# DISEÑO DE SISTEMAS DIGITALES PROYECTO DE LABORATORIO

TEMA DE LA ACTIVIDAD: BEHAVIORAL SIGNAL PROCESSING WITH MACHINE LEARNING BASED ON FPGA

### **OBJETIVOS DE APRENDIZAJE:**

- Elaborar un sistema multi-núcleo utilizando dos microprocesadores Nios II, para la verificación de dos procesos simultáneos utilizando el acceso a memoria para compartición de variables y el acceso a puertos periféricos.
- Predecir el número de pasos de una persona a partir de una red neuronal previamente entrenada y proyectar los resultados en un monitor con puerto VGA.
- Diseñar programación en lenguaje C que ayude a la estimación de número de pasos.

**DURACIÓN:** 60 minutos

### **MATERIALES Y HERRAMIENTAS:**

- Software Quartus 17.0 versión Estándar.
- Software Eclipse Kepler.
- Tarjeta de desarrollo DE10-Standard.

## INTRODUCCIÓN

El proyecto se basa en un sistema embebido capaz de estimar el número de pasos que realizará el usuario en un día determinado, a partir de una red neuronal entrenada con los datos propios del usuario; para lo cual se ha implementado la tarjeta DE10 Standard con FPGA SoC.

Para la realización del proyecto, se utilizará el programa Quartus junto con sus herramientas Qsys y Eclipse que permitirá la creación tanto del software como del hardware respectivamente realizando la interconexión de los bloques que se detallan a continuación: Procesador Nios II, System ID, VGA controller, On chip Memory, Internal Timer, entre otros.

La propuesta cuenta con un sistema multinúcleos, el primer CPU está encargado del almacenamiento de la red neuronal y realiza la predicción del número de pasos a dar; siempre y cuando se ingrese por teclado los siguientes datos: día a predecir, mes a predecir, día de la semana, hora a predecir, temperatura promedio y pasos del día

anterior. El segundo CPU realizará la proyección de los datos a partir de un puerto VGA y un monitor.

Para el entrenamiento de la Red Neuronal utilizada en este trabajo, se realizó el conteo de pasos de un estudiante sano a través de una aplicación móvil y el podómetro incorporado en el Smartphone, dicha aplicación nos muestra el total de pasos en un día entero. De esta manera se logró obtener la cantidad de pasos dados en aproximadamente 2 meses (noviembre y diciembre).

**DESCRIPCION DE LA PRACTICA:** Inicialmente se debe añadir el archivo de Qsys al proyecto y compilarlo, en el que se encuentra la arquitectura de un sistema multi-nucleo. Se abre la plataforma de Eclipse y se añade la programación en lenguaje C, que contiene la red neuronal y predicción de número de pasos.

#### PROCEDIMIENTO:

- 1. Proceda a crear una carpeta en el escritorio con el nombre PROYECTO (escriba sin espacios). Luego copie y pegue todos los archivos que se encuentran en la carpeta compartida, en la carpeta que fue creada en el escritorio.
- 2. Ejecute el programa Quartus Prime 17.0 Standard Edition que se encuentra en el escritorio representado por el ícono.
- Proceda a crear un nuevo Proyecto seleccionando la opción File→New Project Wizard como se observa en la ilustración 1. Aparecerá la ventana de Introducción, luego da clic en Next.

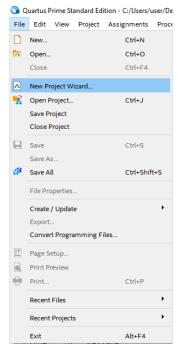


Ilustración 1: Creación de un nuevo proyecto

4. En la primera línea escoja la ruta donde va a crear su proyecto, en este caso, la ruta será la carpeta PROYECTO que fue creada en el procedimiento 1 y que se encuentra en el escritorio. En la siguiente línea escoge como nombre del proyecto PROYECTO, tal como se visualiza en la ilustración 2. Luego da clic en Next.

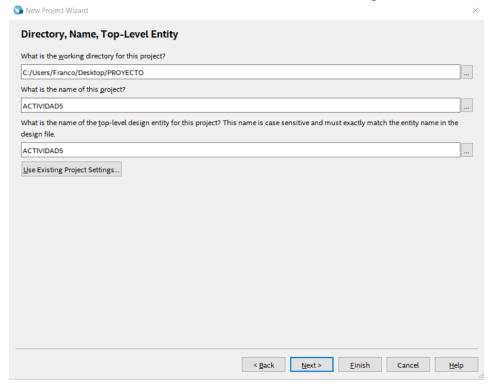


Ilustración 2: Nombre del proyecto

 Se mostrará la ventana para seleccionar el tipo de proyecto. Se escogerá la opción de Proyecto vacío (Empty project), tal como se visualiza en la ilustración 3. Luego da clic en Next.

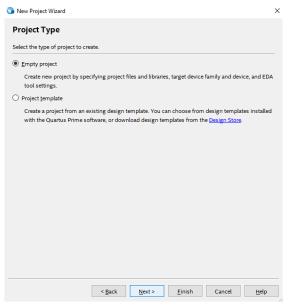


Ilustración 3: Tipo de proyecto

6. Se mostrará la ventana para adhesión de archivos. En esa ventana seleccione la opción Add All, para que se agreguen todos los archivos que fueron copiados a la carpeta del proyecto y que se realizó en el procedimiento 1. En la ilustración 4 se observan los archivos añadidos. Por último, haga clic en Next.

