

Ocenjevanje datoteke testnih vrednosti CLA seštevalnika

Ocenjevanje datoteke testnih vrednosti CLA seštevalnika	1
-- **** STUDENT: 64000225	4
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Za signal X_num, Y_num morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;	
	4
Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)). Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	4
-- **** STUDENT: 64200100	6
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Za signal X8u, Y8u morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;	
	6
Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).	
	6
Zakasnitev CLA ni 2 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	6
-- **** STUDENT: 64200112	8
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Za signal Xcompare, Ycompare je smiselno (ni pa nujno – lahko imate tudi celoten 32-bitni obseg) določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;	
	8
Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).	
	8
Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	8
-- **** STUDENT: 64200163	10
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).	
	10
Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	10
-- **** STUDENT: 64200238	11
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).	
	11
Zakasnitev CLA ni 3 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	11
-- **** STUDENT: 64200288	13
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).	
	13
Zakasnitev CLA ni 3 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	13
-- **** STUDENT: 64200296	15
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).	
	15
Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	15
-- **** STUDENT: 64200385	17
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).	
	17
Za signal x_int, y_int morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;	
	17
Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).	
	17
-- **** STUDENT: 64210113	19

-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	19
	Za signal unsignX, unsignY morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;.....	19
	Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	19
-- **** STUDENT: 64210290.....		21
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	21
-- **** STUDENT: 64210382.....		23
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	23
	Za signal X_int, Y_int morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;.....	23
	Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	23
-- **** STUDENT: 64210384.....		25
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	25
	Za signal X_int, Y_int morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;.....	25
	Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	25
-- **** STUDENT: 64210386.....		27
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	27
	Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	27
-- **** STUDENT: 64210445.....		29
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	29
	Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	29
-- **** STUDENT: 64210455.....		31
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	31
	Za signal A_int, B_int morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;.....	31
	Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	31
-- **** STUDENT: 64240429.....		33
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Zakasnitev CLA ni 2 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	33
-- **** STUDENT: 64210457.....		34
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	34
	Za signal X_n, Y_n morate določiti parametriziran obseg nepredznačenih števil integer range 0 to 2**n-1 := 0;.....	34
	Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	34
-- **** STUDENT: 64240430.....		36
-- KOMENTARJI K OCENI NALOGE	-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	36
	Zakasnitev CLA ni 5 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).....	36

-- **** STUDENT: 64210132.....	38
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map (n=>n)).....	38
Ideja datoteke testnih vrednosti je, da preleti celoten obseg števil x in y ter na drugačen način (torej z VHDL + operatorjem) preveri, če so razlike v izračunu strojne komponente in simulatorja.....	38
NASLEDNJIČ KODO NALOŽITE V USTREZEN RAZDELEK (OSTALO_x), KJER JE X ŠTEVILKA DOMAČE NALOGE.....	38

```

-- *****
-- **** STUDENT: 64000225
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Za signal X_num, Y_num morate določiti parametriziran obseg
nepredznačenih števil integer range 0 to 2**n-1 := 0;
Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map
(n=>n)). Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path
delay).
-- *****

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
    generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS

    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Inputs
    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    signal      X_num, Y_num : integer range 0 to 255 := 0;

    -- Outputs
    signal      S, S_test : std_logic_vector( n-1 downto 0 );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,

```

```

Gout => Gout,
Pout => Pout,
Cout => Cout
);

    -- Stimulus process
stim_proc: process
begin
    -- hold reset state for 100 ns.
wait for 100 ns;
    for i in 0 to ( 2**n )-1 loop
        X_num <= i;
        X      <= std_logic_vector( to_unsigned( i, X'length ) );
        for j in 0 to ( 2**n )-1 loop
            Y_num <= j;
            Y      <= std_logic_vector( to_unsigned( j, Y'length ) );
            wait for 1 ns;
            assert( S_test = S ) report "Test failed. X:" & integer'image( i
) & " Y:" & integer'image( j ) severity error;
            exit when ( S_test /= S );
        end loop;
    end loop;

wait;

end process;

    test_proc: process ( X_num, Y_num )
    begin
        S_test <= std_logic_vector( to_unsigned( ( X_num + Y_num ) mod 2**n,
S_test'length ) );
    end process;
END;

