library ieee;

use ieee.std\_logic\_1164.all;

entity T\_FF is

port( T,Reset: in std\_logic;

Q: out std\_logic);

end T\_FF;

architecture T\_FF\_ARCH of T\_FF is

signal clk: std\_logic:='0';

signal intQ : std\_logic;

begin

tact: process

begin

clk<=not clk;

wait for 50 ns;

end process;

process(clk,reset)

begin

if reset = '1' then

intQ<='0';

elsif clk='1' and clk'event then

if T='0' then

intQ<=intQ;

else

intQ<= not intQ;

end if;

end if;

end process;

Q<=intQ;

end T\_FF\_ARCH;

library ieee;

use ieee.std\_logic\_1164.all;

entity T\_FF\_TB is

end T\_FF\_TB;

architecture A of T\_FF\_TB is

component T\_FF is

port( T,Reset: in std\_logic;

Q: out std\_logic);

end component;

signal T,Reset,Q: std\_logic;

begin

ust: T\_FF port map(T, Reset, Q);

process

begin

Reset<='1';

T<='0';

wait for 50 ns;

Reset<='1';

T<='1';

wait for 50 ns;

Reset<='0';

T<='0';

wait for 50 ns;

Reset<='0';

T<='1';

wait for 50 ns;

end process;

end A;