



Microtecnología y
Sistemas Embebidos

Instituto Politécnico Nacional

Centro de Investigación en Computación

Lenguajes de descripción de hardware

Práctica 7 - Programación de un procesador NIOS II

PROFESOR:

M. EN C. OSVALDO ESPINOSA SOSA

POR:

ING. RICARDO ALDAIR TIRADO TORRES

CIUDAD DE MÉXICO, 7 DE JUNIO DE 2024

Tabla de contenido

1. Objetivos	2
2. Programación del NIOS II	3
3. Conclusiones	6
4. Anexos	8
4.1. Descripciones del hardware	8

1. Objetivos

- Implementar un programa que mande valores, cada segundo, al puerto paralelo de 8 bits del procesador NIOS II, previamente creado.
- Aprender como se utilizan las funciones para escribir en un puerto y para generar retardos.

2. Programación del NIOS II

Actividad 1

Basado en el sistema creado en la práctica 6, crear un programa que mande valores al puerto paralelo (PIO) de 8 bits, cada 1 segundo. Investigar la función para escribir en el puerto y la función para crear un retardo.

En el entorno de programación del procesador Nios II, para escribir en un puerto de salida, generalmente se utilizan macros proporcionadas por el Sistema Operativo en Tiempo Real (RTOS) o por las librerías de hardware. La función más comúnmente usada es IOWR [1], vista en la Programa 1.

```
1 #include "io.h"
2
3 IOWR(BASE_ADDRESS, OFFSET, DATA);
```

Programa 1: Utilización de la función “IOWR”.

Donde:

- **BASE_ADDRESS:** Es la dirección base del puerto de salida.
- **OFFSET:** Es el desplazamiento del registro específico dentro del puerto.
- **DATA:** Es el valor que se va a escribir en el puerto.

Para generar un retardo, se puede usar la función `usleep` (ver Programa 2) que hace una pausa en la ejecución del programa durante un tiempo especificado en microsegundos. [2]

```
1 #include <unistd.h>
2
3 usleep(TIME);
```

Programa 2: Utilización de la función “usleep”.

Donde:

- **TIME:** Es el número de microsegundos para el retardo.

Primeramente, como se visualiza en la Figura 1, se asignaron los pines de entradas y salidas del NIOS II en la tarjeta en donde se iba a implementar el procesador. Siguiendo los pasos indicados en la presentación, se utiliza la herramienta *Eclipse* para generar un programa que imprima un mensaje en la consola, para ello se emplea una de las plantillas predeterminadas que ya trae el software. Una vez configurado el programa y construido el proyecto, se realiza la programación en la tarjeta, usando el puerto JTAG (ver Figura 2) y al momento de ejecutar el proyecto en el hardware del NIOS II, se visualiza en la Figura 3, el resultado en la consola.

