
**Ing. Guadalupe Evaristo Cedillo Garza †**

**26 octubre 1934, 1 Diciembre 2023 †  89 años,**
**66 años de profesor en FIME**

**La verdadera experiencia no se limita a lo vivido, sino que radica en la reflexión sobre lo vivido**

*JAGG*

| Clase | Actividad | Laboratorio |
|---|---|---|
| PF7 | Diseño Combinacional con HDL | Sesión 6 |
| AF3 | Decodificador con Display | Sesión 7 |
| PF9 | Flip Flops | Sesión 8 |
| PF10 | Pulsos de sincronía | Sesión 9 |
| AF4 | Diseño Secuencial | Sesión 10 |
| | | Sesión 11 |
| AF5 | PIA | PIA |

**Lo que se solicitará en últimas sesiones de laboratorio corresponde a:**

**PF6 a la *sesión 6*, AF3 a la *sesión 7*, PF8 a las *sesiones 8 y 9*,**
**AF4 a las *sesiones 10 y 11*,**
**El PIA asignado en clase también vale para el laboratorio**

**Las evidencias de aprendizaje y documentos entregables son los mismos, solo hay que cambiar los datos de la portada**

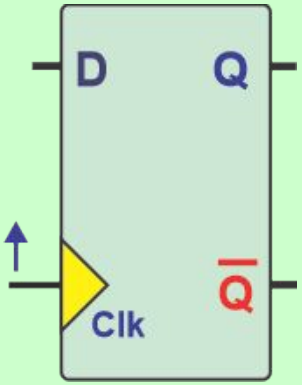
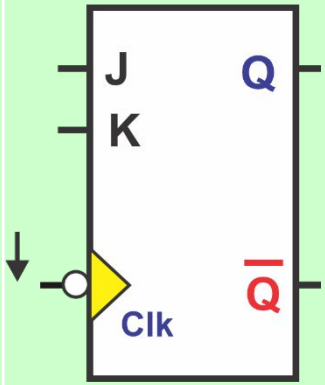
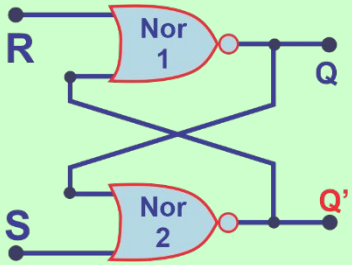
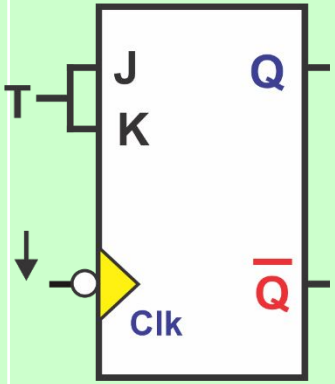
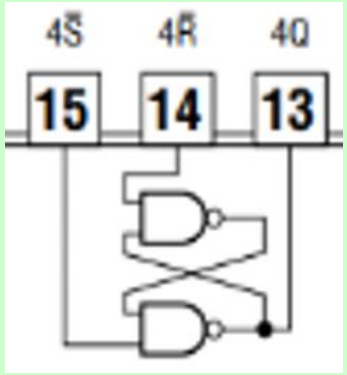
*JAGG*

# Tipos formales de Flip Flops

| Tipo | Nombre | Serie TTL SN 74 | |
|---|---|---|---|
| **D** | **Data o Datos** | **74**, 564, 575, 576, 577, 874, 876 | **Flip Flops** Comerciales (estándares) |
| **JK** | **Jack Kilby** | **73**, 107, **109**, 112, 276 | |
| **SC** | **Set Clear** | **279** también llamado **SR** | |
| **RS** | **Reset Set** | **Flip Flops** basados en compuertas básicas retroalimentadas o partiendo de FF's comerciales | |
| **T** | **Toggle** | | |
| **SC** | **Set Clear** | | |

**El PLD22V10 tiene 10 FFs tipo D**

JAGG

| FF D | FF JK | FF RS | FF T | FF SC |
|---|---|---|---|---|
|  |  |  |  |  |

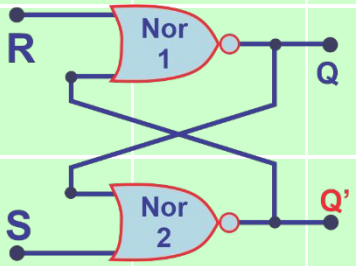


JAGG

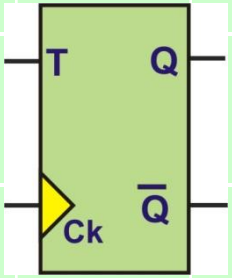
# Tablas características

## Reset - Set

| Entradas de control | | Salida |
|---|---|---|
| R | S | Qn+1 |
| 0 | 0 | Qn |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | X |



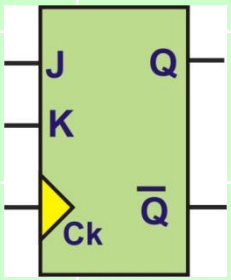

## Toggle



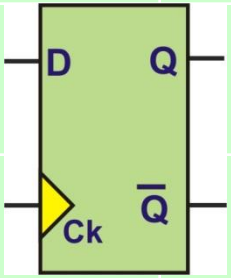

| T | Qn+1 |
|---|---|
| 0 | Qn |
| 1 | ‘Qn |

## J - K

| Entradas de control | | Salida |
|---|---|---|
| J | K | Qn+1 |
| 0 | 0 | Qn |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | Q’ |




## Data

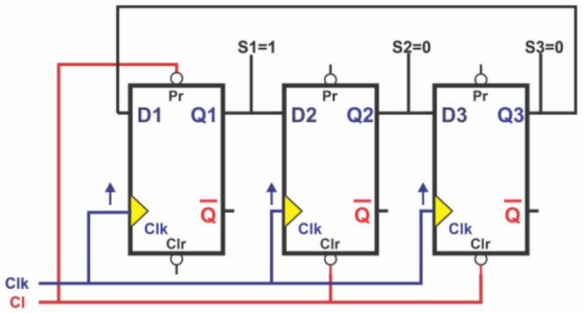


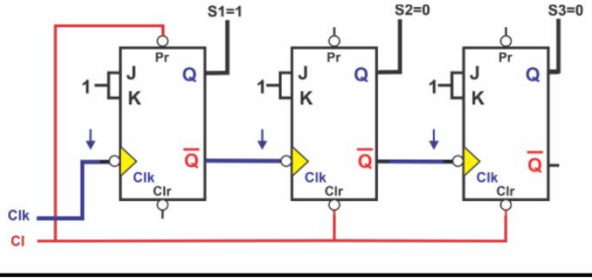

| D | Qn+1 |
|---|---|
| 0 | 0 |
| 1 | 1 |

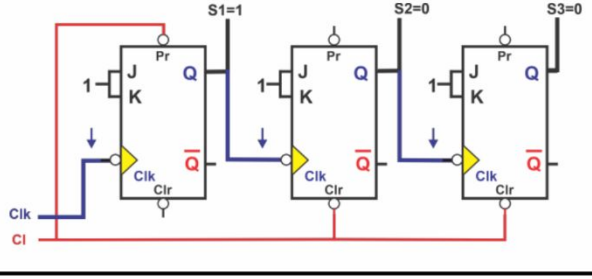
JAGG

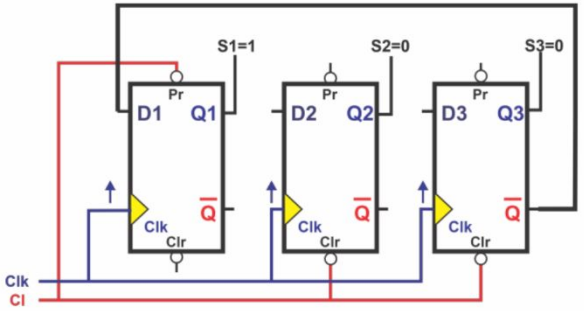
# Proyecto Formativo 9 Sesión 8 Lab

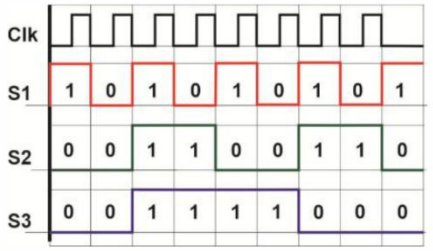
| Circuito | Grafica de tiempos |
|---|---|
| 1 | A |
| 2 | B |
| 3 | C |
| 4 | D |

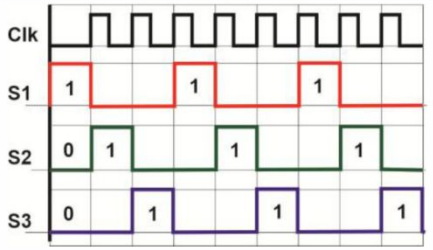


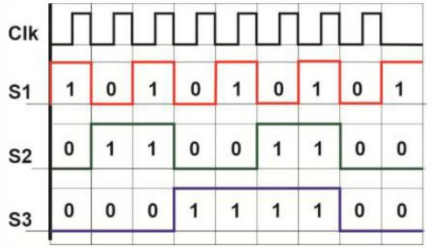



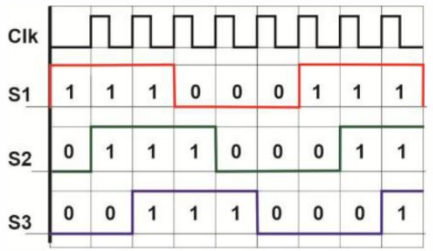



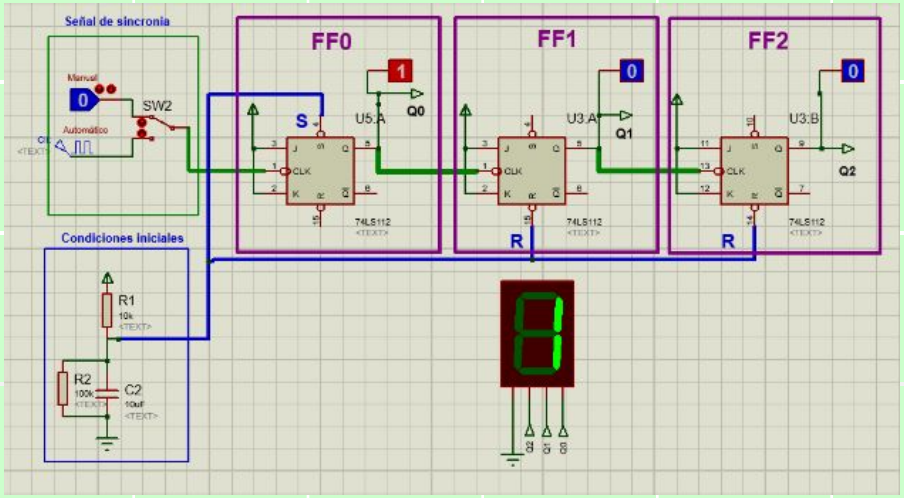



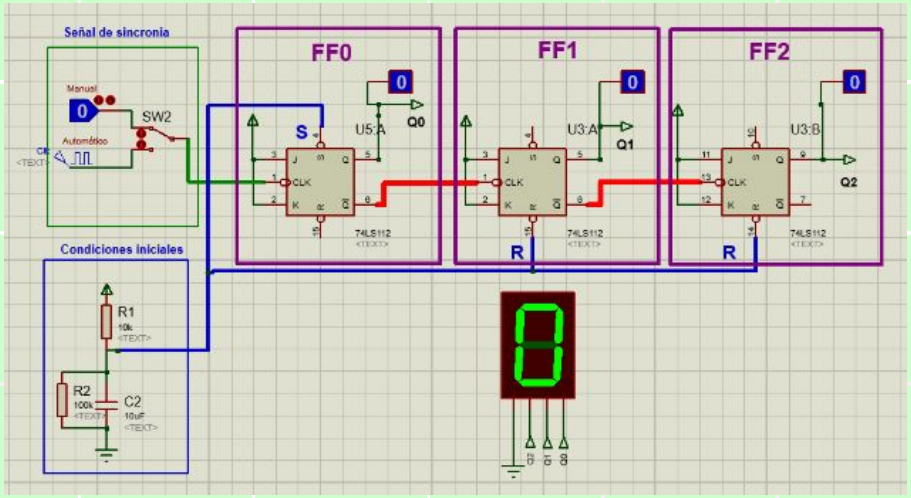



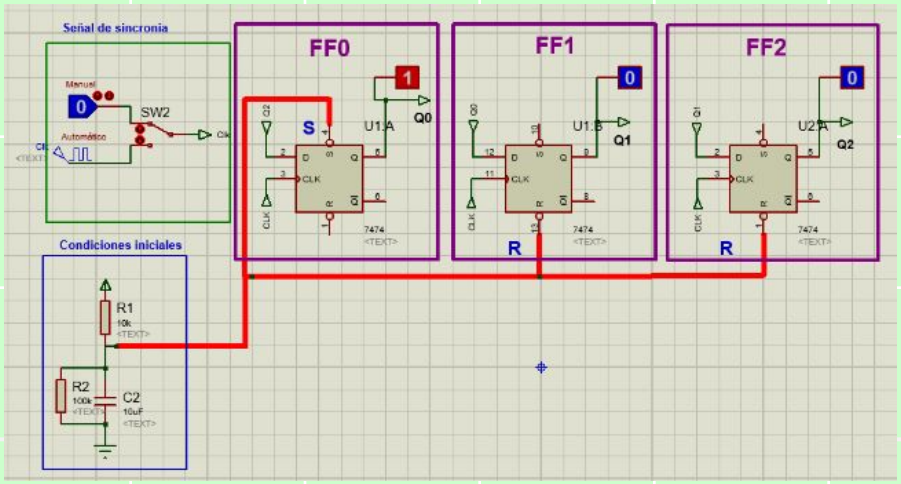



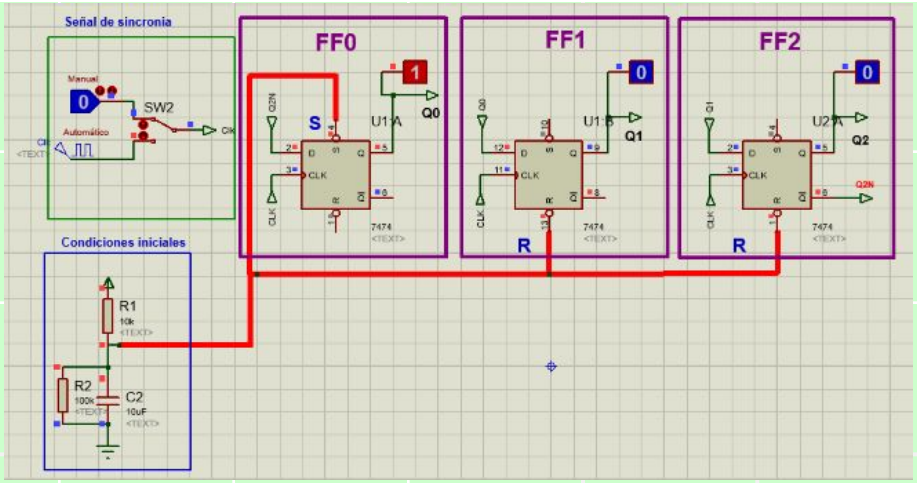



# Sesión 8 lab y PF 9



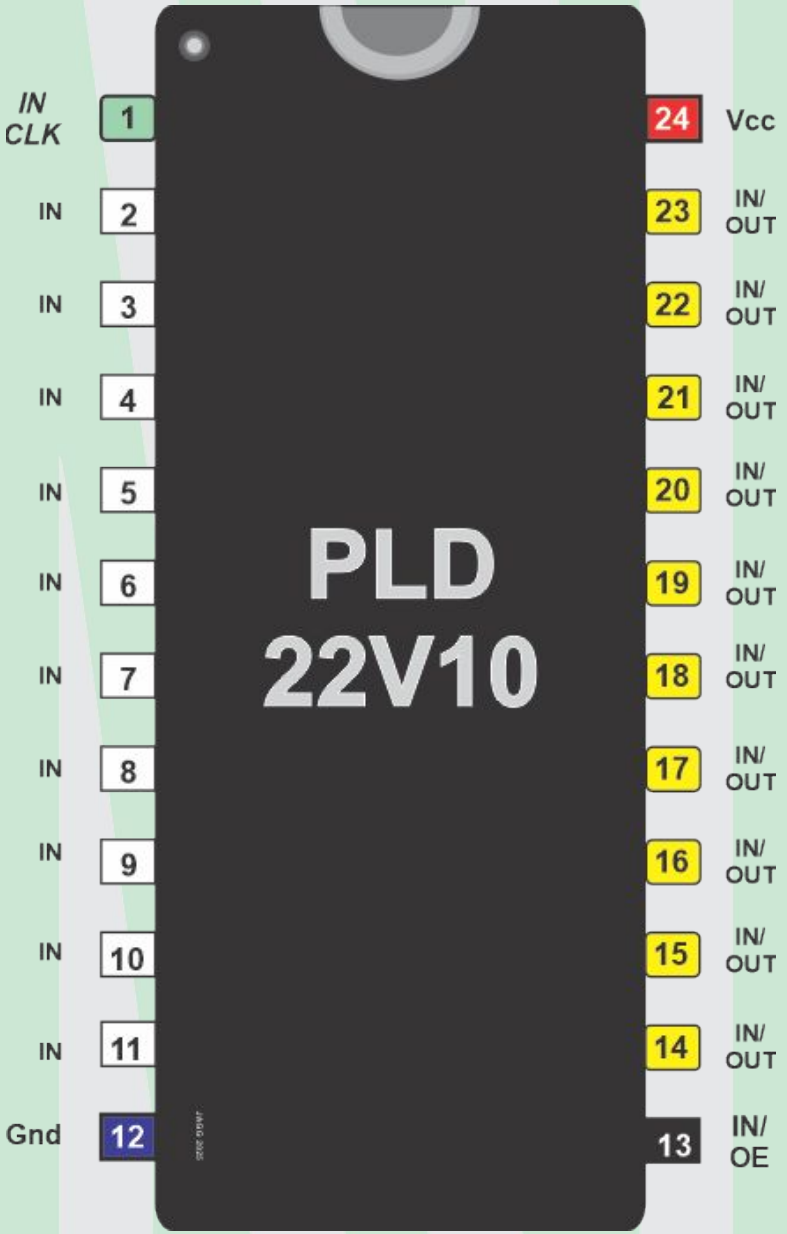




*JAGG*

# Tablas de excitación

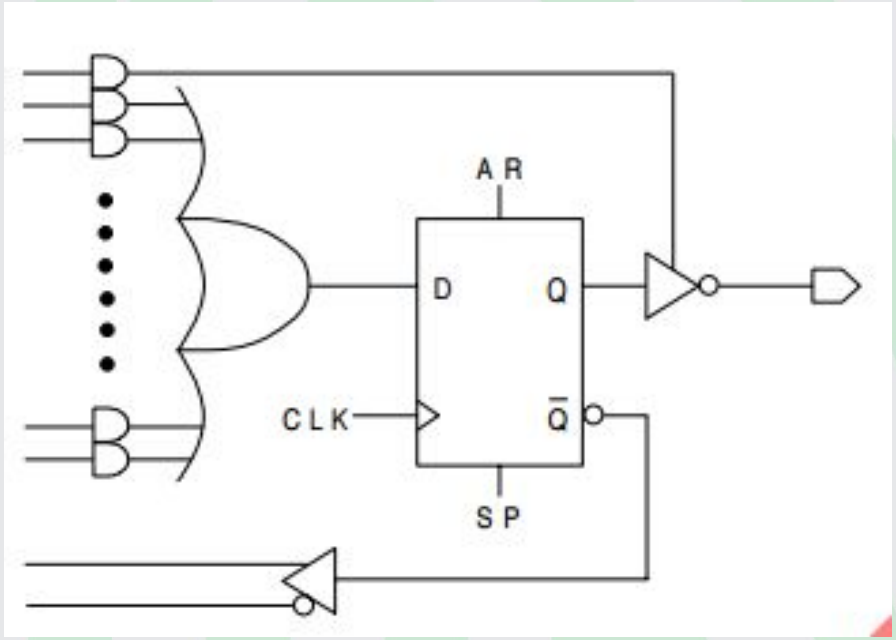
| Estado Presente | Estado Siguiente | Entradas de control | | | | | |
|---|---|---|---|---|---|---|---|
| Qn | Qn+1 | R | S | J | K | T | D |
| 0 | 0 | X | 0 | 0 | X | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | X | 1 | 1 |
| 1 | 0 | 1 | 0 | X | 1 | 1 | 0 |
| 1 | 1 | 0 | X | X | 0 | 0 | 1 |

JAGG




**El PLD22V10 tiene 10 FFs tipo D**

**De las salidas 14 a la 23**

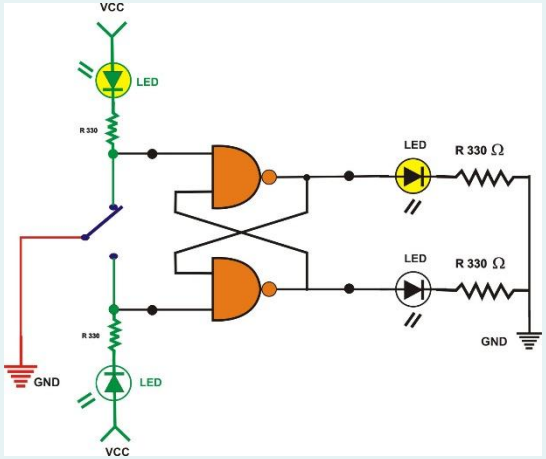
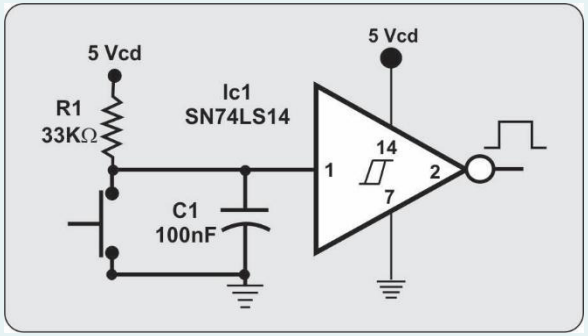
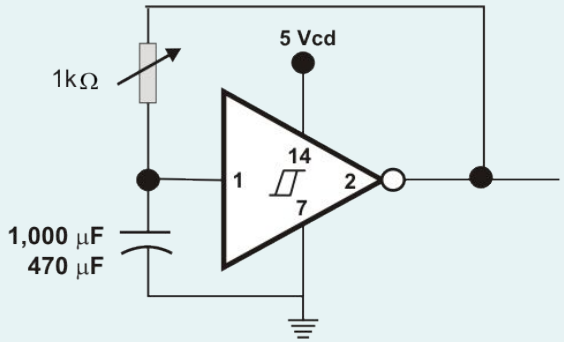



# Istype ‘reg’

JAGG

## Sesión 9 laboratorio

# Generación de Pulsos de Sincronía para Sistemas Secuenciales

| Biestable | Monoestable | Astable |
|---|---|---|
|  |  |  |
| Mantienen dos estados estables | Generan un solo pulso ante una excitación | Producen oscilaciones continuas sin estabilidad |







*JAGG*

# Multivibradores, Generadores de pulsos de sincronía

- **Biestable** (Dos estados estables, Flip-Flops )
  - RS, JK, T, D y **SC** (Eliminador de rebotes)
- **Monoestable** (Un estado estable y otro transitorio)
  - Redisparables (7414, 74121 )
  - No redisparables (555)

    Transición Positiva ↑ y Transición Negativa ↓

**Manual**

- **Astable** (Sin estabilidad)
  - Timer (555 y 7414)

