**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**DISEÑO DE SISTEMA DIGITALES**

**Documentación Técnica Del Proyecto**



**TEMA:** PROCESAMIENTO DE SEÑALES DE COMPORTAMIENTO CON APRENDIZAJE DE MÁQUINA BASADO EN UNA FPGA.

# OBJETIVOS:

* Realizar el entrenamiento de una red neuronal, con el objetivo de realiza predicciones del comportamiento de una persona, pasos que realiza, basado en aspectos de su entorno y actividades anteriores.
* Implementar un sistema multinúcleo que permita la ejecución de la red entrenada y además permita el procesamiento de esa información que luego será analizada.
* Construir la arquitectura del computador empleando los bloques IP necesarios, con el fin de poder enviar a imprimir por pantalla los resultados obtenidos y la comparación con los resultados esperados.
* Diseñar la programación en C requerida para el manejo y conexión de la arquitectura diseñada con el hardware o periféricos de la tarjeta de desarrollo DE10-Standerd.
* Realizar la configuración de los procesadores adecuada, para su comunicación y funcionamiento, incluyendo su modelo de construcción y lanzamiento mediante la herramienta Eclipse para la programación de los procesadores.

# MATERIALES Y HERRAMIENTAS:

* Software Quartus 16.1 versión Estándar.
* Software Eclipse Kepler.
* Tarjeta de desarrollo DE10-Standard.

# INTRODUCCIÓN:

Muchos sistemas requieren que varias tareas se ejecuten de forma paralela en diferentes procesadores o simplemente requieren de más de un procesador para poder ejecutar las funciones que les fueron asignadas. Esos sistemas se conocen con el nombre de sistemas multinúcleos; en los cuales se incluyen varios núcleos realizando tareas que generalmente tienen una relación entre sí.

El aprendizaje de máquina o “Machine Learning”, se puede entender como una técnica de análisis que brinda la capacidad a un sistema de aprender patrones, comportamientos, relaciones entre datos que recibe y así poder analizarlos y realizar tareas de clasificación, predicción u toma de decisiones; todo sin ningún tipo de intervención humana. Las redes neuronales se aplican en el “Machine Learning” y son la base de este aprendizaje. Las redes se encargan de recibir datos y realiza una serie de operaciones y funciones internas para obtener una salida esperada en base a esos datos todo de forma autónoma. La única intervención humana se realiza durante el entrenamiento de la red; lo cual se puede realizar en diferentes softwares como lo es Matlab.

Las tarjetas FPGA se caracterizan por su flexibilidad en la construcción de Hardware, convirtiéndola en una herramienta útil para la implementación de sistemas o computadoras especificas; además de contar, según el modelo de la tarjeta con periféricos de gran utilidad. En base a esto se decide aplicar el sistema de análisis de comportamiento mediante la FPGA con la aplicación de Redes Neuronales y sistemas de varios núcleos para cumplir con las tareas requeridas.

# DESCRIPCIÓN DE LA PRACTICA:

Se cumplirá con el diseño e implementación de un sistema de dos núcleos, para el análisis y predicción del número de pasos de una persona en base a datos de entrada ingresados por consola y su posterior presentación por pantalla; todo basado en el uso de programación C y redes neuronales entrenadas para las tareas de conexión y procesamiento de los datos y su interacción con los periféricos de la tarjeta correspondientes.

# PROCEDIMIENTO:

1. Proceda con la creación de una carpeta denominada **PROYECTO** en el escritorio y copie en el interior de esta el archivo con extensión .txt con el nombre de **NN** y **VGA**.

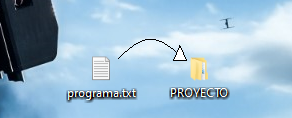


Ilustración 1: Creación de carpeta para la realización del proyecto.

1. Ejecute el programa **Quartus Prime 16.1 Standard Edition** que se encuentra en el escritorio.
2. Cree un nuevo proyecto tomando la opción **New Project Wizard** del recuadro como se observa en la Ilustración 2. Instantáneamente aparecerá la ventana de Introducción, en la cual debe dar clic en **Next**.

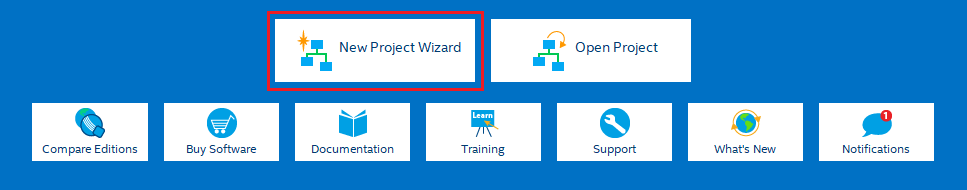


Ilustración 2: Creación de un nuevo proyecto en Quartus Prime 17.0.

1. En la primera línea escoja la ruta donde va a crear su proyecto, en este caso, la ruta será la carpeta denominada **PROYECTO** que fue creada en el paso 1 y que se encuentra en el escritorio, mientras que en la siguiente línea escoge como nombre del proyecto **PROYECTO**, tal como se presenta en la Ilustración 3. Luego da clic en **Next**.



Ilustración 3: Nombre del proyecto creado.

1. A continuación, se presenta la ventana en donde debemos elegir la opción de **Empty Project** para seleccionar el tipo de proyecto, tal como se visualiza en la Ilustración 4. Luego da clic en **Next**.

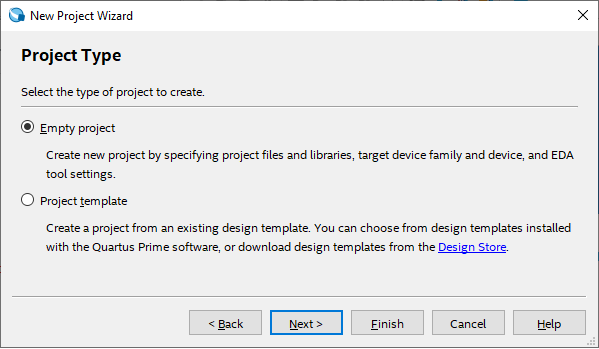


Ilustración 4: Selección del tipo de proyecto.

1. Se mostrará la ventana para adhesión de archivos. En esa ventana haremos clic en **Next**, puesto que vamos a crear la programación de este desde el inicio.
2. La ventana siguiente presenta 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**; lo cual puede ser efectuado mediante la herramienta de filtrado por nombre, tal como se muestra en la Ilustración 5. Seguidamente proceda a hacer clic en **Next**.

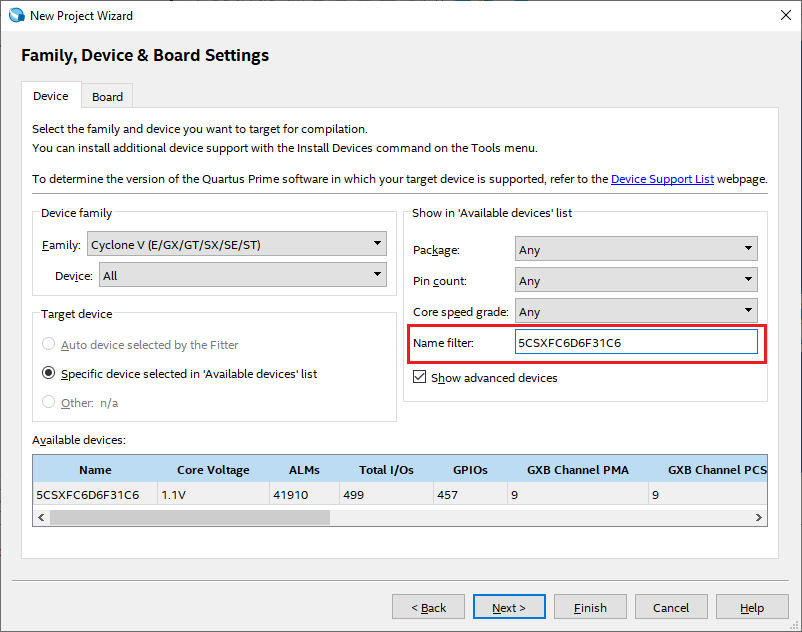


Ilustración 5: Selección de la Familia y modelo del chip FPGA SoC

1. En la siguiente ventana no se seleccionará alguna herramienta de simulación, de forma que debe quedar como se presenta en la Ilustración 6. Seguidamente haga clic en **Next**.

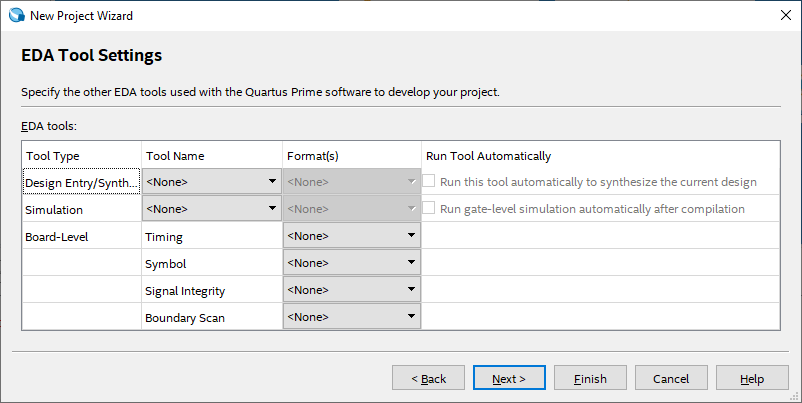


Ilustración 6: Herramientas para simulación del proyecto.

1. Al finalizar la creación del proyecto, aparecerá la ventana **Summary** para resumir todas las especificaciones que tendrá nuestro proyecto. Para finalizar haga clic en **Finish**.
2. Proceda a hacer clic en la opción **Tools🡪Qsys** para efectuar la programación de los bloques del sistema, tal como se presenta en la Ilustración 7.

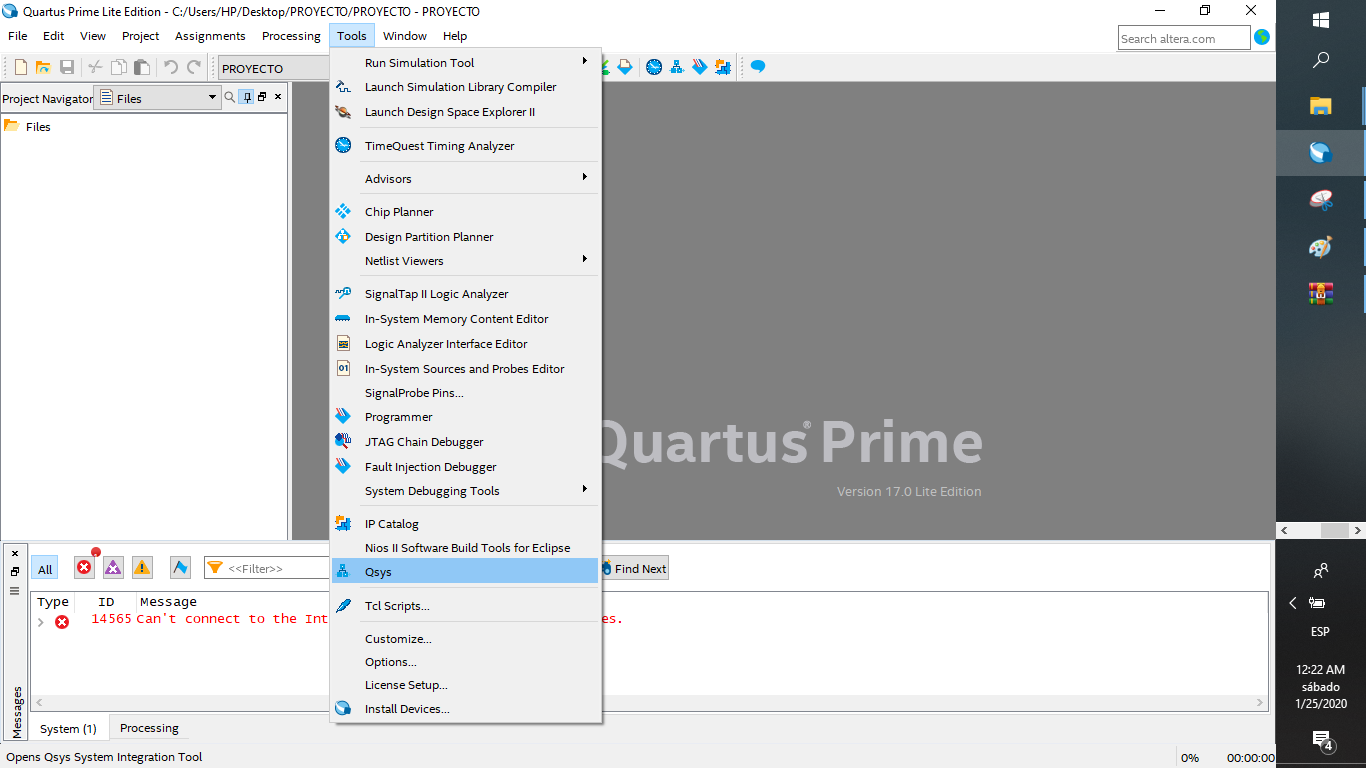


Ilustración 7: Creación de la programación del sistema en la herramienta de Qsys.

