

CONTADOR BINARIO

He aquí uno de los fundamentos teóricos de la lógica digital más básicos y a la vez importantes a tener en cuenta por quien quiera iniciarse en el mundo de la electrónica. En este ejercicio, se pretende ver una serie de circuitos cuya función es la de exponer una sucesión numérica ascendente basada en el código binario.

Primeramente pasamos a explicar qué es y en qué consiste un código binario. El circuito funciona de manera correcta si cada vez que se origina un pulso de reloj, a través del switch, los LEDS encienden de forma secuencial.

Este circuito es un contador binario. Al aplicar un pulso de reloj se observarán cambios en las salidas. Se da inicio a la cuenta que comienza en el número cero, 0000b, (todos los leds apagados) y termina en 15, 1111b (todos los leds encendidos). Del mismo modo hemos colocado al lado de las salidas luminosas (diodos led), un display 7 segmentos para comparar la salida numérica con la salida luminosa.

¿QUÉ ES EL CÓDIGO BINARIO?

Es aquel en que los elementos binarios "bits" se representan solamente con los valores "1" ó "0". Un bit es la unidad más pequeña de información en código binario. El valor de un bit puede ser cero o uno, según determine el conmutador. Un grupo de ocho bits en una fila constituye un byte, la unidad más pequeña que puede almacenar un carácter.

La sucesión de bits (serie de bits) codifica información en grupos, llamados bytes, en base a un sistema numérico binario. El sistema decimal con el que se trabaja normalmente tiene diez dígitos, de cero a nueve. Al igual que en el sistema decimal, cuando los dígitos en una columna llegan hasta el último (9 en decimal, 1 en binario), el dígito comienza en el cero y el de la columna a la izquierda aumenta un valor. Así, en el sistema decimal, el número 09 mas 1 hacen 10. Contando según el código binario y utilizando un byte (ocho columnas o bits) obtenemos: 00000000, 00000001, 00000010, 00000011, 00000100, 00000101 y así sucesivamente. Mas adelante expondremos una tabla representando este código de tal manera que se haga mas sencillo su entendimiento. Al relacionar números binarios específicos con caracteres, números, signos de puntuación y otros símbolos, una sucesión de ceros y unos pueden representar palabras y oraciones.

Todas las "aplicaciones" (software para el "usuario final") que conocemos han debido ser creadas especificando primero sus funciones y traduciendo éstas en "instrucciones" comprensibles por la máquina.

Para ésto, el o los programadores utilizan un determinado "lenguaje de programación", que es como un idioma: cuenta con un diccionario (los "comandos") y una gramática (reglas de sintaxis). El programador traduce las especificaciones de funciones y operaciones (que están en su idioma natural y/o en gráficos que especifican su secuencia) en este idioma, elaborando un producto llamado "programa fuente".

Se dice que el código binario es de "bajo nivel" o "primer nivel" (porque al usar pocos signos logra muy difícilmente expresar cosas complicadas), mientras un lenguaje humano es de "muy alto nivel" (con una cantidad mayor de signos y con reglas combinatorias logra expresar con facilidad cosas muy complicadas). Todo el esfuerzo, entonces, para facilitar la comunicación del hombre con el computador, ha de centrarse en el desarrollo de lenguajes de mayor nivel.