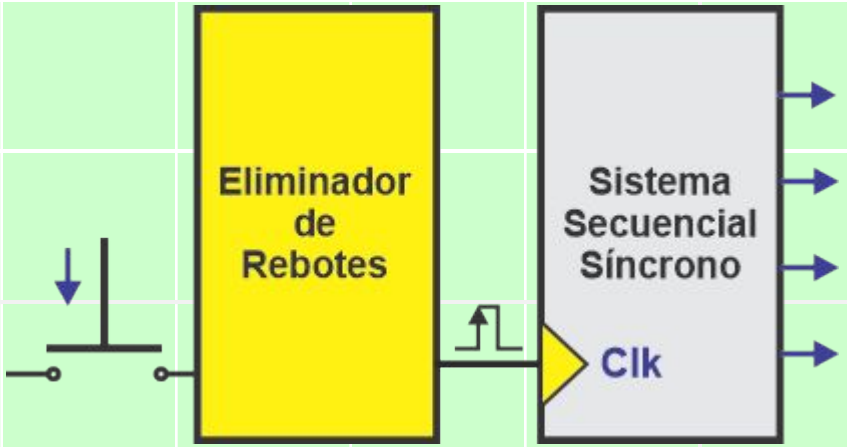
**Automático**
**Frecuencia Variable**

*JAGG*

# Generación de los pulsos de sincronía para los Flip Flops

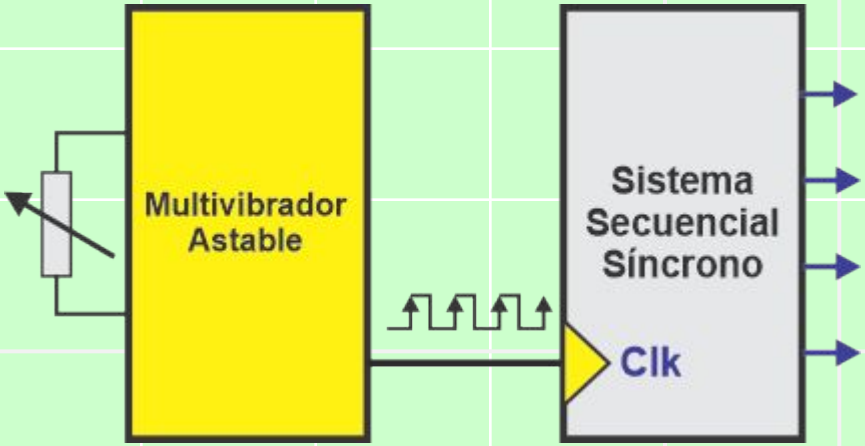
**1. Manual**

**Biestable y Monoestable**




**2.-Automática**

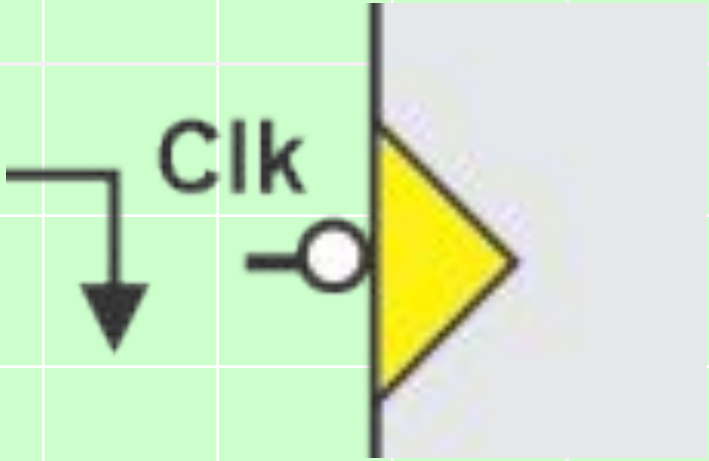
**Astable,** Frecuencia variable




*JAGG*

# Generadores de pulsos de sincronía

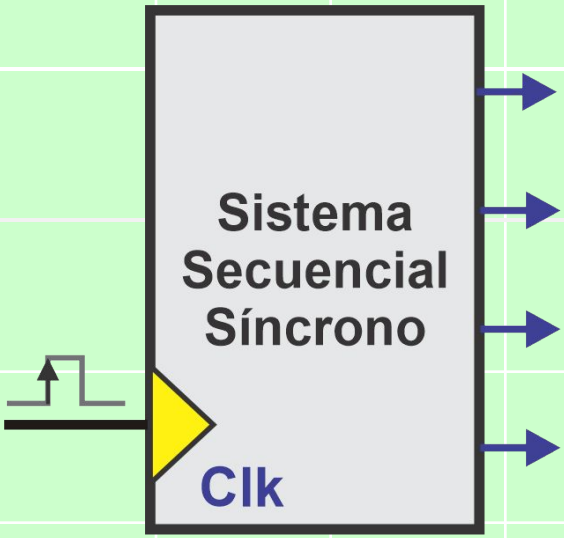

Transición Positiva



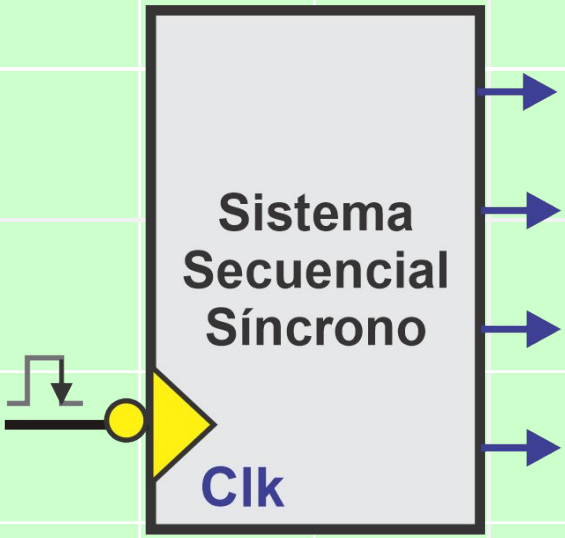

Transición Negativa

JAGG

# Generación de los pulsos de sincronía para los sistemas secuenciales síncronos



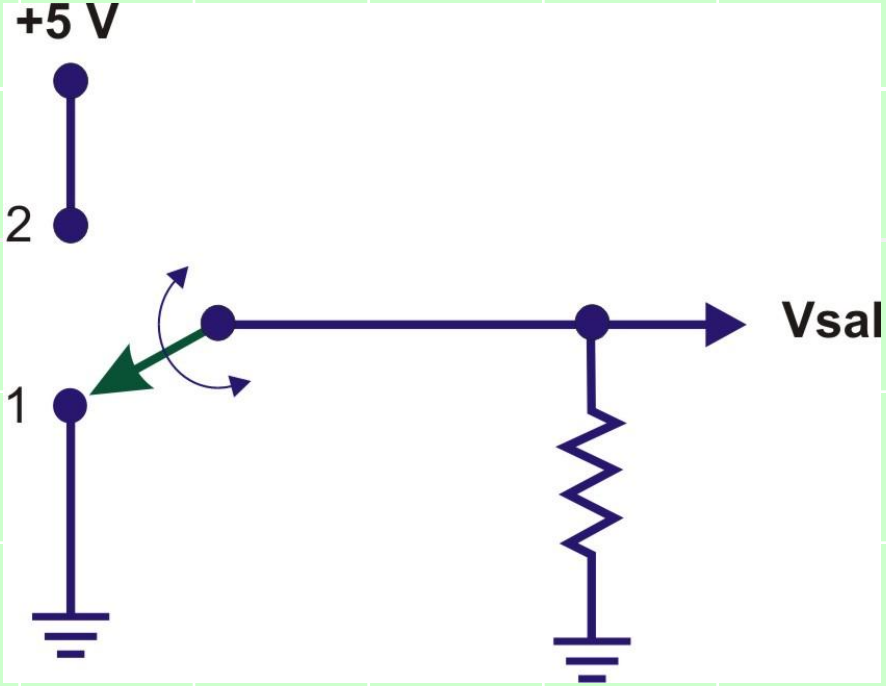

**Transición Positiva**

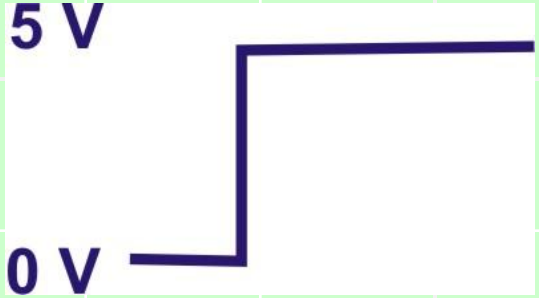



**Transición Negativa**

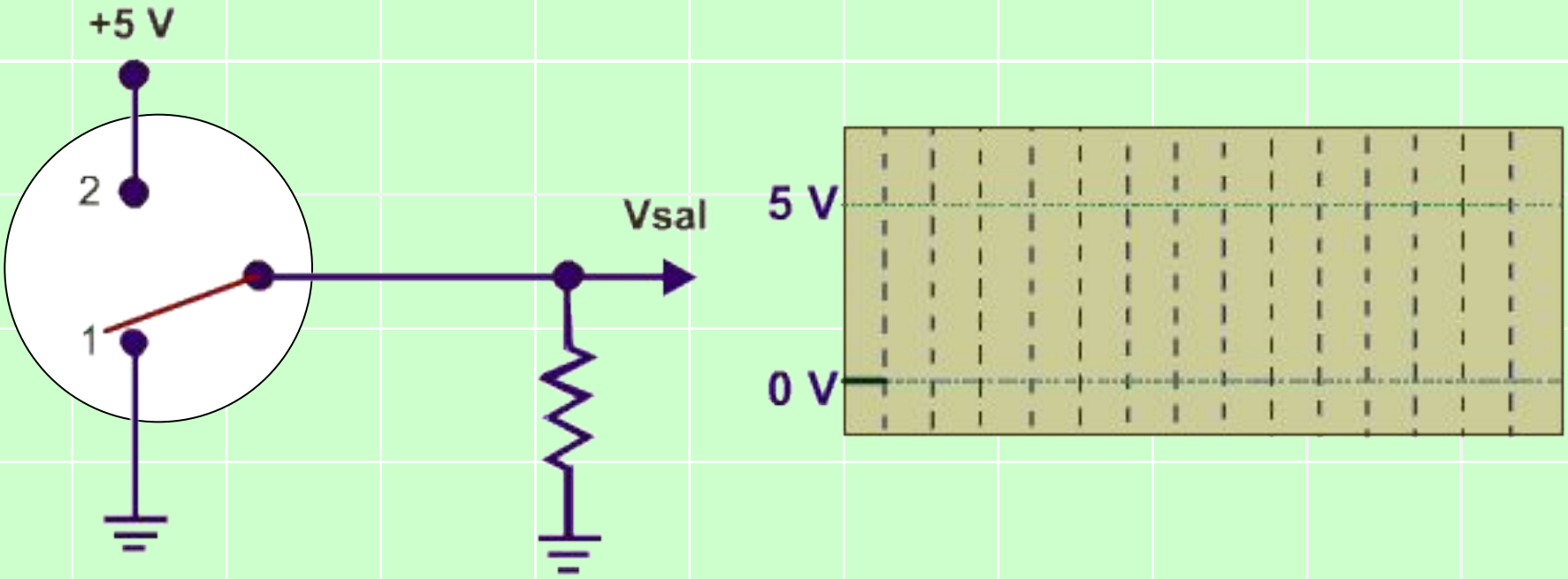
*JAGG*

# Eliminador de rebote

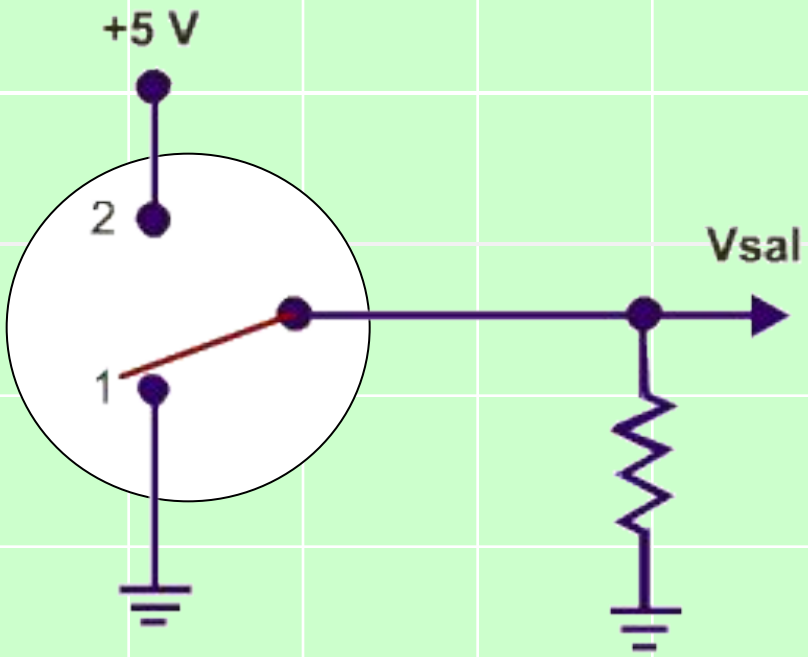


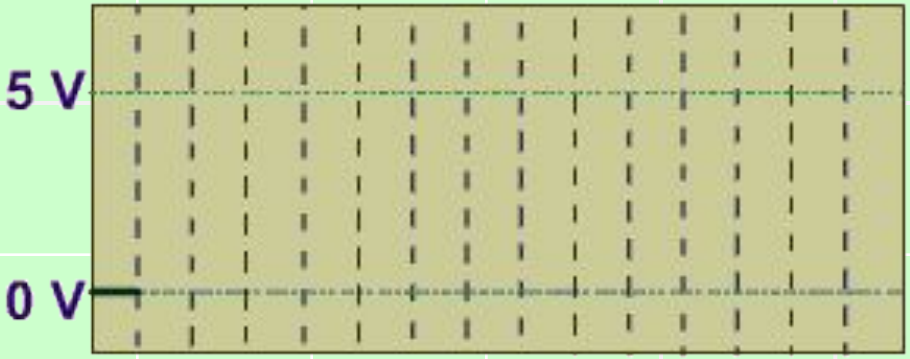




JAGG

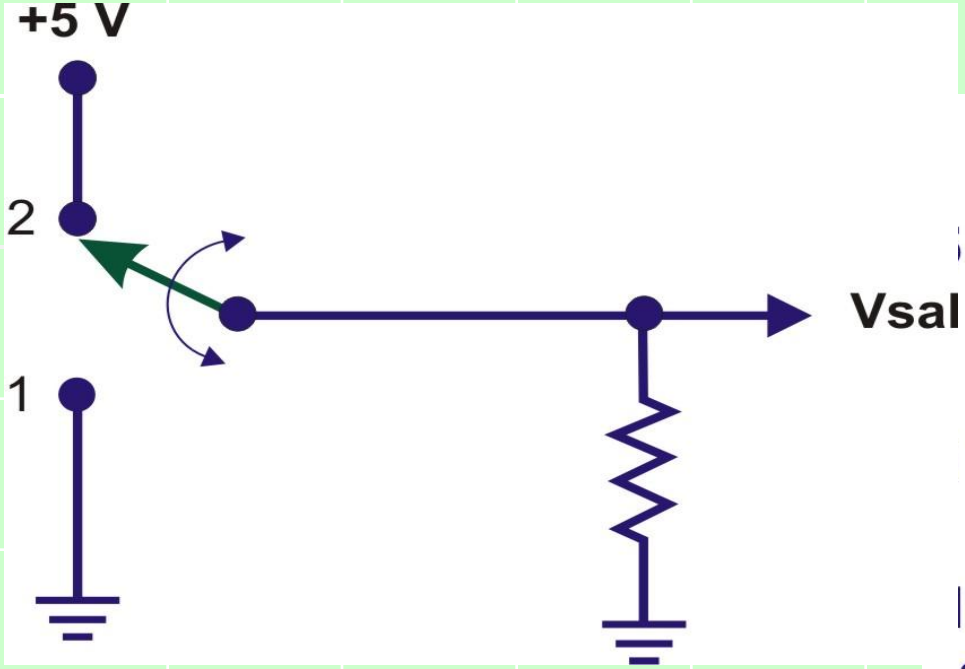


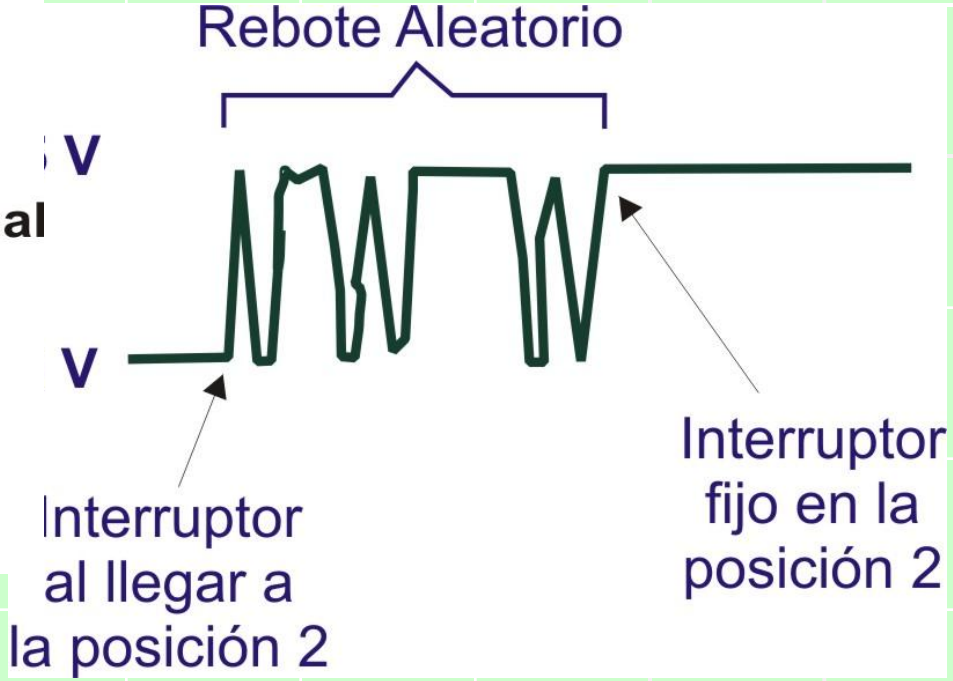

JAGG




JAGG

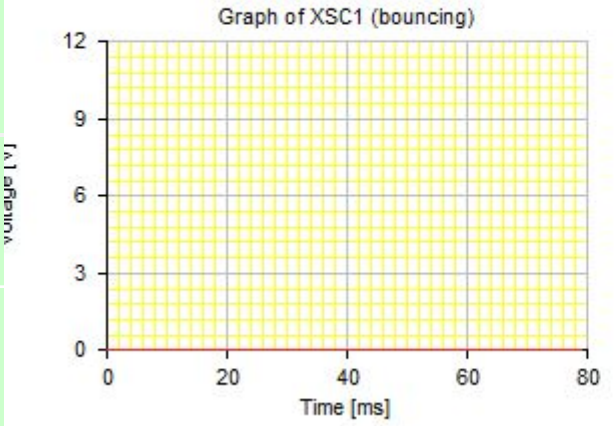

+5 V
2
1
Vsal

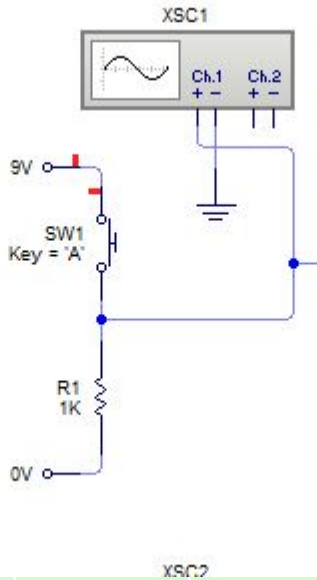

Rebote Aleatorio
V
V
Interruptor
al llegar a
la posición 2
Interruptor
fijo en la
posición 2

*JAGG*

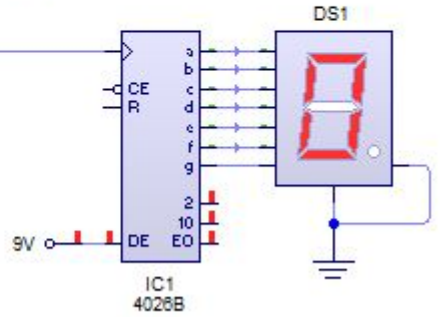
# Rebote de contacto

**Contact Bounce**—Demonstrates switch bounce.




Whenever a mechanical switch is pressed, the switch contacts will **bounce**, producing several very quick on and off signals. Each of these signals would be counted by a digital counter. Pressing switch **SW1** will show this effect.



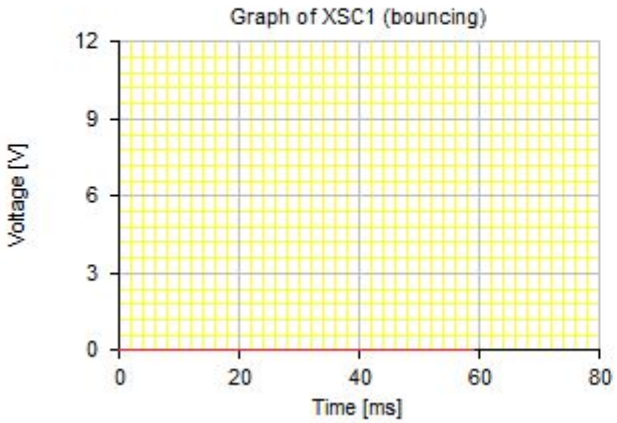

XSC2

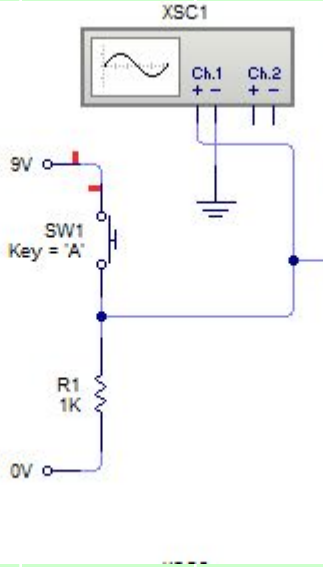
*JAGG*

# Rebote de contacto

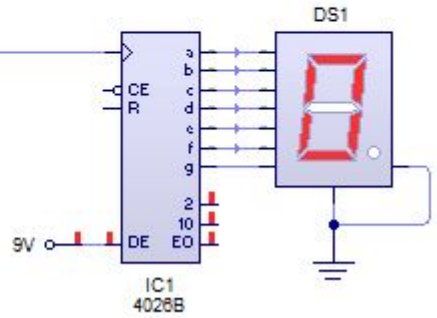
## Contact Bounce

**Contact Bounce**—Demonstrates switch bounce.

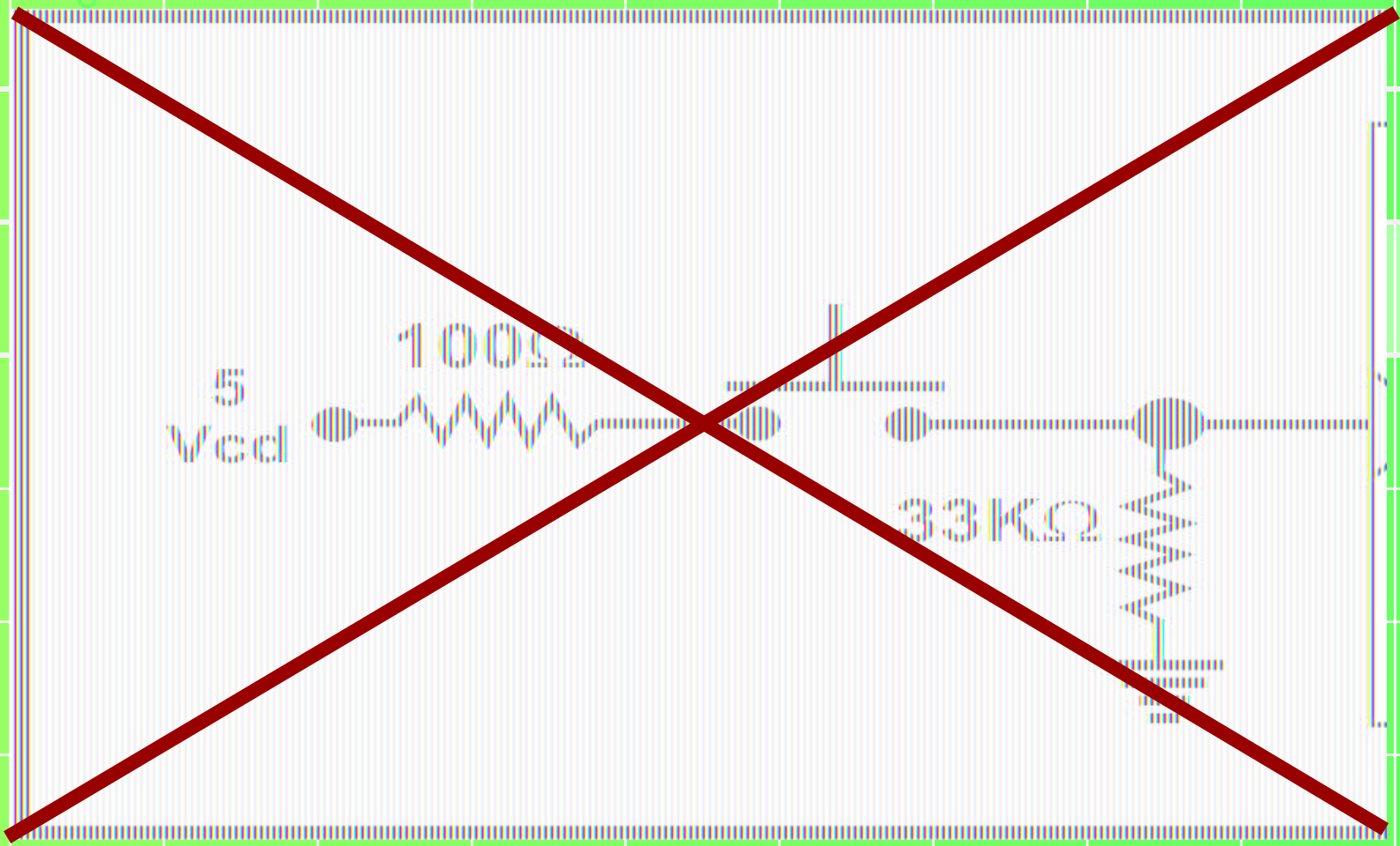



Whenever a mechanical switch is pressed, the switch contacts will **bounce**, producing several very quick on and off signals. Each of these signals would be counted by a digital counter. Pressing switch **SW1** will show this effect.




*JAGG*




JAGG


**Un interruptor mecánico no debe conectarse directamente al pin de reloj (CLK) de los flip-flops ni del PLD22V10, porque al presionarlo sus contactos rebotan y generan muchos pulsos indeseados en lugar de uno solo.**

**Además, los flancos son lentos e irregulares, lo que puede provocar errores en los registros y hacer que el circuito cambie de estado de forma incorrecta**

| ATF22V10 | | | | |
|---|---|---|---|---|
| **Parámetro** | **Descripción** | **Mínimo** | **Máximo** | **Unidad** |
| **tW (tWL, tWH)** | Ancho del pulso de reloj (Clock Width) | 3.0 | 6.0 | ns |

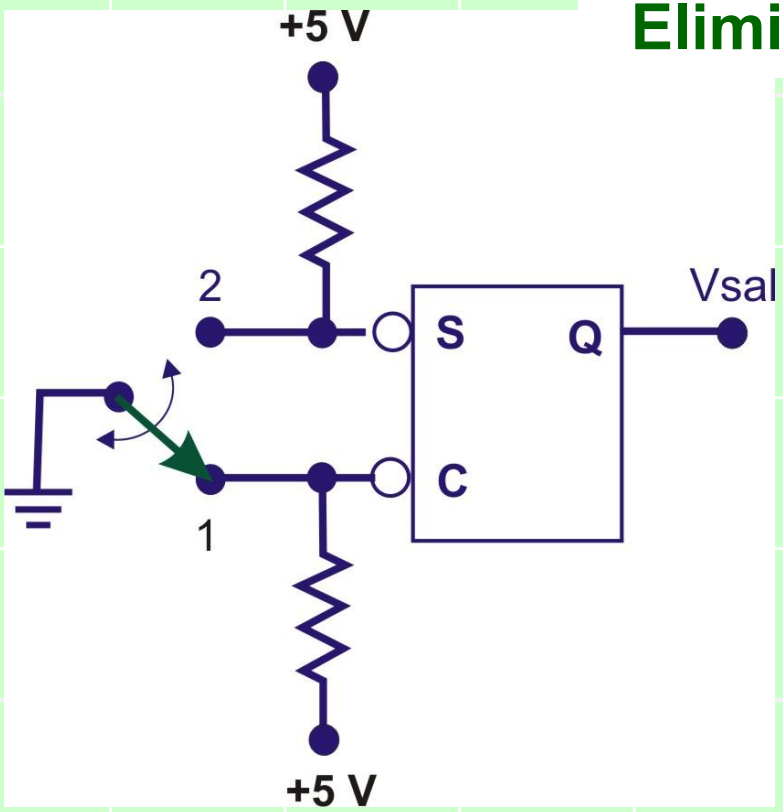
**El ancho mínimo del pulso de reloj (tWH y tWL) debe ser de 3 ns, lo que equivale a una frecuencia máxima teórica de 166 MHz**

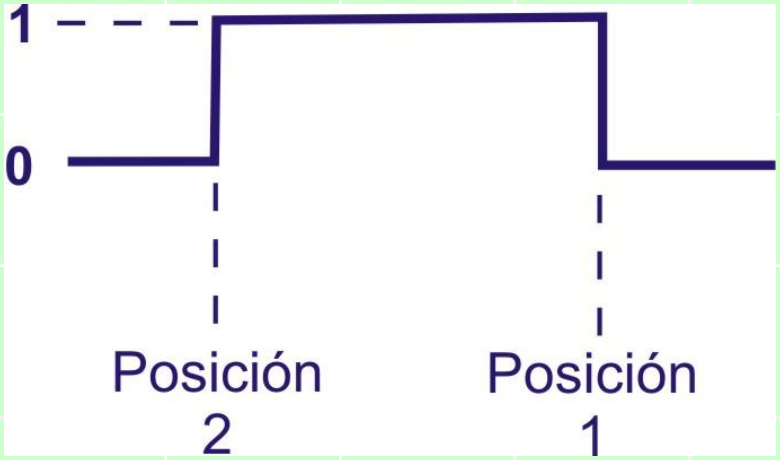
*JAGG*

# Multivibrador Biestable

**Circuito que tiene dos estados estables y permanece en uno de ellos hasta que una señal externa lo cambia al otro.**
**Es la base de los Flip-Flops, usados para almacenar un bit de información**

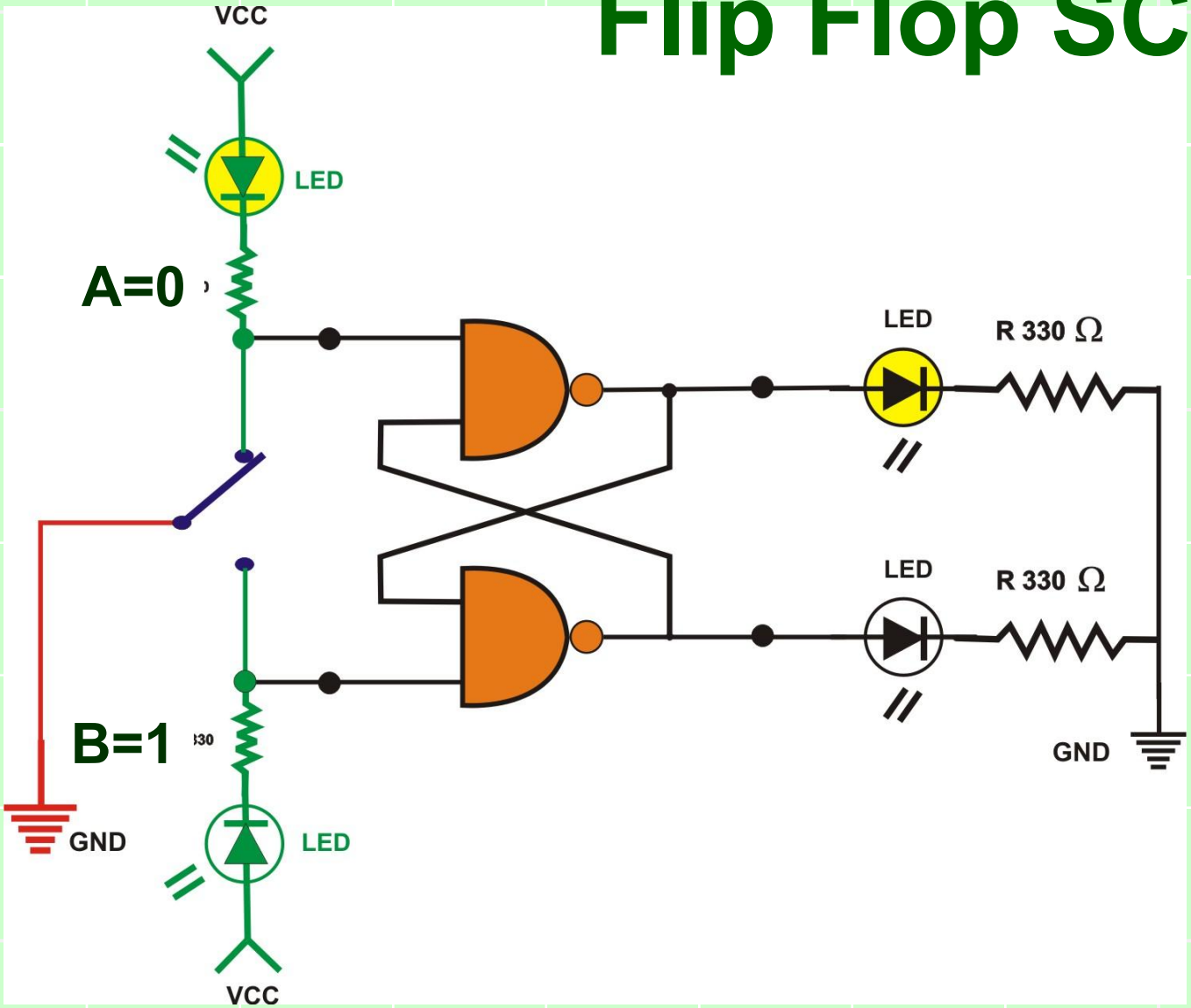
## Flip Flop SC SET CLEAR
## Eliminador de Rebotes



