

TECNOLOGÍA DE COMPUTADORES. CURSO 2021/22.

PRÁCTICA Nº 3: SISTEMAS COMBINACIONALES.

SUBCIRCUITOS

Semana del 8 al 11 de noviembre.

SOLUCIÓN

0.- Introducción.

El objetivo de esta práctica es diseñar un circuito digital compuesto por varios subcircuitos (ver Figura 1). La entrada del circuito es un vector de 8 bits V, que es proporcionado externamente en función de un dato personal del alumno, concretamente, si la cifra correspondiente de su DNI es par pone un 0 y si es impar pone un 1. Por ejemplo, si el DNI es 08.453.392, el vector sería **00011110**.

- El Subcircuito 1 debe visualizar el índice del bit prioritario (el bit de más peso a 1) del vector.
- El Subcircuito 2 implementa la función F, la cual vale 1 si el índice del bit prioritario es un número par, con la excepción del 0, en cuyo caso F=0. En los restantes casos también F=0.
- El Subcircuito 3 implementa la función G, la cual vale 1 si el número de 1's contenidos en el código binario del índice del bit prioritario es impar y 0 si es par.

La Figura 1 muestra el circuito integrado (*main*) y un ejemplo en el que el vector de entrada es V=00011110. En este vector, el índice del bit prioritario es 4. Por lo tanto, el circuito 1 proporciona la salida pertinente para que sea visualizado el número 4 en un display. Ya que 4 es un número par, el subcircuito 2 proporciona F=1. Por último, el código binario de 4 es 100, secuencia que contiene un impar de 1's. Luego el subcircuito 3 proporcionará G=1.

El alumno debe implementar los subcircuitos 1, 2, 3 y, finalmente, implementar el circuito *main*.

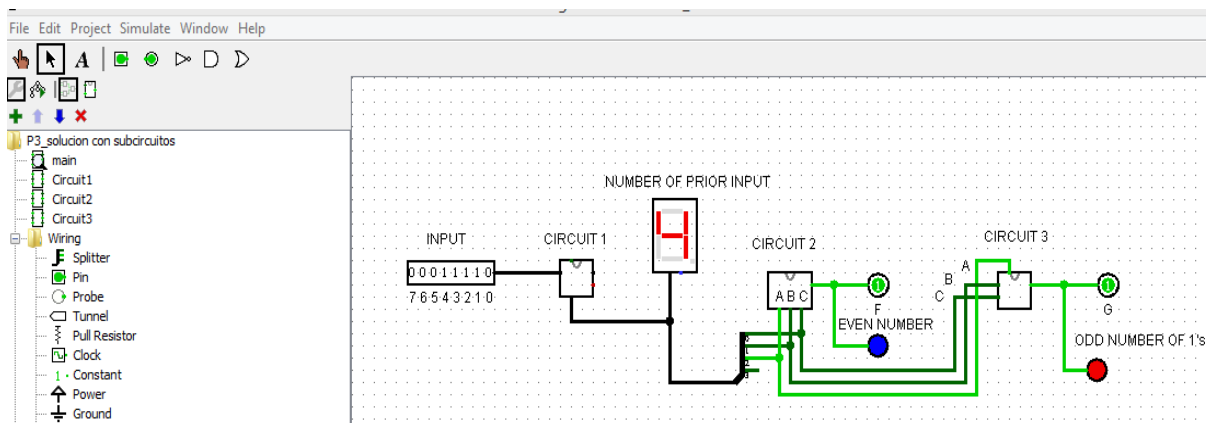


Figura 1

Parte 1. Subcircuito 1

Diseña e implementa el subcircuito 1 utilizando los siguientes módulos de Logisim:

- Codificador de prioridad (Priority Encoder).
- Visor display de 7 segmentos (Hex Digit Display), para visualizar números BCD.

TECNOLOGÍA DE COMPUTADORES. CURSO 2021/22.

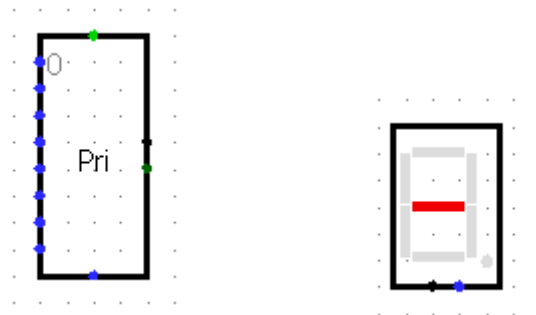
PRÁCTICA N° 3: SISTEMAS COMBINACIONALES.

SUBCIRCUITOS

Semana del 8 al 11 de noviembre.

Sugerencias:

- Pasar el cursor por encima de los pequeños puntos de los módulos para ver y entender qué son las entradas y salidas de los módulos.
- Usar el componente Separador (*Splitter*) (en el panel *Wiring*) para convertir datos (variables) binarios separados en vectores de datos y viceversa.



