

Sistemas Digitales I

Prácticas con la Tarjeta

DE0 nano

Capítulo del curso: Uso de la Tarjeta de Desarrollo DE0Nano

Objetivos de aprendizaje:


- Describir circuitos MSI usando VHDL.
- Importar archivos a un proyecto en Quartus y utilizarlos en el diseño modular de un sistema digital.
- Implementar un sistema combinatorial utilizando la tarjeta DE0 - Nano.

Duración:

Materiales y herramientas:

- Dip Switches
- Protoboard
- Resistencias
- Leds
- Displays de 7 Segmentos
- Jumpers Macho-Hembra
- Tarjeta DE0-Nano
- Cable Mini USB
- Quartus Prime Lite V19.1

Procedimiento para crear un Nuevo Proyecto en Quartus:

1. Cree una nueva carpeta en el escritorio y copie el contenido de la carpeta Bloques de la carpeta compartida.
2. Ejecute el programa Quartus Primer Lite v19.1 que se encuentra en el escritorio representado por el ícono .
3. Proceda a crear un nuevo Proyecto seleccionando la opción **File**→**New Project Wizard** como se observa en la Figura 1.

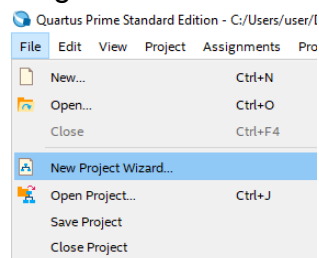


Figura 1. Creación de un nuevo proyecto.

4. El asistente de creación de proyectos mostrará una ventana de inicio; dar clic en “Next” para ir a la siguiente ventana.

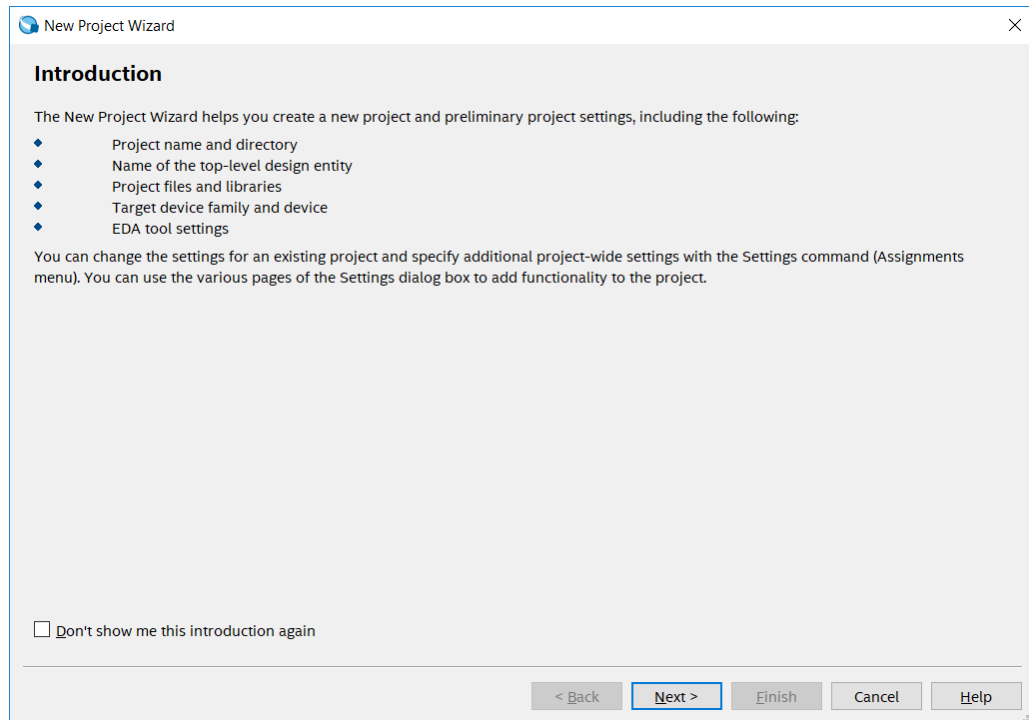


Figura 2. Asistente de Creación de Proyectos de Quartus.

5. Como directorio, seleccione la carpeta del paso #1. **Revise que la ruta y nombre de esta no contenga ni espacios ni caracteres especiales.** Coloque el nombre del proyecto, como se observa en la figura 3. Luego de clic en “next”.

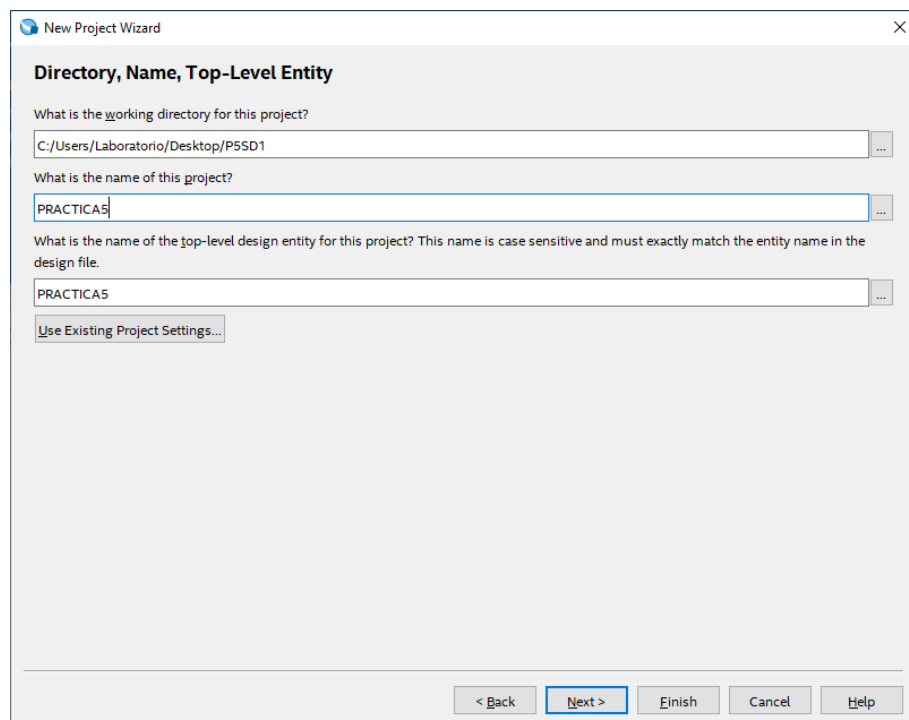


Figura 3. Asignación de Directorio y Nombre del Proyecto.

