



FUNDAMENTOS DE COMPUTADORES

PRÁCTICA 1: MONTAJE DE UN CIRCUITO COMBINACIONAL USANDO PUERTAS LÓGICAS

El objetivo de esta primera práctica es tomar contacto con el puesto de laboratorio, aprender a realizar el montaje de un circuito combinacional y conseguir que funcione correctamente.

Cada una de las prácticas constará de dos fases:

1. **Diseño** del circuito que se pide en el enunciado.
 2. **Montaje y depuración** de dicho diseño hasta conseguir que funcione correctamente.
- La fase de diseño consiste en realizar, en papel, el diseño (tabla de verdad, expresiones de conmutación,...) del problema que se plantea en el enunciado de cada una de las prácticas. Para realizar esta fase hay que aplicar los conocimientos de circuitos combinacionales explicados en las clases de teoría y problemas.
 - La fase de montaje se realizará en el entrenador que hay en cada puesto del laboratorio. El material que se necesita para el montaje del circuito se encuentra en el maletín de laboratorio. Este maletín se debe pedir una vez el alumno se haya dado de alta en laboratorios, a partir del 1 de noviembre.

En esta práctica hay que diseñar y montar en el laboratorio un sistema combinacional que realice la conversión de código Gray de 4 bits a binario (véase figura 1).



Figura 1

El Código Gray es un caso particular de código binario (patentado por Frank Gray en 1953). Consiste en una ordenación de 2^n códigos binarios de tal forma que cada código sólo tenga un dígito binario distinto a su predecesor. Un código Gray de 2 bits sería el de la figura 2:

0	0
0	1
1	1
1	0

Figura 2: Código Gray de 2 bits

Una forma sencilla de construir un código Gray de n bits es utilizar un código Gray de $n-1$ bits y añadirle un cero por la izquierda. A continuación para completar el código Gray

añadiremos un 1 por la izquierda al código Gray de n-1 bits empezando por el último elemento.

Por ejemplo, para obtener un código Gray de 3 bits partimos del código Gray de 2-bits que se muestra en la figura 2. Se añade un “0” a la izquierda del código Gray de 2-bits. Con esto tenemos los 4 primeros códigos del total de 8. Luego se añade un “1” en la columna de la izquierda y se colocan los códigos Gray de 2 bits en orden inverso. El resultado se muestra en la figura 3. En ella se puede ver que la línea roja hace de espejo para las dos columnas de menor peso.

0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

Código Gray 2-bits

Código Gray 2-bits
En orden inverso

Figura 3: Código Gray de 3 bits

Esta técnica de codificación se originó cuando los circuitos lógicos digitales se realizaban con válvulas de vacío y dispositivos electromecánicos. Los contadores necesitaban potencias muy elevadas a la entrada y generaban picos de ruido cuando varios bits cambiaban simultáneamente. El uso de código Gray garantizó que en cualquier transición variaría tan sólo un bit.

En la actualidad, el código Gray se sigue empleando para el diseño de cualquier circuito electrónico combinacional mediante el uso de un Mapa de Karnaugh, ya que el principio de diseño de buscar transiciones más simples y rápidas entre estados sigue vigente, a pesar de que los problemas de ruido y potencia se hayan reducido.

A modo de ejemplo, a continuación se desarrolla un sistema combinacional que realiza la conversión de código Gray de **3 bits** a binario siguiendo las fases descritas anteriormente.

1. Fase de diseño

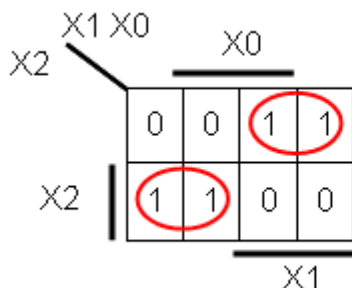
Tabla de verdad:

X2	X1	X0	Z2	Z1	Z0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	1
1	0	1	1	1	0
1	1	0	1	0	0
1	1	1	1	0	1

Mapas de Karnaugh y expresiones simplificadas:

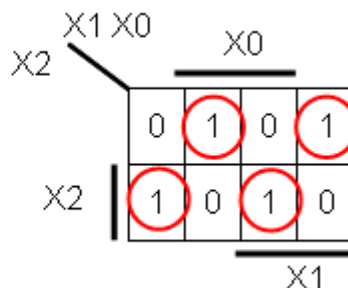
$$Z2 = X2$$

Para la salida **Z1**:



$$Z1 = X2 \oplus X1$$

Para la salida **Z0**:



$$Z0 = X2 \oplus X1 \oplus X0 = (X2 \oplus X1) \oplus X0$$

Implementación lógica:

La figura 4 muestra la implementación (una representación gráfica del diseño hecha mediante componentes lógicas, conexiones entre ellos, entradas y salidas) del conversor de código usando puertas XOR.

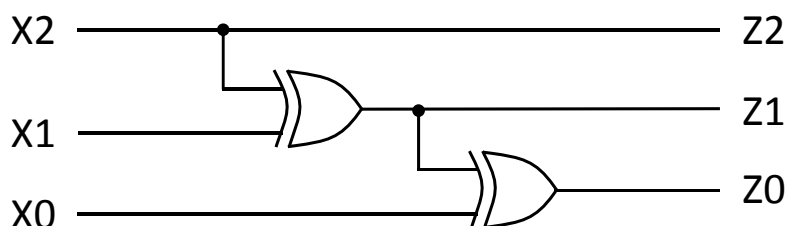


Figura 4: Implementación lógica del conversor de código

En esta práctica el **alumno tendrá que diseñar el conversor de código Gray de 4 bits a binario**: obtener la tabla de verdad y la implementación del sistema usando puertas XOR de dos entradas. **Se tiene que traer en el hecho de casa y reflejado en el correspondiente cuadernillo que debe presentarse al profesor de laboratorio al principio de la sesión.**

2. Fase de montaje y depuración

El montaje de un circuito se realiza en el entrenador (véase figura 5) que hay en cada uno de los puestos del laboratorio. Dicho entrenador está compuesto de varias zonas, cada una de las cuales está destinada a una función.

En las prácticas de esta asignatura se van a usar las siguientes funciones del entrenador:

A. **“DC POWER”**. Es el módulo de alimentación y contiene:

- Un generador de **tensión fijo** (5 V, -5 V)
- Un generador de **tensión variable** (0 ~15 V, 0 ~ -15 V). Girando el potenciómetro “+V” en sentido horario el rango del voltaje varía entre 0 y 15 V. Girando el potenciómetro “-V” observamos que dicha variación se produce entre 0 y -15 V.

- El terminal “**GND**” es el terminal de tierra (0 V).

B. “8 BITS LEDS DISPLAYS”

- Estos LEDS se van a usar para mostrar el valor de las **salidas del circuito**.
- Para conectar cada salida del circuito a un led hay que pinchar un cable en el terminal correspondiente
- Si el led está apagado representa salida “0” y encendido salida “1”

C. “8 BITS DATA SWITCHES”

- Estos switches (interruptores) se van a usar para las **entradas del circuito**.
- Para conectar cada entrada del circuito a un interruptor hay que pinchar un cable en el terminal que hay encima
- Con el interruptor hacia abajo la entrada vale “0” y hacia arriba “1”

D. ZONA CENTRAL

- Esta zona es donde se coloca la placa base (figura 6) que es el lugar donde se monta el circuito.



Figura 5: Entrenador

La placa base está compuesta de terminales de conexión. En estos terminales hay que pinchar:

- Las distintas componentes del circuito.
- Cables para conectar dichas componentes.
- Cables para conectar el circuito con la fuente de alimentación y tierra.
- Cables para conectar las entradas del circuito con los interruptores y las salidas con los leds.

Para poder montar el circuito en la placa primero hay que conocer el interconexionado de dicha placa, es decir cuáles de sus terminales de conexión están conectados entre sí (son el mismo punto, están cortocircuitados).

