

Uso del catalogo IP

Introducción

En esta práctica se utilizará el catalogo IP para generar un recurso de temporización que permita generar una señal de 50 MHz a partir del oscilador de 125MHz disponible. El mismo será aplicado en el receptor serie sin modificar sus parámetros de temporización.

Objetivos

- Usar el catalogo IP para generar un elemento de temporización
- Instanciar el elemento generado
- Generar el archivo de configuración y verificar la funcionalidad en el hardware

Procedimiento

Descripción del Sistema

El sistema consiste en un receptor serie asíncrono (UART) que recibe caracteres y muestra la representación binaria de la parte baja del carácter en 4 Leds (es el mismo utilizado en la Práctica 2). Cuando se presiona un botón, se representa la parte alta del carácter. En la Figura 1 se ve el diagrama en bloques del sistema.

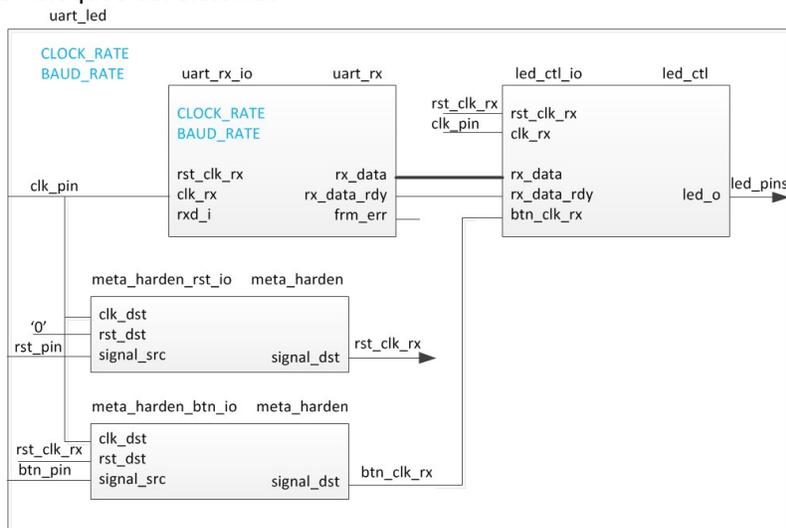
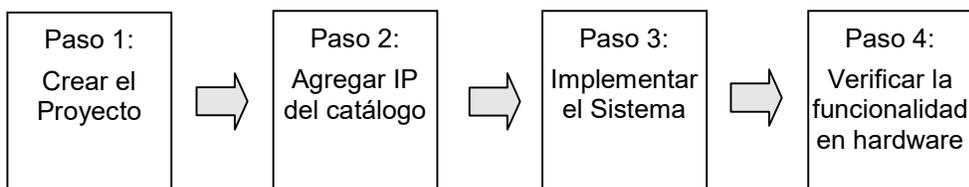


Figura 1. Sistema a sintetizar

Hoja de Ruta de la Práctica



Crear el Proyecto

Paso 1

1-1. Iniciar la herramienta Vivado y crear un proyecto con la placa Arty Z7. Agregar los archivos HDL y las restricciones `uart_led_pins_ArtyZ7.xdc` y `uart_led_timing_ArtyZ7.xdc`.

1-1-1. Iniciar la herramienta Vivado

1-1-2. Presionar **Create Project** para iniciar el asistente. Aparecerá el cuadro de dialogo *Create A New Vivado Project*. Presionar **Next**.

1-1-3. Presionar el botón de navegación del campo *Project location* del formulario **New Project**, elegir la ubicación del proyecto y presionar **Select**.

1-1-4. Ingresar el nombre del proyecto (Práctica04) en el campo *Project name*. Verificar que la opción *Create Project Subdirectory* este seleccionada. Presionar **Next**.

1-1-5. Seleccionar la opción **RTL Project** en el formulario *Project Type*, y presionar **Next**.

1-1-6. Verificar que este seleccionada la opción **VHDL** en los menús desplegables *Target Language* y *Simulator Language* en el formulario *Add Sources*.