1. En la ventana de Qsys que se inicia proceda a hacer clic del bloque clk\_0 para elegir la opción de renombrar y cambiarla por **CLOCK\_50**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 8.

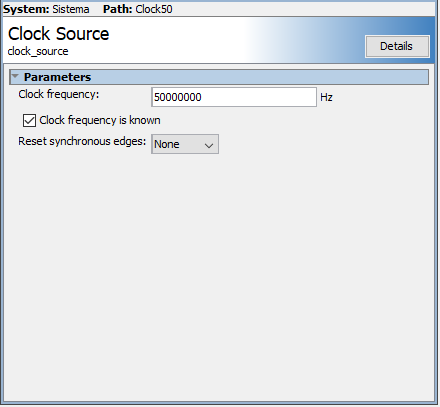


Ilustración 8: Configuración de los parámetros del bloque Clock Source.

1. Proceda a buscar en el catálogo el bloque denominado **Altera PLL**, seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 9.

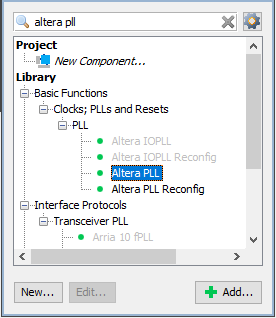


Ilustración 9: Búsqueda del bloque Altera PLL para incluirlo en programación.

1. Proceda a hacer clic del bloque Altera PLL para elegir la opción de renombrar y cambiarla por **PLL**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 10.

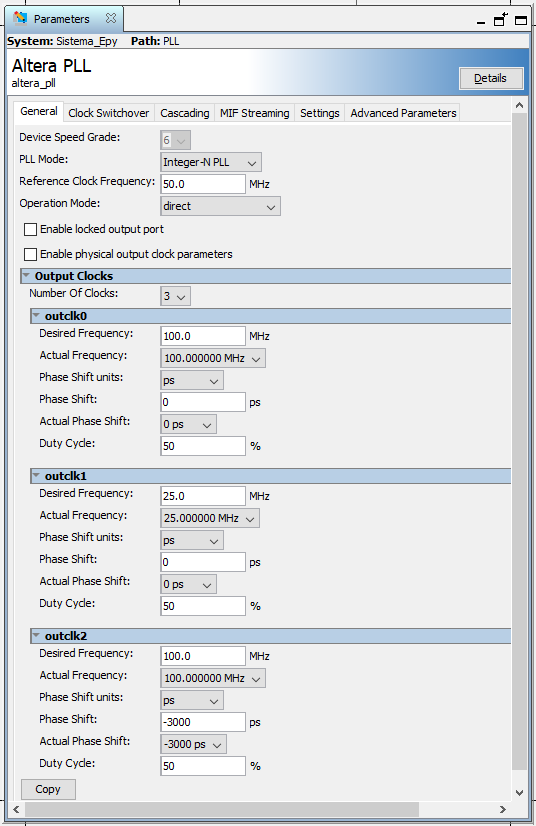


Ilustración 10: Configuración de los parámetros del bloque Altera PLL.

1. Proceda a buscar en el catálogo el bloque denominado **NIOS II Processor**, seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 11.

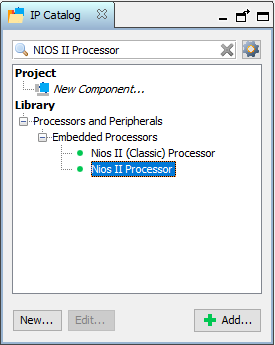


Ilustración 11: Búsqueda del bloque Nios II Processor para incluirlo en programación.

1. Proceda a hacer clic del bloque Nios II Porcessor para elegir la opción de renombrar y cambiarla por **NIOS2\_NN**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 12.

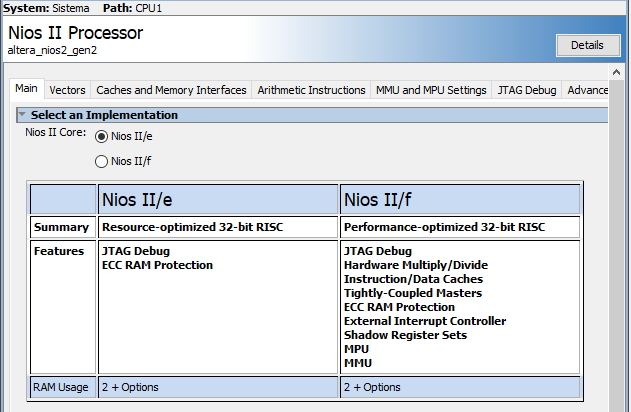


Ilustración 12: Configuración de los parámetros del bloque Nios II Processor.

1. Vuelva a efectuar los pasos 14. y 15. del procedimiento, pero esta vez renombre el Nios II Processor con el nombre **NIOS2\_VGA**. Las configuraciones del Main deben ser las mismas que las que se muestran en la Ilustración 12.
2. Proceda a buscar en el catálogo el bloque denominado **JTAG UART**, seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 13.

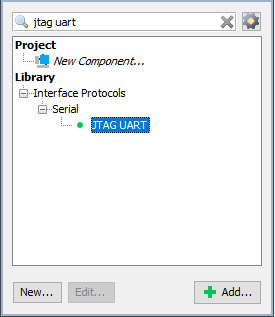


Ilustración 13: Búsqueda del bloque JTAG UART para incluirlo en programación.

1. Proceda a hacer clic del bloque JTAG UART para elegir la opción de renombrar y cambiarla por **JTAG\_UART**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 14.

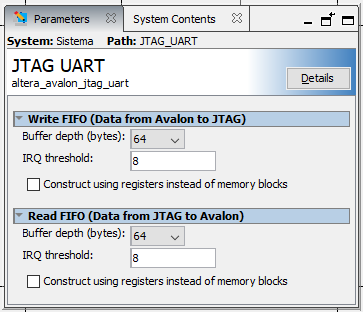


Ilustración 14: Configuración de los parámetros del bloque JTAG UART.

1. Proceda a buscar en el catálogo el bloque denominado **System ID Peripheral**, seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 15.

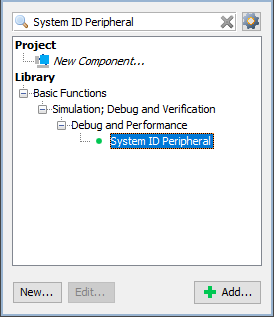


Ilustración 15: Búsqueda del bloque System ID Peripheral para incluirlo en programación.

1. Proceda a hacer clic del bloque System ID Peripheral para elegir la opción de renombrar y cambiarla por **SYSTEM\_ID\_1**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 16.

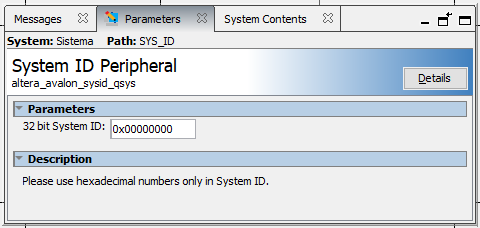


Ilustración 16: Configuración de los parámetros del bloque System ID Peripheral.

1. Proceda a buscar en el catálogo el bloque denominado **Interval Timer**, seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 17.

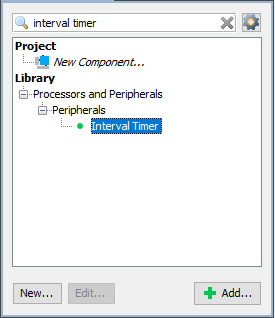


Ilustración 17: Búsqueda del bloque Interval Timer para incluirlo en programación.

1. Proceda a hacer clic del bloque Intervel Timer para elegir la opción de renombrar y cambiarla por **TIMER**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 18.

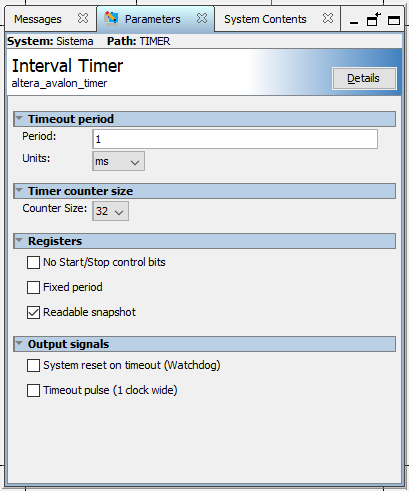


Ilustración 18: Configuración de los parámetros del bloque Interval Timer.

1. Proceda a buscar en el catálogo el bloque denominado **SDRAM Controller**, seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 17.

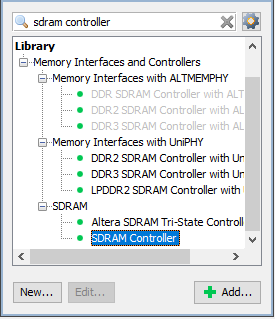


Ilustración 19: Búsqueda del bloque SDRAM Controller para’ incluirlo en programación.

1. Proceda a hacer clic del bloque SDRAM Controller para elegir la opción de renombrar y cambiarla por **SDRAM**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 20.

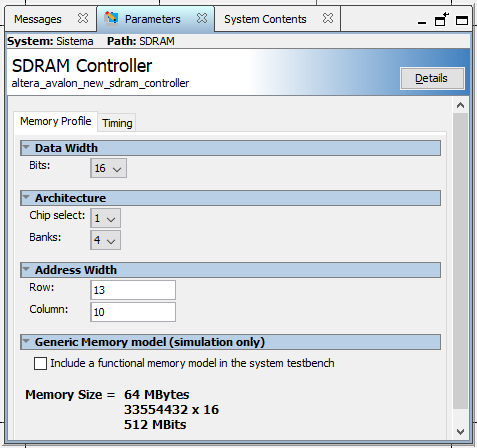


Ilustración 20: Configuración de los parámetros del bloque SRAM Controller.

1. Proceda a buscar en el catálogo el bloque denominado **On-Chip Memory (RAM or ROM)**, seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 21.

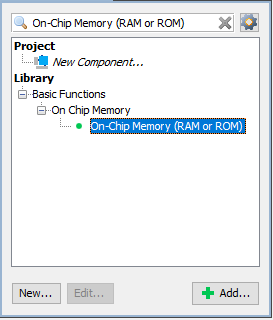


Ilustración 21: Búsqueda del bloque On-Chip Memory (RAM or ROM) para incluirlo en programación.

1. Proceda a hacer clic del bloque On-Chip Memory (RAM or ROM) para elegir la opción de renombrar y cambiarla por **OnChip1**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 22.

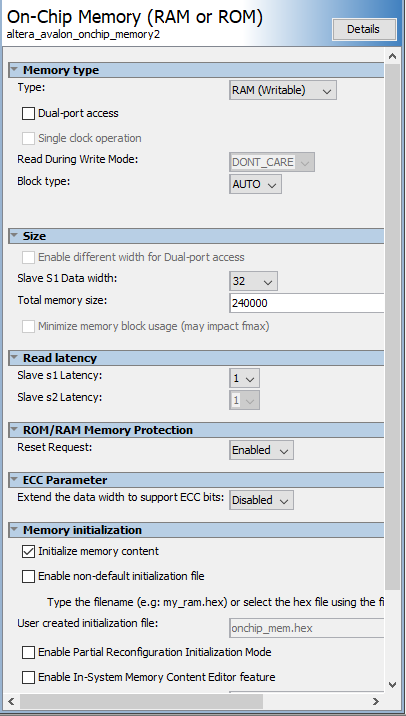


Ilustración 22: Configuración de los parámetros del bloque On-Chip Memory (RAM or ROM).

1. Vuelva a efectuar los pasos 25. y 26. del procedimiento, pero esta vez renombre el On-Chip Memory (RAM or ROM) con el nombre **OnChip2**. Las configuraciones del bloque deben ser las mismas que las que se muestran en la Ilustración 22.
2. Proceda a buscar en el catálogo el bloque denominado **Pixel Buffer DMA Controller** seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 23.

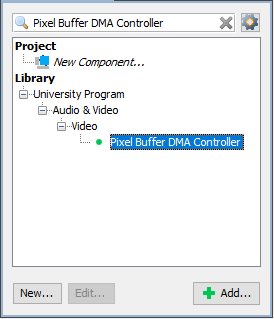


Ilustración 23: Búsqueda del bloque Pixel Buffer DMA Controller para incluirlo en programación.

1. Proceda a hacer clic del bloque Pixel Buffer DMA Controller para elegir la opción de renombrar y cambiarla por **PixelBuffer**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades del mismo y dejarlo como se muestra en la Ilustración 24.



Ilustración 24: Configuración de los parámetros del bloque Pixel Buffer DMA Controller.

1. Proceda a buscar en el catálogo el bloque denominado **RGB Resampler** seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 25.

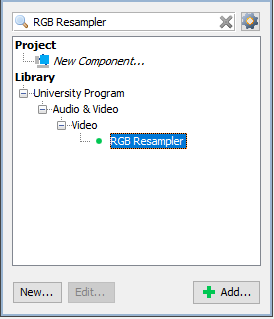


Ilustración 25: Búsqueda del bloque RGB Resampler para incluirlo en programación.

