

**INSTITUTO POLITECNICO NACIONAL**  
**ESCUELA SUPERIOR DE CÓMPUTO**

**Diseño de Sistemas Digitales**  
**Implementación de Flip Flop D, JK, SR y T por medio de VHDL**

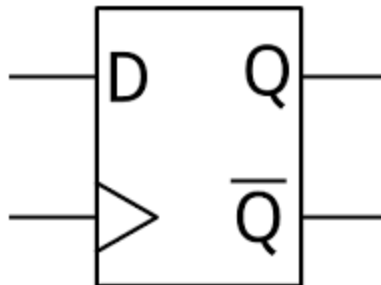
**Alcaraz Fraga Ricardo**  
**Cruz Contreras Karen Tiffany**  
**Montaño Morales Angeles Aranza**

**Grupo 2CM9**  
**Profesor Testa Nava Alexis**

## Diseño de la Práctica

Esta práctica tiene el objetivo de implementar 4 tipos de flip flops por medio de VHDL. Los flip flops son componentes de suma importancia en los circuitos digitales ya que nos dan la oportunidad de saltar de circuitos que dependen exclusivamente de las entradas (circuitos combinacionales) a los circuitos que dependen, además de las entradas, también de componentes de memoria (circuitos secuenciales). Estos dispositivos, como se puede ver en los diagramas, hacen uso de un pulso de reloj, el cual coordina la manipulación de la información.

### Flip Flop D



### Tabla de Verdad Flip Flop D

$D$	$Q(t+1)$
0	0 Restablecer
1	1 Establecer

### Código Flip Flop D

```
library ieee;
use ieee. std_logic_1164.all;
use ieee. std_logic_arith.all;
use ieee. std_logic_unsigned.all;

entity D_FF is
PORT( D,CLOCK: in std_logic;
```

```

Q: out std_logic);
end D_FF;

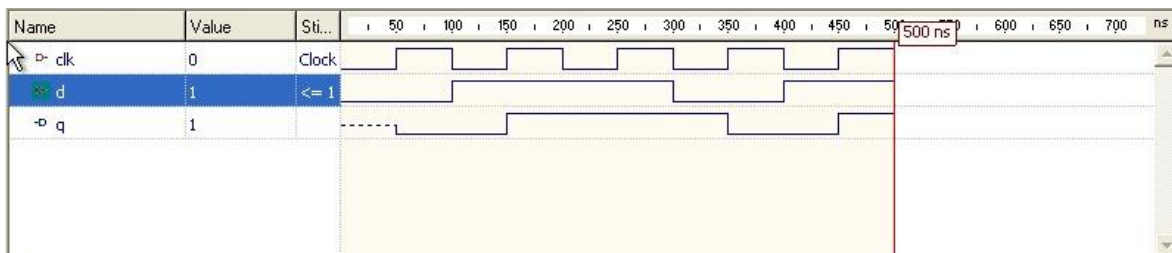
```

```

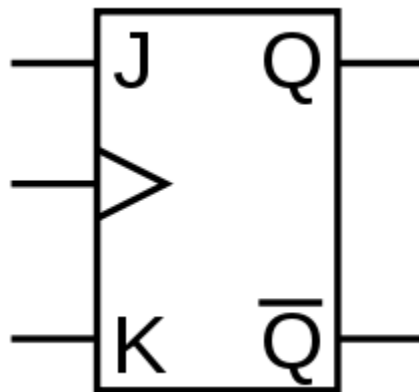
architecture behavioral of D_FF is
begin
  process (CLOCK)
  begin
    if (CLOCK='1' and CLOCK'EVENT) then
      Q <= D;
    end if;
  end process;
end behavioral;

```

## Simulación



## Flip Flop JK



## Tabla de Verdad Flip Flop JK

<i>J</i>	<i>K</i>	<i>Q(t+1)</i>
0	0	<b>Q(t) Sin cambio</b>
0	1	<b>0 Restablecer</b>
1	0	<b>1 Establecer</b>
1	1	<b>Q'(t) Complementar</b>

