

## Tarea 2 | Entorno de desarrollo de hardware

### Objetivos

- Familiarizarse con el uso de la herramienta Quartus para agregar puertos de entrada y salida a un sistema dado.

### Descripción General

El objetivo de la tarea consiste en agregar dos puertos al sistema inicial suministrado con el *Tutorial de hardware*.

Antes de comenzar **el estudiante deberá:**

- Tener instalado *Quartus*.
- Haber realizado previamente el paso 2.2 *Trabajando con un sistema real cargado en la placa DE0* de la *Guía para las herramientas del Laboratorio*.
- Haber seguido todos los pasos de la guía del laboratorio *Tutorial de hardware*.

Luego deberá repetir el procedimiento del *Tutorial de hardware* partiendo nuevamente del sistema inicial suministrado con el tutorial. En esta oportunidad, en lugar de los puertos pedidos en el tutorial, se deben agregar los siguientes dos puertos:

- Un puerto de salida que maneje las salidas  $LEDG[7..0]$  de la placa, mapeado en la dirección **dirout** del espacio de salida.
- Un puerto de entrada mapeado en la dirección **dirin** del espacio de entrada, conectado al XNOR (XOR negado) bit a bit entre las salidas del registro del puerto de salida pedido arriba y la constante **MASK**. Observar que al leer este puerto de entrada con la instrucción “IN A, (dirin)” debe obtenerse el XNOR bit a bit del último valor escrito en el puerto de salida con la constante mencionada y NO el valor de los switches que no se utilizan en este caso.
- Ambos puertos deben estar decodificados sin fantasmas, es decir, podrán accederse únicamente en las direcciones **dirin** y **dirout** especificadas.

La constante **MASK** y las direcciones **dirout** y **dirin** de los puertos se determinan a partir de la cédula de identidad de cada estudiante como sigue:

Si los dígitos de la cédula de identidad son  $d_6, d_5d_4d_3, d_2d_1d_0$  (NO se considera el dígito verificador), entonces, **MASK** es el número  $0xd_1d_0$ , **dirout** es el número obtenido haciendo el OR entre la constante  $0x80$  y el número  $0xd_3d_2$  (la notación  $0x$  indica números hexadecimales (base 16)) y **dirin** haciendo el OR entre  $0x80$  y el número  $0xd_5d_4$ .

Por ejemplo: para la cédula de identidad 1.234.567 los valores de las constantes son:

- MASK** =  $0xd_1d_0 \Rightarrow$  **MASK** =  $0x67$  (0110 0111B)
- dirout** =  $0xd_3d_2$  OR  $0x80 \Rightarrow$  **dirout** =  $0x45$  OR  $0x80 \Rightarrow$  **dirout** =  $0xC5$  (1100 0101)
- dirin** =  $0xd_5d_4$  OR  $0x80 \Rightarrow$  **dirin** =  $0x23$  OR  $0x80 \Rightarrow$  **dirin** =  $0xA3$  (1010 0011B)

Observar que por la forma en que se obtienen, los valores de **dirout** y **dirin** siempre son mayores o iguales a 0x80 y por lo tanto no hay posibilidad de conflicto con los otros puertos presentes en el sistema suministrado con el tutorial (DFF y Bridged\_jtag\_uart).

Para verificar si el circuito funciona correctamente se sugiere escribir un programa de prueba que escriba un byte cualquiera conocido (p. ej. 0x55 = 01010101B) en el puerto **dirout**, seguido de una lectura desde el puerto **dirin**, y comprobar que el byte leído coincida con el XNOR bit a bit de **MASK** con el byte escrito ( $0x67 \text{ XNOR } 0x55 = 0xCD = 1100\ 1101B$  para el ejemplo). Para la corrección se cargará en el FPGA el archivo demo-hw.sof entregado por el estudiante y se verificará con programas de prueba que los puertos estén mapeados en las direcciones correspondientes al número de cédula del estudiante (y solamente en esas direcciones).

### Entrega

La entrega consistirá de un archivo comprimido en **formato zip (cualquier otro tipo de archivo no será tenido en cuenta)** con los siguientes archivos del diseño realizado por el estudiante (con estos mismos nombres):

- sistema.bdf
- demo-hw.sof
- demo-hw.qsf

El archivo comprimido debe tener como nombre el número de C.I. (sin puntos ni guiones y **SIN** el dígito verificador) y debe llevar la extensión “.zip”. Por ejemplo si el nro. de cédula es 1.234.567-8 el archivo debe llamarse 1234567.zip

El archivo debe subirse a través de la tarea que será creada en la página del curso en la plataforma EVA de facultad. La entrega **vence el día jueves 12 de mayo a la medianoche (23:59 hs)**.