

Clase 13

Compuertas lógicas

13.1 Introducción

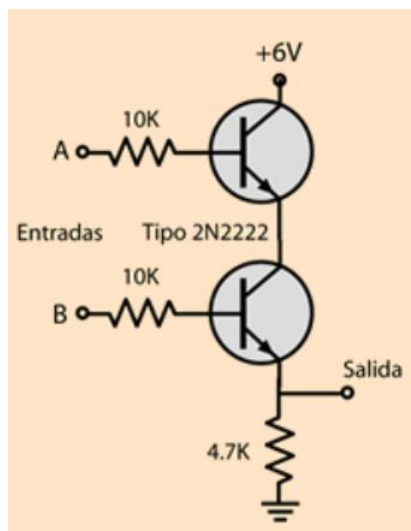
La clase pasada hemos introducido el tema de álgebra de Boole y operadores lógicos, en esta clase concluiremos este concepto llevándolo al área de la programación en circuitos lógicos mediante las compuertas lógicas, algo que a final de la clase pasada ya introducimos.

13.2 Compuertas lógicas

Las compuertas lógicas son los elementos básicos que permiten la construcción de un circuito digital. Las compuertas lógicas son equivalentes a los operadores lógicos, solo que con las compuertas nos referimos a dispositivos físicos contruidos con dispositivos electrónicos tales como transistores y resistencias; mientras que los operadores lógicos son funciones del álgebra de Boole.

Es así que encontraremos diversas compuertas lógicas, como por ejemplo AND, OR, NAND, NOR, XOR y XNOR y con ellas a su vez se pueden armar circuitos más complejos, lo que sería el equivalente a las funciones lógicas de la clase pasada.

Por ejemplo, el siguiente es el circuito eléctrico correspondiente a una compuerta AND construida con dos transistores y tres resistencias:



Obtener un valor 1 en la salida equivaldría a obtener voltaje en la salida. Para esto es necesario que ambos transistores estén en saturación, es decir, que estén conduciendo corriente entre el colector y el emisor; y esto equivale a que ambas entradas A y B reciban voltaje, es decir un 1 lógico.

Con lo cual la salida será 1 si ambas entradas son 1. Quedando formada la tabla de verdad correspondiente a la compuerta AND.

Este modelo de desarrollo con transistores y resistencias se denomina RTL (Resistor Transistor Logic) y fue el utilizado en los inicios de la informática para llevar a cabo secuencias lógicas que permitieron el posterior desarrollo de la tecnología hasta llegar al día de hoy.

En la actualidad las tecnologías de construcción de compuertas y circuitos lógicos han ido avanzando y en la actualidad la tecnología que más se utiliza para la construcción de circuitos lógicos es la NMOS (Negative-channel Metal-Oxide Semiconductor), cuya base es un tipo de semiconductor que se carga negativamente de modo que los transistores se enciendan o apaguen con el movimiento de los electrones. Es una tecnología en donde se ha logrado que los transistores tarden muy poco en conmutar entre sus estados 0 o 1, con lo cual se alcanzan muy altas velocidades, del orden de los Gigahertz.

La gestión de la información binaria en las compuertas Lógicas se encuentra asociada a las operaciones lógicas con variables binarias, y cumplen con el álgebra de Boole.

El sistema de compuertas lógicas tiene la tarea de hacer cumplir las condiciones booleanas para cada operador específico, es decir que se trata de hardware que envían señales "1" o "0" cuando se cumple una determinada condición de entrada lógica. Podría decirse que se trata de circuitos de conmutación integrados en un chip.

13.3 Símbolos de las compuertas lógicas

Más allá de la tecnología con la que estén construidas y su diseño interno, cada compuerta lógica tiene su símbolo con la cual se la representa. Ingresando y saliendo de cada símbolo encontraremos segmentos que representan las entradas y la salida.

