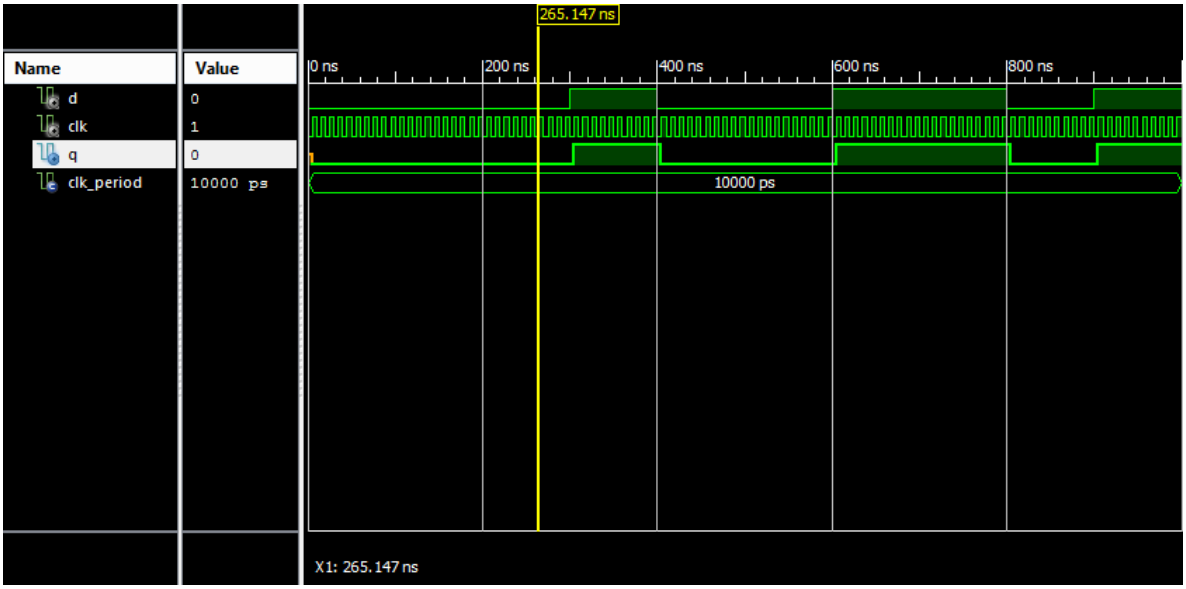
Un Latch D asíncrono

Código

```
1  LIBRARY ieee ;
2  USE ieee.std_logic_1164.all ;
3
4  ENTITY latch IS
5     PORT ( D, Clk : IN STD_LOGIC ;
6        Q : OUT STD_LOGIC) ;
7     END latch ;
8
9  ARCHITECTURE Behavior OF latch IS
10 BEGIN
11    PROCESS ( D, Clk )
12    BEGIN
13       IF Clk = '1' THEN
14          Q <=  D ;
15       END IF ;
16    END PROCESS ;
17 END Behavior ;
18
```