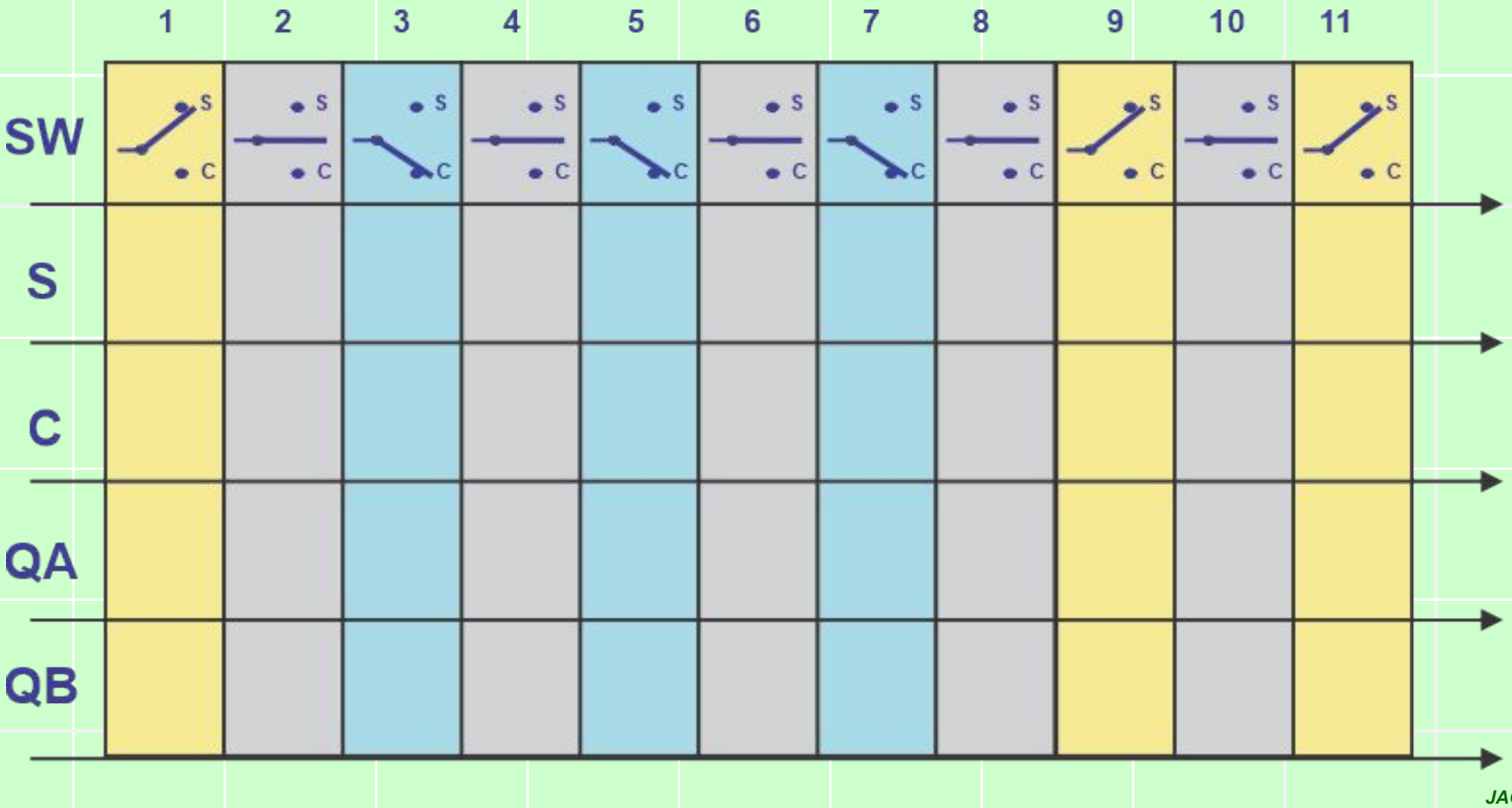

JAGG

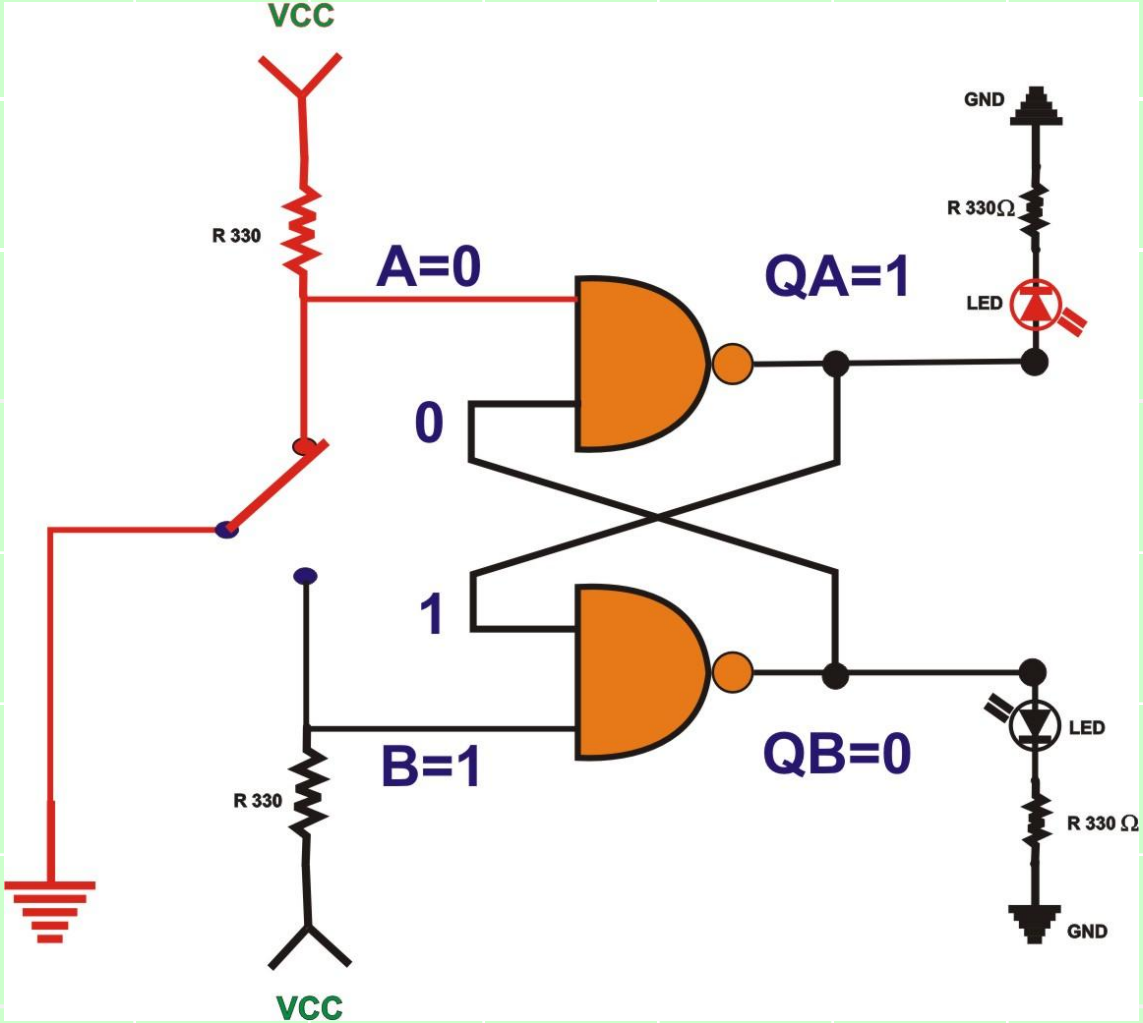
# Flip Flop SC




JAGG

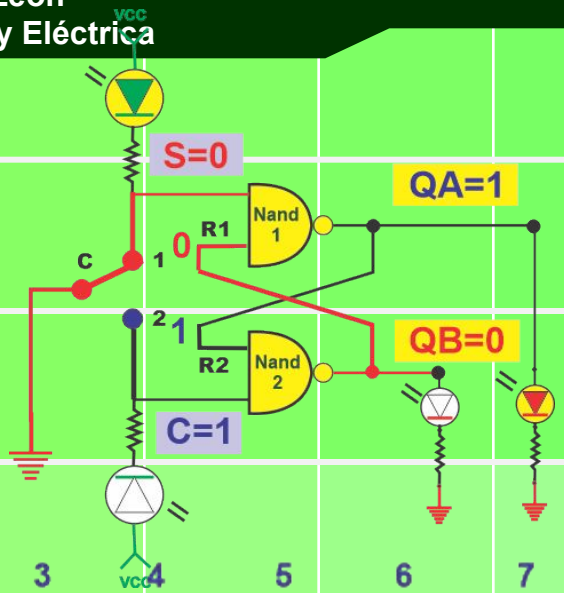
# Grafica de tiempos

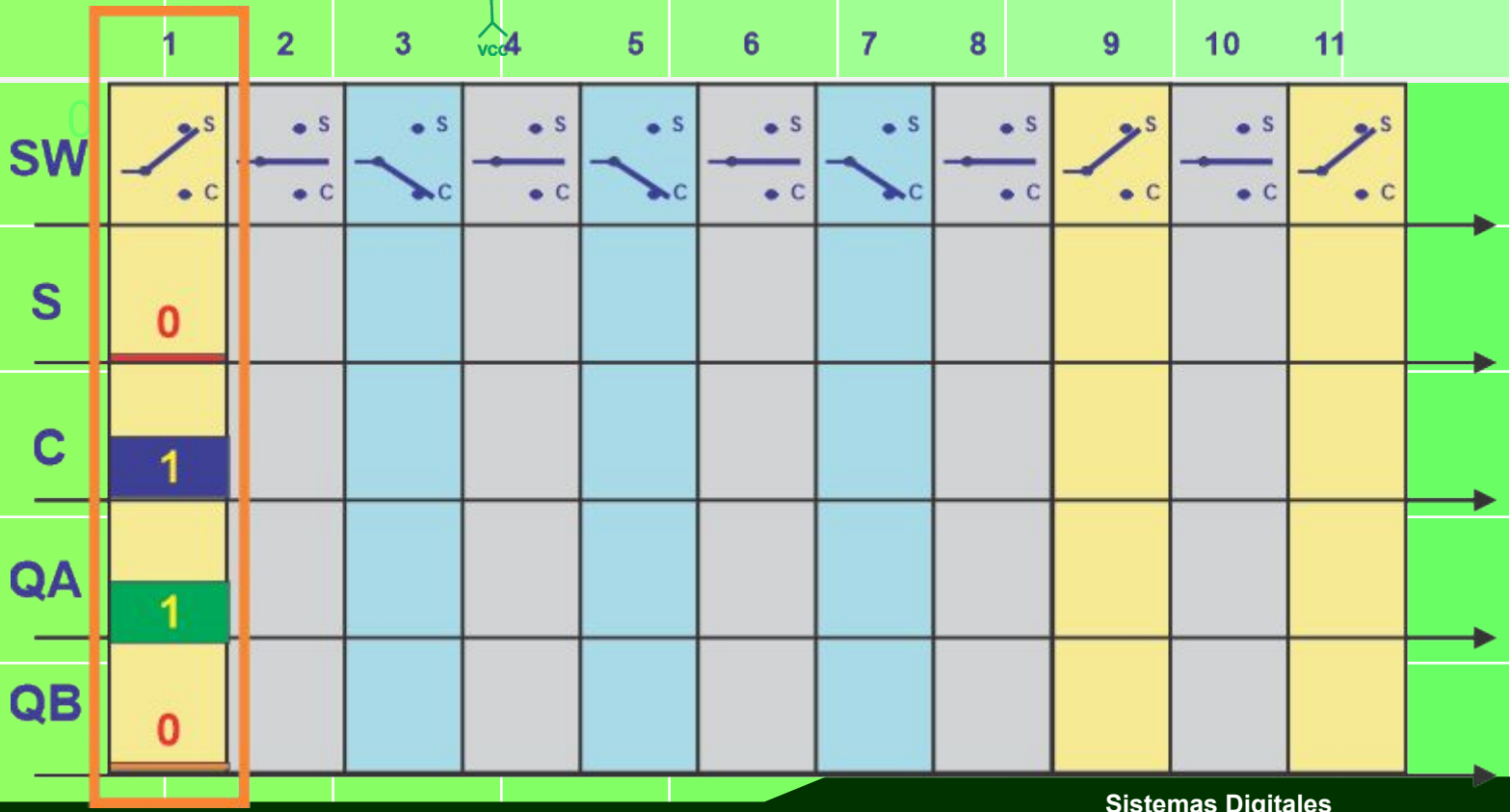


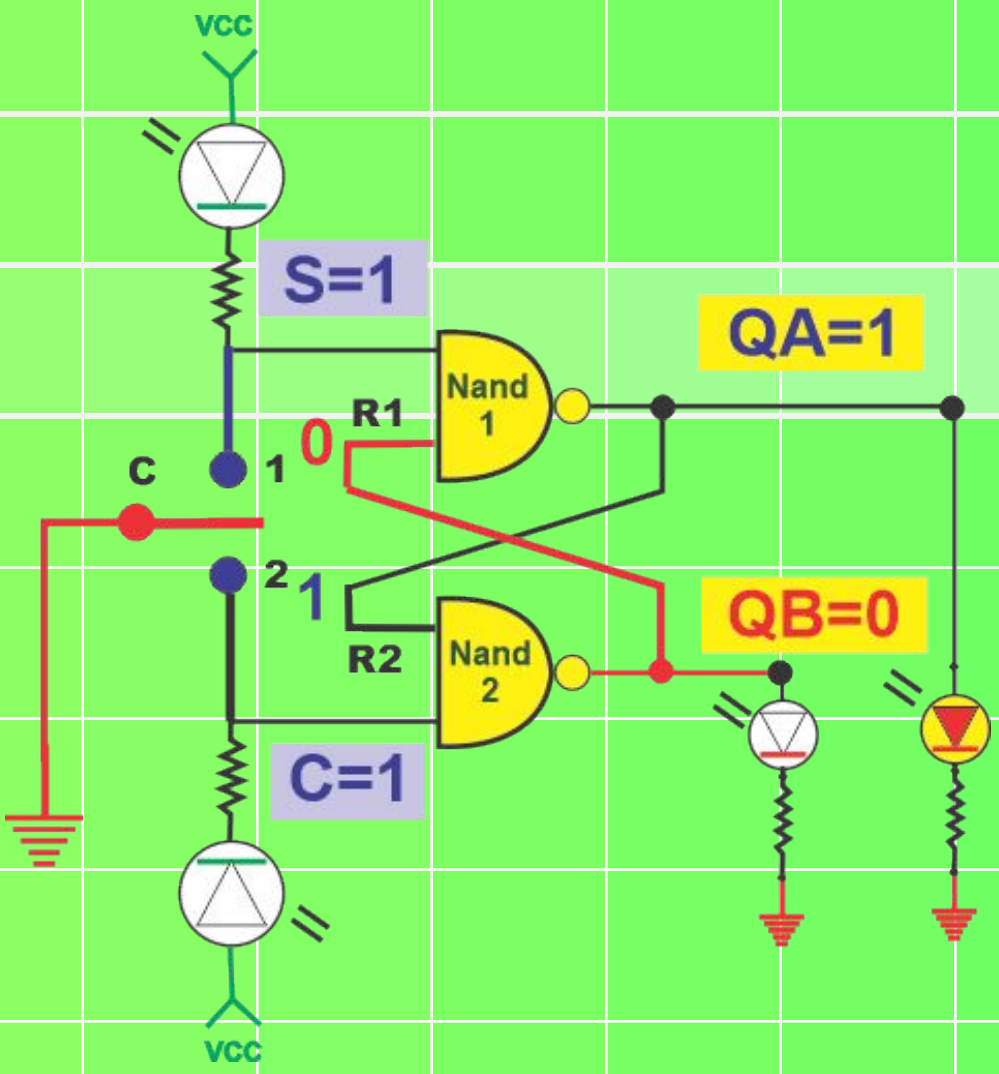



JAGG

# T1

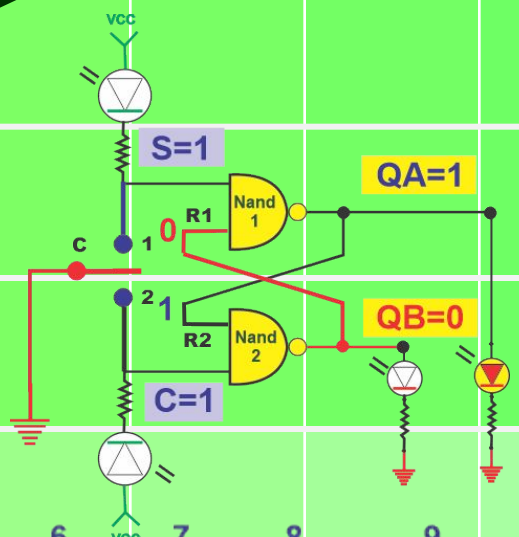

VCC
S=0
QA=1
Nand 1
R1
0
C
1
2
1
QB=0
R2
Nand 2
C=1
VCC

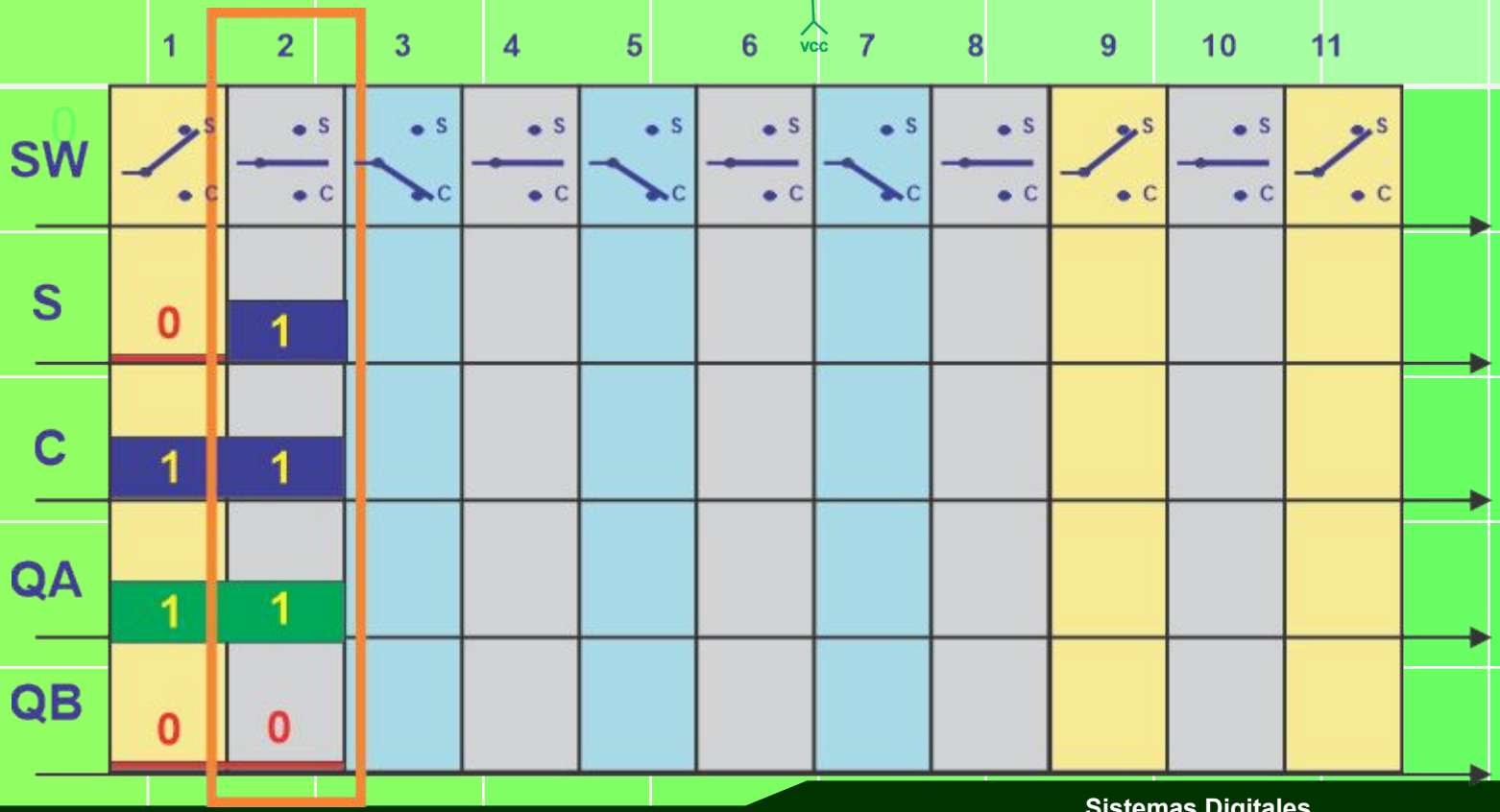

1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
0
1
1
0
JAGG

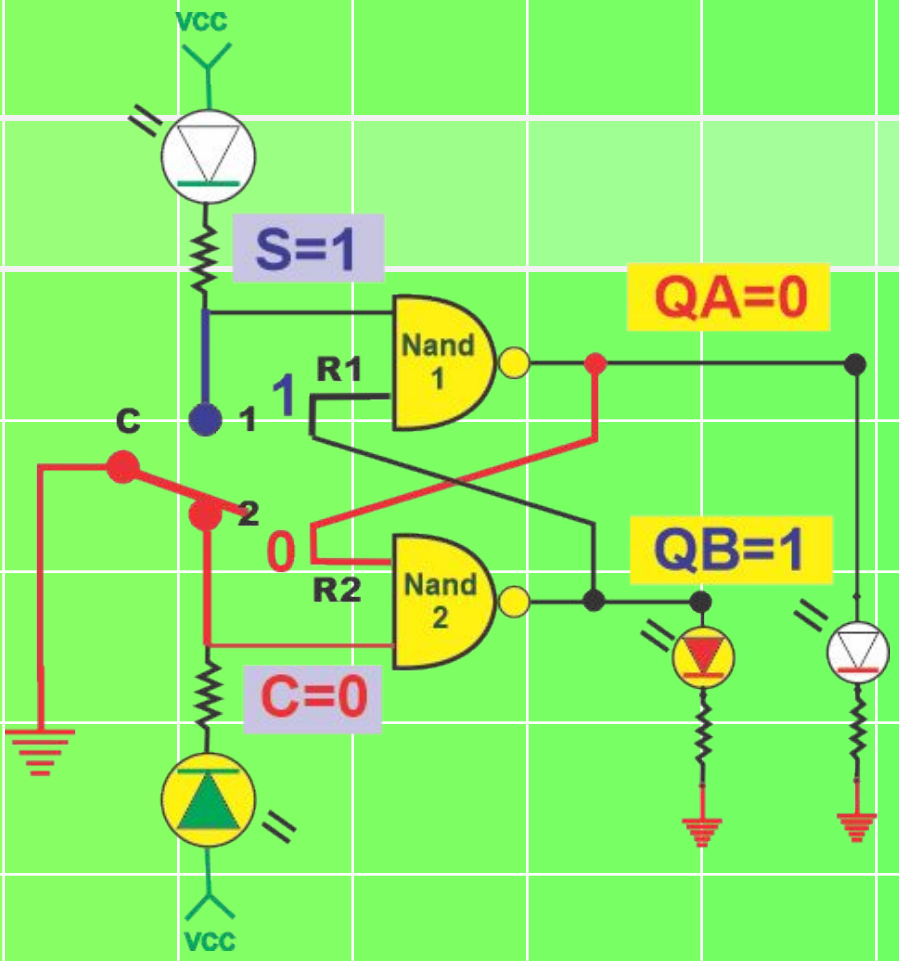



JAGG

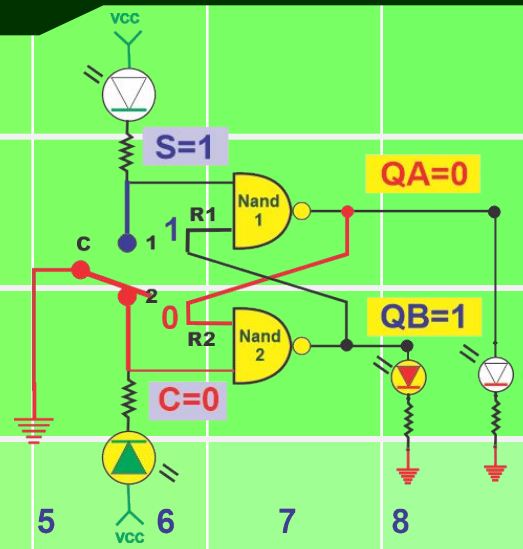
# T2


Vcc
S=1
QA=1
R1
Nand 1
C
1
0
2
1
R2
Nand 2
QB=0
C=1
Vcc

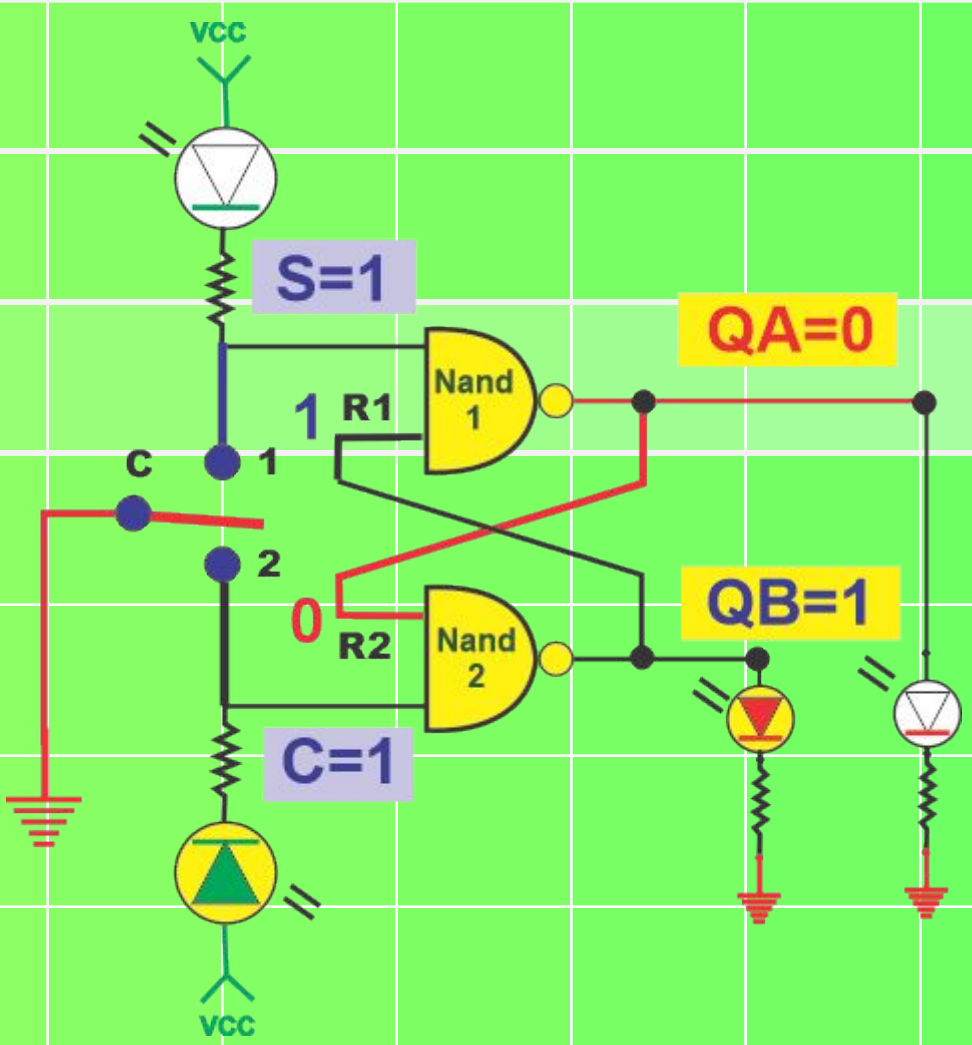

1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
0
1
1
1
1
1
0
0
JAGG


VCC
S=1
QA=0
Nand 1
R1
1
C
1
2
0
R2
Nand 2
QB=1
C=0
VCC
JAGG

# T3

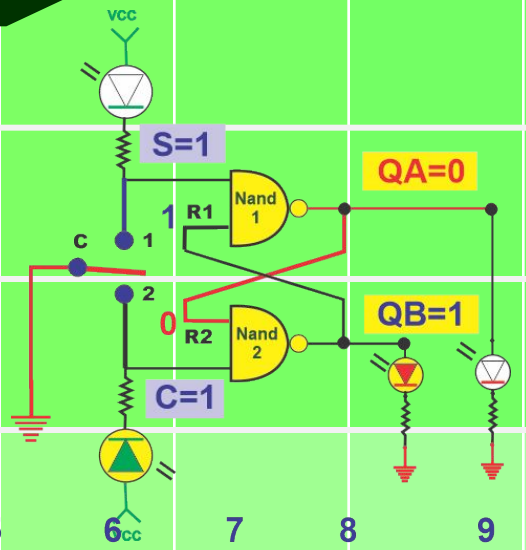

VCC
S=1
QA=0
Nand 1
R1
1
1
C
1
2
0
R2
Nand 2
QB=1
C=0
VCC

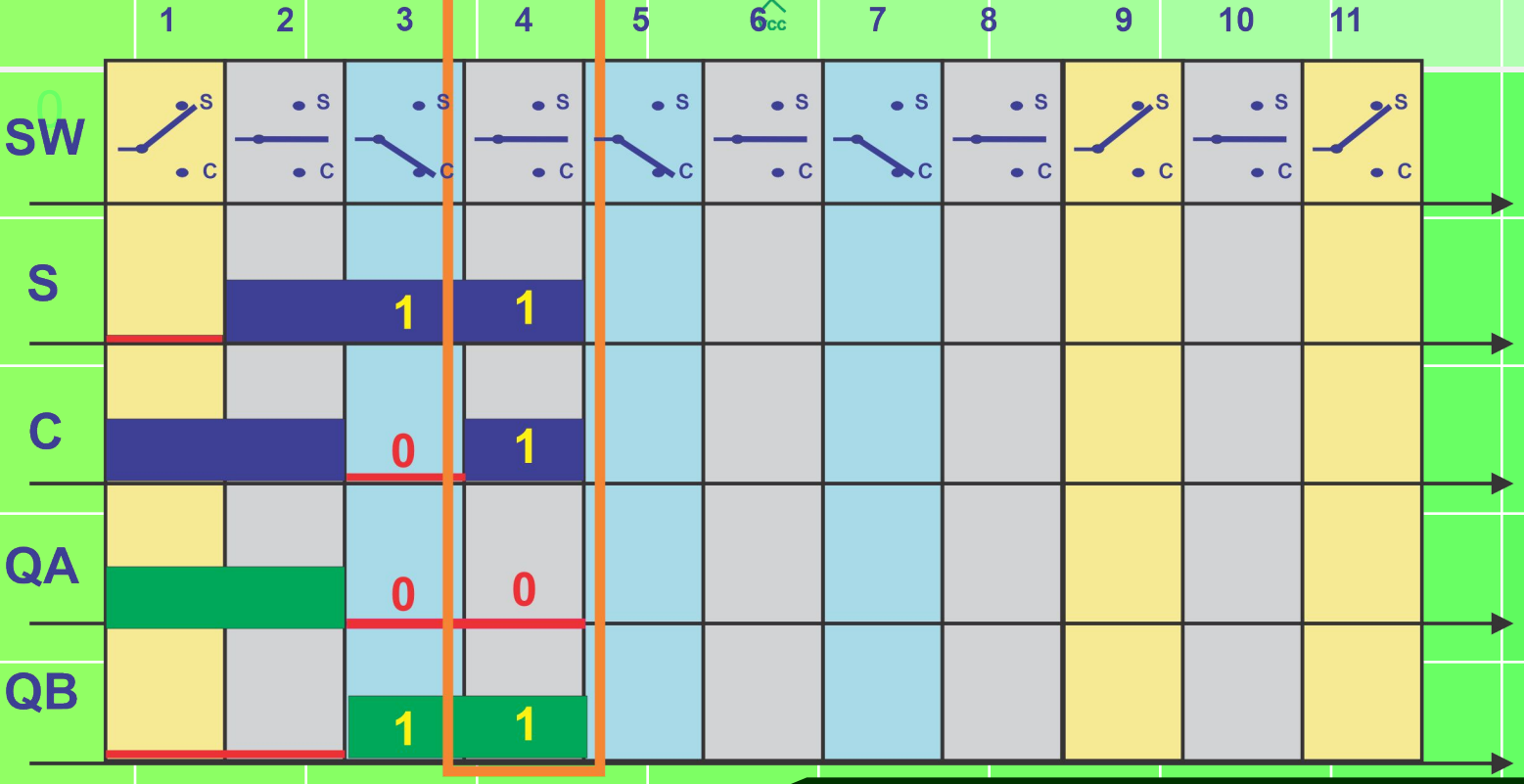

1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
S
C
1
1
1
0
1
0
0
1
JAGG



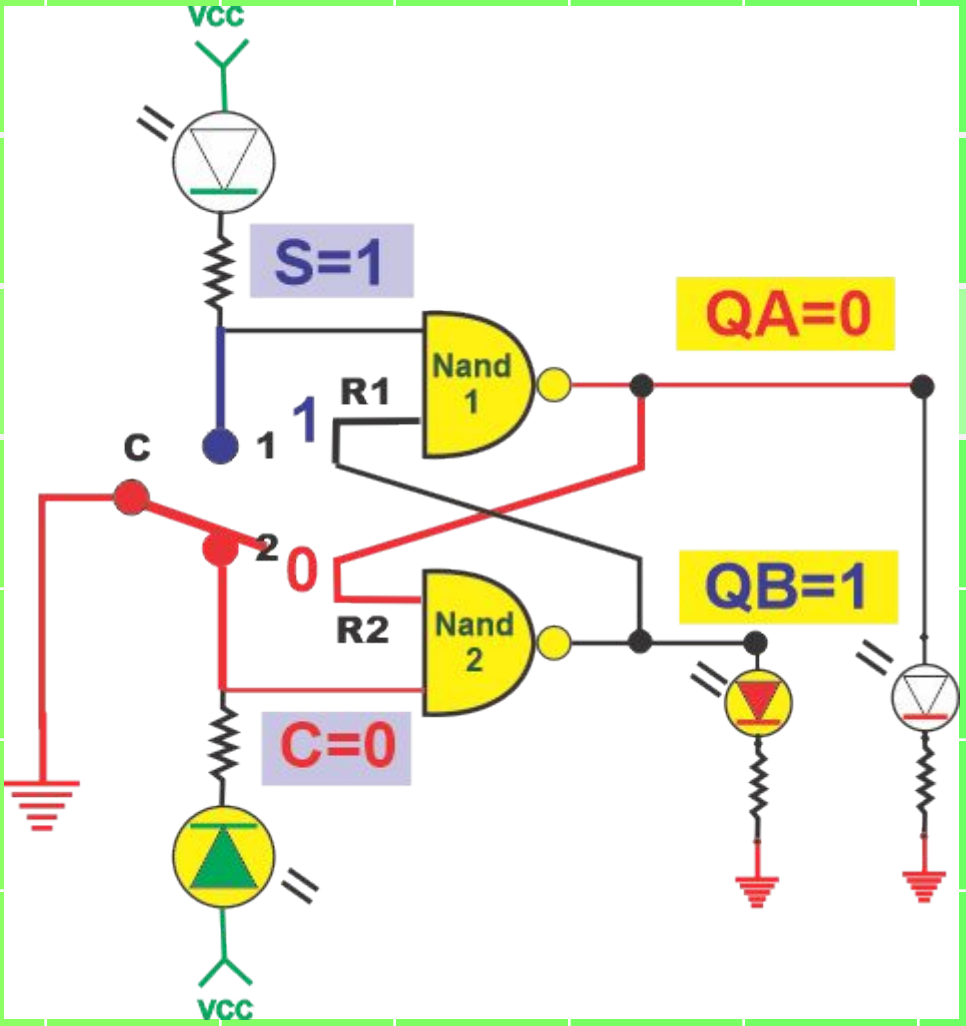

JAGG

# T4


VCC
S=1
R1
Nand 1
QA=0
1
C
1
2
0
R2
Nand 2
QB=1
C=1
Vcc

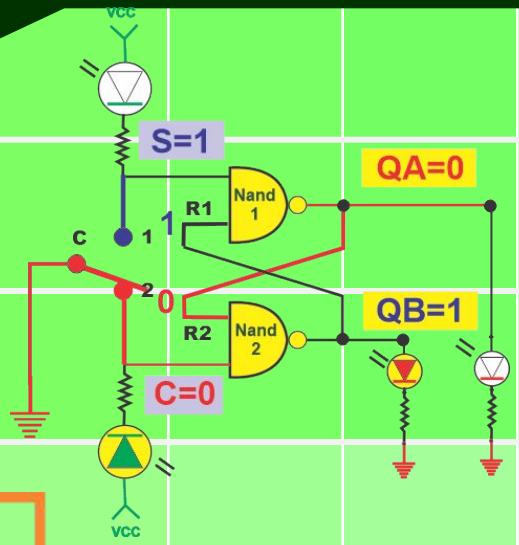

1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
S
C
1
1
0
1
0
0
1
1
JAGG

# T5




JAGG

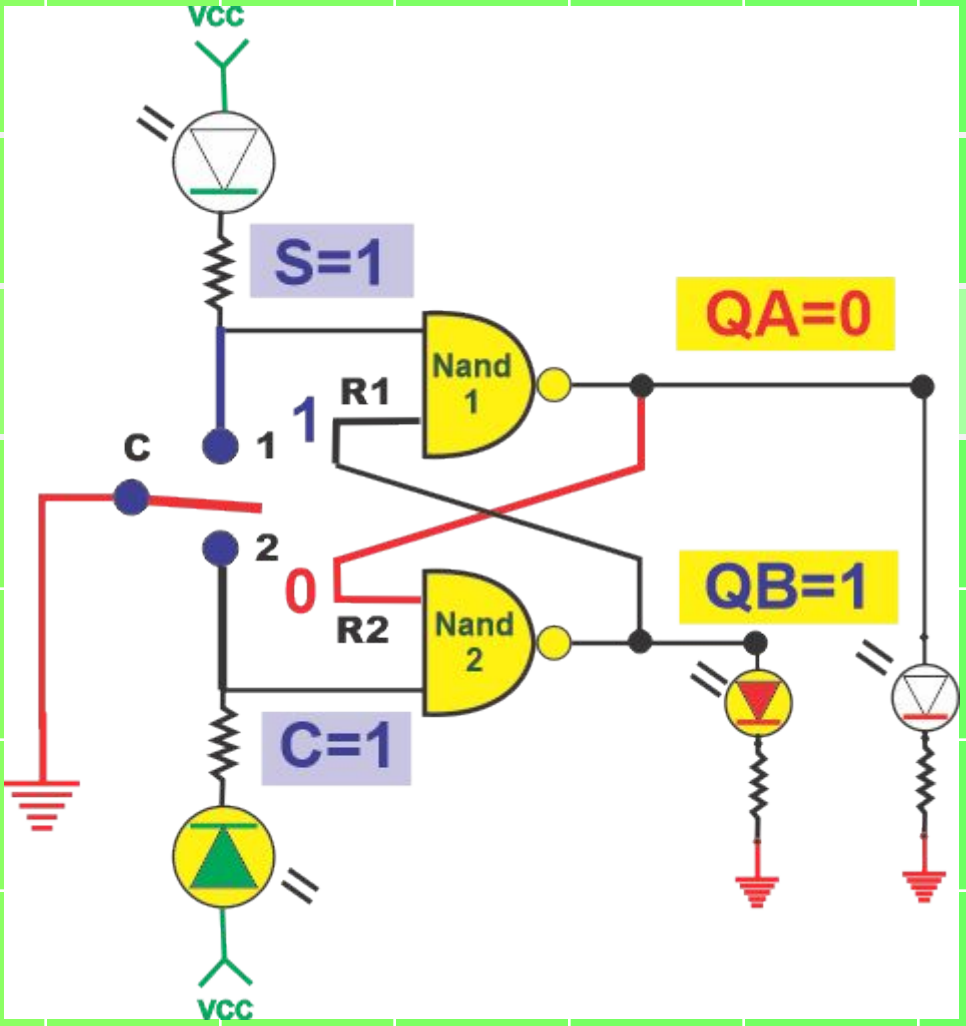
# T5


VCC
S=1
QA=0
Nand 1
R1
1
C
1
2
0
QB=1
R2
Nand 2
C=0
VCC


1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
S
C
1
1
1
0
0
0
1
1

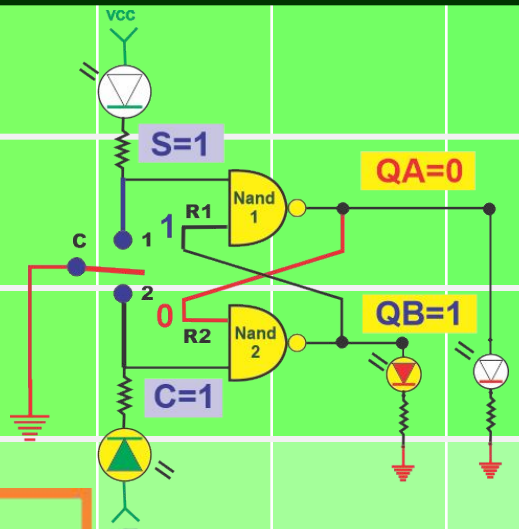
JAGG

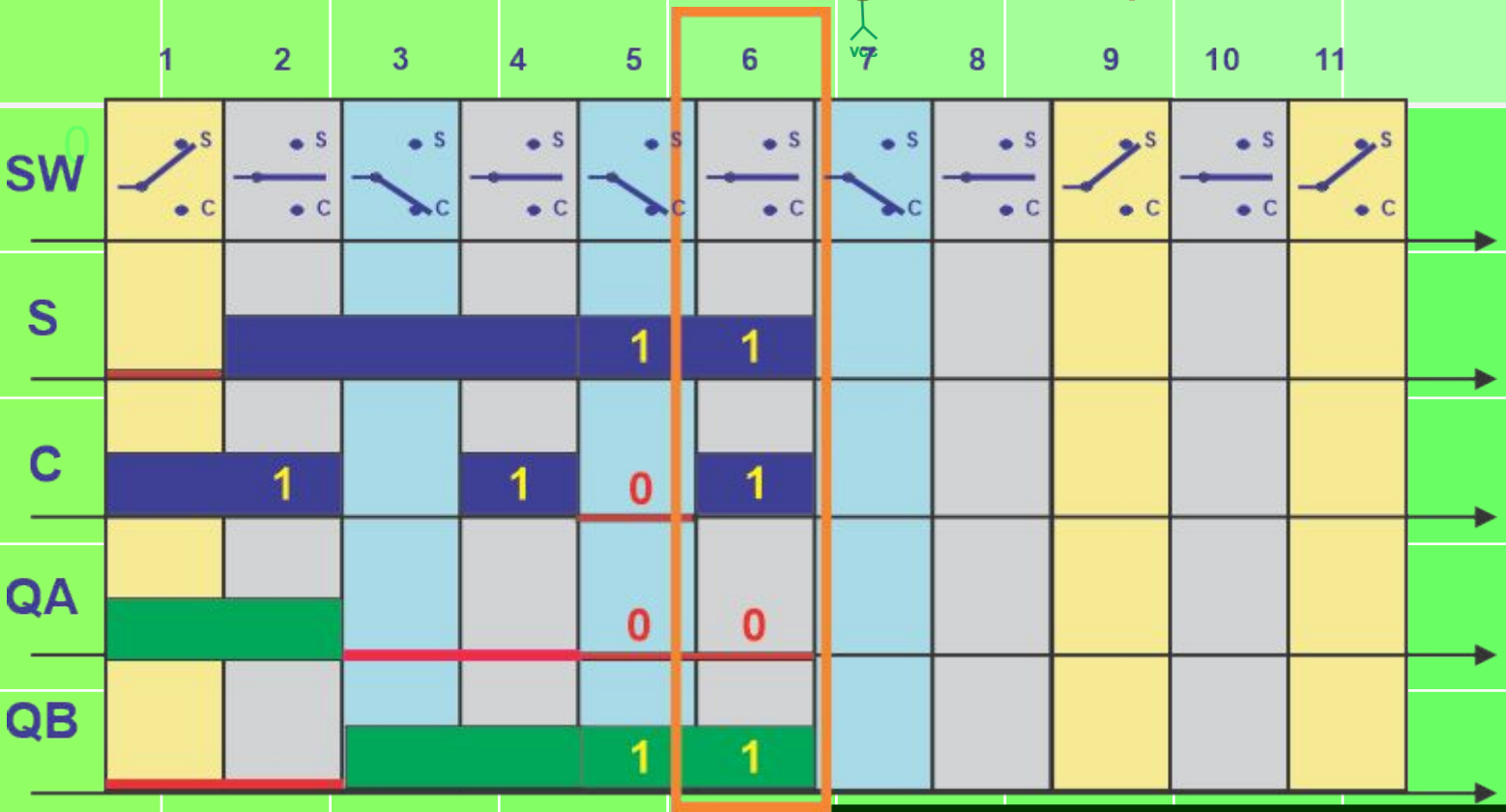
# T6



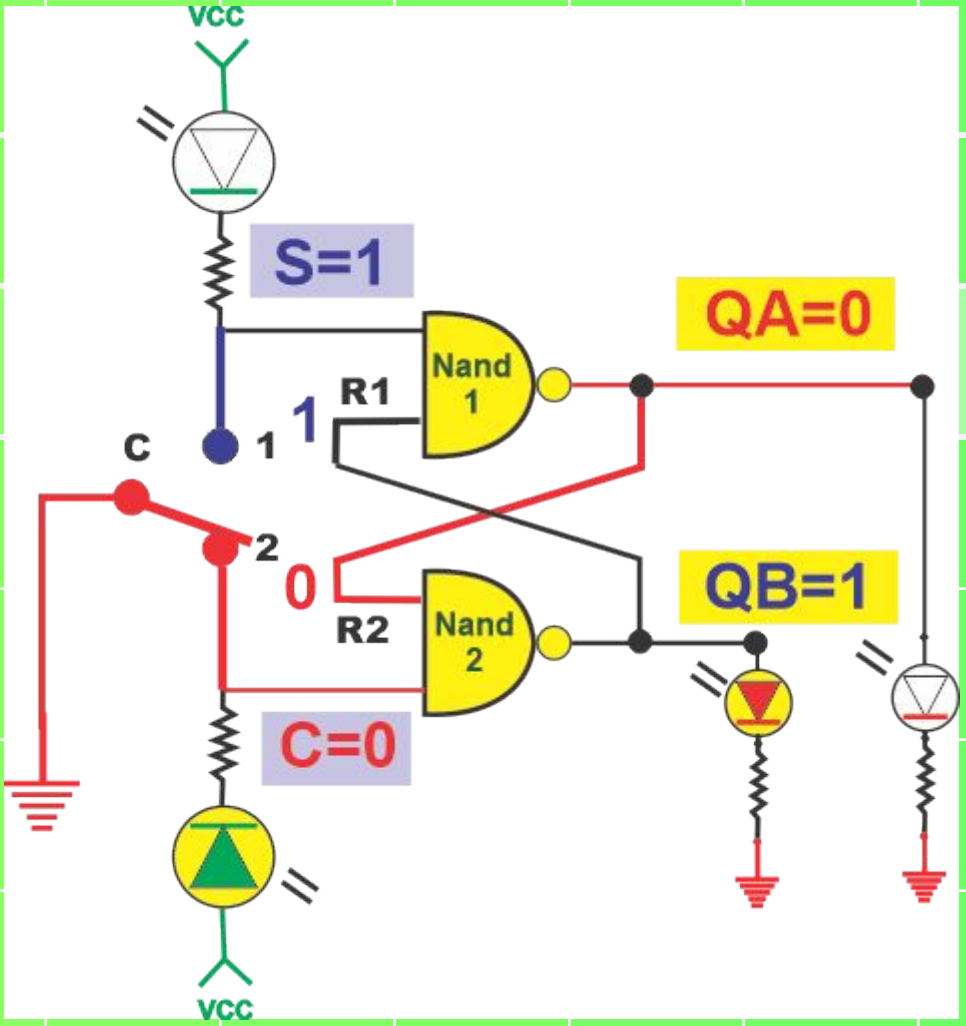

JAGG

# T6


VCC
S=1
QA=0
Nand 1
R1
1
C
1
2
0
R2
Nand 2
QB=1
C=1
VCC

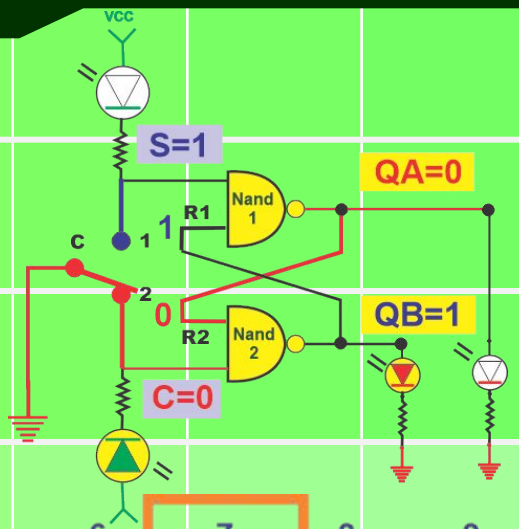

1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
S
C
1
1
1
1
0
1
0
0
1
1
JAGG

