



:: Electrónica Digital - Introducción

Electrónica Digital

A partir de aquí vamos a dar inicio a un nuevo tutorial de electrónica digital, algo sencillo pero con intenciones de que resulte ameno, amigable, práctico y útil, bueno..., eso espero :o))

Por dudas, consultas o esas cosas, ni más que hablar busquen a quien sepa del tema :oP

Las críticas..., constructivas claro está enviarlas a mi correo o al foro, como quieras, por cierto, no existe crítica alguna si no viene acompañada de un buen fundamento, y ahí nos agarramos de los pelos, jejejejeje

En fin, buena suerte, y espero que te sirva...

:: **Electrónica Digital - Índice General**

Primeros contactos

[Lógica Positiva - Lógica negativa](#)

[Compuertas Lógicas Básicas - NOT, AND, OR, OR-EX](#)

[Compuertas Lógicas Combinadas - NAND, NOR, NOR-EX](#)

[Circuitos de Prueba - Disposición de terminales](#)

[Leyes de Morgan](#)

[Mapas de Karnaugh](#)

Circuitos Astables

[Circuitos Astables Parte I - Con Compuertas Lógicas](#)

[Circuitos Astables Parte II - Con Disparadores Schmitt Trigger](#)

[Circuitos Astables Parte III - Controlados o Conmutados](#)

Modulación por ancho de pulso

[Introducción - Ancho de pulso no simétrico](#)

[Modulación por ancho de pulso - Conmutado](#)

[Demodulación de Señales](#)

[Doblador de frecuencia](#)

Circuitos Monoestables

[Monoestable Sencillo](#)

[Monoestable con compuertas NOR](#)

[Monoestable con dos Inversores](#)

[Cerradura con teclado electrónico](#)

Circuitos Biestables

[Circuitos Biestables Parte I - Flip-Flop Básico RS](#)

[Circuitos Biestables Parte II - FF Tipo D y FF Master-Slave](#)

[Circuitos Biestables Parte III - Flip-Flop JK](#)

:: Electrónica Digital - Lección 1

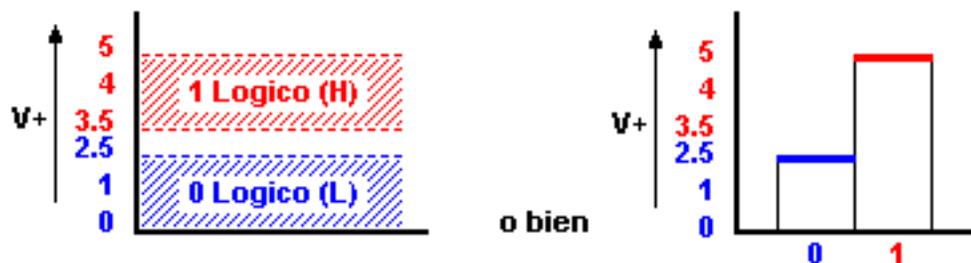
Qué es Electrónica Digital...?

Obviamente es una ciencia que estudia las señales eléctricas, pero en este caso son señales discretas, es decir, están bien identificadas, razón por la cual a un determinado nivel de tensión se lo llama estado alto (**High**) o Uno lógico; y a otro, estado bajo (**Low**) o Cero lógico.

Suponte que las señales eléctricas con que trabaja un sistema digital son 0V y 5V. Es obvio que 5V será el estado alto o uno lógico, pero bueno, habrá que tener en cuenta que existe la **Lógica Positiva** y la **Lógica Negativa**, veamos cada una de ellas.

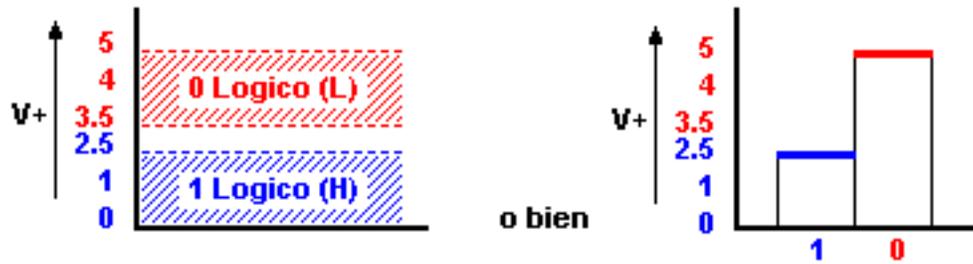
Lógica Positiva

En esta notación al 1 lógico le corresponde el nivel más alto de tensión (positivo, si quieres llamarlo así) y al 0 lógico el nivel mas bajo (que bien podría ser negativo), pero que ocurre cuando la señal no está bien definida...?. Entonces habrá que conocer cuales son los límites para cada tipo de señal (conocido como tensión de histéresis), en este gráfico se puede ver con mayor claridad cada estado lógico y su nivel de tensión.

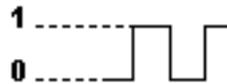


Lógica Negativa

Aquí ocurre todo lo contrario, es decir, se representa al estado "1" con los niveles más bajos de tensión y al "0" con los niveles más altos.



Por lo general se suele trabajar con lógica positiva, y así lo haremos en este tutorial, la forma más sencilla de representar estos estados es como se puede ver en el siguiente gráfico.



De ahora en más ya sabrás a que nos referimos con estados lógicos **1** y **0**, de todos modos no viene nada mal saber un poco más... ;-)

:: Electrónica Digital - Lección 2

Compuertas Lógicas

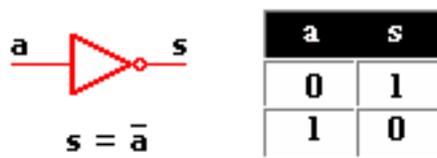
Las compuertas lógicas son dispositivos que operan con aquellos estados lógicos mencionados en la página anterior y funcionan igual que una calculadora, de un lado ingresas los datos, ésta realiza una operación, y finalmente, te muestra el resultado.



Cada una de las compuertas lógicas se las representa mediante un **Símbolo**, y la operación que realiza (**Operación lógica**) se corresponde con una tabla, llamada **Tabla de Verdad**, vamos con la primera...

Compuerta NOT

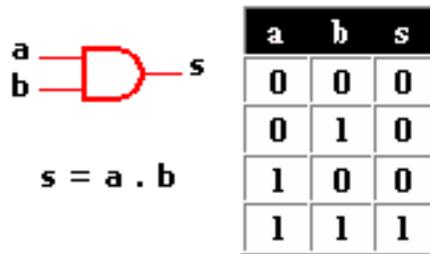
Se trata de un inversor, es decir, invierte el dato de entrada, por ejemplo; si pones su entrada a 1 (nivel alto) obtendrás en su salida un 0 (o nivel bajo), y viceversa. Esta compuerta dispone de una sola entrada. Su operación lógica es **s** igual a **a invertida**



Compuerta AND

Una compuerta AND tiene dos entradas como mínimo y su operación lógica es un producto entre ambas, no es un producto aritmético, aunque en este caso coincidan.

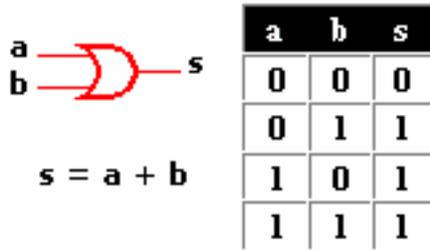
Observa que su salida será alta si sus dos entradas están a nivel alto



Compuerta OR

Al igual que la anterior posee dos entradas como mínimo y la operación lógica, será una suma entre ambas... *Bueno, todo va bien hasta que $1 + 1 = 1$* , el tema es que se trata de una compuerta **O Inclusiva** es como **a y/o b**

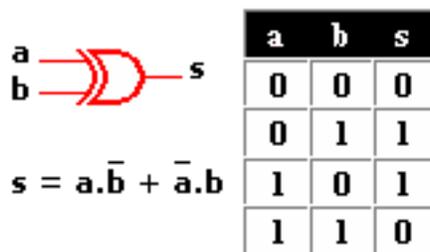
*Es decir, basta que una de ellas sea **1** para que su salida sea también **1***



Compuerta OR-EX o XOR

Es OR EXclusiva en este caso con dos entradas (puede tener mas claro...!) y lo que hará con ellas será una suma lógica entre **a** por **b invertida** y **a invertida** por **b**.

*Al ser **O Exclusiva** su salida será **1** si una y **sólo una** de sus entradas es **1***



Estas serían básicamente las compuertas mas sencillas. Es momento de complicar esto un poco más...

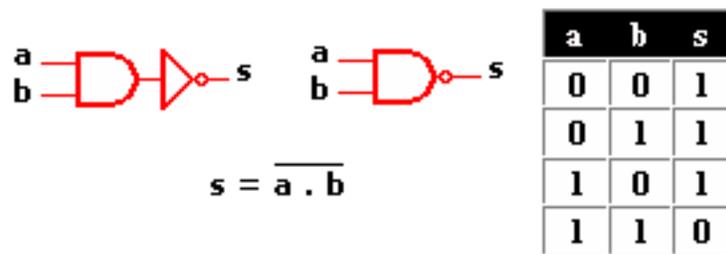
:: Electrónica Digital - Lección 3

Compuertas Lógicas Combinadas

Al agregar una compuerta NOT a cada una de las compuertas anteriores los resultados de sus respectivas tablas de verdad se invierten, y dan origen a tres nuevas compuertas llamadas **NAND**, **NOR** y **NOR-EX**... Veamos ahora como son y cual es el símbolo que las representa...