1. Proceda a hacer clic del bloque RGB Resampler para elegir la opción de renombrar y cambiarla por **RGB\_RESAMPLER**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 26.

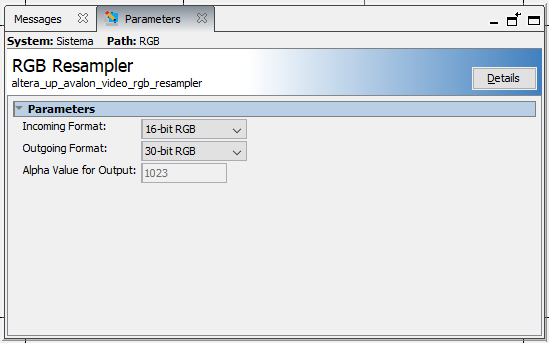


Ilustración 26: Configuración de los parámetros del bloque RGB Resampler.

1. Proceda a buscar en el catálogo el bloque denominado **Scaler** seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 27.



Ilustración 27: Búsqueda del bloque Scaler para incluirlo en programación.

1. Proceda a hacer doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 28.

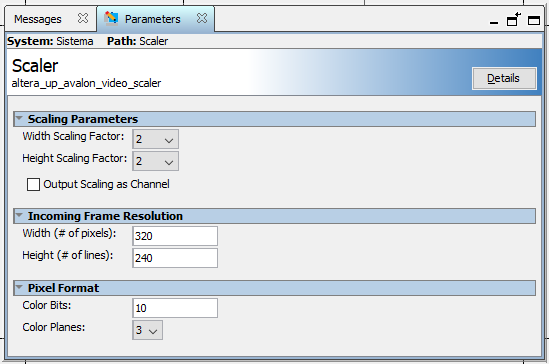


Ilustración 28: Configuración de los parámetros del bloque Scaler.

1. Proceda a buscar en el catálogo el bloque denominado **Dual-Clock FIFO** seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 29.

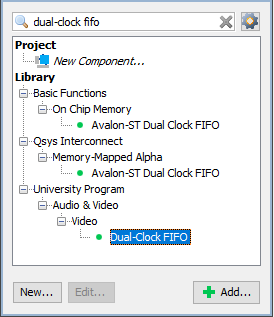


Ilustración 29: Búsqueda del bloque Dual-Clock FIFO para incluirlo en programación.

1. Proceda a hacer clic del bloque Dual-Clock FIFO para elegir la opción de renombrar y cambiarla por **DUAL\_CLOCK\_FIFO**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 30.

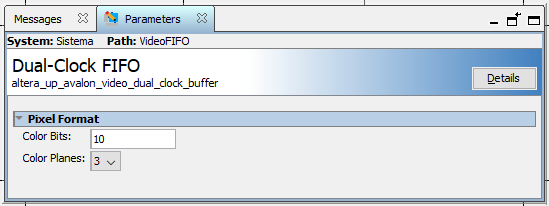


Ilustración 30: Configuración de los parámetros del bloque Dual-Clock FIFO.

1. Proceda a buscar en el catálogo el bloque denominado **VGA Controller** seguidamente haca clic en la opción **Add** para añadirlo a la programación, como se presenta en la Ilustración 31.

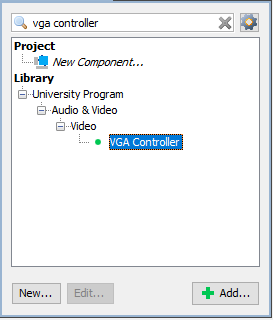


Ilustración 31: Búsqueda del bloque VGA Controller para incluirlo en programación.

1. Proceda a hacer clic del bloque VGA Contoller para elegir la opción de renombrar y cambiarla por **VGA**, seguidamente haga doble clic sobre el bloque para acceder a las propiedades de este y dejarlo como se muestra en la Ilustración 32.

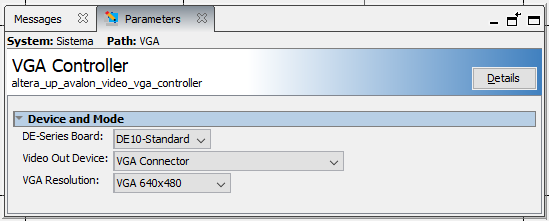


Ilustración 32: Configuración de los parámetros del bloque VGA Controller.

1. Hacer doble clic en el bloque de **NIOS2\_NN** y en la sección de **Verctors** efectuar las siguientes configuraciones, que se presentan en la Ilustración 32. Por otra parte, para el **NIOS2\_VGA** emplear las configuraciones de vectores que se encuentran en la Ilustración 33.



Ilustración 33: Configuración de los parámetros del bloque CPU1 en parámetro de vectores.

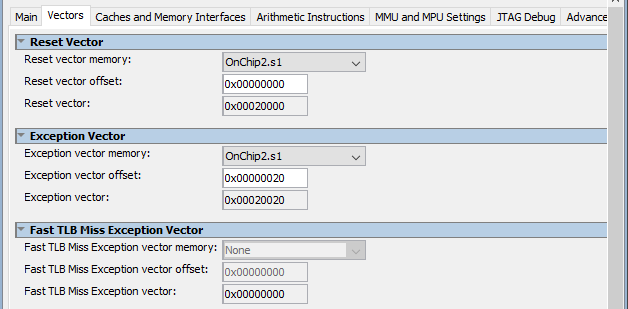


Ilustración 34: Configuración de los parámetros del bloque CPU2 en parámetro de vectores.

1. Seleccionar la opción de **Assing Base Addresses**, que se encuentra en la barra de herramientas en la opción **System**, como se puede observar en la Ilustración 35.

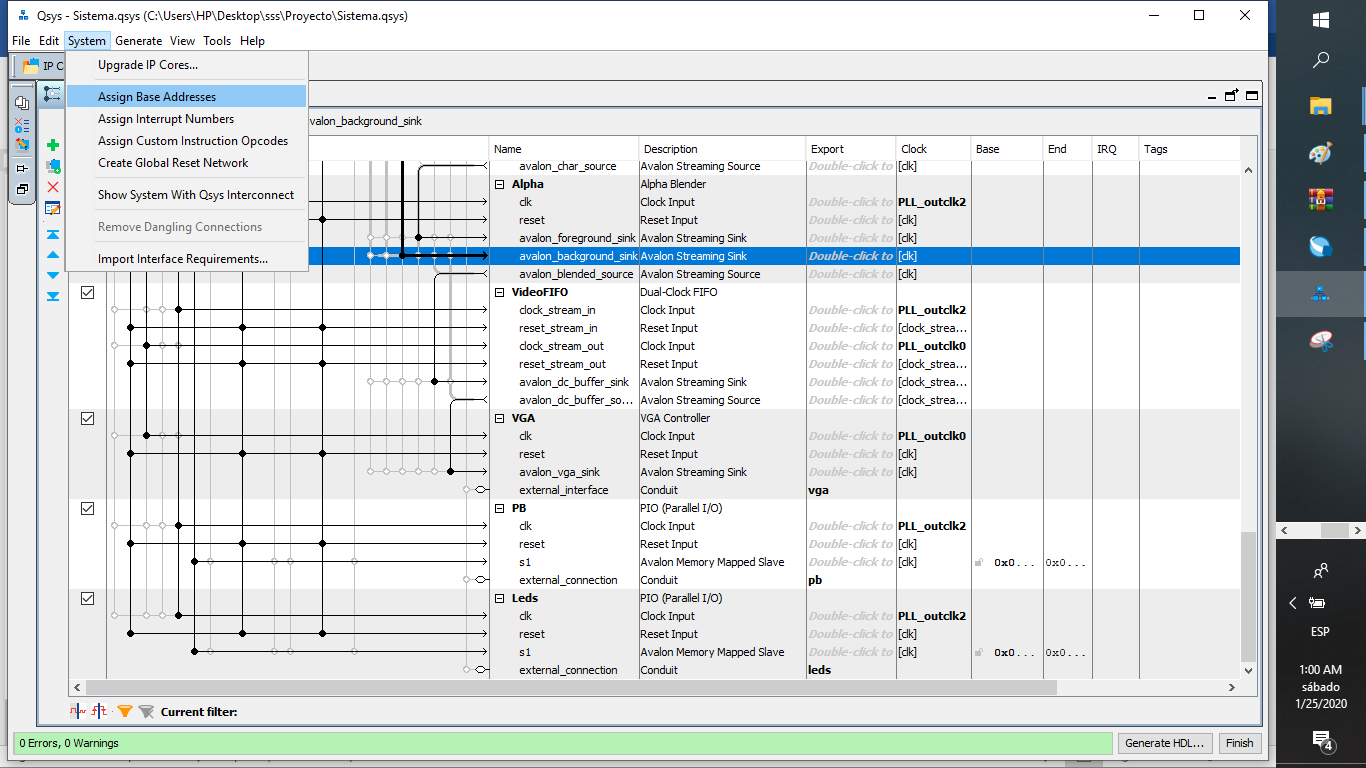


Ilustración 35: Asignación de base de dirección a los bloques de programación del sistema.

1. Seleccionar la opción de **Create Global Reset Network**, que se encuentra en la barra de herramientas en la opción **System**, como se puede observar en la Ilustración 36.

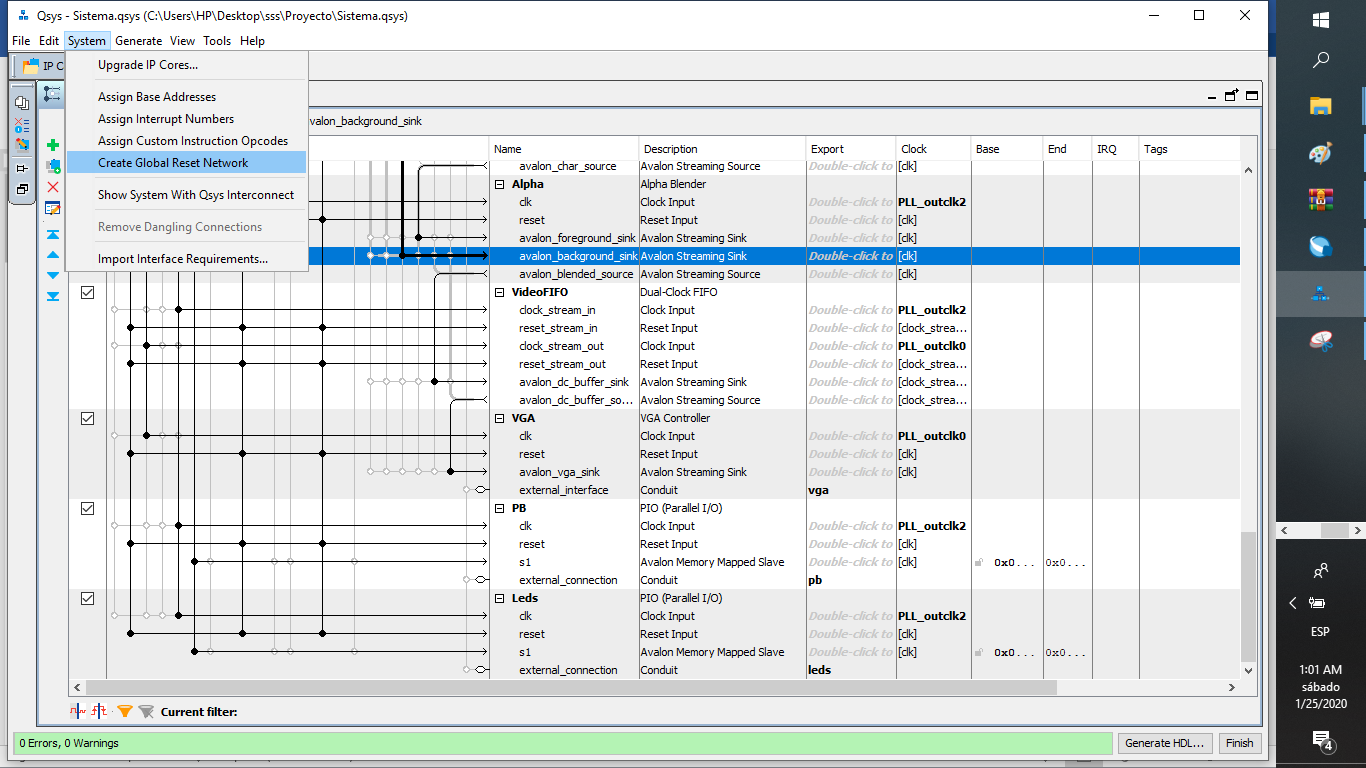


Ilustración 36: Creación del Reset Global del sistema.

1. Proceda a efectuar las conexiones y configuraciones en los bloques de programación, como se muestra en la Ilustración 37.

