

PLAZA, ELMO L.
BSCPE 3A

VHDL CODE FOR T FLIP FLOP:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
```

entity T_flipflop is

```
    Port ( t : in STD_LOGIC; -- T input
          clk : in STD_LOGIC; -- Clock input
          rst : in STD_LOGIC; -- Active-low reset
          q : buffer STD_LOGIC -- Output (buffer allows reading and writing)
    );
end T_flipflop;
```

architecture Behavioral of T_flipflop is

```
    signal div : std_logic_vector(22 downto 0) := (others => '0'); -- Clock divider counter
    signal clkd : std_logic; -- Divided clock
begin
```

-- Clock divider: Slow down the clock

```
process(clk)
begin
    if rising_edge(clk) then
        div <= div + 1;
    end if;
end process;
```


-- Select the slower clock signal

clkd <= div(20); -- You can adjust this value for more or less division

-- T flip-flop behavior

```
process(clkd, rst)
begin
    if rst = '0' then -- Active-low reset
        q <= '0'; -- Reset output to '0'
    elsif rising_edge(clkd) then
        if t = '1' then -- Toggle on T=1
            q <= not q; -- Toggle the output
        end if;
    end if;
end process;
```

end Behavioral;

Clear	T	Clock	Qn+1	$\overline{Q_{n+1}}$
1	0	0	0	$\overline{Q_n}$
0	1		$\overline{Q_n}$	Qn

Behavior of a T Flip-Flop

T	Clock Edge	Q (LED)	What Happens
0	↑ (rising)	Holds last state	No toggle
1	↑ (rising)	Toggles	Changes from 0→1 or 1→0

So if:

- You're feeding in a **slow divided clock**,
- Your LED is connected to **Q**,
- And you **press the T button** to make **T='1'**, then:

The LED will **toggle (flip)** on each clock rising edge **only if T=1** at that moment.

- The LED is toggling **only when you press T**, and depending on **when you press it** (relative to the clock edge), the LED either turns **on or off**.
- This is **normal and expected** behavior for a T flip-flop.

Why It Feels Random:

Because of the clock speed and your **human reaction time**, sometimes:

- You press T **just before** a rising clock edge → it toggles.
- You press T **just after** a rising clock edge → no effect (waits for next one).

This causes the LED to **sometimes turn on, sometimes off**, depending on the **timing**.

The screenshot displays the Quartus Prime IDE interface. The main window shows the VHDL code for a T flip-flop. The code includes a clock divider and a T flip-flop behavior block. The Project Navigator on the left shows the hierarchy of the project, including the T_flipflop.vhd file. The Programmer window on the right shows the hardware setup, including the device (EP4CE6E22), the clock (100MHz), and the programming mode (JTAG). The status bar at the bottom indicates that the compilation was successful.

```

12 end T_flipflop;
13
14 architecture Behavioral of T_flipflop is
15     signal div : std_logic_vector(22 downto 0) := (others => '0'); -- Clock divider counter
16     signal clkd : std_logic; -- Divided clock
17 begin
18     -- Clock divider: Slow down the clock
19     process(clk)
20     begin
21         if rising_edge(clk) then
22             div <= div + 1;
23             end if;
24             clkd <= div(20); -- You can adjust this value for more
25         end process;
26
27     -- Select the slower clock signal
28     clkd <= div(20);
29
30     -- T Flip-flop behavior
31     process(clkd, rst)
32     begin
33         if rst = '0' then -- Active-low reset
34             q <= '0'; -- Reset output to '0'
35         elsif rising_edge(clkd) then
36             if t = '1' then -- Toggle on T=1
37                 q <= not q; -- Toggle the output
38             end if;
39         end if;
40     end process;
41 end Behavioral;
42
43

```

Compilation Report - T_flipflop

Programmer - D:\QUARTUS_PRIME_LITE\LAB_ACTIVITIES\T_flipflop\T_flipflop - T_flipflop - [T_flipflop...]

Hardware Setup: USB-Blaster (USB-0) Mode: JTAG Progress: 100% (Successful)

File Device Checksum Usercode Program/ Verify Blank- Examine Security Err
Configure Bit

output_files/T_fll... EP4CE6E22 00094F18 00094F18

TOI
EP4CE6E22
TDO

Running Quartus Prime EDA Netlist Writer
Command: quartus_eda --read_settings_files=off --write_settings_files=off T_flipflop -c
18236 Number of processors has not been specified which may cause overloading on shared machines. Set the global assignment NUM_PARALLEL_PROCESSORS in your QSF to an appropriate va
204019 Generated File T_FlipFlop.vho in folder "D:\QUARTUS_PRIME_LITE\LAB_ACTIVITIES\T_FlipFlop\simulation\modelsim\" for EDA simulation tool
Quartus Prime EDA Netlist Writer was successful. 0 errors, 1 warning
293000 Quartus Prime Full Compilation was successful. 0 errors, 13 warnings