



**FACULTAD
DE INGENIERIA**

Universidad de Buenos Aires

Implementación automatizada de redes neuronales para sistemas embebidos

Autor:

Ing. José David Alvarado Moreno

Director:

Ing. Federico G. Zacchigna (FIUBA)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 02 de marzo de 2021 y el 22 de abril de 2021.*

Índice

Registros de cambios	3
Acta de constitución del proyecto.	4
Descripción técnica-conceptual del proyecto a realizar	4
Identificación y análisis de los interesados.	6
1. Propósito del proyecto.	6
2. Alcance del proyecto	7
3. Supuestos del proyecto.	7
4. Requerimientos	7
Historias de usuarios (<i>Product backlog</i>)	8
5. Entregables principales del proyecto	9
6. Desglose del trabajo en tareas	9
7. Diagrama de Activity On Node	10
8. Diagrama de Gantt.	11
9. Matriz de uso de recursos de materiales	12
10. Presupuesto detallado del proyecto	13
11. Matriz de asignación de responsabilidades	13
12. Gestión de riesgos.	13
13. Gestión de la calidad	15
14. Comunicación del proyecto	19
15. Gestión de compras.	19
16. Seguimiento y control.	19
17. Procesos de cierre.	20

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
1.0	Creación del documento	02/03/2021
1.1	Avances ítems del 1 al 6	16/03/2021
1.2	Correcciones ítems del 1 al 6 e historias de usuarios	23/03/2021
1.3	Correcciones ítems del 1 al 6 realizadas por el director	30/03/2021
1.4	Correcciones ítems del 1 al 6 e historias de usuarios por el docente	01/04/2021
1.5	Avances ítems del 6 al 11	02/04/2021
1.6	Avances ítems del 12 al 17	06/04/2021
1.7	Correcciones ítems del 6 al 17	13/04/2021

Acta de constitución del proyecto

Buenos Aires, 02 de marzo de 2021

Por medio de la presente se acuerda con el Ing. José David Alvarado Moreno que su Trabajo Final de la Carrera de Especialización en Sistemas Embebidos se titulará “Implementación automatizada de redes neuronales para sistemas embebidos”, consistirá esencialmente en un prototipo preliminar de la implementación de redes neuronales convolucionales automatizadas, para reducir significativamente la complejidad y el tiempo que conlleva el codiseño de HW-SW, y tendrá un presupuesto preliminar estimado de 600 hs de trabajo y un costo de \$150 USD, con fecha de inicio 02 de marzo de 2021 y fecha de presentación pública 30 de noviembre de 2021.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg
Director posgrado FIUBA

Ing. Federico G. Zacchigna
FIUBA

Ing. Federico G. Zacchigna
Director del Trabajo Final

Descripción técnica-conceptual del proyecto a realizar

En la actualidad encontramos diferentes aplicaciones de inteligencia artificial que utilizan redes neuronales (RN). Para este tipo de desarrollos generalmente se requiere procesar un alto volumen de información por lo que es necesario que los dispositivos a utilizar incorporen una gran capacidad de procesamiento y almacenamiento, para lo cual se puede encontrar con diferentes alternativas de sistemas embebidos en donde es posible realizar esta implementación como por ejemplo: MCU, GPU, FPGA, DLP y ASIC. Al evaluar cada uno en términos de desempeño, flexibilidad y capacidad de integración en un sistema embebido (SE) encontramos que los MCU y ASIC presentan una baja *performance*; las GPU presentan baja capacidad de integración y los DLP presentan una flexibilidad para aplicaciones específicas; y por último encontramos que las FPGA presentan una alta flexibilidad, aunque muchas veces se ve opacada por la complejidad que conlleva el desarrollo del sistema.

El proyecto general busca el desarrollo de un software (SW) para la implementación del algoritmo de clasificación de una Red Neuronal (RN) de manera automatizada a partir de un modelo de alto nivel y que incorpore una interfaz necesaria para acceder, controlar y monitorear su funcionamiento desde un microcontrolador de propósito general. para lo cual se utilizará una SOC-FPGA en el cual coexisten un microprocesador de propósito general y un bloque de hardware (HW) dedicado exclusivamente a realizar el procesamiento de datos asociado a la RN, cuyo fin es el de liberar al microprocesador de estas tareas. Al diseño de este tipo de arquitecturas se lo conoce como “codiseño de HW-SW”. En la Figura 1 se muestra un diagramas en bloques simplificado del sistema.