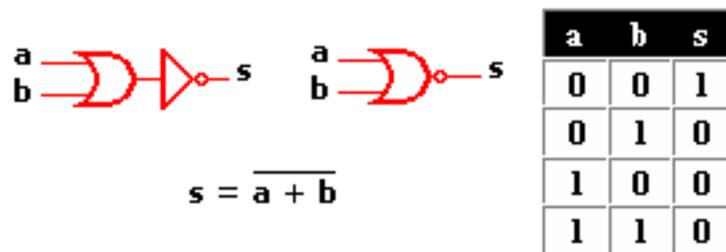
Compuerta NAND

Responde a la inversión del producto lógico de sus entradas, en su representación simbólica se reemplaza la compuerta NOT por un círculo a la salida de la compuerta AND.



Compuerta NOR

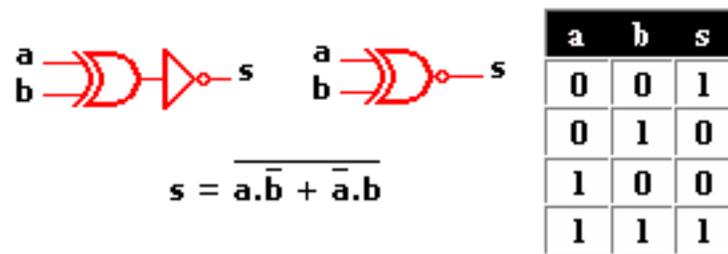
El resultado que se obtiene a la salida de esta compuerta resulta de la inversión de la operación lógica **o inclusiva** es como un **no a y/o b**. Igual que antes, solo agregas un círculo a la compuerta OR y ya tienes una NOR.



Compuerta NOR-EX

Es simplemente la inversión de la compuerta OR-EX, los resultados se pueden apreciar en la tabla de verdad, que bien podrías compararla con la anterior y notar la diferencia, el símbolo que la representa lo tienes en el

siguiente gráfico.



Buffer's

Ya la estaba dejando de lado..., no se si viene bien incluirla aquí pero de todos modos es bueno que la conozcas, en realidad no realiza ninguna operación lógica, su finalidad es amplificar un poco la señal (o refrescarla si se puede decir). Como puedes ver en el siguiente gráfico la señal de salida es la misma que de entrada.



Hasta aquí de teoría, nos interesa más saber como se hacen evidente estos estados en la práctica, y en qué circuitos integrados se las puede encontrar y más adelante veremos unas cuantas leyes que se pueden aplicar a estas compuertas para obtener los resultados que deseas...

:: Electrónica Digital - Lección 4

Circuitos Integrados y Circuito de Prueba

Existen varias familias de Circuitos integrados, pero sólo mencionaré dos, los más comunes, que son los TTL y CMOS:

Estos Integrados los puedes caracterizar por el número que corresponde a cada familia según su composición. Por ejemplo;

Los TTL se corresponden con la serie 5400, 7400, 74LSXX, 74HCXX, 74HCTXX etc. algunos 3000 y 9000.

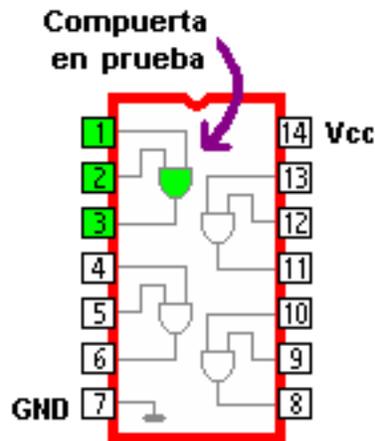
Los C-MOS y MOS se corresponde con la serie CD4000, CD4500, MC14000, 54C00 ó 74C00. en fin...

La pregunta de rigor... Cual es la diferencia entre uno y otro...?, veamos... yo comencé con los C-MOS, ya que disponía del manual de estos integrados, lo bueno es que el máximo nivel de tensión soportado llega en algunos casos a +15V, (especial para torpes...!!!), mientras que para los TTL el nivel superior de tensión alcanza en algunos casos a los +12V aproximadamente, pero claro estos son límites extremos, lo común en estos últimos es utilizar +5V y así son felices.

Otra característica es la velocidad de transmisión de datos, resulta ser, que los circuitos TTL son mas rápidos que los C-MOS, por eso su mayor uso en sistemas de computación.

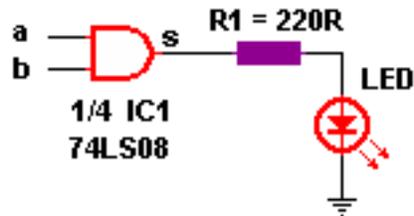
Suficiente... de todos modos es importante que busques la hoja de datos o datasheet del integrado en cuestión, distribuido de forma gratuita por cada fabricante y disponible en Internet... *donde más...?*

Veamos lo que encontramos en uno de ellos; en este caso un Circuito integrado 74LS08, un TTL, es una cuádruple compuerta AND. Es importante que notes el sentido en que están numerados los pines y esto es general, para todo tipo de integrado...

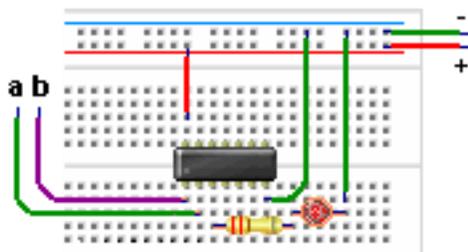


Comenzaremos con este integrado para verificar el comportamiento de las compuertas vistas anteriormente. El representado en el gráfico marca una de las compuertas que será puesta a prueba, para ello utilizaremos un fuente regulada de +5V, un LED una resistencia de 220 ohm, y por supuesto el IC que corresponda y la placa de prueba.

El esquema es el siguiente...



En el esquema está marcada la compuerta, como 1 de 4 disponibles en el Integrado 74LS08, los extremos **a** y **b** son las entradas que deberás llevar a un 1 lógico (+5V) ó 0 lógico (GND), el resultado en la salida **s** de la compuerta se verá reflejado en el LED, LED encendido (1 lógico) y LED apagado (0 lógico), no olvides conectar los terminales de alimentación que en este caso son el pin 7 a GND y el 14 a +5V. Montado en la placa de prueba te quedaría algo así...



Esto es a modo de ejemplo, *Sólo debes reemplazar IC1*, que es el Circuito Integrado que está a prueba para verificar su tabla de verdad.

¿Y en qué Circuito Integrado encuentro todas estas compuertas?...

Sabia que preguntarías eso... Para que puedas realizar las pruebas, en la web dejé los datos de algunos integrados.

:: Electrónica Digital - Lección 5

Antes de seguir... Lo primero y más importante es que trates de interpretar la forma en que realiza sus operaciones cada compuerta lógica, ya que a partir de ahora las lecciones se complican un poco más. Practica y verifica cada una de las tablas de verdad.

Leyes de De Morgan

Se trata simplemente de una combinación de compuertas de tal modo de encontrar una equivalencia entre ellas, esto viene a consecuencia de que en algunos casos no dispones del integrado que necesitas pero si de otros que podrían producir los mismos resultados que estas buscando.

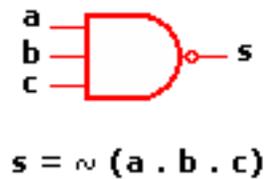
Para interpretar mejor lo que viene, considera a las señales de entrada como variables y al resultado como una función entre ellas. El símbolo de negación (operador NOT) lo representaré por "~", por ejemplo: **a . ~ b** significa **a AND NOT b**, se entendió...?

1º Ley:

El producto lógico negado de varias variables lógicas es igual a la suma lógica de cada una de dichas variables negadas. Si tomamos un ejemplo para 3 variables tendríamos..

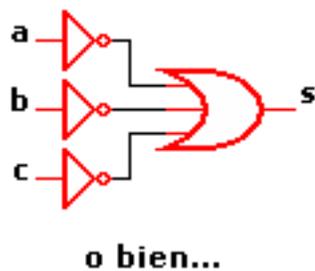
$$\sim (a.b.c) = \sim a + \sim b + \sim c$$

El primer miembro de esta ecuación equivale a una compuerta NAND de 3 entradas, representada en el siguiente gráfico y con su respectiva tabla de verdad.

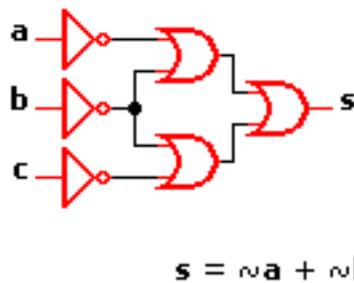


| a | b | c | s |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

El segundo miembro de la ecuación se lo puede obtener de dos formas...



| a | b | c | s |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |



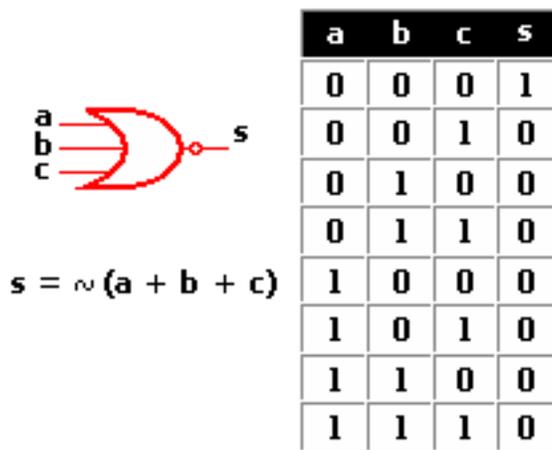
Fíjate que la tabla de verdad es la misma, ya que los resultados obtenidos son iguales. Acabamos de verificar la primera ley.