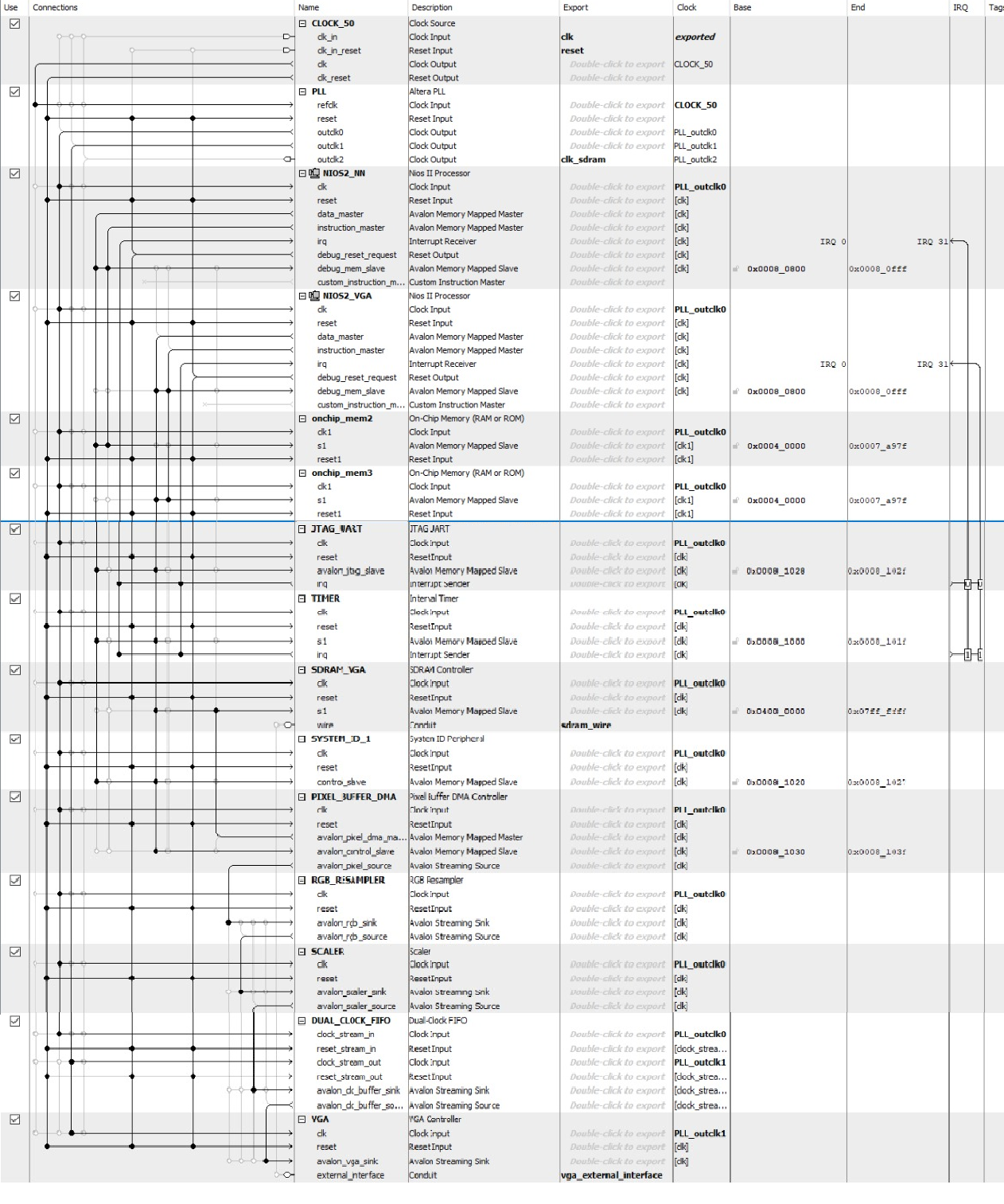


Ilustración 37: Conexiones de los bloques de programación en Qsys.

1. Realice la generación del archivo HDL, haciendo clic en la opción **Generate🡪HDL**, que se encuentra en la parte superior de la barra de herramientas. Seguidamente se abrirá una ventana en la cual debemos realizar las configuraciones que se muestran en la Ilustración 38.

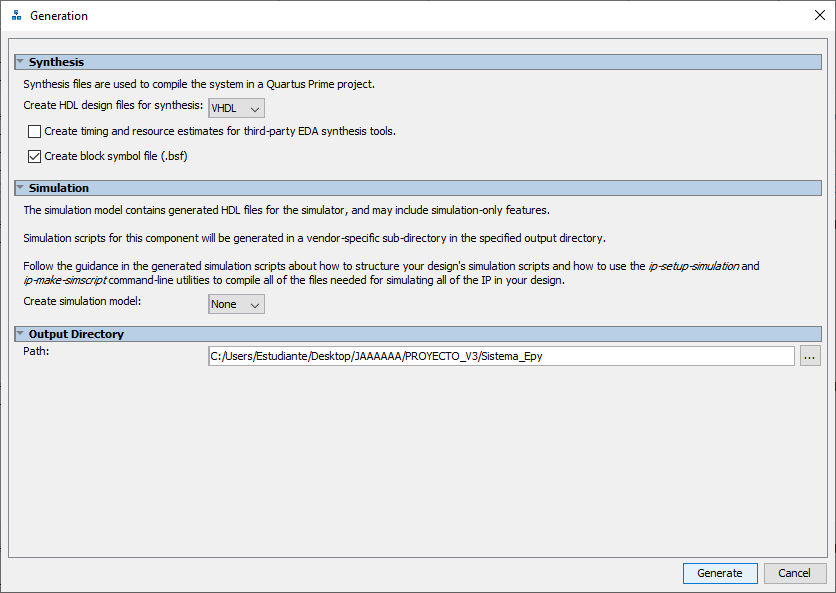


Ilustración 38: Generación del archivo HDL de la programación del sistema.

1. Verificar que se haya efectuado correctamente la operación, seguidamente se abrirá una ventana en la cual se debe hacer clic en **Cerrar**, como se muestra en la Ilustración 39 y posteriormente en acceder a la opción **Finish** que se encuentra en la parte inferior derecha.



Ilustración 39: Ventana de generación exitosa del archivo VHDL de forma exitosa.

1. Proceda a hacer clic en **Assigments🡪Pin Planner** para efectuar la asignación respectiva de lo pines de la programación en Qsys a la tarjeta FPGA, como se presenta en la Ilustración 40.

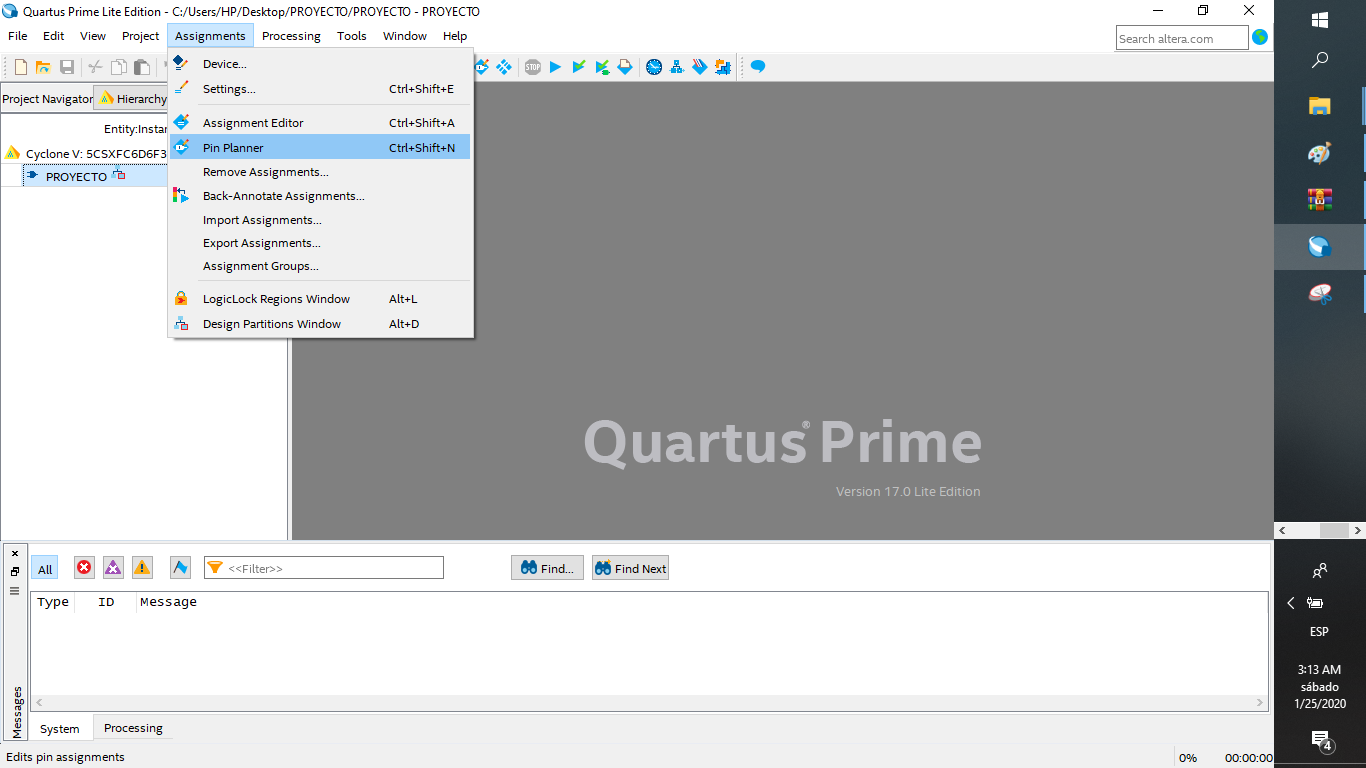


Ilustración 40: Acceso a Pin Planner para la Asignación de pines de la tarjeta.

1. Efectué la asignación de pines entre la FPGA y la programación efectuada en Qsys, tal como se muestra en las siguientes ilustraciones.

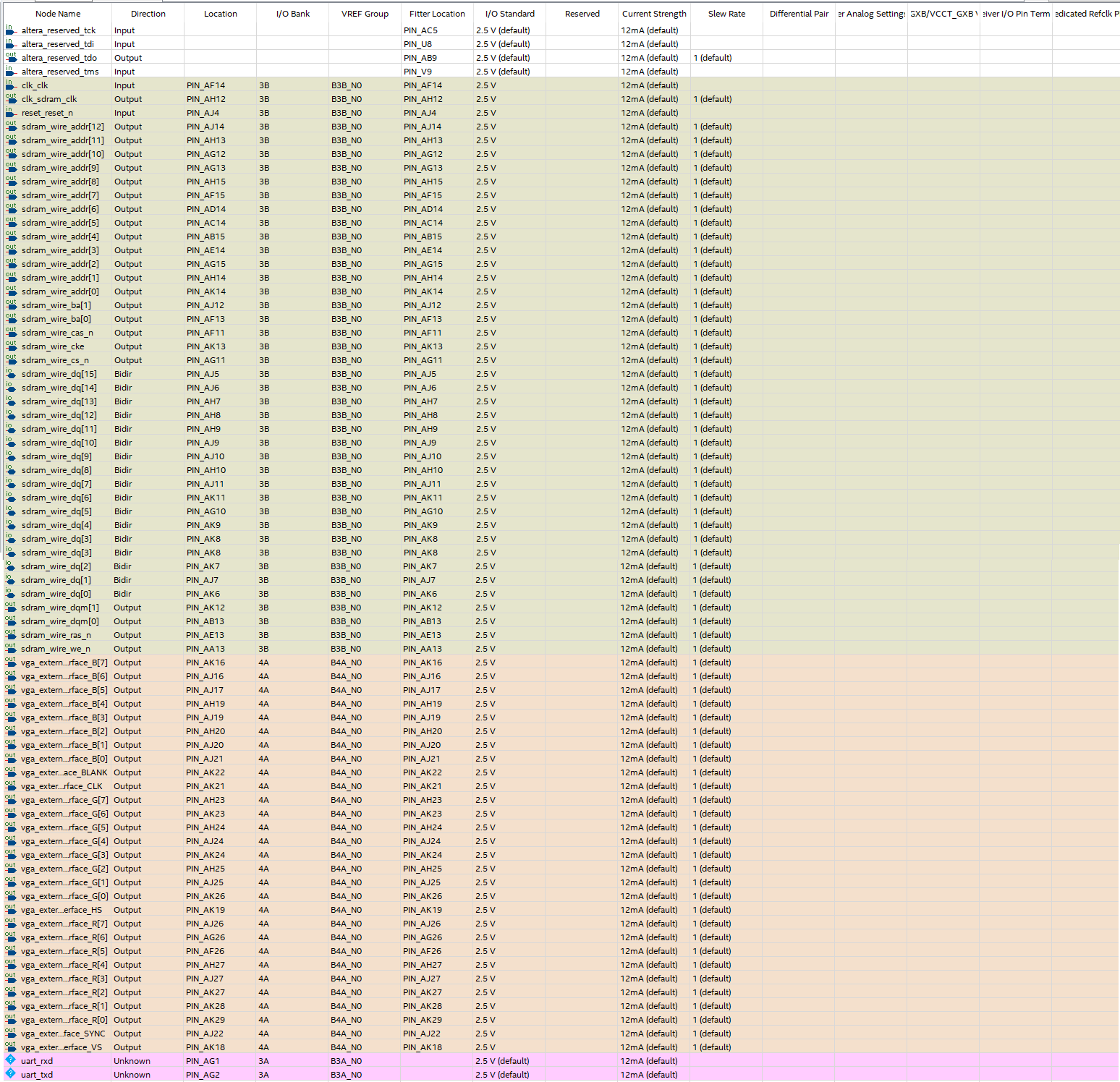


Ilustración 41: Asignación de pines de la FPGA en Pin Planner-Parte1.