1-1-7. Presionar el botón **Add Files** y agregar los archivos vhd `led_ctl.vhd`, `meta_harden.vhd`, `uart_baud_gen.vhd`, `uart_led.vhd`, `uart_rx.vhd`, `uart_rx_ctl.vhd` and `uart_top.vhd`, (son los mismos archivos de la Practica02). Presionar **OK**, y después presionar **Next** para pasar al formulario *Add Constraints*.

1-1-8. Presionar el botón **Add File** y agregar los archivos de restricciones `uart_led_timing_ArtyZ7.xdc` y `uart_led_pins_ArtyZ7.xdc`. Presionar **Ok**. Después presionar **Next**.

Estos archivos de restricciones asignan las restricciones de temporización (periodo, retardo de entrada y retardo de salida) del sistema y la ubicación física de los pines del modulo principal (`UART_TOP`).

1-1-9. En el formulario *Default Part*, seleccionar la pestaña **Boards**. En el recuadro *Vendor* seleccionar **digilentinc.com**, en el recuadro *Name* seleccionar la placa **Arty Z7-10**. En el recuadro *Display Name* seleccionar la fila con la placa Arty Z7-10. Presionar **Next**.

1-1-10. Presionar **Finish** para crear el proyecto.

1-2. Modificar los archivos fuente para adaptarlos a la placa Arty-Z7.

Abrir el archivo de restricciones `uart_led_timing_ArtyZ7.xdc`. Modificar el reloj para que su frecuencia sea de 125MHz. Para ello reemplazar la línea 4:

```
create_clock -period 8.000 -name clk_pin -waveform {0.000 4.000} [get_ports clk_pin]
```

Y la línea 7:

```
create_clock -period 20.000 -name virtual_clock
```

Guardar el archivo de restricciones una vez hecha la modificación

- 1-2-1.** Abrir el archivo de restricciones *uart_led_pins_ArtyZ7.xdc*. Agregar un pin de Tx para tener eco de lo enviado. Para ello agregar la línea 22:

```
set_property -dict { PACKAGE_PIN Y19 IOSTANDARD LVCMOS33 } [get_ports { txd_pin }];
```

Guardar el archivo de restricciones una vez hecha la modificación

- 1-2-2.** Abrir el archivo HDL *uart_top.vhd*. Agregar el puerto de Tx de la uart. Para ello insertar en la línea 31 (entre *rxn_pin* y *led_pins*)

```
txd_pin: out std_logic; -- Uart output
```

Conectar la entrada de *rxn* a la salida *txd*. Para ello insertar en la línea 53 (antes de *Begin*)

```
signal loopback: std_logic; (no olvidar el punto y coma)
```

Luego insertar en la línea 55 y 56:

```
loopback <= not rxn_pin;  
txd_pin <= not loopback; (no olvidar el punto y coma)
```

Modificar la línea 60:

```
CLOCK_RATE => 50E6
```

Y la línea 63:

```
clk_pin => clk50
```

Guardar el archivo *HDL uard_top.vhd* (Ignorar el error, es debido a que todavía no esta declarada la señal *clk50*)

- 1-2-3.** Abrir el archivo HDL *led_ctl.vhd*. Modificar las líneas 62 y 64; en la línea 62 es **char_data(7 downto 4)** y en la línea 64 es **char_data (3 downto 0)** Guardar el archivo

Agregar IP del Catalogo

Paso 2

2-1. Iniciar la herramienta IP Catalog y agregar un recurso de temporización.