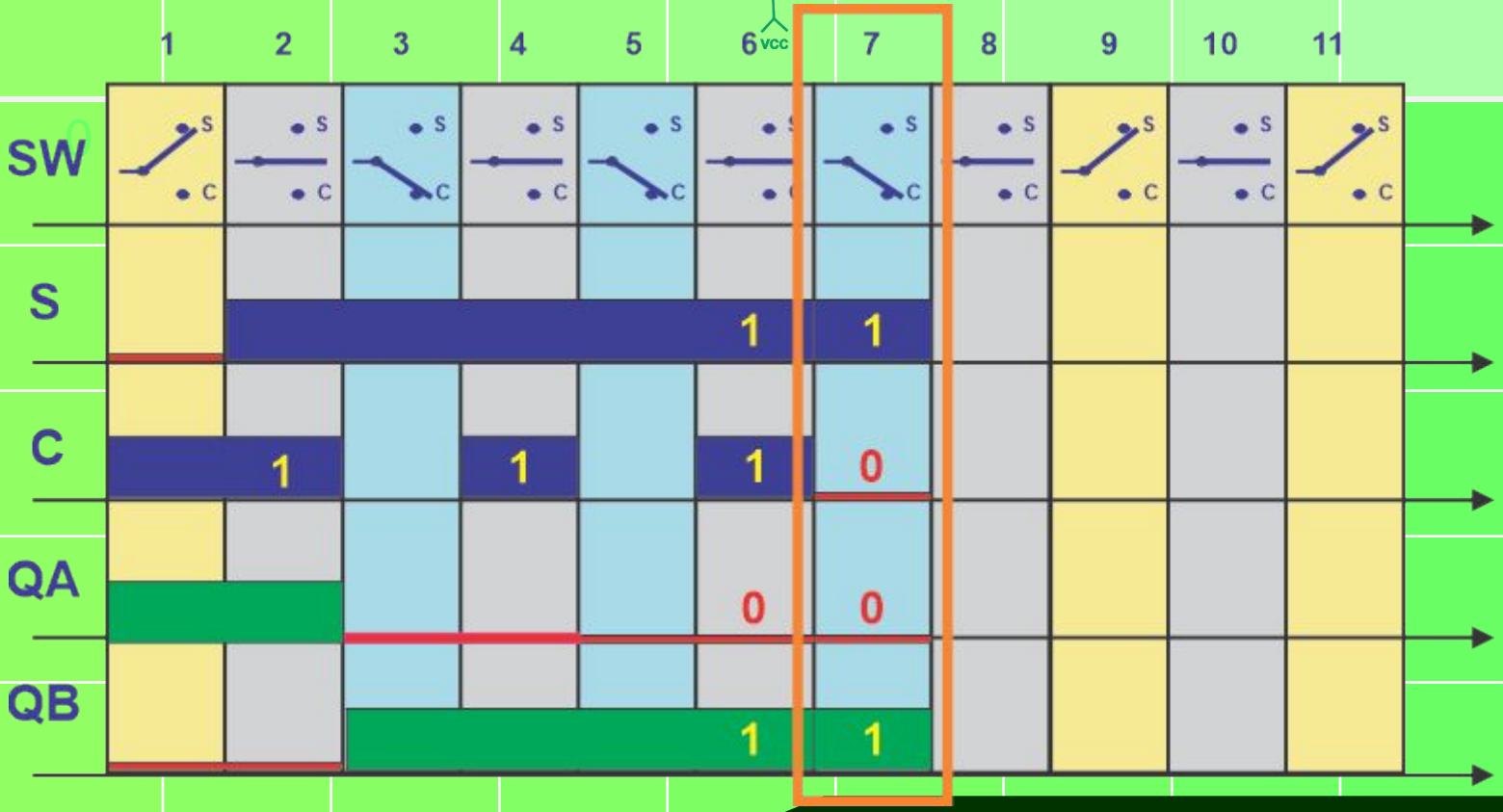
# T7




JAGG

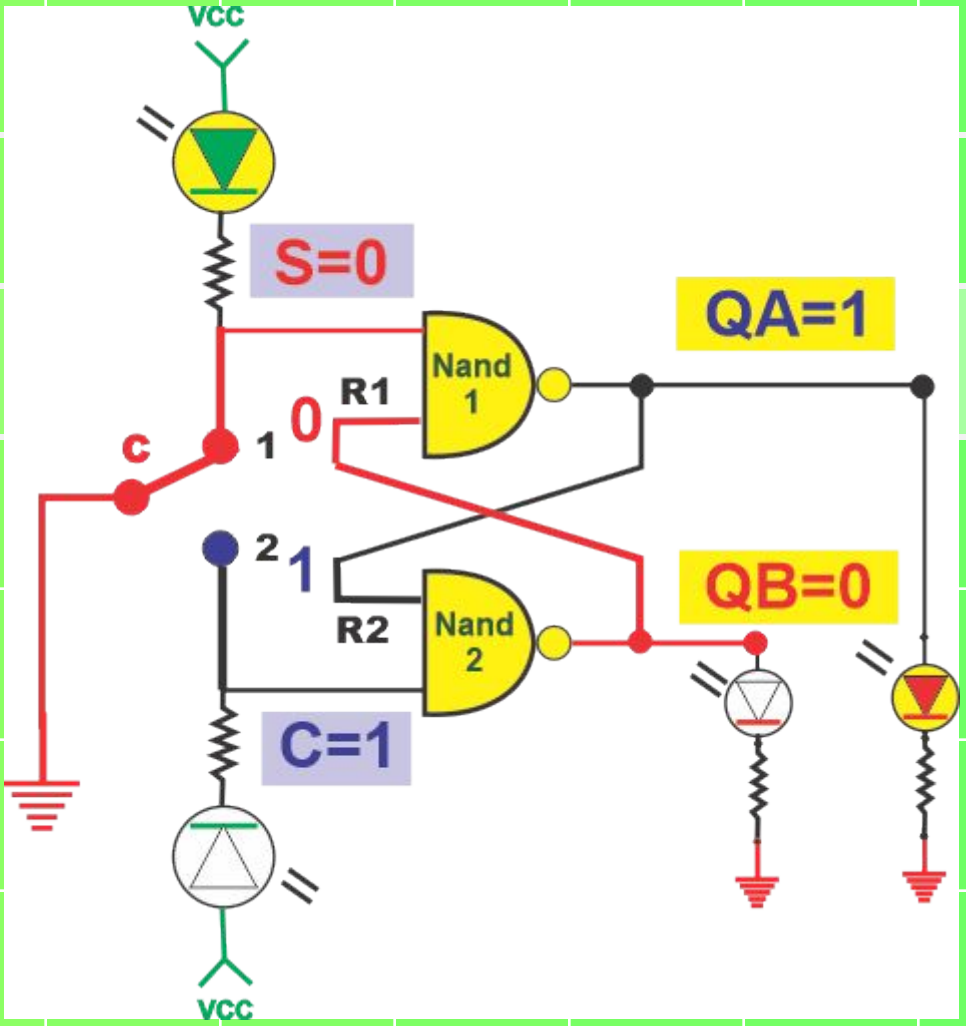
# T7


VCC
S=1
QA=0
Nand 1
R1
C
1
1
2
0
R2
Nand 2
QB=1
C=0
VCC


1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
S
C
1
1
1
1
1
0
0
0
1
1

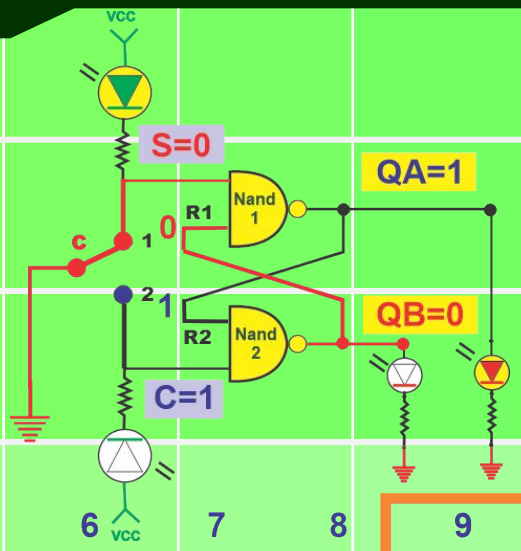
JAGG

# T9




JAGG

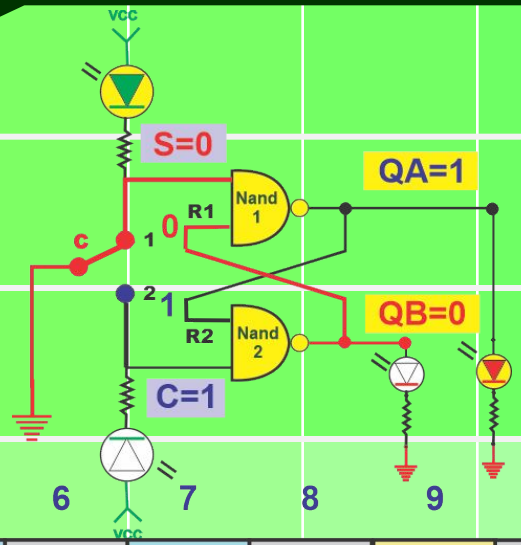
# T9


VCC
S=0
QA=1
Nand 1
R1
0
C
1
2
1
R2
Nand 2
QB=0
C=1
VCC


1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
S
C
1
0
1
1
1
1
1
1
0
JAGG

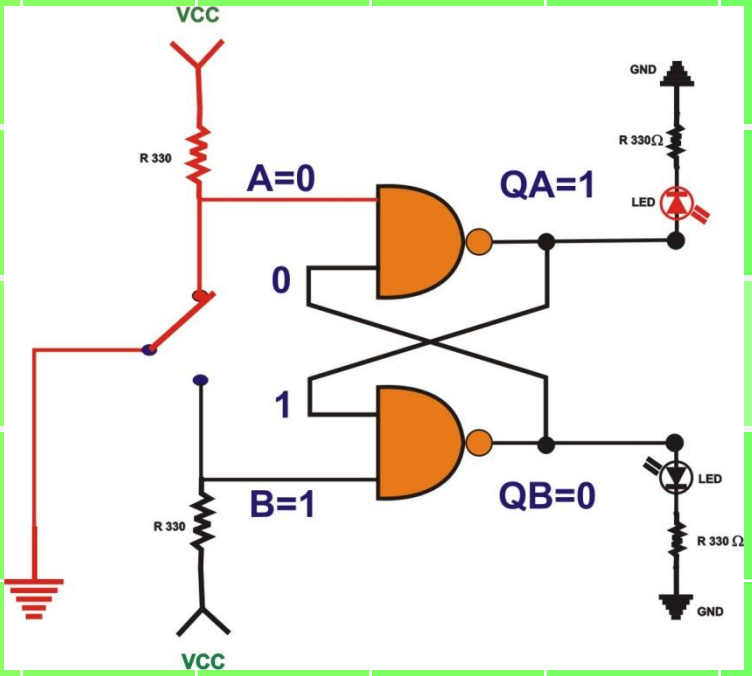
# T1

1


VCC
S=0
QA=1
Nand 1
R1
0
1
c
2
1
QB=0
R2
Nand 2
C=1
VCC


1
2
3
4
5
6
7
8
9
10
11
SW
S
C
QA
QB
S
C
1
0

JAGG

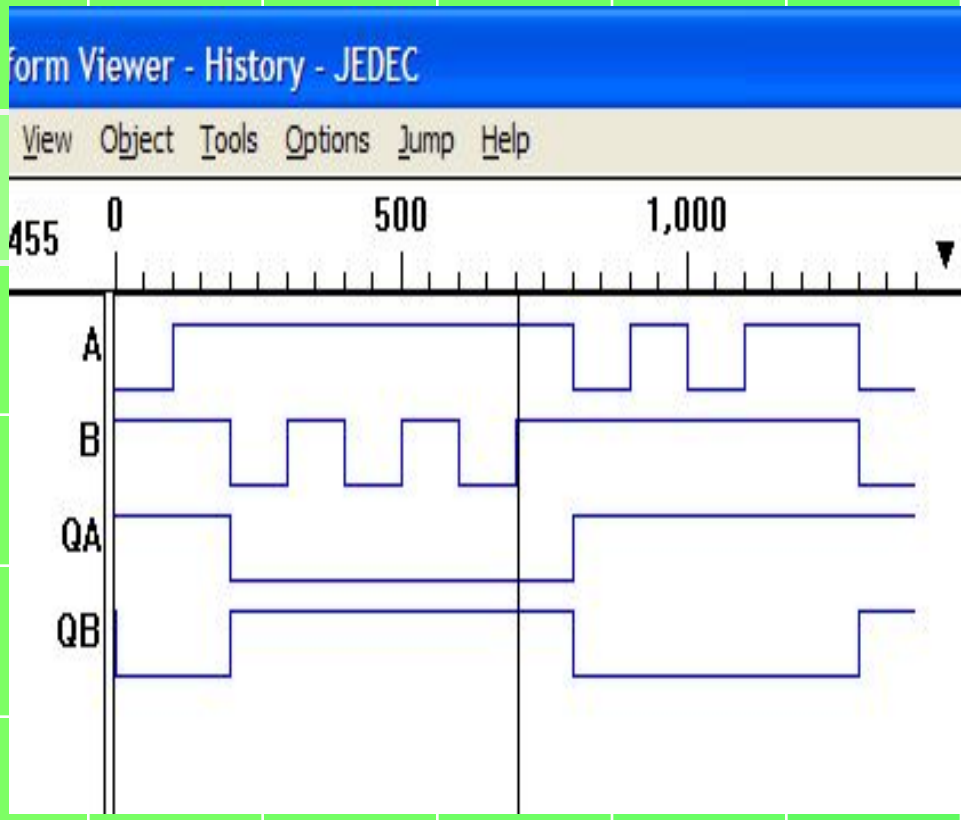



$$QA = !( A \ \& \ QB)$$

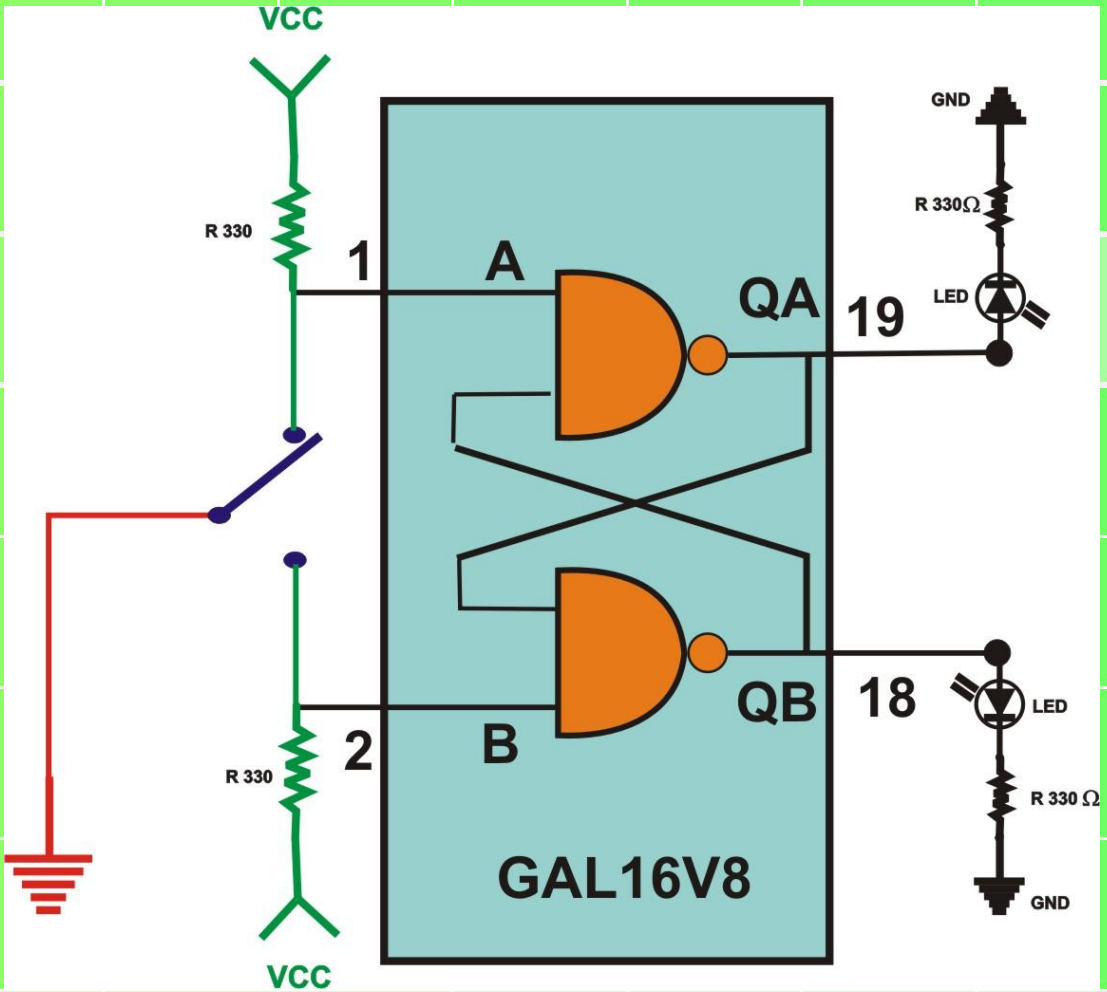
$$QB = !( B \ \& \ QA)$$

JAGG

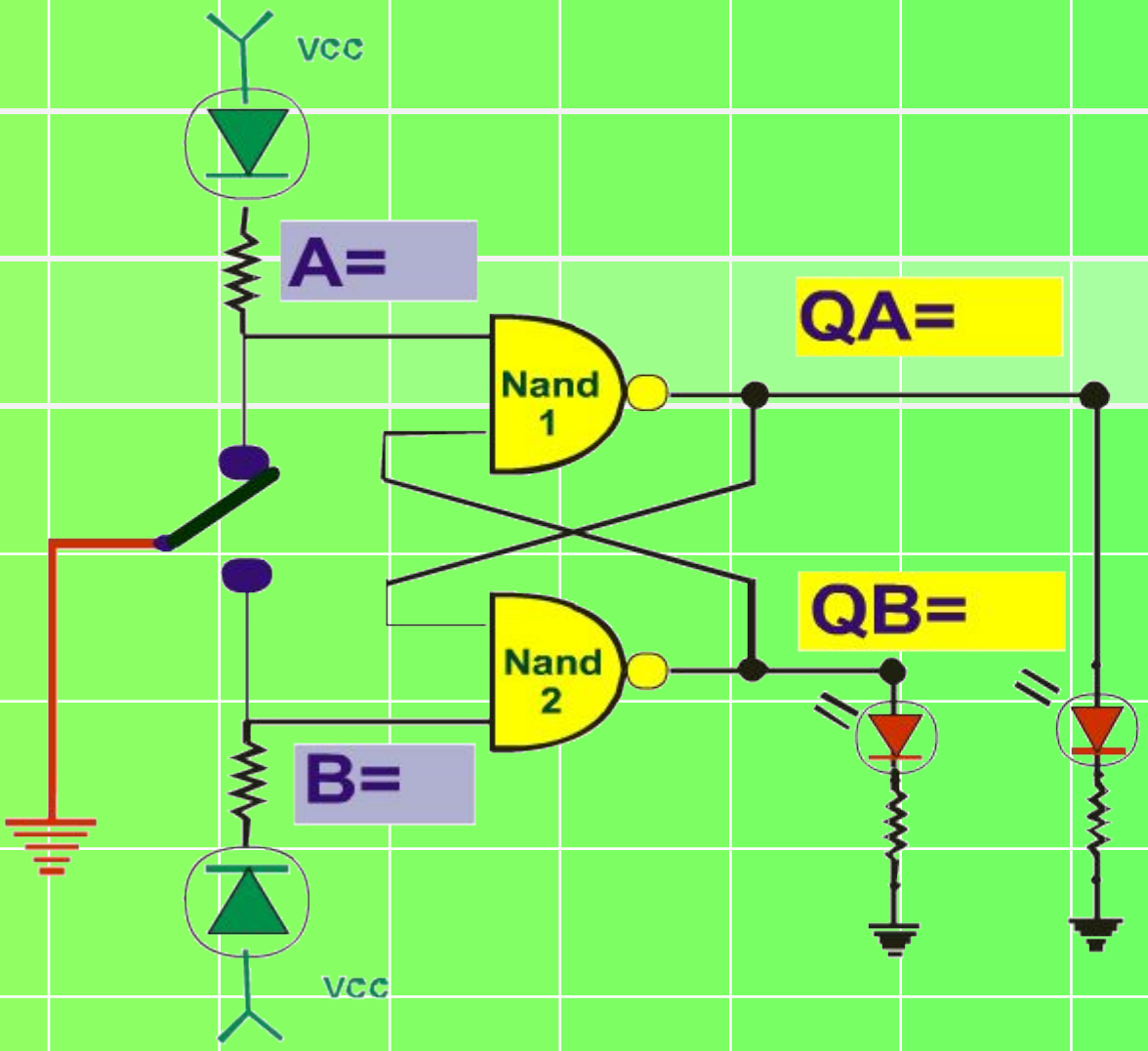
```
MODULE sc
"entradas
A,B pin 1,2;
"salidas
QA,QB pin 19,18 istype 'com';
equations
QA=!(A&QB);
QB=!(B&QA);
test_vectors
([A,B]->[QA,QB])
[0,1]->[.x.,.x.];
[1,1]->[.x.,.x.];
[1,0]->[.x.,.x.];
[1,1]->[.x.,.x.];
[1,0]->[.x.,.x.];
[1,1]->[.x.,.x.];
[1,0]->[.x.,.x.];
[1,1]->[.x.,.x.];
[0,1]->[.x.,.x.];
[1,1]->[.x.,.x.];
[0,1]->[.x.,.x.];
[1,1]->[.x.,.x.];
[1,1]->[.x.,.x.];
[0,0]->[.x.,.x.];
END
```