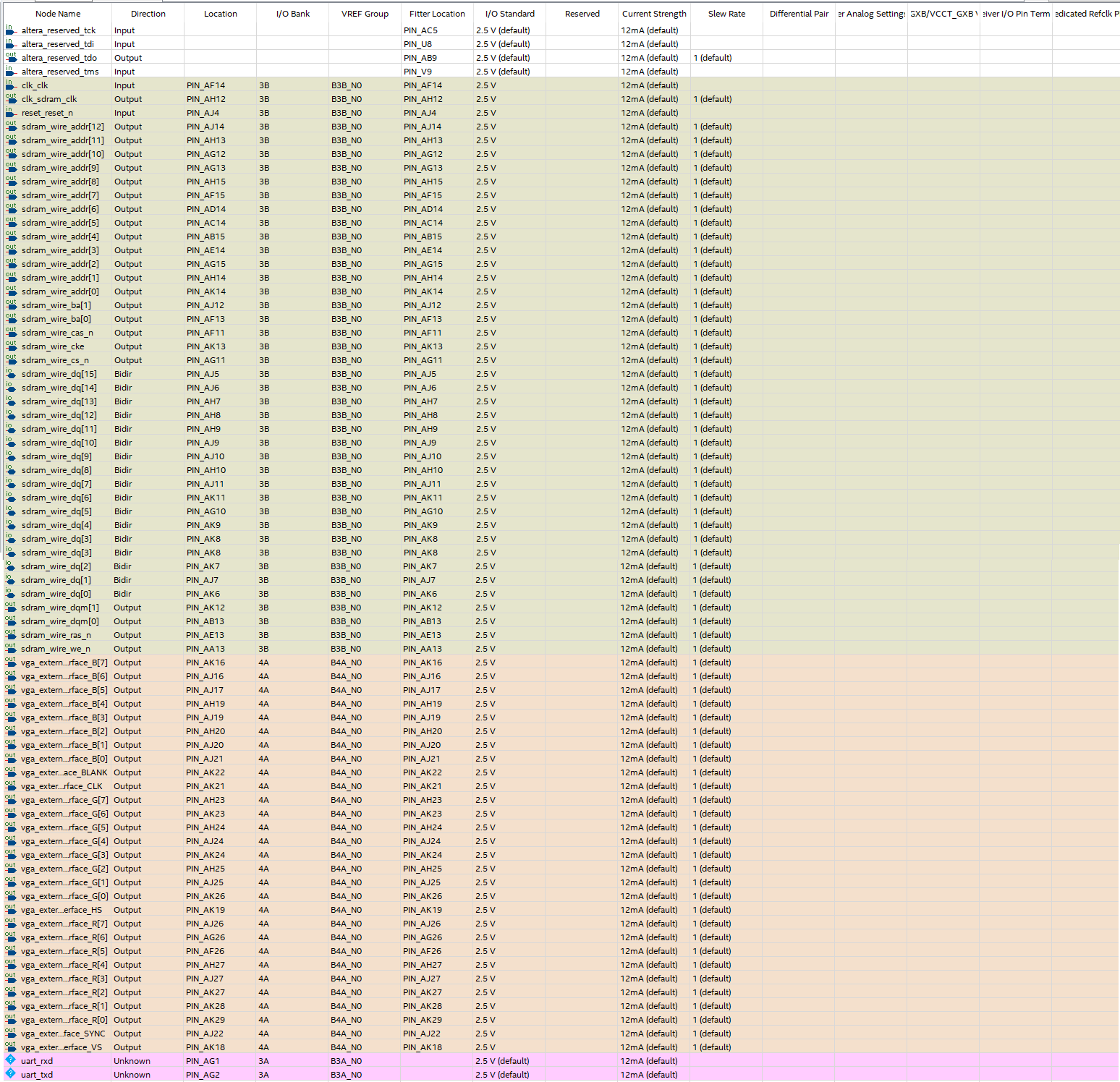


Ilustración 42: Asignación de pines de la FPGA en Pin Planner-Parte2.

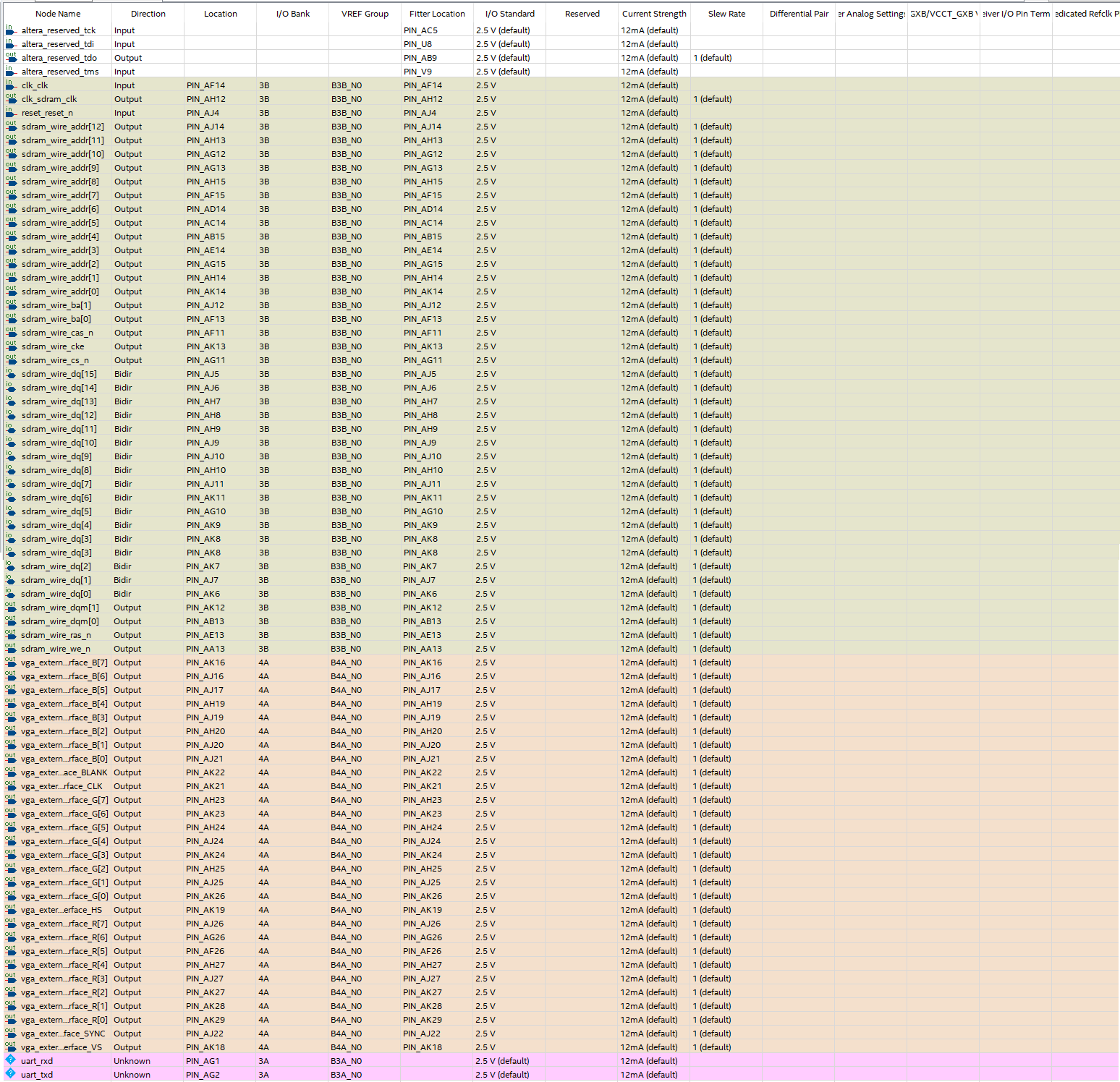
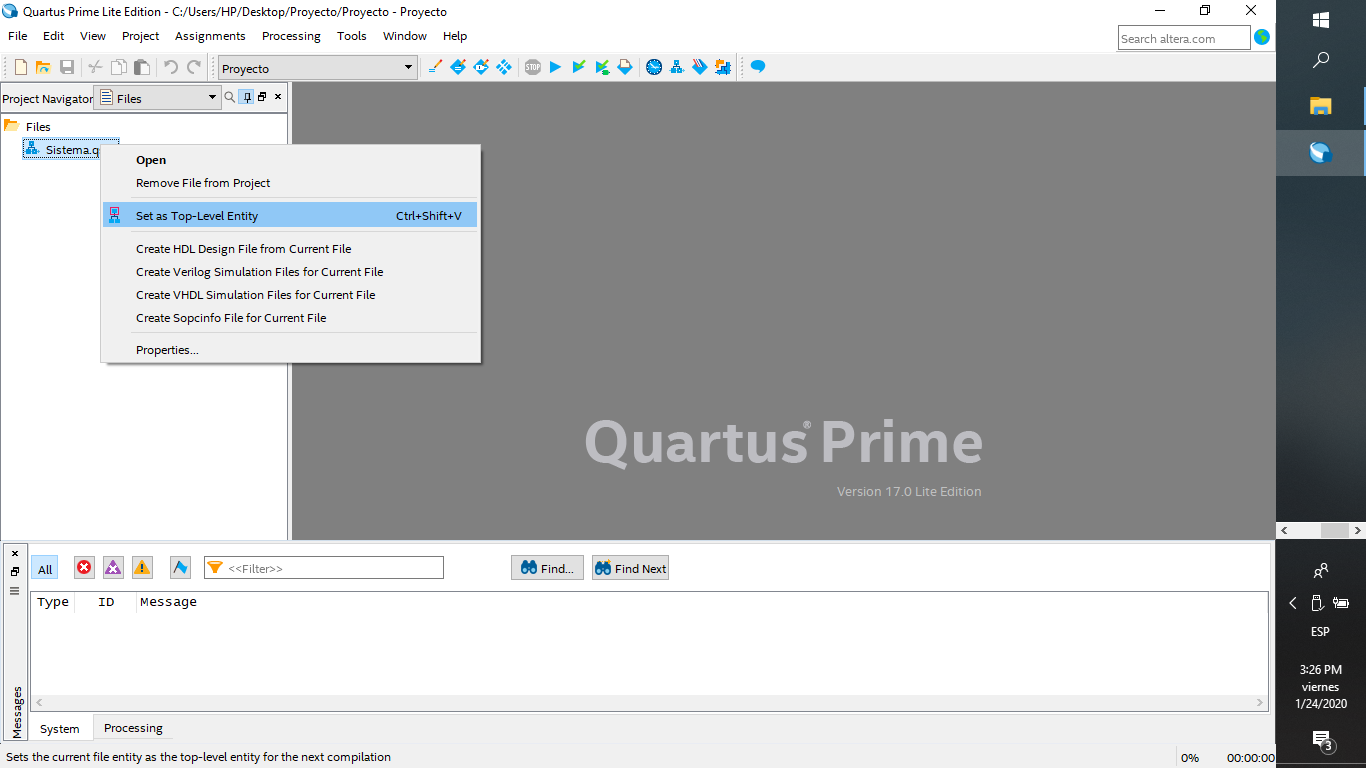


Ilustración 43: Asignación de pines de la FPGA en Pin Planner-Parte3.

1. Proceda a cerrar la herramienta de **PinPlanner**, coloque como **Top Level Entity** el archivo de Qsys que se ha creado y compílelo haciendo clic en el icono , como se muestra en la Ilustración 44.