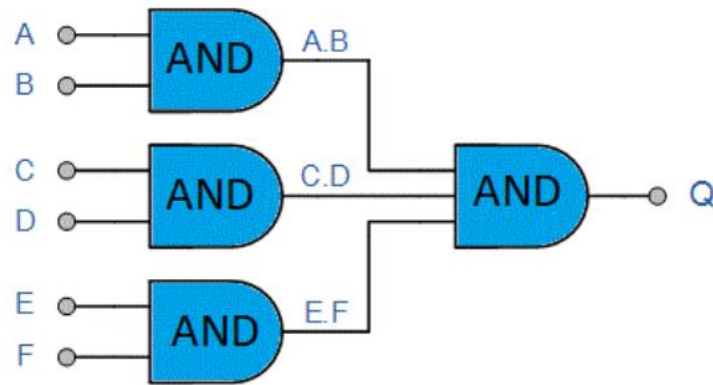
Compuerta AND

Cada una de estas compuertas lógicas posee dos variables de entrada, designadas por A y B y una salida binaria designada por Q. La compuerta AND produce la multiplicación lógica AND, lo que significa que La salida es 1 si la entrada A y la entrada B están ambas en el binario 1, de otra manera, la salida es 0.

La compuerta se representa de la siguiente manera:



Las compuertas AND son capaces de tener más de dos entradas, podemos logrando utilizando varias compuertas AND. En ese caso la salida es 1 si todas las entradas son 1, como se muestra en la siguiente imagen.



Compuerta OR

La compuerta OR se simboliza de la siguiente manera:



La salida es 1 si la entrada A o la entrada B o ambas entradas son 1; de otra manera, la salida es 0.

Ejemplo de una compuerta lógica OR, fabricada con tecnología TTL. El circuito 74LS32 o SN74LS32N conocido como compuerta OR es una puerta lógica digital que implementa la disyunción lógica con dos entradas. Consta de cuatro compuertas OR independientes.

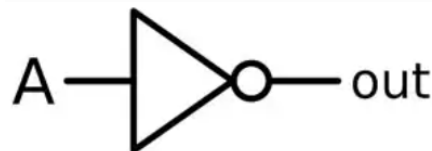


Se pueden construir compuertas OR con más de dos entradas y por definición la salida es 1 si cualquier entrada es 1.

Compuerta NOT

El circuito NOT básicamente es un inversor que es capaz de invertir el nivel lógico de una señal binaria. En el caso que la variable binaria tenga un valor "0", la compuerta NOT cambiará su estado al valor "1" y viceversa.

Se representa de la siguiente manera:



Compuerta NAND

La compuerta NAND es el complemento de la función AND. Las compuertas NAND son capaces de ofrecer más de dos entradas, y la salida es siempre el complemento de la función AND.



Compuerta NOR

La compuerta NOR es el complemento de la compuerta OR. Las compuertas NOR son capaces de tener más de dos entradas, y la salida siempre va a ser el complemento de la función OR.



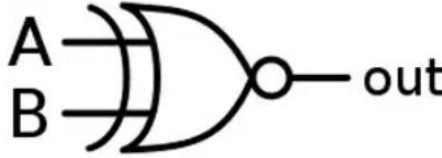
Compuerta XOR

La puerta XOR, también llamada OR exclusiva, cumple que una salida es 1 si una, y solo una de las entradas a la puerta es 1. Si las dos entradas son 0 o son 1, resultará en una salida 0.



Compuerta XNOR

La compuerta XNOR, también conocida como puerta NO-exclusiva, se trata básicamente del complemento de la puerta OR exclusiva, y su función booleana es $A \cdot B + \sim A \cdot \sim B$



13.4 Circuitos lógicos

Un circuito lógico es aquel que maneja información binaria, es decir, que tanto sus entradas como sus salidas se puede representar con valor 1 (alto) o 0 (bajo).

Las compuertas lógicas serían los circuitos lógicos más elementales, sin embargo, hay otros circuitos lógicos que podemos obtener, por ejemplo, mediante combinaciones de las compuertas lógicas.

Un circuito lógico sería el equivalente a una función en el álgebra de Boole.

