

Cómo están tanto tiempo? Espero que por más que no sea regular el curso en estos momentos, sigan investigando por su cuenta. Además es muy importante que estén notificados de las novedades que existen en el mundo de la tecnología, la ciencia y el hacking mismo.

HDC

Hoy estaremos interiorizándonos en el mundo de la **electrónica**, de la mano de lo digital.

Estos dos términos son extranjeros al vocabulario mundano y común, así que es necesario comprender qué significan por separados y juntos.

¿Qué es la electrónica?

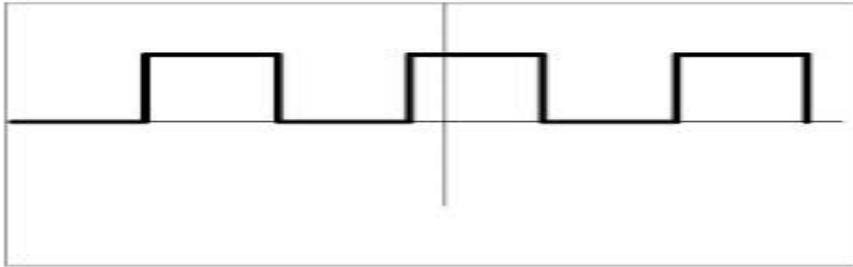
Voy a hacer la comparación más común. Todos conocen qué es la **electricidad** ¿Verdad? Bueno, la electrónica es lo mismo pero para **pequeñas potencias**, pequeños componentes.



¿Qué es lo digital?

Lo digital es el hermano de lo analógico. Para hacer referencias normales, lo analógico puede tomar valores de 0 a 100, con decimales; y un sistema digital toma valores que simplemente varían en prendido o apagado. Supongamos que entonces un termómetro nos da valores analógicos ya que puede tomar cualquier valor de la temperatura. Lo **digital** es, por ejemplo, un sistema que nos dice si hace frío o calor (**un simple bit, que tiene de valor 1 o 0**). Aunque también es posible y común

encontramos sistemas digitales de varios valores, pero que obviamente son **escalonados** y **bien definidos**.



Entonces: **¿Qué es la electrónica digital?**

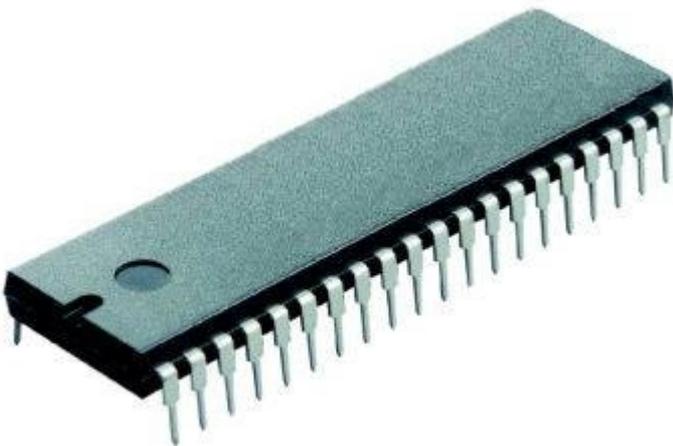
Simple, es la **ciencia** o la **técnica** que involucra a **componentes** de **baja potencia** que trabajan con **valores escalonados**.

“Si yo aprendo esto, te ganaste el premio nobel de enseñanza.”