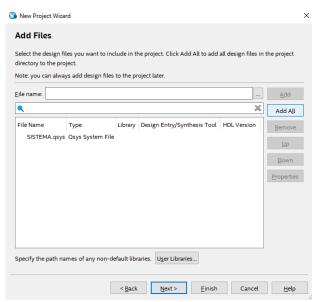


Ilustración 4: Agregar archivos al proyecto

7. En la siguiente ventana aparecerán todas las familias de chips FPGA. Proceda a escoger la familia Cyclone V (E/GX/GT/SX/SE/ST) y seleccione el nombre del

chip FPGA SoC **5CSXFC6D6F31C6**, tal como se muestra en la ilustración 5. Luego haga clic en **Next**.

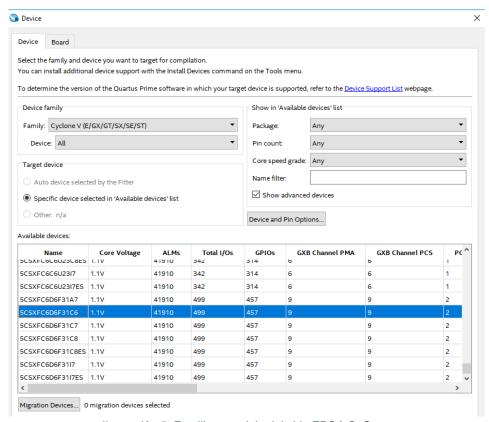


Ilustración 5: Familia y modelo del chip FPGA SoC

8. Debido a que no se realizarán simulaciones en la presente actividad, no se seleccionará alguna herramienta de simulación como se muestra en la ilustración6. Haga clic en Next.

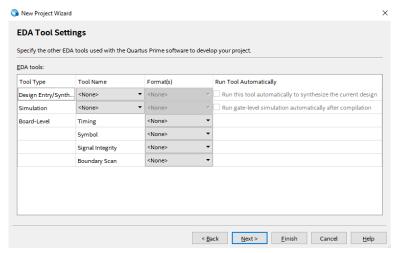


Ilustración 6: Herramientas para simulación.

0	Al finalizar la graggión del proyecto, appropará la ventona <b>Summary</b> para regumin
9.	Al finalizar la creación del proyecto, aparecerá la ventana <b>Summary</b> para resumir todas las especificaciones que tendrá nuestro proyecto. Para finalizar haga clic en <b>Finish</b> .

10. Haga doble clic al archivo **SISTEMA.qsys** dentro del programa **Quartus Prime**. Verifique que se encuentra las siguientes conexiones entre cada componente, como se muestra en la ilustración 7.

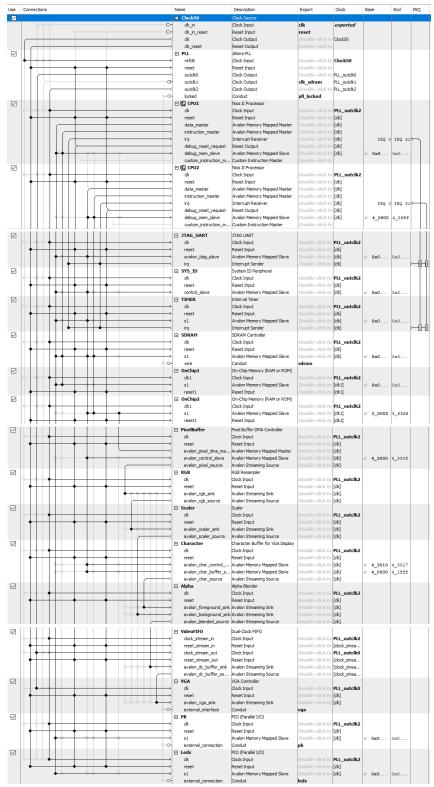


Ilustración 7: Diseño del Qsys realizado

11. Haga clic en **Generate HDL** que se encuentra en la parte inferior derecha, tal como se muestra en la ilustración 8.

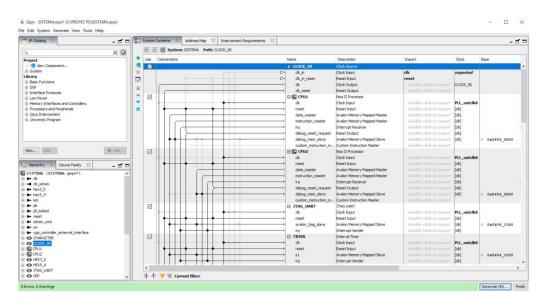


Ilustración 8: Herramienta de Qsys.

12. En la ventana **Generation**, seleccione la creación de archivos VHDL en la sección **Synthesis**, tal como se muestra en la ilustración 9. Por último, haga clic en **Generate**, y cuando finalice haga clic en **Finish**.