```

```

-- *****
-- **** STUDENT: 64200100
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Za signal X8u, Y8u morate določiti parametriziran obseg nepredznačenih
števil integer range 0 to 2**n-1 := 0;
Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map
(n=>n)).
Zakasnitev CLA ni 2 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.all;
use ieee.numeric_std.all;

entity cla_add_n_bit_tb is
generic( n:natural:=8 );
end cla_add_n_bit_tb;

architecture behavior of cla_add_n_bit_tb is
component cla_add_n_bit
    PORT (
        Cin      : in      std_logic ;
        X, Y      : in      std_logic_vector( n-1 downto 0 );
        S          : out     std_logic_vector( n-1 downto 0 );
        Gout, Pout, Cout : out  std_logic );
end component;

signal      Cin: std_logic := '0';
signal      X: std_logic_vector( n-1 downto 0 ):= ( others=>'0' );
signal      Y: std_logic_vector( n-1 downto 0 ):= ( others=>'0' );
signal      S, Sprocs: std_logic_vector( n-1 downto 0 );
signal      Gout: std_logic;
signal      Pout: std_logic;
signal      Cout: std_logic;
signal      X8u, Y8u: integer range 0 to 255:=0;--

begin
    uut: cla_add_n_bit port map(
        Cin=>Cin,
        X=>X,
        Y=>Y,
        S=>S,
        Gout=>Gout,
        Pout=>Pout,
        Cout=>Cout
    );

    stimproc:process
    begin
        for i in 0 to ( 2**n )-1 loop
            X8u    <=i;
            X      <=std_logic_vector( to_unsigned( i,n ) );
            for j in 0 to ( 2**n )-1 loop
                Y8u    <=j;
                Y      <= std_logic_vector( to_unsigned( j,n ) );
            wait for 2 ns;

```

```
assert( Sprocs=S );
exit when ( Sprocs/=S );
end loop;
end loop;
wait;
end process;

procs: process ( X8u,Y8u )
begin
Sprocs <= std_logic_vector( to_unsigned( ( X8u+Y8u ) mod 2**n, n ) );
end process;
end;
```

```

-- *****
-- **** STUDENT: 64200112
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Za signal Xcompare, Ycompare je smiselno (ni pa nujno – lahko imate
tudi celoten 32-bitni obseg) določiti parametriziran obseg nepredznačenih števil
integer range 0 to 2**n-1 := 0;
Povezava med cla_add_n_bit in testbench mora biti parametrizirana (manjka generic map
(n=>n)).
Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
use ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
generic( n : natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS

    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    signal          Cin : std_logic := '0';
    signal          X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal          Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    signal          S : std_logic_vector( n-1 downto 0 );
    signal          Gout : std_logic;
    signal          Pout : std_logic;
    signal          Cout : std_logic;

    signal          Xcompare, Ycompare : integer;
    signal          Scompare : std_logic_vector( n-1 downto 0 );

BEGIN

    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,
        Gout => Gout,
        Pout => Pout,
        Cout => Cout
    );

```



```

);

stim_proc: process
begin
    for i in 0 to 2**n-1 loop
        X      <= std_logic_vector( to_unsigned( i, n ) );
        Xcompare <= i;
        for j in 0 to 2**n-1 loop
            Y      <= std_logic_vector( to_unsigned( j, n ) );
            Ycompare <= j;
            wait for 10 ns;
            assert( Scompare = S ) report "Fail at: X=" & integer'image( i )
& " Y=" & integer'image( j ) severity error;
            exit when( Scompare /= S );
        end loop;
    end loop;
wait;
end process;

    comp_proc: process ( Xcompare, Ycompare )
begin
    Scompare      <= std_logic_vector( to_unsigned( ( Xcompare + Ycompare )
mod 2**n, n ) );
end process;

END;

```

```

-- *****
-- **** STUDENT: 64200163
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****

library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity cla_add_n_bit_tb is
    generic( n: natural := 8 );
end cla_add_n_bit_tb;

architecture behavior of cla_add_n_bit_tb is
    component cla_add_n_bit
    port( Cin: in std_logic;
        x: in std_logic_vector( n-1 downto 0 );
        y: in std_logic_vector( n-1 downto 0 );
        S: out std_logic_vector( n-1 downto 0 );
        Gout: out std_logic;
        Pout: out std_logic;
        Cout: out std_logic
        );
    end component;

    signal      Cin : std_logic := '0';
    signal      x  : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      y  : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      S  : std_logic_vector( n-1 downto 0 );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;

begin
    uut: cla_add_n_bit
        port map(
            Cin => Cin,          x => x,
            y => y,          S => S,          Gout => Gout,          Pout => Pout,          Cout => Cout );
    stim_proc: process
    begin
        for i in 0 to ( 2**n )-1 loop
            x <= std_logic_vector( to_unsigned( i, n ) );
            for j in 0 to ( 2**n )-1 loop
                y <= std_logic_vector( to_unsigned( j, n ) );
                wait for 10 ns;
            end loop;
        end loop;

        wait;
    end process;
end;