El interconexionado de las placas que hay en el laboratorio es como se muestra en la figura 6. Los terminales de conexión sobre los que se ha trazado una línea de color rojo están cortocircuitados. Esto implica que **no se pueden poner en la misma línea dos o más patillas de un mismo componente**.

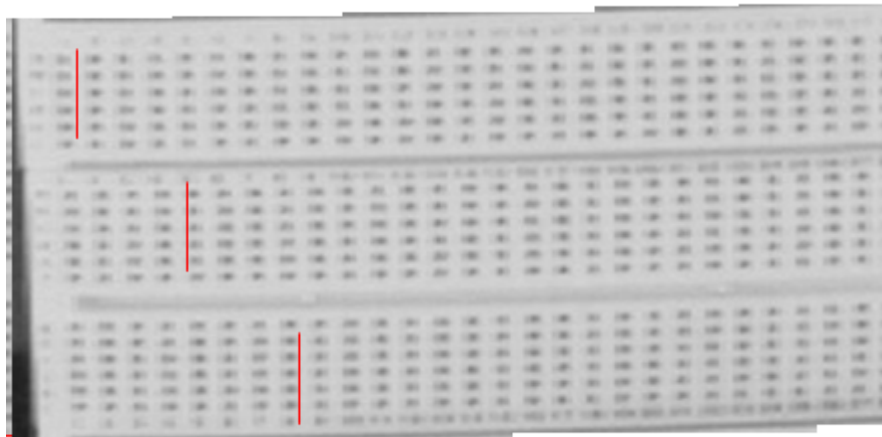


Figura 6: Placa base

Para montar estos circuitos se necesitan puertas lógicas. Estas se presentan encapsuladas en un circuito integrado denominado coloquialmente “cucaracha” o “chip”. Estos chips se encuentran en el maletín del laboratorio.

El único “chip” que se necesita en esta práctica es el **7486** que contiene **4 puertas XOR de 2 entradas** (véase figura 7).

Para saber cómo funcionan estos “chips” y cómo están interconectados por dentro hay que usar las llamadas hojas de características que proporciona el fabricante, las cuales contienen una especificación completa del “chip”. Estas se encuentran en el laboratorio dentro de la carpeta de electrónica.

A modo de ejemplo, la figura 7 muestra el interconexionado del “chip” 7486 que proporciona la hoja de características de este circuito integrado.

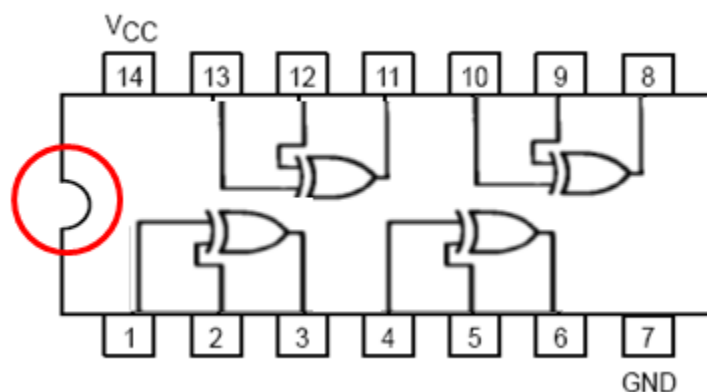


Figura 7: Contenido del circuito integrado 7486

Como se puede ver en la figura 7, este circuito integrado tiene en su interior cuatro puertas XOR de dos entradas cada una y en el exterior 14 patillas, cada una de las cuales tiene una función.

- Las patillas 14 y 7 son las encargadas de alimentar al circuito (darle energía) para que funcione: Conectar **Vcc** a **5V** (normalmente) y **GND** a **tierra**.
- El resto de las patillas son las entradas y salidas de las cuatro puertas XOR tal y como indican los dibujos

Para saber la numeración de las patillas en un “chip” hay que buscar una muesca como la que se ve en el dibujo de la figura 7. Colocando el “chip” en la misma dirección que en dicha figura, la numeración siempre empieza por la patilla inferior izquierda y continúa en sentido antihorario.