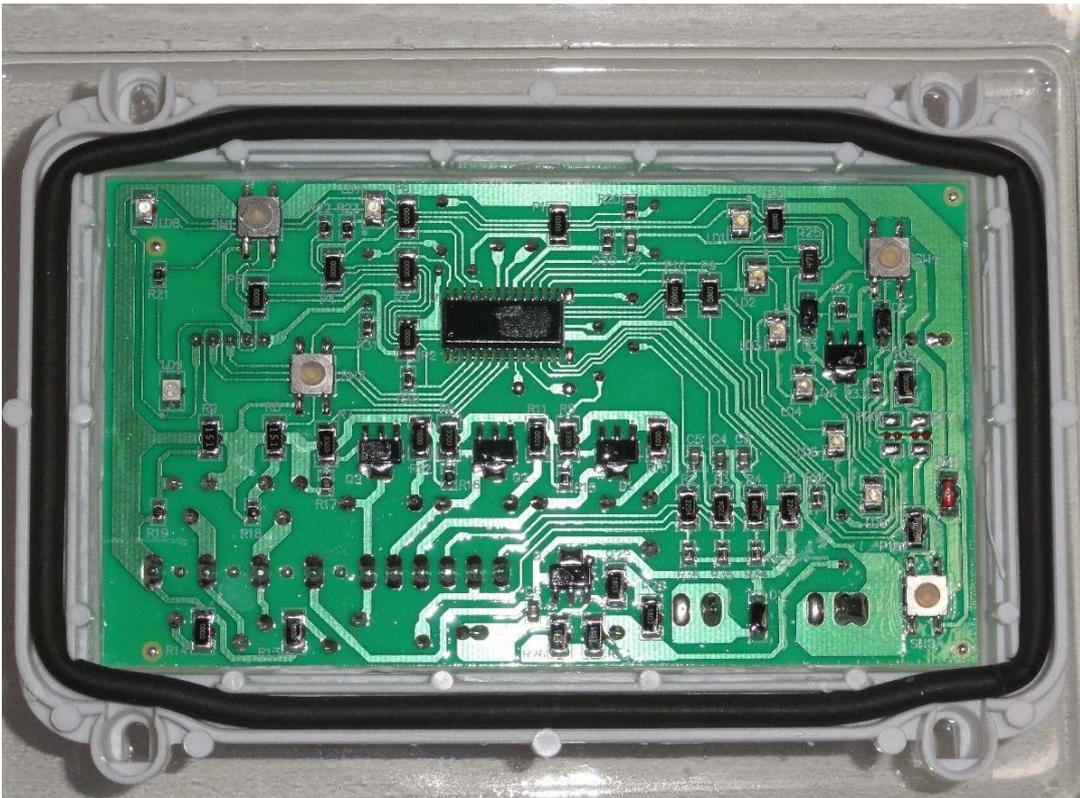
Tranquilo Manolo. Verás que no es tan difícil como parece.

Paso a paso. Vamos a saltarnos un par de fundamentos teóricos ahora mismo, para luego poder verlo correctamente cuando leamos electrónica analógica. Así que quienes sepan, no se alteren:D

En la electrónica, cada “cosito” que hay, es llamado **componente**. En la rama digital, encontramos sobre todo **circuitos integrados** -que llamaremos simplemente “integrados”- que son aquellas cucarachitas negras que encontramos en las **plaquetas**. Se llaman así porque dentro suyo hay circuitos con decenas, miles, y hasta millones de componentes (sí, todo en un cosito chiquito chiquito).

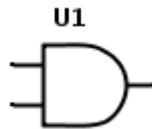


Las **plaquetas** son aquellas cosas verdes que encontramos en todos los objetos de electrónica que tienen cosas inentendibles para los mortales.



Veamos con qué componentes nos podemos encontrar:

Compuerta AND



“Se llama igual que la operación lógica. ¿Tiene algo que ver?”

¡Muy bien, Manolo! No tiene “algo” que ver. **Es justamente eso.** Entra **1 bit en cada patita** y la respectiva compuerta hace todo el trabajo. Se dice que “la salida está en 1 o en 0”.

Entonces, si en la entrada encontramos 1 y 1, en la salida tendremos o estará un 1, ya que corresponde a la operación **AND**.

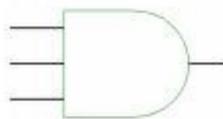
También tenemos la compuerta **NAND** donde la **salida esta negada**. Es decir que el bit que tenemos de salida se invierte -o se le aplica un NOT-. Para diferenciarla de la compuerta AND común, se le dibuja un círculo a la salida.



Otra cosa que también se utiliza (aunque no tanto), es que alguna de las **entradas estén negadas**. Para ésto no se utiliza ningún tipo de nombre específico, simplemente se dice cuántas entradas están negadas.



También tengamos en cuenta que no sólo hay compuertas con dos entradas sino que podemos tener una cantidad enorme de entradas. Aunque siempre contamos con **una sola salida**.



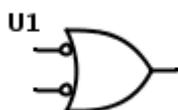
Aclaro que estos dibujitos con los cuales ilustramos a las compuertas se denominan **esquemáticos**. Luego conectaremos todo y veremos un **circuito esquemático** bien hecho.