- 2-1-1.** En el menú “Flow Navigator” , en la sección “Project Manager” , dar click en “IP Catalog”, se abrirá la ventana “IP Catalog”

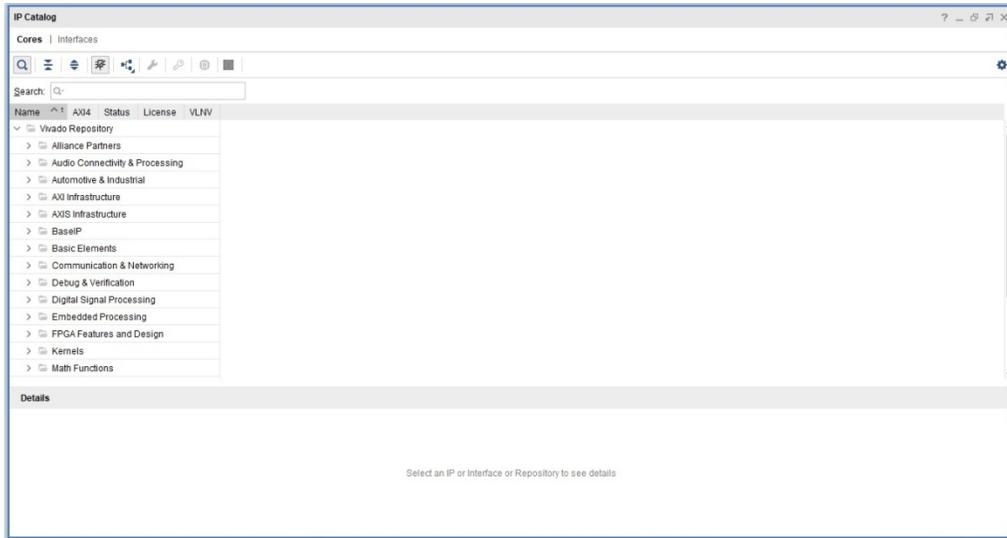


Figura 2: IP Catalog

- 2-1-2. Dar doble click en el Clocking Wizard, que se encuentra en “FPGA Features and Design” -> “Clocking” -> “Clocking Wizard”

