

Unarni operatorji po VHDL-93

Unarni operatorji po VHDL-93.....	1
-- **** STUDENT: 64000225.....	4
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Generično konstanto N ste spremenili iz 64 na 10, zato nimate pravih rezultatov sinteze v poročilu o sintezi, sicer je naloga rešena pravilno.	4
-- **** STUDENT: 64190088.....	6
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	6
-- **** STUDENT: 64200100.....	8
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	8
-- **** STUDENT: 64200112.....	10
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	10
-- **** STUDENT: 64200163.....	12
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	12
-- **** STUDENT: 64200238.....	14
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	14
-- **** STUDENT: 64200288.....	16
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	16
-- **** STUDENT: 64200296.....	18
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	18
-- **** STUDENT: 64200385.....	20
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Generično konstanto N ste spremenili iz 64 na 10, zato nimate pravih rezultatov sinteze v poročilu o sintezi, sicer je naloga rešena pravilno.	20
-- *****	20
-- **** STUDENT: 64210132.....	22
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	22
-- **** STUDENT: 64210290.....	24
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	24
-- **** STUDENT: 64210382.....	26
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	26
-- **** STUDENT: 64210384.....	28
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	28
-- **** STUDENT: 64210386.....	30

-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	30
-- **** STUDENT: 64210445.....	32
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	32
-- **** STUDENT: 64210455.....	34
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Spremenili ste ime arhitekture v "vhdl_altered_process". Spremenili ste ime generične konstante iz N na WIDTH. Napake sintetizatorja:	35
ERROR:HDLCompiler:210 - "unary_operators_vhdl93_IDEAL_tb.vhd" Line 37: Architecture vhdl_93_process not found in entity reduction_operators.....	35
ERROR:Simulator:777 - Static elaboration of top level VHDL design unit reduction_operators_tb in library work failed	35
Če oboje popravimo, vseeno ne dobimo pravih rezultatov:.....	35
Datoteka testnih vrednosti z xor, and in or unarnimi operatorji z zaporednim izračunom XOR.....	35
A, reduced_OR, reduced_AND, reduced_XOR	35
0,0,0,0.....	35
1,1,0,1	35
2,1,0,0.....	35
3,1,0,0.....	35
4,1,0,1	35
5,1,0,1	35
6,1,0,1	35
7,1,0,0.....	35
OR dela pravilno, AND bi moral pri 7 (111) biti '1', xor3 bi moral biti V(1,2,4,7), ne V(1,4,5,6). To je zato, ker sintetizator vrne opozorila o kombinacijski povratni vezavi (ang. signal(s) form a combinatorial loop), ker spremenljivke niso inicializirane znotraj procesa (result_or := '0'), ampak samo na začetku (npr. variable result_or: STD_LOGIC := '0';). Razlika je v tem, da se priredba na začetku izvede samo **ob začetku simulacije**, priredba znotraj procesa pa vsakič, ko se proces izvede – torej ko se vektor A spremeni. Po vaši rešitvi ob vsaki naslednji spremembi signala A temp_or ni inicializiran, zato opozorila. Če pa eksplicitno zahtevate, da je vrednost spremenljivke na začetku določena, potem kombinacijska povratna vezava ne more obstajati (temu se reče implicitni spomin znotraj procesa).	35
Opozorila sintetizatorja:	35
WARNING:Xst:2170 - Unit reduction_operators : the following signal(s) form a combinatorial loop: Mmux_process_or.result_or7, Mmux_process_or.result_or3, Mmux_process_or.result_or19, reduced_OR, Mmux_process_or.result_or15, Mmux_process_or.result_or5, Mmux_process_or.result_or1, Mmux_process_or.result_or14, Mmux_process_or.result_or10, Mmux_process_or.result_or18, Mmux_process_or.result_or13, Mmux_process_or.result_or, Mmux_process_or.result_or9, process_or.result_or_A[63]_OR_1_o, Mmux_process_or.result_or11,	

