

Práctica 9.

DISEÑO DE UN RECEPTOR PARA COMUNICACIÓN SERIAL

OBJETIVO:

Demostrar a los estudiantes mediante el diseño de un módulo receptor (RX), usado en comunicaciones de tipo serial UART (*Universal Asynchronous Receiver Transmitter*), la utilidad de este módulo, así como la importancia de su presencia en la arquitectura de un procesador para aplicaciones electrónicas de recepción de información. Mostrar su aplicación en el control de dispositivos periféricos desde una terminal remota.

ESPECIFICACIONES:

Utilizando un FPGA y 8 LEDS, diseñar un sistema receptor serial, el cual sea capaz de recibir un carácter ASCII del teclado de una computadora, procesarlo en el FPGA y posteriormente mostrar su código binario en los 8 LEDS. La conexión entre el FPGA y la computadora deberá realizarse empleando un circuito convertidor USB TTL-Serial. La figura 9.1 muestra el diagrama de bloques del sistema.

DIAGRAMA DE BLOQUES:

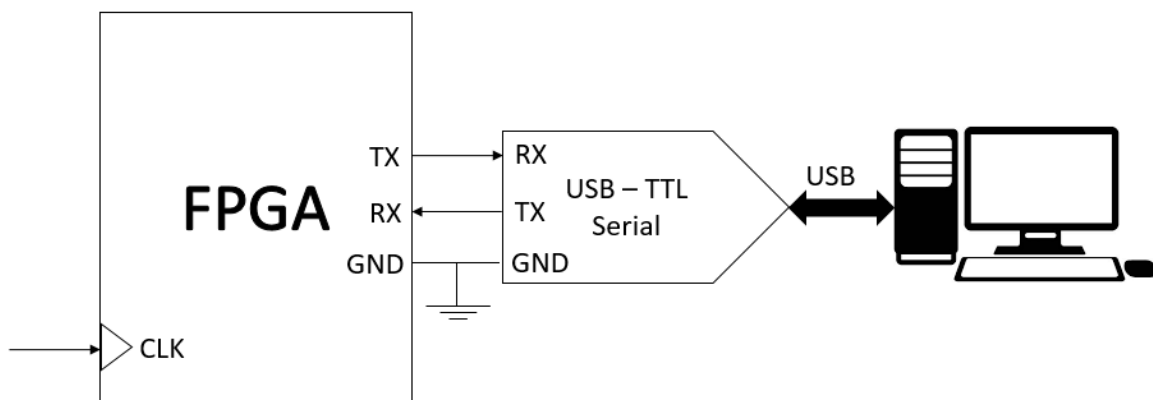


Figura 9.1. Diagrama de bloques para la comunicación serial.

Al igual que el sistema transmisor, su contraparte receptora resulta útil de ser implementada en un dispositivo FPGA, dadas las características de reconfiguración de éste. Aunado a ello

y a la capacidad de emplear recursos de hardware mínimos, un módulo RX en la comunicación UART permite a su vez una amplia gama de aplicaciones electrónicas y de cómputo.

Es importante resaltar que, en este punto, será posible observar que la implementación de este sistema será más simple que en el caso del transmisor, en donde gran parte de la lógica que establece la velocidad de transmisión es idéntica.

La figura 9.2 muestra los bloques funcionales del sistema Receptor, donde las señales se muestran como flechas de color azul, mientras que las terminales físicas se muestran en color rojo.

BLOQUES FUNCIONALES:

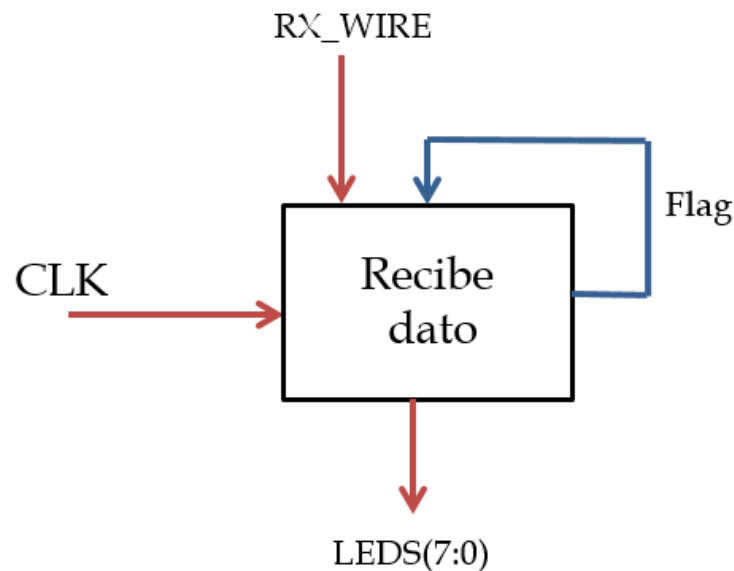


Figura 9.2. Bloque funcional del sistema Receptor Serial

La figura 9.3 muestra la parte entidad del sistema receptor de comunicación serial. Las terminales físicas corresponden al reloj maestro del FPGA de 50 MHz, 8 LEDS y la línea de recepción (RX_WIRE).