2º Ley:

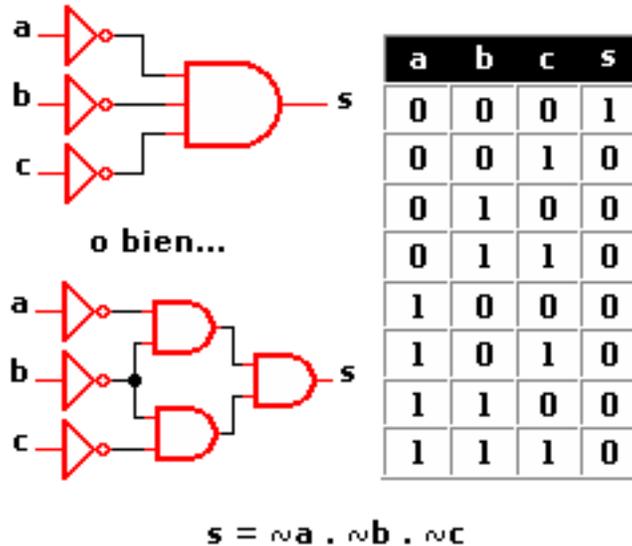
La suma lógica negada de varias variables lógicas es igual al producto de cada una de dichas variables negadas...

$$\sim (a + b + c) = \sim a . \sim b . \sim c$$

El primer miembro de esta ecuación equivale a una compuerta NOR de 3 entradas y la representamos con su tabla de verdad...



El segundo miembro de la ecuación se lo puede obtener de diferentes forma, aquí cité solo dos...



Nuevamente... Observa que la tabla de verdad es la misma que para el primer miembro en el gráfico anterior. Acabamos así de verificar la segunda ley de De Morgan.

Para concluir... Con estas dos leyes puedes llegar a una gran variedad de conclusiones, por ejemplo...

Para obtener una compuerta AND puedes utilizar una compuerta NOR con sus entradas negadas, o sea...

$$a \cdot b = \sim(\sim a + \sim b)$$

Para obtener una compuerta OR puedes utilizar una compuerta NAND con sus entradas negadas, es decir...

$$\mathbf{a + b = \sim (\sim a . \sim b)}$$

Para obtener una compuerta NAND utiliza una compuerta OR con sus dos entradas negadas, como indica la primera ley de De Morgan...

$$\sim (\mathbf{a.b}) = \sim \mathbf{a} + \sim \mathbf{b}$$

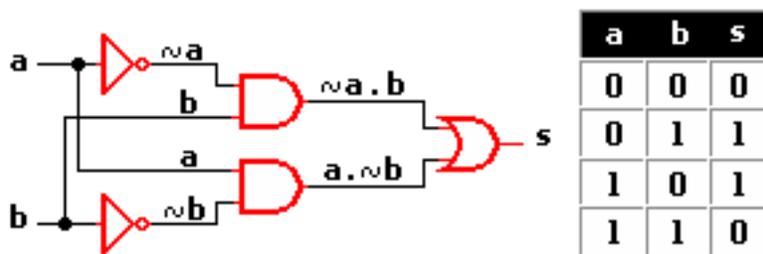
Para obtener una compuerta NOR utiliza una compuerta AND con sus entradas negadas, ...eso dice la 2º ley de De Morgan, así que... habrá que obedecer...

$$\sim (\mathbf{a + b}) = \sim \mathbf{a} . \sim \mathbf{b}$$

La compuerta OR-EX tiene la particularidad de entregar un nivel alto cuando una y sólo una de sus entradas se encuentra en nivel alto. Si bien su función se puede representar como sigue...

$$\mathbf{s = a . \sim b + \sim a . b}$$

te puedes dar cuenta que esta ecuación te indica las compuertas a utilizar, y terminarás en esto...

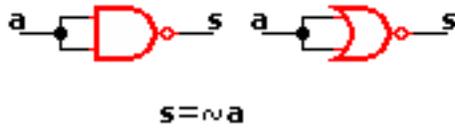


$$\mathbf{s = a . \sim b + \sim a . b}$$

Para obtener una compuerta NOR-EX agregas una compuerta NOT a la salida de la compuerta OR-EX vista anteriormente y ya la tendrás. Recuerda que su función es...

$$\mathbf{s = \sim (a . \sim b + \sim a . b)}$$

Para obtener Inversores (NOT) puedes hacer uso de compuertas NOR o compuertas NAND, simplemente uniendo sus entradas.



Existen muchas opciones más, pero bueno... ya las irás descubriendo, o las iremos citando a medida que vayan apareciendo, de todos modos valió la pena. *No crees...?*

:: Electrónica Digital - Lección 6

A estas alturas ya estamos muy familiarizados con las funciones de todos los operadores lógicos y sus tablas de verdad, todo vino bien..., pero... qué hago si dispongo de tres entradas (a, b y c) y deseo que los estados altos sólo se den en las combinaciones 0, 2, 4, 5 y 6 (decimal)...? Cómo combino las compuertas...? y lo peor, Qué compuertas utilizo...?. No te preocupes, yo tengo la solución, ...pégate un tiro... :o))

Bueno... NO...!!!, mejor no. Trataré de dar una solución verdadera a tu problema, preparado...?

Mapas de Karnaugh

Podría definirlo como un método para encontrar la forma más sencilla de representar una función lógica.

Esto es... Encontrar la función que relaciona todas las variables disponibles de tal modo que el resultado sea el que se está buscando.

Para esto vamos a aclarar tres conceptos que son fundamentales

a)- Minitérmino Es cada una de las combinaciones posibles entre todas las variables disponibles, por ejemplo con 2 variables obtienes 4 minitérminos; con 3 obtienes 8; con 4, 16 etc., como te darás cuenta se puede encontrar la cantidad de minitérminos haciendo 2^n donde n es el número de variables disponibles.

b)- Numeración de un minitérmino Cada minitérmino es numerado en decimal de acuerdo a la combinación de las variables y su equivalente en binario así...

| a | b | Minit. |
|---|---|----------|
| 0 | 0 | Minit. 0 |
| 0 | 1 | Minit. 1 |
| 1 | 0 | Minit. 2 |
| 1 | 1 | Minit. 3 |

Bien... El Mapa de Karnaugh representa la misma tabla de verdad a través de una matriz, en la cual en la primer fila y la primer columna se indican las posibles combinaciones de las variables. Aquí tienes tres mapas para 2, 3 y 4 variables...

Para 2 Variables

| a\b | 0 | 1 |
|-----|---|---|
| 0 | 0 | 1 |
| 1 | 2 | 3 |

Para 3 Variables

| a\bc | 00 | 01 | 11 | 10 |
|------|----|----|----|----|
| 0 | 0 | 1 | 3 | 2 |
| 1 | 4 | 5 | 7 | 6 |

Para 4 Variables

| ab\cd | 00 | 01 | 11 | 10 |
|-------|----|----|----|----|
| 00 | 0 | 1 | 3 | 2 |
| 01 | 4 | 5 | 7 | 6 |
| 11 | 12 | 13 | 15 | 14 |
| 10 | 8 | 9 | 11 | 10 |

Analicemos el mapa para cuatro variables, las dos primeras columnas (columnas adyacentes) difieren sólo en la variable **d**, y c permanece sin cambio, en la segunda y tercer columna (columnas adyacentes) cambia **c**, y d permanece sin cambio, ocurre lo mismo en las filas. En general se dice que...

Dos columnas o filas adyacentes sólo pueden diferir en el estado de una de sus variables

Observa también que según lo dicho anteriormente la primer columna con la última serían adyacentes, al igual que la primer fila y la última, ya que sólo difieren en una de sus variables

c)- Valor lógico de un minitérmino (esos que estaban escritos en rojo), bien, estos deben tener un valor lógico, y es el que resulta de la operación que se realiza entre las variables. lógicamente **0** ó **1**

Listo... Lo que haremos ahora será colocar el valor de cada minitérmino según la tabla de verdad que estamos buscando... diablos...!!! en este momento no se me ocurre nada, bueno si, trabajemos con esta...

| a\bc | 00 | 01 | 11 | 10 |
|------|----|----|----|----|
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 |

| a | b | c | s |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

El siguiente paso, es agrupar los unos adyacentes (horizontal o verticalmente) en grupos de potencias de 2, es decir, en grupos de 2, de 4, de 8 etc... y nos quedaría así...

| a\bc | 00 | 01 | 11 | 10 |
|-------------|-----------|-----------|-----------|-----------|
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 |

Te preguntarán que pasó con la fila de abajo... bueno, es porque no estas atento...!!! Recuerda que la primer columna y la última son adyacentes, por lo tanto sus minitérminos también lo son.