6. En la siguiente ventana, se selecciona el tipo de proyecto que se está creando. Por ahora, se trabajará con el tipo “Empty project”. Dar clic en Next.

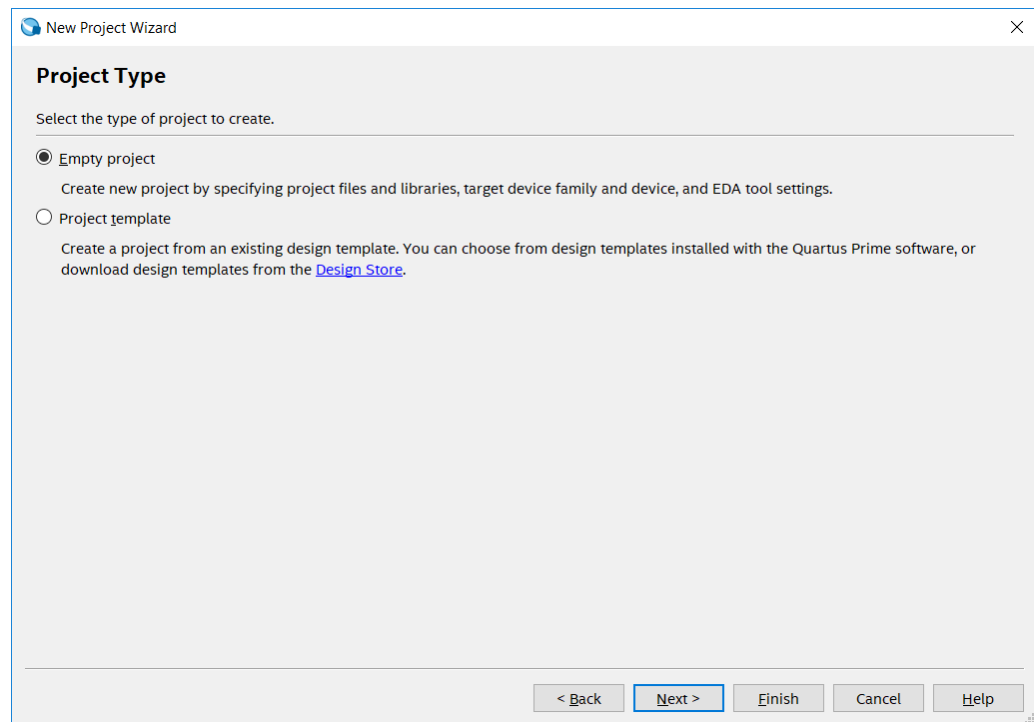


Figura 4. Tipo de Proyecto.

- La siguiente ventana del asistente permite agregar archivos previamente creados al proyecto, de ser necesario haga clic en **Add All** y anada los archivos, y luego de clic en **Next**, caso contrario al no tener archivos para añadir dar click en **Next**.

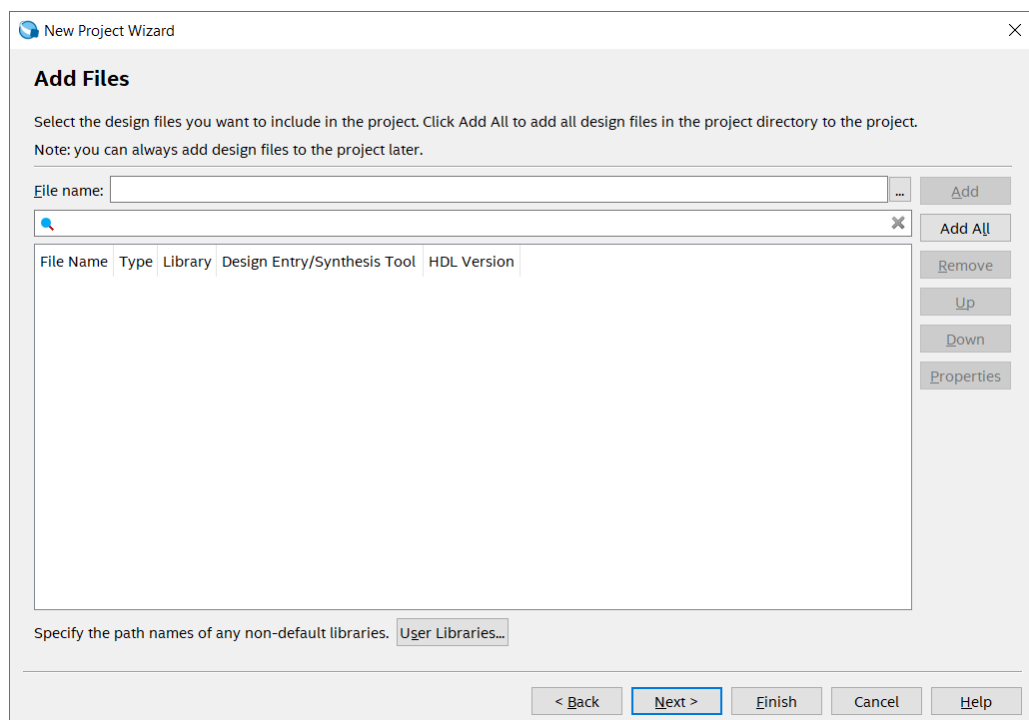


Figura 5. Añadir archivos.

- En la siguiente ventana del asistente, se selecciona el dispositivo por programar; usaremos la tarjeta de desarrollo DE0 Nano. Seleccione la familia **Cyclone IV E** y el nombre del dispositivo de la tarjeta, en este caso corresponde a **EP4CE22F17C6**. De clic en Next.

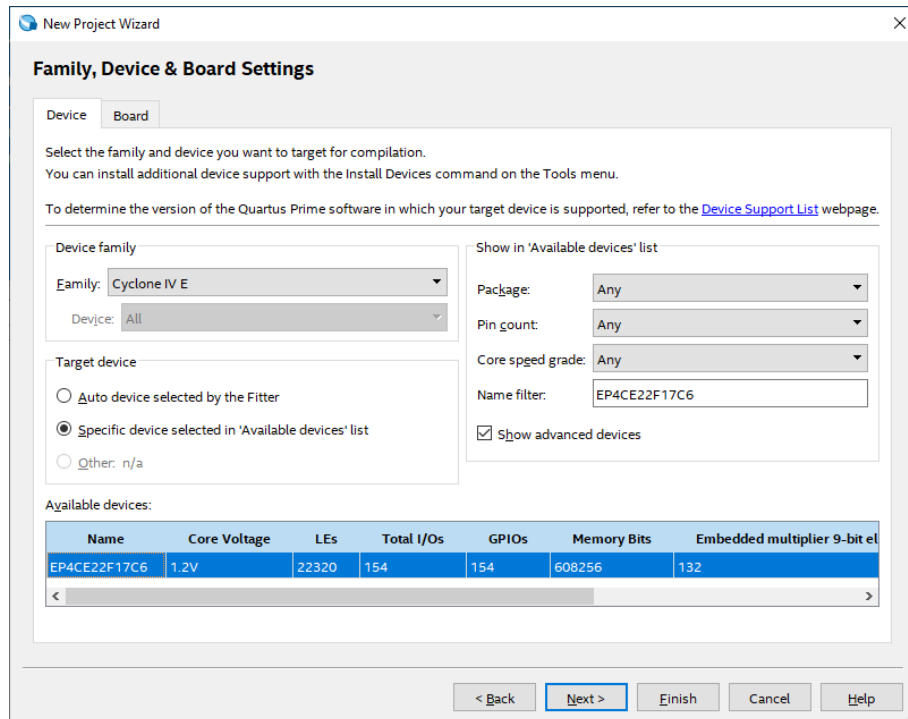


Figura 6. Configuración de la tarjeta.

9. En la ventana de EDA Tool Settings, daremos clic en Next.

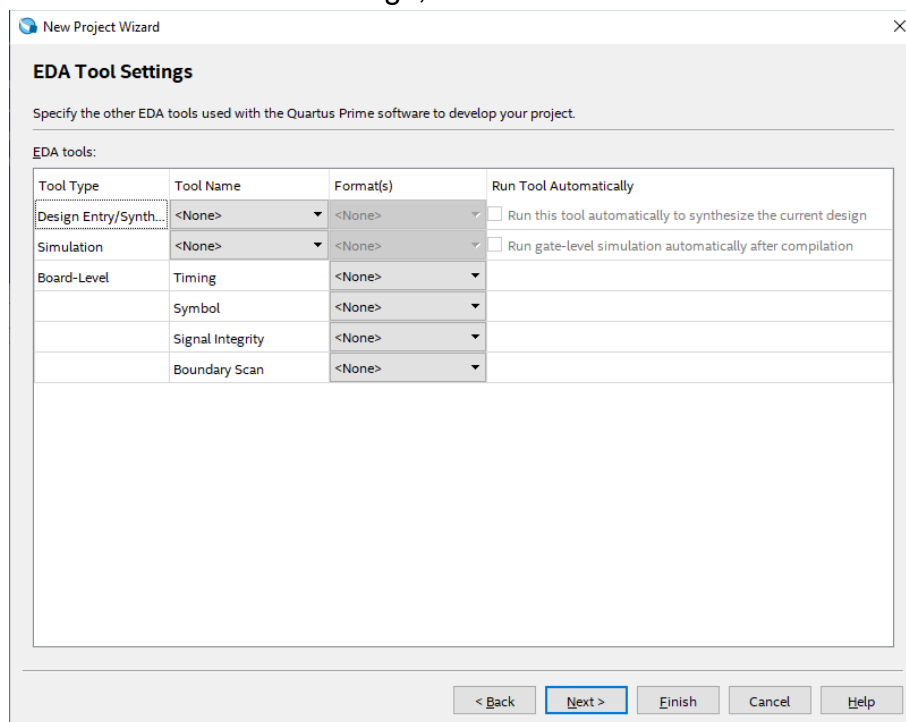


Figura 7. Configuración de la herramienta EDA.

10. Cree un nuevo archivo .VHD y escriba el código de la practica asignada para el uso de la tarjeta, póngalo como mayor jerarquía (**Top-Level-Entity**) y compile.

Practica #1: Encendido y Apagado de un led con un switch

11. Una vez creado el archivo .VHD con el nombre Switch_LED, hay que recordar que es el mismo nombre con el cual se llama la sección entity. póngalo como mayor jerarquía (**Top-Level-Entity**) y compile.

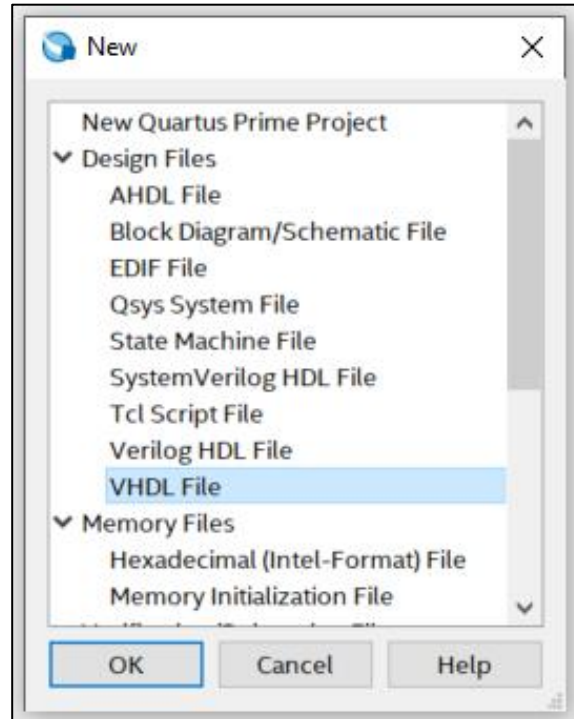


Figura 8. Creación de archivo VHDL

```

Switch_LED.vhd
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity Switch_LED is
5  Port (
6      Switch : in STD_LOGIC; -- Entrada del interruptor
7      LED    : out STD_LOGIC -- Salida del LED
8  );
9  end Switch_LED;
10
11 architecture Behavioral of Switch_LED is
12 begin
13     process (Switch)
14     begin
15         if Switch = '1' then
16             -- Enciende el LED cuando el interruptor está en posición "1"
17             LED <= '1';
18         else
19             -- Apaga el LED cuando el interruptor está en posición "0"
20             LED <= '0';
21         end if;
22     end process;
23 end Behavioral;
24

```

Figura 9. Código VHDL de Switch_LED

12. Haga clic derecho en el archivo Switch_LED, y seleccione “Create Symbol File for Current File” como se muestra en la imagen a continuación.

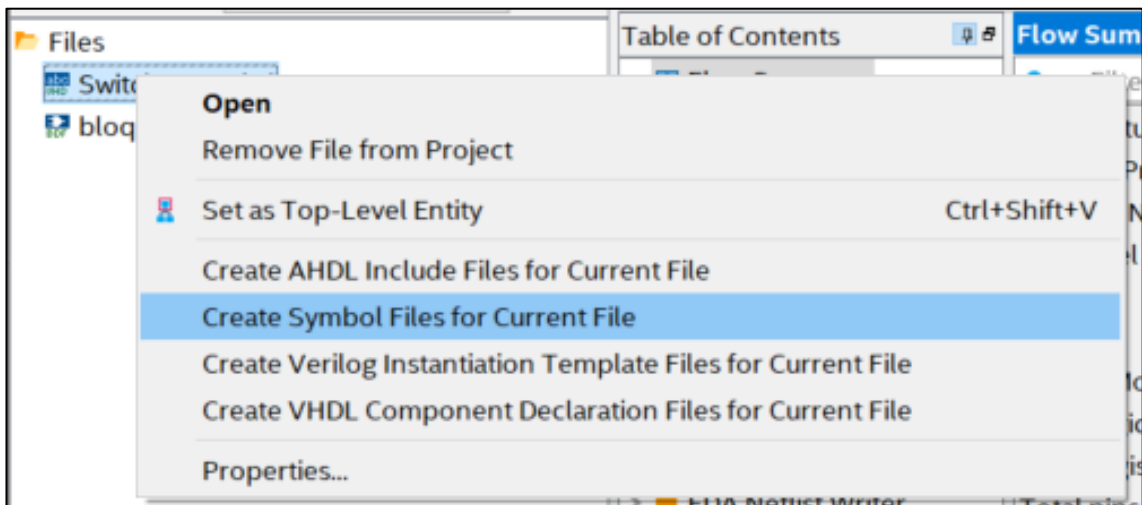


Figura 10. Creación del bloque de símbolo

13. Crear un nuevo archivo .bdf en el cual realizaremos las conexiones y el diagrama de bloques en partición funcional de nuestro proyecto.

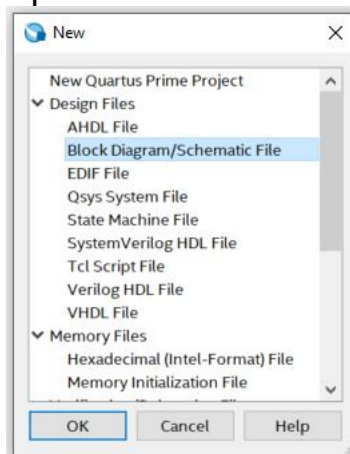


Figura 11. Creación de archivo .BDF

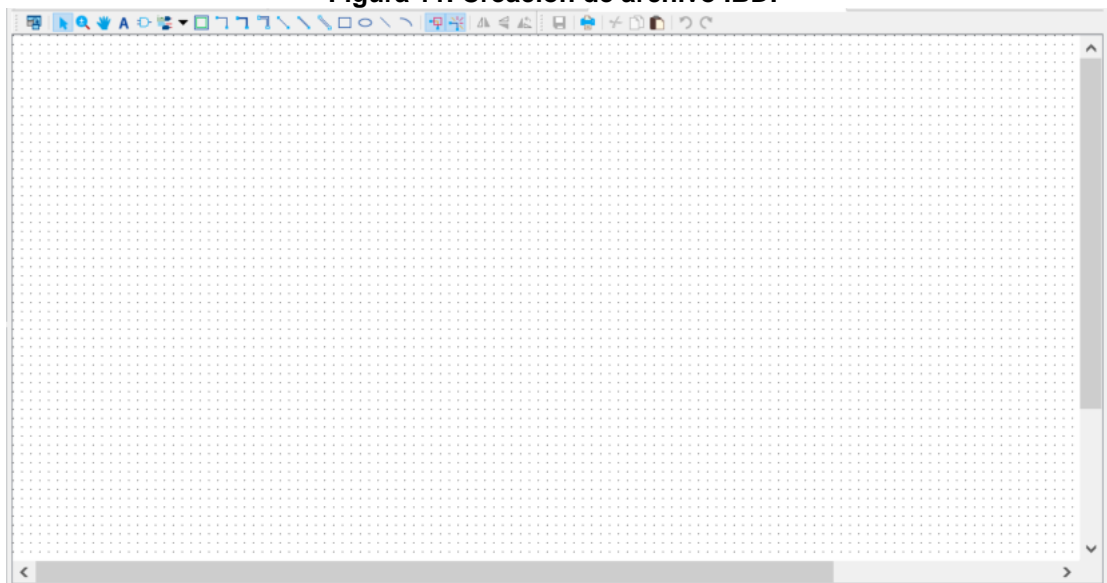


Figura 12. Área de trabajo para el diagrama de bloques

14. Daremos click en el icono llamado **Symbol Tool** para que podamos abrir las librerías de nuestro proyecto y componente de Quartus.

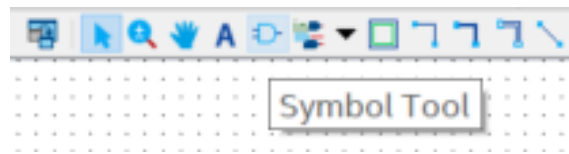


Figura 13. Icono Symbol Tool

15. Para encontrar nuestro bloque creado iremos a la carpeta de nuestro proyecto llamada Project, en la cual tendremos siempre todos nuestros bloques creados para una vez seleccionarlos y dar click en ok.

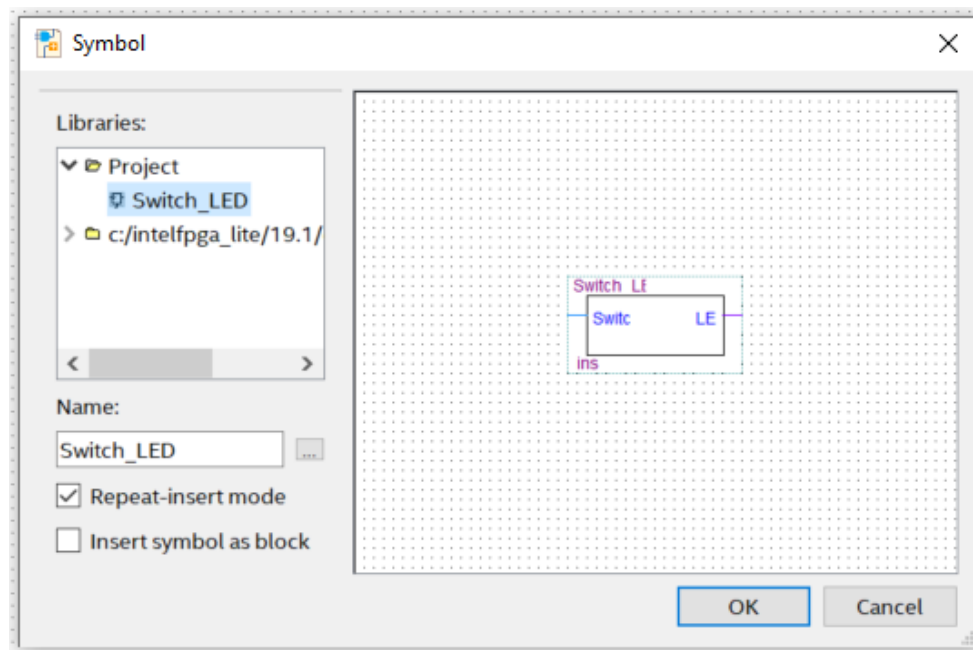


Figura 14. Símbolo del Bloque Switch_LED

16. Una vez posicionemos el bloque en nuestra área de trabajo, le daremos click derecho sobre el y seleccionaremos la opción **Generate Pins for Symbol Ports** para poder generar las entradas y salidas del bloque.

