

INSTITUTO TECNOLÓGICO DE COSTA RICA

ÁREA ACADÉMICA DE INGENIERÍA MECATRÓNICA

**MT 4001
Electrónica Digital**

Tarea de diseño 3

David Rodríguez Camacho – 2019024546
Andrew Eliécer Vargas Puffenberger – 2020425926
Emmanuel Naranjo Blanco – 2019053605

Profesor: Ing. Carlos Salazar García, MSc.

Semestre II - 2021

I. Desarrollo del diseño

Se presenta a continuación el procedimiento para llegar al circuito diseñado. Este consiste en una división en distintas etapas para posteriormente realizar un ensamble que cumpla con las especificaciones de diseño.

i. Requerimientos y restricciones del problema.

Se requiere un circuito secuencial que permita llevar el conteo del número de pulsos de entrada con duración mínima de 4 ciclos de reloj, cuyas entradas y salidas se muestran en la Figura 1.

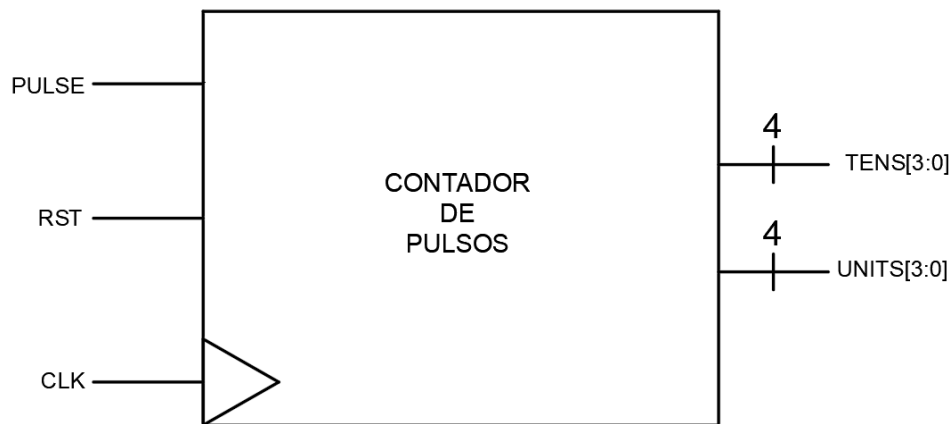


Figura 1. Diagrama de entradas y salidas del sistema a diseñar.

Para esto se tiene un reloj que opera a una frecuencia de 100 MHz y un reset sincronizado con el reloj. Como restricción, se tiene que el sistema puede contar hasta 20 y cuando reciba un pulso posterior, se restaure el conteo.

A modo de resultado se tiene una salida de dos señales de 4 bits codificadas en BCD que permitirá llevar el conteo desde 0 hasta 20.

ii. *Pulso de inicio Pulse.*

El conteo se da en base a cada pulsación que recibe el circuito en la entrada *Pulse*. Sin embargo, puede haber pulsos no deseados como ruido, quienes se consideran aquellos con una duración menor a los 4 ciclos de reloj, es decir 40 ns. Para esto se colocó un contador que cuenta ciclos de reloj una vez recibido el pulso a través de su *enable*. Este permite llevar control de la duración de *Pulse*.

iii. *Uso del Enable y Flip-Flops.*

Una vez recibido el pulso de entrada, el primer contador se activa y junto con el comparador, emiten un pulso cuando se cumple la condición de los 4 ciclos. Se utiliza el FF-D indicado en la Figura 3 con un recuadro rojo para recibir una señal que indica cuando el pulso fue válido. Este se controla a partir de la salida de un segundo FF-D conectada al *enable*. Este Flip-Flop de control normalmente permanece en alto y se encarga de controlar cuando llega un pulso y que tarde más de los 4 ciclos del CLK. Esto se debe a que el pulso de entrada puede durar más del tiempo mínimo y aun así equivale a un único pulso. De este modo, el FF-D marcado en la Figura 4 con un recuadro azul envía un cero cuando este recibe el un pulso a partir de la configuración conectada a su RST, y hace que el FF-D del recuadro rojo permanezca en memoria hasta que finalice la pulsación.

Así, este último emitirá la señal que permita el conteo del contador que cuenta hasta 20.