```

```

-- *****
-- **** STUDENT: 64200238
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Zakasnitev CLA ni 3 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
    GENERIC ( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS
    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Signals for inputs and outputs
    SIGNAL Cin : std_logic := '0';
    SIGNAL X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    SIGNAL Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    SIGNAL S : std_logic_vector( n-1 downto 0 );
    SIGNAL Gout : std_logic;
    SIGNAL Pout : std_logic;
    SIGNAL Cout : std_logic;

    signal sum : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

);

BEGIN

    -- Unit Under Test ( UUT ) instantiation
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,
        Gout => Gout,
        Pout => Pout,
        Cout => Cout
    );

    -- Stimulus process for testing
    stim_proc: process

```

```

        -- Variable to store the sum as unsigned

begin
    -- Loop over possible values for X and Y
    for j in 0 to 2**n-1 loop

        for k in 0 to 2**n-1 loop
            X      <= std_logic_vector( to_unsigned( j, n ) );    -- Assign
value to X
            Y      <= std_logic_vector( to_unsigned( k, n ) );    -- Assign value to Y

            sum    <= std_logic_vector( to_unsigned( ( to_integer( unsigned( X ) ) + to_integer(
unsigned( Y ) ) + 1 ) mod 2**n, n ) );

            assert ( sum = S ) report "NAPAKA. X: " & integer'image( j ) & "
Y: " & integer'image( k )
                severity error;

        wait for 3 ns;
        end loop;
    end loop;

    wait;
    end process;

END behavior;

```

```

-- *****
-- **** STUDENT: 64200288
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Zakasnitev CLA ni 3 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
    GENERIC ( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS
    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Signali za vhode in izhode
    signal Cin : std_logic := '0';    -- Nosilni bit ( prenos )
    signal X : std_logic_vector( n-1 downto 0 ) := ( others => '0' ); --
    Prvi_vhodni_operand
    signal Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' ); --
    Drugi_vhodni_operand
    signal S : std_logic_vector( n-1 downto 0 );    -- Izhodni vsota
    signal Gout : std_logic;    -- Generacijski bit ( za prenos )
    signal Pout : std_logic;    -- Propagacijski bit ( za prenos )
    signal Cout : std_logic;    -- Nosilni izhod
    signal sum : std_logic_vector( n-1 downto 0 ); -- Pričakovana vsota za
    preverjanje

BEGIN
    -- Instanca preizkušane enote ( UUT )
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,
        Gout => Gout,
        Pout => Pout,
        Cout => Cout
    );

    -- Proces za generiranje stimulacije

```

```

stim_proc: process
begin
    -- iteracija skozi vse mozne vrednosti za X in Y
    for Xgen in 0 to 2**n-1 loop
    for Ygen in 0 to 2**n-1 loop
        -- nastavitev testnih vrednosti
        X    <= std_logic_vector( to_unsigned( Xgen, n ) );
        Y    <= std_logic_vector( to_unsigned( Ygen, n ) );

        -- izracun pricakovane vsote
        sum  <= std_logic_vector(
            to_unsigned( ( to_integer( unsigned( X ) ) + to_integer( unsigned( Y ) ) + 1 ) mod
2**n, n )
        );

        -- pocakaj da se sistem stabilizira
        wait for 3 ns;

        -- preveri rezultat
        assert ( sum = S )
        report "NAPAKA. X: " & integer'image( Xgen ) &
        " Y: " & integer'image( Ygen ) &
        " PriÄakovana S: " & to_hstring( sum ) &
        " Dejanska S: " & to_hstring( S )
        severity error;
    end loop;
    end loop;

    wait;
end process;

END behavior;

```

```

-- *****
-- **** STUDENT: 64200296
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
use ieee.numeric_std.all;

ENTITY cla_add_n_bit_tb IS
    generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS

    -- Component Declaration for the Unit Under Test ( UUT )

    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Inputs
    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    -- Outputs
    signal      S, s_test: std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;
    signal      X_test, Y_test : integer := 0;
    -- No clocks detected in port list. Replace <clock> below with
    -- appropriate port name

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,
        Gout => Gout,