Mmux_process_or.result_or6, Mmux_process_or.result_or2, Mmux_process_or.result_or12, Mmux_process_or.result_or16, Mmux_process_or.result_or4, Mmux_process_or.result_or8, Mmux_process_or.result_or17.	35
WARNING:Xst:2170 - Unit reduction_operators : the following signal(s) form a combinatorial loop:	
Mmux_process_and.result_and18, Mmux_process_and.result_and2, Mmux_process_and.result_and15, Mmux_process_and.result_and4, Mmux_process_and.result_and16, process_and.result_and_A[63]_AND_1_o, Mmux_process_and.result_and12, reduced_AND, Mmux_process_and.result_and5, Mmux_process_and.result_and20, Mmux_process_and.result_and1, Mmux_process_and.result_and14, Mmux_process_and.result_and6, Mmux_process_and.result_and8, Mmux_process_and.result_and3, Mmux_process_and.result_and19, Mmux_process_and.result_and11, Mmux_process_and.result_and7, Mmux_process_and.result_and10, Mmux_process_and.result_and13, Mmux_process_and.result_and, Mmux_process_and.result_and17, Mmux_process_and.result_and9.....	35
WARNING:Xst:2170 - Unit reduction_operators : the following signal(s) form a combinatorial loop: reduced_XOR.	36
-- **** STUDENT: 64210457.....	38
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	38
-- **** STUDENT: 64240429.....	40
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	40
-- **** STUDENT: 64240430.....	42
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Generično konstanto N ste spremenili iz 64 na 10, zato nimate pravih rezultatov sinteze v poročilu o sintezi, sicer je naloga rešena pravilno.	42
-- *****.....	42
-- **** STUDENT: 64210113.....	44
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	44
-- *****.....	44
-- **** PREDLOGA VAJE.....	46
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	46

```

-- *****
-- **** STUDENT: 64000225
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Generično konstanto N ste spremenili iz 64 na 10, zato nimate pravih rezultatov sinteze v poročilu o
sintezi, sicer je naloga rešena pravilno.
-- *****

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 10 );
port (        A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin

    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );    -- ce je v A samo en element, je rezultat ta element ( or 0 )
        exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i ); -- ce je v A samo en element, je rezultat ta element ( and 1 )
        exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

```

```

xor_reduction: process ( A )
variable temp_xor: std_logic;
begin
    temp_xor := '0';
    for i in A'range loop
--      temp_xor := temp_xor xor A( i ); -- ce je v A samo en element, je rezultat ta element ( xor 0 )
        if A( i ) = '1' then
            temp_xor := not temp_xor; -- ce je v A( i ) '1' je rezultat negacija prejšnje vrednosti ( xor 0 )
        else
            temp_xor := temp_xor;      -- ce je v A( i ) '0' je rezultat enak prejšnji vrednosti ( xor 0 )
        end if;
    end loop;
    reduced_XOR <= temp_xor;
end process;

end vhd1_93_process;

```

```

-- *****
-- **** STUDENT: 64190088
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNT0 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    -- KODO VPISUJTE SEM

    or_reduction: process ( A )
        variable temp_or: std_logic;
        begin
            temp_or := '0';
            for i in A'range loop
                temp_or := temp_or or A( i );
            exit when temp_or = '1';
            end loop;
            reduced_OR    <= temp_or;
        end process;

    and_reduction: process ( A )
        variable temp_and: std_logic;
        begin
            temp_and := '1';
            for i in A'range loop
                temp_and := temp_and and A( i );
            exit when temp_and = '0';
            end loop;
            reduced_AND    <= temp_and;
        end process;

```

```
xor_reduction: process( A )  
    variable temp_xor : std_logic;  
    begin  
        temp_xor := '0';  
        for i in A'range loop  
            temp_xor := temp_xor xor A( i );  
        end loop;  
        reduced_XOR <= temp_xor;  
    end process;  
end vhd1_93_process;
```

```
-- *****
-- **** STUDENT: 64200100
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
```

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNT0 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    -- KODO VPISUJTE SEM
    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A' range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A' range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;
```