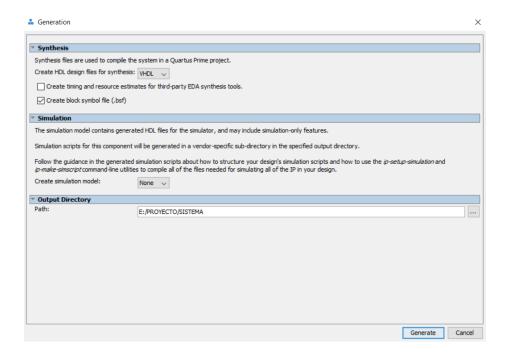


Ilustración 9: Ventana Generation del Qsys

13. Proceda a compilar el archivo **SISTEMA.qsys** estableciéndolo previamente como más alta jerarquía (dando clic derecho en el mismo y seleccionando la opción **Set as Top-Level Entity**), como se muestra en la ilustración 10. Luego seleccione el ícono.

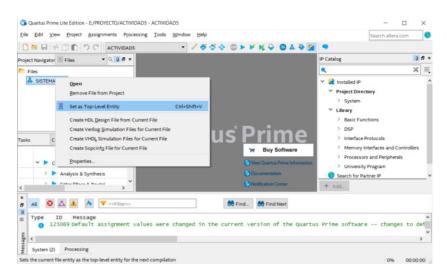


Ilustración 10: Compilación del proyecto

14. A continuación, se procede a la asignación de pines. En la ventada de Assigments elegir la opción de Pin Planner, como se muestra en la ilustración 11.