```

```

Pout => Pout,
Cout => Cout
);

-- Stimulus process
stim_proc: process
begin
    for i in 0 to ( 2**n )-1 loop
        x_test<=i;
        x      <= std_logic_vector( to_unsigned( i, n ) );
        for j in 0 to ( 2**n )-1 loop
            y_test<= j;
            y      <= std_logic_vector( to_unsigned( j, n ) );
            wait for 1 ns;
            assert ( s_test = s ) report "Error. X: " & integer'image( i ) &
" Y: " & integer'image( j ) severity error;
            end loop;
        end loop;
        wait;
    end process;

    test: process ( x_test, y_test )
    begin
        s_test<= std_logic_vector( to_unsigned( ( X_test + Y_test ) mod 2**n, n
) );
    end process;

END;

```



```

-- *****
-- **** STUDENT: 64200385
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Za signal x_int, y_int morate določiti parametriziran obseg nepredznačenih števil
-- integer range 0 to 2**n-1 := 0;.
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.all;

ENTITY cla_add_n_bit_tb IS
generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS
COMPONENT cla_add_n_bit
PORT(
Cin : IN std_logic;
X : IN std_logic_vector( n-1 downto 0 );
Y : IN std_logic_vector( n-1 downto 0 );
S : OUT std_logic_vector( n-1 downto 0 );
Gout : OUT std_logic;
Pout : OUT std_logic;
Cout : OUT std_logic
);
END COMPONENT;

-- Inputs
signal Cin : std_logic := '0';
signal X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
signal Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
signal x_int, y_int : integer range 0 to 255 := 0;

-- Outputs
signal S : std_logic_vector( n-1 downto 0 );
signal S_Comp : std_logic_vector( n-1 downto 0 );
signal Gout : std_logic;
signal Pout : std_logic;
signal Cout : std_logic;

BEGIN

-- Instantiate the Unit Under Test ( UUT )
 uut: cla_add_n_bit PORT MAP (
Cin => Cin,
X => X,
Y => Y,
S => S,
Gout => Gout,
Pout => Pout,
Cout => Cout
);

```

```

-- Stimulus process
stim_proc: process
begin
for i in 0 to ( 2**n )-1 loop
    x_int <= i;
    X      <= std_logic_vector ( to_unsigned( i,n ) );
    for j in 0 to ( 2**n )-1 loop
        y_int <=j;
        y      <= std_logic_vector ( to_unsigned( j,n ) );
        wait for 1 ns;
        assert ( S_Comp=S ) report "Fail => x:"
        &integer'image( i )& " Y:" & integer'image( j ) severity error;
        exit when ( S_Comp/=S );
    end loop;
end loop;

wait;
end process;
comp_proc:
process( x_int, y_int )
begin
S_Comp <= std_logic_vector( to_unsigned( ( x_int+y_int )mod 2**n, n ) );
end process;
END;

```

```

-- *****
-- **** STUDENT: 64210113
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n))
-- Za signal unsignX, unsignY morate določiti parametriziran obseg nepredznačenih števil
-- integer range 0 to 2**n-1 := 0;
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
    generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS
    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    signal      Cin : std_logic := '0';
    signal      X, Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      S, Rez : std_logic_vector( n-1 downto 0 );
    signal      Gout, Pout, Cout : std_logic;
    signal      unsignX, unsignY : integer range 0 to 255 := 0;

BEGIN

    uut: cla_add_n_bit
    PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,
        Gout => Gout,
        Pout => Pout,
        Cout => Cout
    );

    stim_proc: process
        variable i, j : integer := 0;

    begin

```

```

while i < ( 2**n ) loop
    unsignX    <= i;
    X    <= std_logic_vector( to_unsigned( i, n ) );
    j := 0;
    while j < ( 2**n ) loop
        unsignY    <= j;
        Y    <= std_logic_vector( to_unsigned( j, n ) );

        wait for 1 ns;

        if Rez /= S then
            report "Error x=" & integer'image( i ) & " x=" & integer'image( j )
            severity error;
        end if;
        j := j + 1;
    end loop;
    i := i + 1;
end loop;
wait;

end process;

comp_proc: process ( unsignX, unsignY )
begin
    Rez    <= std_logic_vector( to_unsigned( ( unsignX + unsignY ) mod ( 2**n ), n
    ) );
end process;

END behavior;

```

```

-- *****
-- **** STUDENT: 64210290
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial
path delay).
-- *****

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
generic( n : natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS
COMPONENT cla_add_n_bit IS
    GENERIC( N : natural := 8 );
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( 7 downto 0 );
        Y : IN std_logic_vector( 7 downto 0 );
        S : OUT std_logic_vector( 7 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
END COMPONENT;

    -- Inputs
    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( 7 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( 7 downto 0 ) := ( others => '0' );

    -- Outputs
    signal      S : std_logic_vector( 7 downto 0 );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;

    signal      clock: std_logic;
    constant    clock_period : time := 10 ns;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: cla_add_n_bit
        GENERIC MAP( n => N )
        PORT MAP (
            Cin => Cin,
            X => X,
            Y => Y,
            S => S,
            Gout => Gout,
            Pout => Pout,