```
xor_reduction: process ( A )  
variable temp_xor: std_logic;  
begin  
temp_xor := '0';  
for i in A' range loop  
temp_xor := temp_xor xor A( i );  
end loop;  
reduced_XOR <= temp_xor;  
end process;  
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64200112
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic ( N: Natural := 4096 );
port ( A: in STD_LOGIC_VECTOR (N-1 DOWNT0 0);
       reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC);
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin

or_reduction: process (A)
variable temp_or: std_logic;
begin
    temp_or := '0';
    for i in A'range loop
        temp_or := temp_or or A(i);
        exit when temp_or = '1';
    end loop;
    reduced_OR <= temp_or;
end process;

and_reduction: process (A)
variable temp_and: std_logic;
begin
    temp_and := '1';
    for i in A'range loop
        temp_and := temp_and and A(i);
        exit when temp_and = '0';
    end loop;
    reduced_AND <= temp_and;
end process;

xor_reduction: process (A)

```

```
variable temp_xor: std_logic;  
begin  
    temp_xor := '0';  
    for i in A'range loop  
        temp_xor := temp_xor xor A(i);  
    end loop;  
    reduced_XOR <= temp_xor;  
end process;
```

```
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64200163
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic( N: Natural := 64 );      -- Matej Možek: popravil iz 4096 na 64
port( A: in STD_LOGIC_VECTOR ( N-1 DOWNT0 0 );
      reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;

architecture vhdl_93_process of reduction_operators is
begin
    or_reduction: process( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_OR <= temp_or;
    end process;

    and_reduction: process( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_AND <= temp_and;
    end process;

    xor_reduction: process( A )

```

```
variable temp_xor: std_logic;
begin
    temp_xor := A( 0 );
    for i in 1 to N-1 loop
        temp_xor := temp_xor xor A( i );
    end loop;
    reduced_XOR <= temp_xor;
end process;
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64200238
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (        A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    -- OR
    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin
        temp_or:= '0';
        for i in A' range loop
            temp_or := A( i ) or temp_or;
            exit when temp_or = '1';
        end loop;
        reduced_OR  <= temp_or;
    end process;
    -- AND
    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin
        temp_and:= '1';
        for i in A' range loop
            temp_and:= A( i ) and temp_and;
            exit when temp_and = '0';
        end loop;
        reduced_AND  <= temp_and;
    end process;
    -- XOR
    xor_reduction: process ( A )

```

```
variable temp_xor: std_logic;
begin
    temp_xor:= '0';
    for i in A' range loop
        temp_xor:= A( i ) xor temp_xor;
    end loop;
    reduced_xor <= temp_xor;
end process;

end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64200288
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    or_reduction: process ( A )
        variable trenutni_or: std_logic;
    begin
        trenutni_or:= '0';
        for i in A' range loop
            trenutni_or := A( i ) or trenutni_or;
            exit when trenutni_or = '1';
        end loop;
        reduced_OR    <= trenutni_or;
    end process;

    and_reduction: process ( A )
        variable trenutni_and: std_logic;
    begin
        trenutni_and:= '1';
        for i in A' range loop
            trenutni_and:= A( i ) and trenutni_and;
            exit when trenutni_and = '0';
        end loop;
        reduced_AND    <= trenutni_and;
    end process;

    xor_reduction: process ( A )
        variable trenutni_xor: std_logic;

```



```
begin
    trenutni_xor:= '0';
    for i in A' range loop
        trenutni_xor:= A( i ) xor trenutni_xor;
    end loop;
    reduced_xor <= trenutni_xor;
end process;

end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64200296
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    OR_reduced: process( A )
    variable temp_or: std_logic;
    begin
        temp_or:='0';
        for i in A' range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1'; -- return when the first element is 1
        end loop;
        reduced_OR    <= temp_or;
    end process;

    AND_reduced: process( A )
    variable temp_and: std_logic;
    begin
        temp_and:='1';
        for i in A' range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0'; -- exit when is no longer true
        end loop;
        reduced_AND    <= temp_and;
    end process;