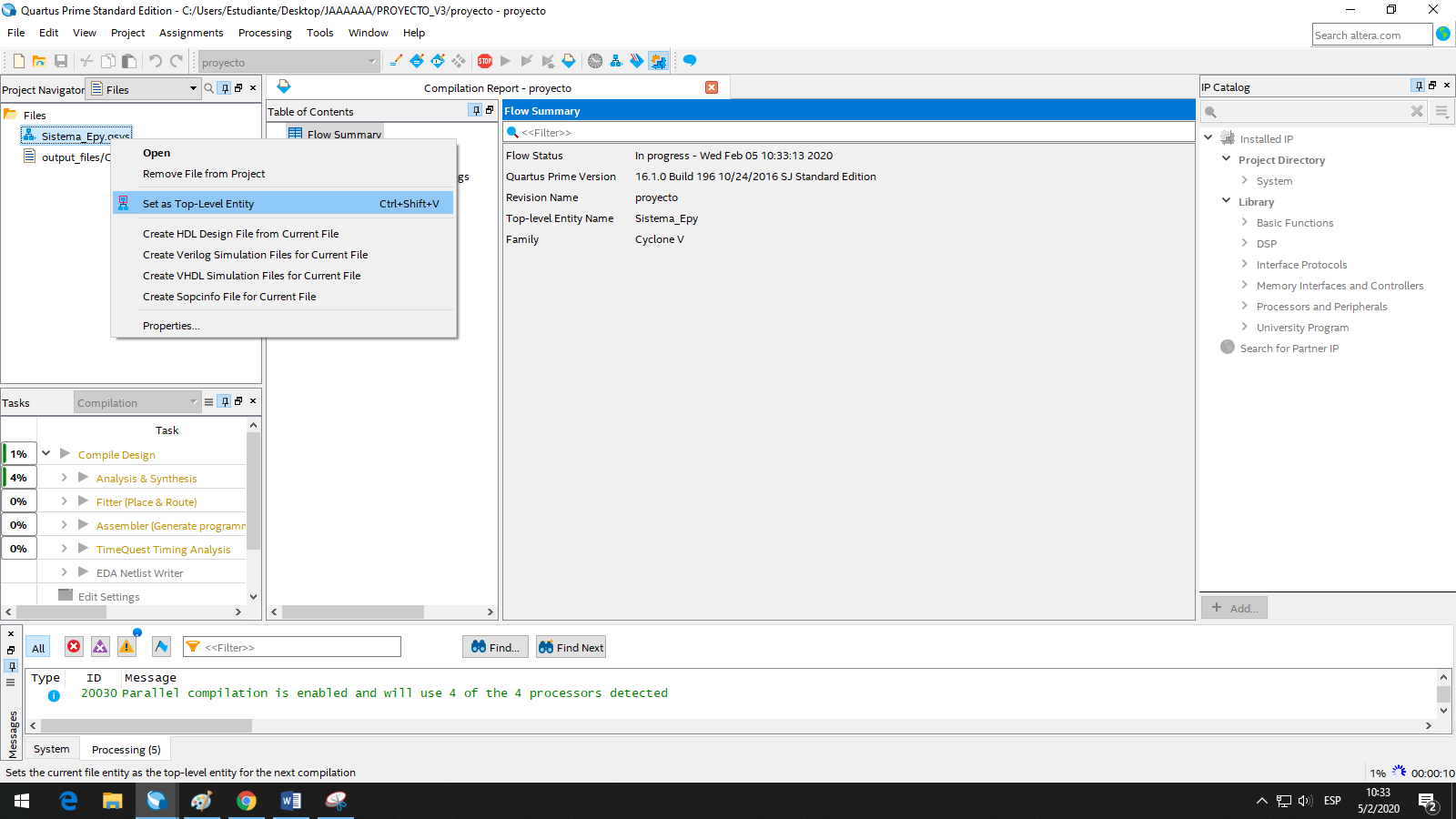
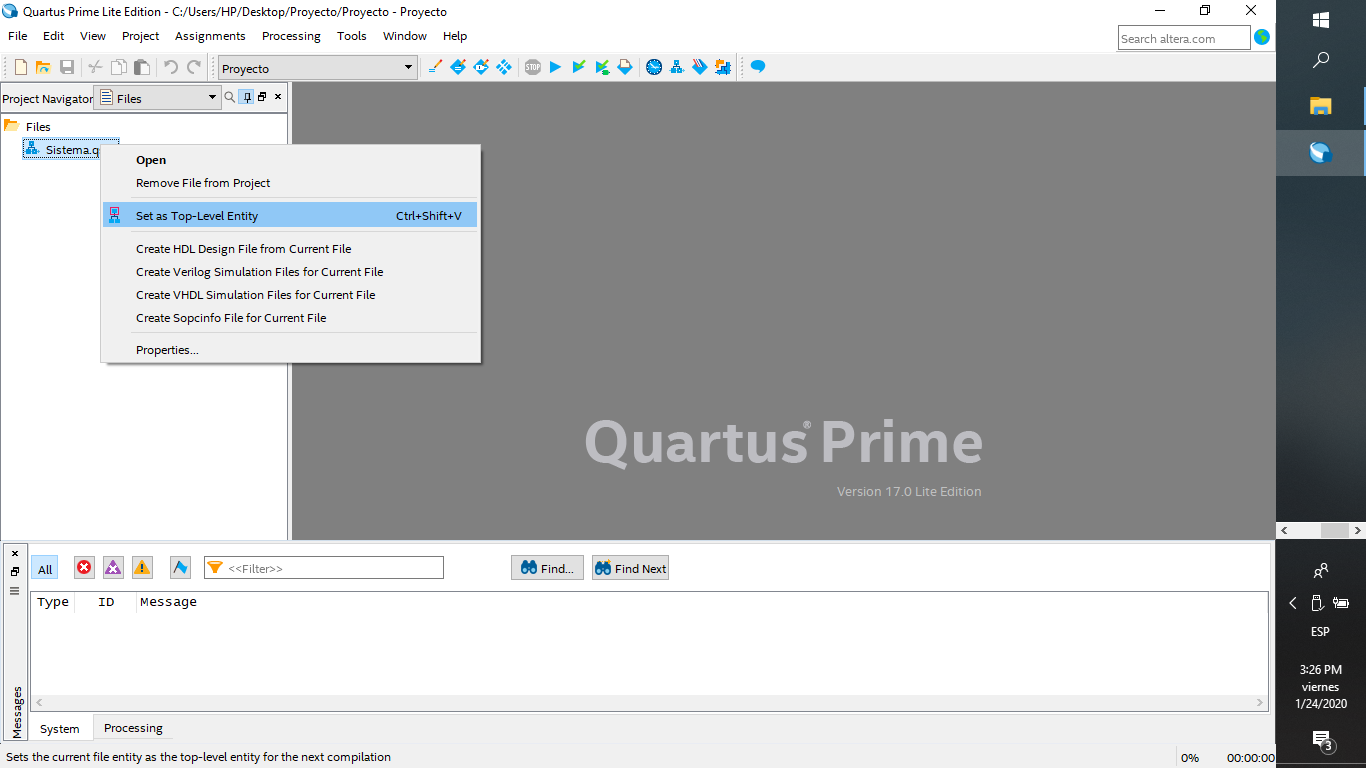


Ilustración 44: Colocación de Top Level Entity al archivo de Qsys y compilación de este.

1. Una vez se haya compilado completamente y sin errores, proceda conectar la tarjeta al puerto USB de la computadora y continúe haciendo clic en el icono .
2. En la ventana que se presenta, proceda a eliminar los archivos que se encuentran presentes, seguidamente haga clic en **Autodetect**. En ventana que se origina se debe elegir la penúltima opción, tal como se muestra en la Ilustración 45.

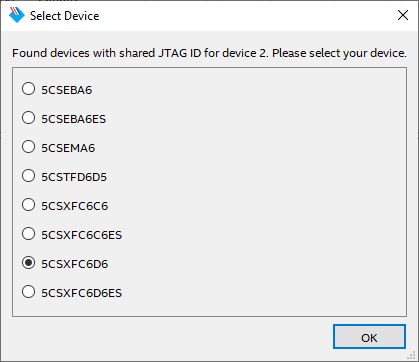


Ilustración 45: Selección del dispositivo para la carga de la programación.

1. Proceda a hacer clic en el segundo archivo y seleccionar la opción **Change File** y reemplácelo por el archivo que se encuentra en la dirección de **OutputFiles**, como se presenta en la Ilustración 46.

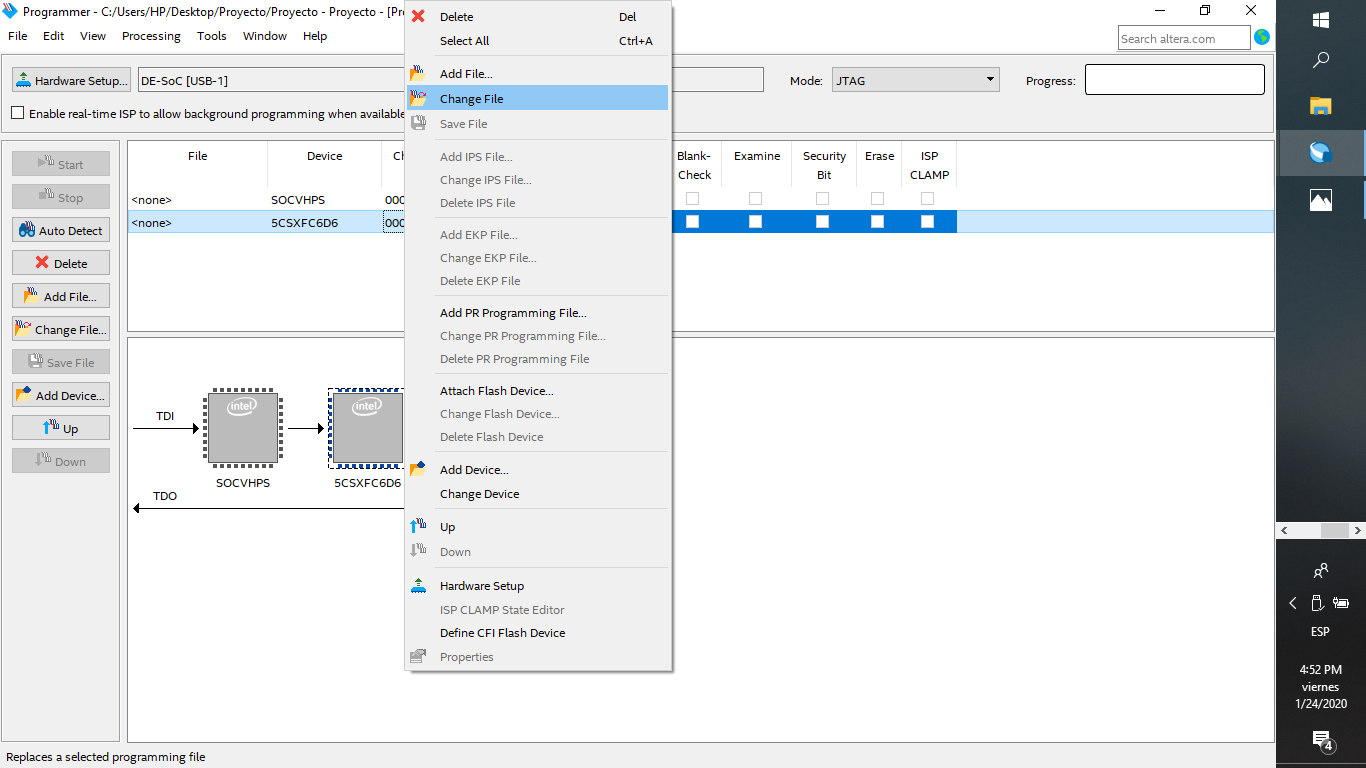


Ilustración 46: Carga del nuevo archivo al sistema de la tarjeta FPGA.

1. Efectué la activación de la casilla **Program/Configure** del archivo que se acaba de cambiar, como se puede observar en la Ilustración 47 y continúe haciendo clic en **Start**.

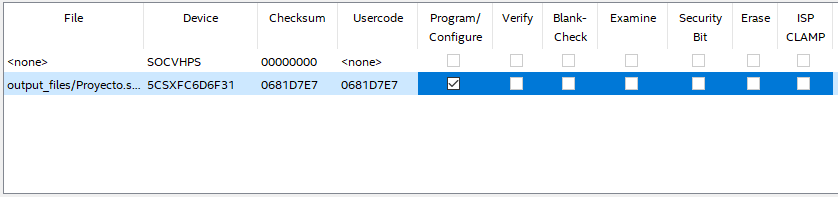


Ilustración 47: Configuración de parámetros de los archivos del proyecto que serán cargados.

1. Una vez que se haya presentado el mensaje de **100%Successful** que se presenta en la parte superior derecha de la ventana, cierre la misma y diríjase a la opción **Tools🡪Nios II Software Builds Tools for Eclipse**, seguidamente se abrirá una ventana en la cual debe tomar la ruta en la que se encuentra guardado el proyecto, como se presenta en la Ilustración 48 y seguidamente proceda haciendo clic en **Ok**.

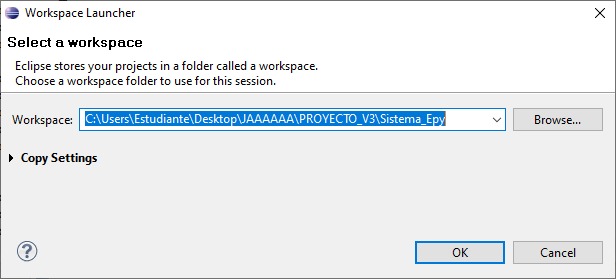


Ilustración 48: Selección de la ruta para el Workspace de la programación del sistema.

1. Aparecerá de esta manera el ambiente de programación de la herramienta Eclipse; donde se realizará la creación de un proyecto nuevo. Con este fin se realiza la selección de las siguientes opciones, **File -> New -> Nios II Application and BSP from Template**, tal como se detalla en la siguiente ilustración.

