

## Trabajo entregable

### 1. Introducción

El objetivo de este trabajo es diseñar una aplicación empotrada con componentes hardware y software a medida, y control remoto. El sistema elegido es el control remoto de un servo que establece la posición de un mecanismo. El servo es un actuador que controla la posición de un motor de corriente continua. La posición del servo se determina mediante el ciclo de trabajo (DC o Duty Cycle) de una señal PWM.

El sistema a diseñar tendrá 3 modos diferentes de funcionamiento que dependen de la lectura de un sensor de color que detecta 3 colores diferentes:

- Modo 1: Generación de señal PWM controlada de forma remota, a través de un protocolo de comunicaciones Bluetooth. Este protocolo permite realizar el control desde cualquier terminal compatible, como un teléfono móvil.
- Modo 2: Generación de señal PWM proporcional a la medida de la luz ambiente obtenida a través de un sensor.
- Modo 3: Generación de señal PWM controlada a través del PC/portátil.

Se podrá modificar el modo (nueva lectura del sensor de color) activando los switches de la placa de desarrollo.

Por simplicidad, la posición del servo y el ciclo de trabajo de la señal PWM coincidirán exactamente. La posición puede variar entre 10 y 90, lo que se corresponde por tanto con un ciclo de trabajo entre el 10% y el 90%. Al iniciar el sistema, el IP generará una señal con un ciclo de trabajo del 50%.

Para la generación de la señal PWM se utilizará un módulo hardware a medida que se facilitará a los alumnos (los alumnos no lo tienen que diseñar).

En general, el diseño se ha simplificado con el objeto de que su implementación sirva a los propósitos del aprendizaje. Salvo en aquellos aspectos que están claramente especificados, el diseño es libre. El estudiante podrá utilizar la aproximación que considere más oportuna para implementar la funcionalidad especificada.

## 2. Especificaciones del sistema

### 2.1. Descripción del hardware

El sistema consta de una placa Zybo, un módulo PMOD BT2 (conectado al conector PMOD JD de la placa), un sensor de color PMOD COLOR (conectado al conector PMOD JB de la placa) y un sensor de luz ambiente PMOD ALS (conectado al conector PMOD JA de la placa).

El hardware del sistema implementado en el dispositivo Zynq 7000 incorporado en la placa Zybo constará de los siguientes componentes:

- Microprocesador: Zynq Processing System (PS)
- Interfaz PMOD BT2
- Interfaz PMOD COLOR
- Interfaz PMOD ALS
- Módulo hardware PWM
- Temporizadores (AXI Timer)
- 1 LED
- 1 Pin de salida (PWM\_OUT)
- AXI GPIO para controlar los 4 switches de la placa de desarrollo

A continuación, se resume brevemente las características de cada uno de los elementos:

- Microprocesador:  
Se utilizará un solo núcleo del microprocesador ARM con la configuración por defecto.
- Interfaz PMOD BT2:  
El módulo PMOD BT2 permitirá el control remoto del sistema. Este interfaz se genera automáticamente siguiendo los pasos descritos en las transparencias de clase. Desde el lado del microprocesador, la interfaz consta de los siguientes elementos:
  - AXI\_LITE\_UART: Es la interfaz principal a través de la cual se reciben y se envían los mensajes. Es equivalente a una interfaz UART 16550
  - AXI\_LITE\_GPIO: Permite el control de las señales CTS/RTS de la UART
- Interfaz PMOD COLOR:  
Este interfaz y el sensor PMOD COLOR se estudiarán durante las prácticas.
- Interfaz PMOD ALS:  
Este interfaz y el sensor PMOD ALS se estudiarán durante las prácticas.
- Módulo hardware PWM:  
Para la generación de la señal PWM se utilizará un módulo hardware (IP). Este módulo se facilitará a los alumnos, que lo integrarán en el sistema, conectándolo al microprocesador como un periférico.

El módulo PWM tiene una salida pwm\_out y una interfaz AXI. A través de la interfaz AXI se puede acceder a los registros dados en la siguiente tabla:

| Registro         | Offset | Tamaño  | Descripción  |
|------------------|--------|---------|--|
| Ciclo de trabajo | 0      | 8 bits  | Ciclo de trabajo en porcentaje. El valor por defecto es 50 |
| Habilitación     | 4      | 1 bit   | Habilita la generación de la señal PWM                     |
| Preescalado      | 8      | 32 bits | Preescalado. Valor por defecto 1,000,000 (100 ms)          |

La salida del módulo hardware, pwm\_out, será una salida externa del sistema. Debe conectarse a un pin externo de la placa a través de un pin de un conector PMOD. También se conectará al LED, que actuará como testigo de la señal generada.

- Temporizadores:

En modo Standalone (sin Sistema Operativo) se necesita un temporizador para la realización de medidas de tiempo periódicas. En tal caso, se utilizará un AXI Timer.

En el caso de utilizar el Sistema Operativo FreeRTOS, el temporizador es innecesario ya que el sistema operativo dispone de funciones de temporización. No obstante, se debe incluir para mayor generalidad y evitar tener que configurar el hardware según el caso.

- Pines de salida:

La salida del módulo PWM (pwm\_out) es el pin en el que se generará la señal PWM y además esa salida también se conectará a un LED.

- Switches:

Los switches de la placa de desarrollo se utilizarán para indicar al sistema que debe leer de nuevo el sensor de color y posibilitar un cambio de modo de funcionamiento. También se utilizará un switch para forzar el modo de funcionamiento desde consola (modo 3).