Según la cantidad de compuertas que compongan cada circuito, se lo clasifica tradicionalmente de la siguiente manera:

- SSI (Small Scale Integration), escala de integración pequeña, hasta 10 compuertas por circuito.
- MSI (Medium Scale Integration), escala de integración media, de 10 a 100 compuertas por circuito.
- LSI (Long Scale Integration), escala de integración grande, de 100 a 10000 compuertas por circuito.
- VLSI (Very Long Scale Integration), escala de integración muy grande, más de 1000 compuertas por circuito.

Hoy en día se ha superada esta clasificación, puesto que los microprocesadores modernos tienen varios millones de compuertas en el mismo chip.

13.4.1 Sumador lógico

Como ejemplo de un circuito lógico sencillo, vamos a explicar y mostrar el funcionamiento de un sumador lógico.

El siguiente circuito recibe dos bits y devuelve el resultado de la suma binaria de dichos bits, es así que las operaciones posibles pueden ser:

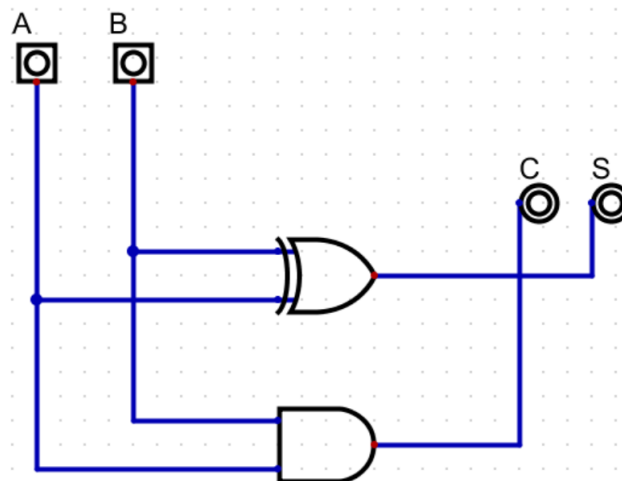
$$0 + 0 = 0$$

$$1 + 0 = 1$$

$$0 + 1 = 1$$

$$1 + 1 = 10$$

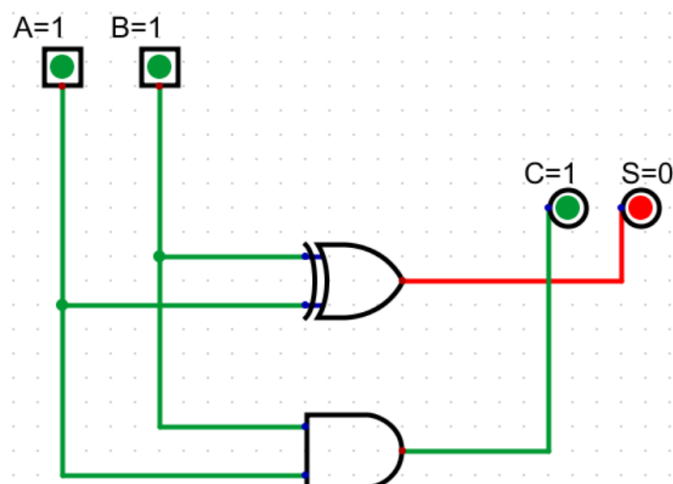
Vemos que en el último caso necesitaremos dos bits de salida.



Como vemos el circuito incluye una compuerta XOR y una AND, la compuerta XOR tal como introducimos al final de la clase pasada equivale a la suma de dos bits para los casos $0+0$, $0+1$ y $1+0$. En estos casos el bit de salida C será 0 como corresponde, pues es la salida de una compuerta AND.