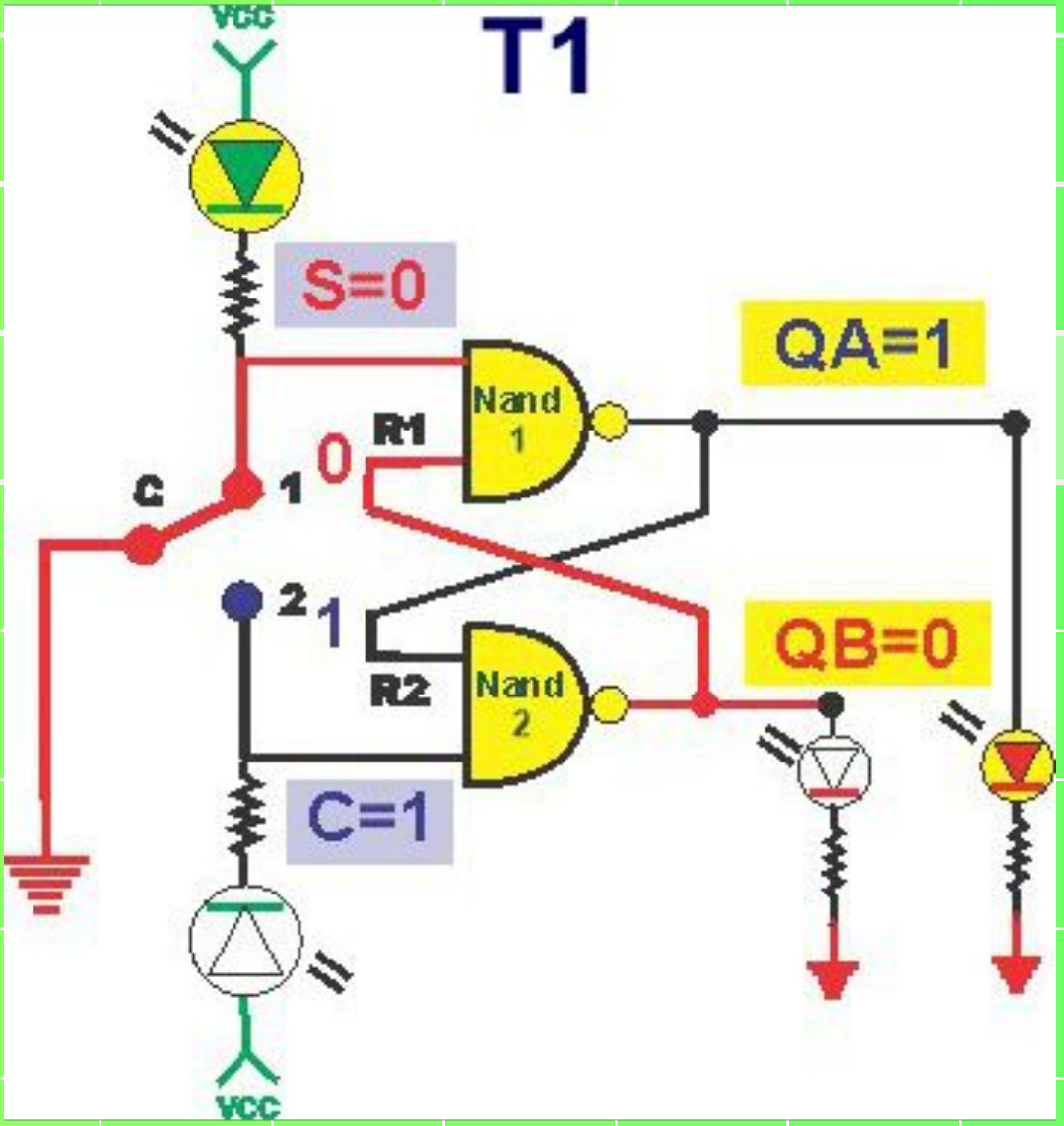



JAGG


VCC
R 330
1
A
QA
19
GND
R 330Ω
LED
2
B
QB
18
LED
R 330 Ω
GND
R 330
VCC
GAL16V8

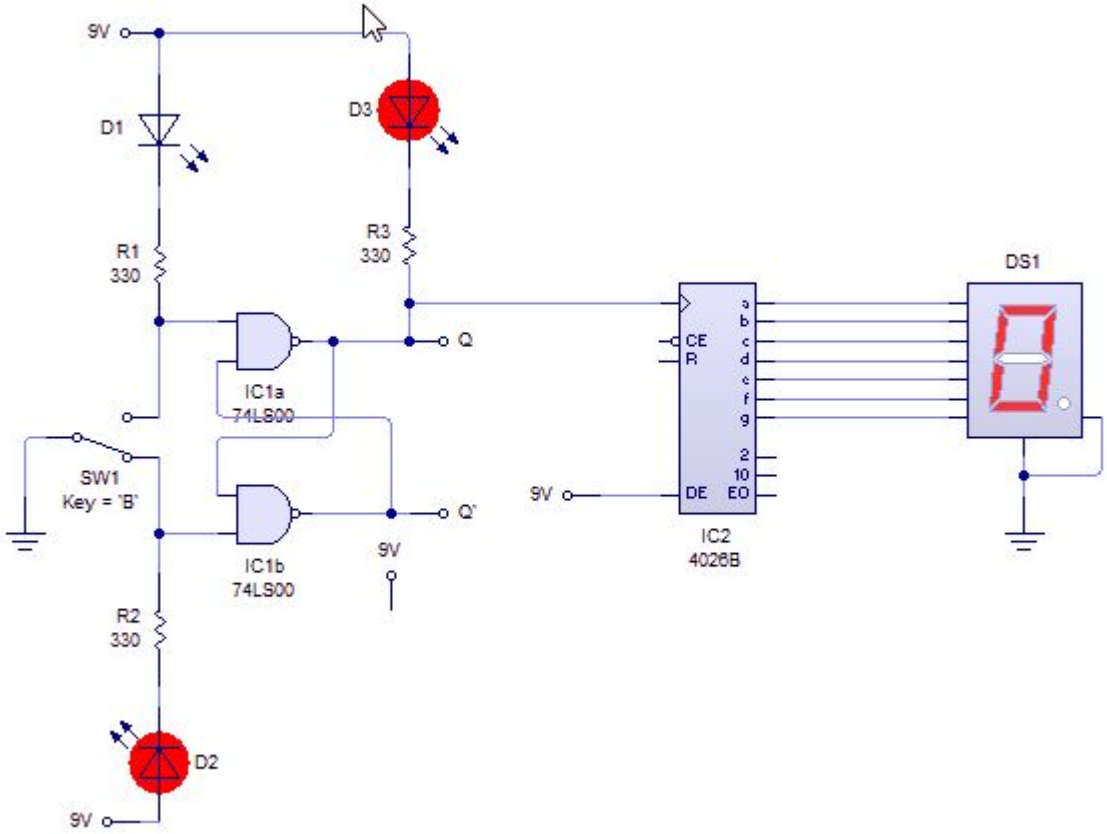
JAGG


1
0
1
0
Vcc
A=
Nand
1
QA=
Nand
2
QB=
B=
Vcc
JAGG




JAGG

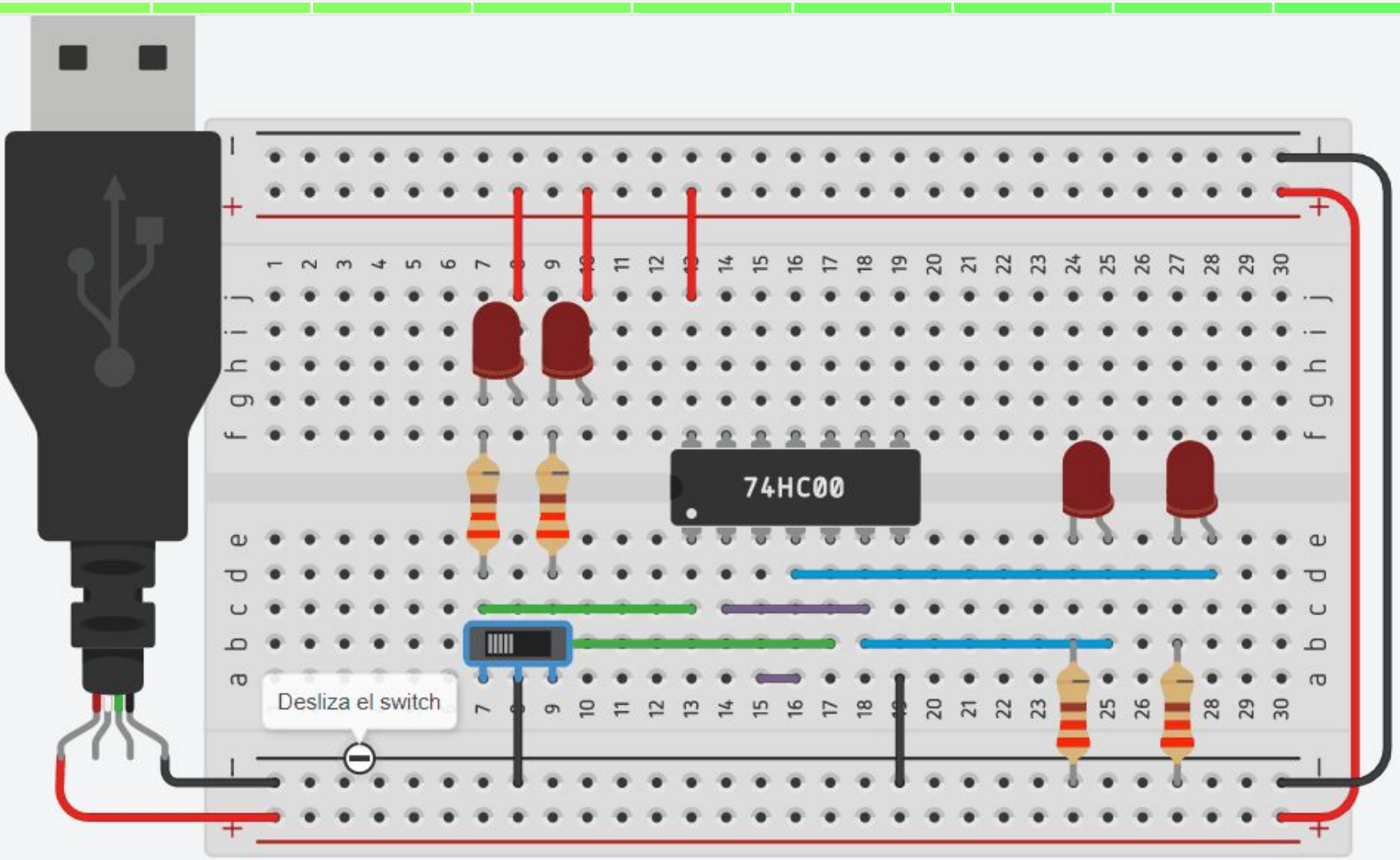
# Flip Flop SC




# Simulación en LiveWire

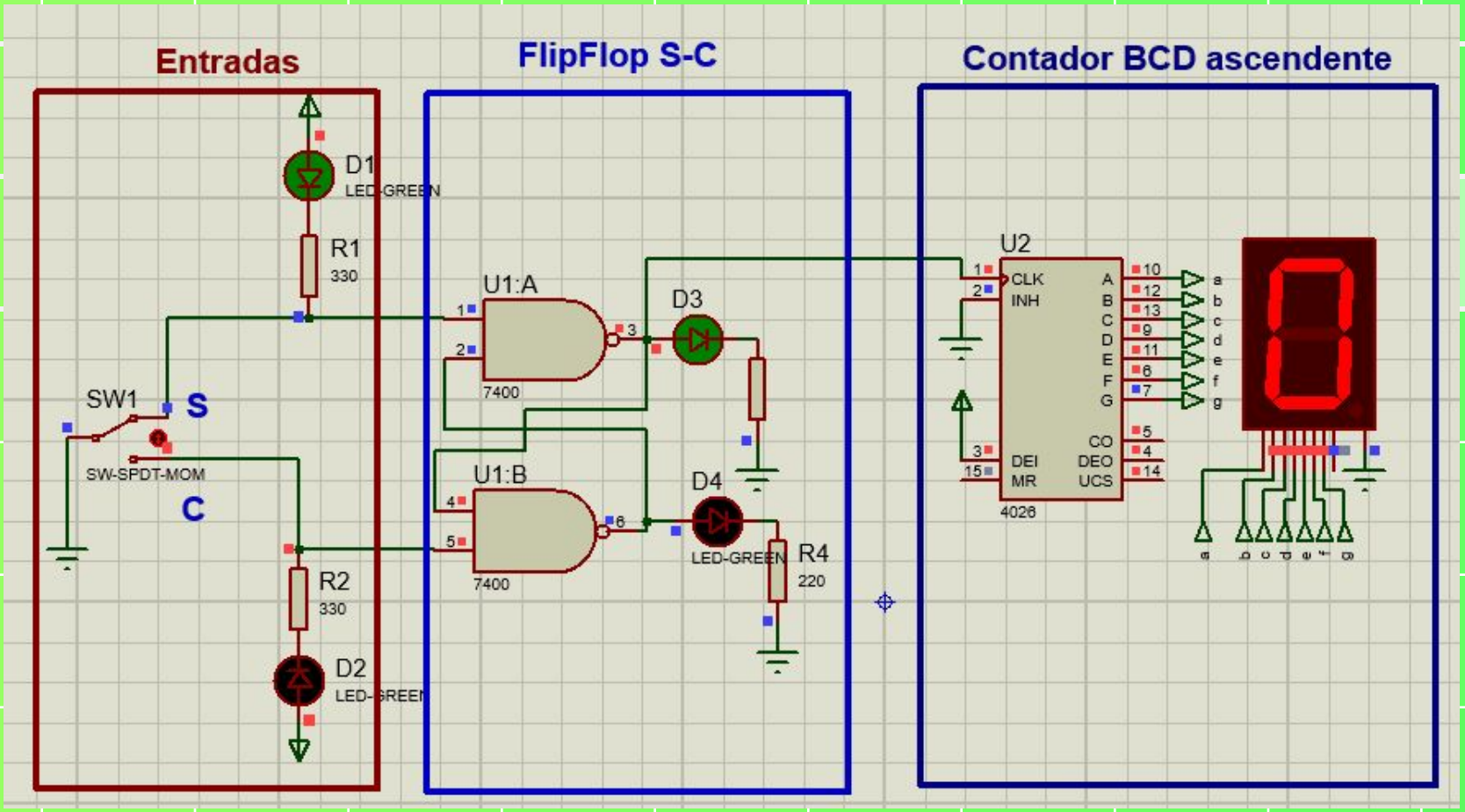
JAGG

# Tinkercad




JAGG

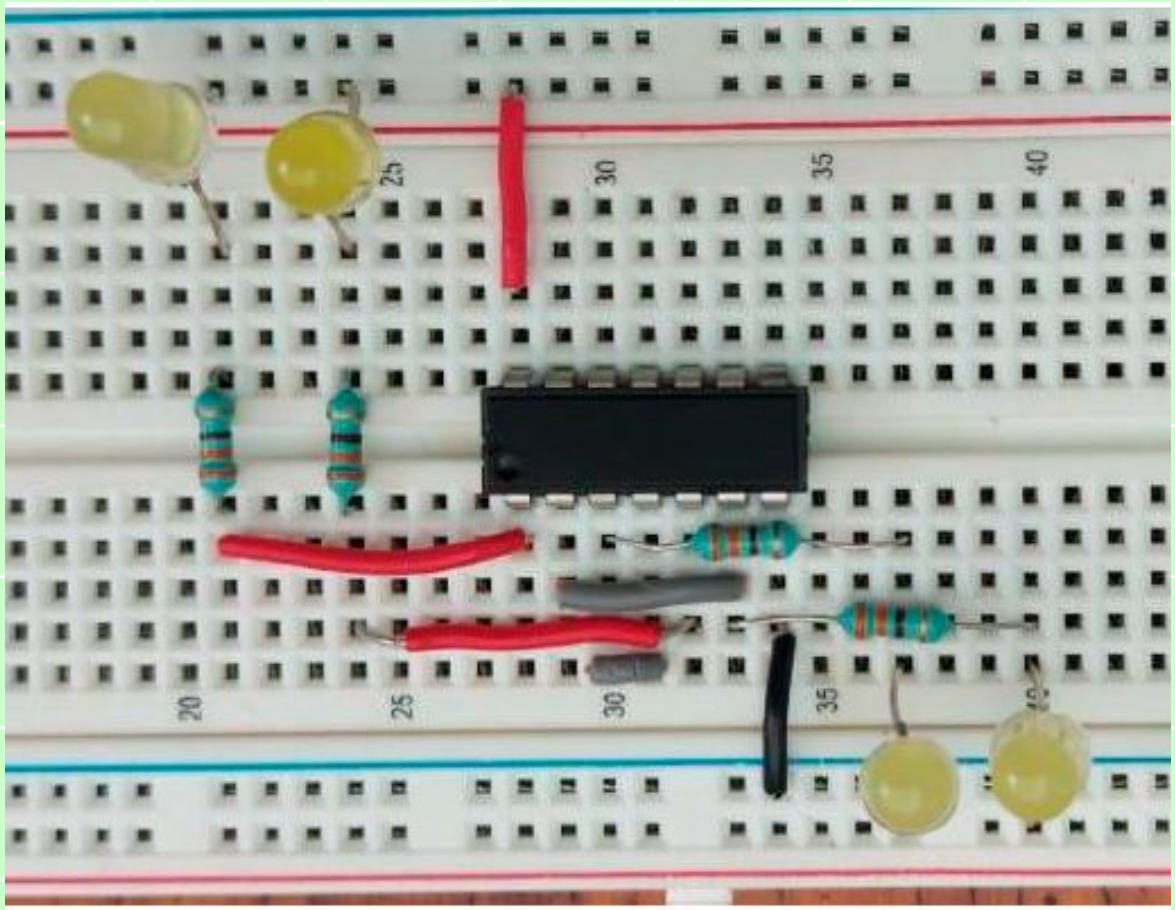
# Flip Flop SC




## PROTEUS

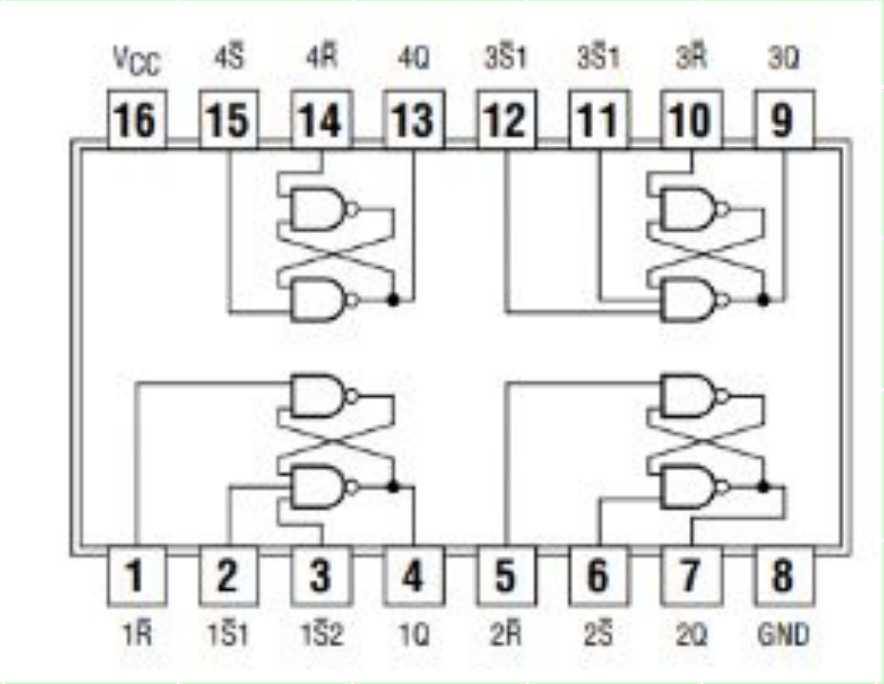
JAGG

# Flip Flop SC




JAGG

# Sn74LS279
# QUADRUPLE S-R LATCHES




**Distribución de terminales**
**PIN OUT**

*JAGG*

# Sn74279
# QUADRUPLE S-R LATCHES

| INPUTS | | OUTPUT |
|---|---|---|
| $\bar{S}$† | $\bar{R}$ | Q |
| H | H | $Q_0$ |
| L | H | H |
| H | L | L |
| L | L | H‡ |

H = high level  L = low level

†For latches with double $\bar{S}$ inputs:

$Q_0$ = the level of Q before the indicated input conditions were established.

‡This configuration is nonstable: that is, it may not persist when the $\bar{S}$ and $\bar{R}$ inputs return to their inactive (high) level.

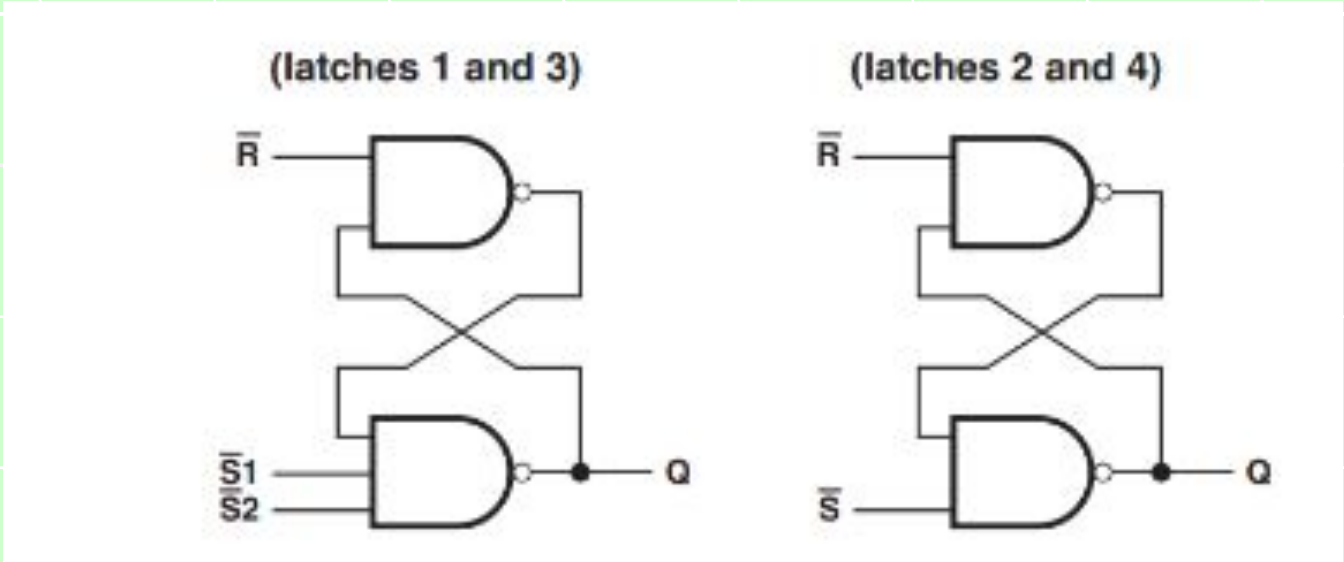
H = both $\bar{S}$ inputs high

L = one or both $\bar{S}$ inputs low

**Tabla Característica o de funcionamiento**

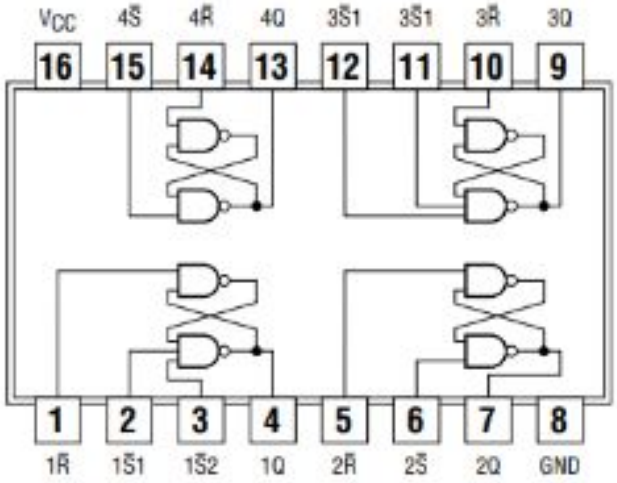
*JAGG*

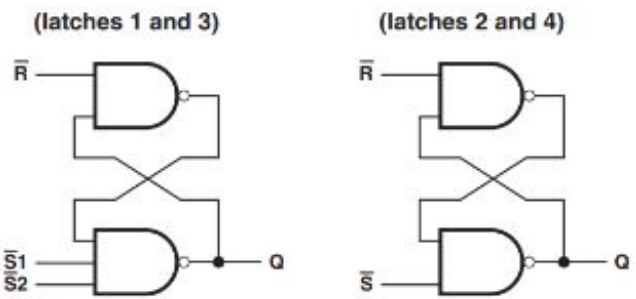
# Sn74279
# QUADRUPLE S-R LATCHES




JAGG

# Sn74279
# QUADRUPLE S-R LATCHES

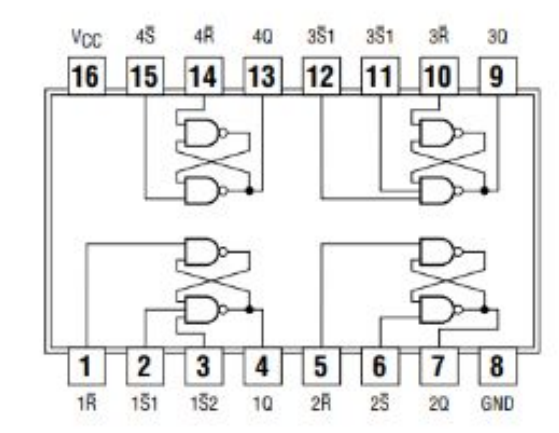


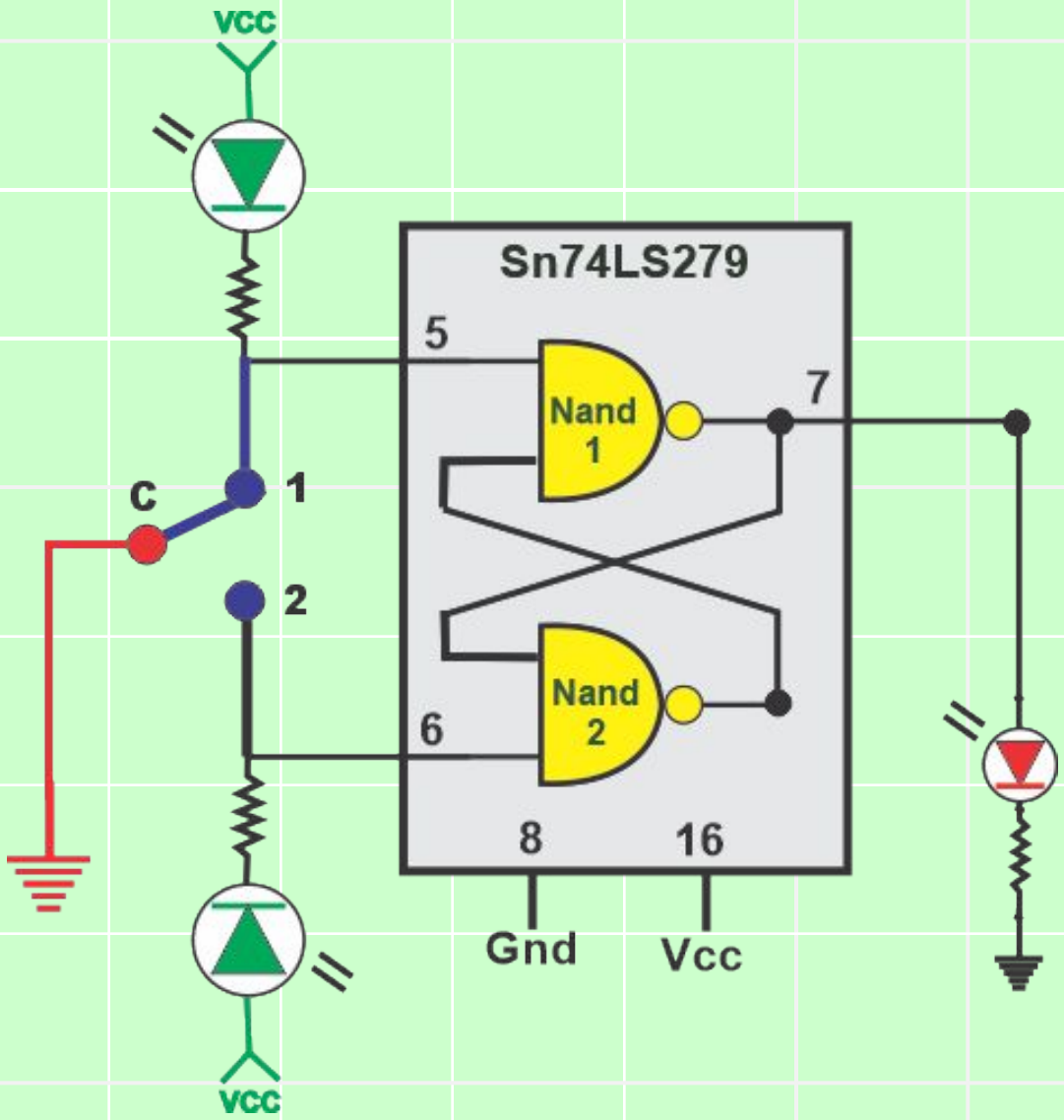




| INPUTS | | OUTPUT |
|---|---|---|
| $\bar{S}^{\dagger}$ | $\bar{R}$ | Q |
| H | H | $Q_0$ |
| L | H | H |
| H | L | L |
| L | L | H‡ |

H = high level  L = low level
†For latches with double S inputs:
$Q_0$ = the level of Q before the indicated input conditions were established.
‡This configuration is nonstable: that is, it may not persist when the $\bar{S}$ and $\bar{R}$ inputs return to their inactive (high) level.
H = both $\bar{S}$ inputs high
L = one or both $\bar{S}$ inputs low

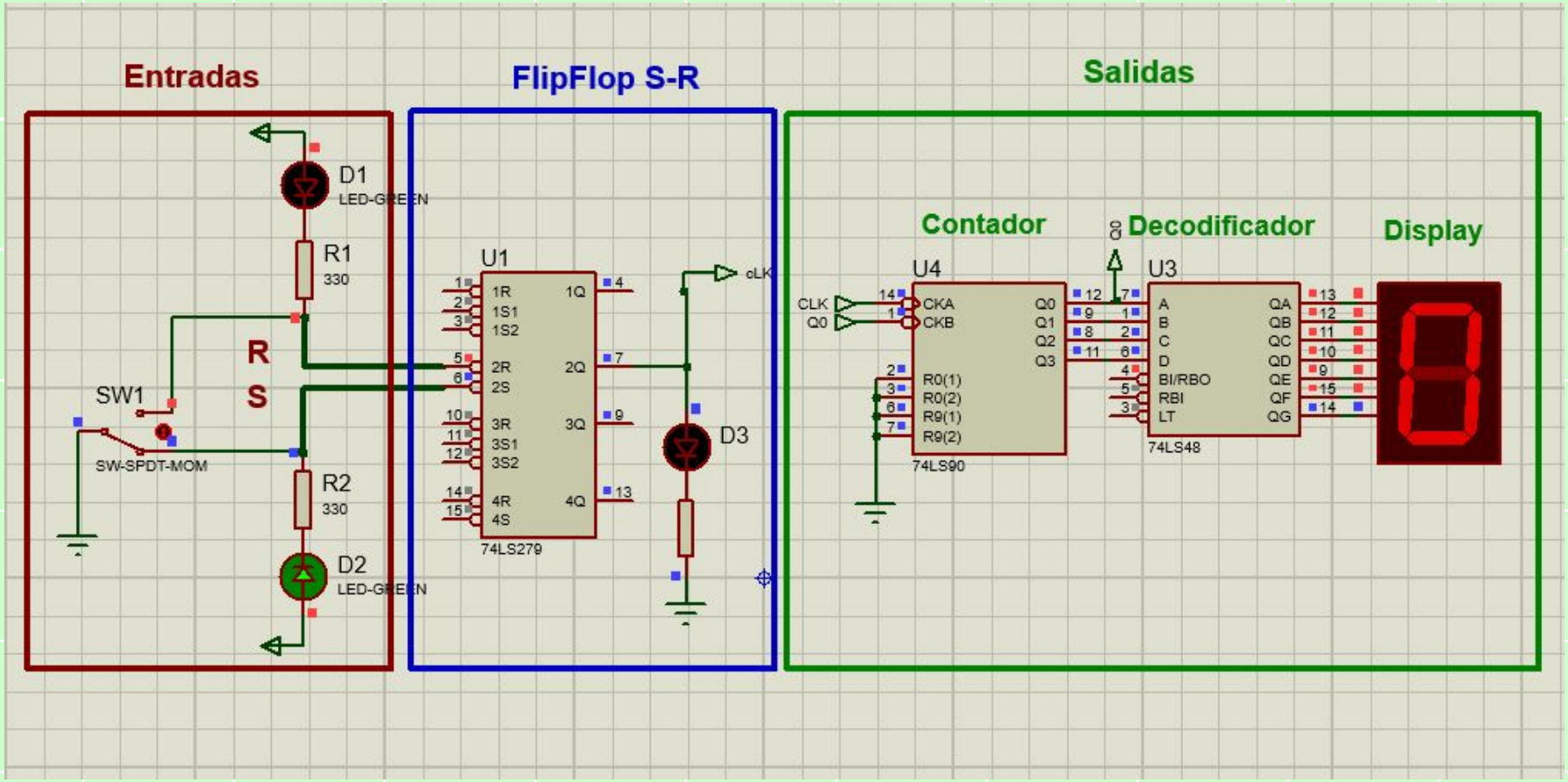
*JAGG*


VCC 4S̄ 4R̄ 4Q 3S1 3S̄1 3R̄ 3Q
16 15 14 13 12 11 10 9
1 2 3 4 5 6 7 8
1R̄ 1S̄1 1S̄2 1Q 2R̄ 2S̄ 2Q GND


VCC
Sn74LS279
5
7
Nand 1
C
1
2
Nand 2
6
8
16
Gnd
Vcc
VCC

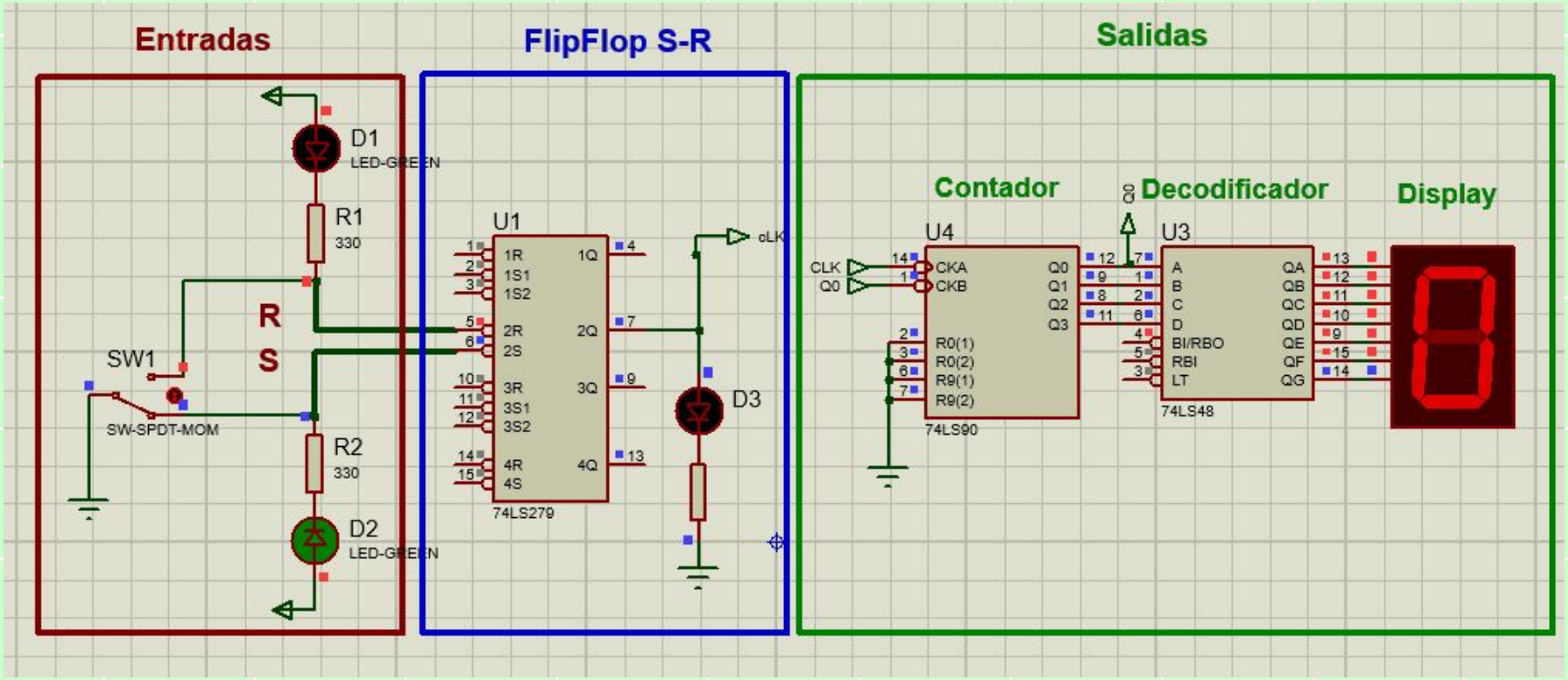
*JAGG*

# Sn74279
# QUADRUPLE S-R LATCHES




*JAGG*

# Sn74279
# QUADRUPLE S-R LATCHES

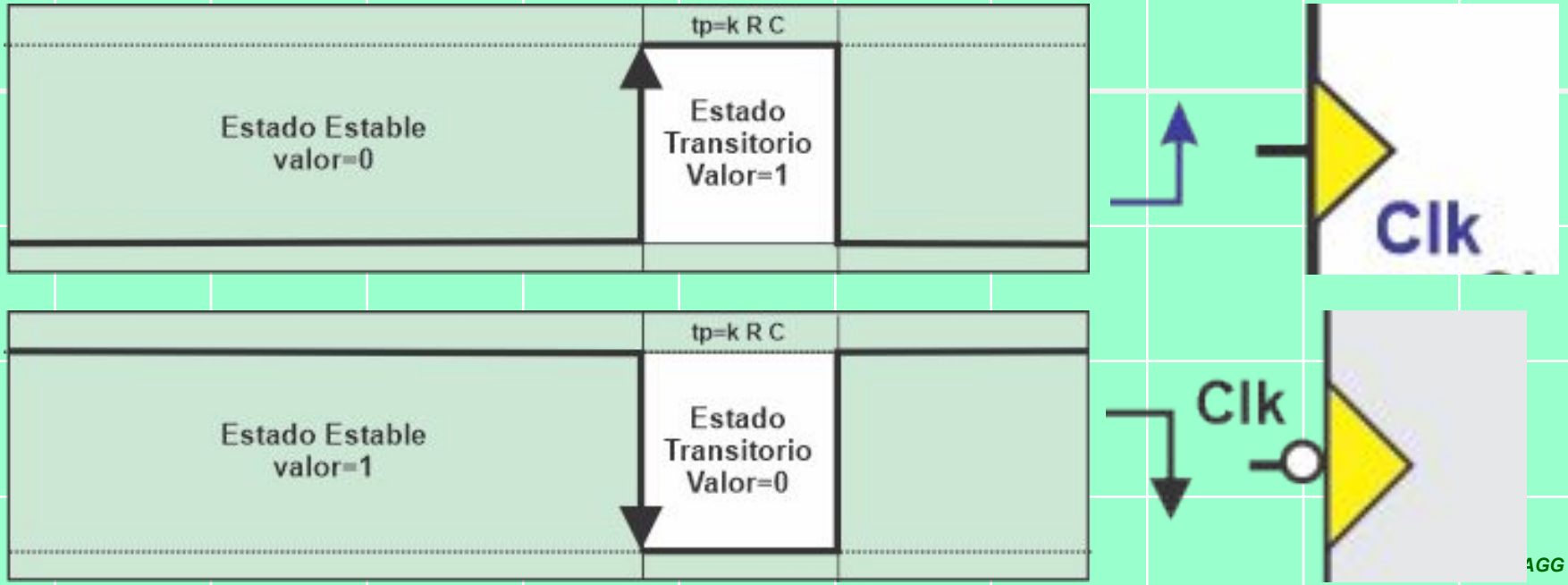



*JAGG*

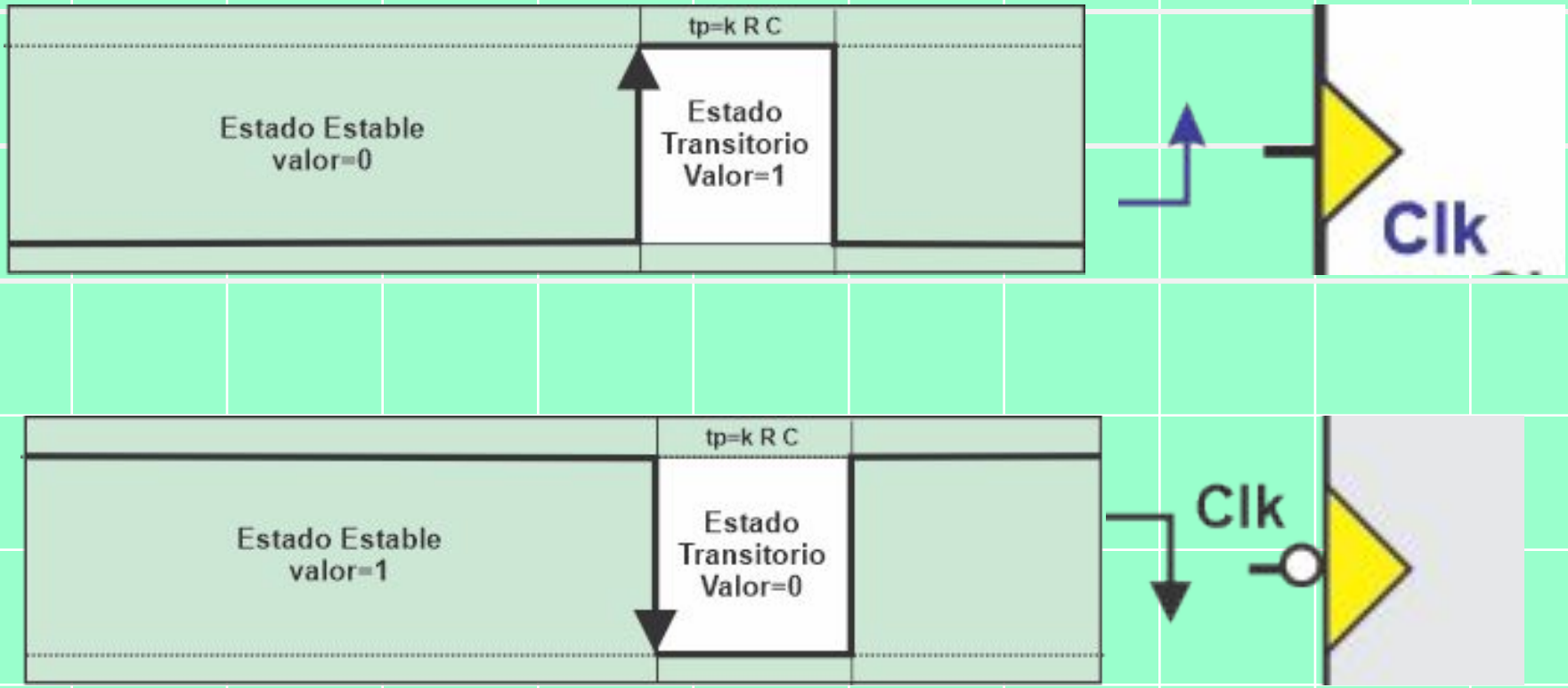
# Multivibradores Monoestables

**Un multivibrador monoestable es un circuito que permanece estable en un solo estado y, al recibir un pulso de disparo, cambia temporalmente al otro estado por un tiempo determinado antes de volver al inicial.**