```

```

Cout => Cout
);

    -- Clock process definitions
clock_process :process
begin
    clock <= '0';
    wait for clock_period/2;
    clock <= '1';
    wait for clock_period/2;
end process;

    -- Stimulus process
stim_proc: process
begin
    -- hold reset state for 100 ns.
    wait for 100 ns;

    Cin    <= '0';
    for idx in 0 to ( 2**n )-1 loop
        X    <= std_logic_vector( to_unsigned( idx,n ) );
        for idy in 0 to ( 2**n )-1 loop
            Y    <= std_logic_vector( to_unsigned( idy,n ) );
            wait for 10 ns;
        end loop;
    end loop;

    wait;
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210382
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n))
-- Za signal X_int, Y_int morate določiti parametriziran obseg nepredznačenih števil
-- integer range 0 to 2**n-1 := 0;
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
use ieee.numeric_std.all;

ENTITY cla_add_n_bit_tb IS
    generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS
    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    signal      X_int, Y_int : integer range 0 to 255 := 0;

    signal      S, S_comp : std_logic_vector( n-1 downto 0 );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;

BEGIN
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,
        Gout => Gout,
        Pout => Pout,
        Cout => Cout
    );

    stim_proc: process
        begin

```

```

        for i in 0 to ( 2**n )-1 loop
            X_int <= i;
            X      <= std_logic_vector( to_unsigned( i, n ) );
            for j in 0 to ( 2**n )-1 loop
                Y_int <= j;
                Y      <= std_logic_vector( to_unsigned( j, n ) );
                wait for 1 ns;
                assert( S_comp = S ) report "Test failed. X:" & integer'image( i
) & " Y:" & integer'image( j ) severity error;
                exit when ( S_comp /= S );
            end loop;
        end loop;
    wait;
end process;

-- Calculate sum for checking the adder result
comp_proc: process ( X_int, Y_int )
begin
    S_comp <= std_logic_vector( to_unsigned( ( X_int + Y_int ) mod 2**n, n )
);
end process;
END;

```



```

-- *****
-- **** STUDENT: 64210384
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Za signal X_int, Y_int morate določiti parametriziran obseg nepredznačenih števil
-- integer range 0 to 2**n-1 := 0;
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
    generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS
    -- Component Declaration for the Unit Under Test ( UUT )
    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Inputs
    signal Cin : std_logic := '0';
    signal X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    signal X_int, Y_int : integer range 0 to 255 := 0;

    -- Outputs
    signal S, S_com : std_logic_vector( n-1 downto 0 );
    signal Gout : std_logic;
    signal Pout : std_logic;
    signal Cout : std_logic;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,
        Y => Y,

```

```

S => S,
Gout => Gout,
Pout => Pout,
Cout => Cout );

stim_proc: process
begin
  for i in 0 to ( 2**n )-1 loop
    X_int <= i;
    X      <= std_logic_vector( to_unsigned( i,n ) );
    for j in 0 to ( 2**n )-1 loop
      Y_int <= j;
      X      <= std_logic_vector( to_unsigned( j,n ) );
      wait for 1 ns;
      assert( S_com = S ) report "Test failed. X:" & integer'image( i )
& "Y:" & integer'image( j ) severity error;
      exit when ( S_com /= S );
    end loop;
  end loop;
  wait;
end process;

comp_proc: process ( X_int, Y_int )    -- calculating the sum for checking the
adder result
begin
  S_com <= std_logic_vector( to_unsigned( ( X_int + Y_int ) mod 2**n, n )
);
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210386
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n))
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.all;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
    generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS

    -- Component Declaration for the Unit Under Test ( UUT )

    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Inputs
    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    -- Outputs
    signal      S, S_sig : std_logic_vector( n-1 downto 0 );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;

    -- No clocks detected in port list. Replace <clock> below with
    -- appropriate port name

    constant    clock_period : time := 10 ns;
    signal      Xi, Yi : integer range 0 to ( 2**n )-1 := 0;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )