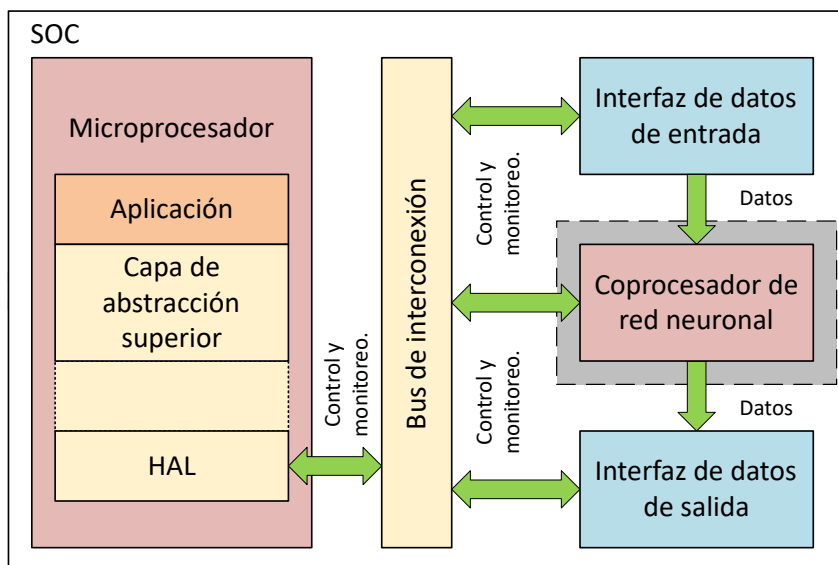


Figura 1. Diagrama en bloques del sistema

En este proyecto se realizará específicamente el bloque de hardware en la FPGA denominado ***coprocesador de red neuronal*** (bloque resaltado en color gris de la Figura 1), para un modelo de red neuronal convolucional (RNC). En la Figura 2 se presenta el modelo de una RNC, en donde se pueden identificar las diferentes etapas para el proceso de clasificación. En este tipo de RNC las etapas de convolución y *subsampling* se pueden realizar en varias veces en las

diferentes capas dependiendo del modelo diseñado, y lo que se busca con el sistema propuesto de red automatizada es que sea posible ajustar el número y el tamaño de las diferentes capas internas de la red como se muestra en la Figura 3.

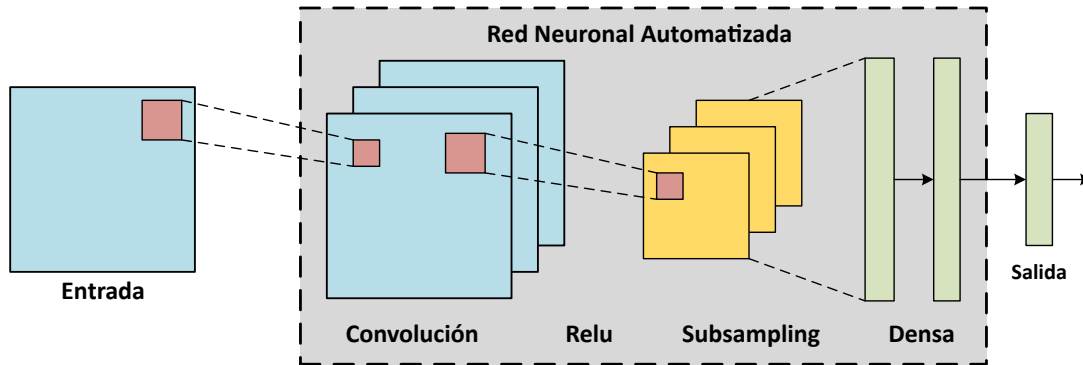


Figura 2. Diagrama en bloques red neuronal convolucional

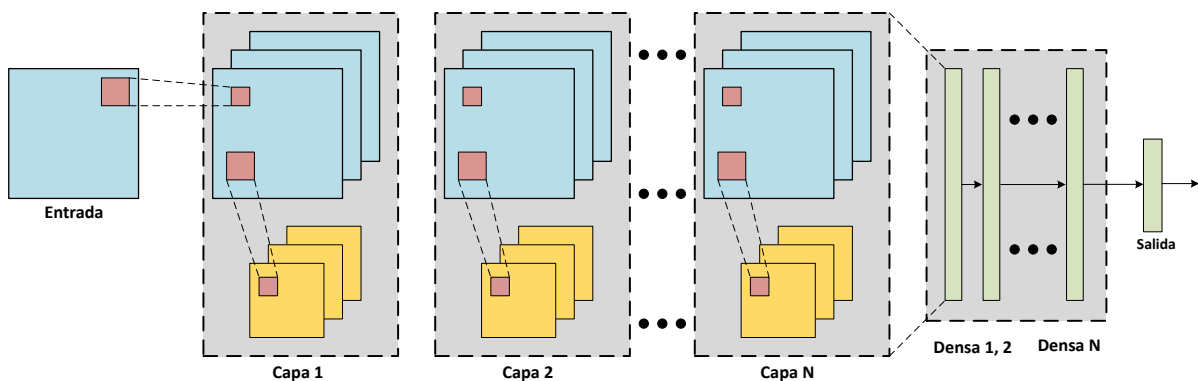


Figura 3. Diagrama en bloques red neuronal automatizada

Este trabajo tiene como objetivo principal reducir significativamente la complejidad y el tiempo que conlleva el codiseño de HW-SW e implementación de una RN en una SoC-FPGA y de esta manera eliminar una de las grandes barreras para su utilización.

Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Ing. Federico G. Zacchigna	FIUBA	-
Responsable	Ing. José David Alvarado Moreno	FIUBA	Alumno
Orientador	Ing. Federico G. Zacchigna	FIUBA	Director Trabajo final

1. Propósito del proyecto

El propósito de este proyecto es el desarrollo de un módulo de hardware para un coprocesador de red neuronal a partir de un modelo de alto nivel, en el cual se implementarán todas las capas de la red neuronal (RN), *subsampling* y clasificación de la RN, en donde sea posible ajustar la arquitectura de la RNC a partir de los parámetros definidos por el tamaño de los datos de entrada y las características definidas de la arquitectura para el número de capas de la RN.

2. Alcance del proyecto

En el presente proyecto se realizarán las siguientes actividades:

- Establecer el modelo de alto nivel de red neuronal utilizando una herramienta de software especializada.
- Diseñar e implementar la arquitectura de alto nivel y la micro-arquitectura de cada uno de los módulos internos.
- Implementar el modelo de red neuronal automatizada utilizando lenguaje de alto nivel.
- Verificación y validación del funcionamiento del sistema.
- Realizar la documentación del trabajo final.

El presente proyecto no incluye realizar procesos de optimización en las etapas de cuantización y arquitectura de alto nivel, adicionalmente no se complementará con dispositivos para realizar la captura o para la visualización de los patrones. Por último, no se realizarán desarrollos de hardware o incorporación de periféricos externos a los que están en la placa PYNQ-Z2.

3. Supuestos del proyecto

Para el desarrollo de este proyecto se supone que es posible implementar las diferentes capas de la RNC en la placa TUL PYNQ-Z2 (ZYNQ XC7Z020-1CLG400C), y con la herramienta de software Vivado HLx versión webpack.

4. Requerimientos

1. Requerimientos funcionales etapa de convolución.

- 1.1. La herramienta debe permitir que los datos de entrada y salida en la capa de convolución se realicen por medio de un interfaz de *streaming* de datos.
- 1.2. La herramienta debe permitir cargar los valores de los pesos y el tamaño del kernel.
- 1.3. El sistema debe realizar la convolución para procesar la información de la capa anterior.

2. Requerimientos funcionales etapa de *subsampling*.

- 2.1. La herramienta debe permitir que los datos de entrada y salida en la capa de *subsampling* se realicen por medio de un interfaz *streaming* de datos.
- 2.2. La herramienta debe permitir seleccionar el método de *subsampling*.
- 2.3. El sistema debe realizar la *subsampling* para procesar la información de la capa anterior.
3. Requerimientos funcionales etapa de clasificación (Capa densa).
 - 3.1. La herramienta debe permitir que los datos de entrada y salida en la capa de densa se realicen por medio de un interfaz de *streaming* de datos.
 - 3.2. El sistema debe realizar la clasificación para procesar la información de la capa anterior.
4. Requerimientos funcionales de software.
 - 4.1. La herramienta debe tomar como entrada un modelo de una red neuronal de alto nivel (Keras).
 - 4.2. La herramienta debe generar un modelo de la arquitectura digital del alto nivel.
 - 4.3. La herramienta debe generar un modelo de la arquitectura de alto nivel a partir del modelo cuantizado.
 - 4.4. La herramienta debe generar una *Golden Reference* (GR) a partir de la arquitectura de alto nivel.
 - 4.5. La herramienta debe generar el código HDL sintetizable a partir de la arquitectura de alto nivel.
 - 4.6. La herramienta debe generar un *testbench* de simulación, que permita comparar los resultados de HW con los resultados de la GR.
5. Requerimientos funcionales del código HDL generado.
 - 5.1. El sistema debe presentar un funcionamiento equivalente al modelo cuantizado.
 - 5.2. El sistema debe utilizar un número de multiplicadores fijo por cada etapa de acuerdo a la arquitectura de alto nivel.
 - 5.3. El sistema debe ser configurado, controlado y monitoreado a través de un banco de registros.
6. Requerimientos documentales.
 - 6.1. Se debe realizar el documento de informe de avance.
 - 6.2. Se debe realizar el documento de informe final.