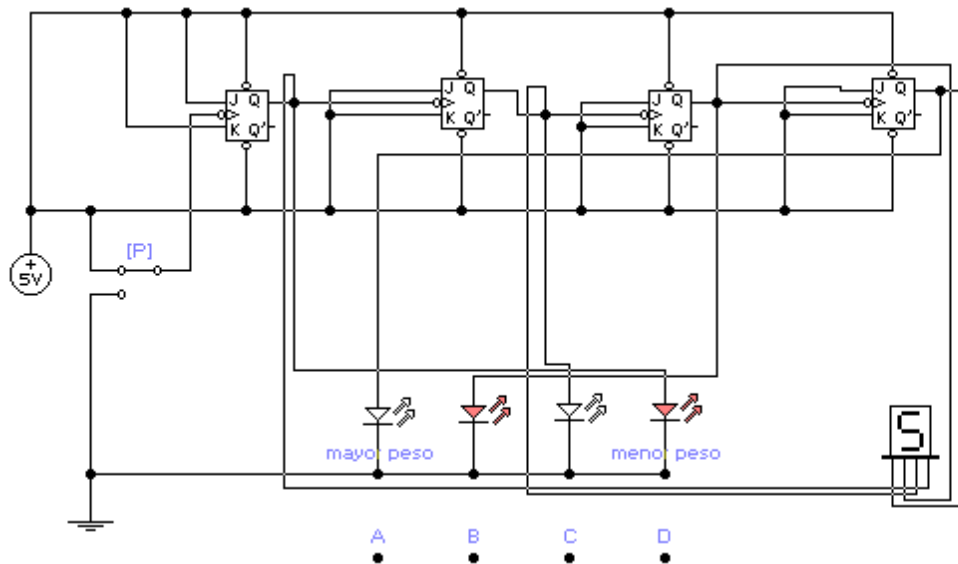
TABLA REPRESENTATIVA DEL CÓDIGO BINARIO

Vamos a pasar a representar una tabla de visualización del código binario. En esta tabla solo se ven representados unos y ceros, seguidamente pondremos para su entendimiento, el equivalente decimal.

A Mayor peso	B	C	D menor peso	N° DECIMAL
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10 (A)
1	0	1	1	11 (B)
1	1	0	0	12 (C)
1	1	0	1	13 (D)
1	1	1	0	14 (E)
1	1	1	1	15 (F)

Los pesos de la tabla indican el bit más importante ó significativo de una secuencia.

Seguidamente pasamos a representar el diagrama de conexiones eléctrico del circuito que será el contador que utilizemos para visualizar una escala ascendente del código binario.

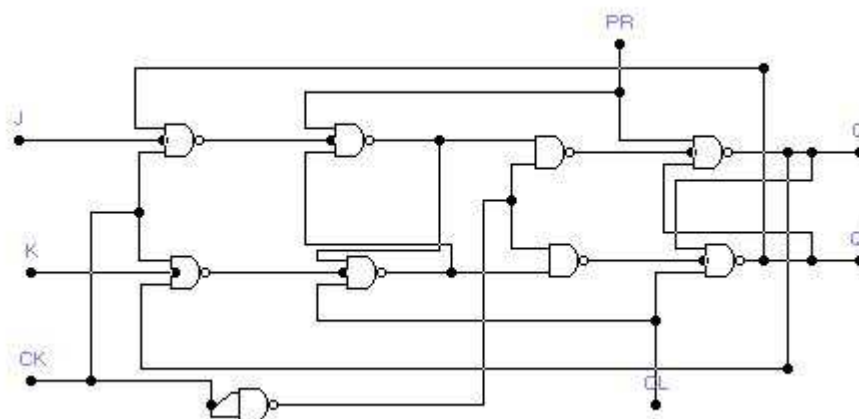


Como se puede observar, el diseño está basado en biestables JK master slave.

Un biestable actúa fundamentalmente como contador. En este caso hemos optado por los biestables JK master Slave. Su cualidad principal es que pasan la información desde su entrada a la salida, una vez que la señal de reloj se encuentra en su estado bajo, por lo que no existe el problema de que se puedan superponer dos informaciones distintas, induciendo esto al error en la comunicación.

Esta siguiente es la configuración de un biestable JK master Slave:

Biestable J-K master - slave



En los dos últimos biestables se puede observar también como las entradas dependen de la señal CK de reloj, mientras que el preset y el clear son totalmente asíncronos.

La construcción de dicho biestable está basada en las puertas lógicas NAND, que son operaciones de producto de dos bits negada, es decir, si nos encontramos a la entrada de una puerta NAND un 1 y un 0, en lógica digital $1 \times 0 = 0$, sin embargo al negar el resultado del producto, el resultado final será un 1.

Volviendo al diseño de nuestro contador, se observan dos tipos de salidas, una luminosa y otra representada por un display 7 segmentos. Se pueden ver dos diodos led iluminados, lo cual significa que esas salidas están a nivel alto (1). Cada secuencia de diodos led iluminados, representa un número binario; la representación que hemos realizado en páginas anteriores de la tabla del código binario nos ayuda junto con el diseño del contador a entender en qué consiste este código.

En vez de situar a la entrada de reloj de los biestables un reloj, lo cual nos daría una secuencia en cada pulso, hemos situado un pulsador, con lo que la secuencia de ascenso la controlamos nosotros.

Se puede comprobar que el diseño es muy sencillo, y muy entretenido a la hora de ver una secuencia de luces y de entender en qué consiste el código binario y a lo que equivale en numeración decimal.