Compuerta OR:

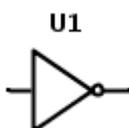


También llamado como la operación lógica, hace lo mismo que ella. En las entradas tenemos **un bit por pata** -que también puede ser llamado **terminal**-, y en la única salida saldrá el resultado de la operación.

Y al igual que la NAND tenemos la compuerta **NOR**, y con múltiples entradas y de entradas negadas.

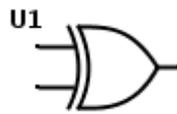


Compuerta NOT:



Ya que todas las compuertas tienen una única salida, esta compuerta NOT tiene únicamente **una entrada**. No hace más que cambiar el valor de la entrada.

Compuerta XOR:

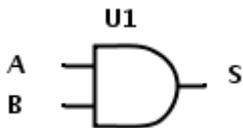


Hace el cálculo XOR con la entrada. Aviso que ésta no hace el cálculo lógico normal cuando tiene múltiples entradas. Lo veremos más adelante. A parte, **el negado es XNOR**.



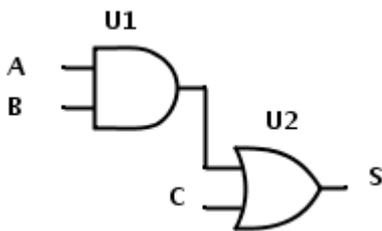
Ahora vamos a idear el esquemático de una función. Supongamos entonces que nuestra salida es S y decimos que

$$A \wedge B = S$$



Si a esto le agregamos algo más podría quedar de la siguiente manera

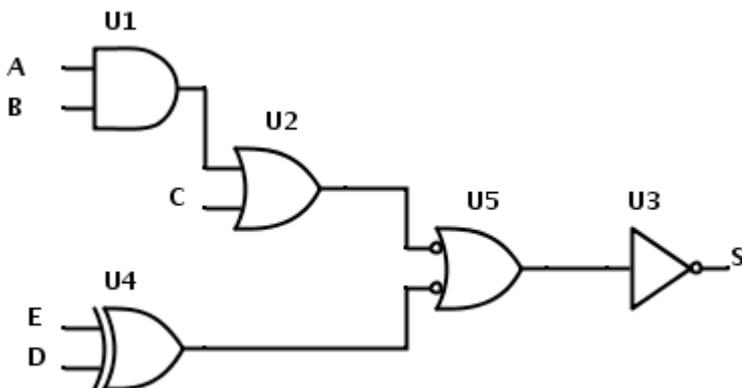
$$(A \wedge B) \vee C = S$$



(perdonen que la pata de U2 donde entra C, no llega al fondo. Debería pero el programa no me dejó.)

Y ahora algo más complicado

$$\neg(\neg(D \vee E) \vee \neg(A \wedge B) \vee C) = S$$



Veán que use una compuerta NOT en vez de negar la compuerta OR final. Esto sucede muchas veces en ingeniería porque se trabaja con los recursos que se tiene pues es mejor gastar la menor cantidad de recursos. Otra anotación es que cada compuerta tiene un nombre distinto y sirve mucho para la organización.

“¿Por qué letras? Yo trabajo con números.”

Bueno porque esto es un concepto teórico. **Así, no importa cual fuera el estado de las entradas, la función se aplica y corresponderá la salida debida.** Así que usá letras.

Ejercicios para hacer los esquemáticos:

1- $((A \wedge B) \vee D) \vee C = S$

2- $E \vee \neg(B \wedge A \wedge C \wedge D) = S$

3- $(A \vee B \vee C) \wedge \neg D = S$

NOTA: para terminar pueden hacer, también, la tabla de verdad y ver que sucedería en cada estado de las entradas.

La página que use para hacer los esquemáticos es www.digikey.com.

Cualquier cosa pueden mandarme mail a: r0add@hotmail.com

Para donaciones, pueden hacerlo en bitcoin en la dirección siguiente:

1HqpPJbbWJ9H2hAZTmpXnVuoLKkP7RFSvw

Roadd.

Este tutorial puede ser copiado y/o compartido en cualquier lado siempre poniendo que es de mi autoría y de mis propios conocimientos.