Historias de usuarios (*Product backlog*)

A continuación, se encuentran las historias de usuario:

- Como usuario he diseñado y entrenado una RN utilizando Keras y quiero migrarla a un sistema embebido..
- Como usuario quiero explorar distintas estrategias de cuantización de una RN y relevar el impacto en la utilización de recursos al implementarla en una FPGA.

- Como usuario quiero explorar distintas arquitecturas de RN para su implementación en FPGA y evaluar el rendimiento y la utilización de recursos en cada caso.
- Como usuario quiero realizar pruebas de campo de RN implementadas en un sistema embebido para evaluar su rendimiento en un ambiente real.

5. Entregables principales del proyecto

- Plan de trabajo del proyecto.
- SW de generación automática del código VHDL.
- Documento del SW de generación automática.
- Documento del módulo generado automáticamente.
- Documento de informe de avance.
- Documento de informe final.

6. Desglose del trabajo en tareas

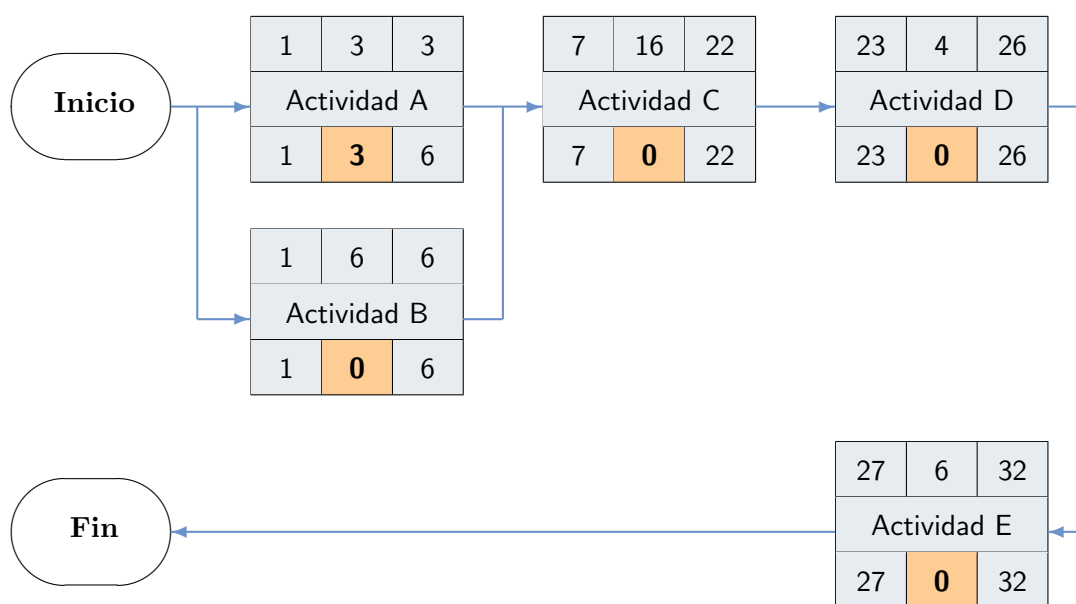
A continuación se presenta el desglose de las actividades y la duración estimada para el desarrollo del proyecto.

1. Planificación general del proyecto.
 - 1.1. Elaboración del plan de trabajo del proyecto 30 hs
 - 1.2. Planificación del proyecto 10 hs
2. Revisión bibliográfica y especificaciones.
 - 2.1. Revisión de la bibliografía específica 40 hs
 - 2.2. Especificación de requisitos 20 hs
 - 2.3. Definición de los casos de uso 20 hs
3. Desarrollo de red neuronal automatizada.
 - 3.1. Desarrollo del algoritmo de ventaneo 30 hs
 - 3.2. Desarrollo del algoritmo de convolución 60 hs
 - 3.3. Desarrollo del algoritmo de *subsampling* 40 hs
 - 3.4. Desarrollo de las unidades de control 40 hs
 - 3.5. Desarrollo del algoritmo de clasificación 60 hs
 - 3.6. Desarrollo del sistema de integración de la arquitectura 60 hs
4. Verificación y validación.
 - 4.1. Pruebas de las capas: convolución, *subsampling* y densa. 20 hs

- | | | |
|-----------------------|--|-------|
| 4.2. | Pruebas integrales del sistema y validación completa del proceso de automatización | 20 hs |
| 4.3. | Ejecución y pruebas de los casos de uso | 15 hs |
| 4.4. | Evaluar el cumplimiento de los requerimientos | 15 hs |
| 5. Proceso de cierre. | | |
| 5.1. | Elaboración del documento del SW de generación automática | 10 hs |
| 5.2. | Elaboración del documento del sistema generado automáticamente | 10 hs |
| 5.3. | Elaboración del informe de avance del proyecto | 20 hs |
| 5.4. | Elaboración del informe de final del proyecto | 50 hs |
| 5.5. | Elaboración de la presentación de la defensa | 10 hs |