```

```

 uut: cla_add_n_bit PORT MAP (
  Cin => Cin,
  X => X,
  Y => Y,
  S => S,
  Gout => Gout,
  Pout => Pout,
  Cout => Cout
 );

  -- Stimulus process
stim_proc: process
begin
    for i in 0 to ( 2**n )-1 loop
        Xi    <= i;
        X      <= std_logic_vector( to_unsigned( i, n ) );

        for u in 0 to ( 2**n )-1 loop
            Yi    <= u;
            Y      <= std_logic_vector( to_unsigned( u, n ) );
            wait for clock_period;

            assert( S_sig = S ) report "Failed X:" & integer'image( i ) & "
Y:" & integer'image( u ) severity error;
        end loop;
    end loop;

    wait;
end process;

    test_p: process ( Xi, Yi )
    begin
        S_sig <= std_logic_vector( to_unsigned( ( Xi + Yi ) mod 2**n, n ) );
    end process;

END;

```

```

-- *****
-- **** STUDENT: 64210445
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Zakasnitev CLA ni 10 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.all;

ENTITY cla_add_n_bit_tb IS
    generic( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE behavior OF cla_add_n_bit_tb IS

    -- Component Declaration for the Unit Under Test ( UUT )

    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Inputs
    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    -- Outputs
    signal      S, S_test : std_logic_vector( n-1 downto 0 );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;
    -- No clocks detected in port list. Replace <clock> below with
    -- appropriate port name

    constant    period : time := 10 ns;

    signal      Xval, Yval : integer range 0 to ( 2**n )-1 := 0;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,
        X => X,

```

```

Y => Y,
S => S,
Gout => Gout,
Pout => Pout,
Cout => Cout
);

-- Stimulus process
stim_proc: process
begin

    for i in 0 to ( 2**n )-1 loop
        Xval <= i;
        X <= std_logic_vector( to_unsigned( i, n ) );

        for j in 0 to ( 2**n )-1 loop
            Yval <= j;
            Y <= std_logic_vector( to_unsigned( j, n ) );
            wait for period;
            assert( S_test = S ) report "Fail at X:" & integer'image( i ) & "
Y:" & integer'image( j ) severity error;
            end loop;
        end loop;

    wait;
end process;

test_proc: process ( Xval, Yval )
begin
    S_test <= std_logic_vector( to_unsigned( ( Xval + Yval ) mod 2**n, n )
);
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210455
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n)).
-- Za signal A_int, B_int morate določiti parametriziran obseg nepredznačenih števil
-- integer range 0 to 2**n-1 := 0;
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY cla_add_n_bit_tb IS
    GENERIC ( n : natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE testbench OF cla_add_n_bit_tb IS

    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 DOWNT0 0 );
        Y : IN std_logic_vector( n-1 DOWNT0 0 );
        S : OUT std_logic_vector( n-1 DOWNT0 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    SIGNAL      Cin_tb : std_logic := '0';
    SIGNAL      A_tb, B_tb : std_logic_vector( n-1 DOWNT0 0 ) := ( OTHERS => '0' );
    SIGNAL      A_int, B_int : integer RANGE 0 TO 255 := 0;

    SIGNAL      Sum_tb, Expected_sum : std_logic_vector( n-1 DOWNT0 0 );
    SIGNAL      Gout_tb, Pout_tb, Cout_tb : std_logic;

BEGIN

    uut: cla_add_n_bit
    PORT MAP (
        Cin => Cin_tb,
        X => A_tb,
        Y => B_tb,
        S => Sum_tb,
        Gout => Gout_tb,
        Pout => Pout_tb,
        Cout => Cout_tb
    );

    test_process: PROCESS
    BEGIN

```

```

FOR a_val IN 0 TO ( 2**n - 1 ) LOOP
A_int<= a_val;
A_tb  <= std_logic_vector( to_unsigned( a_val, n ) );
FOR b_val IN 0 TO ( 2**n - 1 ) LOOP
B_int<= b_val;
B_tb  <= std_logic_vector( to_unsigned( b_val, n ) );
WAIT FOR 1 ns;

ASSERT ( Expected_sum = Sum_tb )
REPORT "Napaka: Sum ni pravilen za A=" & integer'image( a_val ) &
" in B=" & integer'image( b_val ) SEVERITY ERROR;
END LOOP;
END LOOP;

WAIT;
END PROCESS;

expected_sum_process: PROCESS( A_int, B_int, Cin_tb )
BEGIN

Expected_sum<= std_logic_vector( to_unsigned( ( A_int + B_int + to_integer(
unsigned( Cin_tb ) ) ) MOD ( 2**n ), n ) );
END PROCESS;

END testbench;

