

-- **** STUDENT: 64000225.....	3
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	3
-- **** STUDENT: 64190088.....	4
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	4
-- **** STUDENT: 64200100.....	5
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	5
-- **** STUDENT: 64200112.....	6
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	6
-- **** STUDENT: 64200163.....	7
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	7
-- **** STUDENT: 64200238.....	8
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	8
-- **** STUDENT: 64200288.....	9
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	9
-- **** STUDENT: 64200296.....	10
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	10
-- **** STUDENT: 64200385.....	11
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	11
-- **** STUDENT: 64210113.....	12
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	12
-- **** STUDENT: 64210132.....	13
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	13
-- **** STUDENT: 64210290.....	14
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	14
-- **** STUDENT: 64210382.....	15
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	15
-- **** STUDENT: 64210384.....	16
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	16
-- **** STUDENT: 64210386.....	17
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	17
-- **** STUDENT: 64210445.....	18
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	18
-- **** STUDENT: 64210455.....	19
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	19

-- **** STUDENT: 64210457 .....	20
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	20
-- **** STUDENT: 64240430 .....	21
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	21
-- **** PREDLOGA VAJE .....	22
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	22

```

-- *****
-- **** STUDENT: 64000225
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal          ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64190088
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64200100
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal      ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64200112
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64200163
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal          ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64200238
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal          ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```



```

-- *****
-- **** STUDENT: 64200288
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64200296
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal          ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64200385
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal      ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210113
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal          ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210132
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210290
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal          ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210382
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal      ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210384
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal          ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```



```

-- *****
-- **** STUDENT: 64210386
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210445
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210455
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64210457
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
        clk,      -- signal      ure ( prožen na sprednjo fronto )
        nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
        Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** STUDENT: 64240430
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal          ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;      -- signal          za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );        -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';          -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

```

-- *****
-- **** PREDLOGA VAJE
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY dff IS
PORT (      D,      -- vhod D flip-flopa
         clk,      -- signal      ure ( prožen na sprednjo fronto )
         nCLEAR : IN STD_LOGIC;  -- signal      za asinhrono brisanje ( nCLEAR = '0' se postavi Q=>'0' )
         Q : OUT STD_LOGIC );    -- izhod D flip-flopa
END dff;

ARCHITECTURE ideal OF dff IS
BEGIN
PROCESS ( clk )
    BEGIN
        IF ( nCLEAR = '0' ) THEN
            Q      <= '0';      -- asinhrono brisanje izhoda ( clear )
        ELSIF rising_edge( clk ) THEN
            Q      <= D;
        END IF;
    END PROCESS;
END ideal;

```