Simulación

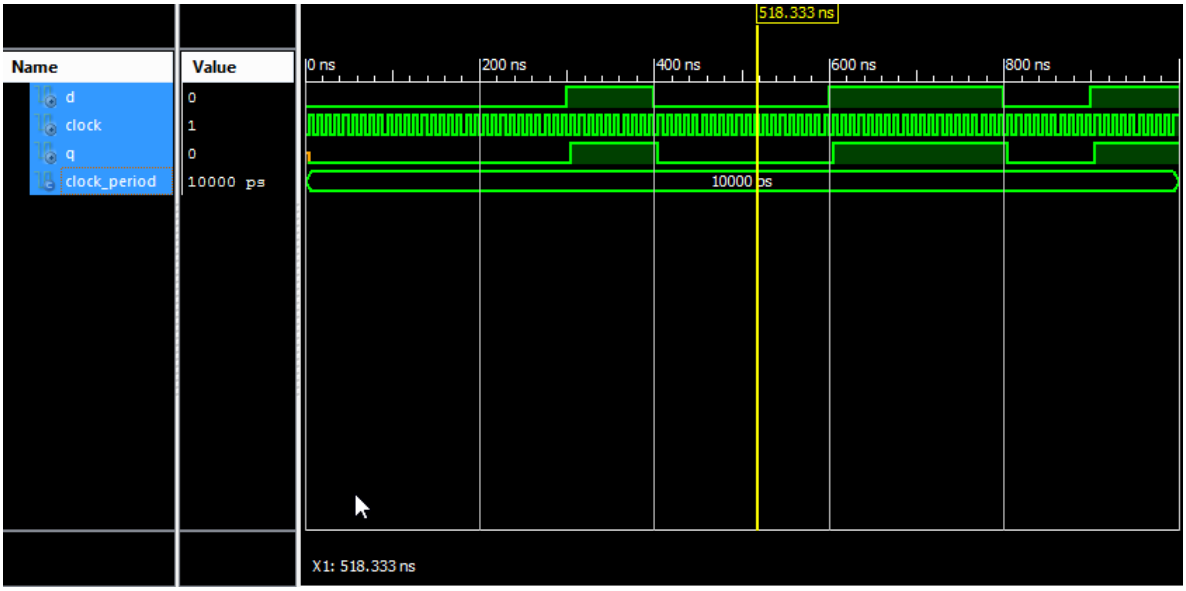


Un Flip Flop D

Código

```
1  LIBRARY ieee ;
2  USE ieee.std_logic_1164.all ;
3
4  ENTITY FF_D IS
5     PORT ( D, Clock : IN STD_LOGIC ;
6        Q : OUT STD_LOGIC) ;
7  END FF_D ;
8
9  ARCHITECTURE Behavior OF FF_D IS
10 BEGIN
11    PROCESS ( Clock )
12    BEGIN
13       IF Clock'EVENT AND Clock =  '1' THEN
14          Q <=  D ;
15       END IF ;
16    END PROCESS ;
17 END Behavior ;
```

Simulación



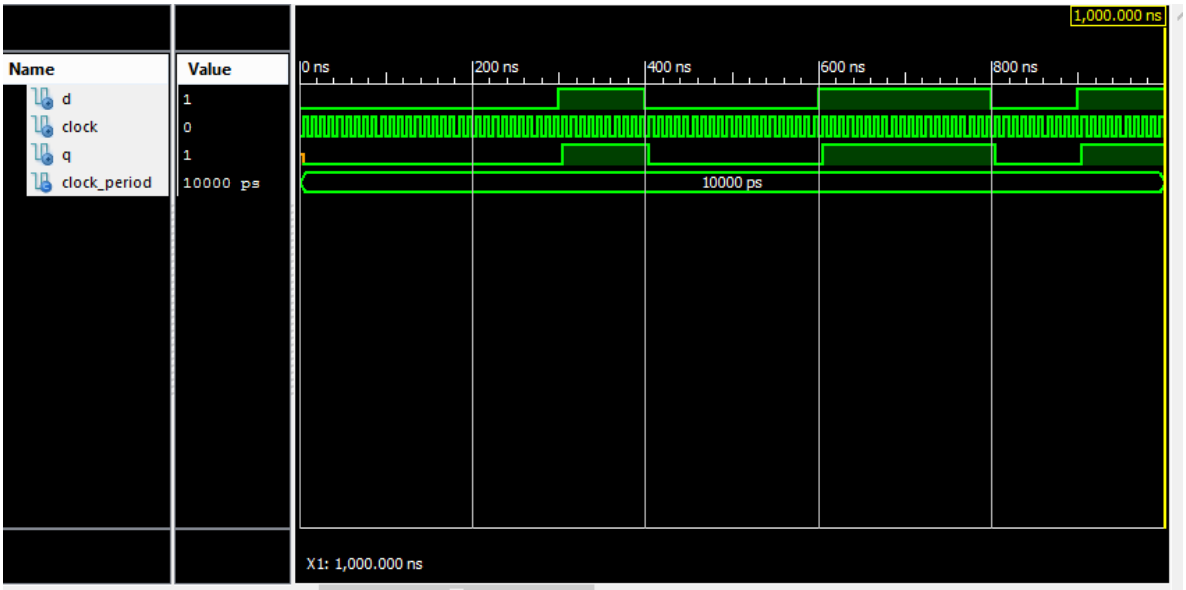
Otra manera de construir un Flip Flop D.