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength	Slew Rate	Differential Pair	er Analog Setting	GXB/VCCT_GXB	siver I/O Pin Term	edicate
altera_reserved_tck altera_reserved_tdi	Input				PIN_AC5 PIN_U8	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)						
altera_reserved_tdo altera_reserved_tms	Output				PIN_AB9 PIN V9	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default)					
clk_clk	Input	PIN_AF14	3B	B3B_N0	PIN_AF14	3.3-V LVTTL		16mA (default)						
clk_sdram_clk	Output	PIN_AH12 PIN_AD20	38 4A	83B_N0 84A_N0	PIN_AH12 PIN_AD20	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[27] hex3_0_export[26]	Output	PIN_AA19	4A	B4A_NO	PIN_AA19	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[25]	Output	PIN_AC20	4A 4A	B4A_NO B4A_NO	PIN_AC20	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[24] hex3_0_export[23]	Output	PIN_AA20 PIN_AD19	4A 4A	B4A_NO B4A_NO	PIN_AA20 PIN_AD19	3.3-V LVTTL 3.3-V LVTTL		16mA (default)	1 (default) 1 (default)					
hex3_0_export[22]	Output	PIN_W19	4A	B4A_NO	PIN_W19	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[21] hex3_0_export[20]	Output	PIN_Y19 PIN_W16	4A 4A	B4A_NO B4A_NO	PIN_Y19 PIN_W16	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[19]	Output	PIN_AF18	4A	B4A_NO	PIN_AF18	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[18] hex3_0_export[17]	Output	PIN_Y18 PIN_Y17	4A 4A	B4A_NO B4A_NO	PIN_Y18 PIN_Y17	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[16]	Output	PIN_AA18	4A	B4A_NO	PIN_AA18	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[15] hex3_0_export[14]	Output	PIN_AB17 PIN_AA21	4A 4A	84A_N0 84A_N0	PIN_AB17 PIN_AA21	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[13]	Output	PIN_V17	4A	B4A_NO	PIN_V17	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[12] hex3_0_export[11]	Output	PIN_AE17 PIN_AE18	4A 4A	B4A_NO B4A_NO	PIN_AE17 PIN_AE18	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[10]	Output	PIN_AD17	4A	B4A_NO	PIN_AD17	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[9] hex3_0_export[8]	Output	PIN_AE16 PIN V16	4A 4A	B4A_NO B4A_NO	PIN_AE16 PIN_V16	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[7]	Output	PIN_AF16	4A	B4A_N0	PIN_AF16	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[6] hex3_0_export[5]	Output	PIN_AH18 PIN AG18	4A 4A	B4A_NO B4A_NO	PIN_AH18 PIN AG18	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[4]	Output	PIN_AH17	4A	B4A_N0	PIN_AH17	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[3] hex3_0_export[2]	Output	PIN_AG16 PIN_AG17	4A 4A	B4A_NO B4A_NO	PIN_AG16 PIN_AG17	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex3_0_export[1]	Output	PIN_V18	4A	B4A_NO	PIN_V18	3.3-V LVTTL		16mA (default)	1 (default)					
hex3_0_export[0] hex5_4_export[13]	Output Output	PIN_W17 PIN_AB21	4A 4A	B4A_NO B4A_NO	PIN_W17 PIN_AB21	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex5_4_export[12]	Output	PIN_AF19	4A	B4A_NO	PIN_AF19	3.3-V LVTTL		16mA (default)	1 (default)					
hex5_4_export[11] hex5_4_export[10]	Output	PIN_AE19 PIN_AG20	4A 4A	B4A_NO B4A_NO	PIN_AE19 PIN_AG20	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex5_4_export[9]	Output	PIN_AF20	4A	B4A_N0	PIN_AF20	3.3-V LVTTL		16mA (default)	1 (default)					
hex5_4_export[8] hex5_4_export[7]	Output	PIN_AG21 PIN_AF21	4A 4A	B4A_N0 B4A_N0	PIN_AG21 PIN_AF21	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex5_4_export[6]	Output	PIN_AH22	4A	B4A_N0	PIN_AH22	3.3-V LVTTL		16mA (default)	1 (default)					
hex5_4_export[5]	Output	PIN_AF23 PIN_AG23	4A 4A	B4A_NO B4A_NO	PIN_AF23 PIN_AG23	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex5_4_export[4] hex5_4_export[3]	Output	PIN_AE23	4A	B4A_NO	PIN_AE23	3.3-V LVTTL		16mA (default)	1 (default)					
hex5_4_export[2]	Output	PIN_AE22	4A 4A	B4A_NO B4A_NO	PIN_AE22	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
hex5_4_export[1] hex5_4_export[0]	Output Output	PIN_AG22 PIN_AD21	4A	B4A_NO	PIN_AG22 PIN_AD21	3.3-V LVTTL		16mA (default)	1 (default)					
led_export[9] led_export[8]	Output	PIN_AC22 PIN_AB22	4A 5A	B4A_NO B5A_NO	PIN_AC22 PIN_AB22	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
led_export[8] led_export[7]	Output	PIN_AF24	4A	B4A_NO	PIN_AF24	3.3-V LVTTL		16mA (default)	1 (default)					
led_export[6]	Output	PIN_AE24 PIN_AF25	4A 4A	B4A_NO B4A_NO	PIN_AE24	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
led_export[5] led_export[4]	Output Output	PIN_AG25	4A 4A	B4A_NO	PIN_AF25 PIN_AG25	3.3-V LVTTL		16mA (default)	1 (default) 1 (default)					
led_export[3]	Output	PIN_AD24	4A 4A	B4A_NO	PIN_AD24	3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
led_export[2] led_export[1]	Output	PIN_AC23 PIN_AB23	5A	B4A_N0 B5A_N0	PIN_AC23 PIN_AB23	3.3-V LVTTL 3.3-V LVTTL		16mA (default)	1 (default) 1 (default)					
led_export[0]	Output	PIN_AA24	5A	B5A_N0	PIN_AA24	3.3-V LVTTL		16mA (default)	1 (default)					
pb_export[2] pb_export[1]	Input	PIN_AA15 PIN_AA14	3B 3B	B3B_NO B3B_NO	PIN_AA15 PIN_AA14	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)						
pb_export[0]	Input	PIN_AK4	38	B3B_N0	PIN_AK4	3.3-V LVTTL		16mA (default)						
pll_locked_export reset_reset_n	Output	PIN AJ4	38	B3B NO	PIN_AE7 PIN AJ4	3.3-V Lefault) 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default)					
