```
--esim1
library ieee;
use ieee.std_logic_1164.all;

entity tristate is
        port (A, EN: in std_logic; Y: out in std_logic);
end tristate;

archtitecture toiminta of tristate is
begin
        process(A, EN)
        begin
                if(EN='0') then Y<=A;
                else Y<='Z' ;     --Y suuri-imp. tilaan
                end if;
        end process;
end toiminta;


--esim2
library ieee;
use ieee.std_logic_1164.all;

entity sekvenssi  is
        port (A, B: out std_logic;  CLK: in std_logic);
end sekvenssi;

archtitecture toiminta of sekvenssi is
begin
        process(CLK)
                variable apu: std_logic_vector(1 downto 0);
        begin
                if (CLK'event and CLK='1') then
                        if(apu="00") then apu:= "01" ;
                        elsif (apu="01") then apu:="11" ;
                        elsif (apu="11") then apu:="00" ;   --turha
                        else apu="00" ;
                        end if;
                end if;
                A<=apu(0);
                B<=apu(1);
        end process;
end toiminta;
```

```
--esim3
ENTITY state_machine IS
  PORT(
    clk     : IN   STD_LOGIC;
    input   : IN   STD_LOGIC;
    reset   : IN   STD_LOGIC;
    output  : OUT  STD_LOGIC_VECTOR(1 downto 0));
END state_machine;

ARCHITECTURE a OF state_machine IS
  TYPE STATE_TYPE IS (s0, s1, s2);  --oma tietotyyppi
  SIGNAL state   : STATE_TYPE; --signaalimuuttuja state
BEGIN
  PROCESS (clk, reset)
  BEGIN
    IF reset = '1' THEN
      state <= s0;
    ELSIF (clk'EVENT AND clk = '1') THEN
      CASE state IS
        WHEN s0=>
          IF input = '1' THEN
            state <= s1;
          ELSE
            state <= s0;
          END IF;
        WHEN s1=>
          IF input = '1' THEN
            state <= s2;
          ELSE
            state <= s1;
          END IF;
        WHEN s2=>
          IF input = '1' THEN
            state <= s0;
          ELSE
            state <= s2;
          END IF;
      END CASE;
    END IF;
  END PROCESS;

  PROCESS (state)             --lähtöjen kirjoituksesta
  BEGIN                       --oma prosessi
    CASE state IS
      WHEN s0 =>
        output <= "00";
      WHEN s1 =>
        output <= "01";
      WHEN s2 =>
        output <= "10";
    END CASE;
  END PROCESS;
END a;
```

```
--esim4, ls374
library ieee;
use ieee.std_logic_1164.all;

entity ls374 is
        port (D: in std_logic_vector(0 to 7);
                Q: out std_logic_vector(0 to 7);
                CLK,  OC: in std_logic);
end ls374;

archtitecture toiminta of ls374 is
begin
        process(CLK, OC)
                variable apu: std_logic_vector(0 to 7);
        begin
                if (CLK='1' and CLK'event) then apu:=D;
                end if;

                if (OC='0') then Q<=apu;
                                else Q<="ZZZZZZZZ";
                                end if;
        end process;
end toiminta;
```