    XOR_reduced: process( A )
    variable temp_xor: std_logic;

```

```
begin
    temp_xor:='0';
    for i in A' range loop
        temp_xor := temp_xor xor A( i ); -- Loop trough all the elements
    end loop;
    reduced_XOR <= temp_xor;
end process;

end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64200385
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Generično konstanto N ste spremenili iz 64 na 10, zato nimate pravih rezultatov sinteze v poročilu o
sintezi, sicer je naloga rešena pravilno.
-- *****

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 10 );
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNT0 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin

    -- OR
    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin temp_or := '0';
        for i in A' range loop
            temp_or := temp_or or A( i );    -- ce je v A samo en element, je rezultat ta element ( or 0 )
        exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    -- AND
    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin temp_and := '1';
        for i in A' range loop
            temp_and := temp_and and A( i );
        exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

```

```
-- XOR
xor_reduction: process ( A )
variable temp_xor: std_logic;
begin temp_xor := '0';
    for i in A' range loop
        temp_xor := temp_xor xor A( i );
    end loop;
    reduced_XOR <= temp_xor;
end process;

end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64210132
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (        A: in STD_LOGIC_VECTOR ( N-1 DOWNT0 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    -- KODO VPISUJTE SEM

    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );    -- ce je v A samo en element, je rezultat ta element ( or 0 )
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

```

```
xor_reduction: process ( A )  
variable temp_xor: std_logic;  
begin  
    temp_xor := '0';  
    for i in A'range loop  
        temp_xor := temp_xor xor A( i );  
    end loop;  
    reduced_XOR <= temp_xor;  
end process;  
  
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64210290
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNT0 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhdl_93_process of reduction_operators is
begin

    or_reduction: process( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_or    <= temp_or;
    end process;

    and_reduction: process( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_and    <= temp_and;
    end process;

    xor_reduction: process( A )

```



```
variable temp_xor: std_logic;  
begin  
    temp_xor := '0';  
    for i in A'range loop  
        temp_xor := temp_xor xor A( i );  
    end loop;  
    reduced_xor <= temp_xor;  
end process;  
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64210382
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    or_reduction: process ( A )
    variable temp_or: STD_LOGIC;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    and_reduction: process ( A )
    variable temp_and: STD_LOGIC;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

    xor_reduction: process ( A )
    variable temp_xor: STD_LOGIC;

```

```
begin
    temp_xor := '0';
    for i in A'range loop
        temp_xor := temp_xor xor A( i );
    end loop;
    reduced_XOR <= temp_xor;
end process;
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64210384
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin

    or_reduction: process ( A )
    variable temp_or: STD_LOGIC;
    begin
        temp_or := '0';
        for i in A' range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    and_reduction: process ( A )
    variable temp_and: STD_LOGIC;
    begin
        temp_and := '1';
        for i in A' range loop
            temp_and := temp_and and A( i );
            exit when temp_AND = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

    xor_reduction: process ( A )

```

```
variable temp_xor: STD_LOGIC;  
begin  
    temp_xor := '0';  
    for i in A' range loop  
        temp_xor := temp_xor xor A( i );  
    end loop;  
    reduced_XOR <= temp_xor;  
end process;  
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64210386
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (        A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    -- KODO VPISUJTE SEM

    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

```

```
xor_reduction: process ( A )
variable temp_xor: std_logic;
begin
    temp_xor := '0';
    for i in A'range loop
        temp_xor := temp_xor xor A( i );
    end loop;
    reduced_XOR <= temp_xor;
end process;

end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64210445
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNT0 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;

architecture vhdl_93_process of reduction_operators is
begin
    -- KODO VPISUJTE SEM