sdram_wire_addr[12]	Output	PIN_AJ14	3B	B3B_NO	PIN_AJ14	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_addr[11] sdram_wire_addr[10]	Output	PIN_AH13 PIN_AG12	3B 3B	B3B_N0 B3B_N0	PIN_AH13 PIN_AG12	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_addr[9]	Output	PIN_AG13	38	B3B_NO	PIN_AG13	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_addr[8] sdram_wire_addr[7]	Output	PIN_AH15 PIN_AF15	38	B3B_NO B3B_NO	PIN_AH15 PIN_AF15	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_addr[6]	Output	PIN_AD14	38	B3B_N0	PIN_AD14	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_addr[5] sdram_wire_addr[4]	Output	PIN_AC14 PIN_AB15	38 38	B3B_NO B3B_NO	PIN_AC14 PIN_AB15	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_addr[3]	Output	PIN_AE14	38	83B_N0	PIN_AE14	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_addr[2] sdram_wire_addr[1]	Output	PIN_AG15 PIN_AH14	3B 3B	B3B_N0 B3B_N0	PIN_AG15 PIN_AH14	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_addr[0]	Output	PIN_AK14	38	B3B_N0	PIN_AK14	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_ba[1] sdram_wire_ba[0]	Output	PIN_AJ12 PIN_AF13	38	838_NO 838_NO	PIN_AJ12 PIN_AF13	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_cas_n	Output	PIN_AF11	38	838_NO	PIN_AF11	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_cke	Output	PIN_AK13 PIN_AG11	3B 3B	838_NO 838_NO	PIN_AK13 PIN AG11	3.3-V LVTTL 3.3-V LVTTL		16mA (default)	1 (default) 1 (default)					
sdram_wire_cs_n sdram_wire_dq[15]	Output Bidir	PIN_AG11 PIN_AJ5	38	B3B_N0 B3B_N0	PIN_AG11 PIN_AJ5	3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_dq[14]	Bidir	PIN_AJ6	38	B3B_N0	PIN_AJ6	3.3-V LVTTL 3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_dq[13] sdram_wire_dq[12]	Bidir	PIN_AH7 PIN_AH8	38	838_NO 838_NO	PIN_AH7 PIN_AH8	3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_dq[11]	Bidir	PIN_AH9	38	83B_N0	PIN_AH9	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_dq[10] sdram_wire_dq[9]	Bidir Bidir	PIN_AJ9 PIN_AJ10	38 38	838_NO 838_NO	PIN_AJ9 PIN_AJ10	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_dq[8]	Bidir	PIN_AH10	38	83B_N0	PIN_AH10	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_dq[7] sdram_wire_dq[6]	Bidir Bidir	PIN_AJ11 PIN_AK11	3B 3B	B3B_N0 B3B_N0	PIN_AJ11 PIN_AK11	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_dq[5]	Bidir	PIN_AG10	38	B3B_N0	PIN_AG10	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_dq[4] sdram_wire_dq[3]	Bidir Bidir	PIN_AK9 PIN_AK8	38 38	B3B_N0 B3B_N0	PIN_AK9 PIN_AK8	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_dq[2]	Bidir	PIN_AK7	38	838_NO	PIN_AK7	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_dq[1] sdram_wire_dq[0]	Bidir	PIN_AJ7 PIN AK6	3B 3B	838_NO 838_NO	PIN_AJ7 PIN AK6	3.3-V LVTTL		16mA (default)	1 (default) 1 (default)					
sdram_wire_dqm[1]	Output	PIN_AK12	3B	B3B_N0	PIN_AK12	3.3-V LVTTL		16mA (default)	1 (default)					
sdram_wire_dqm[0] sdram_wire_ras_n	Output Output	PIN_AB13 PIN_AE13	38 38	838_NO 838 NO	PIN_AB13 PIN_AE13	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)	1 (default) 1 (default)					
sdram_wire_we_n	Output	PIN_AA13	38	B3B_NO	PIN_AA13	3.3-V LVTTL		16mA (default)	1 (default) 1 (default)					
sw_export[9] sw_export[8]	Input	PIN_AA30 PIN_AC29	5B 5B	B5B_N0 B5B_N0	PIN_AA30 PIN_AC29	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)						
sw_export[7]	Input	PIN_AD30	5B	B5B_NO	PIN_AD30	3.3-V LVTTL		16mA (default)						
sw_export[6]	Input	PIN_AC28	58 58	858_NO 858_NO	PIN_AC28 PIN_V25	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)						
sw_export[5] sw_export[4]	Input	PIN_V25 PIN_W25	58	B5B_NO	PIN_W25	3.3-V LVTTL		16mA (default)						
sw_export[3]	Input	PIN_AC30 PIN_AB28	58 58	858_NO 858_NO	PIN_AC30 PIN_AB28	3.3-V LVTTL 3.3-V LVTTL		16mA (default) 16mA (default)						
sw_export[2] sw_export[1]	Input	PIN_Y27	58	B5B_N0	PIN_Y27	3.3-V LVTTL		16mA (default)						
sw_export[0]	Input	PIN_AB30	58 4A	B5B_NO	PIN_AB30	3.3-V LVTTL		16mA (default)	1 (default)					
/ga_contrface_B[7] /ga_contrface_B[6]	Output	PIN_AK16 PIN_AJ16	4A	B4A_N0 B4A_N0	PIN_AJ16 PIN_AJ20	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default)					
/ga_contrface_B[5]	Output	PIN_AJ17	4A	B4A_N0	PIN_AK18	3.3-V Lefault)		16mA (default)	1 (default)					
/ga_contrface_B[4] /ga_contrface_B[3]	Output Output	PIN_AH19 PIN_AJ19	4A 4A	B4A_NO B4A_NO	PIN_AD25 PIN_AH25	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default) 1 (default)					
vga_contrface_B[2]	Output	PIN_AH20	4A	B4A_NO	PIN_AJ19	3.3-V Lefault)		16mA (default)	1 (default)					
vga_contrface_B[1] vga_contrface_B[0]	Output	PIN_AJ20 PIN_AJ21	4A 4A	B4A_NO B4A_NO	PIN_AH19 PIN_AJ21	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default) 1 (default)					
vga_contce_BLANK	Output	PIN_AK22	4A	B4A_N0	PIN_AH20	3.3-V Lefault)		16mA (default)	1 (default)					
vga_contrface_CLK vga_contrface_G[7]	Output	PIN_AK21 PIN_AH23	4A 4A	B4A_NO B4A_NO	PIN_A13 PIN AK19	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default) 1 (default)					
vga_contrface_G[7] vga_contrface_G[6]	Output	PIN_AK23	4A 4A	B4A_NO	PIN_AC18	3.3-V Lefault)		16mA (default)	1 (default) 1 (default)					
vga_contrface_G[5]	Output	PIN_AH24	4A	B4A_NO	PIN_AK24	3.3-V Lefault)		16mA (default)	1 (default) 1 (default)					
	Output	PIN_AJ24 PIN_AK24	4A 4A	B4A_NO B4A_NO	PIN_AJ24 PIN_AJ26	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default) 1 (default)					
vga_contrface_G[4] vga_contrface_G[3]	Output				PIN AK23	3.3-V Lefault)		16mA (default)	1 (default)					
vga_contrface_G[4] vga_contrface_G[3] vga_contrface_G[2]	Output	PIN_AH25	4A	B4A_NO					a felicity and					
vga_contrface_G[4] vga_contrface_G[3] vga_contrface_G[2] vga_contrface_G[1] vga_contrface_G[0]		PIN_AH25 PIN_AJ25 PIN_AK26	4A 4A 4A	84A_NO 84A_NO 84A_NO	PIN_AJ25 PIN_Y16	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default) 1 (default)					
vga_contrface_G[4] vga_contrface_G[3] vga_contrface_G[2] vga_contrface_G[1] vga_contrface_G[0] vga_contrerface_HS	Output Output Output Output	PIN_AJ25 PIN_AK26 PIN_AK19	4A 4A 4A	B4A_NO B4A_NO B4A_NO	PIN_AJ25 PIN_Y16 PIN_AJ27	3.3-V Lefault) 3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default) 16mA (default)	1 (default) 1 (default)					
vga_contrface_G[4]	Output Output Output	PIN_AJ25 PIN_AK26	4A 4A	B4A_NO B4A_NO	PIN_AJ25 PIN_Y16	3.3-V Lefault) 3.3-V Lefault)		16mA (default) 16mA (default)	1 (default)					