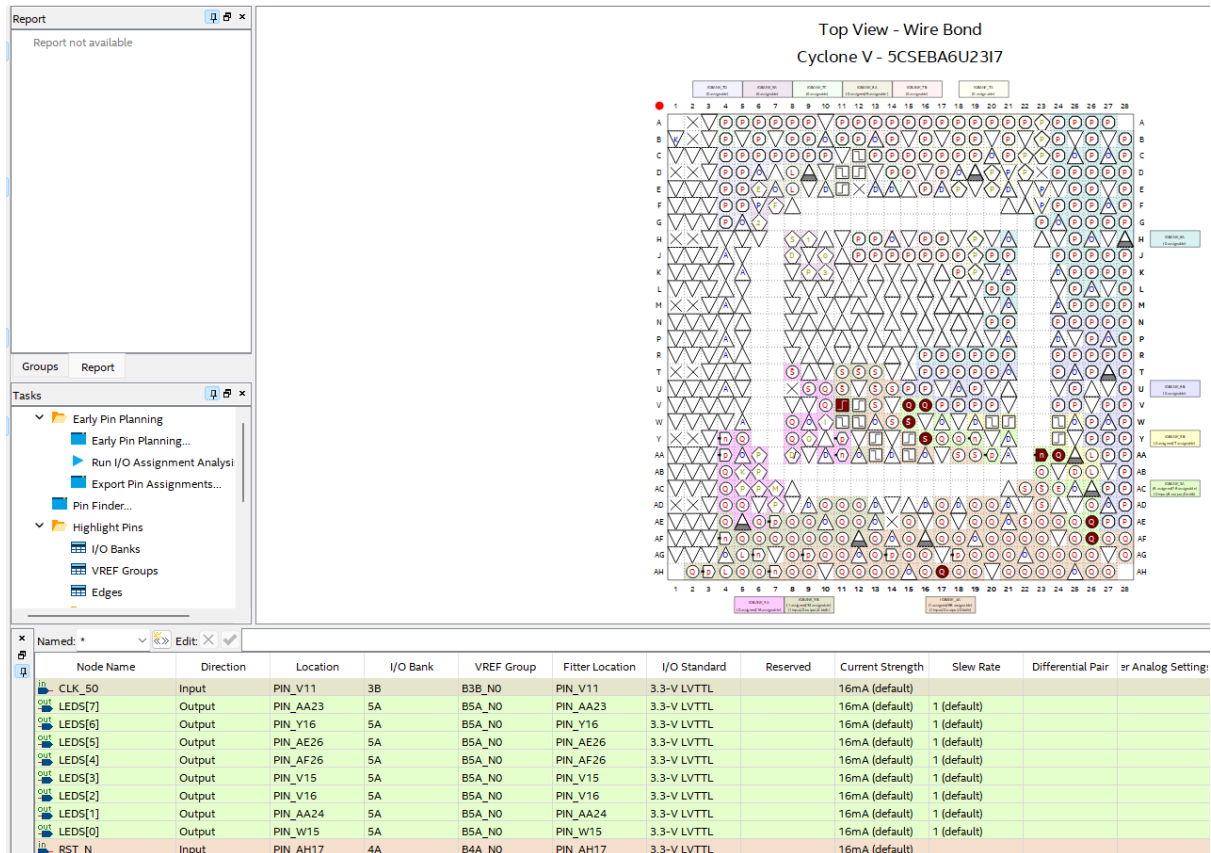


Figura 1: Asignación de los puertos de entradas y salidas procesador NIOS II en la tarjeta Cyclone V - 5CSEBA6U23I7.