    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

```



```
xor_reduction: process ( A )
variable temp_xor: std_logic;
begin
    temp_xor := '0';
    for i in A'range loop
-- if A( i ) = '1' then
-- temp_xor := NOT( temp_xor );
-- end if;
        temp_xor := temp_xor xor A( i );
    end loop;
    reduced_XOR <= temp_xor;
end process;

end vhd1_93_process;
```

```
-- *****  
-- **** STUDENT: 64210455  
-- *****
```

-- KOMENTARJI K OCENI NALOGE

-- Matej Možek: Spremenili ste ime arhitekture v "vhdl_altered_process".

Spremenili ste ime generične konstante iz N na WIDTH.

Napake sintetizatorja:

ERROR:HDLCompiler:210 - "unary_operators_vhdl93_IDEAL_tb.vhd" Line 37: Architecture vhdl_93_process not found in entity reduction_operators

ERROR:Simulator:777 - Static elaboration of top level VHDL design unit reduction_operators_tb in library work failed

Če oboje popravimo, vseeno ne dobimo pravih rezultatov:

Datoteka testnih vrednosti z xor, and in or unarnimi operatorji z zaporednim izračunom XOR

A, reduced_OR, reduced_AND, reduced_XOR

0,0,0,0

1,1,0,1

2,1,0,0

3,1,0,0

4,1,0,1

5,1,0,1

6,1,0,1

7,1,0,0

OR dela pravilno, AND bi moral pri 7 (111) biti '1', xor3 bi moral biti V(1,2,4,7), ne V(1,4,5,6).

To je zato, ker sintetizator vrne opozorila o kombinacijski povratni vezavi (ang. signal(s) form a combinatorial loop), ker spremenljivke niso inicializirane znotraj procesa (result_or := '0'), ampak samo na začetku (npr. variable result_or: STD_LOGIC := '0';). Razlika je v tem, da se priredba na začetku izvede samo **ob začetku simulacije**, priredba znotraj procesa pa vsakič, ko se proce izvede – torej ko se vektor A spremeni. Po vaši rešitvi ob vsaki naslednji spremembi signala A temp_or ni inicializiran, zato opozorila. Če pa eksplicitno zahtevate, da je vrednost spremenljivke na začetku določena, potem kombinacijska povratna vezava ne more obstajati (temu se reče implicitni spomin znotraj procesa).

Opozorila sintetizatorja:

WARNING:Xst:2170 - Unit reduction_operators : the following signal(s) form a combinatorial loop:

Mmux_process_or.result_or7, Mmux_process_or.result_or3, Mmux_process_or.result_or19, reduced_OR,
Mmux_process_or.result_or15, Mmux_process_or.result_or5, Mmux_process_or.result_or1, Mmux_process_or.result_or14,
Mmux_process_or.result_or10, Mmux_process_or.result_or18, Mmux_process_or.result_or13, Mmux_process_or.result_or,
Mmux_process_or.result_or9, process_or.result_or_A[63]_OR_1_o, Mmux_process_or.result_or11,
Mmux_process_or.result_or6, Mmux_process_or.result_or2, Mmux_process_or.result_or12, Mmux_process_or.result_or16,
Mmux_process_or.result_or4, Mmux_process_or.result_or8, Mmux_process_or.result_or17.

WARNING:Xst:2170 - Unit reduction_operators : the following signal(s) form a combinatorial loop:

Mmux_process_and.result_and18, Mmux_process_and.result_and2, Mmux_process_and.result_and15,
Mmux_process_and.result_and4, Mmux_process_and.result_and16, process_and.result_and_A[63]_AND_1_o,