Ilustración 11: Asignación de pines

- 15. Nuevamente, proceda a compilar el archivo **SISTEMA.qsys** estableciéndolo previamente como más alta jerarquía (dando clic derecho en el mismo y seleccionando la opción **Set as Top-Level Entity** y seleccione el ícono.
- 16. Proceda a conectar la tarjeta de desarrollo **DE10-Standard** tanto a la PC (cable **USB-Blaster**) como a la fuente de alimentación.
- 17. Una vez que se haya completado la compilación, diríjase al ícono que se encuentra en la barra de tareas del programa **Quartus Prime**, y servirá para programar la tarjeta de desarrollo **DE10-Standard**.
- 18. En la ventana Programmer se visualizará un archivo, haga clic derecho en el mismo y elimínelo escogiendo la opción Delete, tal como se visualiza en la ilustración 12.

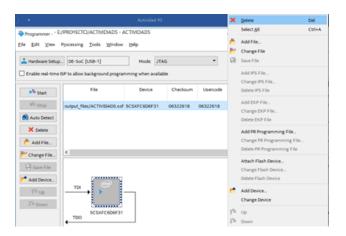


Ilustración 12: Ventana Programmer

19. Una vez eliminado el archivo y que no se encuentre ningún otro archivo en la ventana Programmer, proceda a escoger la opción **Auto Detect** representada por la siguiente botonera Auto Detect. Seleccione la opción **5CSXFC6D6** tal como se muestra en la ilustración 13. Haga clic en OK.

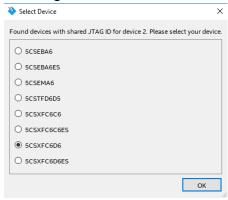


Ilustración 13: Ventana Auto Detect

20. Aparecerán dos archivos: SOCVHPS y 5CSXFC6D6. Haga clic derecho en el segundo y escoja la opción Change File tal como se muestra en la ilustración 14, y reemplácelo por el archivo ACTIVIDAD5.sof que se encuentra en la carpeta Output\_Files tal como se muestra en la ilustración 15.

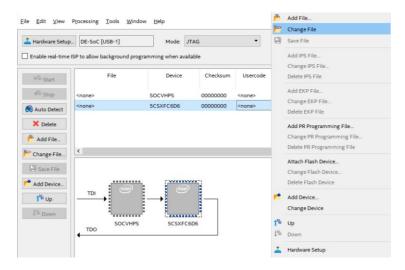


Ilustración 14: Cambio de archivo en la ventana Programmer.

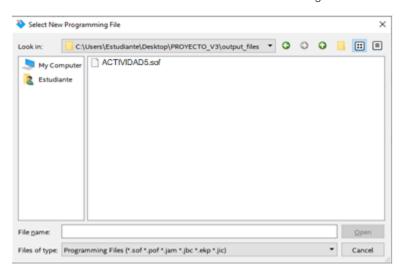


Ilustración 15:Elección del archivo. sof

21. Haga clic en la casilla correspondiente a la columna **Program/Configure** del archivo **ACTIVIDAD5.sof**, y luego seleccione **Start** para que se programe el chip FPGA. Para que la programación sea exitosa se debe de alcanzar el 100%, tal cual se muestra en la ilustración 16.

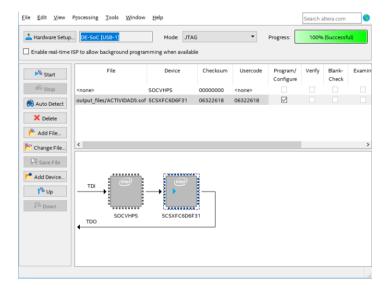


Ilustración 16: Programación exitosa del chip FPGA

- 22. En este momento se acaba de construir físicamente la arquitectura de computadora embebida con los 2 microprocesadores Nios II en el chip FPGA.
- 23. Realizamos la programacion del microprocesador Nios II utilizando lenguaje C. En menu **Tools**, seleccione la opcion **Nios II Software Build Tools for Eclipse**, tal como se muestra en la ilustracion 17.