Cantidad total de horas: 600 hs

7. Diagrama de Activity On Node



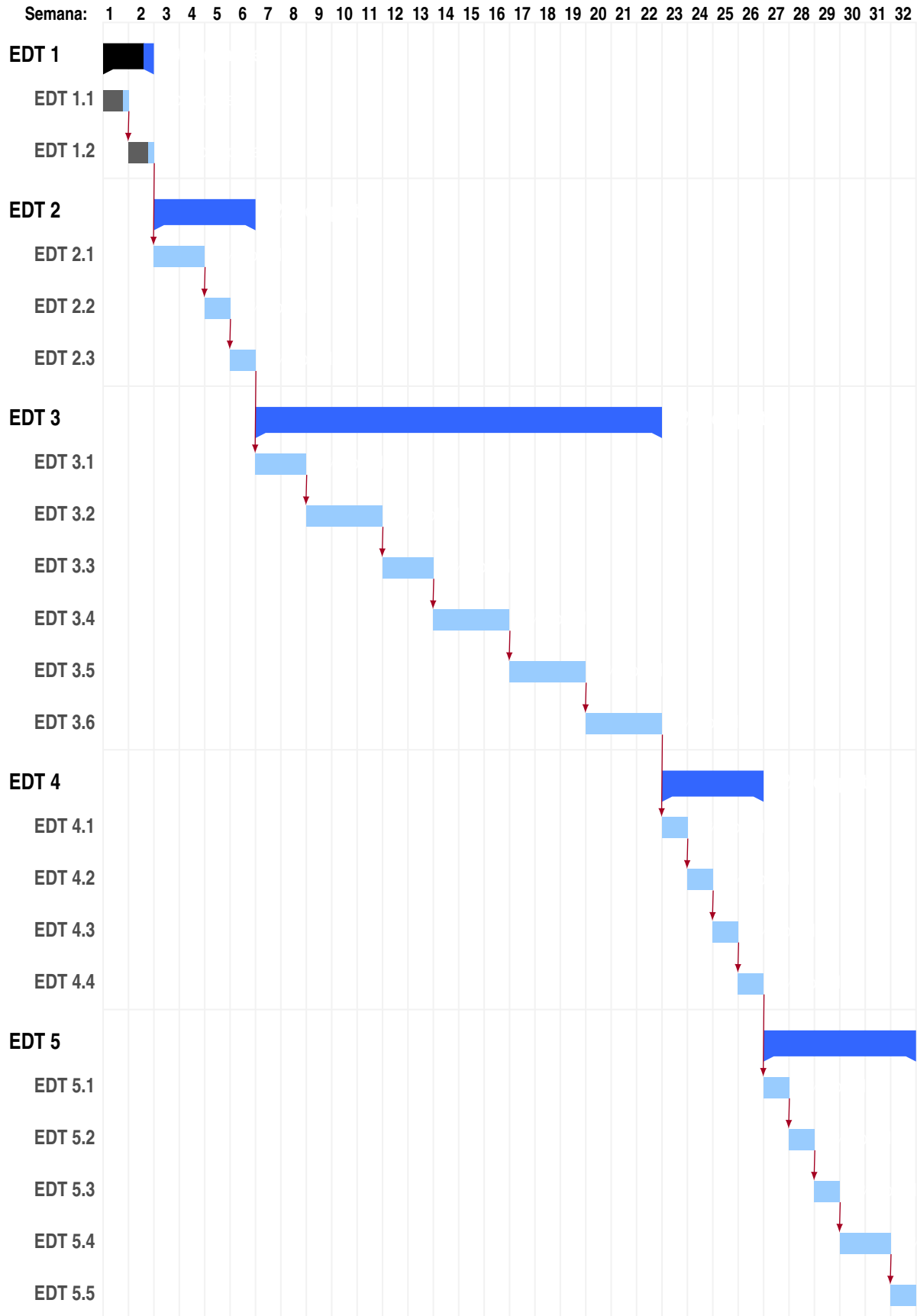
Lista de Actividades.

Actividad A: Planificación general del proyecto.
 Actividad B: Revisión bibliográfica y especificaciones.
 Actividad C: Desarrollo de red neuronal automatizada.
 Actividad D: Verificación y validación.
 Actividad E: Proceso de cierre.

El tiempo de las actividades está estimado en semanas.