De ahora en más a cada grupo de unos se le asigna la unión (producto lógico) de las variables que se mantienen constante (ya sea uno o cero) ignorando aquellas que cambian, tal como se puede ver en esta imagen...

En este grupo cambia c por lo tanto le corresponde...

| a\bc | 00 | 01 | 11 | 10 |
|-------------|-----------|-----------|-----------|-----------|
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 |

En este grupo cambia b por lo tanto le corresponde...

$$\bar{a} \cdot \bar{b}$$

$$a \cdot \bar{c}$$

Para terminar, simplemente se realiza la suma lógica entre los términos obtenidos dando como resultado la función que estamos buscando, es decir...

$$f = (\bar{a} \cdot \bar{b}) + (a \cdot \bar{c})$$

Puedes plantear tu problema como una función de variables, en nuestro ejemplo quedaría de esta forma...

$$f(a, b, c) = S(0, 1, 4, 6)$$

F es la función buscada

(a, b, c) son las variables utilizadas

(0, 1, 4, 6) son los minitérminos que dan como resultado **1** o un nivel alto.

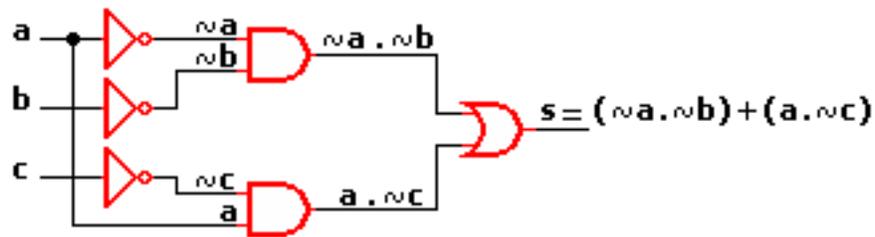
S La sumatoria de las funciones que producen el estado alto en dichos minitérminos.

Sólo resta convertir esa función en su circuito eléctrico correspondiente. Veamos, si la función es...

$$f = (\sim a \cdot \sim b) + (a \cdot \sim c) \text{ o sea...}$$

(NOT a AND NOT b) OR (a AND NOT c)

El esquema eléctrico que le corresponde es el que viene a continuación...



El resultado de todo este lío, es un circuito con la menor cantidad de compuertas posibles, lo cual lo hace más económico, por otro lado cumple totalmente con la tabla de verdad planteada al inicio del problema, y a demás recuerda que al tener menor cantidad de compuertas la transmisión de datos se hace más rápida.

En fin... Solucionado el problema...!!!

Por cierto, un día, mientras merodeaba por la red me encontré con un pequeño programa que hace todo este trabajo por tu cuenta, El programa se llama **Karma** Creado por Pablo Fernández Fraga, mis saludos Pablo...!!! está muy, pero muy bueno...!!! y puedes bajarlo desde la web.

Basta por hoy, muy pronto utilizaremos toda esta teoría y el programa de pablo (Karma) para diseñar una tarjeta controladora de motores paso a paso, mientras tanto averigua como funcionan estos motores.

Saludos lógicos para todos...!!!

:: Electrónica Digital - Lección 7

Osciladores, Multivibradores o Circuitos Astables - Parte I

Volví...!!!, esta vez prepara tu placa de pruebas y unos cuantos integrados, que lo que viene será todo práctica... Seguramente te llama la atención el nombre **Astable**. Bien... lo explicaré brevemente...

Existen tres circuitos clasificados según la forma en que retienen o memorizan el estado que adoptan sus salidas, estos son...

- **Circuitos Biestables o Flip-Flop (FF):** Son aquellos que cambian de estado cada vez que reciben una señal de entrada (ya sea nivel bajo o alto), es decir retienen el dato de salida aunque desaparezca el de entrada.
Conclusión: Poseen dos estados estables
- **Circuitos Monoestables:** Estos circuitos cambian de estado sólo si se mantiene la señal de entrada (nivel alto o bajo), si ésta se quita, la salida regresa a su estado anterior.
Conclusión: Poseen un sólo estado estable y otro metaestables
- **Circuitos Astables o Aestables:** Son circuitos gobernados por una red de tiempo R-C (Resistencia-Capacitor) y un circuito de realimentación, a diferencia de los anteriores se puede decir que **no poseen un estado estable sino dos metaestables**

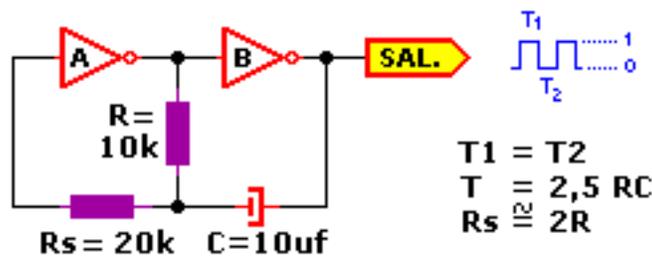
Y a estos últimos nos dedicaremos ahora, los otros dos los trataremos en las próximas lecciones...

De todos los circuitos astables el más conocido es el que se construye con un integrado NE555, el cual ya vimos como hacerlo tiempo atrás (en nuestro tutorial de electrónica básica). La idea es que veas todas las posibilidades que te brindan las compuertas lógicas y ésta es una de ellas, considerando que en muchos circuitos o diseños quedan compuertas libres (sin ser utilizadas) vamos a aprovecharlas para armar circuitos astables, timer's o temporizadores, o yo que se, como le quieras llamar.

Comencemos...

Oscilador Simétrico con compuertas NOT

Fue el primero que se me ocurrió y utiliza dos inversores o compuertas NOT.



Descripción:

Suponte que en determinado momento la salida del inversor B está a nivel "1", entonces su entrada esta a "0", y la entrada del inversor "A" a nivel "1". En esas condiciones **C** se carga a través de **R**, y los inversores permanecen en ese estado.

Cuando el capacitor alcanza su carga máxima, se produce la conmutación del inversor "A". Su entrada pasa a "0", su salida a "1" y la salida del inversor "B" a "0", se invierte la polaridad del capacitor y este se descarga, mientras tanto los inversores permanecen sin cambio, una vez descargado, la entrada del inversor "A" pasa nuevamente a "1", y comienza un nuevo ciclo.

Este oscilador es simétrico ya que el tiempo que dura el nivel alto es igual al que permanece en nivel bajo, este tiempo esta dado por **$T = 2,5 RC$**

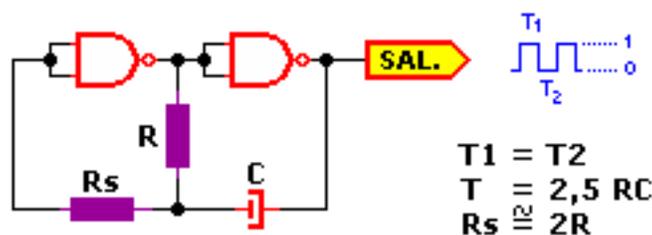
T expresado en segundos

R en Ohms

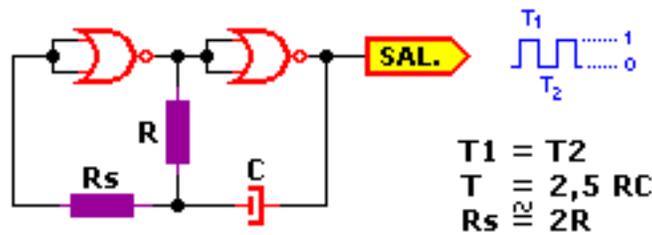
C en Faradios

Creo yo que fue fácil y sencillo hacerlo, ahora bien, si recordamos aquello de [las leyes de De Morgan](#) sabrás que uniendo las entradas de compuertas NAND o compuertas NOR obtienes la misma función que los inversores o compuertas NOT, esto me lleva a las siguientes conclusiones...

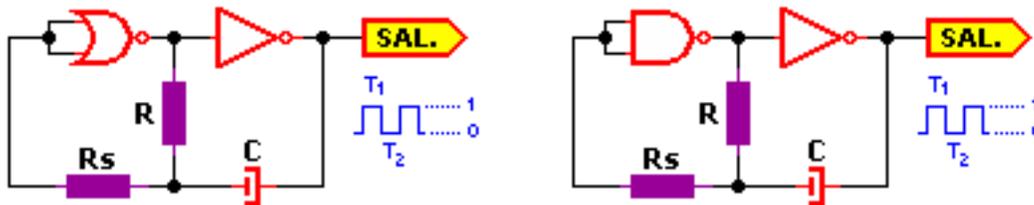
Oscilador Simétrico con compuertas NAND



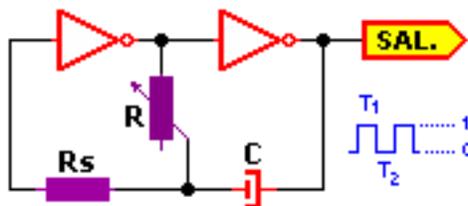
Oscilador Simétrico con compuertas NOR



Como veras, todo se basa en el primero que vimos, y hay más combinaciones todavía..., por ejemplo...



Y así... hasta que me cansé, algo que no mencioné es que puedes controlar la velocidad de estos circuitos, *Cómo...?*, Muy fácil mira...