**Se usa para generar un solo pulso de duración controlada**

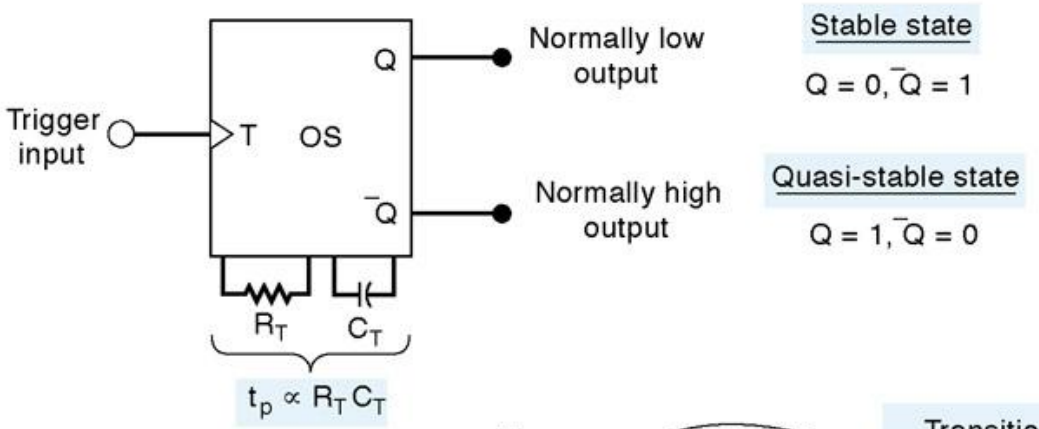


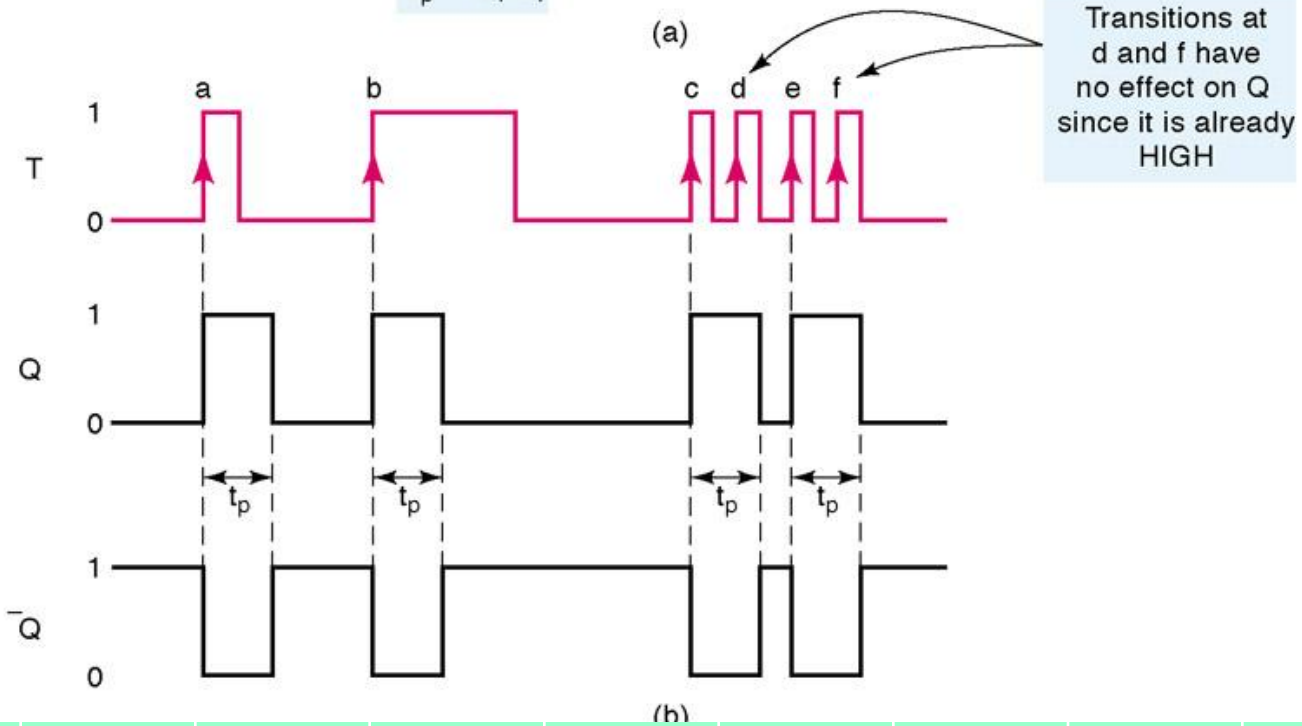
# Multivibradores Monoestables



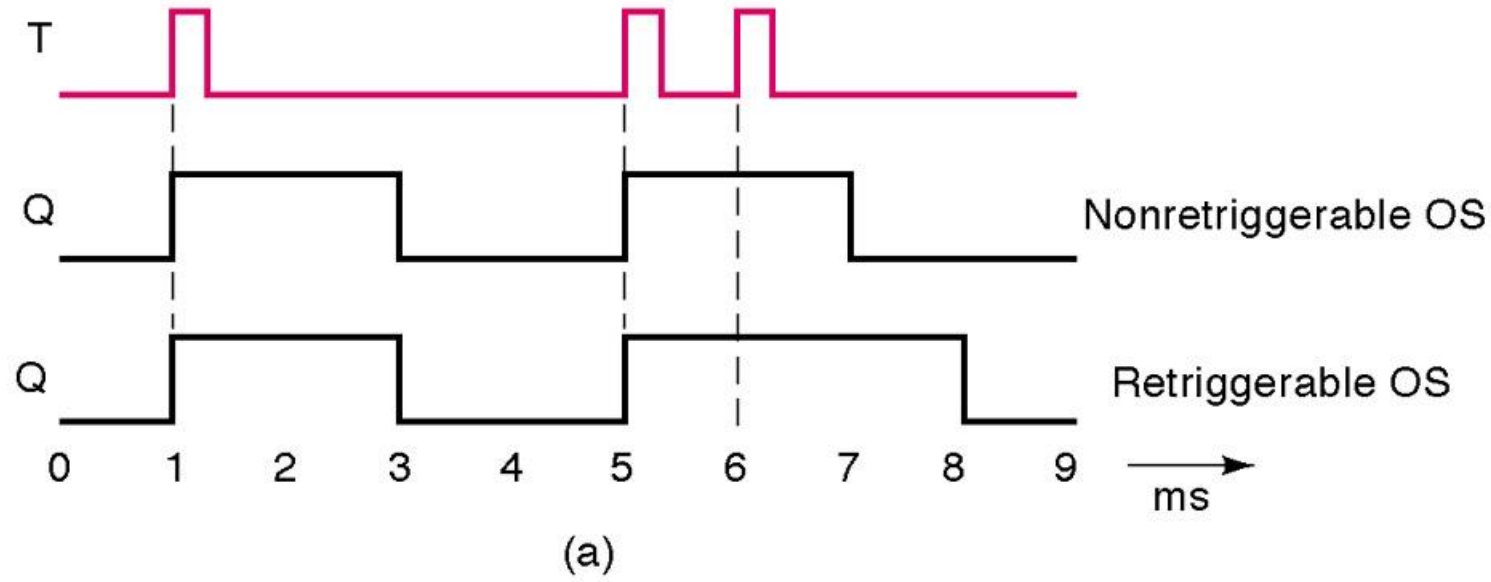

JAGG

# Multivibrador Monoestable

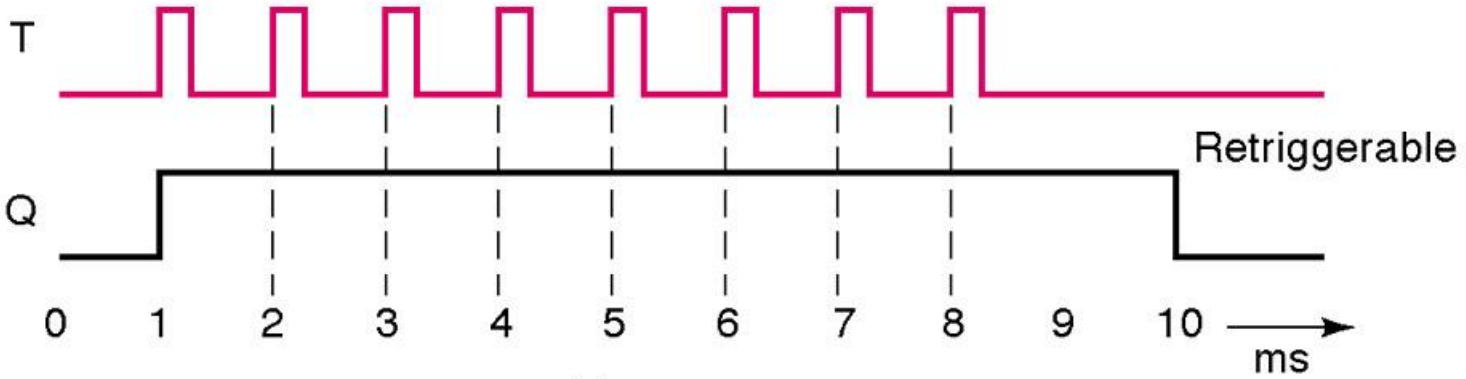



JAGG

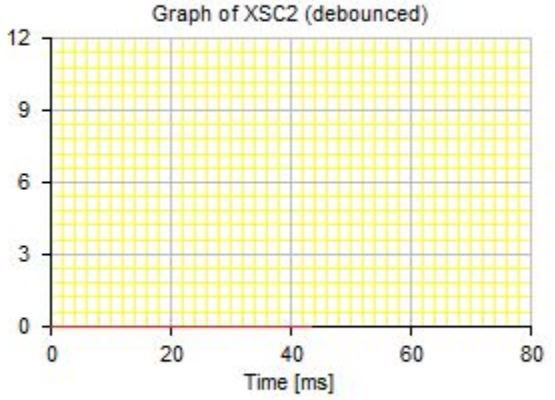

T
Q
Nonretriggerable OS
Q
Retriggerable OS
0 1 2 3 4 5 6 7 8 9
ms

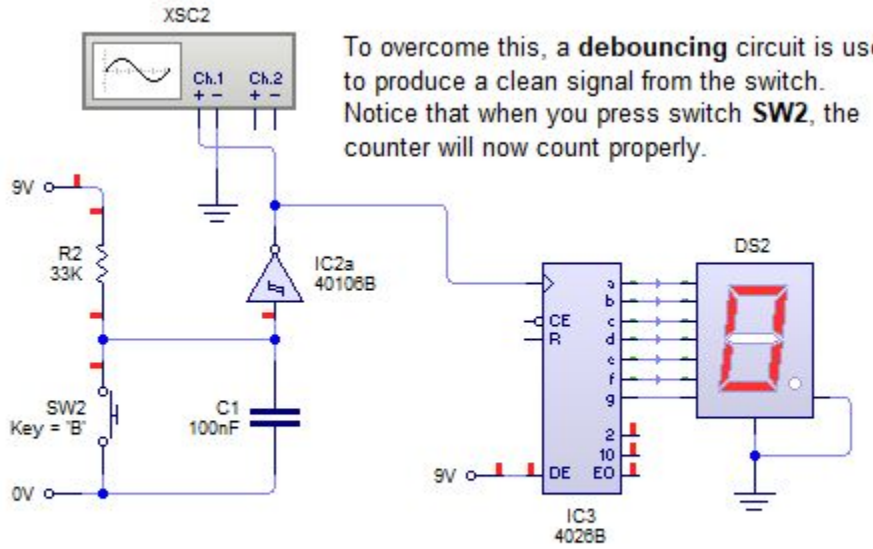
(a)


T
Retriggerable
Q
0 1 2 3 4 5 6 7 8 9 10
ms

(b)

# Monoestable Redisparable



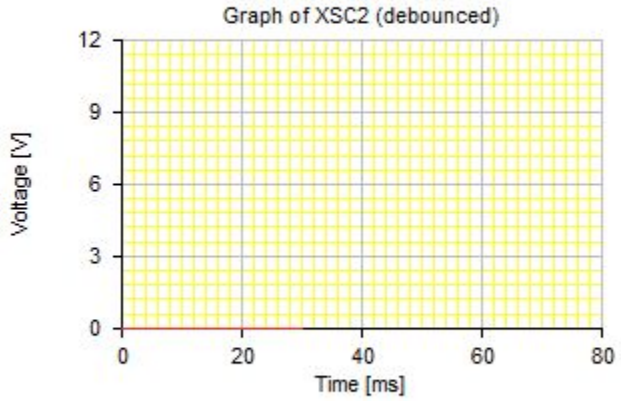

To overcome this, a **debouncing** circuit is used to produce a clean signal from the switch. Notice that when you press switch **SW2**, the counter will now count properly.

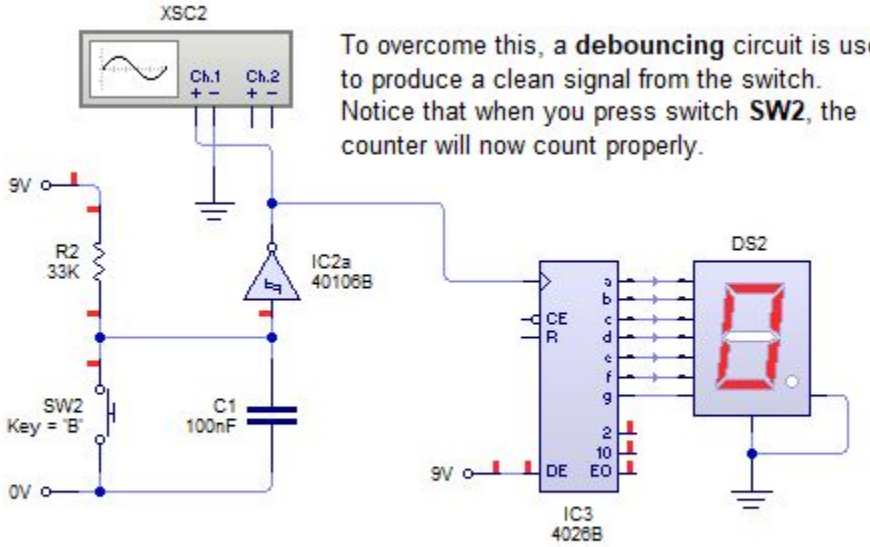
**Tip!** You can select between American and European circuit symbols by choosing **Options** from the **Tools** menu and clicking on the **Symbols** tab.

# LiveWire

*JAGG*

# Monoestable redisparable




To overcome this, a **debouncing** circuit is used to produce a clean signal from the switch. Notice that when you press switch **SW2**, the counter will now count properly.

**Tip!** You can select between American and European circuit symbols by choosing **Options** from the **Tools** menu and clicking on the **Symbols** tab.

# LiveWire

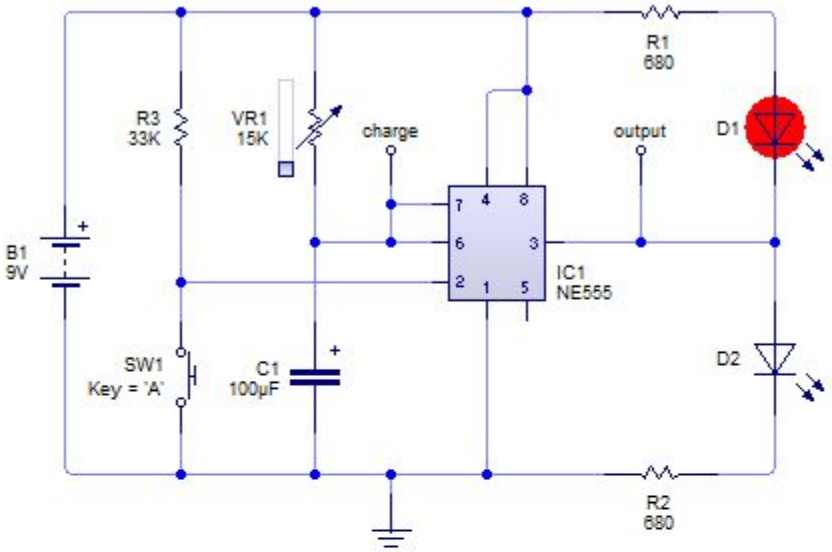
*JAGG*

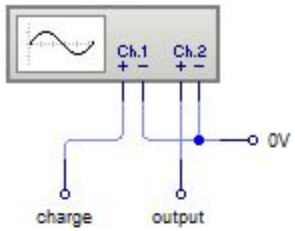
# Monoestable redisparable

Ejemplo práctico: **Secador automático de manos**
Cuando una persona coloca las manos debajo del sensor infrarrojo, el circuito activa el ventilador y la resistencia calefactora.
Este evento genera un pulso de duración determinada (por ejemplo, 3 segundos).
Si la persona mantiene las manos en el área del sensor, el circuito recibe nuevos disparos que reinician el temporizador antes de que el pulso anterior termine.
De esta manera, el secador permanece encendido mientras el sensor sigue detectando presencia.
Cuando las manos se retiran, el sensor deja de generar pulsos, el temporizador termina su ciclo y el secador se apaga automáticamente después de unos segundos.

JAGG

# Monoestable No Redisparable




**Tip!** You can select between American and European circuit symbols by choosing **Options** from the **Tools** menu and clicking on the **Symbols** tab.

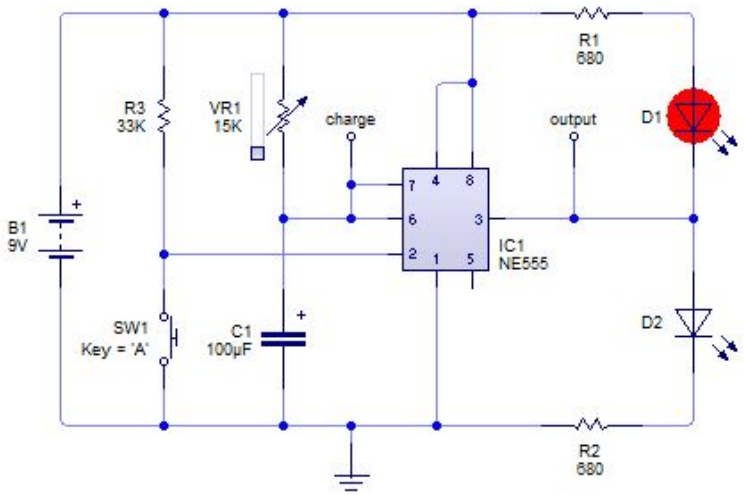
**555 Monostable**—Uses a 555 timer in its monostable mode to provide a delayed output.

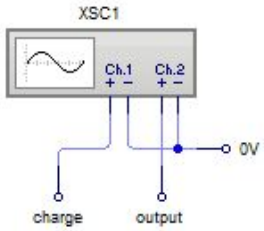
Press **SW1** to start the timer. The length of the delay can be changed by adjusting **VR1**.


## LiveWire

*JAGG*

# Monoestable No Redisparable

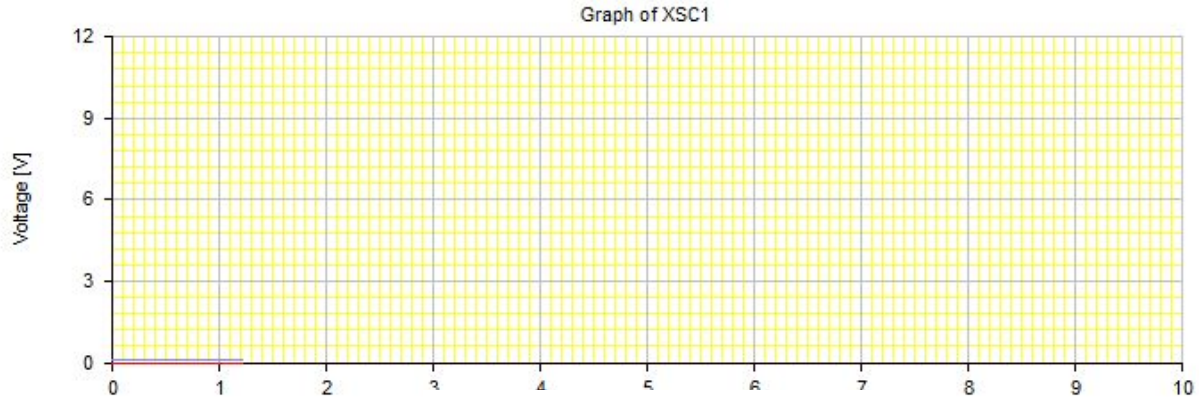



**Tip!** You can select between American and European circuit symbols by choosing **Options** from the **Tools** menu and clicking on the **Symbols** tab.

**555 Monostable**—Uses a 555 timer in its monostable mode to provide a delayed output.

Press **SW1** to start the timer. The length of the delay can be changed by adjusting **VR1**.




# LiveWire

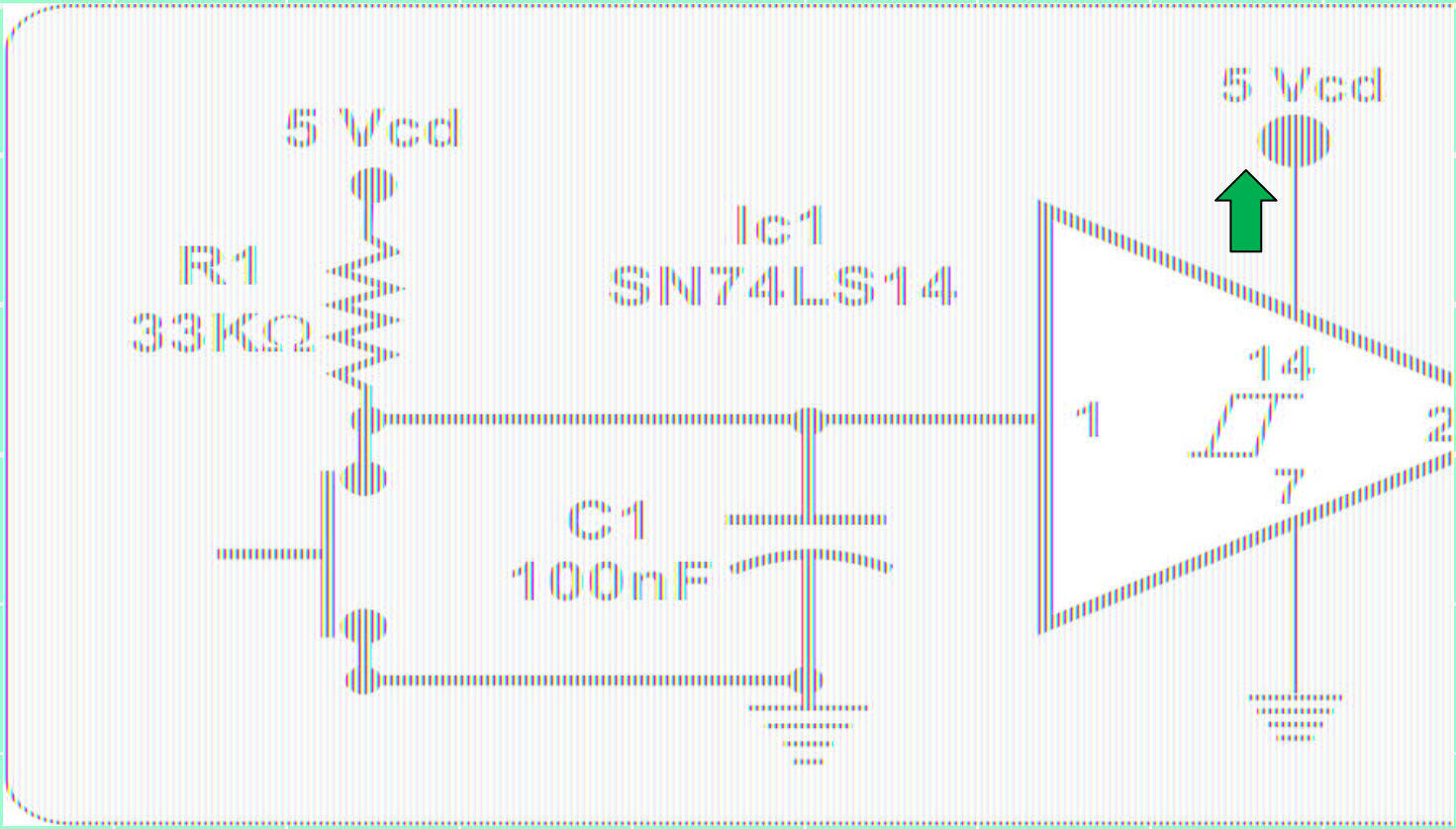
*JAGG*

# Monoestable No Redisparable

Aplicación práctica: **dispensador automático de papel**
En un dispensador automático de papel, el usuario pasa la mano frente a un sensor infrarrojo (IR).
Este sensor detecta el movimiento y envía un pulso de activación al circuito monoestable.
El multivibrador monoestable genera un pulso de salida con una duración predeterminada, por ejemplo 2 segundos, suficiente para que el motor gire y entregue una hoja de papel.
Durante esos 2 segundos, el circuito no responde a nuevas detecciones del sensor (porque es no redisparable), evitando que el motor siga girando sin control si el usuario mantiene la mano frente al sensor.