Implementación física:

Una vez se sabe qué “chips” se necesitan para montar el circuito, debe completarse el diseño lógico realizado con anterioridad indicando en qué “chip” se ubicará cada puerta del diseño (en caso de haber más de un “chip”) y qué pin del “chip” se utilizará para cada entrada/salida de cada puerta. Asimismo deberá indicarse qué switch/led del entrenador se utilizará para cada entrada/salida primaria del circuito. El resultado, que llamaremos implementación física, puede verse en la figura 8.

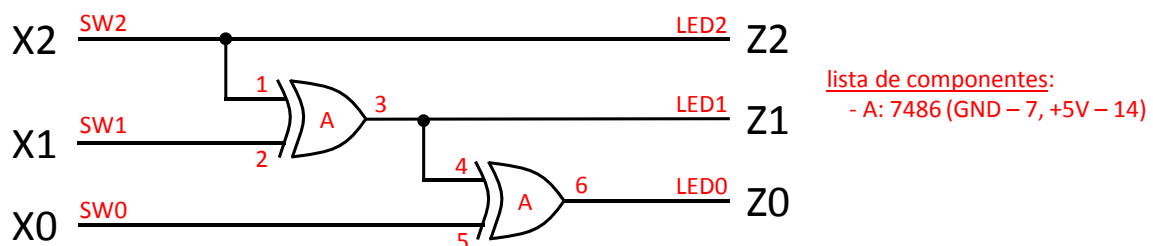


Figura 8: Implementación física del convertidor de código

El alumno tendrá que hacer lo mismo con el circuito que haya obtenido para el convertidor de código Gray de 4 bits a binario. **El dibujo del circuito implementado se tiene que hacer en casa y en el correspondiente cuadernillo.**

Montaje:

Llegados a este punto hay que montar el circuito. Con el entrenador apagado, se pincharán en la placa cada uno de los chips necesarios y a continuación, tomando como base la implementación física, se irán pinchando los cables necesarios uno a uno. Para no olvidar ninguno se recomienda que cada vez que se ponga un cable se tachen del dibujo los correspondientes números de pin que han quedado interconectados. La figura 9 muestra esquemáticamente como quedaría montado el convertidor de 3 bits en el entrenador.

Para que un chip funcione adecuadamente debe tenerse en cuenta que:

- Tiene que estar alimentado ($V_{cc} = +5V$ y $GND = 0V$).
- Puede tener salidas que no estén interconectadas a nada.
- Si se está utilizando una puerta lógica todas sus entradas **se utilicen o no** deben estar conectadas a un valor.
- Las puertas lógicas que no se estén utilizando no necesitan tener sus entradas conectadas a un valor.