Utiliza la instrucción WAIT UNTIL Clock'EVENT AND Clock= '1', que produce el mismo efecto que la instrucción IF del código anterior

Código

```
1   LIBRARY ieee;
2   USE ieee.std_logic_1164.all;
3
4   ENTITY FF_D_2 IS
5      PORT ( D, Clock : IN STD_LOGIC ;
6         Q : OUT STD_LOGIC ) ;
7   END FF_D_2 ;
8
9   ARCHITECTURE Behavior OF FF_D_2 IS
10  BEGIN
11     PROCESS
12     BEGIN
13     WAIT UNTIL Clock'EVENT AND Clock =  '1' ;
14        Q <=  D ;
15     END PROCESS ;
16  END Behavior ;
17
```

Simulación

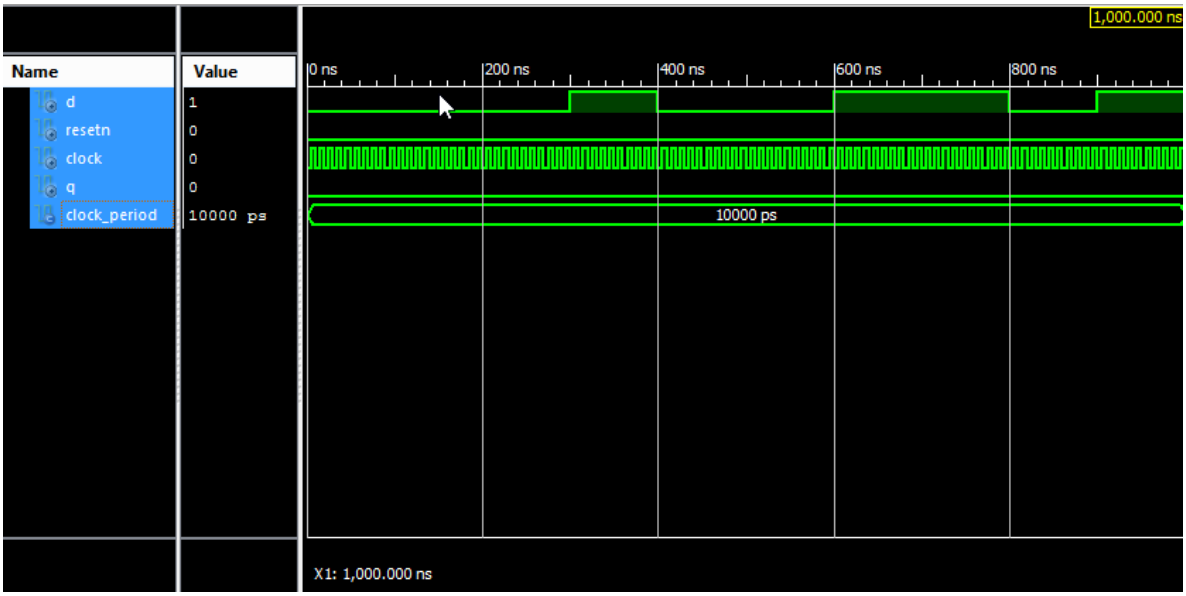


Borrado asíncrono

Código

```
1  LIBRARY ieee ;
2  USE ieee.std_logic_1164.all ;
3
4  ENTITY BorrAsin IS
5     PORT ( D, Resetn, Clock : IN STD_LOGIC ;
6        Q : OUT STD_LOGIC) ;
7  END BorrAsin ;
8
9  ARCHITECTURE Behavior OF BorrAsin IS
10 BEGIN
11    PROCESS ( Resetn, Clock )
12    BEGIN
13       IF Resetn = '0' THEN
14          Q <=  '0' ;
15       ELSIF Clock'EVENT AND Clock =  '1' THEN
16          Q <=  D ;
17       END IF ;
18    END PROCESS ;
19 END Behavior ;
```

Simulación



Borrado síncrono

Código

```
1  LIBRARY ieee ;
2  USE ieee.std_logic_1164.all ;
3
4  ENTITY BorrSin IS
5     PORT ( D, Resetn, Clock : IN STD_LOGIC ;
6        Q : OUT STD_LOGIC) ;
7  END BorrSin ;
8
9  ARCHITECTURE Behavior OF BorrSin IS
10 BEGIN
11    PROCESS
12    BEGIN
13    WAIT UNTIL Clock'EVENT AND Clock = '1' ;
14       IF Resetn = '0' THEN
15          Q <= '0' ;
16       ELSE
17          Q <= D ;
18       END IF ;
19    END PROCESS ;
20 END Behavior ;
21
```

Simulación