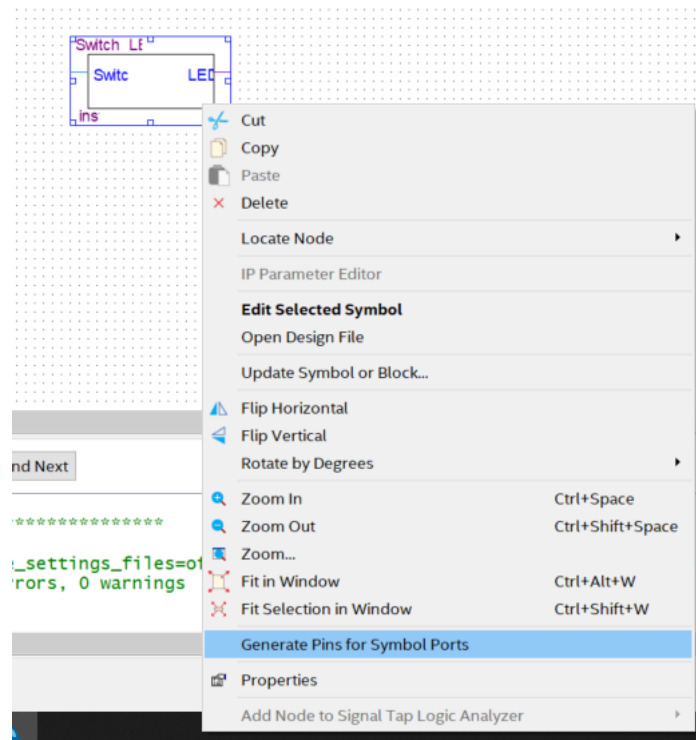


Figura 15. Generando los pines de entradas y salida con Generate Pins for Symbol Ports

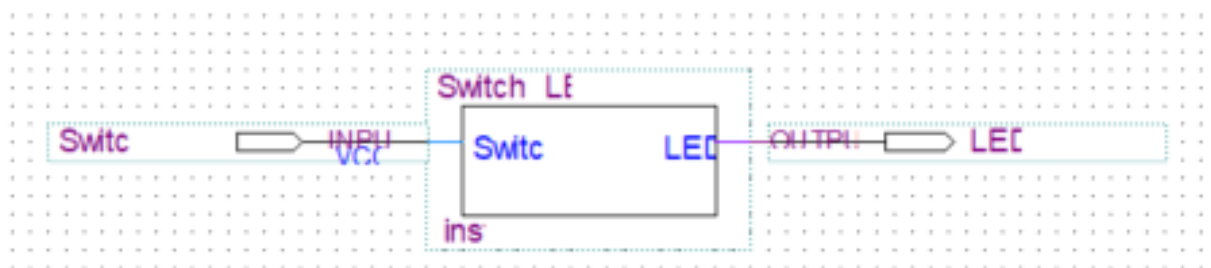


Figura 16. Bloque con entradas y salidas.

17. Guarde el archivo dentro de la misma carpeta de proyecto con un nombre acorde a la función del bloque, seleccione el archivo como **Top-Level-Entity** y compílelo.
18. Haga clic en el menú **Assignments** y seleccione **Pin-Planner**.

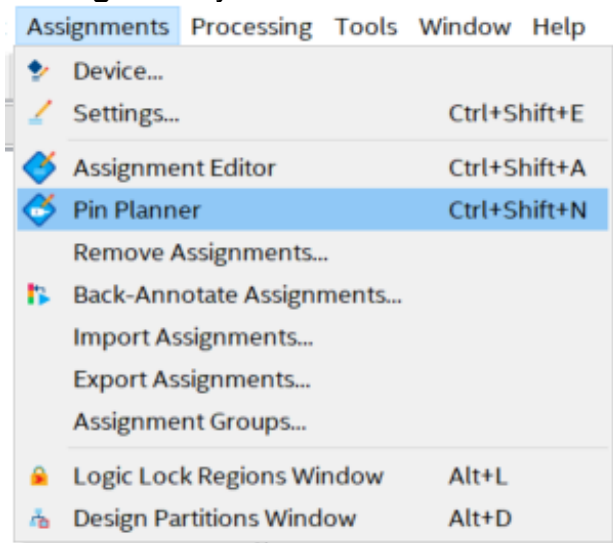


Figura 17. Pin Planner

19. Aparecerá una nueva ventana como la que se muestra a continuación. Esta ventana permite asignar los pines que usaremos de la tarjeta DE0-Nano asociado a las entradas y salidas que se crearon del proyecto. Verifique los pines estén asignados como se muestra en la imagen.