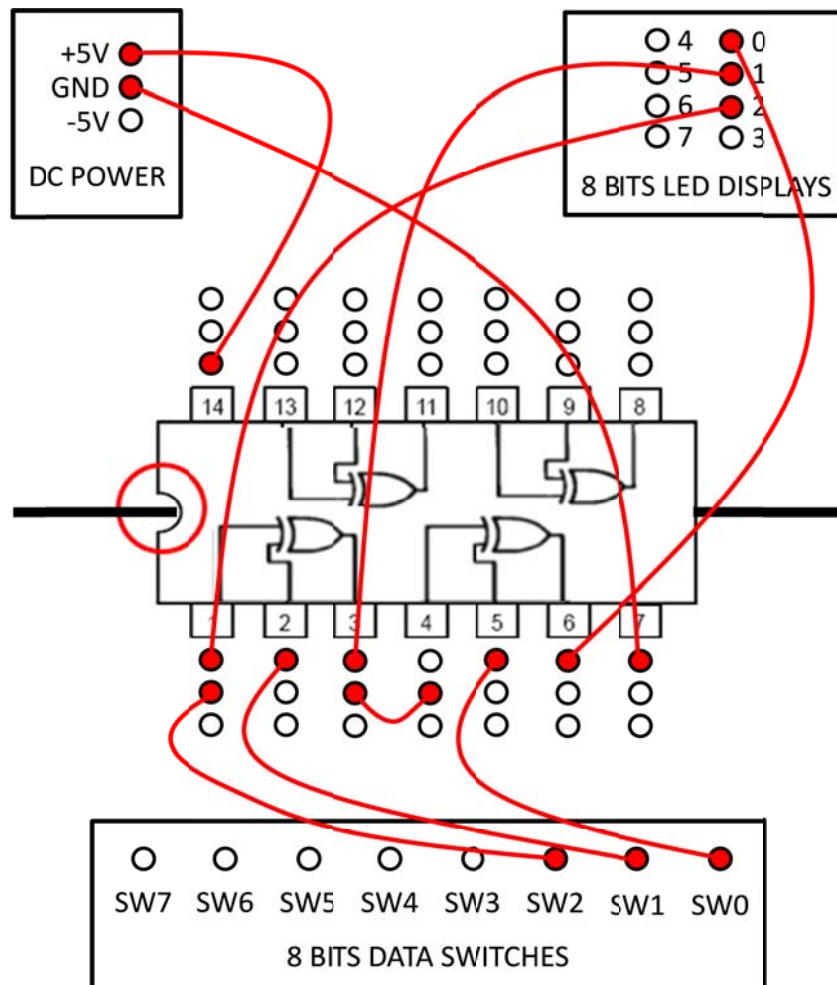


Figura 9: El conversor de código montado en el entrenador

Prueba:

Una vez montado el circuito hay que comprobar que funciona. Para ello, se encenderá el entrenador y se irán introduciendo mediante los switches las diferentes combinaciones de valores de la tabla de verdad y se irá comprobando en los leds si el resultado es el que corresponde.

Si la tabla de verdad calculada por el circuito es la misma que la tabla de verdad fijada por la especificación, el circuito es correcto. En otro caso será incorrecto y deberá depurarse. A continuación se explican los pasos a seguir para encontrar errores en circuitos combinacionales.

Depuración de un circuito combinacional

Si al comprobar el funcionamiento del circuito se detecta que el estado de una de las salidas no se corresponde con el esperado para la configuración binaria presente en las entradas, se debe buscar la causa del error. Para ello se procede del siguiente modo:

- i) Se debe repasar el diseño que se está implementando para ver si se ha producido un error en la fase de diseño. En caso contrario, se debe pasar al punto ii.
- ii) Una vez que se está seguro que el error se encuentra en la implementación hardware o física del circuito, se debe introducir al circuito la configuración binaria de entrada que produce la salida incorrecta. A continuación, se aísla la

- salida incorrecta S. Para ello se desconecta esta salida de todos los puntos del circuito donde estuviera conectada. En este momento pueden ocurrir dos cosas:
- La salida S cambia y da el valor correcto. Luego esta salida seguramente esté conectada a otra salida del circuito que la está “forzando”, pasar a la fase iii.
 - La salida S sigue dando un valor incorrecto, luego hay un módulo que produce una salida incorrecta, pasar a la fase iv.
- iii) La salida S está siendo “forzada” por la salida de otro módulo, está conectada a tierra o está conectada a polarización. Para localizar donde está el error, hay dos procedimientos:
- Seguir en la implementación de donde viene cada uno de los cables conectados a la salida S.
 - Aislar la salida S e ir conectando, uno por uno, los cables a la salida. En el momento en el que el valor de la salida S vuelva a ser incorrecto se habrá identificado la conexión errónea. Corregir la conexión errónea.
- iv) Se ha localizado un módulo M que produce una salida incorrecta. Si la salida del módulo lógico se corresponde con el valor de sus entradas significa que el módulo está funcionando correctamente. Por tanto, alguna de las entradas debe tener un valor lógico incorrecto. Esta entrada se corresponde con una salida de otro módulo lógico. Repítase el proceso a partir del paso II para esta nueva salida. Si la salida del módulo M no se corresponde con el de las entradas, entonces la puerta no está funcionando correctamente debido a que el “chip” está mal polarizado o está estropeado.

El procedimiento de depuración de sistemas combinatoriales se presenta esquemáticamente en el siguiente organigrama:

