

```

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

entity dkiikku is --esittely, liitynnät
  port( D, CLK: in std_logic; Q, QINV: out std_logic); --
end dkiikku;

architecture toiminta of dkiikku is --toiminnan kuvaus
begin
  process (CLK)
  begin
    if (CLK='1' and CLK'event) then
      Q <= D;
      QINV <= not D;
    end if;
  end process;
end toiminta;

```

```

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

entity dkiikkuf is --esittely, liitynnät
  port( D, CLK: in std_logic; Q, QINV: out std_logic); --
end dkiikkuf;

architecture toiminta of dkiikkuf is --toiminnan kuvaus
begin
  process (CLK)
  --aliohjelma nousureuna
  function nousureuna(signal s: std_logic) return boolean is
    variable val: boolean;
  begin
    if(s='1' and s'event) then val:=true;
    else val:=false;
    end if;
    return val;
  end nousureuna;

  begin --prosessilause alkaa
    if nousureuna(CLK)=true then
      Q <= D;
      QINV <= not D;
    end if;
  end process;
end toiminta;

```

```

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

entity dkiikkup is --esittely, liitynnät
    port( D, CLK: in std_logic; Q, QINV: out std_logic); --
end dkiikkup;

architecture toiminta of dkiikkup is --toiminnan kuvaus
begin
    process (CLK)
        --aliohjelma nousureuna
        procedure dfloppi (signal cclk, dd: in std_logic; signal qq, qqinv: out std_logic) is
        begin
            if(cclk='1' and cclk'event) then
                qq <= dd;
                qqinv <= not dd;
            end if;
        end dfloppi;

        begin --prosessilause alkaa
            dfloppi(CLK, D, Q, QINV);
        end process;
    end toiminta;

```

```

library ieee;
use ieee.std_logic_1164.all;
entity dkiikku_fp is
    port(CLK, D: in std_logic; Q: out std_logic);
end dkiikku_fp;

architecture toiminta of dkiikku_fp is
begin --architecturen begin
    process (CLK)
        function nousureuna(signal s: std_logic) return boolean is
        begin
            if(s='1' and s'event) then return true;
            else return false;
            end if;
        end nousureuna;

        --proseduuri-tyyppinen aliohjelma ei palauta arvoa
        procedure kiikku(signal cc, dd: in std_logic; signal qq: out std_logic) is
        begin --proceduren begin
            if (nousureuna(cc)=true) then qq<=dd;
            end if;
        end kiikku;

        begin --process-lauseen begin
            kiikku(CLK, D, Q); --kutsutaan aliohjemiaa
        end process;
    end toiminta;
end

```