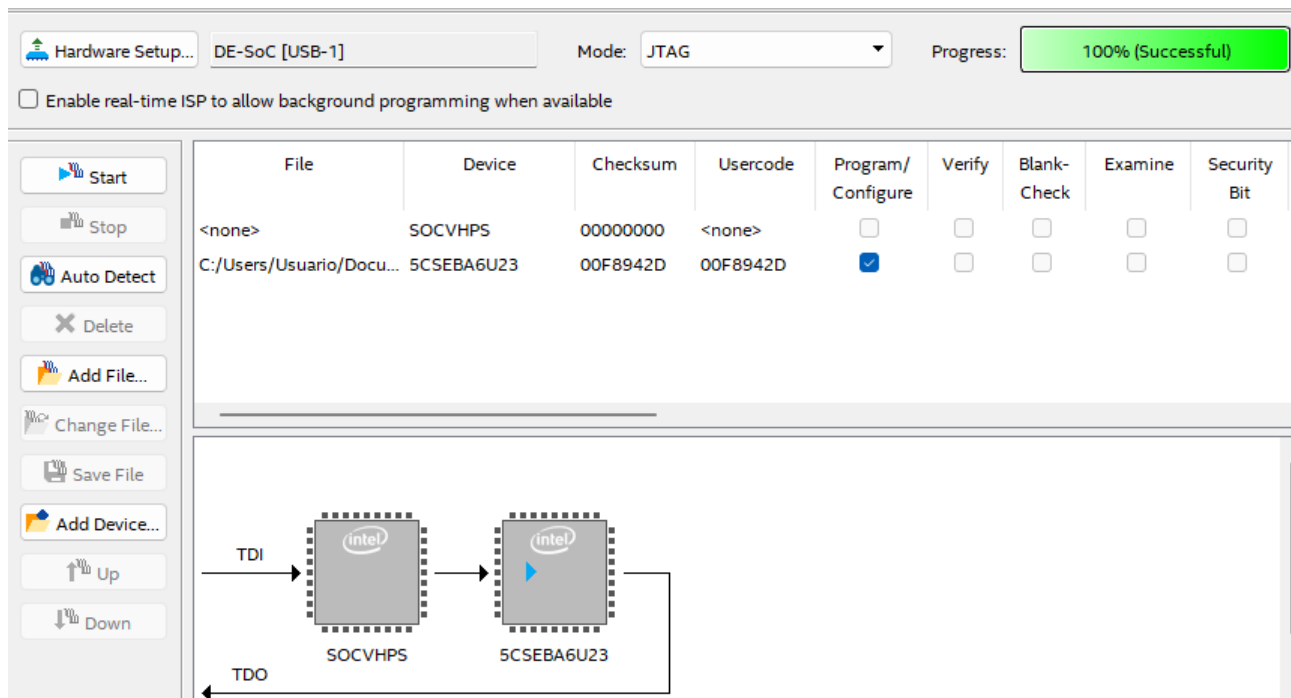


Figura 2: Programación del procesador NIOS II en la tarjeta Cyclone V - 5CSEBA6U23I7.

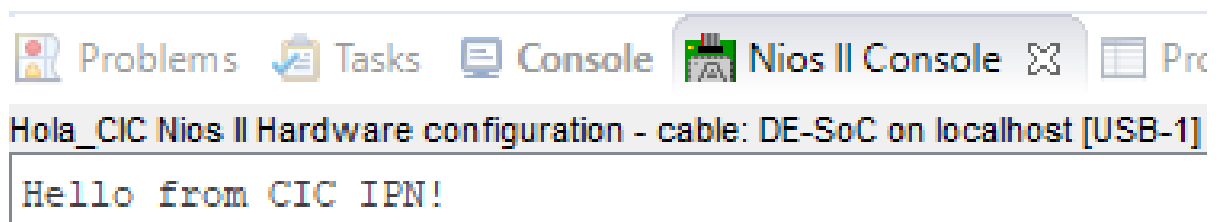


Figura 3: Resultado en la consola del Nios II.

3. Conclusiones

En conclusión, se implementó el programa de asignación de valores en la salida del procesador NIOS II de manera exitosa.

Utilizando el procedimiento de la presentación vista en clase para programar un NIOS II, se creó un programa que imprime en la pantalla un mensaje y además asigna un valor de 8 bits a la salida, con un cierto retardo. Para ello se empleó la herramienta de *Eclipse* y se usó el lenguaje C para indicarle al procesador la función que debía realizar.

Se asignaron los pines correspondientes en la placa de desarrollo y se programo en un dispositivo Cyclone V, observándose en los leds asignados, el funcionamiento del programa generado.

En los Anexos se pueden encontrar los códigos implementados.

Referencias

- [1] “Nios ii c language about sdram iowr & iord function and fopen & fclose,” <https://community.intel.com/t5/Programmable-Devices/NIOS-II-C-language-about-SDRAM-IOWR-IORD-function-and-fopen/m-p/17296>, [Accessed 06-06-2024].
- [2] “Busy-wait sleep function has incorrect timing on nios cpu,” <https://electronics.stackexchange.com/questions/352295/busy-wait-sleep-function-has-incorrect-timing-on-nios-cpu>, [Accessed 06-06-2024].

4. Anexos

4.1. Descripciones del hardware

```
1 module NiosII(  
2     input      CLK_50, RST_N,  
3     output [7:0] LEDS  
4 );  
5  
6 MinionsII xXxNiosxXx(  
7     .clk_clk(CLK_50),  
8     .leds_out_export(LEDS),  
9     .reset_reset_n(RST_N)  
10 );  
11  
12 endmodule
```

Programa 3: Instanciación del procesador NIOS II en el proyecto generado.

```
1 #include "sys/alt_stdio.h"  
2 #include "io.h"  
3 #include "unistd.h"  
4  
5 int main()  
6 {  
7     alt_putstr("Hello from CIC IPN!\n");  
8     alt_u8 led_initial = 0xF0;  
9  
10    IOWR(PIO_0_BASE, 0, led_initial);  
11    while (1){  
12        led_initial-=1;  
13        IOWR(PIO_0_BASE, 0, led_initial);  
14        usleep(1000000);  
15    }  
16    return 0;  
17 }
```

Programa 4: Programa en C, que imprime un mensaje en la consola y asigna valores a la salida del procesador NIOS II.