## Código Flip Flop JK

```
library ieee;
use ieee. std_logic_1164.all;
use ieee. std_logic_arith.all;
use ieee. std_logic_unsigned.all;

entity JK_FF is
PORT( J,K,CLOCK: in std_logic;
Q, QB: out std_logic);
end JK_FF;

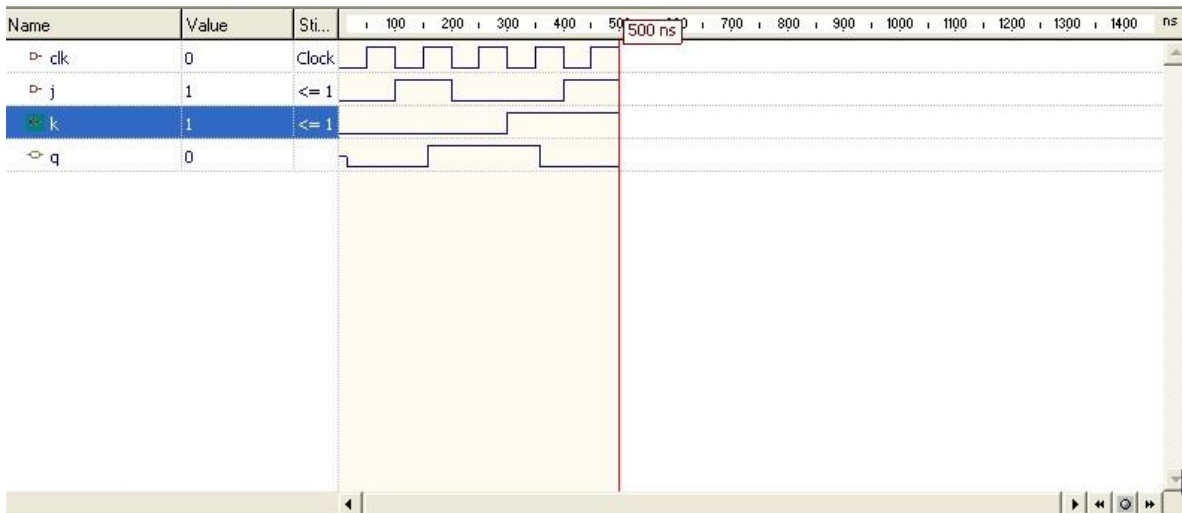
Architecture behavioral of JK_FF is
begin
PROCESS (CLOCK)
variable TMP: std_logic;
begin
if (CLOCK='1' and CLOCK'EVENT) then
if (J='0' and K='0') then
TMP:=TMP;
elsif (J='1' and K='1') then
TMP:= not TMP;
elsif (J='0' and K='1') then
TMP:='0';
else
TMP:='1';
end if;
end if;
end if;
```

```

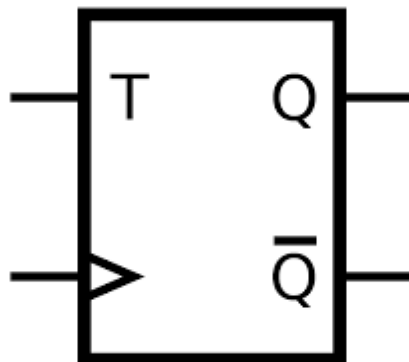
Q<=TMP;
Q <=not TMP;
end PROCESS;
end behavioral;

```

## Simulación



## Flip Flop T



## Tabla de Verdad Flip Flop T

$T$	$Q(t+1)$
0	Q(t) Sin cambio
1	Q'(t) Complementar

## Código Flip Flop T

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

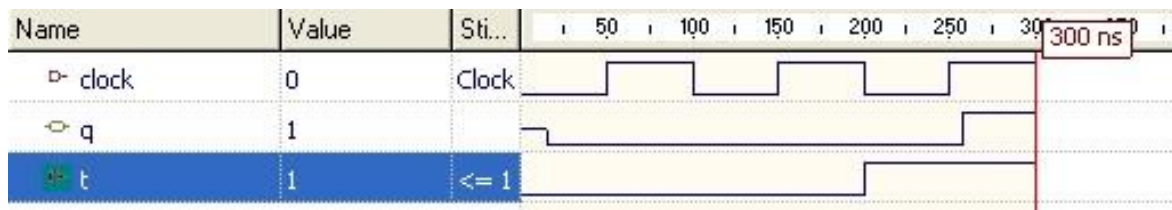
entity T_FF is
port( T: in std_logic;
Clock: in std_logic;
Q: out std_logic);
end T_FF;

architecture Behavioral of T_FF is
signal tmp: std_logic;
begin
process (Clock)
begin
if Clock'event and Clock='1' then

