

Practica 1: Comunicación serie con un PC. Diseño del receptor

1. Objetivos

El objetivo de esta práctica es el diseño de un receptor serie RS-232 para recibir datos de un PC y mostrarlos en binario en los LEDs de la tarjeta DE1

2. Tiempo de laboratorio

1 sesión

3. Trabajo previo

Ha de realizarse un diagrama de bloques en el que se muestren cada uno de los bloques en los que se dividirá el sistema, incluyendo los nombres de todas las señales de entrada y salida a dichos bloques.

4. Memoria

Se entregará el diagrama de bloques del diseño así como los esquemas y el código VHDL del diseño.

5. Introducción

Los datos que se reciben desde el PC constan de 1 bit de arranque, 8 bits de datos, un bit de paridad (paridad impar) y un bit de *stop*. La velocidad de transmisión será de 19200 baudios. Como en la transmisión serie RS-232 no se transmite el reloj, es necesario sincronizar el reloj del receptor en cada byte, para lo que se utiliza el bit de arranque. En la figura 1 se muestra un diagrama de tiempos de la transmisión junto con la señal de lectura. Esta señal ha de generarse de forma que la lectura de cada bit se realice justo en la mitad del periodo de dicho bit, tal como se muestra en la figura. Con esta estrategia se minimiza la probabilidad de que el desfase existente entre los relojes del emisor y del receptor resulte en una lectura incorrecta del bit.

La señal de lectura será la encargada de gobernar un registro de desplazamiento para realizar la conversión serie-paralelo. Nótese que todo el diseño ha de ser **síncrono**.

Por último es necesario realizar un circuito para detectar errores de paridad y activar un led si se realiza una recepción incorrecta.

6. Diseño del circuito

Como en todo diseño complejo, es obligatorio separar el diseño en ruta de datos y control. La ruta de datos consistirá en los siguientes bloques:

1. Un registro de desplazamiento para la recepción de los datos.

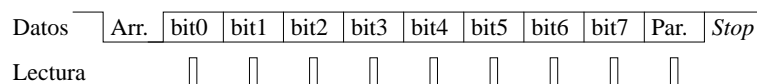


Figura 1: Formato de transmisión y señal de lectura

2. Un circuito para calcular la paridad para poder comprobar si se han producido errores en la transmisión.

Cada uno de estos bloques es conveniente diseñarlo y simularlo por separado para verificar su funcionamiento correcto antes de integrarlos con el circuito de control, el cual consistirá en una máquina de estados asistida por un temporizador para poder esperar la llegada de cada uno de los bits.

6.1. Diseño del registro de desplazamiento

En primer lugar es necesario realizar un registro de desplazamiento de 8 bits para almacenar el dato recibido. Ha de tenerse en cuenta que dicho registro ha de ser síncrono, es decir, su reloj ha de ser el mismo que el de los demás componentes de la FPGA (14,7456 MHz). Por tanto ha de diseñarse el registro de manera que sólo realice un desplazamiento mientras esté activa una señal de control (enable).

6.2. Diseño del detector de errores de paridad

En segundo lugar es preciso diseñar un circuito **secuencial** para verificar que la transmisión ha sido correcta. En el caso de paridad impar, la suma de todos los bits, incluido el de paridad, ha de ser impar. Nótese que no es necesario calcular la suma total, sino sólo el bit menos significativo de la suma y verificar que éste vale uno.

6.3. Diseño del circuito de control de recepción

El circuito de control ha de desplazar y calcular la paridad en los instantes mostrados por la señal de lectura de la figura 1. El circuito debe sincronizarse con la transición de 1 a 0 que se produce en el bit de arranque. A partir de dicha transición ha de esperarse un periodo equivalente a 1.5 bits antes de generar el primer desplazamiento/cálculo de paridad y a continuación generar los 8 restantes con un periodo equivalente a 1 bit. Una vez finalizado el proceso, el circuito ha de volver al estado inicial y esperar un nuevo bit de arranque.

6.4. Diseño del receptor

Una vez realizados los tres bloques anteriores y después de verificar mediante simulación que funcionan correctamente, se diseñará el receptor completo. La entrada al circuito será la línea serie de la que dispone la tarjeta DE1 (conector DB-9 etiquetado como RS232). La señal de recepción de dicha línea serie se denomina `UART_RXD` y está conectada a la patilla `PIN_F14` de la FPGA. La salida para los datos serán los 8 LEDs verdes de la tarjeta (el bit menos significativo se conectará al `LED_G0`). Si existe un error de paridad se iluminará el `LED_R0` para indicarlo.

7. Prueba del circuito

Una vez verificado mediante simulación que el circuito se comporta correctamente, se conectará el puerto serie del PC con el puerto serie montado en la tarjeta DE1 y mediante un programa de comunicaciones se enviarán una serie de caracteres a la FPGA, comprobando que el valor en binario del código ASCII¹ de los caracteres enviados aparece en la barra de LED. Es muy importante que tanto la velocidad de transmisión (19200 baudios) como el formato de ésta (8 bits de datos, paridad impar y 1 bit de stop) se configuren correctamente en el programa de comunicaciones.

Como programa de comunicaciones se usará el HyperTerminal. Para configurarlo existe una pantalla de configuración en `archivo->propiedades`. Una vez configurado es necesario pulsar sobre el icono del teléfono (barra de herramientas) para comenzar a transmitir. Si es necesario cambiar la configuración hace falta antes cortar la comunicación pulsando sobre el icono de colgar el teléfono.

¹ Si no se sabe de memoria la tabla ASCII, recuerde que los dígitos de 0 a 9 se codifican mediante 0x30 a 0x39 respectivamente