8. Diagrama de Gantt

EDT	Nombre	Duración (semanas)
1.	Planificación general del proyecto	2
1.1	Elaboración del plan de trabajo del proyecto	1
1.2	Planificación del proyecto	1
2.	Revisión bibliográfica y especificaciones	4
2.1	Revisión de la bibliografía específica	2
2.2	Especificación de requisitos	1
2.3	Definición de los casos de uso	1
3.	Desarrollo de red neuronal automatizada	16
3.1	Desarrollo del algoritmo de ventaneo	2
3.2	Desarrollo del algoritmo de convolución	3
3.3	Desarrollo del algoritmo de <i>subsampling</i>	2
3.4	Desarrollo de las unidades de control	3
3.5	Desarrollo del algoritmo de clasificación	3
3.6	Desarrollo del sistema de integración de la arquitectura	3
4.	Verificación y validación	4
4.1	Pruebas de las capas: convolución, subsampling y densa	1
4.2	Pruebas integrales del sistema y validación completa del proceso de automatización	1
4.3	Ejecución y pruebas de los casos de uso	1
4.4	Evaluar el cumplimiento de los requerimientos	1
5.	Proceso de cierre	6
5.1	Elaboración del documento del SW de generación automática	1
5.2	Elaboración del documento del sistema generado automáticamente	1
5.3	Elaboración del informe de avance del proyecto	1
5.4	Elaboración del informe de final del proyecto	2
5.5	Elaboración de la presentación de la defensa	1



9. Matriz de uso de recursos de materiales

Código EDT	Nombre actividad	Recursos Requeridos (Horas)	
		Computador	FPGA
1.	Planificación general del proyecto	40	0
2.	Revisión bibliográfica y especificaciones	80	0
3.	Desarrollo de red neuronal automatizada	310	310
4.	Verificación y validación	70	70
5.	Proceso de cierre	100	0

10. Presupuesto detallado del proyecto

COSTOS DIRECTOS			
Descripción	Cantidad	Valor Unitario	Valor total
PYNQ Z2 FPGA	1	\$ 123 USD	\$ 123 USD
Subtotal			\$ 123 USD
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor Unitario	Valor total
20 % del costo directo	1	\$ 24,6 USD	\$ 24,6 USD
Subtotal			\$ 24,6 USD
TOTAL			\$ 147,6 USD

11. Matriz de asignación de responsabilidades

Código WBS	Nombre de la actividad	Lista de todos los nombres del proyecto		
		Responsable Ing. José David Alvarado Moreno	Orientador Ing. Federico G. Zacchigna	Cliente Ing. Federico G. Zacchigna
1.	Planificación general del proyecto	P	I	I
2.	Revisión bibliográfica y especificaciones	P	I	I
3.	Desarrollo de red neuronal automatizada	P	A	I
4.	Verificación y validación	P	A	I
5.	Proceso de cierre	P	A	I

Referencias:

- P = Responsabilidad Primaria
- S = Responsabilidad Secundaria
- A = Aprobación
- I = Informado
- C = Consultado

12. Gestión de riesgos

A continuación se relacionan los riesgos identificados y el análisis efectuado sobre los mismos:

Riesgo 1: La FPGA seleccionada no tiene los recursos de hardware mínimos para implementar la red neuronal.

- Severidad (10): No es posible realizar la implementación en la FPGA, lo cual no permitirá cumplir los objetivos del proyecto.
- Probabilidad de ocurrencia (5): Las redes neuronales requieren un número significativo de operaciones de suma y multiplicación por lo que es posible que el dispositivo utilizado no tenga los recursos de hardware mínimos para realizar la implementación.

Riesgo 2: El código HDL generado por la herramienta de alto nivel no es posible sintetizarlo para la implementación en la FPGA.

- Severidad (5): Las herramientas de alto nivel cuando generan el código HDL es posible que no se pueda sintetizar debido a la arquitectura de la FPGA.
- Probabilidad de ocurrencia (3): Se espera que los ajustes que se deben realizar sobre el código VHDL sean mínimos debido a la robustez de este tipo de herramientas.

Riesgo 3: Obtener porcentajes de acierto bajos al realizar la prueba GR debido a el parámetro definido para la cuantización.

- Severidad (3): El parámetro de la cuantización afecta los resultados obtenidos comparados con el modelo computación, el cual se puede ajustar para reducir los recursos de hardware o para mejorar el nivel de similitud con el modelo computación.
- Probabilidad de ocurrencia (2): Debido a que es un parámetro que se puede definir por el usuario el efecto se puede ajustar dependiendo de lo que se desea mejorar.

Riesgo 4: Retrasos de tiempo en el desarrollo de las actividades.

- Severidad (8): El retraso en el desarrollo de las actividades afecta directamente la fecha de entrega propuesta para finalizar el proyecto.
- Probabilidad de ocurrencia (2): Se espera que se pueda cumplir con las horas de trabajo programadas para cada semana.

Riesgo 5: Fallas en la placa TUL PYNQ-Z2 (ZYNQ XC7Z020-1CLG400C).

- Severidad (10): No se puede evaluar el funcionamiento de la herramienta propuesta.
- Probabilidad de ocurrencia (1): Se debe manipular con precaución para realizar las pruebas.

Tabla de gestión de riesgos (El RPN se calcula como $RPN=S \times O$) *

Riesgo	S	O	RPN	S*	O*	RPN*
1	10	5	50	6	3	18
2	5	3	15			
3	3	2	6			
4	8	2	16			
5	10	1	10			

Criterio adoptado:

Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 20.

Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

Riesgo 1.

- Plan de mitigación: Se evaluarán diferentes herramientas de alto nivel para realizar la codificación en VHDL y reducir significativamente la cantidad de recursos lógicos necesarios para la implementación. Adicionalmente se realizarán asesorías con el director y se consultara bibliografía especializada complementaria.
- Severidad (6): Se disminuye el riesgo por posibilidad de diferentes alternativas de implementación.
- Probabilidad de ocurrencia (3): Con mayor documentación se reduce la posibilidad de que se presente este inconveniente.

13. Gestión de la calidad

Para cada uno de los requerimientos del proyecto se presenta el procedimiento de verificación y validación:

Req #1: La herramienta debe permitir que los datos de entrada y salida en la capa de convolución se realicen por medio de un interfaz de *streaming* de datos.

- Verificación: Se comprobará que la interfaz de *streaming* de datos para la capa convolucional cumpla los requisitos funcionales.
- Validación: Se realizará pruebas de funcionamiento de la interfaz.

Req #2: La herramienta debe permitir cargar los valores de los pesos y el tamaño del kernel.

- Verificación: Se comprobará que se puedan cargar los valores de los pesos y sea posible ajustar el tamaño del kernel.
- Validación: Se realizará pruebas de funcionamiento de cargar los pesos y el tamaño del kernel.

Nota: los valores marcados con () en la tabla corresponden luego de haber aplicado la mitigación.

Req #3: El sistema debe realizar la convolución para procesar la información de la capa anterior.

- Verificación: Se comprobará que se pueda realizar la convolución para los parámetros establecidos.
- Validación: Se realizará pruebas de funcionamiento de la convolución.

Req #4: La herramienta debe permitir que los datos de entrada y salida en la capa de *subsampling* se realicen por medio de un interfaz *streaming* de datos.

- Verificación: Se comprobará que la interfaz de *streaming* de datos para la capa *subsampling* cumpla los requisitos funcionales.
- Validación: Se realizará pruebas de funcionamiento de la interfaz.

Req #5: La herramienta debe permitir seleccionar el método de *subsampling*.

- Verificación: Se comprobará que se puedan seleccionar el método de *subsampling*.
- Validación: Se realizará pruebas de funcionamiento del método de *subsampling*.

Req #6: El sistema debe realizar la *subsampling* para procesar la información de la capa anterior.

- Verificación: Se comprobará que se pueda realizar el *subsampling* para los parámetros establecidos.
- Validación: Se realizará pruebas de funcionamiento del método de *subsampling*.

Req #7: La herramienta debe permitir que los datos de entrada y salida en la capa de densa se realicen por medio de un interfaz *destreaming* de datos.

- Verificación: Se comprobará que la interfaz de *streaming* de datos para la capa densa cumpla los requisitos funcionales.
- Validación: Se realizará pruebas de funcionamiento de la interfaz.

Req #8: El sistema debe realizar la clasificación para procesar la información de la capa anterior.

- Verificación: Se comprobará que se pueda realizar la capa densa para los parámetros establecidos.
- Validación: Se realizará pruebas de funcionamiento de la capa densa.

Req #9: La herramienta debe tomar como entrada un modelo de una red neuronal de alto nivel (Keras).

- Verificación: Se comprobará que se pueda generar los parámetros de la red neuronal de alto nivel.
- Validación: Se realizará pruebas de funcionamiento del modelo de red neuronal de alto nivel.

Req #10: La herramienta debe generar un modelo de la arquitectura digital del alto nivel.

- Verificación: Se comprobará que se pueda generar la arquitectura de alto nivel.
- Validación: Se realizará pruebas de funcionamiento del modelo de la arquitectura de alto nivel.

Req #11: La herramienta debe generar un modelo de la arquitectura de alto nivel a partir del modelo cuantizado.

- Verificación: Se comprobará que se pueda generar la arquitectura de alto nivel con un modelo cuantizado.
- Validación: Se realizará pruebas de funcionamiento la arquitectura de alto nivel con un modelo cuantizado.

Req #12: La herramienta debe generar una *Golden Reference* (GR) a partir de la arquitectura de alto nivel.

- Verificación: Se comprobará que se pueda obtener los datos de la *Golden Reference*.
- Validación: Se generarán reportes de los datos de *Golden Reference*.

Req #13: La herramienta debe generar el código HDL sintetizable a partir de la arquitectura de alto nivel.

- Verificación: Se comprobará que se pueda generar los códigos en VHDL sintetizable.
- Validación: Se realizará pruebas de funcionamiento de los códigos obtenidos de VHDL.

Req #14: La herramienta debe generar un *testbench* de simulación, que permita comparar los resultados de HW con los resultados de la GR.

- Verificación: Se comprobará que se estén los códigos para relizar el *testbench*.
- Validación: Se realizará la simulación del HW para comparar los resultados con la GR.

Req #15: La herramienta debe presentar un funcionamiento equivalente al modelo cuantizado.

- Verificación: Se comprobará que se puedan modificar en la herramienta de HW los parámetros definidos de cuantización.
- Validación: Se realizará las pruebas para comparar el modelo de HW con el GR.

Req #16: La herramienta debe utilizar un número predefinido de multiplicadores por cada etapa.

- Verificación: Se comprobará que se puedan modificar en la herramienta de HW el número de multiplicadores por cada capa.
- Validación: Se realizará pruebas para evaluar el comportamiento de la herramienta de HW.

Req #17: La herramienta debe permitir la configuración, control y monitoreo del sistema se realiza mediante un banco de registros.

- Verificación: Se comprobará que se pueda modificar los parámetros de configuración, control y monitoreo.
- Validación: Se realizará pruebas para evaluar el comportamiento de la herramienta de HW.

Req #18: Se debe realizar el documento de informe de avance.

- Verificación: Se elaborará el documento de informe avance.
- Validación: Se presentará el documento de informe avance para su revisión.

Req #19: Se debe realizar el documento de informe final.

- Verificación: Se elaborará el documento de informe final.
- Validación: Se presentará el documento de informe final para su revisión.

14. Comunicación del proyecto

El plan de comunicación del proyecto es el siguiente:

PLAN DE COMUNICACIÓN DEL PROYECTO					
¿Qué comunicar?	Audiencia	Propósito	Frecuencia	Método de comunicac.	Responsable
Plan de trabajo del proyecto	Director y demás interesados	Información a los interesados	Única vez	Correo electrónico y/o video llamada	Ing. José David Alvarado Moreno
Avances del proyecto	Director	Seguimiento y control	Mensual	Correo electrónico y/o video llamada	Ing. José David Alvarado Moreno
Informe de avance	Director y demás interesados	Seguimiento y control	Única vez	Correo electrónico y/o video llamada	Ing. José David Alvarado Moreno
Informe final	Director y demás interesados	Finalizar el proyecto	Única vez	Correo electrónico y/o video llamada	Ing. José David Alvarado Moreno
Presentación final	Director y demás interesados	Presentación final del proyecto	Única vez	Correo electrónico y/o video llamada	Ing. José David Alvarado Moreno

15. Gestión de compras

En el presente proyecto no se realizarán compras.

16. Seguimiento y control

El avance del proyecto se realizará en la siguiente escala:

- 0 % : Tarea sin iniciar.
- 25 % : Tarea iniciada.
- 50 % : Tarea en desarrollo.
- 75 % : Tarea finalizada sin documentación.
- 100 % : Tarea finalizada.

SEGUIMIENTO DE AVANCE					
Tarea del WBS	Indicador de avance	Frecuencia de reporte	Resp. de seguimiento	Persona a ser informada	Método de comunic.
1.	%	Única vez al comienzo	Ing. José David Alvarado Moreno	Ing. Federico G. Zacchigna	email o video llamada
2.	%	Mensual mientras dure la tarea	Ing. José David Alvarado Moreno	Ing. Federico G. Zacchigna	email o video llamada
3.	%	Mensual mientras dure la tarea	Ing. José David Alvarado Moreno	Ing. Federico G. Zacchigna	email o video llamada
4.	%	Mensual mientras dure la tarea	Ing. José David Alvarado Moreno	Ing. Federico G. Zacchigna	email o video llamada
5.	%	Única vez al finalizar	Ing. José David Alvarado Moreno	Ing. Federico G. Zacchigna	email o video llamada

17. Procesos de cierre

- Documentación final.
 - Responsable: Ing. José David Alvarado Moreno.
 - Elaboración de la documentación para finalizar el proyecto, en donde se pueda evidenciar los resultados obtenidos en función del plan de trabajo propuesto.
- Presentación del proyecto.
 - Responsable: Ing. José David Alvarado Moreno.
 - Realizar la presentación de los resultados del proyecto ante el director, jurados y demás interesados.
- Realizar acto de agradecimiento.
 - Responsable: Ing. José David Alvarado Moreno.
 - organizar un acto de agradecimiento para el director, jurados y demás interesados.