```



```

-- *****
-- **** STUDENT: 64240429
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Zakasnitev CLA ni 2 ns, ampak ca. 11 ns (glej vrednost combinatorial
path delay).
-- *****

LIBRARY ieee;
USE ieee.std_logic_1164.all;
use IEEE.STD_LOGIC_MISC.ALL;      -- reduction operators ( AND_REDUCE, OR_REDUCE )
use ieee.numeric_std.all;

ENTITY cla_add_n_bit_tb IS
generic ( n: natural := 8 );
END cla_add_n_bit_tb;

ARCHITECTURE NDV OF cla_add_n_bit_tb IS

COMPONENT cla_add_n_bit IS
generic( n: natural );
PORT (
    Cin      :    in    std_logic ;
    X, Y      :    in    std_logic_vector( n-1 downto 0 );
    S          :    out   std_logic_vector( n-1 downto 0 );
    Gout, Pout, Cout :    out   std_logic );
END COMPONENT;

signal      Cin      :    std_logic := '0';
signal      X, Y      :    std_logic_vector( n-1 downto 0 );
signal      S          :    std_logic_vector( n-1 downto 0 );
signal      Gout, Pout, Cout: std_logic;

BEGIN
uut: cla_add_n_bit
generic map ( n => n )
port map (
    Cin => Cin, X => X, Y => Y, S => S, Gout => Gout, Pout => Pout, Cout => Cout);

stim: process
begin
    for i in 0 to ( 2**n - 1 ) loop
        X      <= std_logic_vector( to_unsigned( i, n ) );
        for j in 0 to ( 2**n - 1 ) loop
            Y      <= std_logic_vector( to_unsigned( j, n ) );
            Cin     <= '0';

            wait for 2 ns;

            Cin     <= '1';
            wait for 2 ns;

        end loop;
    end loop;

    wait;
end process;

END NDV;

```

```

-- *****
-- **** STUDENT: 64210457
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n))
-- Za signal X_n, Y_n morate določiti parametriziran obseg nepredznačenih števil integer
-- range 0 to 2**n-1 := 0;
-- Zakasnitev CLA ni 1 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
-- n_bit additoon alu test

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
use ieee.numeric_std.all;

ENTITY alu_cla_add_n_bit_tb IS
generic( n: natural := 8 );
END alu_cla_add_n_bit_tb;

ARCHITECTURE behavior OF alu_cla_add_n_bit_tb IS

    -- Component Declaration for the Unit Under Test ( UUT )

    COMPONENT cla_add_n_bit
    PORT(
        Cin : IN std_logic;
        X : IN std_logic_vector( n-1 downto 0 );
        Y : IN std_logic_vector( n-1 downto 0 );
        S : OUT std_logic_vector( n-1 downto 0 );
        Gout : OUT std_logic;
        Pout : OUT std_logic;
        Cout : OUT std_logic
    );
    END COMPONENT;

    -- Inputs
    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    -- Outputs
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;

    -- signal      for comparing the results
    signal      S, S_check : std_logic_vector( n-1 downto 0 );
    signal      X_n, Y_n : integer range 0 to 255 := 0;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: cla_add_n_bit PORT MAP (
        Cin => Cin,

```

```

X => X,
Y => Y,
S => S,
Gout => Gout,
Pout => Pout,
Cout => Cout
);

-- Stimulus process
stim_proc: process

begin
for i in 0 to ( 2**n )-1 loop
    X_n    <= i;
    X      <= std_logic_vector( to_unsigned( i, n ) );
    for j in 0 to ( 2**n )-1 loop
        Y_n    <= j;
        Y      <= std_logic_vector( to_unsigned( j, n ) );
        wait for 1 ns;
        assert( S_check = S ) report "Test failed. X:" & integer'image( i
) & " Y:" & integer'image( j ) severity error;
        exit when ( S_check /= S );
    end loop;
end loop;
wait;
end process;

-- process for checking the result
check_proc: process ( x_n, y_n )
begin
-- sum for checking the result
S_check    <= std_logic_vector ( to_unsigned( ( X_n + Y_n ) mod 2**n, n ) );

end process;

END;