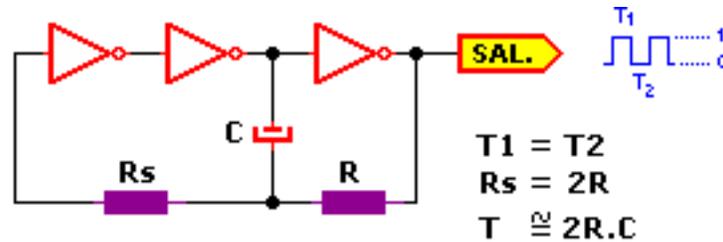


Aquí **R** es de 100k pero puedes usar otro a ver que ocurre, o cambia el capacitor, bueno, ya verás que hacer... pero sigamos con esto que aquí no termina...

:: Electrónica Digital - Lección 8

Osciladores, Multivibradores o Circuitos Astables - Parte II

Sabes que probé los osciladores anteriores con un parlante pequeño (*de esos de PC*) pero nada..., hasta que encontré una solución...

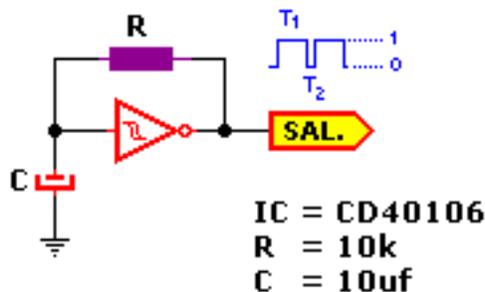


y este sí me dio resultado, hasta puedes reemplazar **R** por un potenciómetro y regular el sonido (es decir, su frecuencia) .

Disparadores Schmitt Trigger

Algo que no vimos hasta ahora son las compuertas SCHMITT TRIGGER o disparadores de Schmitt, son iguales a las compuertas vistas hasta ahora pero tienen la ventaja de tener umbrales de conmutación muy definidos llamados V_{T+} y V_{T-} , esto hace que puedan reconocer señales que en las compuertas lógicas comunes serían una indeterminación de su estado y llevarlas a estados lógicos definidos, mucho más definidos que las compuertas comunes que tienen un solo umbral de conmutación.

Se trata de esto...



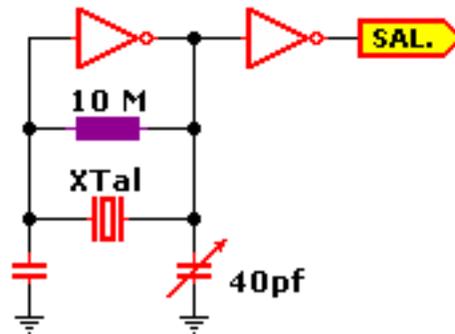
Suponte la salida a nivel lógico **1**, **C** comienza a cargarse a través de **R**, a medida que la tensión crece en la entrada de la compuerta esta alcanza el nivel V_{T+} y produce la conmutación de la compuerta llevando la salida a nivel **0** y el capacitor comienza su descarga.

Cuando el potencial a la entrada de la compuerta disminuye por debajo del umbral de V_T , se produce nuevamente la conmutación pasando la salida a nivel **1**, y se reinicia el ciclo.

No sólo existen inversores Schmitt Trigger, sino también compuertas AND, OR, NOR, etc, y ya sabes como utilizarlas, pero veamos una posibilidad más de obtener circuitos así...

Oscilador a Cristal

Se trata de un oscilador implementado con dos inversores y un Cristal de cuarzo, el trimer de 40pf se incluye para un ajuste fino de la frecuencia de oscilación, mientras el circuito oscilante en si funciona con un solo inversor, se incluye otro para actuar como etapa separadora. *extraído de un libro de por ahí... :-P*



Hasta aquí..., Te cuento que los Osciladores vistos hasta el momento pueden ser controlados fácilmente, y eso es lo que haremos ahora...

:: Electrónica Digital - Lección 9

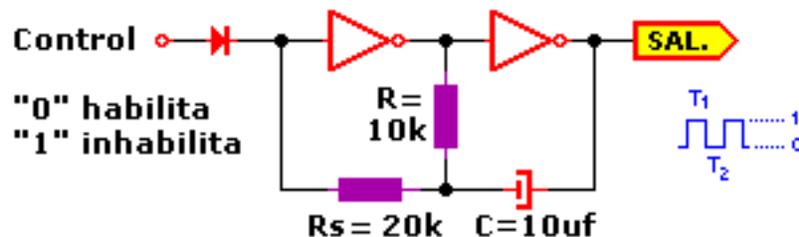
Osciladores, Multivibradores o Circuitos Astables - Parte III

Osciladores Controlados

Se trata simplemente de controlar el momento en que estos deben oscilar. Veamos..., tenemos dos opciones, que sean controlados por un nivel alto o por un nivel bajo.

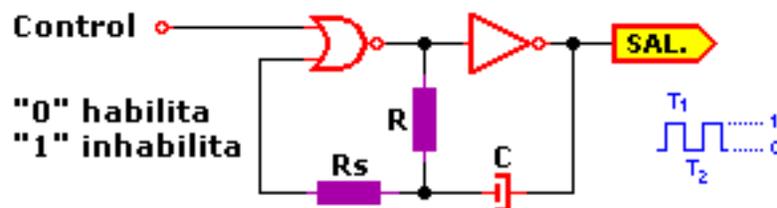
Si tienes en cuenta que los osciladores vistos hasta el momento solo pueden oscilar cambiando el estado de sus entradas en forma alternada, lo que haremos será forzar ese estado a un estado permanente, como dije anteriormente ya sea a **1** o **0**

Vamos al primer ejemplo; lo haremos utilizando un diodo en la entrada del primer inversor, así...

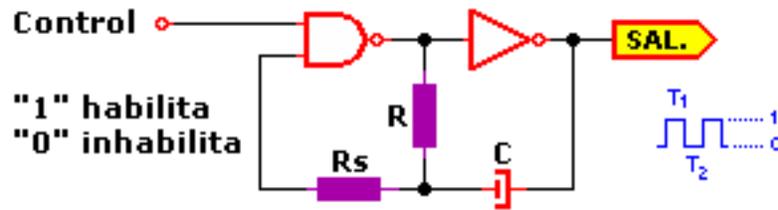


Creo que está claro, si el terminal de control está a nivel **0** el circuito oscilará, si está a nivel **1** dejará de hacerlo.

Lo mismo ocurre con las otras compuertas, observa esta con una compuerta NOR, una de sus entradas forma parte del oscilador y la otra hace de Control.



Si lo quieres hacer con compuertas NAND, es igual que el anterior, solo que esta vez un "1" en la entrada de Control habilita al oscilador y un "0" lo inhabilita.

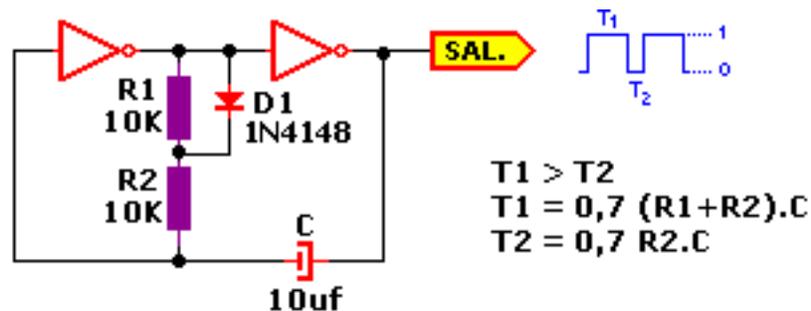


Debes estar cansado ya de tantos osciladores, pero la tentación me mata, el tema es que cierta vez quería controlar la velocidad de un motor de CC y mi única solución era disminuir la tensión lo malo es que también disminuía el torque del motor (fuerza de giro). Hasta que... un día supe que podía controlarla con un circuito astable regulando el ancho de pulso de salida, *como...?*, bueno en la siguiente lección te cuento..., hoy estoy agotado...

:: Electrónica Digital - Lección 10

Modulación por ancho de pulso

Nuevamente aquí, a ver si le damos una solución al problema planteado anteriormente, o sea, tratar de que los pulsos de salida no sean simétricos, por ejemplo que el nivel alto en la salida dure más que el nivel bajo, o quizás al revés, bueno veamos el primero.



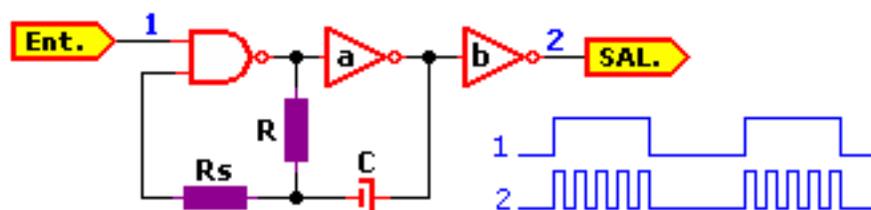
Bien, de entrada ya sabemos que es un circuito astable, solo que esta vez el capacitor se descarga más rápidamente utilizando el diodo como puente y evitando así pasar por **R1**

El efecto obtenido es que **T1** es de mayor duración que **T2**. Puedes ajustar T1 si reemplazas **R1** por un potenciómetro. Los periodos de tiempo para T1 y T2 están dados en la grafica...