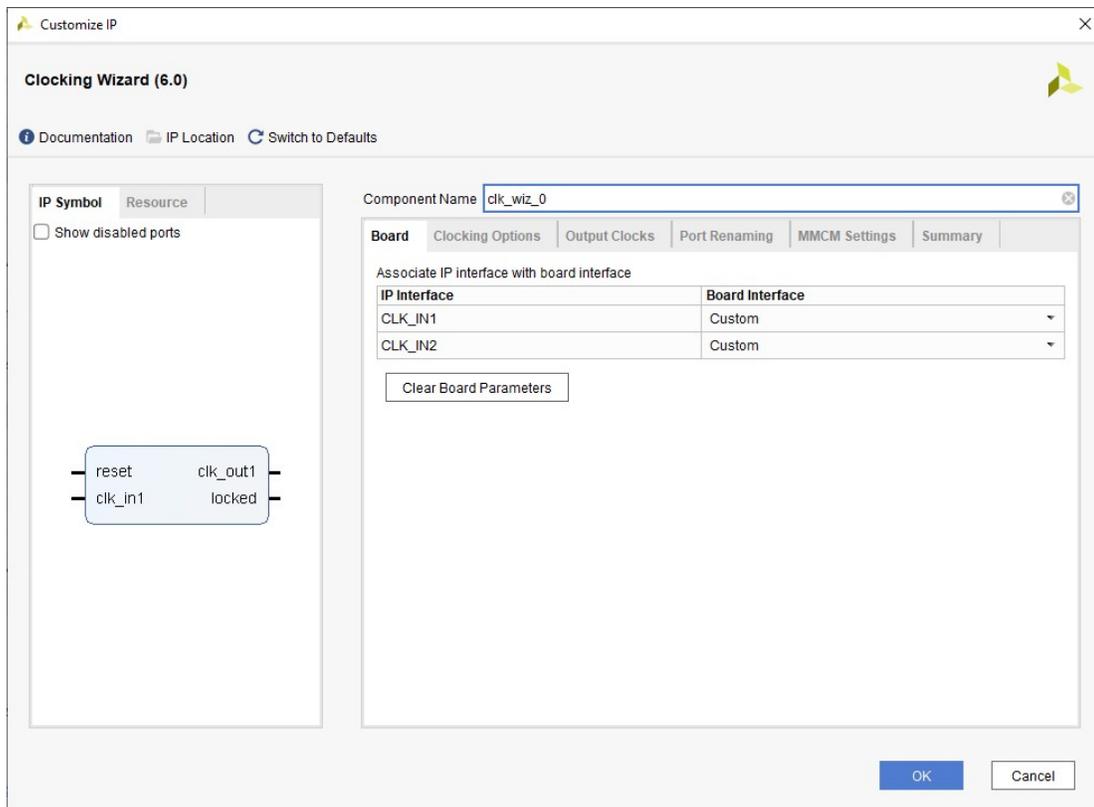


Figura 3: Clocking Wizard

- 2-1-3.** En la pestaña “Clocking Options”, deslizar hasta la sección “Input Clock information” y configurar la entrada “clk_in1” a 125MHz

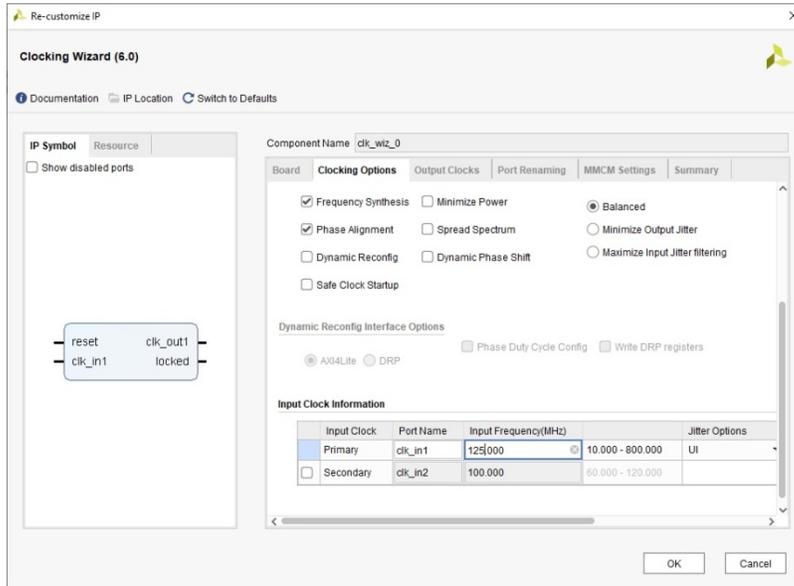


Figura 4: Configurando el reloj de entrada

- 2-1-4.** En la pestaña “Output Clocks”, en la fila “clk_out1”, cambiar el valor de la columna “Requested” a 50 MHz. De esta manera se genera una señal de reloj de 50MHz a partir del oscilador de entrada de 125MHz.

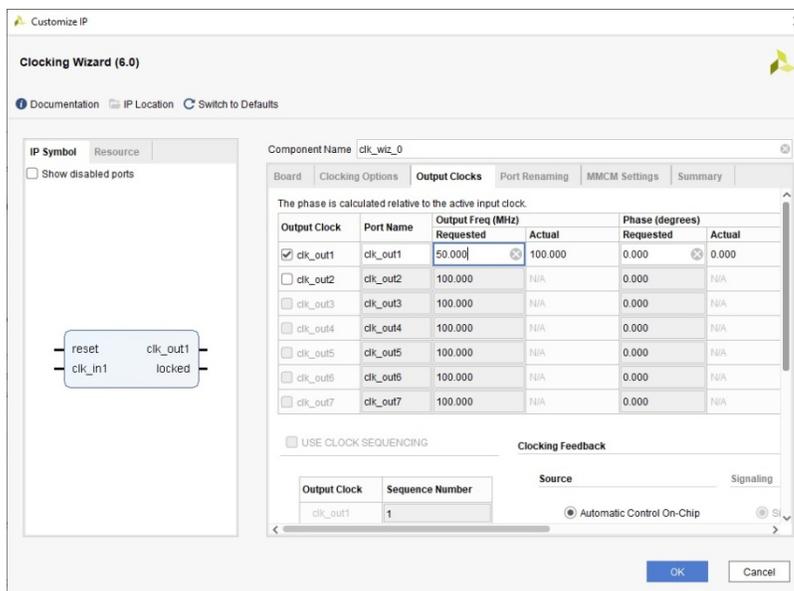


Figura 5: Configurando el reloj de salida

Presionar el botón “OK”. Aparecera una ventana de resumen de los elementos que se crearan.