```

Mmux_process_and.result_and12, reduced_AND, Mmux_process_and.result_and5, Mmux_process_and.result_and20,
Mmux_process_and.result_and1, Mmux_process_and.result_and14, Mmux_process_and.result_and6,
Mmux_process_and.result_and8, Mmux_process_and.result_and3, Mmux_process_and.result_and19,
Mmux_process_and.result_and11, Mmux_process_and.result_and7, Mmux_process_and.result_and10,
Mmux_process_and.result_and13, Mmux_process_and.result_and, Mmux_process_and.result_and17,
Mmux_process_and.result_and9.
WARNING:Xst:2170 - Unit reduction_operators : the following signal(s) form a combinatorial loop: reduced_XOR.
-- *****

```

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
  generic ( WIDTH: Natural := 64 );      -- Matej Možek: popravil iz 4096 na 64
  port (
    A: in STD_LOGIC_VECTOR ( WIDTH-1 DOWNT0 0 );
    reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC
  );
end reduction_operators;

architecture vhdl_altered_process of reduction_operators is
begin

  process_or: process ( A )
    variable result_or: STD_LOGIC := '0';
  begin
    for idx in A'range loop
      result_or := result_or or A( idx );
      if result_or = '1' then
        exit;
      end if;
    end loop;
    reduced_OR <= result_or;
  end process process_or;

  process_and: process ( A )
    variable result_and: STD_LOGIC := '1';
  begin
    for idx in A'range loop
      result_and := result_and and A( idx );
      if result_and = '0' then

```

```
exit;
end if;
end loop;
reduced_AND <= result_and;
end process process_and;

process_xor: process ( A )
variable result_xor: STD_LOGIC := '0';
begin
for idx in A'range loop
result_xor := result_xor xor A( idx );
end loop;
reduced_XOR <= result_xor;
end process process_xor;

end vhdl_altered_process;
```

```

-- *****
-- **** STUDENT: 64210457
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 ); -- Matej Možek: popravil iz 4096 na 64
port (        A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
-- 1. REALIZACIJA REDUCIRANIH OPERATORJEV Z PROCESNIM STAVKOM
-- realizacija z dvoVhodnimi vrati

-- or reduction
or_reduction: process ( A )
variable temp_or: std_logic;
begin
temp_or := '0';    -- inicializacija na 0
for i in A'range loop
temp_or := temp_or or A( i );
exit when temp_or = '1';
end loop;
reduced_OR    <= temp_or;
end process;

-- and reduction

and_reduction: process ( A )
variable temp_and: std_logic;
begin
temp_and := '1';    -- inicializacija na 1
for i in A'range loop
temp_and := temp_and and A( i );
exit when temp_and = '0';

```

```
end loop;  
reduced_AND <= temp_and;  
end process;
```

-- xor reduction

```
xor_reduction: process ( A )  
variable temp_xor: std_logic;  
begin  
temp_xor := '0';    -- 1 ti flipa bit 0 ti ga pusti  
for i in A'range loop  
temp_xor := temp_xor xor A( i );  
end loop;    -- ce je st enic liho je rezultat xor na vektorju 1, ce je sodo pa 0  
reduced_XOR <= temp_xor;  
end process;  
  
end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64240429
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 50 );
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;

architecture vhdl_93_process of reduction_operators is
begin

    -- OR operator
    or_reduction: process ( A )
    variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );    -- ce je v A samo en element, je rezultat ta element ( or 0 )
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    -- AND operator
    and_reduction: process ( A )
    variable temp_and: std_logic;
    begin
        temp_and := '1';    -- start at 1
        for i in A'range loop
            temp_and := temp_and and A( i ); -- if current A is not 1, then 0, else 1
            exit when temp_and = '0'; -- if at least one 0 is met, the result is 0
        end loop;
        reduced_AND    <= temp_and;
    end process;
end architecture;

```



```

end process;

-- XOR operator
xor_reduction: process ( A )    -- output 1 if there is an odd number of 1's on the input
variable temp_xor: std_logic;
begin
    temp_xor := '0';
    for i in A'range loop
--      -- with A( i ) select
--      temp_or := not temp_or when '1',    -- invert the value when there is a 1
--      temp_or when '0',    -- the value remains the same if there is a 0
--      'X' when others;
-- to comply with the way the task is supposed to be synthesized
        temp_xor := temp_xor xor A( i );
    end loop;
    reduced_XOR <= temp_xor;
end process;

end vhd1_93_process;