*JAGG*

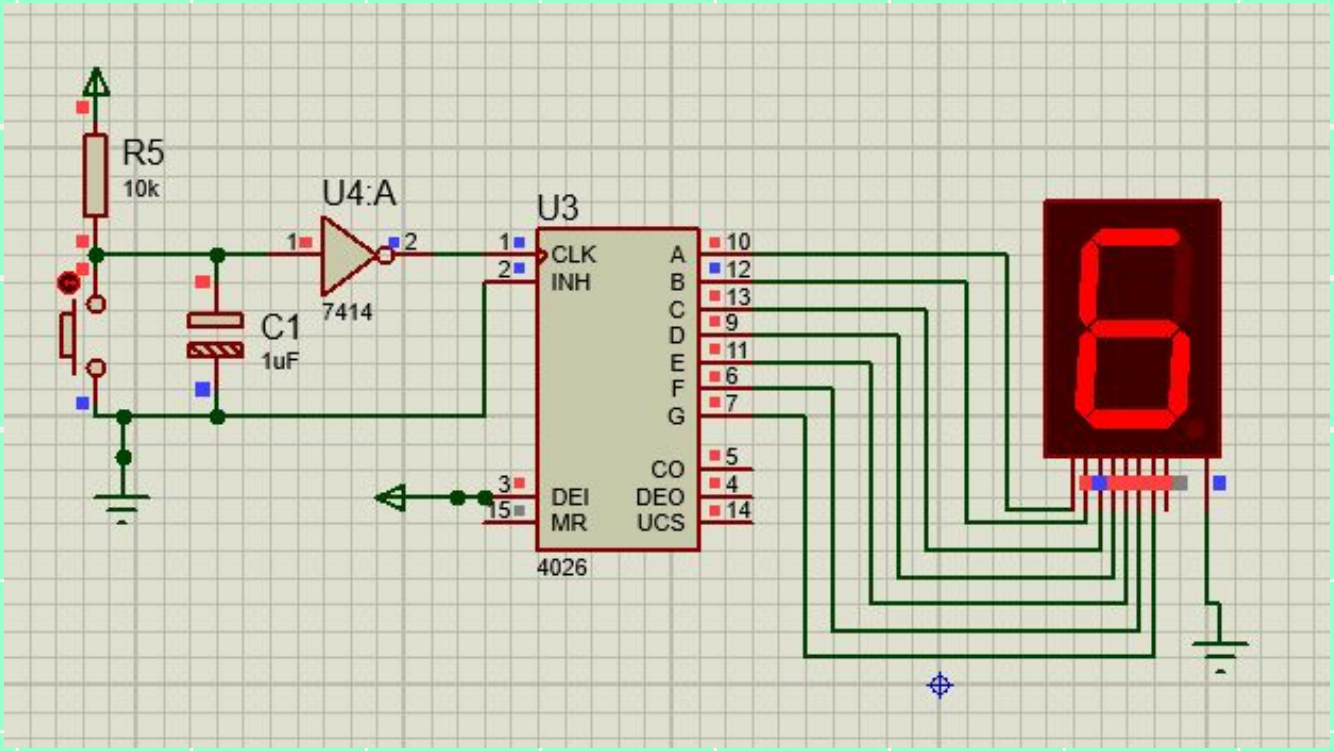
# Eliminador de rebotes




**Transición Positiva**

*JAGG*

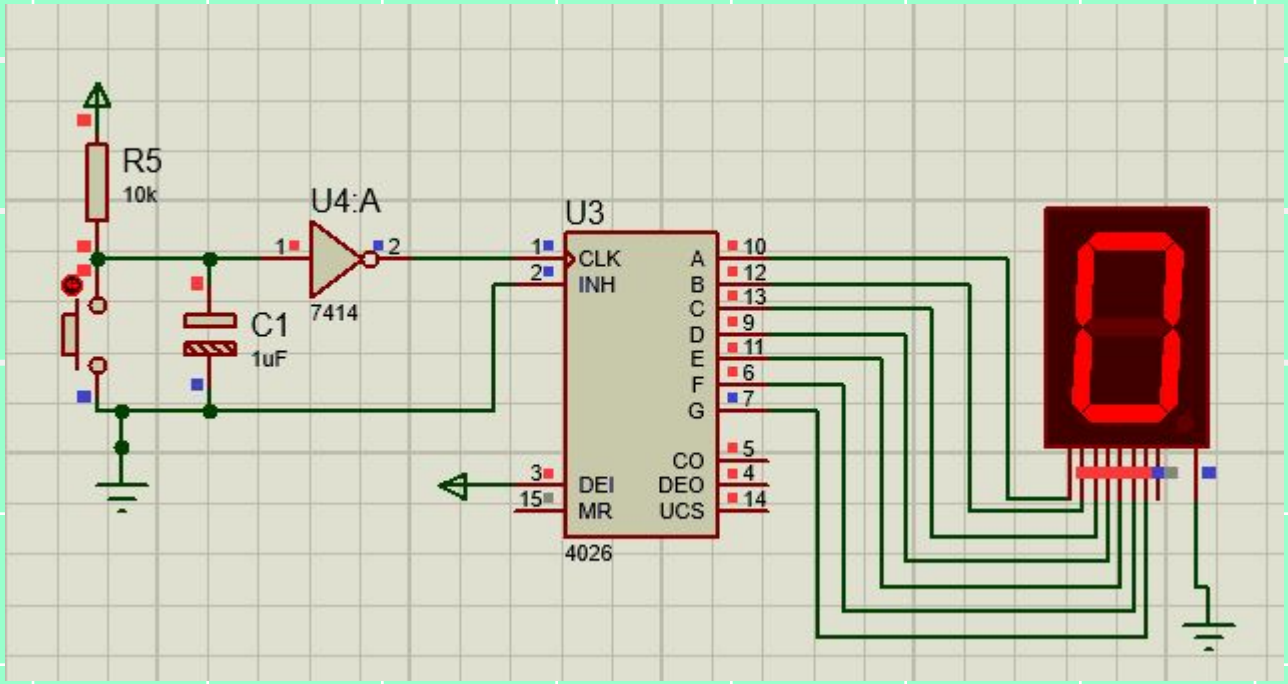
# Monoestable Transición Positiva




# PROTEUS

*JAGG*

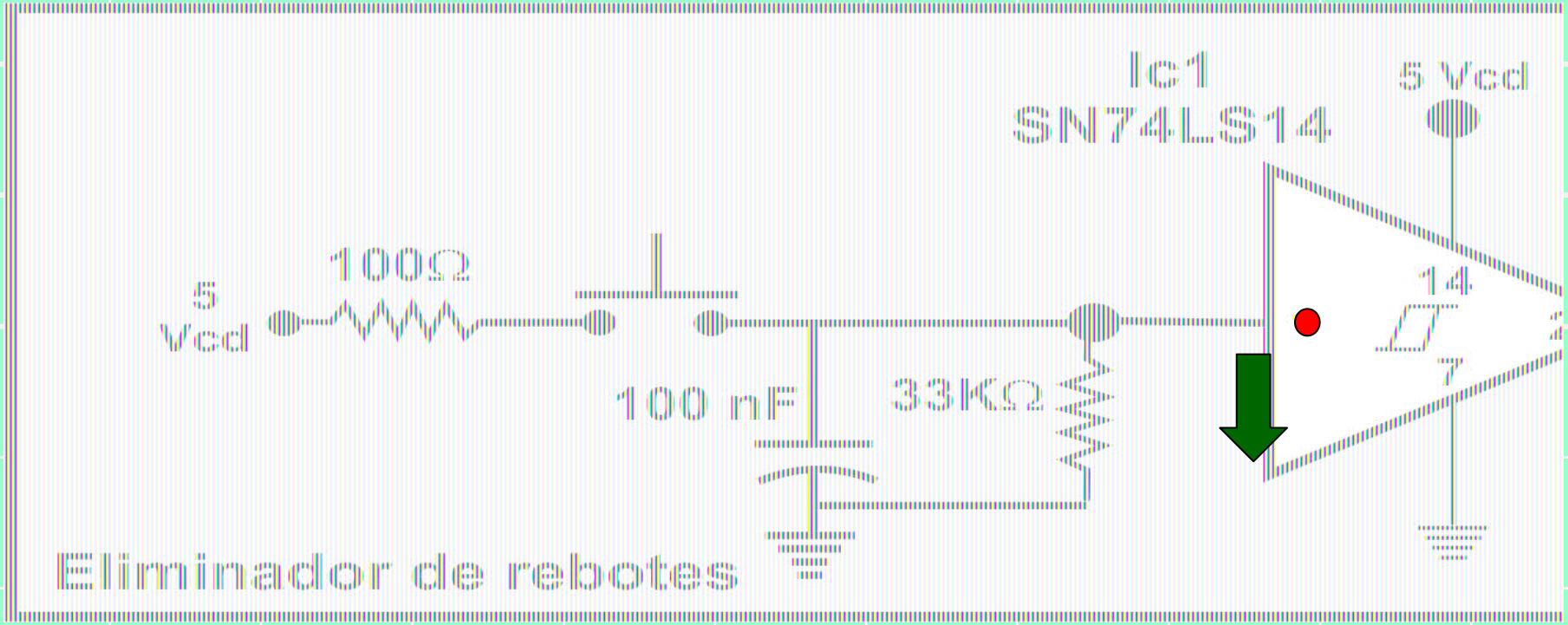
# Monoestable Transición Positiva




# PROTEUS

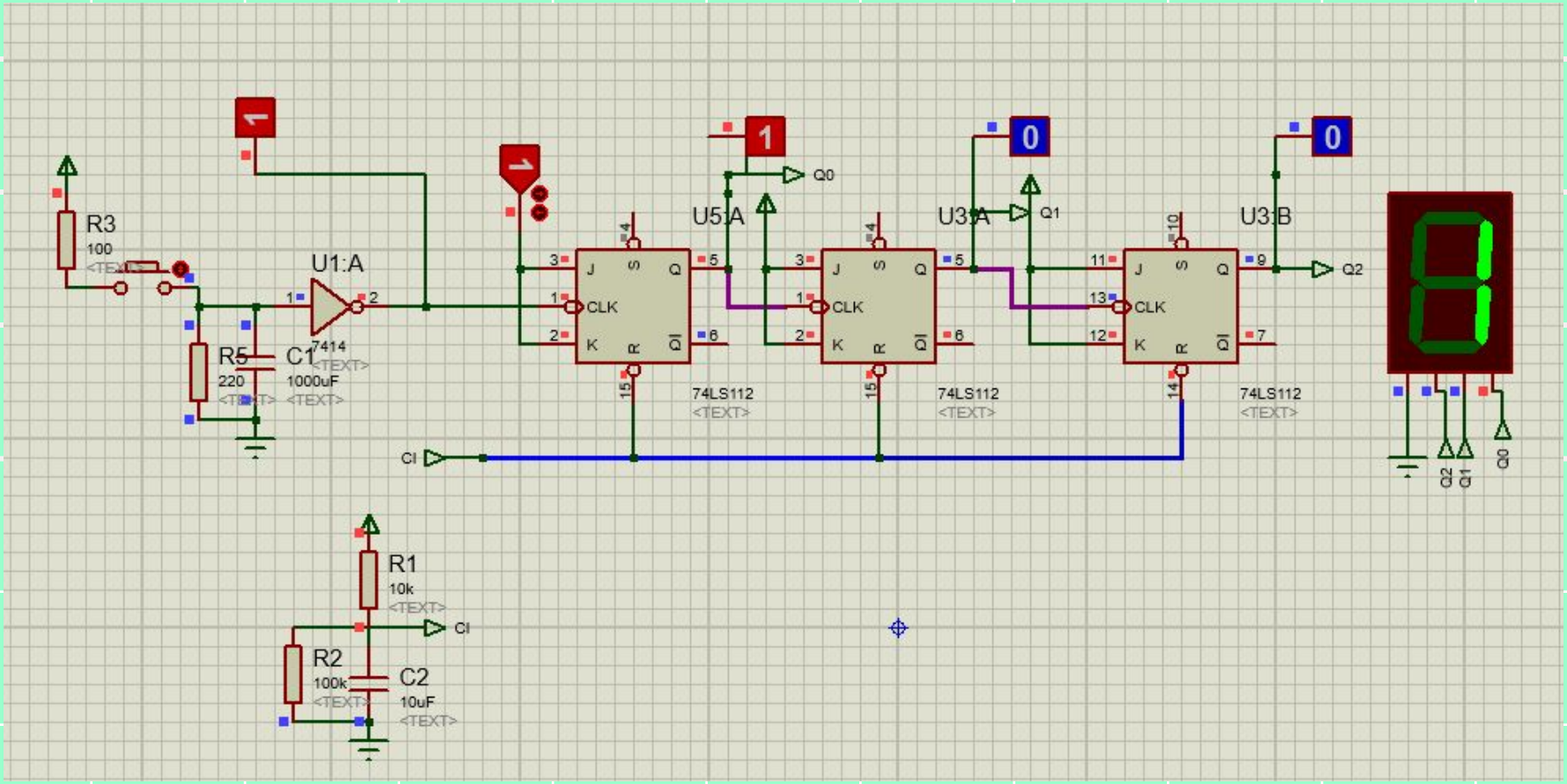
*JAGG*

# Eliminador de rebotes




**Transición Negativa**

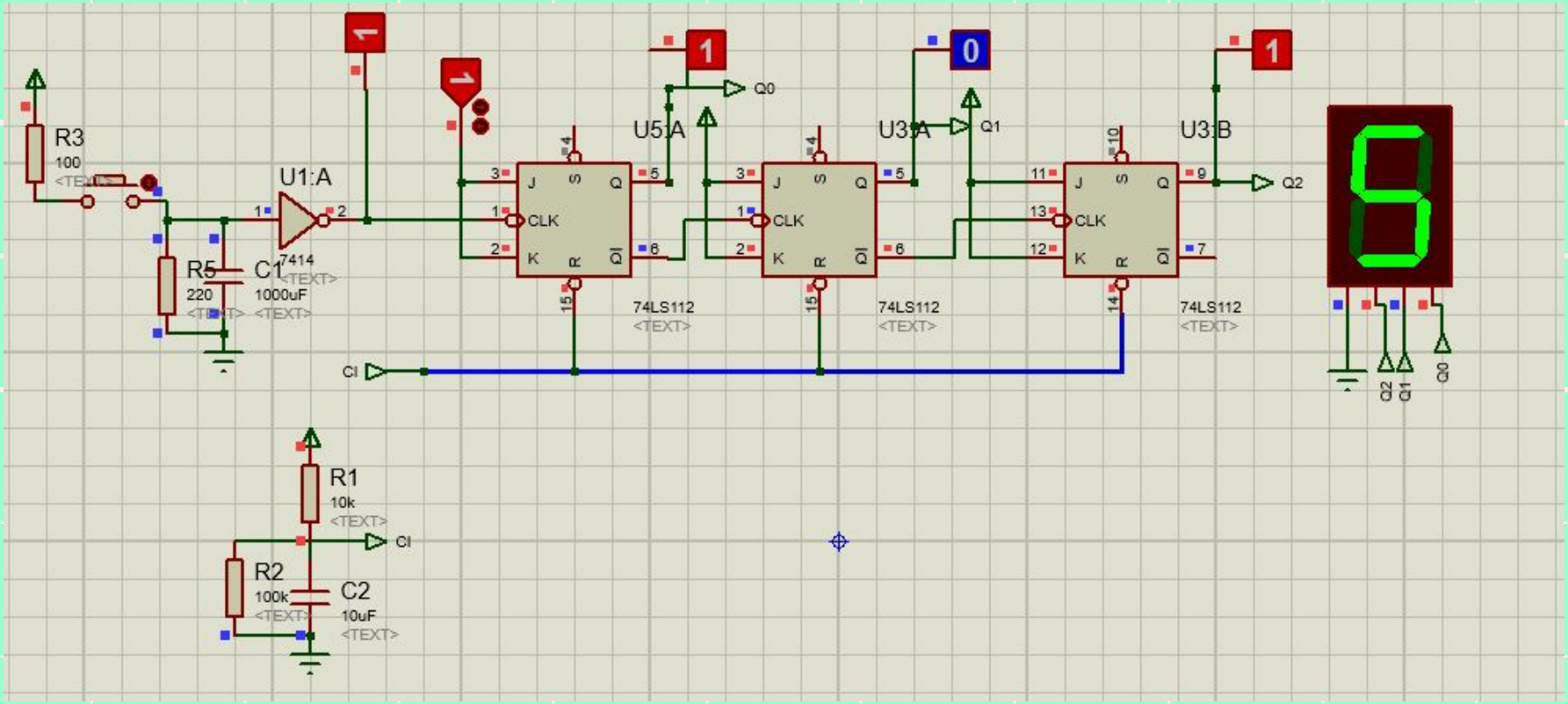
*JAGG*




# PROTEUS

**Transición Negativa**

*JAGG*




# PROTEUS

**Transición Negativa**

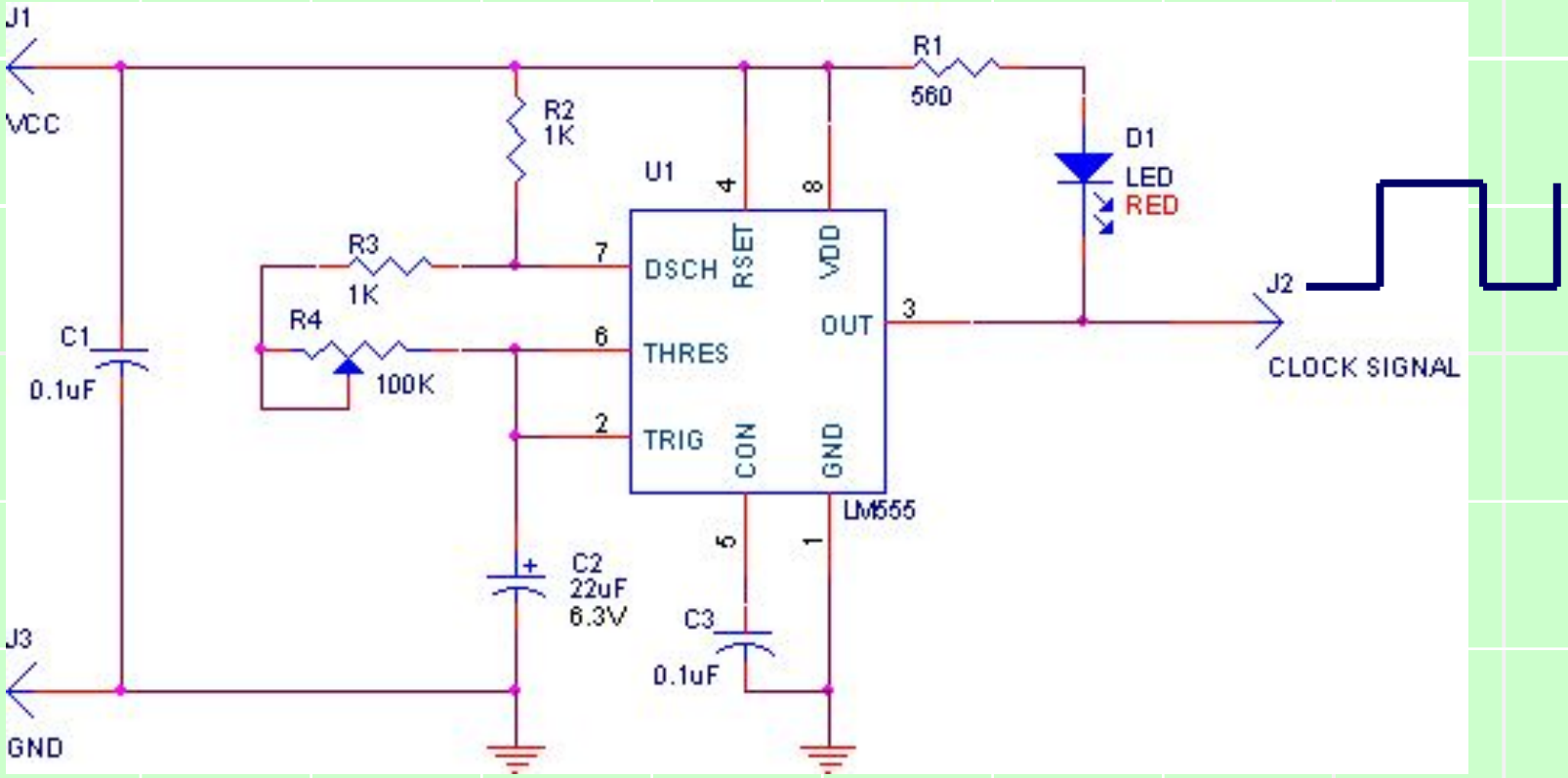
*JAGG*

# Astable

Un multivibrador astable es un circuito electrónico que oscila continuamente entre dos estados, generando una señal cuadrada periódica sin necesidad de una señal externa. Es decir, produce pulsos automáticamente, como un generador de reloj o de sincronía

*JAGG*

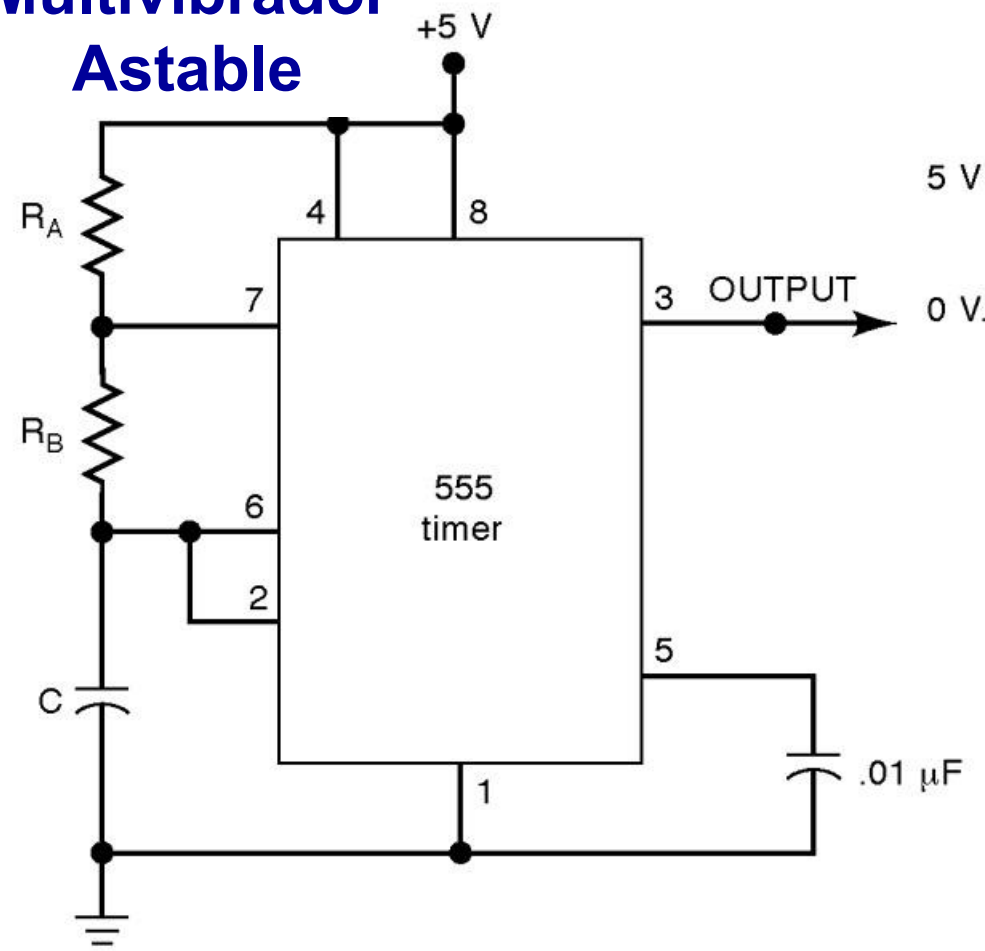
# Astable

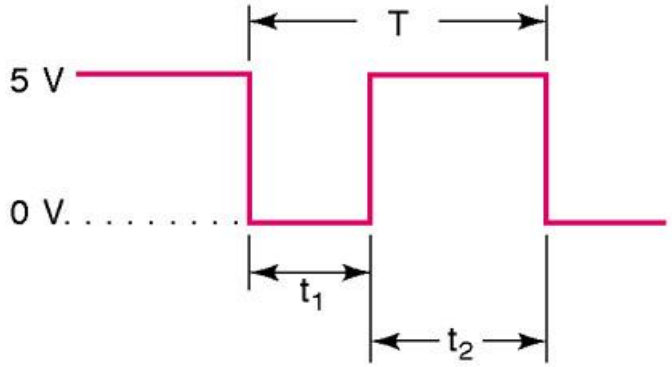



## Livewire

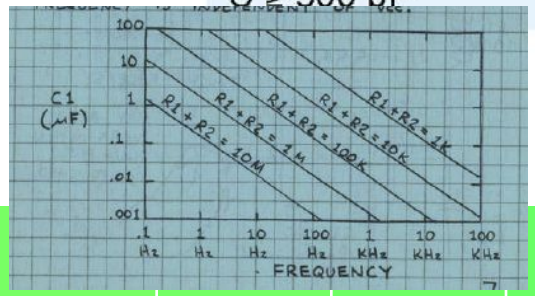
*JAGG*

# Multivibrador Astable



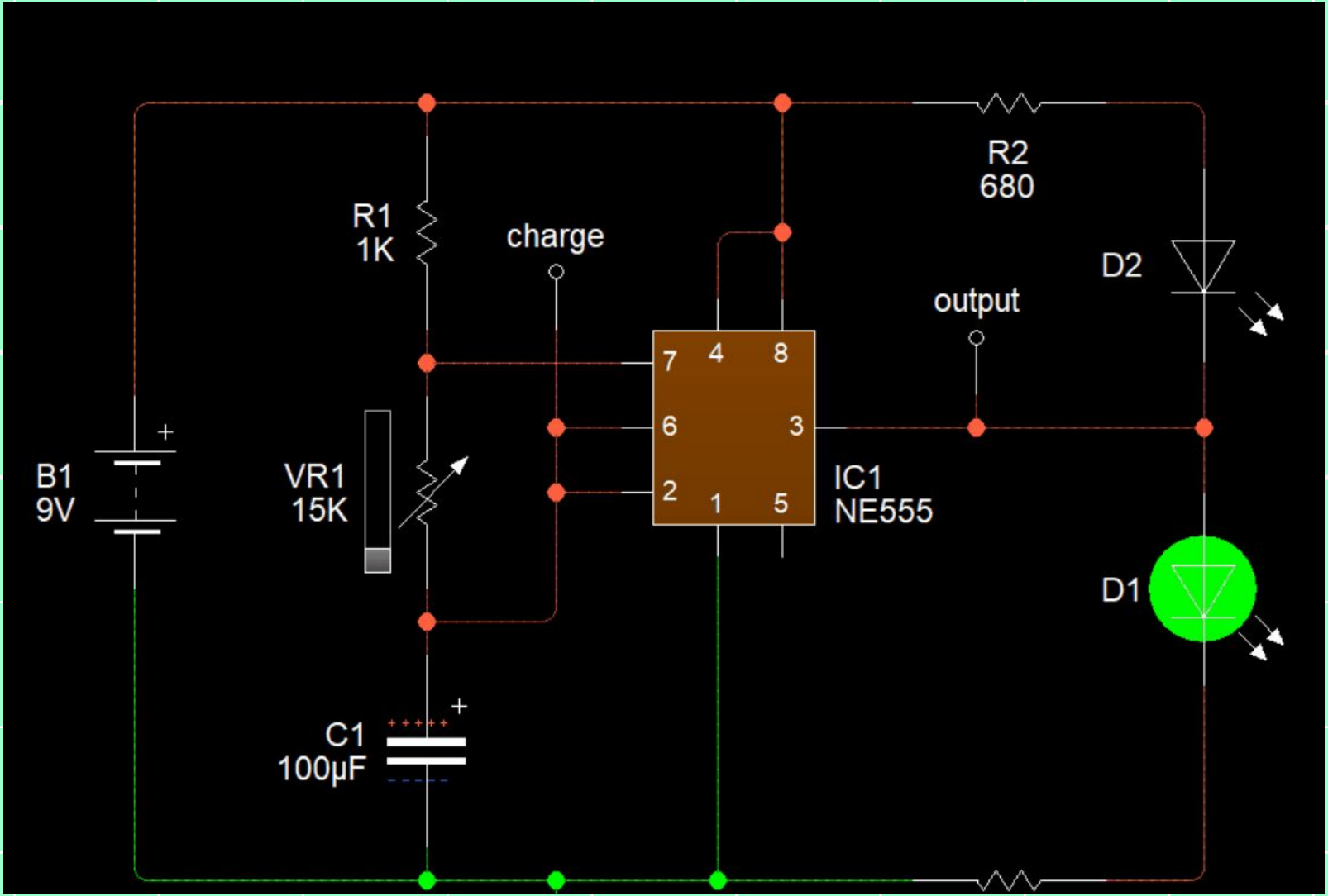




$t_1 = 0.693\ R_B C$
$t_2 = 0.693\ (R_A + R_B)C$
$T = t_1 + t_2$
frequency = 1/T
duty cycle = $t_2/T \times 100\%$
$R_A \geq 1\ k\Omega$
$R_A + R_B \leq 6.6\ M\Omega$
$C \geq 500\ pF$

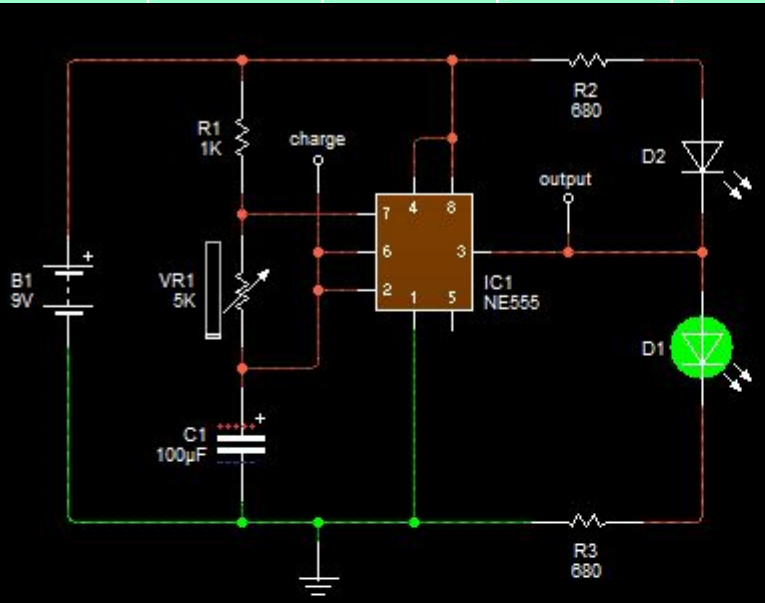


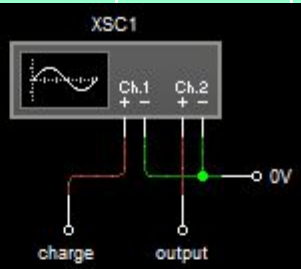

JAGG

# NE555 Astable



# Multivibrador Astable

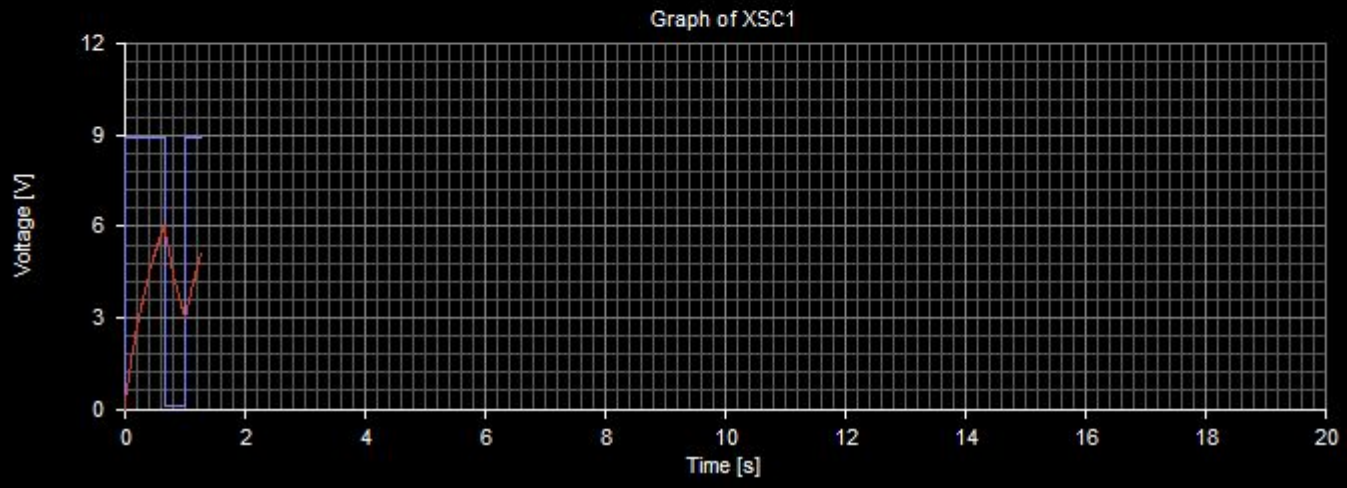



**Tip!** You can select between American and European circuit symbols by choosing **Options** from the **Tools** menu and clicking on the **Symbols** tab.

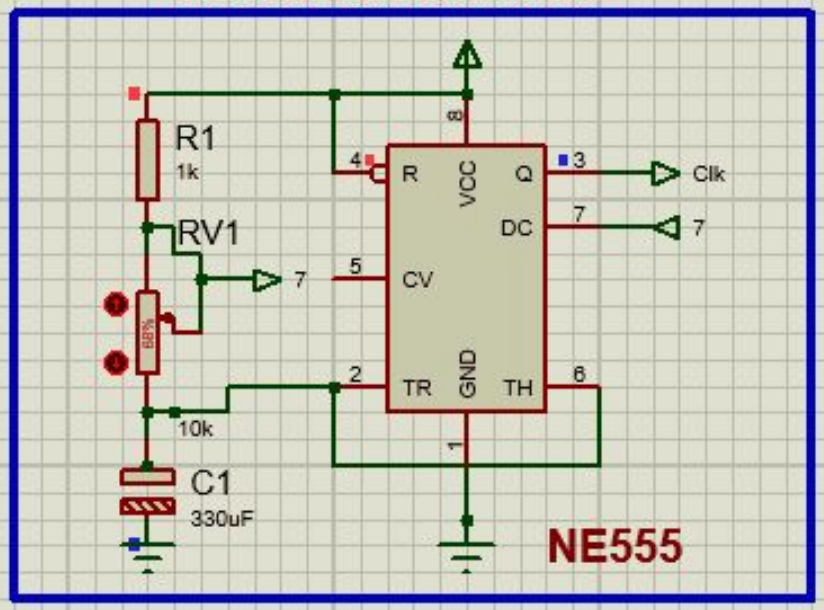
**555 Astable**—Uses a 555 timer in its astable mode to provide an output that constantly oscillates between high and low states.

The pulse rate can be changed by adjusting **VR1**.

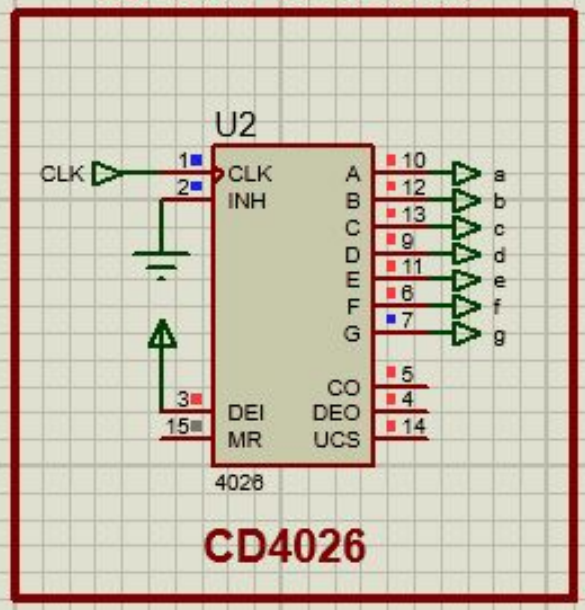



Livewire

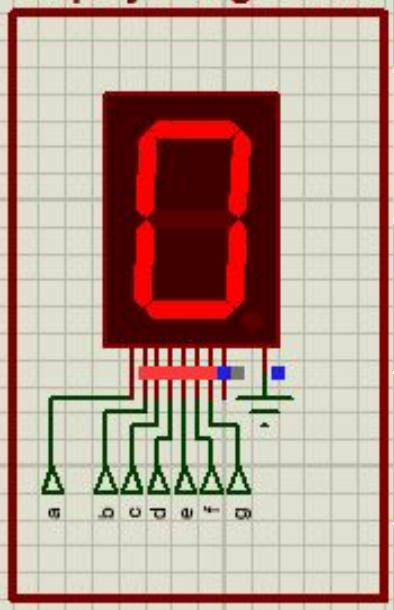
## Multivibrador Astale 555



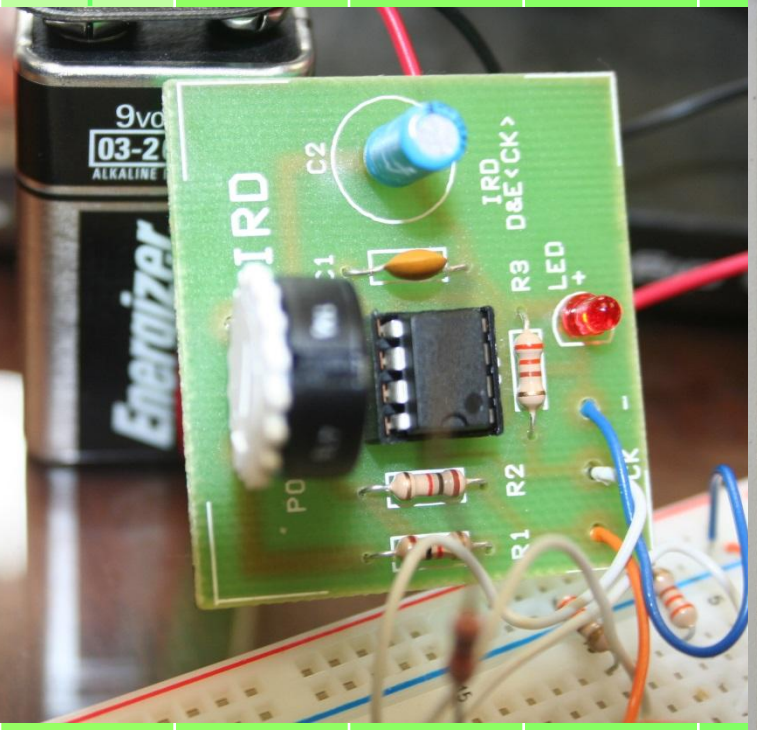

## Contador de Décadas

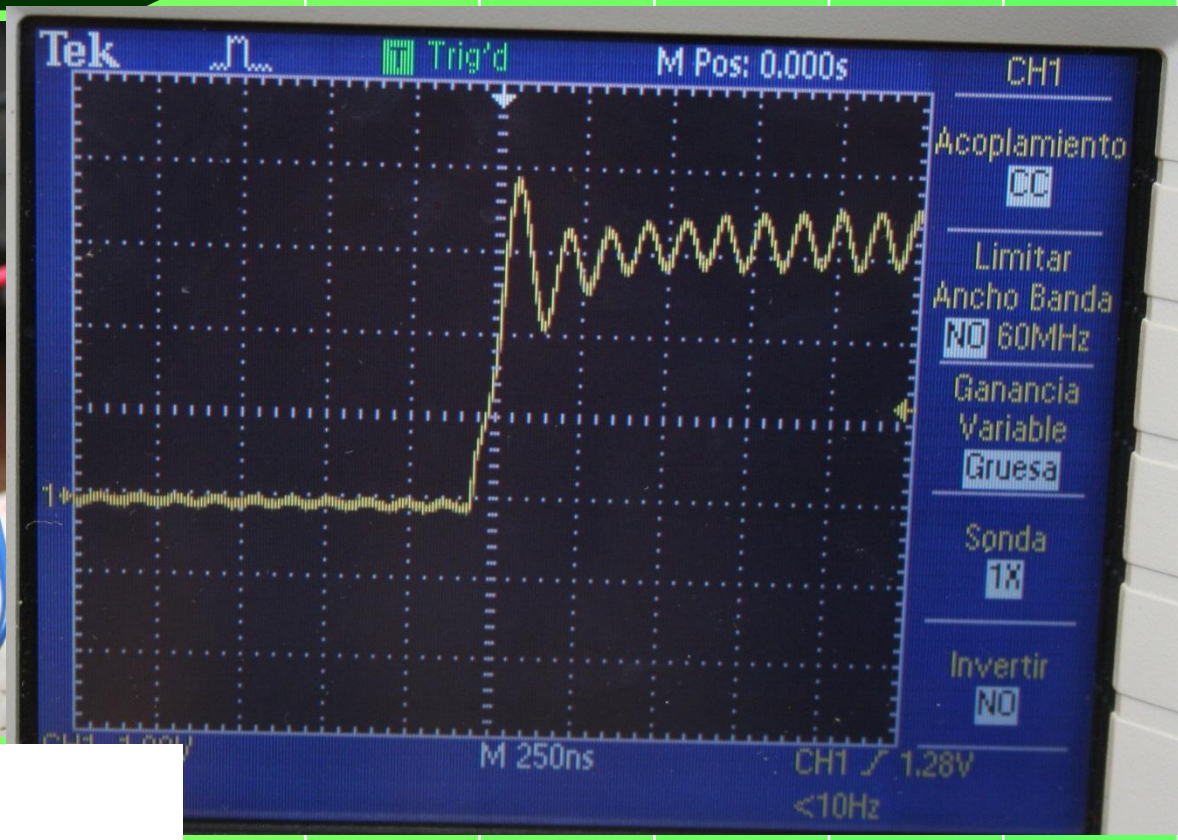


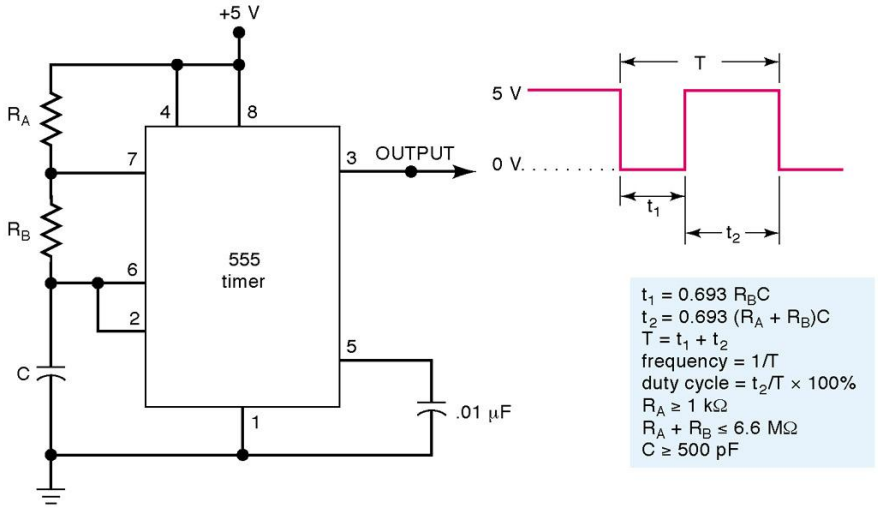

## Dispaly 7 segmentos




JAGG



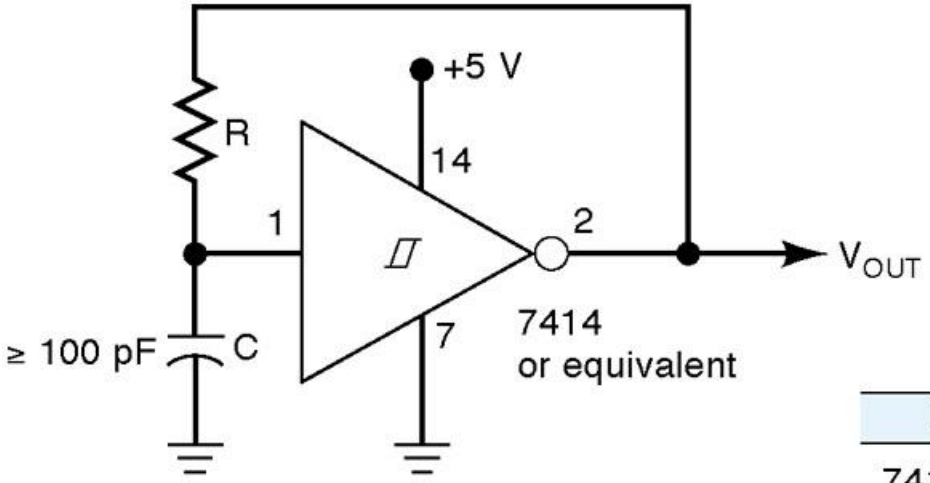

$t_1 = 0.693\ R_B C$
$t_2 = 0.693\ (R_A + R_B)C$
$T = t_1 + t_2$
frequency = 1/T
duty cycle = $t_2/T \times 100\%$
$R_A \geq 1\ k\Omega$
$R_A + R_B \leq 6.6\ M\Omega$
$C \geq 500\ pF$