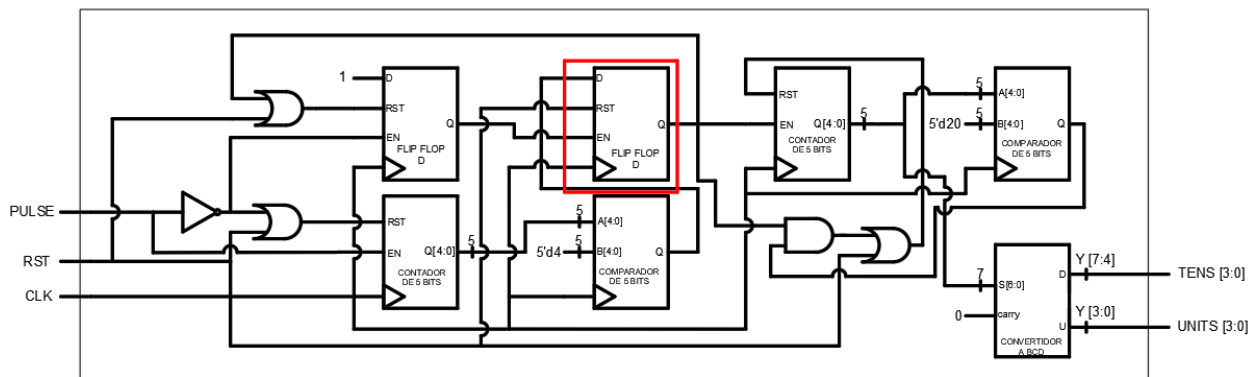


Figura 3. Diagrama donde se muestra el Flip Flop D2.

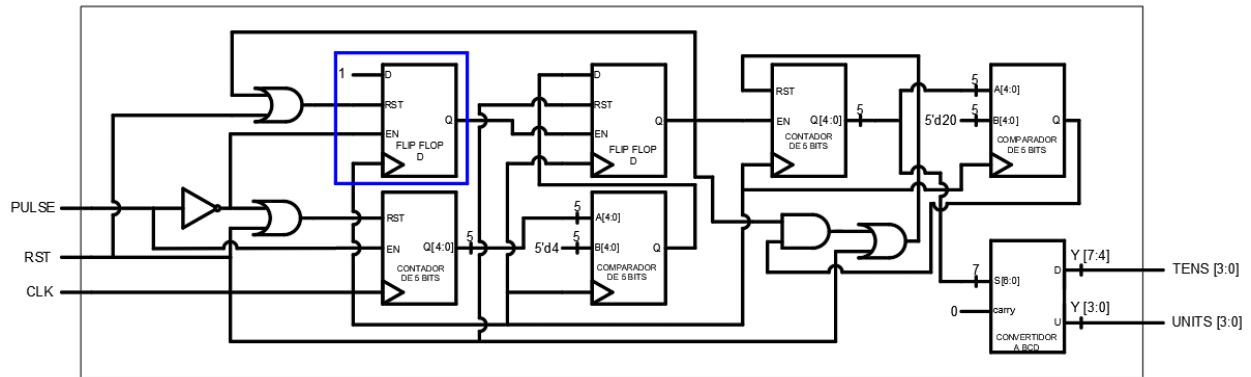


Figura 4. Diagrama donde se muestra el Flip Flop D1.

iv. Bloques Contadores y Comparadores.

Primeramente, se tiene un bloque contador de 4 pulsos de reloj y un comparador que detecta cuándo se llega a los 4 pulsos mínimos establecidos por la restricción del enunciado, que enviará la señal a un FF-D como se mencionó anteriormente.

Por otra parte, se utiliza un comparador y un contador de 5 bits para contar hasta los 20 pulsos válidos de entrada. Cuando este contador alcance el número 20 y el circuito reciba el siguiente pulso válido, el comparador se activará y enviará una señal en alto que reiniciará el conteo.

Ambos comparadores que reciben la salida de su respectivo contador se muestran en la Figura 5.