## 2.2. Descripción del software

El software tendrá que implementar las siguientes funciones:

- Configuración e inicialización de todos los componentes hardware para el propósito del sistema
- Control del Sistema. Elección del modo de funcionamiento al iniciar el sistema y resto de acciones de control dependientes del modo de funcionamiento.
- Generación de la señal PWM\_OUT, actuando sobre el módulo hardware PWM

La implementación se realizará utilizando el sistema operativo FreeRTOS. De forma orientativa, se sugiere la siguiente descomposición:

- Cree una tarea para leer el sensor de color y cambiar el modo de operación.
- Cree una tarea para recibir los mensajes desde la interfaz Bluetooth y otra para recibir los mensajes desde la interfaz serie del PC/portátil. Envíe los mensajes mediante una cola a otra tarea que los interprete, independientemente de dónde procedan.
- Cree una tarea para interpretar y ejecutar los mensajes.
- Cree una tarea para la medida periódica.
- Cuando se modifique el valor del ciclo de trabajo, envíelo a una cola y cree una tarea para reportar los datos presentes en dicha cola.

Con carácter general, todos los componentes se controlarán mediante los mecanismos proporcionados por el sistema operativo.

### 2.2.1. Control del Sistema

El control del sistema dependerá del modo de funcionamiento elegido. A continuación, se resumirá el comportamiento del sistema para cada uno de los modos.

#### 2.2.1.1 Modo 1

En el caso de encontrarnos en el modo 1, el control se hará de forma remota por Bluetooth. En un sistema real, el control remoto se puede hacer por diferentes medios de comunicación, mediante cable o inalámbricos. En este trabajo utilizaremos Bluetooth como mecanismo de comunicación. Una interfaz Bluetooth es básicamente un puerto serie inalámbrico. A través de Bluetooth se enviarán mensajes de control y se recibirán mensajes de indicación del estado del sistema.

Para realizar el control remoto del sistema, debe conectarse desde un dispositivo Bluetooth, como un móvil. Existen varias apps que permiten operar como un terminal básico a través de Bluetooth, como por ejemplo Serial Bluetooth.

Para controlar el sistema se enviarán mensajes o comandos por Bluetooth desde el terminal remoto. Los comandos válidos se indican en la siguiente tabla:

| Mensaje (Comando) | Observaciones   | Descripción  | Mensaje de confirmación     |
|-------------------|---|--|-----------------------------|
| e                 | 1 solo carácter   | Habilitación del IP. Mientras el IP esté habilitado se generará la señal de control del servo        | "Habilitado"                |
| d                 | 1 solo carácter   | Inhabilitación del IP. Mientras el IP esté inhabilitado no se generará la señal de control del servo | "Inhabilitado"              |
| <número>          | Dos caracteres entre 0 y 9. El valor debe estar entre 10 y 90 | Indica la posición deseada del servo   | "Posición actual = <valor>" |
| +                 | 1 solo carácter   | Incrementa la posición en 5. Si el resultado es mayor que 90, la posición se fijará en 90.           | "Posición actual = <valor>" |
| -                 | 1 solo carácter   | Decrementa la posición en 5. Si el resultado es menor que 10, la posición se fijará en 10.           | "Posición actual = <valor>" |

Cuando se reciba un comando válido, el sistema enviará el mensaje de confirmación indicado en la tabla. Si el mensaje no es válido, el mensaje de confirmación será "Mensaje incorrecto".

Por defecto, el sistema estará inicialmente inhabilitado.

#### 2.2.1.2. Modo 2

En el caso de encontrarnos en el modo de funcionamiento 2, se medirá cada 4 segundos la luz ambiente con el sensor PMOD ALS y se actualizará el DC de la señal PWM generada. El DC de la señal PWM generada será proporcional a la medida proporcionada por el sensor de luz ambiente. El sistema enviará a través de la conexión bluetooth la medida de luz realizada mediante el siguiente mensaje:

"Luz <valor>"

#### 2.2.1.3. Modo 3

El modo de funcionamiento 3 será similar al modo 1, pero los comandos se enviarán desde la consola del ordenador y se recibirán por el mismo medio.

El modo de funcionamiento 3 se podrá forzar en cualquier momento, sin necesidad de leer el sensor de color, mediante la activación de un switch de la placa.

#### 2.2.2. Generación de la señal PWM

Para los modos de funcionamiento 1 y 3, los comandos e y d habilitarán o deshabilitarán el módulo hardware PWM. Para todos los modos, la posición se controlará configurando el registro que determina el ciclo de trabajo del módulo hardware PWM.

#### 2.2.3. Medidas Periódicas

Para realizar medidas periódicas se puede utilizar un temporizador o una tarea FreeRTOS que se desbloquee en un tiempo preestablecido. Este temporizador debería generar interrupciones periódicas (con el intervalo de tiempo deseado) que activarían el proceso de medida.

### 3. Entrega y evaluación

Los alumnos tendrán que entregar a través de Aula Global, con la fecha límite que se indique, o en todo caso antes del examen final, el siguiente material:

- Memoria del trabajo realizado, en la que se explique el diseño, en sus aspectos hardware y software. Es necesario incluir el block design realizado con Vivado y el código software comentado.
- Proyecto Vivado completo en fichero comprimido.

Adicionalmente, se habilitará una sesión, al final del curso, para que los alumnos muestren el funcionamiento del sistema diseñado.

En la evaluación se tendrá en cuenta el grado de completitud en la consecución de los objetivos y especificaciones del sistema, la calidad de la solución, la claridad del código software y la calidad de la memoria.