Un detalle más... Si inviertes la polaridad del diodo obtendrás la situación inversa, es decir $T2 > T1$.

Modulación por ancho de pulso Conmutado

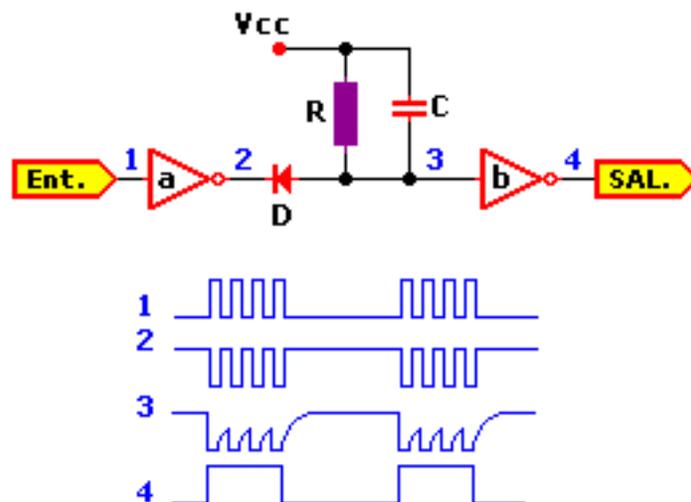
Nada raro... Los mismos circuitos vistos anteriormente pero adaptados para esta tarea. Aquí la cantidad de pulsos de salida depende de la duración del pulso de entrada. Ni para que probar, ya los conocemos y sabemos como funcionan *verdad...?*



Aquel terminal que usábamos antes como terminal de control, ahora está como entrada de señal, y la salida del circuito entregará una cierta cantidad de pulsos mientras dure el pulso de entrada. Si observas la forma de onda en la entrada y la comparas con la salida te darás cuenta de su funcionamiento.

Demodulación de señales

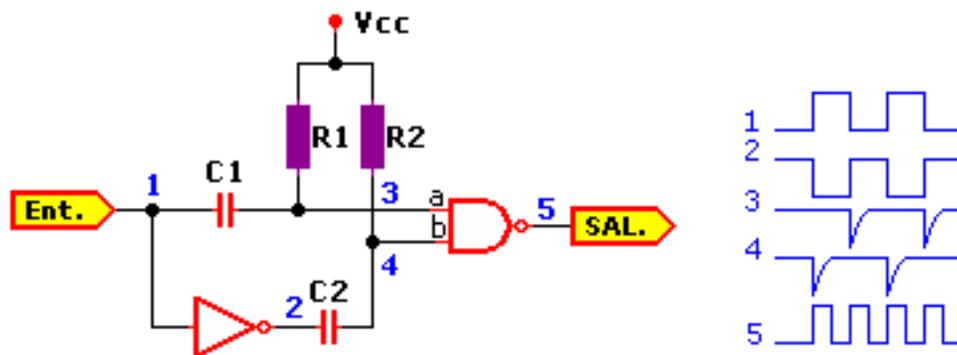
Todo lo opuesto al anterior, es decir tomamos una señal modulada y la demodulamos :o))



Esta vez el tren de pulsos ingresa por el Inversor **a**, en el primer pulso positivo, la salida de **a** se pone a **0** y se carga el capacitor **C** a través del diodo **D**. Cuando la entrada de **a** se invierte el diodo queda bloqueado y **C** se descarga a través de **R**. Ahora bien, durante toda la transmisión de pulsos la salida de **b** permanece a nivel **1** ya que el tiempo de descarga del capacitor es mucho mayor que el tiempo de duración de cada pulso que ingresa por la entrada del inversor **a**

Doblador de frecuencia

Otra aplicación que se pueden dar a las compuertas lógicas es duplicar la frecuencia de una señal, como en este circuito.



Observa la forma de onda obtenidas en los puntos marcados en azul

Analicemos su funcionamiento; El flanco de descenso de la señal de entrada es diferenciada por R1 y C1, y es aplicada a la entrada "a" de la compuerta NAND, esto produce un pulso a la salida de esta compuerta según su tabla de verdad "basta que una de las entradas este a nivel lógico bajo para que la salida vaya a nivel lógico alto"

El flanco de subida del pulso de entrada, luego de ser invertido, es diferenciado y aplicado a la entrada "b" de la compuerta NAND, de modo que para un tren de pulsos de entrada de frecuencia f , hay un tren de pulsos de salida de frecuencia $2f$.

Basta de circuitos astables, veamos como hacer un monoestable...

:: Electrónica Digital - Lección 11

Circuitos Monoestables

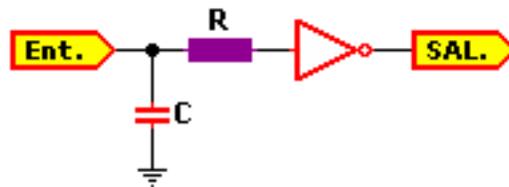
Habrás que recordar de que se trata esto no...?

De acuerdo..., son aquellos que tienen un único nivel de salida estable. Para aclarar un poco las ideas... La mayoría de los edificios disponen de un pulsador que enciende momentáneamente las luces de los pasillos, transcurrido un cierto tiempo éstas se apagan. Conclusión; sólo disponen de un estado estable (apagado) y un estado metaestable (encendido). *se entendió...?*

Bien, veamos el primero:

Monoestables sencillo

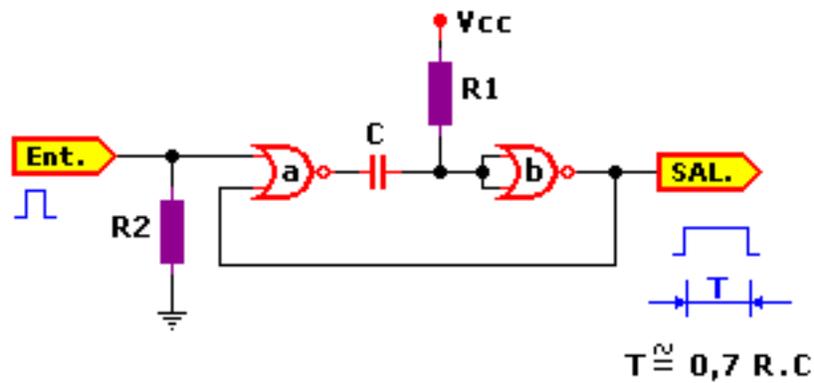
Primero lo básico, un monoestable sencillo con un inversor...



Considera inicialmente la entrada del inversor en nivel bajo a través de **R** y **C**, entonces su salida estará a nivel alto, ahora bien, un **1** lógico de poca duración en la entrada, hace que se cargue el capacitor y conmute el inversor entregando un **0** lógico en su salida, y este permanecerá en ese estado hasta que la descarga del capacitor alcance el umbral de histéresis de la compuerta y entonces conmutará y regresará a su estado inicial...

Monoestables con dos compuertas NOR

Fíjate que la compuerta **b** la puedes cambiar por un inversor...Tratemos ahora de interpretar su funcionamiento



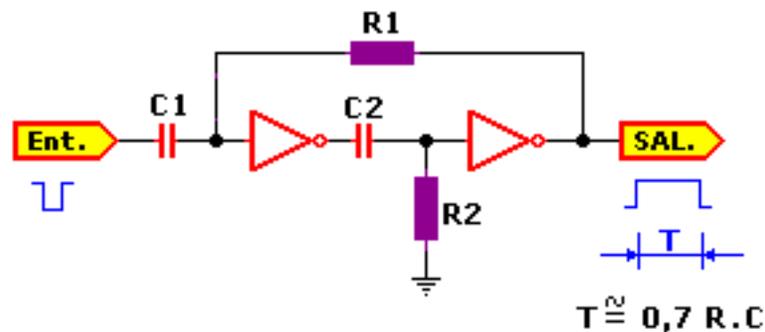
Suponte que no existe señal en la entrada, entonces la compuerta **b** tiene su entrada a nivel "1" por intermedio de **R1**, y su salida a nivel "0", la cual alimenta una de las entradas de **a**. al estar ambas entradas de "a" a nivel "0" la salida de "a" estará a nivel "1". Como el capacitor **C** tiene sus dos extremos al mismo nivel no adquiere carga alguna.

Si entregas un impulso positivo a la entrada de **a**, su salida pasa inmediatamente a nivel "0" y **C** comienza a cargarse a través de **R1**, la entrada de **b** se hace **0** y su salida **1**, como ésta realimenta la compuerta **a** la deja *enganchada* con su salida a **0**.

Cuando la carga del capacitor alcanza el umbral de conmutación de "b" su salida pasa a **0** y la de **a** pasa a **1**, esto hace que el capacitor se descargue a través de **R1** y la línea de alimentación, dejando al circuito listo para un nuevo disparo.

Monoestables con dos inversores

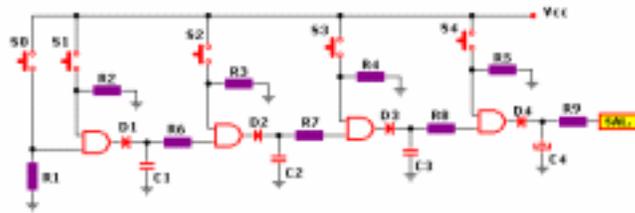
La diferencia aquí, esta en que el gatillado se realiza durante la excursión negativa del pulso de entrada.