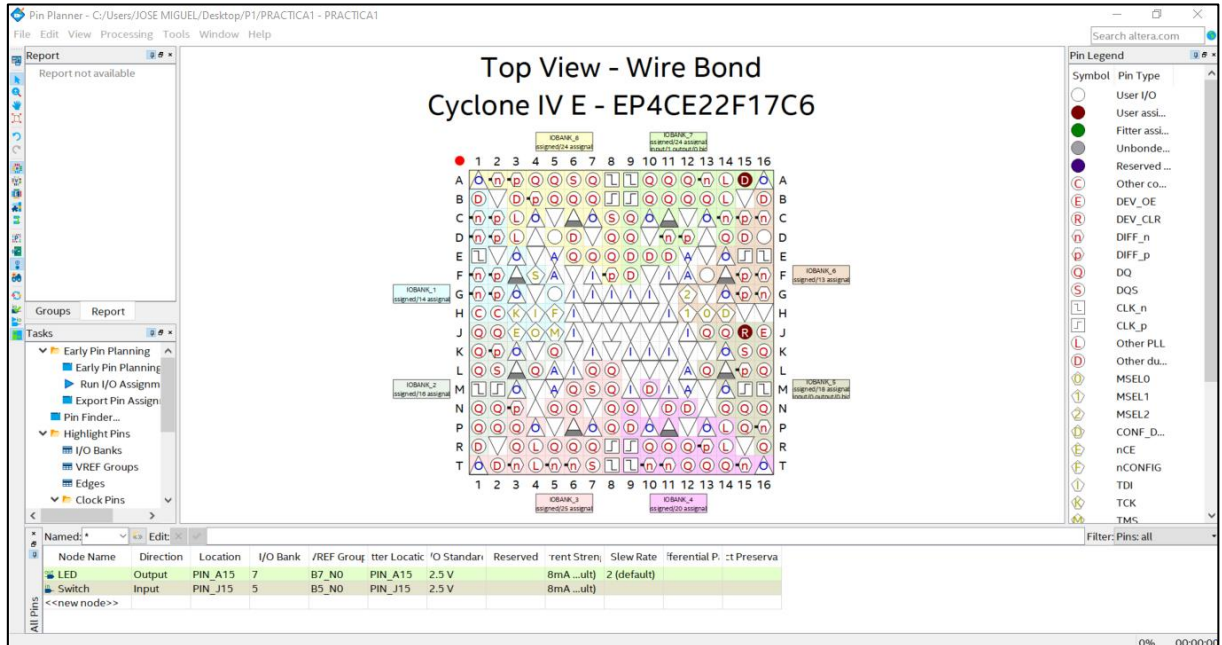


Figura 18. Asignación de pines de entradas y de salida.

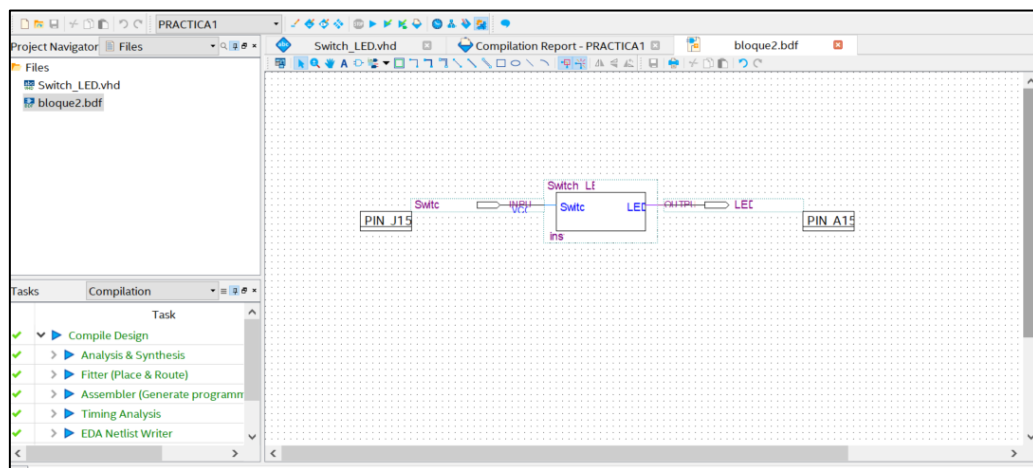




Figura 19. Visualización de diagrama de bloques y pines de entrada y de salida

20. Cerrar la ventana y volver a compilar el archivo **.bdf** con el símbolo 
21. Con estas configuraciones lo que se va a usar un botón como entrada y un led como salida que ya vienen incorporado en la tarjeta de desarrollo.

Entradas		Salidas	
Nombre de GPIO	Pin en FPGA	Nombre de GPIO	Pin en FPGA
KEY[0]	PIN_J15	LED[0]	PIN_A15

22. Conectar el bus de datos al conector JP1 de la tarjeta de desarrollo DE0 Nano
1. Luego se debe configurar la FPGA:

1. Conecte la Tarjeta de desarrollo a la PC.
2. Asegúrese de haber realizado la compilación del circuito.
3. Click en: **Tools** → **Programmer**, o en el ícono  del menú de herramientas. Se visualizará una ventana similar a la que se encuentra en la Figura 7.

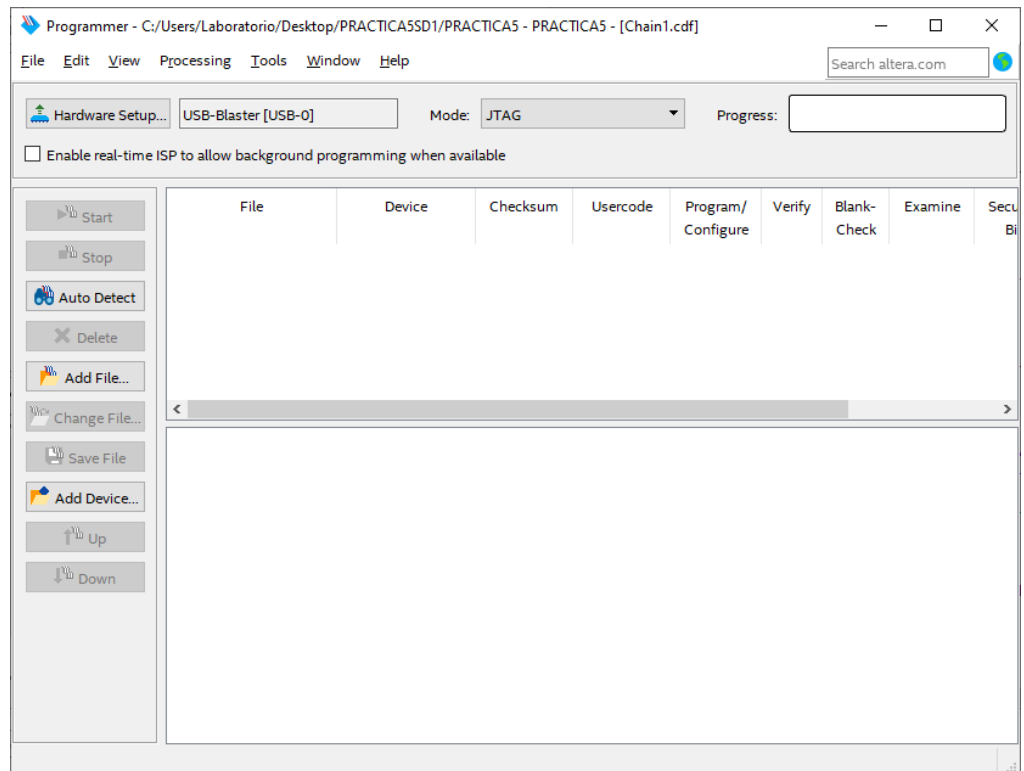


Figura 20. Ventana Programmer de Quartus Prime.