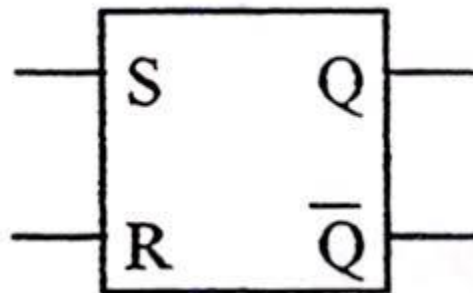
if T='0' then
tmp <= tmp;
elsif T='1' then
tmp <= not (tmp);
end if;
end if;
end process;
Q <= tmp;
end Behavioral;

```

## Simulación



## Flip Flop SR



## Tabla de Verdad SR

S	R	Q(t+1)
0	0	Q(t) Sin cambio
1	0	1 Establecer
0	1	0 Restablecer
1	1	--- Prohibido

## Código Flip Flop SR

```

library ieee;
use ieee. std_logic_1164.all;
use ieee. std_logic_arith.all;
use ieee. std_logic_unsigned.all;

entity SR_FF is
PORT( S,R,CLOCK: in std_logic;
Q, QBAR: out std_logic);

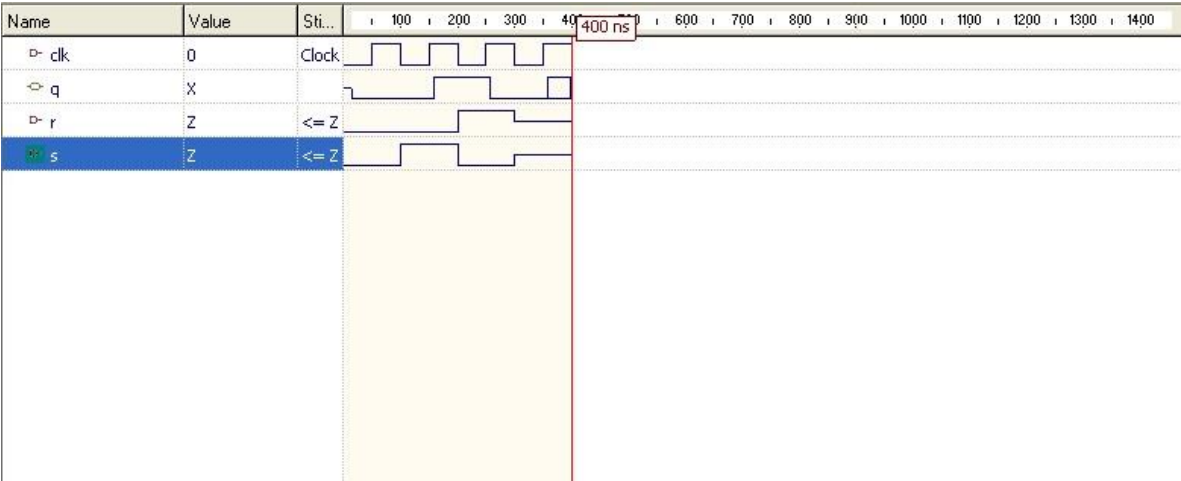
```

```
end SR_FF;
```

Architecture behavioral of SR\_FF is

```
begin
PROCESS (CLOCK)
variable tmp: std_logic;
begin
if (CLOCK='1' and CLOCK'EVENT) then
if (S='0' and R='0') then
tmp:=tmp;
elsif (S='1' and R='1') then
tmp:='Z';
elsif (S='0' and R='1') then
tmp:='0';
else
tmp:='1';
end if;
end if;
Q <= tmp;
QBAR <= not tmp;
end PROCESS;
end behavioral;
```

Simulación





## **Conclusiones**

El funcionamiento del flip flop t es muy parecido al del flip flop jk, por otro lado, el que es, con bastante diferencia, el de menor complejidad y mayor facilidad de uso e implementación es el flip flop d.

Al final, todos sirven para almacenar información, esta información puede funcionar como entrada para un circuito combinacional.

Además de las entradas respectivas a cada flip flop, estos hacen uso de un reloj. En nuestro caso, programamos los flip flops de manera que funcionaran con flancos de subida.

## **Bibliografía**

[1] Diseño Digital. M. Morris Mano. Pearson, Tercera Edición. Pag 167-217.