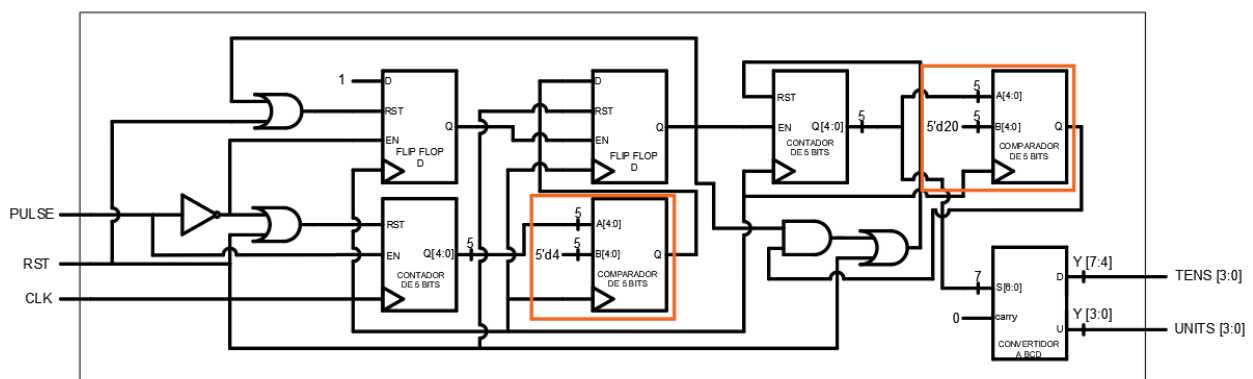


Figura 5. Diagrama donde se muestran los contadores.

v. Uso del RST.

La señal de *reset* es síncrona al sistema, es decir, cuando su valor esté en alto y haya un flanco positivo de reloj, el circuito se restaurará a cero y se esperará por la siguiente pulsación para iniciar nuevamente el conteo.

Además, una vez que el conteo llegue al pulso 21, se activará el RST para comenzar en el estado cero y esperar a que se active nuevamente la señal *Pulse*. Para esto se utilizan compuertas lógicas conectadas a las entradas RST de cada módulo y las respectivas salidas de los Flip-Flops y comparadores.

vi. Conversor de binario a BCD.

Se requiere de un conversor de binario natural a BCD para que la salida del circuito se muestre en unidades y decenas codificadas en BCD. Para esto se apoya del módulo implementado en la Tarea de Diseño 1 (Rodríguez, Vargas, Naranjo, 2021), quien recibe 8 bits y puede retornar tres salidas: unidades, decenas y centenas. En este caso solo se requiere de las unidades *Units* y decenas *Tens*, que se muestran en la Figura 9 como los bits de la salida Y [7:4] y Y [3:0].

En resumen, este módulo se compone de un arreglo de bloques denominados *bloque convertidor*, cuya funciones lógicas se muestran a continuación y representación final se muestra en la Figura 7.

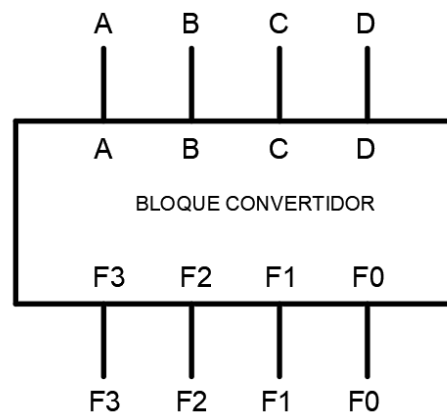


Figura 6. Bloque base para realizar la conversión de binario a BCD.

Tomando las ecuaciones de la Tarea 1:

$$F3 = A + BD + BC$$

$$F2 = AD + BC'D'$$

$$F1 = B'C + CD + AD'$$

$$F0 = AD' + A'B'D + BCD'$$

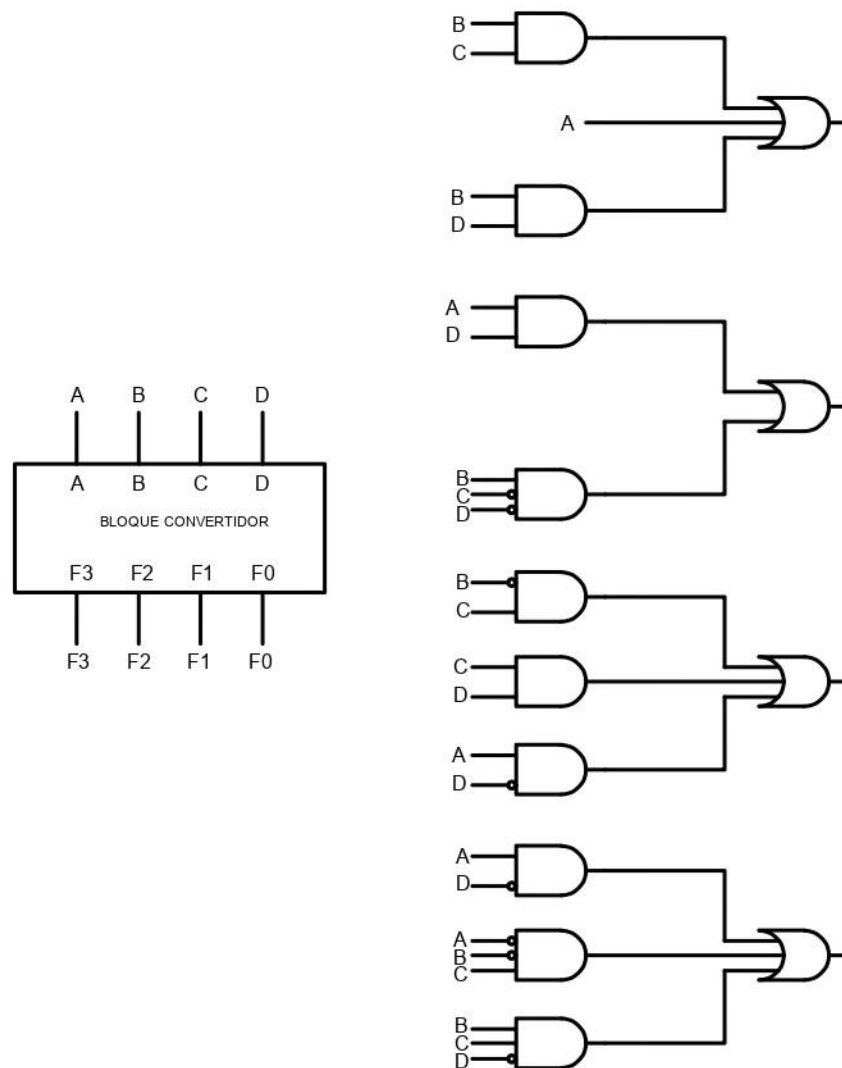


Figura 7. Diseño para el módulo de conversión de la figura 6.