NOTA: Asegúrese que en la opción **Hardware Setup** se encuentre **USB-Blaster [USB-0]**.

4. Hacer click en **Add file**, y en la carpeta **Output_files** seleccione el archivo **.sof**
5. Poner un visto haciendo clic en las 3 casillas: **Program/Configure**, **Verify** y **Blank-Check**.
6. Asegúrese que en la casilla **Hardware Setup** esté **USB-Blaster**.
7. Clic en **Start**.

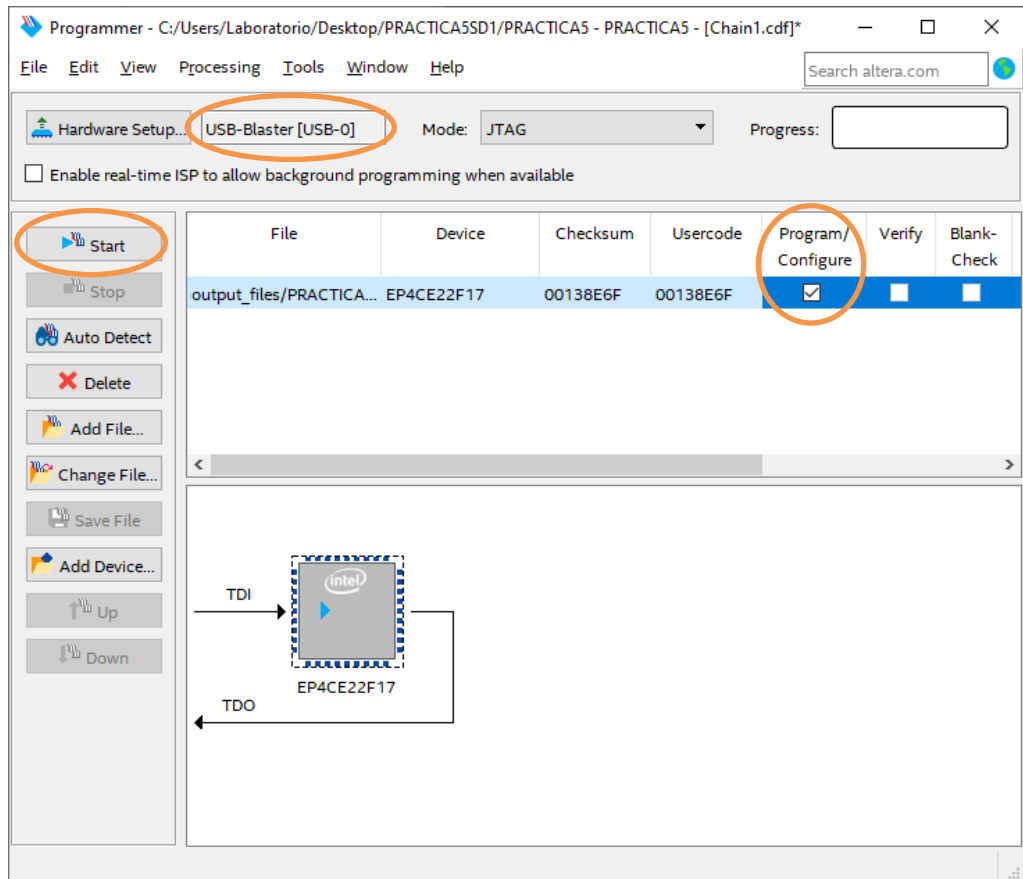


Figura 21. Ventana Programmer de Quartus Prime con el archivo a programar.

8. Verifique que en la barra **Progress** se presente el mensaje **Successful**.

23. Presione las botonerías y compruebe el funcionamiento del circuito digital, como cambia sus salidas dependiendo de las entradas. Mostrar al profesor.

24. Una alternativa a la practica seria usar switch o botoneras y leds externos a la tarjeta, lo que nos llevaría a conocer mejor las conexiones de pines y descripción de la tarjeta de desarrollo, y que a partir del paso 19 seria de la siguiente manera:

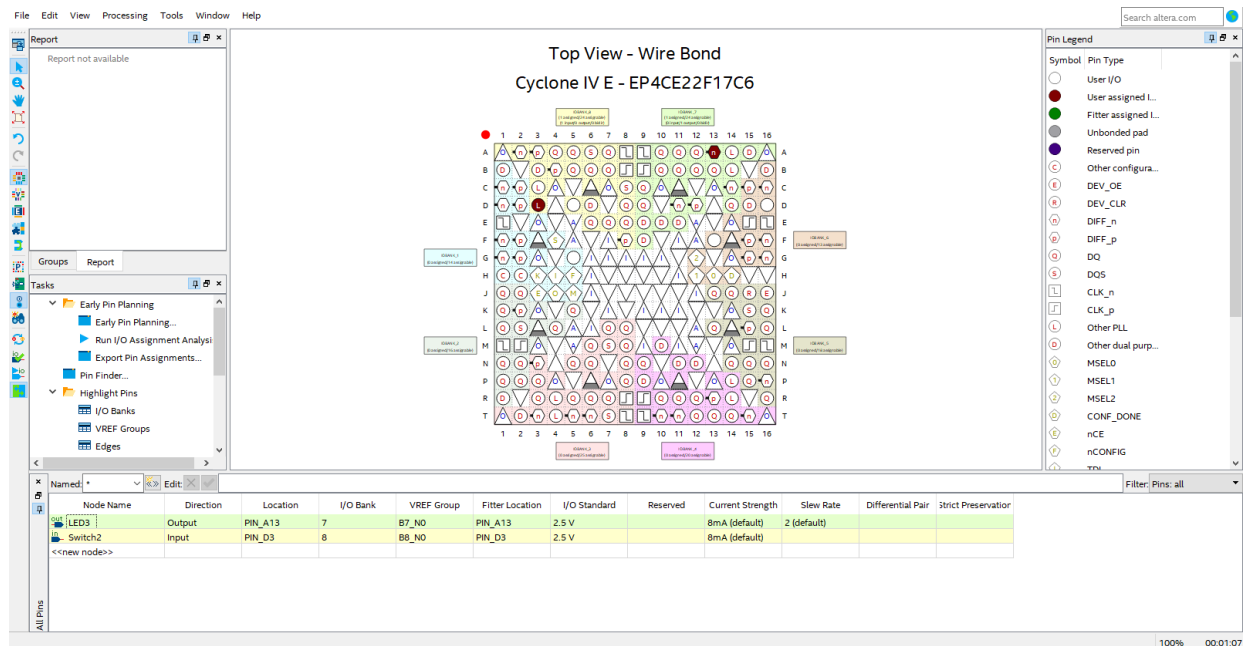


Figura 22. Asignación de pines de entradas y de salida.