```

```

-- *****
-- **** STUDENT: 64240430
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n))
-- Zakasnitev CLA ni 5 ns, ampak ca. 11 ns (glej vrednost combinatorial path delay).
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
use ieee.numeric_std.all;

ENTITY cla_add_n_bit_tb IS
END cla_add_n_bit_tb;

ARCHITECTURE NDV OF cla_add_n_bit_tb IS

    -- constant n
    constant    n : integer := 8;    -- broj bitova

    -- Component Declaration for the Unit Under Test ( UUT )

    COMPONENT cla_add_n_bit
    PORT(
    Cin : IN std_logic;
    X : IN std_logic_vector( n-1 downto 0 );
    Y : IN std_logic_vector( n-1 downto 0 );
    S : OUT std_logic_vector( n-1 downto 0 );
    Gout : OUT std_logic;
    Pout : OUT std_logic;
    Cout : OUT std_logic
    );
    END COMPONENT;

    -- Inputs
    signal      Cin : std_logic := '0';
    signal      X : std_logic_vector( n-1 downto 0 ) := ( others => '0' );
    signal      Y : std_logic_vector( n-1 downto 0 ) := ( others => '0' );

    -- Outputs
    signal      S : std_logic_vector( n-1 downto 0 );
    signal      Gout : std_logic;
    signal      Pout : std_logic;
    signal      Cout : std_logic;

    signal      sum : std_logic_vector( n-1 downto 0 );

    -- clock
    constant    clk_period : time := 5 ns;
    signal      clk : std_logic;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: cla_add_n_bit PORT MAP (

```

```

Cin => Cin,
X => X,
Y => Y,
S => S,
Gout => Gout,
Pout => Pout,
Cout => Cout
);

-- Clock process definitions
clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

-- Stimulus process
stim_proc: process

begin
    z1: for k in 0 to 1 loop -- dvije moguće vrijednosti za Cin ( 0 in 1 )
        for i in 0 to 2**n-1 loop -- sve vrijednosti za x
            for j in 0 to 2**n-1 loop -- sve vrijednosti za y
-- postavljjanje ulaza
                x    <= std_logic_vector( to_unsigned( i, n ) );
                y    <= std_logic_vector( to_unsigned( j, n ) );
-- Cin    <= std_logic( to_unsigned( k ) ); imam tezave s ovom realizcijom

-- racunanje ocekivanog rezultata sabiranja
-- rem ali mod ( 2^n ) ne dozvoljavaju "sum" da izađe iz opsega 2^n
                sum    <= std_logic_vector( to_unsigned( ( i+j+k ) rem ( 2**n ), n
) );
-- stabilizacija izlaza
                wait for clk_period;

-- provjera: da li je "sum" jednako "S"
                assert( S = sum )
                report "Test case: X = " & integer'image( i ) &
                    ", Y = " & integer'image( j ) &
                    ", Cin = " & integer'image( k )
severity error;

                exit z1 when ( s /= sum );
            end loop; -- untrasnja petlja za y
        end loop; -- srednja petlja za x
        Cin    <= not Cin;
    end loop; -- vanjska petlja za Cin
-- zavretak simulacije
wait;
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210132
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Povezava med cla_add_n_bit in testbench mora biti parametrizirana
-- (manjka generic map (n=>n))
Ideja datoteke testnih vrednosti je, da preleti celoten obseg števil x in y ter na
drugačen način (torej z VHDL + operatorjem) preveri, če so razlike v izračunu strojne
komponente in simulatorja.
NASLEDNJIČ KODO NALOŽITE V USTREZEN RAZDELEK (OSTALO_x), KJER JE X ŠTEVILKA DOMAČE
NALOGE
-- *****

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.numeric_std.all;

ENTITY cla_add_n_bit_tb IS
generic(n: natural := 8);
END cla_add_n_bit_tb;

ARCHITECTURE sim OF cla_add_n_bit_tb IS

    component cla_add_n_bit
generic(n: natural := 8);
PORT (Cin      : in      std_logic;
      X, Y     : in      std_logic_vector(n-1 downto 0);
      S        : out     std_logic_vector(n-1 downto 0);
      Gout, Pout, Cout : out std_logic);
    end component;

    signal Cin: std_logic := '0';
    signal X, Y: std_logic_vector(n-1 downto 0) := (others => '0');
    signal S: std_logic_vector(n-1 downto 0) := (others => '0');
    signal Gout, Pout, Cout: std_logic := '0';

BEGIN

    uut: cla_add_n_bit port map (
        Cin => Cin,
        X => X,
        Y => Y,
        S => S,
        Cout => Cout,
        Gout => Gout,
        Pout => Pout
    );

    stim_proc: process
    begin

        wait for 100ns;    X <= x"05";    Y <= x"04";
        wait for 100ns;    X <= x"7F";    Y <= x"01";
        wait for 100ns;    X <= x"7F";    Y <= x"80";

        wait for 100ns;    X <= x"01";    Y <= x"80";

```

```
wait for 100ns;    X <= x"80";    Y <= x"80";  
wait for 100ns;    X <= x"7F";    Y <= x"7F";  
wait for 100ns;    X <= x"80";    Y <= x"7E";  
wait;  
end process;
```

```
END sim;
```