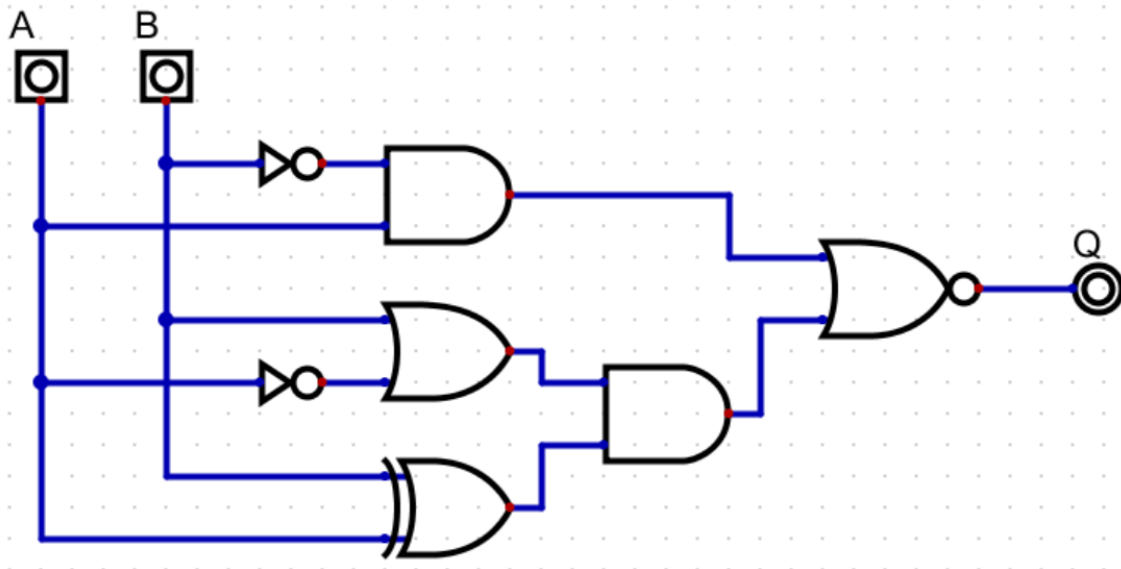
Para el caso $1+1$ el bit de la salida S es el correcto (es decir 0) y el 1 de la salida C se logra mediante la compuerta AND, pues cuando A y B sean 1, C será 1.

A continuación, la imagen del caso $1 + 1 = 10$



13.4.2 Tabla de verdad de un circuito lógico

Vamos a hacer, a modo de ejemplo, un ejercicio en el cual tendremos que obtener la función lógica y la tabla de verdad a partir del siguiente circuito lógico:



Como vemos, tenemos 7 compuertas lógicas:

- 2 compuertas NOT
- 2 compuertas AND
- 1 compuerta OR
- 1 compuerta XOR
- 1 compuerta NOR

Vamos a armar la función lógica correspondiente, para esto se puede encarar el problema de distintas maneras, a mi en particular me parece más sencillo comenzar de atrás hacia adelante en este caso en el cuál hay una única salida, Q.

Voy a dejar un video en el aula con la explicación de la obtención de esta función lógica.

La función lógica correspondiente, cuyo valor es Q, sería:

$$Q = \sim ((A \cdot \sim B) + ((\sim A + B) \cdot ((\sim A \cdot B) + (A \cdot \sim B))))$$

Finalmente, realizamos la tabla de verdad:

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

Como dato adicional, vemos que esta función da por resultado 0 si los valores de las entradas son distintos, es decir si un valor es 1 y el otro no. En caso de que sean iguales da 1. Por lo tanto, esta función es equivalente a la negación del O exclusivo, es decir al XNOR.

Para diseñar estos ejemplos utilicé el software denominado "Digital":

[GitHub - hneemann/Digital: A digital logic designer and circuit simulator.](#)

El simulador permite variar los valores de las entradas y observar la salida correspondiente, así como obtener la fórmula de la función lógica, entre otros aspectos.

Hay varios simuladores de este estilo, incluso algunos que funcionan en el navegador web, como el siguiente perteneciente a la Universidad Nacional de La Plata:

[LogicSim - Simulador de Compuertas Lógicas V1.02 \(unlp.edu.ar\)](#)

Con esto finalizamos nuestra introducción a compuertas y circuitos lógicos que al fin y al cabo es la manera en la que está programado internamente un dispositivo informático, y de los cuales se pueden hacer uso mediante los lenguajes de bajo y alto nivel.

Continuaremos con algunas actividades en los foros. La próxima, y última clase del taller, será luego del receso de invierno.