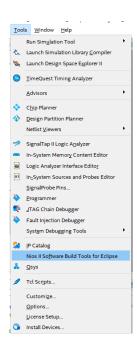


Ilustración 17 Selección de la herramienta Eclipse.

24. Seleccionamos la carpeta **SISTEMA** que se encuentra dentro de nuestra carpeta de proyecto para alojar el nuevo proyecto de Eclipse, comose muestra en la ilustracion 18.

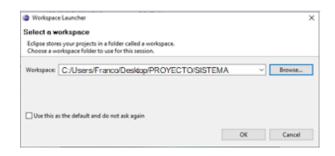


Ilustración 18: Ubicación del proyecto en la plataforma Eclipse

25. Inmediatamente se abrira el entorno de programacion Eclipse, procedemos a crear un nuevo proyecto, seleccionamos File → New→ Nios II Application and BSP from Template tal como se ve en la ilustracion 19.

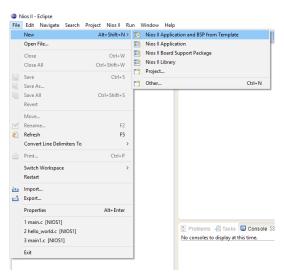


Ilustración 19:Selección de una nueva Aplicación para Nios II.

26. En la opción **SOPC Information File name**, buscamos el archivo **Sistema.sopcinfo** que tendrá toda la información de la computadora embebida, como se muestra en la ilustración 20.

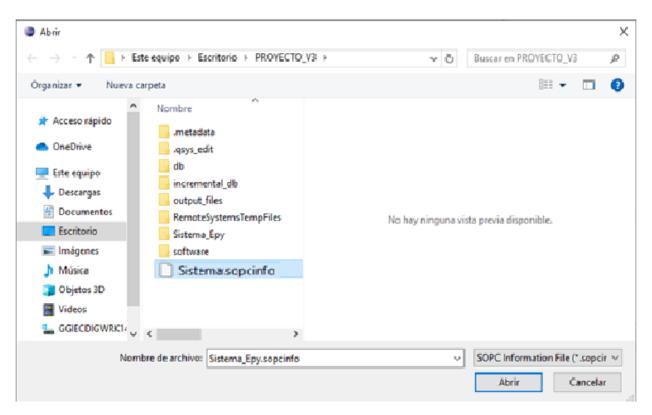


Ilustración 20: Elección del archivo .sopcinfo

27. En la opción **CPU name** escogemos el núcleo CPU1 disponibles en la computadora embebida. En **Project name** elegimos el nombre del proyecto, utilice el nombre **NIOS1**. En **Project Template** seleccionamos la opción **Hello World** y clic en **FINISH**, tal como se muestra en la ilustración 21.

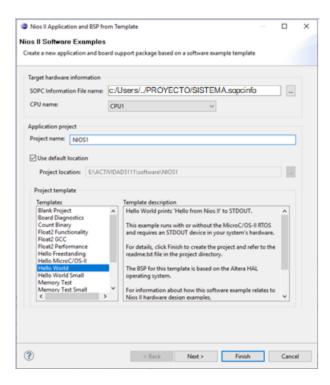


Ilustración 21 Creación de la aplicación del primer núcleo en software.

28. Inmediatamente se abrira el entorno de programacion Eclipse, procedemos a crear un nuevo proyecto, seleccionamos File → New→ Nios II Application and BSP from Template tal como se ve en la ilustracion 22.

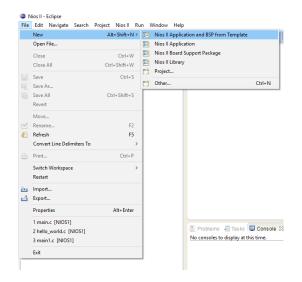


Ilustración 22: Selección de una nueva Aplicación para Nios II.

- 29. En la opción **SOPC Information File name**, buscamos el archivo **Sistema.sopcinfo** que tendrá toda la información de la computadora embebida.
- 30. En la opción **CPU name** escogemos el núcleo CPU2 disponibles en la computadora embebida. En **Project name** elegimos el nombre del proyecto, utilice el nombre **NIOS2**. En **Project Template** seleccionamos la opción **Hello World** y clic en **FINISH**, tal como se muestra en la ilustración 23.

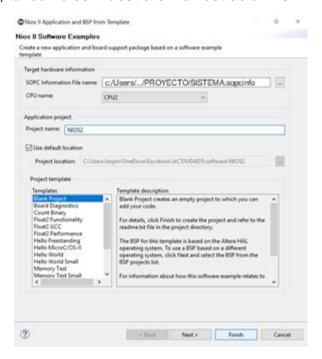


Ilustración 23 Creación de la aplicación del primer núcleo en software.

31. En la subventana de **Projetc Explorer** visualizamos 4 carpetas: Nios1 Nios2 y Nios1\_bsp y Nios2\_bsp. En la carpeta de NIOS1 y Nios2 podemos crear los archivos de programacion en lenguaje C/C++ para nuestros microprocesadores, y las carpetas restante contiene todos los componentes del hardware como se observa en la ilustracion 24.



Ilustración 24: Ventana del navegador de proyectos.

