilizando el método de *sytolic arrays*. Se describirá el hardware utilizando Verilog y se verificará su correcto funcionamiento mediante test benchs realizados con las herramientas de Xilinx. Los resultados obtenidos se comprobarán mediante un entorno de simulación de Matlab, el cual contiene funciones nativas para calcular la operación de la convolución. En este proyecto las limitaciones son el tiempo disponible, recursos y licencias.

**3.2.4 Diseño del modelo de referencia en alto nivel**

Se diseña la arquitectura del convolucionador para una y dos dimensiones en base a las arquitecturas de systolic arrays propuestas por Kung [1982].

**3.2.4.1 Implementación de algoritmos en C**

Se programan diferentes algoritmos conocidos para resolver la convolución utilizando el lenguaje de programación C. Entre ellos están el algoritmo desde la entrada, el algoritmo desde la salida, algoritmo de la convolución circular y el algoritmo en el dominio de la frecuencia.

**3.2.4.2 Pruebas a nivel de software**

Se realiza una comparación entre los algoritmos implementados con la ayuda de las funciones predeterminadas con las que cuenta Matlab para verificar su correcto funcionamiento. 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.

**3.2.4.3 Diseño de arquitectura de alto nivel**

Se diseña la arquitectura del convolucionador. El resultado de este paso es un diagrama de bloques que describe de manera detallada los diferentes componentes de hardware que constituyen al convolucionador. Se decidió utilizar el método de *systolic arrays* para resolver la operación de la convolución en una y dos dimensiones, es por esto que se requiere de un vector de elementos de procesamiento que en conjunto sean capaces de realizar la convolución de dos señales, siendo esta la pieza más importante del diseño.

**3.2.5 Implementación a nivel de hardware**

Se desarrolla el hardware del proyecto utilizando el lenguaje de descripción de hardware Verilog.

**3.2.5.1 Descripción de hardware en Verilog**

Utilizando un lenguaje de descripción de hardware, en este caso Verilog, se describe la arquitectura diseñada en el paso anterior en el programa de Xilinx Vivado 2019.2. Haciendo uso de una metodología de diseño top-down se describen los diferentes bloques que forman parte de la arquitectura, posteriormente se unen estos elementos con un módulo top-level para formar un sistema capaz de convolucionar dos señales.

**3.2.5.3 Pruebas tipo test bench con herramientas de Xilinx**

Se simula la arquitectura mediante el uso de test benchs con las herramientas proporcionadas por Xilinx Vivado. De manera similar a lo realizado con los resultados obtenidos en la fase de implementación de algoritmos en C, los resultados obtenidos con el convolucionador desarrollado en Verilog se comparan con resultados obtenidos para las mismas señales utilizando la función predeterminada de MATLAB llamada conv() la cual realiza la operación de la convolución de dos señales. En caso de obtener los mismos resultados, se concluye que la arquitectura del convolucionador funciona de manera correcta, en caso de que los resultados difieran, se analiza la causa raíz de la falla y se vuelve a la fase de diseño y desarrollo de la arquitectura.

**3.2.6 Documentación**

Como paso final, se realiza un escrito en donde se documenta de manera científica los aspectos más relevantes del proyecto. Este documento contiene la investigación, diseño, desarrollo, resultados y conclusiones obtenidos en la elaboración de este trabajo. La finalidad del este documento es servir como referencia para futuras investigaciones en el área de desarrollo digital.

**3.3 Materiales y equipo**

- Laptop HP con procesador i7

- Windows 10

- Ubuntu 20.24

- Xilinx Vivado 2019.2

- MathWorks MATLAB R2015b

- GNU Compiler Collection gcc version 9.3.0

- Microsoft Office 2016