Como verás, estos circuitos disponen de algún método de realimentación y un capacitor que es quien retiene momentáneamente una determinada señal lógica en la entrada de alguna de las compuertas implicadas en el circuito...

Cerradura con teclado electrónico

Lo mejor que se me pudo ocurrir para la aplicación de un monoestable fue una cerradura electrónica sencilla, aquí la puedes ver...



La secuencia para activar la salida es el orden en que están numerados los pulsadores, eso sí, nota que debes pulsar S0 y sin liberarlo activar S1, luego de eso puedes continuar con la secuencia correspondiente.

Los botones libres del teclado deberían ir unidos a un sistema de alarma o hacer que se desactive momentáneamente todo el sistema antes de ingresar un nuevo código, en fin tienes muchas opciones. En la salida deberás conectar un relé, un optoacoplador o algo por el estilo para accionar la cerradura electrónica.

En realidad la intención era darle una utilidad a los circuitos monoestables, y esta me pareció ideal

Los componentes utilizados son los siguientes:

D1 a D4 = 1N4148

R1 a R6 = 1k

R7 a R9 = 2k2

C1 a C3 = 1nf

C4 = 1000 uf / 16V

IC1 = CD4081

S1 a S5 = Teclado

Que lo disfrutes y hasta la próxima...!!!

:: Electrónica Digital - Lección 12

Circuitos Biestables - Parte I

Nuevamente aquí... y luego de esto me tomaré un buen descanso, *eso creo...!!!*

Comencemos... Los circuitos biestables son muy conocidos y empleados como elementos de memoria, ya que son capaces de almacenar un bit de información. En general, son conocidos como Flip-Flop y poseen dos estados estables, uno a nivel alto (1 lógico) y otro a nivel bajo (cero lógico), *Se entendió...?, aplausos para mí... gracias, muchas gracias...!!!*

Comencemos... Los circuitos biestables son muy conocidos y empleados como elementos de memoria, ya que son capaces de almacenar un bit de información. En general, son conocidos como Flip-Flop y poseen dos estados estables, uno a nivel alto (1 lógico) y otro a nivel bajo (cero lógico), *Se entendió...?, aplausos para mí... gracias, muchas gracias...!!!*

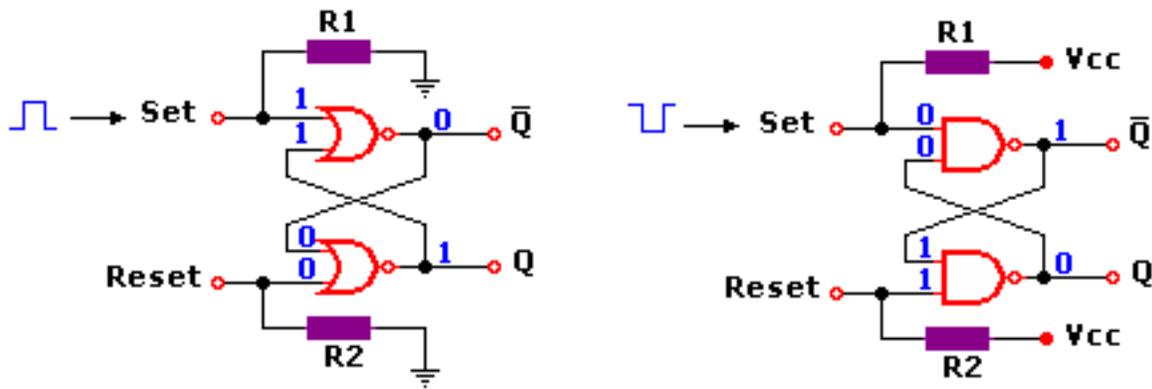
Perdón, me estaba olvidando de un pequeño detalle, es posible que al presionar el pulsador se produzcan rebotes eléctricos, es como haberlo presionado varias veces, y sí... los resultados serán totalmente inesperados, así que lo de los cablecitos para probar estos circuitos no nos servirán de mucho, es conveniente utilizar un pulso de reloj para realizar estas pruebas, *ya sabes...!!! un circuito astable, de los que hicimos en lecciones anteriores, de ahora en más lo llamaremos **pulso de reloj** o **Clock** o **CK**.*

Por lo general un Flip-Flop dispone de dos señales de salida, una con el mismo valor de la entrada y otra con la negación del mismo o sea su complemento.

Primero lo básico, como siempre, y luego lo enredamos un poco más.

FLIP FLOP BÁSICO RS

Se puede construir uno fácilmente utilizando dos compuertas NAND o NOR conectadas de tal forma de realimentar la entrada de una con la salida de la otra, quedando libre una entrada de cada compuerta, las cuales serán utilizadas para control **Set** y **Reset**...



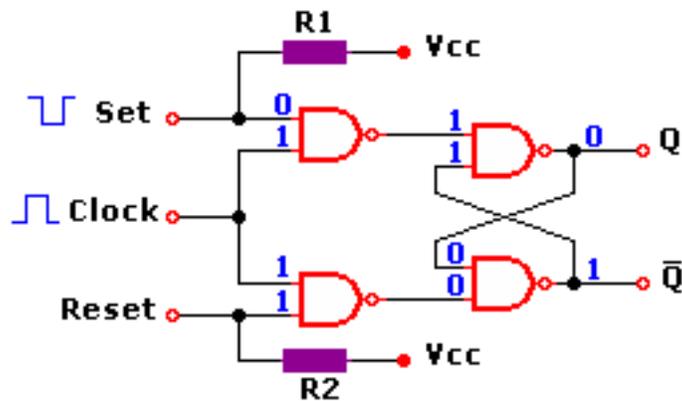
Las resistencias R1 y R2 utilizadas en ambos casos son de 10k y las puse solamente para evitar estados indeterminados, observa el circuito con compuertas NOR... Un nivel alto aplicado en **Set**, hace que la salida negada $\sim Q$ sea **0** debido a la tabla de verdad de la compuerta NOR, al realimentar la entrada de la segunda compuerta y estando la otra a masa, la salida normal **Q** será **1**. Ahora bien, esta señal realimenta la primer compuerta, por lo tanto no importan los rebotes, y el FF se mantendrá en este estado hasta que le des un pulso positivo a la entrada **Reset**

Conclusión: El biestable posee dos entradas **Set** y **Reset** que trabajan con un mismo nivel de señal, provee dos salidas, una salida normal **Q** que refleja la señal de entrada Set y otra $\sim Q$ que es el complemento de la anterior.

Si comparas los dos flip-flop representados en el gráfico, verás que sólo difieren en los niveles de señal que se utilizan, debido a la tabla de verdad que le corresponde a cada tipo de compuerta.

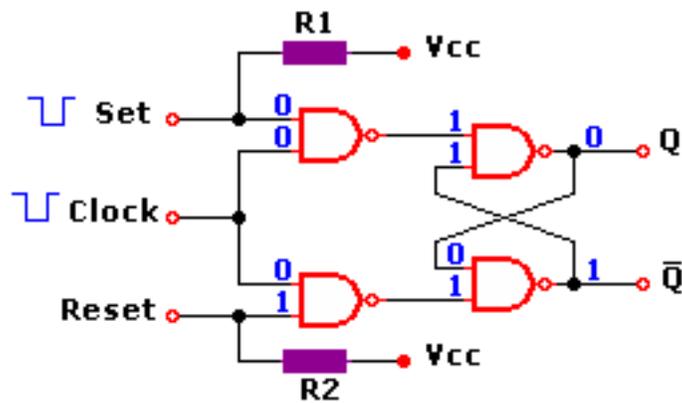
FLIP FLOP RS - Controlado por un pulso de reloj:

En este caso voy a utilizar el ejemplo de las compuertas NAND, pero le agregaremos dos compuertas mas, y uniremos la entrada de cada una a una señal de Reloj...



Lo dicho mas arriba, necesitamos un generador de pulsos (Astable) para conectarlo en la entrada Clock, una vez lo tenemos pasamos a interpretar el circuito...

Si pones un **0** en **Set** y la entrada **Clock** está a **1** ocurrirá todo lo que se describe en el esquema anterior, veamos que ocurre cuando **Clock** pasa a **0**...



Sorpresaaaaaaaaa...!!!, el FF se mantiene sin cambios en **Q** y $\sim\mathbf{Q}$. Fíjate que ahora no importa el estado de **Set** y **Reset**, esto se debe a su tabla de verdad (basta que una de sus entradas sea **0** para que su salida sea **1**) por lo tanto **Set** y **Reset** quedan inhabilitadas.

Es decir que se leerán los niveles de **Set** y **Reset** sólo cuando la entrada **Clock** sea **1**.

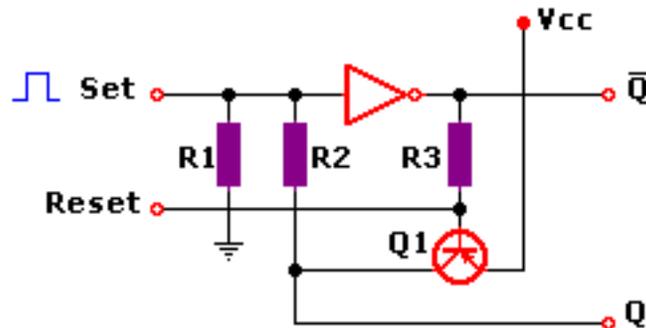
NOTA 1: El primer circuito que vimos (Flip-Flop simple) es llamado **Flip-Flop Asíncrono** ya que puede cambiar el estados de sus salidas en cualquier momento, y sólo depende de las entradas Set y Reset.