32. Proceda a hacer clic derecho a la carpeta **NIOS1\_bsp[SISTEMA**] y seleccione la opcion **Properties**, tal como se muestra en la ilustracion 25

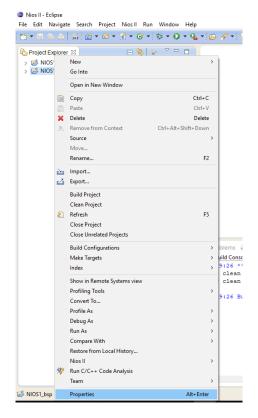


Ilustración 25: Propiedades del archivo NIOS1\_bsp.

33. Inmediatamente se abrirá la ventana de propiedades, diríjase a la categoría Nios II BSP Properties, y deshabilite la casilla Support C++ y habilite la casilla Reduce device drives. Tal como se muestra en la ilustración 26. Luego dar clic en OK.

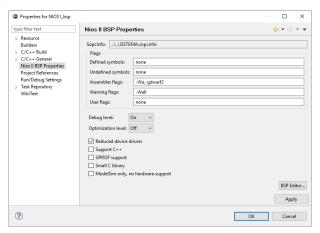


Ilustración 26: Selección para librerías reducidas de la carpeta NIOS1\_bsp.

- 34. Copiamos la programación del archivo **RedNeuronal.txt** que se encuentra en la carpeta del proyecto, en el archivo **hellow\_world.c** de la carpeta **NIOS1**. Asimismo, copiamos la programación del archivo **ProyeccionVGA.txt** que se encuentra en la carpeta del proyecto, en el archivo **hellow\_world.c** de la carperta **NIOS2**.
- 35. Proceda a hacer clic derecho a la carpeta NIOS1\_bsp[SISTEMA], seleccione la opción NIOS II y luego seleccione la opción BSP EDITOR, tal como se visualiza en la ilustración 27.

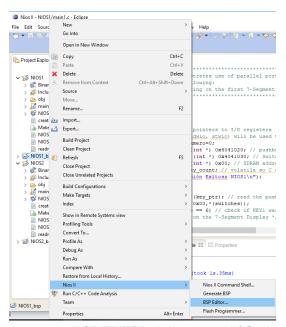


Ilustración 27: BSP EDITOR de la carpeta NIOS1\_bsp

36. Asegúrese que tanto **NIOS1** y **NIOS2** se encuentren ejecutando desde sus espacios de memoria, como se visualiza en la ilustración 28. En el caso de **NIOS1** se ejecuta desde **onchip\_mem1**, y **NIOS2** se ejecuta desde **onchip\_mem2**.

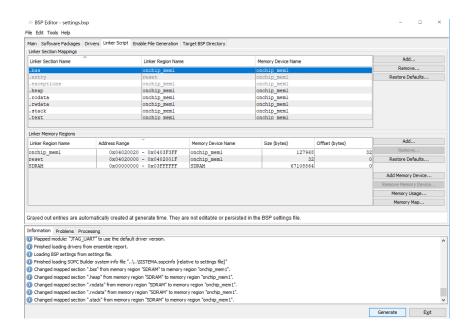


Ilustración 28: Revisión de los espacios de memoria de cada procesador.

37. Luego, se procede a guardar y compilar todo el sistema nuevamente.



Ilustración 29: Compilación de todo el sistema

38. Inicialmente, se ejecutará el procesador **NIOS1** que contiene la red neuronal con la predicción de pasos. Ingrese los datos, en **Nios II Console,** correspondientes a los pasos a predecir por la Red Neuronal y observe los resultados, tal como se muestra en la ilustración 30.

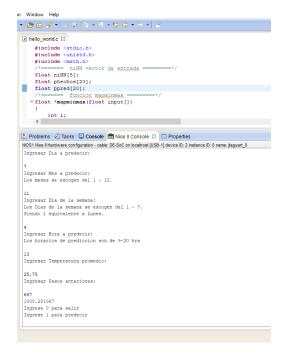


Ilustración 30: Salida por consola del programa Eclipse.

39. Luego, con los datos obtenidos, se procede a graficar los pasos reales vs los pasos predichos; para lo cual se compila y ejecuta el procesador **NIOS2**, tal como se muestra en la ilustración 31.

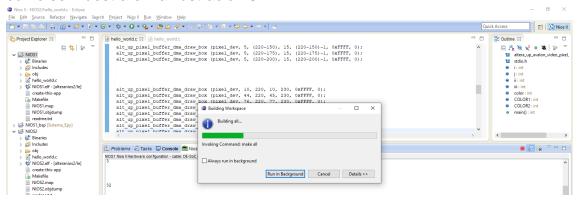


Ilustración 31: Ejecución del NIOS2 para la proyección en VGA

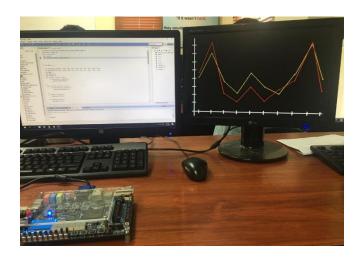


Ilustración 32: Proyección de datos estadísticos por puerto VGA.