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity RX is
    port( Clk      : IN  STD_LOGIC;
          LEDS     : OUT STD_LOGIC_VECTOR(7 downto 0);
          RX_WIRE  : IN  STD_LOGIC);
end entity;

```

Figura 9.3. Entidad del sistema Receptor Serial RX.vhd

La figura 9.4 muestra la parte declaratoria de la arquitectura del módulo **RX.vhd**, en donde se declaran todas las señales involucradas en el proceso de recepción del dato proveniente de la computadora.

```

architecture behavioral OF RX IS
    signal BUFF: STD_LOGIC_VECTOR(9 downto 0);
    signal Flag: STD_LOGIC := '0';
    signal PRE: INTEGER RANGE 0 TO 5208 := 0;
    signal INDICE: INTEGER RANGE 0 TO 9 := 0;
    signal PRE_val: INTEGER range 0 to 41600;
    signal baud: STD_LOGIC_VECTOR(2 downto 0);

```

Figura 9.4. Parte declaratoria en la arquitectura del sistema Transmisor Serial

La figura 9.5 presenta el código del proceso “RX_dato”, correspondiente a la descripción del bloque funcional “Recibe dato”. Dicho proceso es el encargado de generar la velocidad de transmisión “*Baudrate*” y recibir los bits asociados al dato proveniente de la terminal física RX_WIRE, para posteriormente ser transferido a los 8 LEDS.

```

begin
  RX_dato : process(Clk)
  begin
    if (Clk'EVENT and Clk = '1') then
      if (Flag = '0' and RX_WIRE = '0') then
        Flag<= '1';
        INDICE <= 0;
        PRE <= 0;
      end if;
      if (Flag = '1') then
        BUFF(INDICE)<=RX_WIRE;
        if(PRE < PRE_val) then
          PRE <= PRE + 1;
        else
          PRE <= 0;
        end if;
        if(PRE = PRE_val/2) then
          if(INDICE < 9) then
            INDICE <= INDICE + 1;
          else
            if(BUFF(0) = '0' and BUFF(9)= '1') then
              LEDS <= BUFF(8 DOWNT0 1);
            else
              LEDS <= "00000000";
            end if;
            Flag <= '0';
          end if;
        end if;
      end if;
    end if;
  end process RX_dato;

```

Figura 9.5. Proceso RX_Dato del sistema Receptor Serial

Finalmente, la figura 9.6 muestra la última parte de la arquitectura del sistema Receptor Serial, en donde se selecciona la velocidad de recepción.

```

  baud<="011";
  with (baud) select
    PRE_val <= 41600 when "000", -- 1200 bauds
                20800 when "001", -- 2400 bauds
                10400 when "010", -- 4800 bauds
                5200 when "011", -- 9600 bauds
                2600 when "100", -- 19200 bauds
                1300 when "101", -- 38400 bauds
                866 when "110", -- 57600 bauds
                432 when others; --115200 bauds

end architecture behavioral;

```

Figura 9.6. Código para selección de velocidad del sistema Receptor Serial

ACTIVIDADES COMPLEMENTARIAS:

1.- A partir del código presentado, el alumno desarrollará un bloque funcional genérico para realizar la recepción serial de un byte, el cual cuente con un parámetro genérico para seleccionar el *baud rate* y tenga una bandera de salida que nos indique cuando se ha recibido un caracter y está listo para ser leído, cuyo diagrama de bloques se muestra en la figura 9.7, y que pasará a ser parte de la biblioteca de módulos funcionales del alumno.

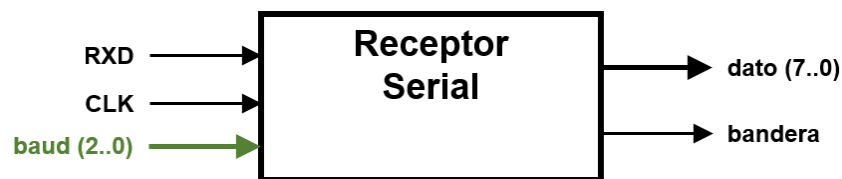


Figura 9.7. Diagrama de bloques del módulo de recepción serial

2.- Las demás actividades complementarias serán presentadas el día de la práctica.