**Ventaja**
**Amplio rango de frecuencias**

**Desventaja**
**Mala calidad del pulso**

*JAGG*

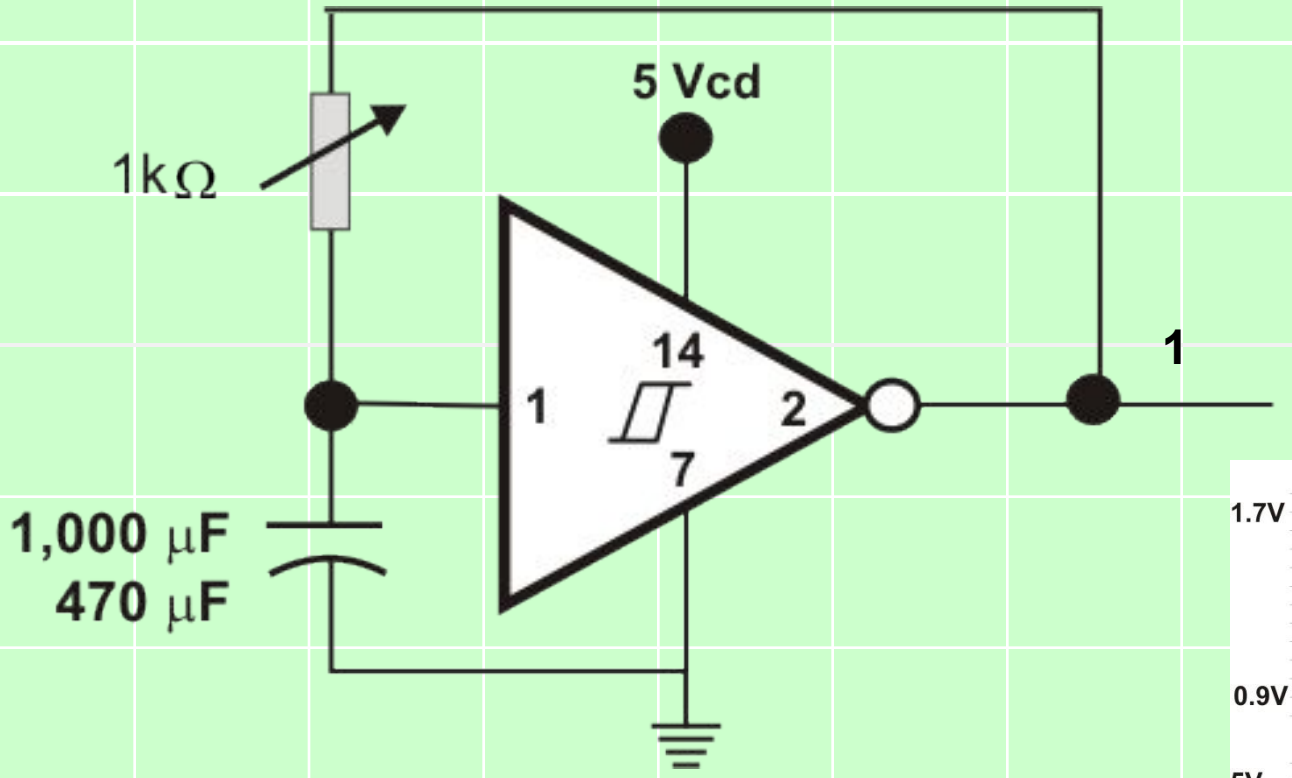
# Multivibrador Astable

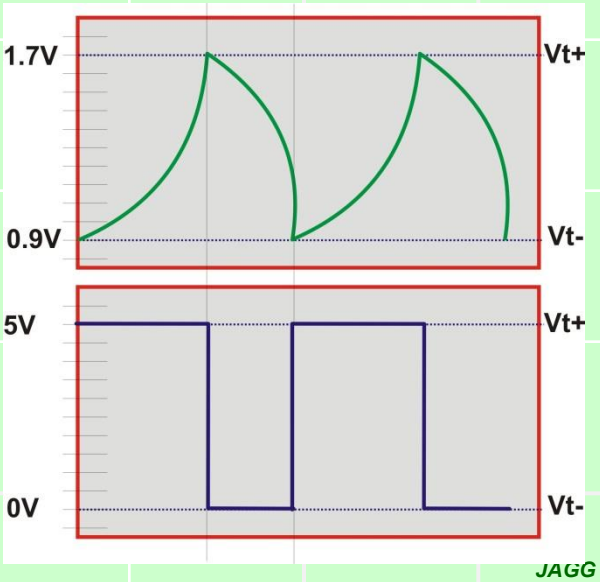


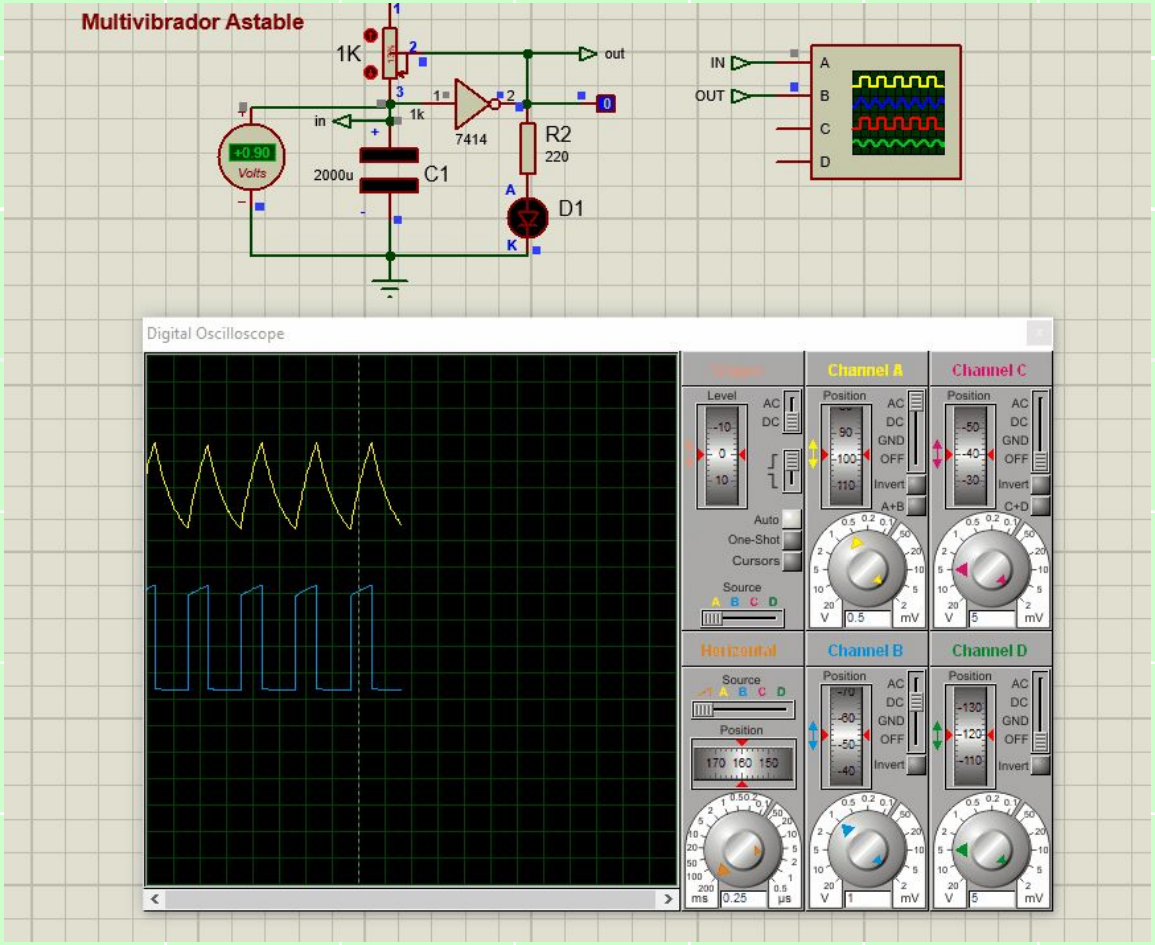


| IC | Frequency | |
|---|---|---|
| 7414 | ≈ 0.8/RC | (R ≤ 500 Ω) |
| 74LS14 | ≈ 0.8/RC | (R ≤ 2 kΩ) |
| 74HC14 | ≈ 1.2/RC | (R ≤ 10 MΩ) |

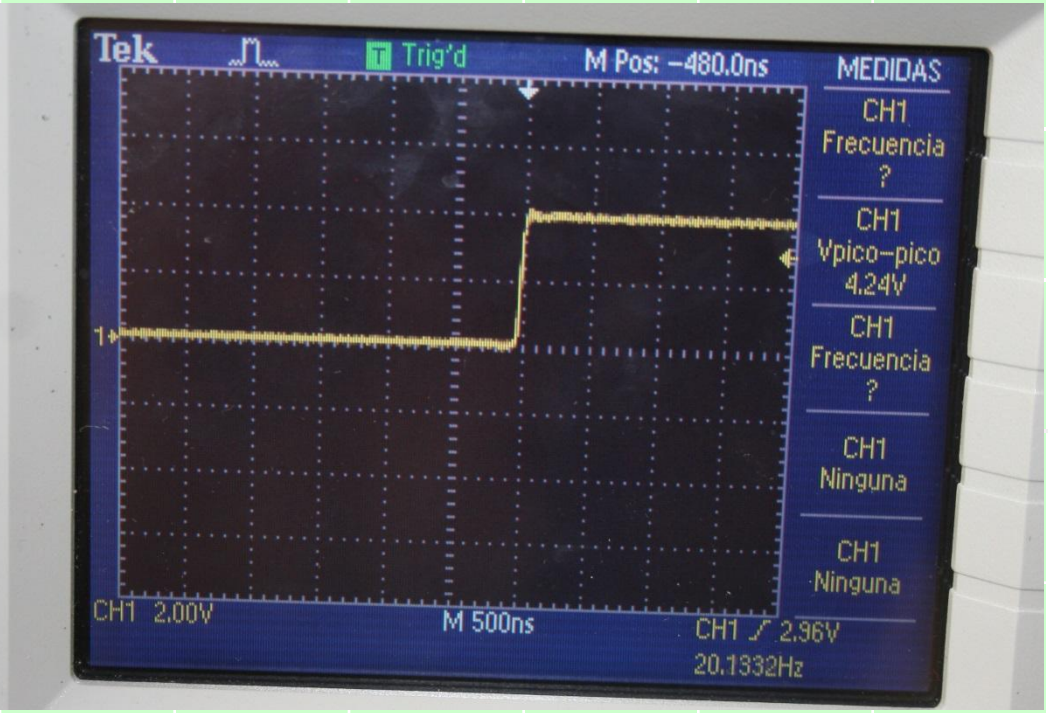
JAGG

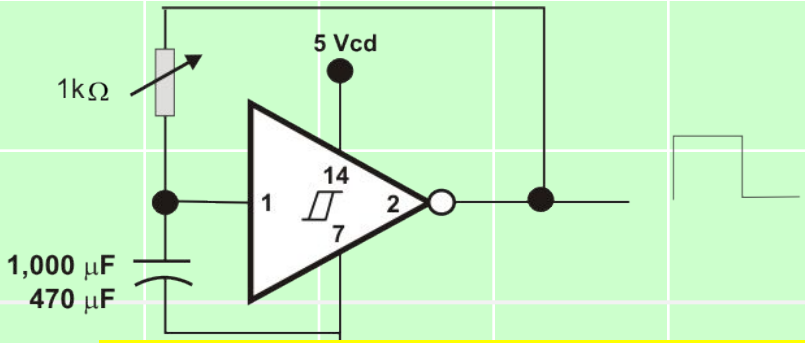

5 Vcd
1kΩ
14
1
2
7
1
1,000 µF
470 µF


1.7V
Vt+
0.9V
Vt-
5V
Vt+
0V
Vt-
JAGG

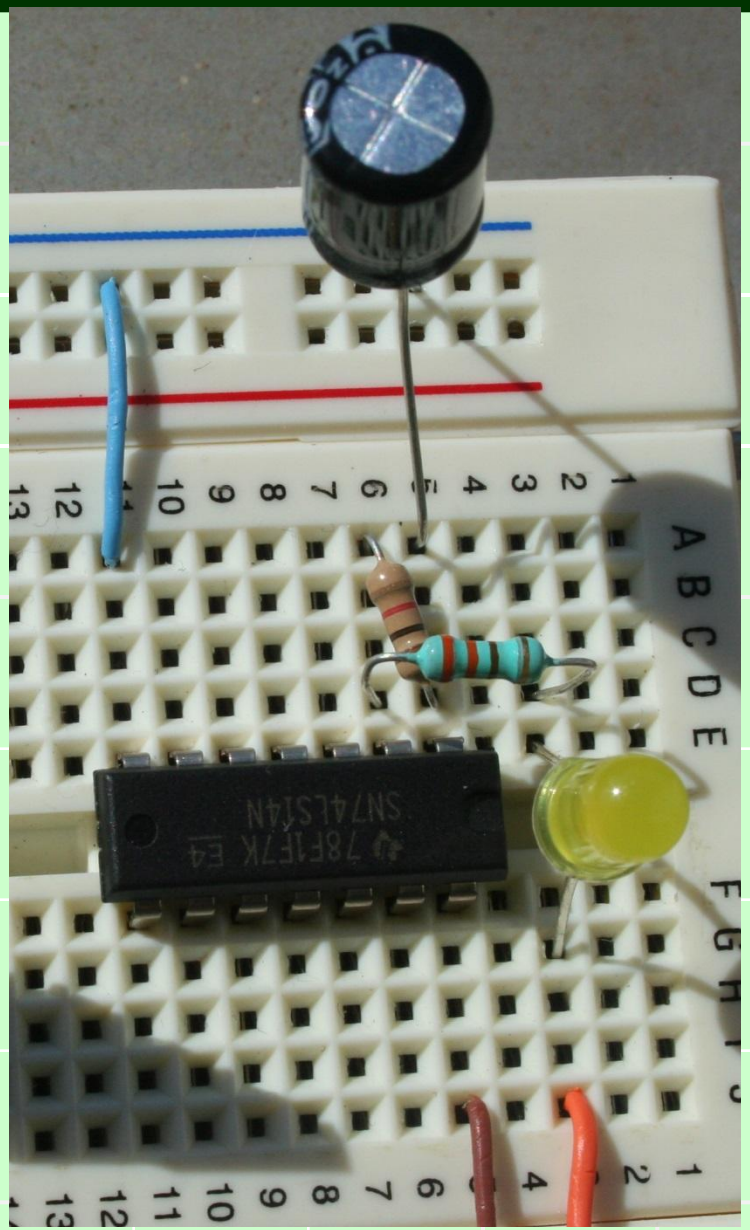

Multivibrador Astable
1K
out
in
1k
7414
R2
220
+0.90
Volts
2000u
C1
D1
IN
OUT
A
B
C
D
Digital Oscilloscope
Channel A
Channel C
Channel B
Channel D
Horizontal
Level
Position
Source
Auto
One-Shot
Cursors
Invert
A+B
C+D

JAGG



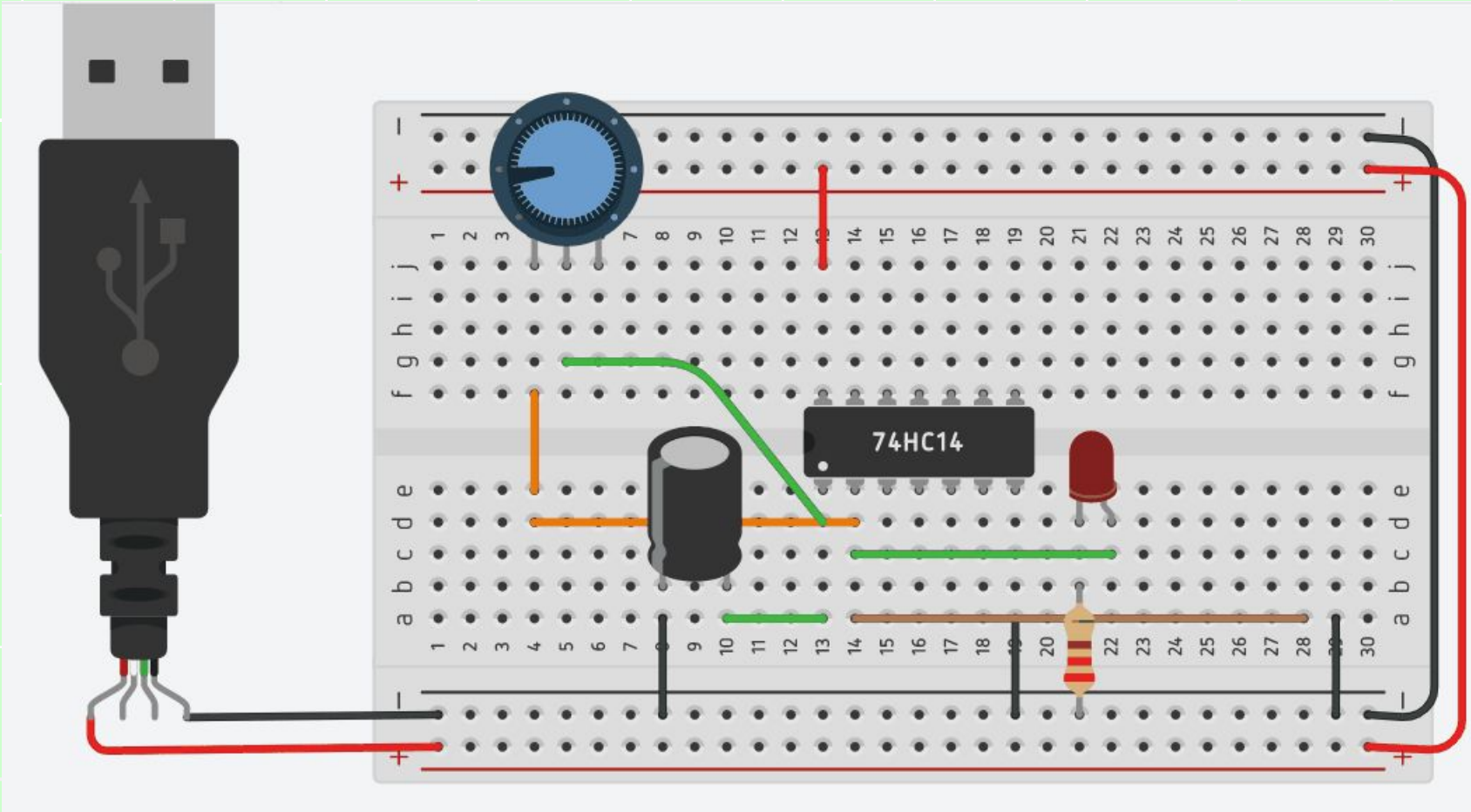




**Ventaja: Buena calidad del pulso**




**Desventaja : poco rango de frecuencias**

# Tinkercad




JAGG

# Proyecto Formativo 10

**1.- Manual: (Biestable)**

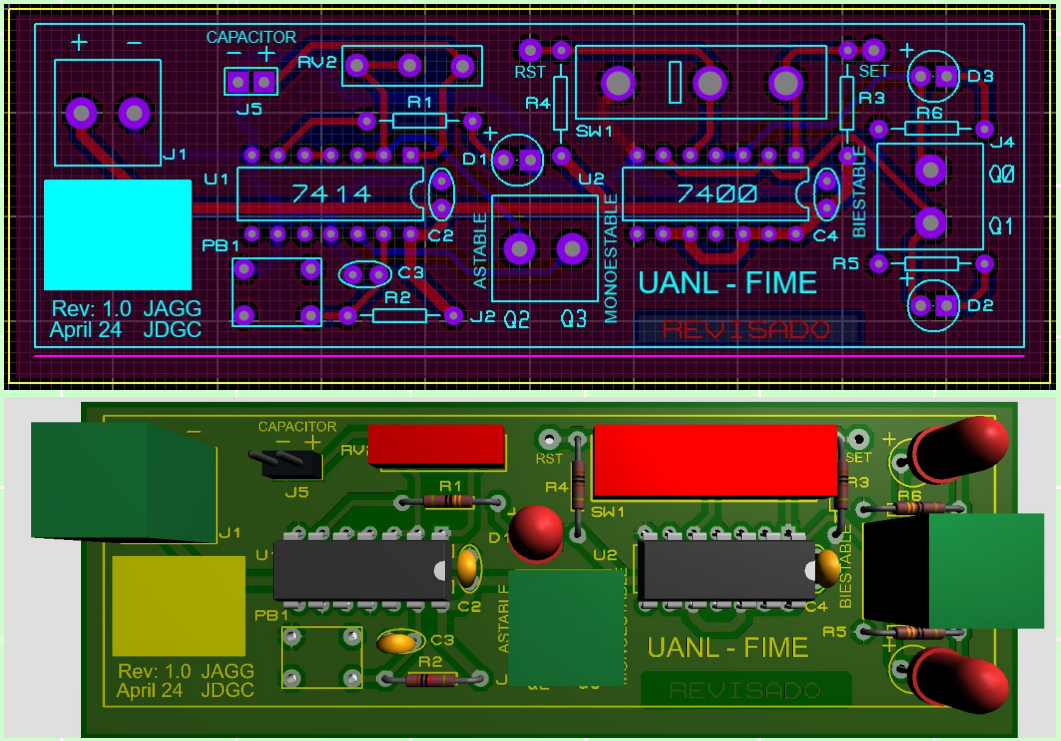
Flip Flop SC (SN74LS00), Eliminador de rebotes

**2.- Manual: (Monoestable)**

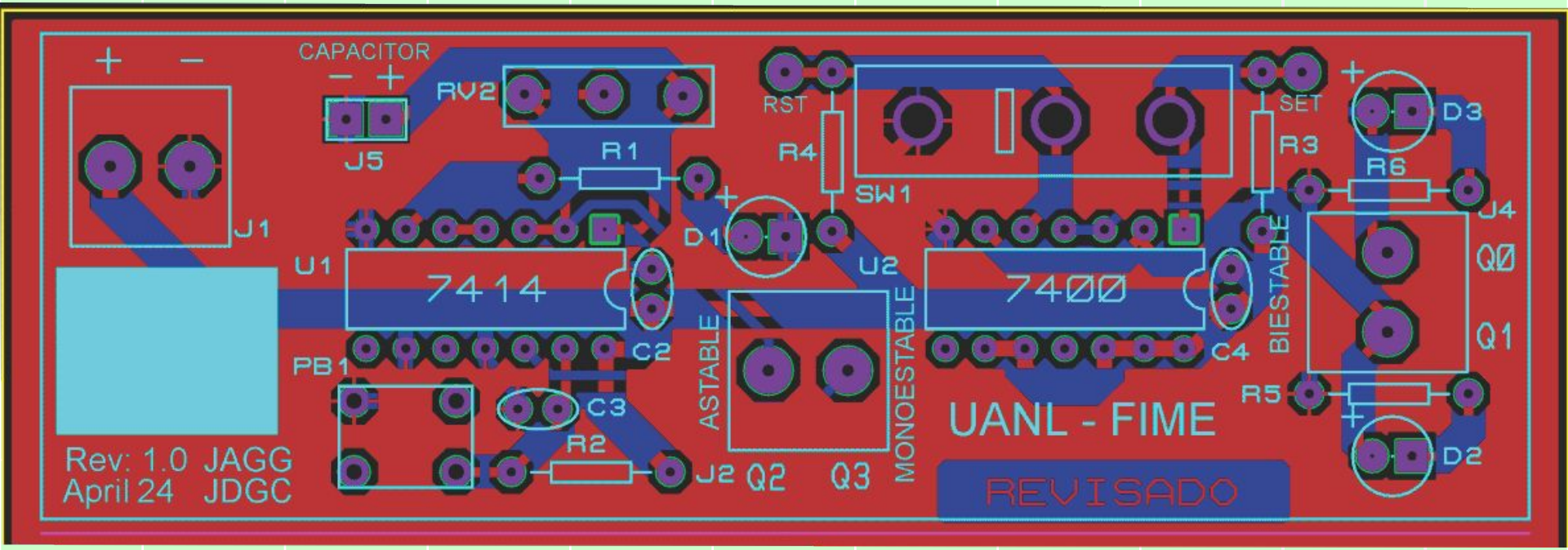
Compuerta Not, Schmitt Trigger (SN74LS14).

**3.- Automático: (Astable).**

Compuerta Not, Schmitt Trigger retroalimentada (SN7414).




JAGG


+ −
CAPACITOR
− +
RV2
J5
J1
R1
RST
R4
SW1
SET
R3
D3
R6
J4
Q0
Q1
D1
U1
7414
U2
7400
BIESTABLE
C2
C4
PB1
C3
ASTABLE
MONOESTABLE
UANL - FIME
R5
D2
R2
J2
Q2
Q3
Rev: 1.0 JAGG
April 24 JDGC
REVISADO

*JAGG*

# Multivibradores, Generadores de pulsos de sincronía

- **Biestable** (dos estados estables, Flip Flops )
  - RS, JK, T, D y **SC** (eliminador de rebotes)
- **Monoestable** (un estado estable y otro transitorio)
  - Redisparables (7414, 74121 )
  - No redisparables (555)

    Transición Positiva ↑ y Transición Negativa ↓

Manual

- **Astable** (sin estabilidad)
  - Timer (555 y 7414)

Automático
Frecuencia Variable

*JAGG*

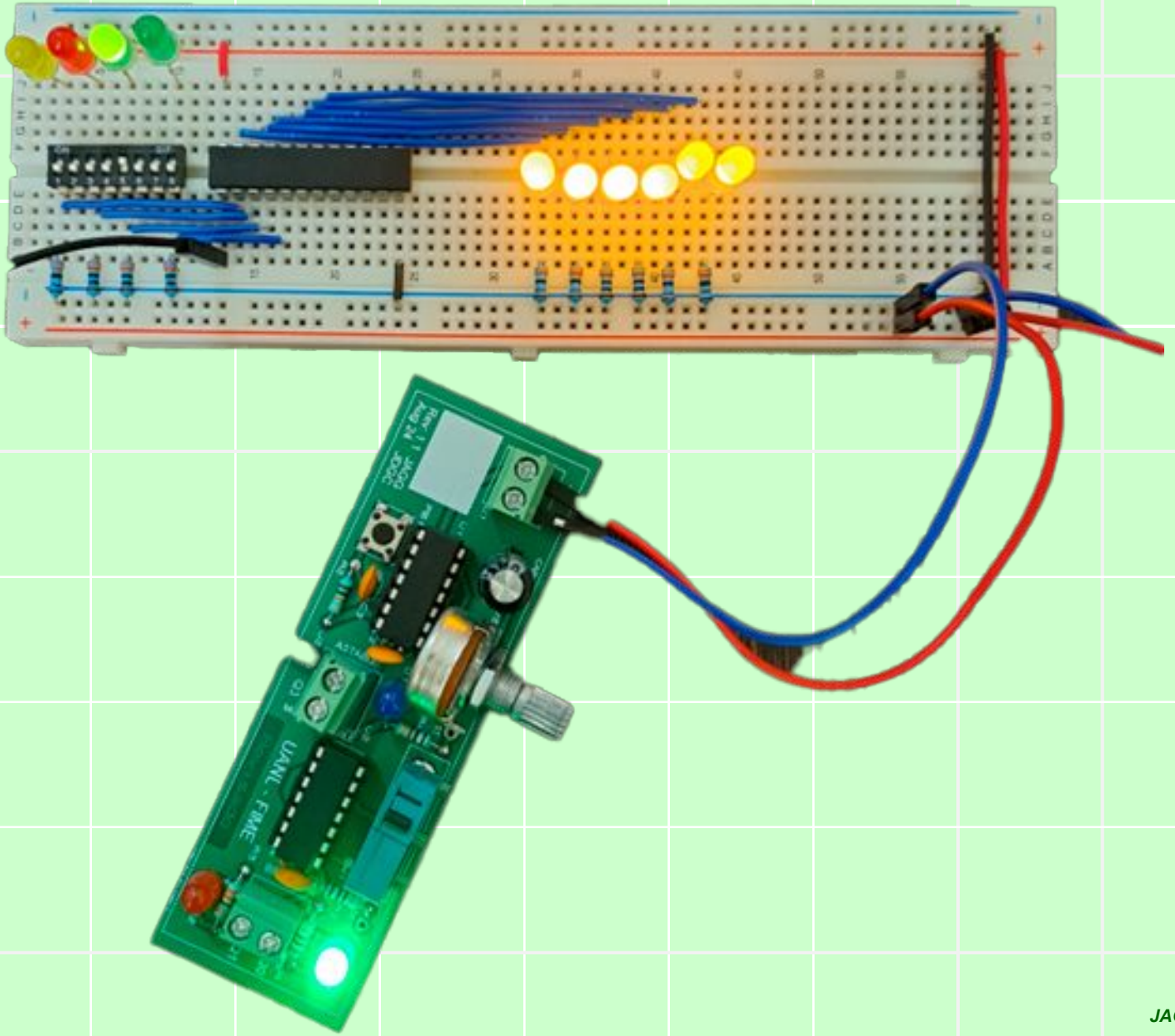
**En esta sesión comprendimos la importancia de los pulsos de sincronía en los sistemas secuenciales síncronos.**

- Los multivibradores biestables permiten almacenar un estado lógico y eliminar rebotes.
- Los monoestables generan un único pulso controlado por la duración del RC.
- Los astables producen trenes de pulsos periódicos para sincronizar circuitos.
- Aplicamos estos principios en la práctica PF10, utilizando compuertas SN7400, SN7414, y 555 para construir generadores de reloj manuales y automáticos

***La correcta generación y control de los pulsos de sincronía garantiza el funcionamiento estable y predecible de cualquier sistema secuencial, evitando errores de temporización y condiciones de carrera.***

*JAGG*

# PIA



# PF10

*JAGG*