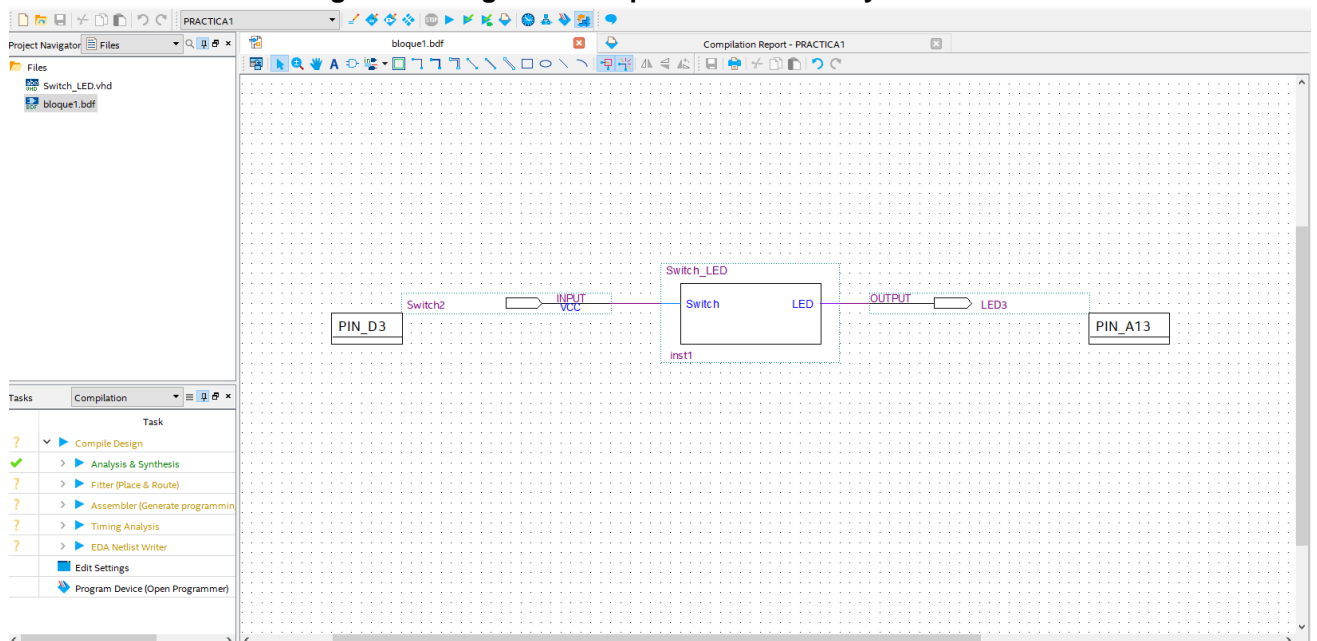


Figura 23. Visualización de diagrama de bloques y pines de entrada y de salida

25. Cerrar la ventana y volver a compilar el archivo **.bdf** con el símbolo
26. Con estas configuraciones lo que se va a usar un botón como entrada y un led como salida que ya vienen incorporados en la tarjeta de desarrollo.

Entradas		Salidas	
Nombre de GPIO	Pin en FPGA	Nombre de GPIO	Pin en FPGA
GPIO_00	PIN_D3	LED[1]	PIN_A13

Bibliografía:

Elaborado por:
 Ing. José Miguel Larrea Gando
 Jefe del Laboratorio Sistemas Digitales Básicos

27. El esquemático del ejercicio sería el siguiente:

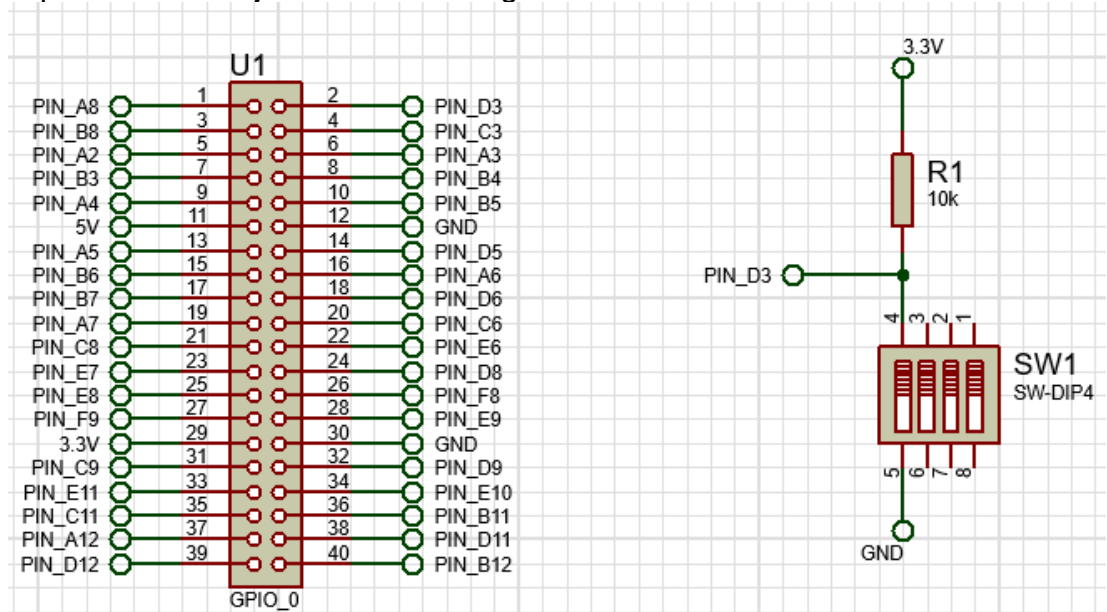


Figura 24. Esquemático del ejercicio.

28. Por último, paso después de tener la programación y el armado físico, se carga el programa a la tarjeta de desarrollo y verificar su funcionamiento. Mostrar al Maestro.

Actividades:

1. cambiar el pin de entrada y ejecutar el programa con botonera o switch y leds externos.
2. cambiar la entrada y la salida para poder conectar una resistencia con led externo.

Bibliografía:

[1]. Fundamentos de Lógica Digital, Stephen Brown & Zvonko Vranesic, Segunda Edición, Mc.Graw Hill, 2009.

[2]. Sistemas Digitales: Principios y Aplicaciones, Ronald Tocci, Octava Edición, Prentice Hall, 2003.

[3]. Digital Design with RTL Design, Verilog and VHDL, Frank Vahid, Second Edition, John Wiley and Sons, 2010.

[4]. DE10-Lite User Manual, Terasic Inc, 2017.