NOTA 2: El segundo circuito es controlado por una entrada Clock y es llamado **Flip-Flop Síncrono** ya que el cambio de estado de sus salidas esta sincronizado por un pulso de reloj que realiza la lectura de las entradas en un determinado instante.

Antes de continuar quiero mostrarte algo muy interesante, no es la única forma de obtener un Flip-Flop, observa esto...

FLIP FLOP - Con un inversor

La ventaja aquí es la cantidad de compuertas utilizadas, esta bueno, no te parece...?



Bueno, lo dejo para que lo analices...

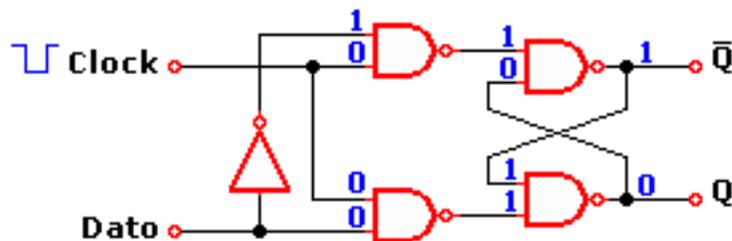
:: Electrónica Digital - Lección 13

Circuitos Biestables - Parte II

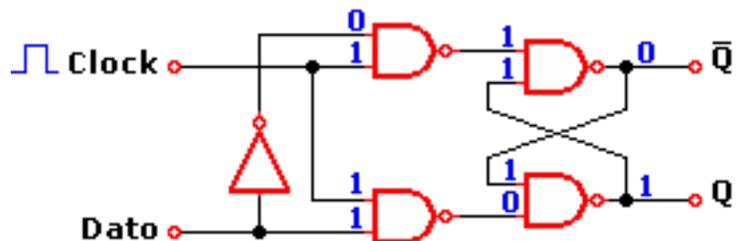
El flip-flop presentado anteriormente conocido como **flip-flop RS** suele presentar un estado indeterminado cuando sus dos entradas R y S se encuentran en estado alto así que veamos como se puede solucionar este inconveniente...

FLIP FLOP D:

En este circuito no existe la posibilidad de que las dos entradas estén a nivel alto ya que posee un inversor entre la una y la otra de tal modo que $R = \sim S$, observa el siguiente gráfico, aquí se supone la entrada Dato a nivel 0...

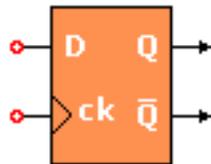


Veamos que ocurre cuando la entrada Dato, pasa a 1 y CK cambia de estado pasando también a 1, según como se van transmitiendo los datos por las compuertas resulta $Q=1$ y $\sim Q=0$.



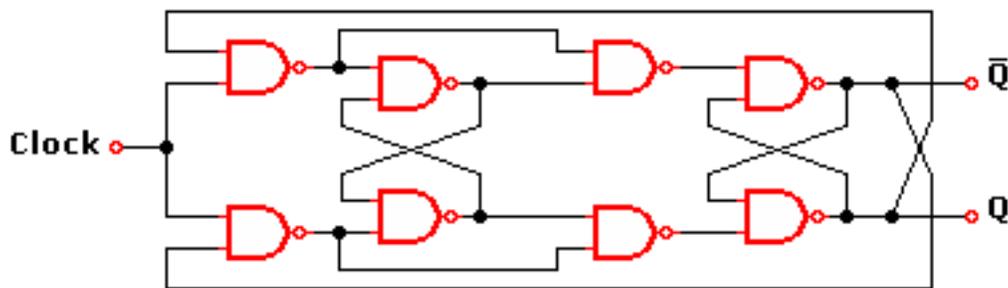
Para que el flip-flop retorne a su estado inicial, la entrada Dato **D** deberá pasar a **0** y sólo se transferirá a la salida si Ck es **1**. Nuevamente se repite el caso que para leer el datos debe ser $ck=1$.

En forma general se representa el flip-flop D con el siguiente símbolo



FLIP FLOP Master-Slave:

Se trata de un arreglo de dos FF independientes. El primero actúa como Master y el otro como Slave. Con la diferencia de que en este caso las entradas Set y Reset son realimentadas por las salidas Q y $\sim Q$ respectivamente, quedando libre únicamente la entrada CK.



Ya se, será complicado de analizar, pero lo haremos fácil, veamos...

Considerando $CK=0$, será la salida $Q=0$ y $\sim Q=1$, al momento del cambio de nivel de CK ($CK=1$), sólo cambiarán las salidas del primer flip-flop (Master) sin afectar las salidas Q y $\sim Q$.

Ahora bien, cuando CK regrese a su estado inicial ($CK=0$) el Slave conmutará las salidas Q y $\sim Q$ quedando $Q=1$ y $\sim Q=0$. Al cambiar de estado CK ($CK=1$) las salidas no serán afectadas. Esto se puede resumir en una pequeña tabla de verdad, como ésta...

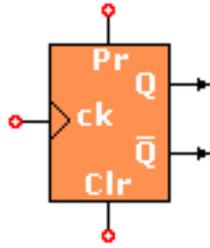
| CK | Q | $\sim Q$ |
|----|---|----------|
| 0 | 0 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 1 | 0 |
| 0 | 0 | 1 |

Bueno, le agregué una fila más, por si preguntas ;-)

A este tipo de Flip-flop, se le a dado la posibilidad de preestablecer el estado de sus salidas, adicionándole dos entradas más, Preset (**Pr**) y Clear (**Clr**), que vendrían a ser algo así como Set y Reset respectivamente, pero

claro, hay que advertir que se debe evitar la situación $Pr=Clr=0$

También tiene una forma de representación simbólica...



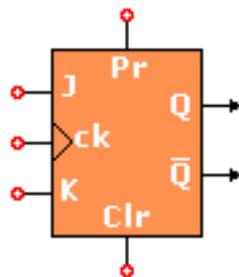
Y aún queda más, el flip-flop JK, así que vamos por él...

:: Electrónica Digital - Lección 14

Circuitos Biestables - Parte III

FLIP FLOP JK:

Un flip-flop JK es muy similar al visto anteriormente pero mucho más complejo que éste, y existen Circuitos integrados que ya lo traen incorporado así que por cuestiones de sencillez y para no complicarte demasiado utilizaré su representación simbólica.



Lo vamos a analizar de forma sencilla haciendo uso de la tabla de verdad que corresponde al funcionamiento del flip-flop...

| Clr | Pr | J | K | CK | Q | $\sim Q$ |
|-----|----|---|---|----|----|----------|
| 0 | 1 | X | X | X | 0 | 1 |
| 1 | 0 | X | X | X | 1 | 0 |
| 1 | 1 | X | X | 0 | Sc | Sc |
| 1 | 1 | X | X | 1 | Sc | Sc |
| 1 | 1 | X | X | ↑ | Sc | Sc |
| 1 | 1 | 0 | 0 | X | Sc | Sc |
| 1 | 1 | 0 | 1 | ↓ | 0 | 1 |
| 1 | 1 | 1 | 0 | ↓ | 1 | 0 |
| 1 | 1 | 1 | 1 | ↓ | Cs | Cs |

Comencemos:

Las dos primeras líneas indican que las entradas Clr y Pr establecen el estado de las salidas Q y $\sim Q$ sin importar el estado en que se encontraban anteriormente, ni el estado de las otras entradas (J, K y CK).

En la tercera y cuarta línea se han establecido las entradas Clr y Pr a nivel 1

y las salidas Q y $\sim Q$ permanecen en cualquiera de los dos estados mencionados anteriormente, según el que se haya establecido...!!!, ahora bien si se mantiene $CK=0$ las salidas Q y $\sim Q$ permanecen sin cambio (Sc), lo mismo ocurre si se mantiene $CK=1$, y continúa así en los cambios ascendentes de CK , y como podrás notar en la siguiente línea, si estableces $J=K=0$ queda sin importancia la entrada CK y sin cambio las salidas.

En la séptima y octava línea se transfieren los datos de las entradas J y K a las salidas Q y $\sim Q$ respectivamente, pero esto sólo ocurrirá en la transición ascendente de CK .

Finalmente con $Clr=Pr=J=K=1$ el flip-flop Cambiará Siempre (Cs) cada vez que se presente una transición descendente de CK .

Y hasta aquí..., la idea fue mostrarte las ventajas y desventajas de cada uno de estos circuitos, te recuerdo que no necesitas armar uno de estos embrollos de compuertas, ya que existen integrados que las contienen, como el CD4027 que es un doble flip-flop JK maestro-esclavo o el CD4013 que es un doble flip-flop tipo D, al cual le voy a dedicar una página especial, por sus variadas aplicaciones, muy utilizado en robótica...!!!

Es más estoy pensando que sería bueno describir las funciones de algunos integrados, los más utilizados según el tipo de aplicación y diseño que se requiera, ya veré si me doy tiempo para ello, pero creo que me merezco un descanso, así que..., ahí nos vemos.

Hasta la próxima...!!!