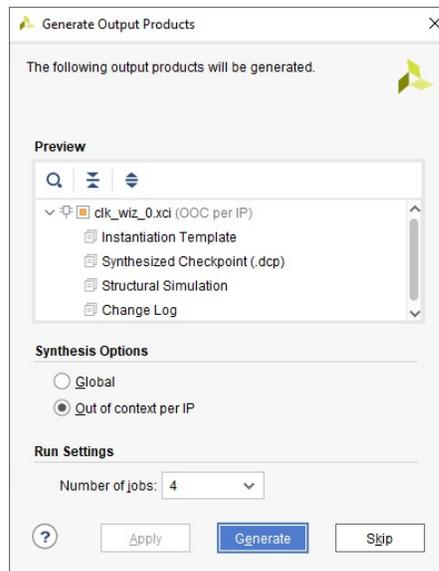


Figura 6: Productos generados

Presionar el botón “Generate” para generar el bloque IP

En el recuadro “Sources” aparecera un elemento “clk_wiz_0” dentro de “Design Sources”

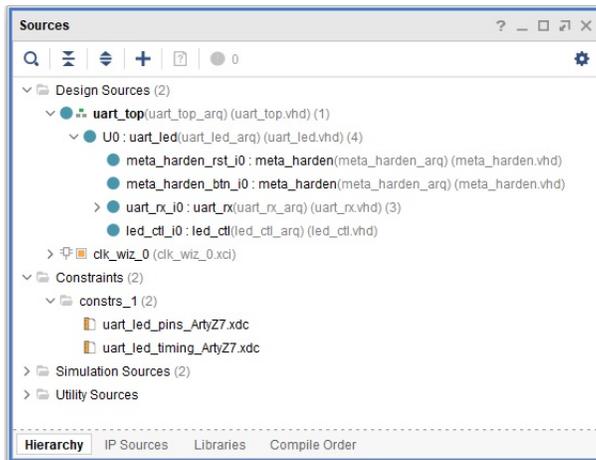


Figura 7: Nuevo elemento en “Sources”

2-2. Agregar el recurso de temporización al proyecto.

- 2-2-1.** Para agregar el recurso de temporización, es necesario utilizar una plantilla de instanciado. Como parte de los productos generados hay plantillas de instanciado en VHDL y Verilog. En este caso se utilizará la plantilla en VHDL. Para acceder a dicha plantilla se abre el archivo correspondiente utilizando el menú “Open File” (símbolo de archivo debajo del menú “File”). La ubicación del archivo es Practica04\Practica04.ip_user_files\ip\clk_wiz_0\clk_wiz_0_stub.vhdl.

```

1  |----- Copyright 1986-2018 Xilinx, Inc. All Rights Reserved.
2  |-----
3  |----- Tool Version: Vivado v.2018.3 (win64) Build 2405991 Thu Dec 6 23:38:27 MST 2018
4  |----- Date       : Sun Sep 27 16:17:47 2020
5  |----- Host        : Laptop running 64-bit major release (build 9200)
6  |----- Command     : write_vhdl -force -mode synth_stub
7  |----- Design      : clk_wiz_0
8  |----- Purpose     : Stub declaration of top-level module interface
9  |----- Device      : xc7z020clg400-1
10 |-----
11 |-----
12 | library IEEE;
13 | use IEEE.STD_LOGIC_1164.ALL;
14 |
15 | entity clk_wiz_0 is
16 |   Port (
17 |     clk_out1 : out STD_LOGIC;
18 |     reset    : in  STD_LOGIC;
19 |     locked   : out STD_LOGIC;
20 |     clk_in1  : in  STD_LOGIC
21 |   );
22 |
23 | end clk_wiz_0;
24 |
25 | architecture stub of clk_wiz_0 is
26 |   attribute syn_black_box : boolean;
27 |   attribute black_box_ped_pin : string;
28 |   attribute syn_black_box of stub : architecture is true;
29 |   attribute black_box_ped_pin of stub : architecture is "clk_out1,reset,locked,clk_in1";
30 | begin