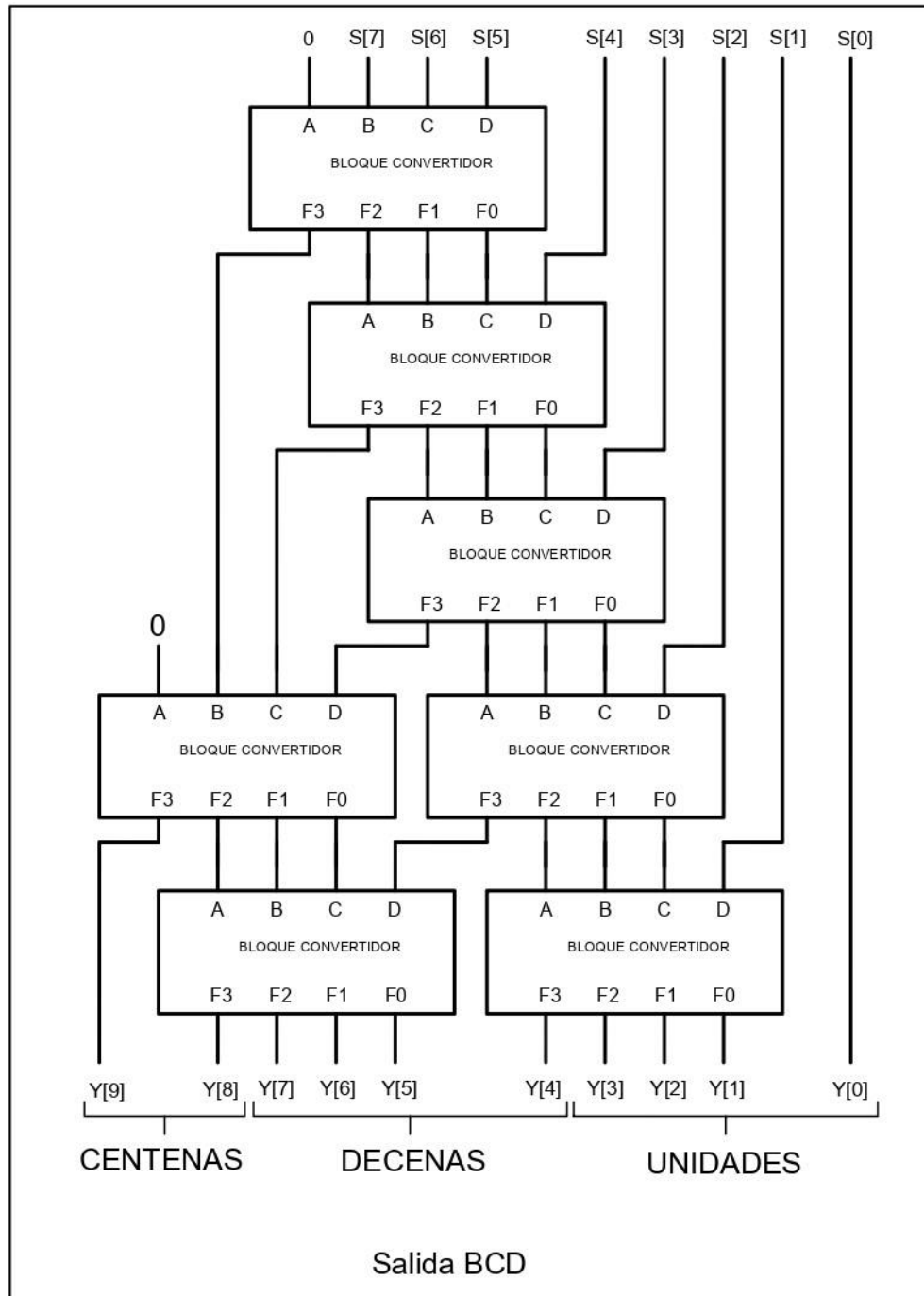


Figura 8. Diseño para el conversor de binario a BCD de 8 bits.

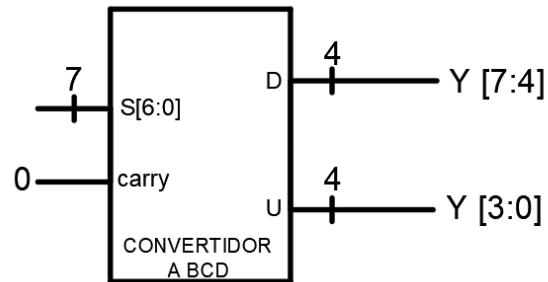


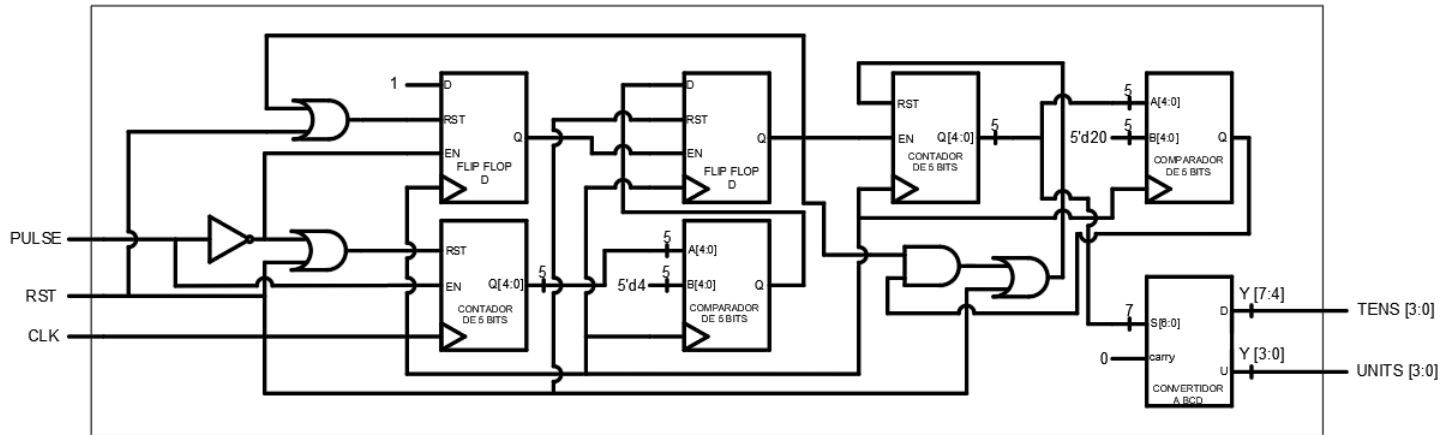
Figura 9. Diseño de bloque para el conversor de binario a BCD de 8 bits.

vii. Implementación.

Para finalizar, se implementó cada componente requerido en Verilog, se simularon los módulos propuestos y se empleó la simulación del diseño final en el *main*. Los archivos .v utilizados en la ejecución se adjuntan con el presente trabajo, en la carpeta *implementation*.

II. Diseño Final

Se presenta a continuación el circuito final diseñado.



Referencias

Rodríguez, D., Vargas, A., Naranjo, E. (2021, 5 de octubre). *Tarea de diseño 1*. [Documento PDF en TEC Digital].