Assignment 5

MAC CIRCUIT

Mohit , 20D070052

The circuit is shown in the below figure:

Elemental Logic Blocks :

Roll No. - 20D070052
Last Digit - 2
AND - 42ps
XOR - 54ps
A+B.C - 64ps
A.B + C.(A+B) - 64ps
3-carry = 2*XOR - 2*54ps - 108ps
3-SUM - A.B + C.(A+B) - 64ps

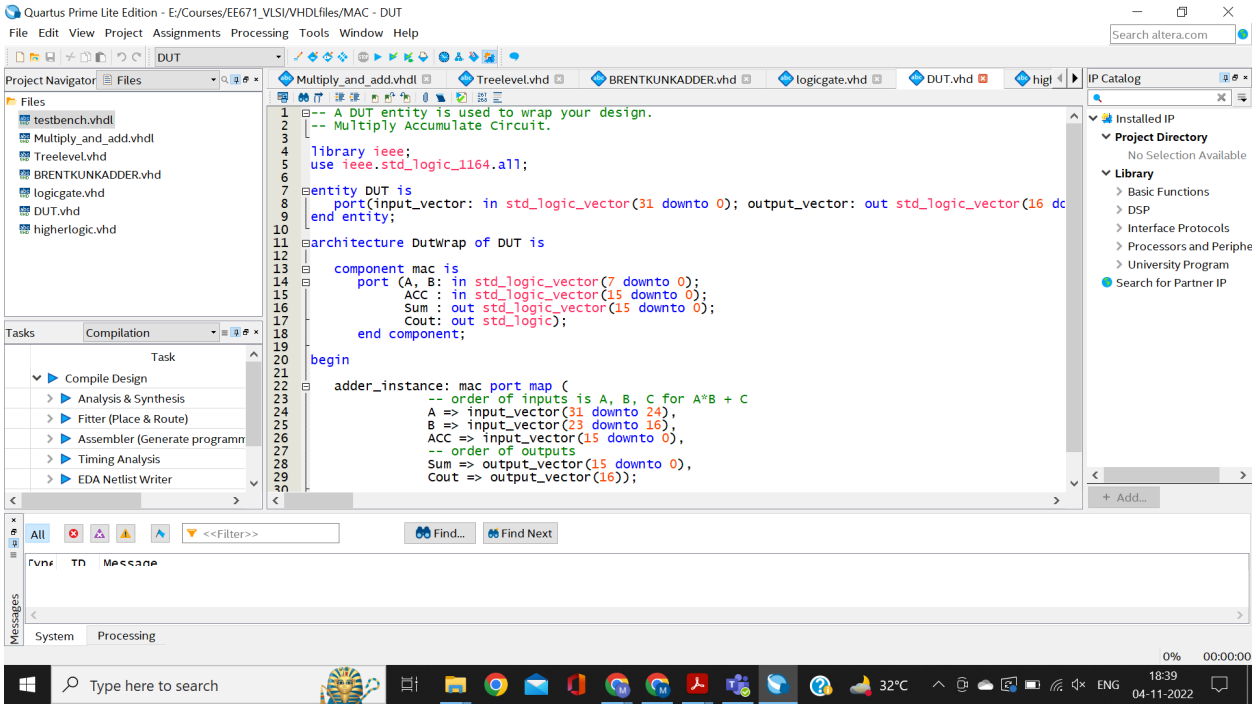
• We generate the various orders of G and P values using the logic blocks in the previous part. Also the brent kung adder is taken from the previous assignment directly.

• We write the VHDL code that reads a pair of 8-bit numbers along with a 16 bit number that it will add to the product of the two 8 bit numbers and generates the expected 16-bit sum and 1 bit output carry using a MAC Circuit