```

Figura 8: Plantilla de instanciado en VHDL

2-2-2. Incorporar la plantilla de instanciado mediante el menú “Add Sources” de manera similar a cuando se agregaron los fuentes del proyecto al principio (Project Manager -> Add Sources), seleccionar “Add or create design sources”, presionar “Next”. En la siguiente ventana seleccionar “Add Files” e incorporar el archivo “clk_wiz_0_stub.vhdl” y presionar “OK”. Luego presionar el botón “Finish”

2-2-3. Instanciar el elemento de temporización en el archivo “uart_top.vhdl”, para ello insertar entre las líneas 53 y 54 (antes del “Begin”)

signal clk50: std_logic; (no olvidar el punto y coma)

2-2-4. Insertar en la línea 38

```

component clk_wiz_0 is
  Port (
    clk_out1 : out STD_LOGIC;
    reset    : in  STD_LOGIC;
    locked   : out STD_LOGIC;
    clk_in1  : in  STD_LOGIC
  );
end component;

```

2-2-5. Insertar en la línea 65 (antes de U0)

```

clk0: clk_wiz_0
  port map(
    clk_out1 => clk50,
    reset => rst_pin,
    locked => open,
    clk_in1 => clk_pin
);

```

- 2-2-6.** Elaborar el diseño y ver el diagrama en bloques del mismo para verificar el agregado del elemento de temporización, para ello dar click en “Schematic” dentro del menú “Project Manager”, en la sección “RTL Analysis”. Presionar “OK” en el cuadro de dialogo

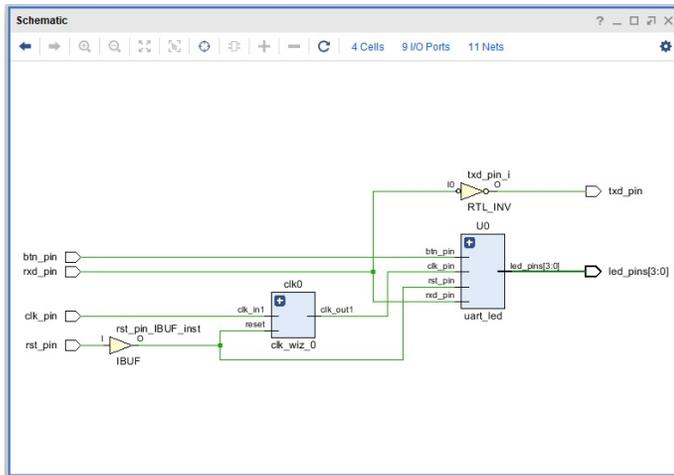


Figura 9: Diagrama en bloques con el recurso de temporización agregado

Implementar el Sistema

Paso 3

3-1. Ejecutar los procesos de Síntesis e Implementación.

- 3-1-1.** En el menú *Flow Navigator*, seleccionar *Implementation*, presionar **Run Implementation**.

- 3-1-2.** Presionar **OK** en el cuadro de dialogo para que se ejecute el proceso de síntesis antes de la implementación. Presionar nuevamente **OK** para iniciar ambos procesos

Los procesos de síntesis e implementación se ejecutaran sobre el modulo `uart_top.vhd` y todos sus submodulos. Cuando el proceso de implementación finalice, aparecerá el cuadro de dialogo *Implementation Completed* con 3 opciones.

- 3-1-3.** Seleccionar la opción *Open Implemented Design* y presionar **OK**.