Módulos Priority Encoder y Hex Digit Display

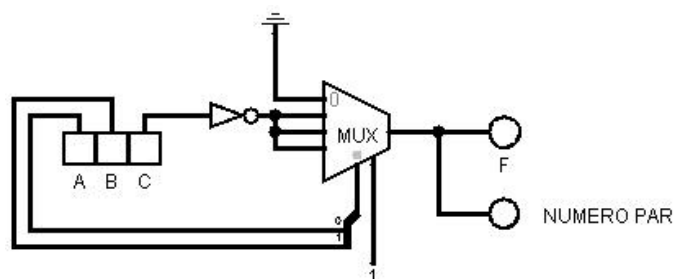
Simplemente se toma el componente Codificador con prioridad y se le introduce el dato. El componente se encarga de codificar en binario la posición del 1 más alto. La única precaución que hay que tener es poner el dígito más significativo de la entrada en el dígito más significativo del codificador con prioridad. En el circuito que se ofrece como solución se da por supuesto que hay al menos un bit a 1. En caso contrario la solución queda indeterminada.

Parte 2. Subcircuito 2

Diseña e implementa el subcircuito 2 utilizando un MUX 4x1 y las puertas lógicas necesarias.

Sean ABC los bits que han excitado el display. La tabla de verdad que resuelve el apartado 2 será:

| A | B | C | F | Justificación |
|---|---|---|---|---------------|
| 0 | 0 | 0 | 0 | 0 excepción |
| 0 | 0 | 1 | 0 | 1 impar |
| 0 | 1 | 0 | 1 | 2 par |
| 0 | 1 | 1 | 0 | 3 impar |
| 1 | 0 | 0 | 1 | 4 par |
| 1 | 0 | 1 | 0 | 5 impar |
| 1 | 1 | 0 | 1 | 6 par |
| 1 | 1 | 1 | 0 | 7 impar |



cuya implementación con un MUX 4x1 podría ser la indicada en la Figura adjunta dada. En principio, A sería la variable más significativa (peso 1) y en esta solución está como menos significativa (peso 0). En este caso particular es indiferente, puesto que la salida F para AB=01 y AB=10 es la misma.

TECNOLOGÍA DE COMPUTADORES. CURSO 2021/22.

PRÁCTICA N° 3: SISTEMAS COMBINACIONALES.

SUBCIRCUITOS

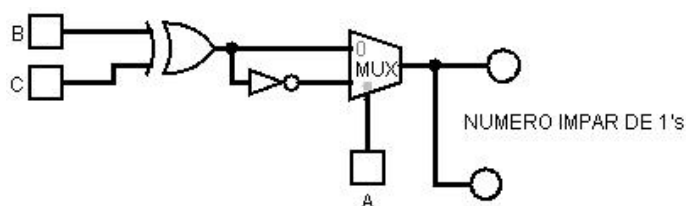
Semana del 8 al 11 de noviembre.

Parte 3. Subcircuito 3

Diseña e implementa el subcircuito 3 de la forma más sencilla posible, pero colocando un MUX 2x1 en la salida.

Sean ABC los bits que han excitado el display. Son los mismos del apartado 2. La tabla de verdad que resuelve el apartado 3 será:

| A | B | C | G | Justificación |
|---|---|---|---|-----------------------|
| 0 | 0 | 0 | 0 | 000, cero unos, par |
| 0 | 0 | 1 | 1 | 001, un uno, impar |
| 0 | 1 | 0 | 1 | 010, un uno, impar |
| 0 | 1 | 1 | 0 | 011, dos unos, par |
| 1 | 0 | 0 | 1 | 100, un uno, impar |
| 1 | 0 | 1 | 0 | 101, dos unos, par |
| 1 | 1 | 0 | 0 | 110, dos unos, par |
| 1 | 1 | 1 | 1 | 111, tres unos, impar |



Se puede observar que G es una puerta XOR de B y C para A=0 y su negada (XNOR) para A=1, con lo que bastaría poner una puerta de cada tipo, o una puerta XOR y un inversor, y elegir con la variable A qué entrada sacar.

A modo de resumen, los apartados 2 y 3 se pueden resumir en la tabla adjunta:

| ÍNDICE DEL BIT PRIORITARIO SUBC. 1 | ¿ES PAR? SUBC. 2 | ¿ES IMPAR EL NÚMERO DE UNOS? SUBC. 3 |
|---------------------------------------|---------------------|---|
| 0 | NO, excepción | NO (G=0) |
| 1 | NO (F=0) | SÍ (G=1) |
| 2 | SÍ (F=1) | SÍ (G=1) |
| 3 | NO (F=0) | NO (G=0) |
| 4 | SÍ (F=1) | SÍ (G=1) |
| 5 | NO (F=0) | NO (G=0) |
| 6 | SÍ (F=1) | NO (G=0) |
| 7 | NO (F=0) | SÍ (G=1) |

Parte 4. Circuito main. Integración

Implementa el circuito *main* utilizando los subcircuitos 1, 2 y 3.

Se trata simplemente de tomar los subcircuitos y “ensamblarlos”. El vector de 8 bits y el display se colocan en *main*, así como las salidas F y G.

En la Figura 1 del enunciado se puede ver el aspecto que tendría *main*.