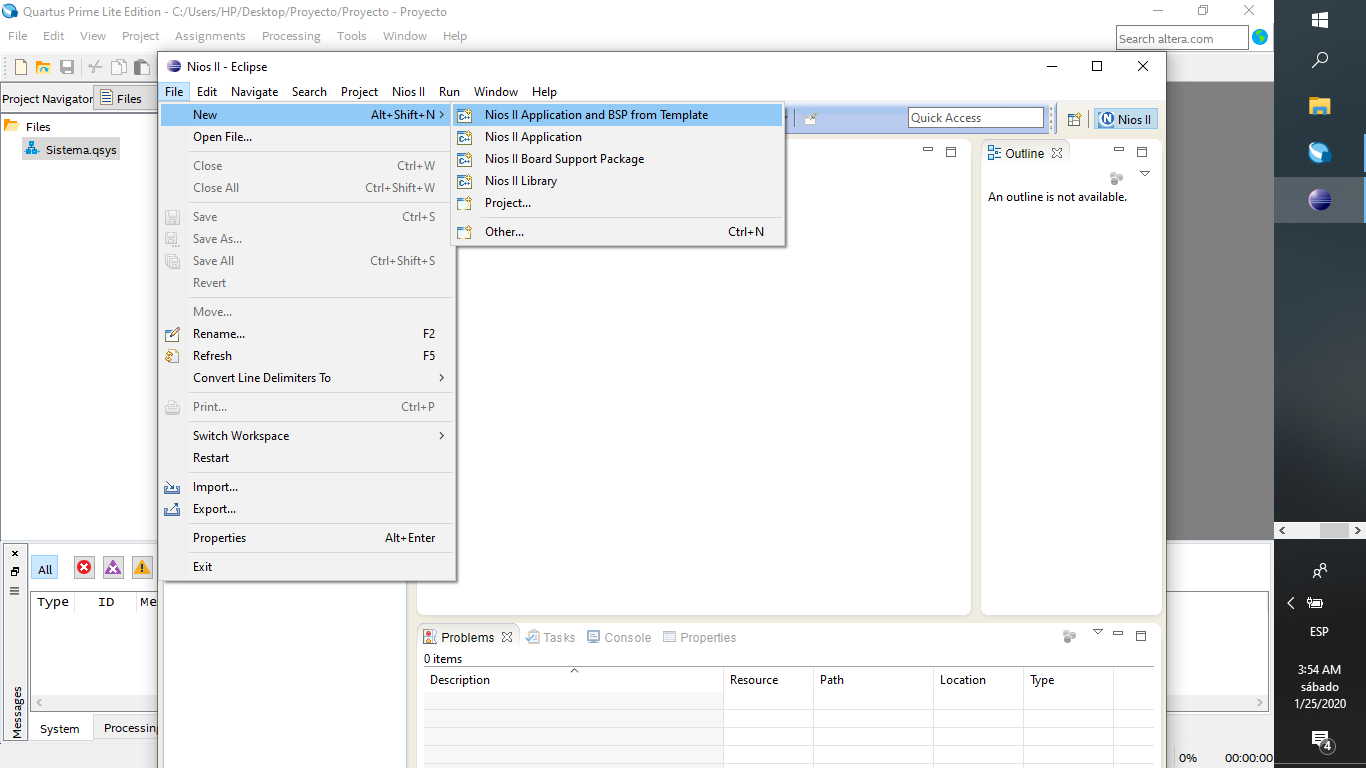


Ilustración 49: Creación de un nuevo proyecto en Eclipse.

1. Emergerá una ventana en donde para la sección de **SOPC Information File name**, se debe buscar el archivo de extensión “**.sopcinfo”**  generado anteriormente y que contiene la información del sistema embebido, ubicado en la carpeta del proyecto.

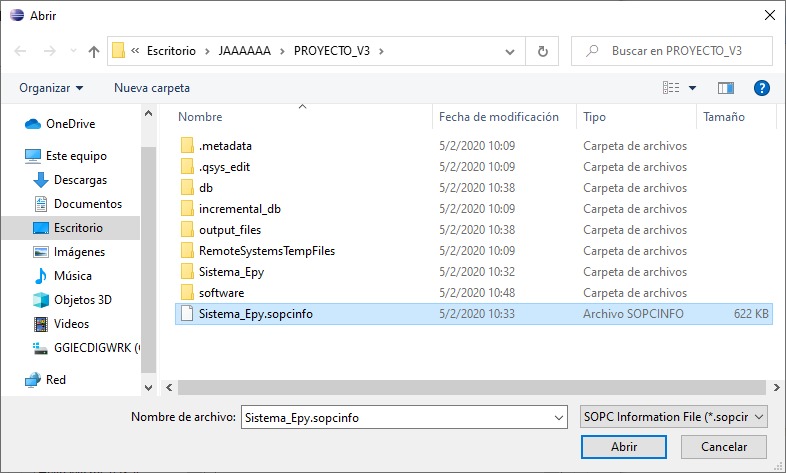


Ilustración 50: Selección del archivo con la información del computador.

1. Una vez realizado el paso anterior es posible seleccionar el elemento en la opción **CPU name**, en donde se seleccionará el núcleo **NIOS2\_VGA** de la computadora embebida. En **Project name** se escribirá el nombre deseado para el proyecto; en este caso se escribirá **VGA2**. En **Project Template**, se escogerá el tipo de proyecto **Hellow World** y luego se dará clic en **FINISH** para terminar con la creación.

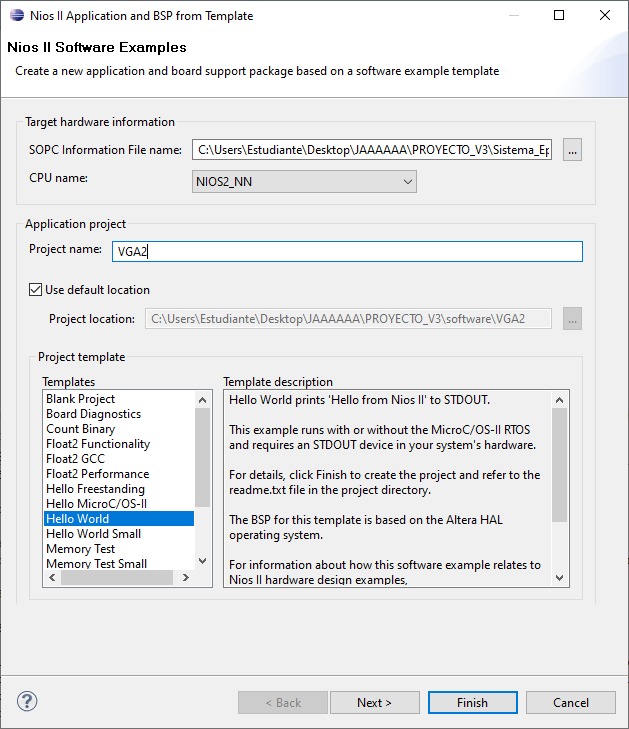


Ilustración 51: Selección del proyecto y demás configuraciones.

1. A continuación, se procede a realizar la configuración de la carpeta **VGA2\_bsp[SISTEMA]** dando clic derecho sobre esta y seleccionando la opción **Properties**, como se detalla en la siguiente imagen.

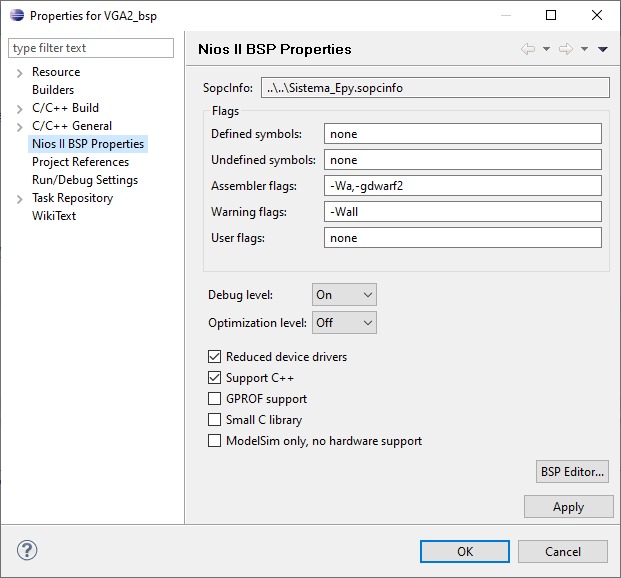


Ilustración 52: Configuración de la carpeta BSP.

1. Se continua con la programación a utilizar en el archivo creado en el procesador **VGA2**. Para esto se debe copiar el contenido del archivo **NN.txt**, disponible en la carpeta del proyecto, en el programa **HelloWorld.c** de la capeta **VGA2.** Si se desea se puede cambiar el nombre del archivo de programación una vez guardado los datos.
2. Previo a seguir, guardar todos los cambios realizados en el proyecto. Ahora se continúa dando clic sobre el icono de **Build All** () para realizar la compilación de las carpetas que se encuentran en el proyecto, al realizarlo aparecerá la siguiente ventana.

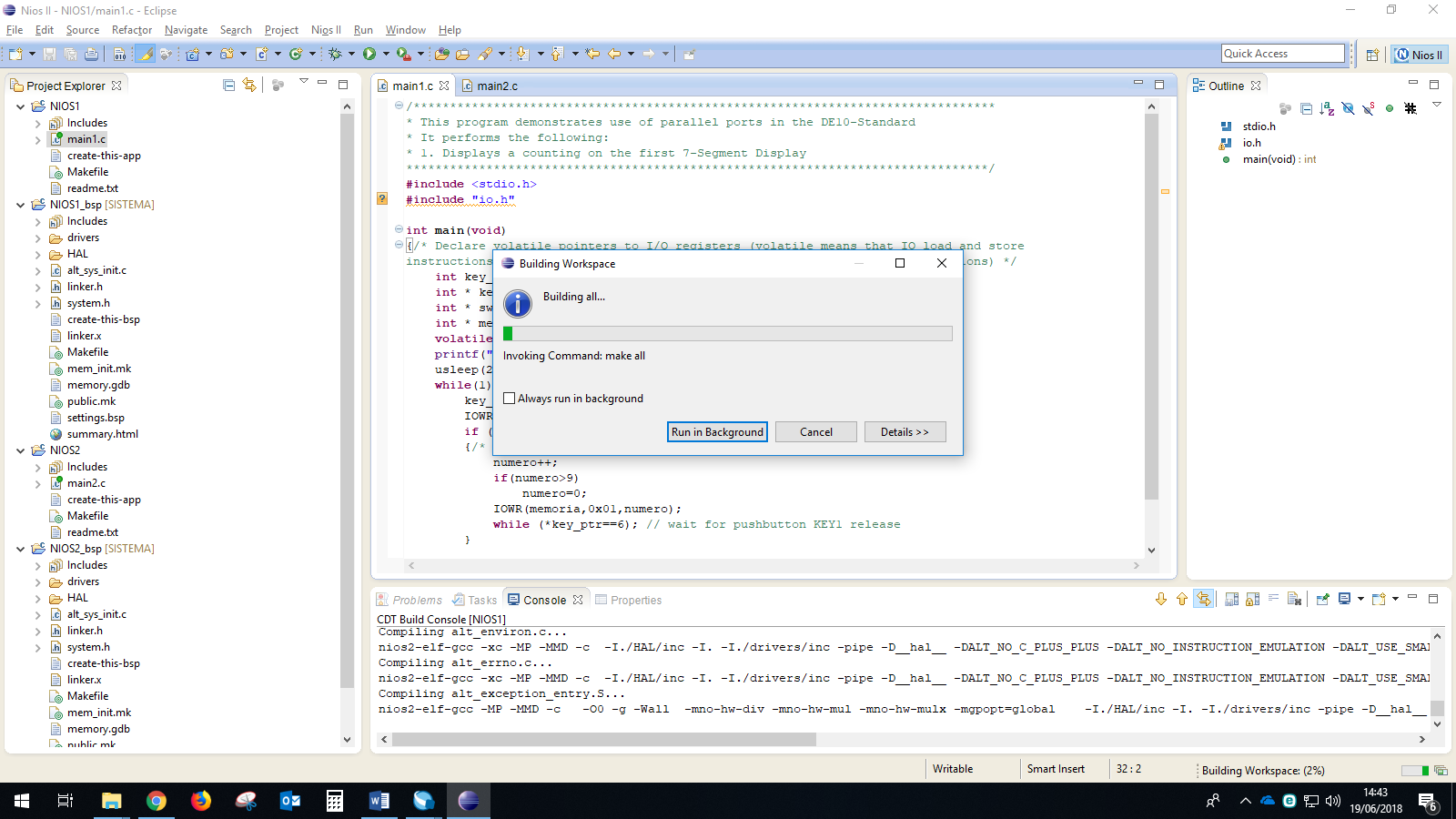


Ilustración 53: Construcción de los archivos del proyecto.

1. Acto seguido se debe hacer clic sobre la carpeta **VGA\_bsp [SISTEMA]**, escogemos la opción de **NIOS II** para luego dar clic sobre la opción de **BSP EDITOR**, como se demuestra en la siguiente ilustración.