- 3-1-4.** Verificar en el reporte de temporización “Report Timing Summary” que no hay inconvenientes de temporización

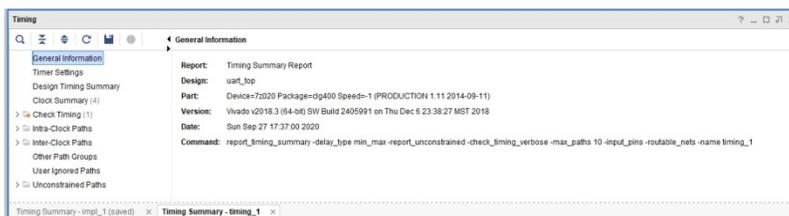


Figura 10: Reporte de temporización del sistema implementado

- 3-1-5.** Generar el archivo de configuración, para ello en el panel *Flow Navigator*, seleccionar **Program and Debug**, presionar **Generate Bitstream**. En el cuadro de dialogo presionar el botón “Cancel”

Verificar la Funcionalidad en el Hardware

Paso 4

4-1. Conectar la placa, conectar el adaptador usb-serie. Abrir una sesión de hardware y configurar la FPGA.

4-1-1. Conectar el cable micro-usb al conector prog/uart de la placa Arty Z7. Este conector se encuentra al costado del conector ethernet. Conectar el otro extremo a un puerto usb de la computadora

4-1-2. Conectar el adaptador usb-serie a la computadora. Verificar el puerto COM al que se mapea

4-1-3. Conectar los 3 cables de Tx; Rx y GND entre el conector USB-serie y el puerto JA de la placa ArtyZ7: TxD en el pin 1; RxD en el pin 2 y GND en el pin 5

Primero conectar la placa ArtyZ7 a la computadora, luego conectar el adaptador usb-serie

4-1-4. En el panel *Flow Navigator*, seleccionar **Program and Debug**, presionar *Open Hardware Manager*.

Aparecerá la interface del administrador de hardware (Hardware Manager) indicando el estado desconectado (unconnected).

4-1-5. Presionar en el enlace **Open target**, y luego en la opción **Auto Connect** del menú.

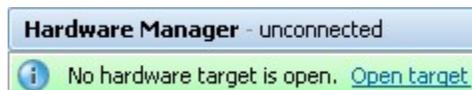


Figura 15. Hardware Manager

4-1-6. El estado de la sesión de hardware pasa de desconectado a tener el nombre del servidor (en este caso local host porque la placa está conectada en forma local). También se indica que el estado del dispositivo es *not programmed*.

4-1-7. Seleccionar el dispositivo (xc7z010_1) en la ventana de hardware. Presionar el botón derecho y seleccionar **Program Device...**

4-1-8. En el cuadro de dialogo, presionar el botón **Program** .

Se transferirá el archivo de configuración y el led verde DONE se prendera cuando la FPGA este configurada.

4-2. Iniciar un programa de emulación de terminal (por ejemplo TeraTerm), configurarlo con el puerto serie del adaptador usb-serie y con los parámetros de comunicación 115200 N-8-1.

4-2-1. Iniciar el programa de comunicaciones (por ejemplo TeraTerm).

4-2-2. Seleccionar el puerto serie (COM) del adaptador usb-serie.

4-2-3. Configurar el puerto serie a 115200, sin paridad, 8 bits de datos, 1 bit de parada .

-
- 4-2-4. En el programa de comunicaciones, escribir algunos caracteres. Los mismos aparecerán en la pantalla reenviados por la placa y en los 4 Leds se verá el nibble bajo del ASCII equivalente del caracter
 - 4-2-5. Presionar y mantener presionado el botón BTN0 y ver como ahora los Leds muestran el nibble alto del ASCII equivalente del caracter escrito en el programa de comunicaciones
 - 4-2-6. Seleccionar **File -> Close Hardware Manager**. Presionar **OK**.
 - 4-2-7. Cerrar el programa de comunicaciones. Desconectar el adaptador usb-serie de la computadora. Desconectar la placa ArtyZ7 de la computadora. **Hacerlo en ese orden, sino se puede dañar la placa ArtyZ7**
 - 4-2-8. Cerrar la aplicación Vivado