```

```

-- *****
-- **** STUDENT: 64240430
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Generično konstanto N ste spremenili iz 64 na 10, zato nimate pravih rezultatov sinteze v poročilu o
sintezi, sicer je naloga rešena pravilno.
-- *****

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 10 );
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin
    -- OR_REDUCTION
    or_reduction : process ( A )
        variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );
            exit when temp_or = '1';
        end loop;
        reduced_OR    <= temp_or;
    end process;

    -- AND_REDUCTION
    and_reduction : process ( A )
        variable temp_and: std_logic;
    begin
        temp_and := '1';
        for i in A'range loop
            temp_and := temp_and and A( i );
            exit when temp_and = '0';
        end loop;
        reduced_AND    <= temp_and;
    end process;

```

```
-- XOR_REDUCTION
xor_reduction : process ( A )
variable temp_xor: std_logic;
begin
    temp_xor := '0';
    for i in A'range loop
        temp_xor := temp_xor xor A( i );
    end loop;
    reduced_XOR <= temp_xor;
end process;

end vhd1_93_process;
```

```

-- *****
-- **** STUDENT: 64210113
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****

```

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic ( N: Natural := 64 );
port ( A: in STD_LOGIC_VECTOR (N-1 DOWNT0 0);
       reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC);
end reduction_operators;
architecture vhd1_93_process of reduction_operators is
begin

```

```

-----OR
or_reduction: process (A)

    variable temp_or: std_logic;

begin
    temp_or := '0';
    for i in A'range loop
        temp_or := temp_or or A(i);
        exit when temp_or = '1';
    end loop;

    reduced_OR <= temp_or;

end process;

```

```

-----AND
and_reduction: process (A)

    variable temp_and: std_logic;

begin

```

```

    temp_and := '1';
    for i in A'range loop
        temp_and := temp_and and A(i);
        exit when temp_and = '0';
    end loop;

    reduced_AND <= temp_and;

end process;

-----XOR
xor_reduction: process (A)

    variable temp_xor: std_logic;

begin
    temp_xor := '0';
    for i in A'range loop
        temp_xor := temp_xor xor A(i);
    end loop;

    reduced_XOR <= temp_xor;

end process;

end vhd1_93_process;

```

```

-- *****
-- **** PREDLOGA VAJE
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity reduction_operators is
generic (      N: Natural := 64 );
port (      A: in STD_LOGIC_VECTOR ( N-1 DOWNTO 0 );
        reduced_OR, reduced_AND, reduced_XOR: out STD_LOGIC );
end reduction_operators;

architecture vhd1_93_comb of reduction_operators is begin
    reduced_OR   <= '0' when ( A = ( A'range => '0' ) ) else '1';    -- redukcijski OR
    reduced_AND  <= '1' when ( A = ( A'range => '1' ) ) else '0';    -- redukcijski AND
end vhd1_93_comb;

-- architecture vhd1_2002_built_in of reduction_operators is begin
-- reduced_OR<= or_reduce( A );
-- reduced_AND    <= and_reduce( A );
-- reduced_XOR    <= xor_reduce( A );
-- end vhd1_2002_built_in;

architecture vhd1_93_process of reduction_operators is
begin
    or_reduction: process ( A )
        variable temp_or: std_logic;
    begin
        temp_or := '0';
        for i in A'range loop
            temp_or := temp_or or A( i );    -- ce je samo en element, je rezultat ta element ( or 0 )
            exit when temp_or = '1';
        end loop;
        reduced_OR   <= temp_or;
    end process;

    and_reduction: process ( A )

```

```

variable temp_and: std_logic := '1';
begin
    temp_and := '1';
    for j in A'range loop
        temp_and := temp_and and A( j ); -- ce je samo en element, je rezultat ta element ( and 1 )
        exit when temp_and = '0';
    end loop;
    reduced_AND <= temp_and;
end process;

xor_reduction: process ( A )
variable temp_xor: std_logic;
begin
    temp_xor := '0';
    for k in A'range loop
        temp_xor := temp_xor xor A( k );
    end loop;
    reduced_XOR <= temp_xor;
end process;
end vhd1_93_process;

```