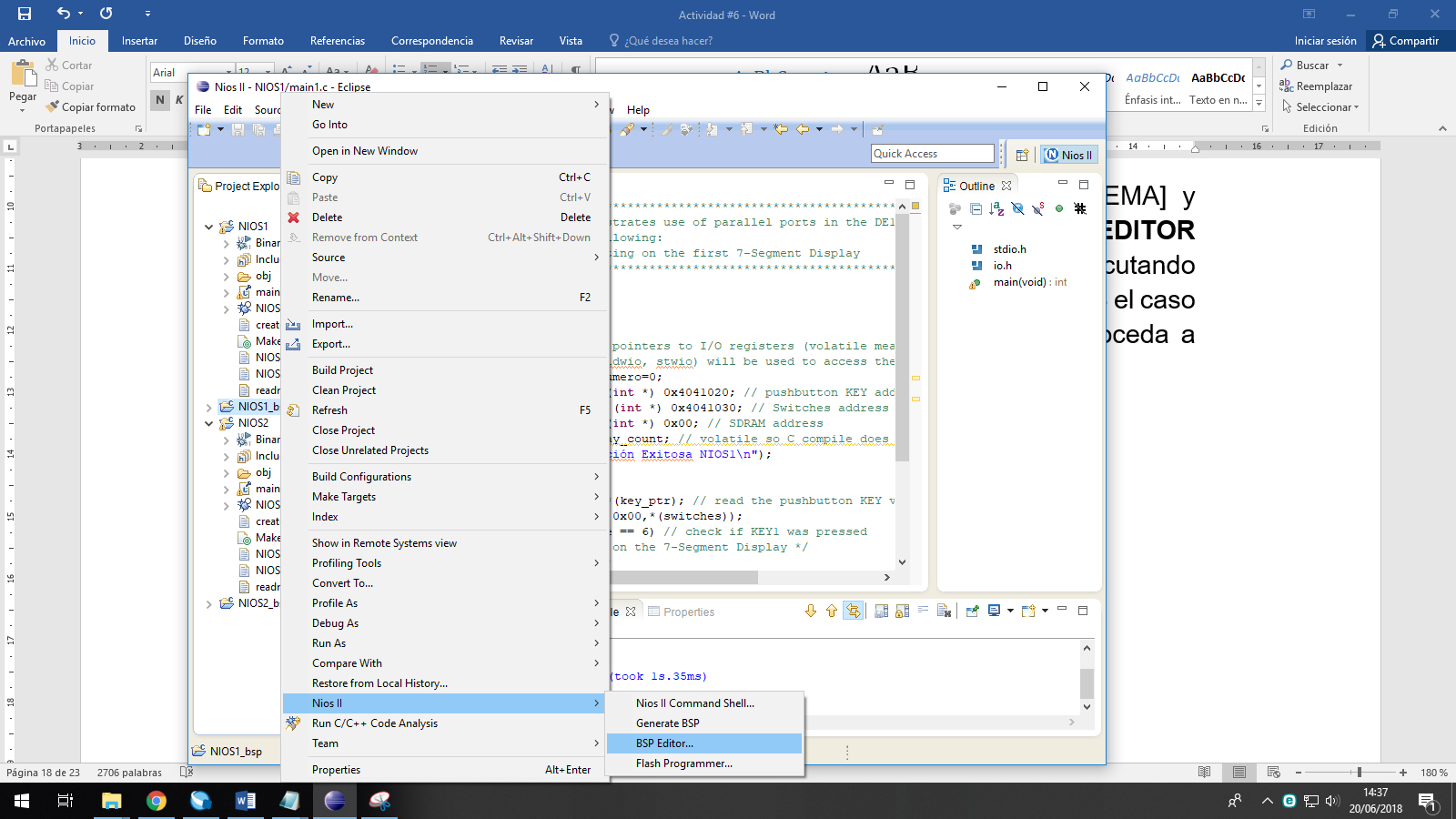


Ilustración 54: Instrucciones para abrir el editor BSP del NIOS1.

1. En la ventana del editor se debe verificar que el **procesador**, se encuentren funcionando desde sus espacios respectivos de memorias; es decir que el **VGA2** se ejecute sobre el **onchip\_mem2**; realizado esto se compila dando clic en **Generate**, como se aprecia en la figura.

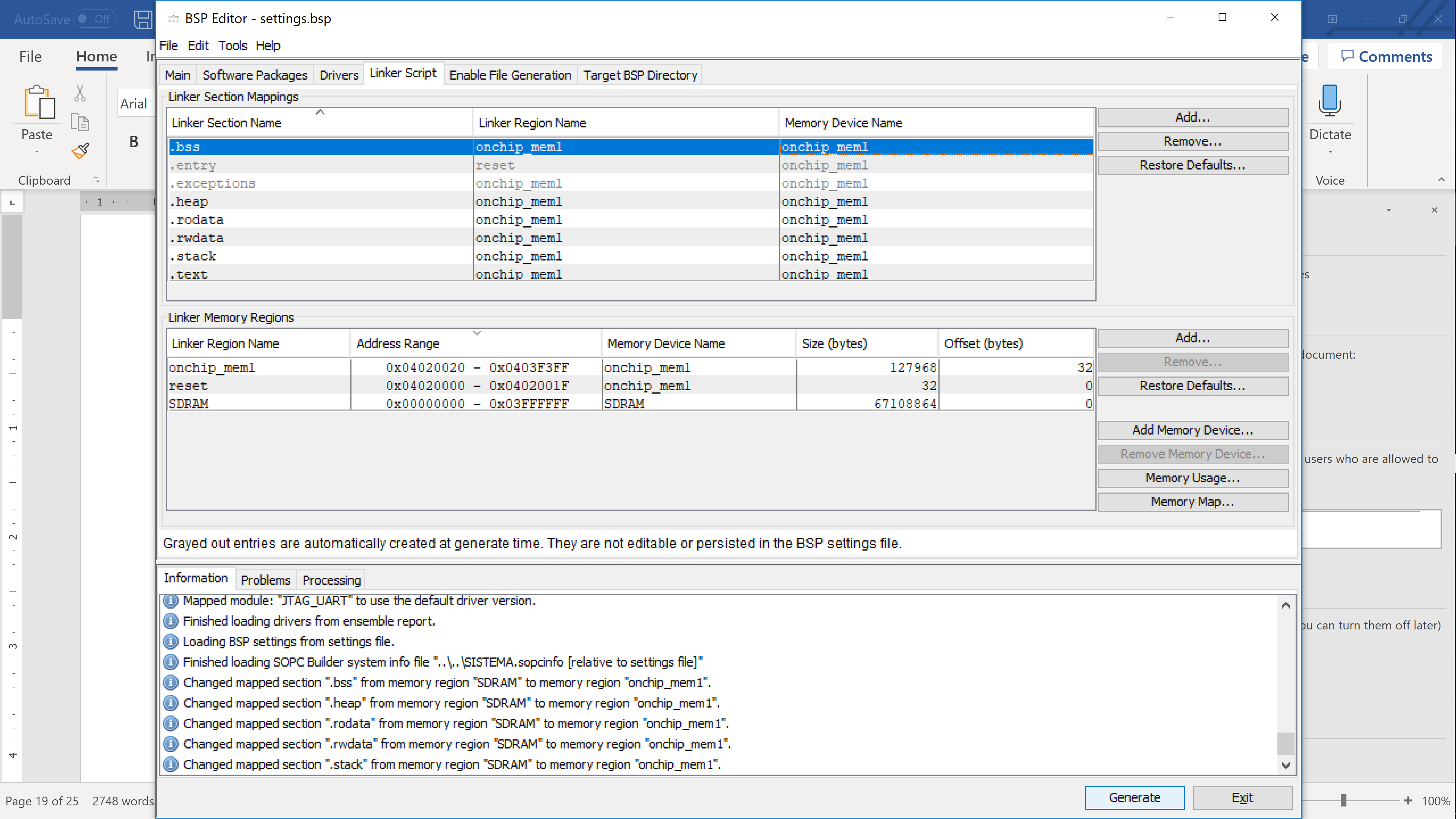


Ilustración 55: Compilación de las configuraciones realizadas.

1. Terminada la compilación, se debe dar clic derecho en la carpera VGA2 y luego buscar la opción **RUN AS** y así luego seleccionar la opción de **NIOS II HARDWARE.**

# RESULTADOS

Se realizó la ejecución de la red neuronal entrenada en uno de los procesadores de la tarjeta de desarrollo; en donde se observó los resultados de predicción de pasos del sujeto en estudio en base a los datos ingresados por consola. En la siguiente imagen se observa el ingreso de datos y número de pasos predicho.

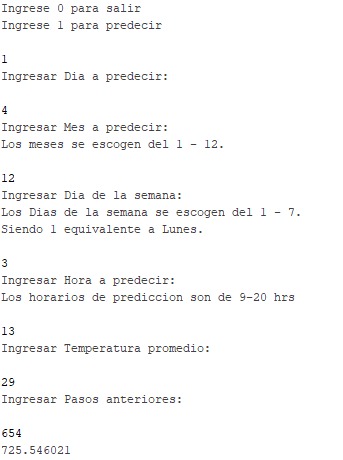


Ilustración : Resultados obtenidos de la red neuronal ejecutándose en el primer procesador.

Además, en cada interacción realizada se muestra el dato predicho y la verificación de los pasos reales realizados en la entra anterior; mostrando así una comparación gráfica mostrando la dispersión de los datos y la efectividad de la red empleada.

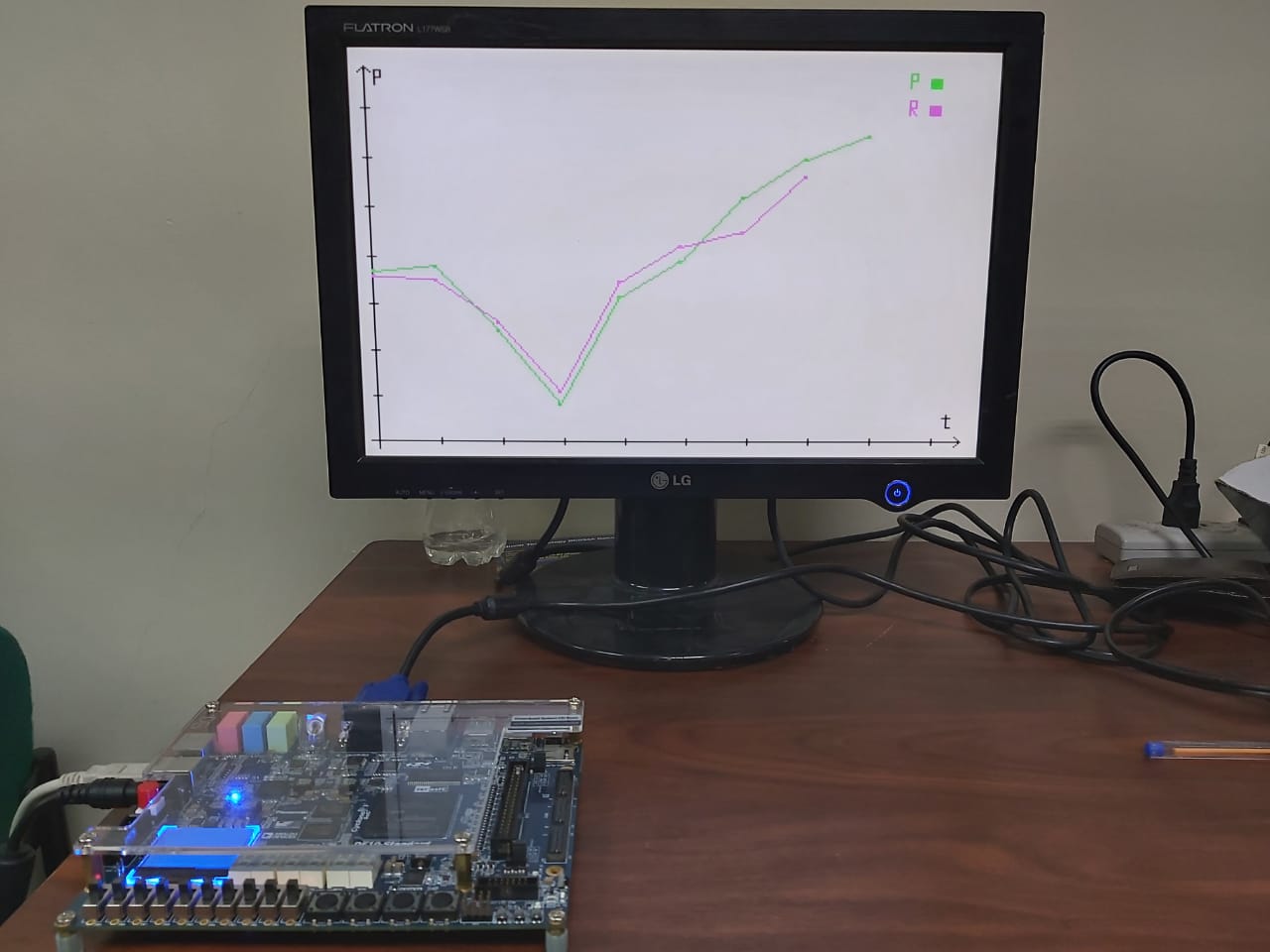


Ilustración : Impresión de resultados por pantalla.