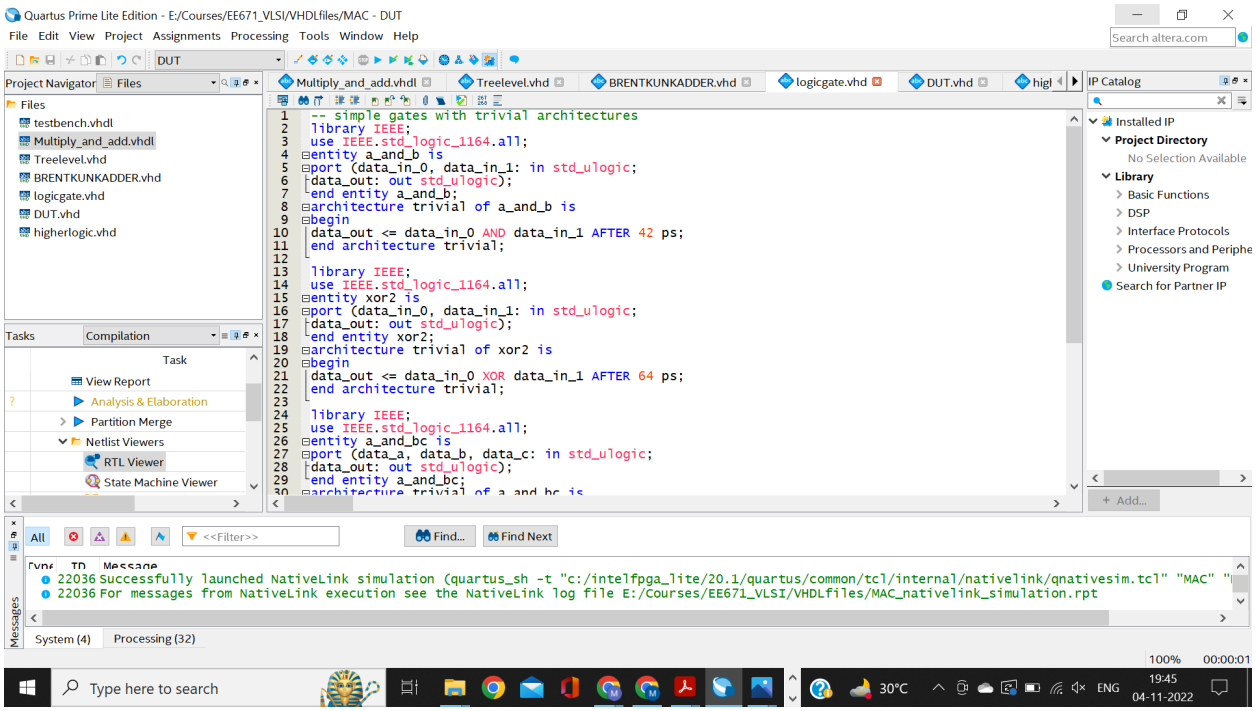
• The MAC Circuit defined using VHDL, is simulated using ModelSim considering 10 randomly chosen pairs of 8-bit numbers and 16-bit input that needs to be summed with the product.

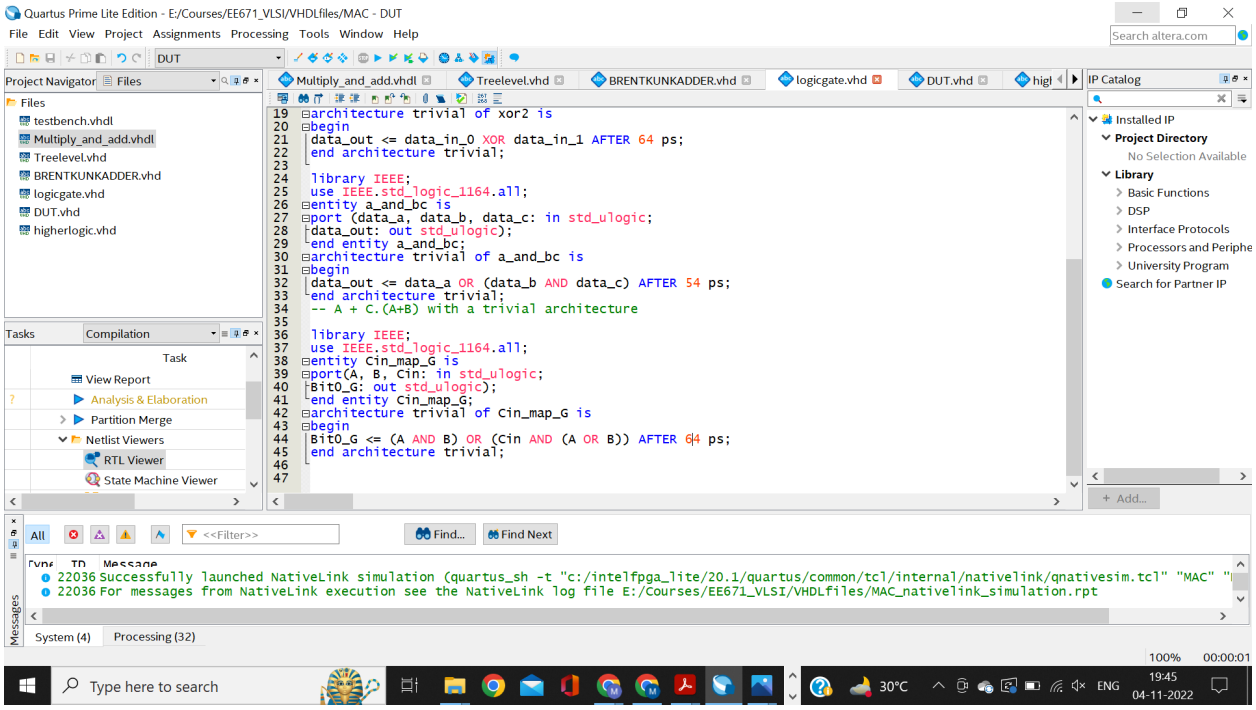
Code :

DUT FILE:

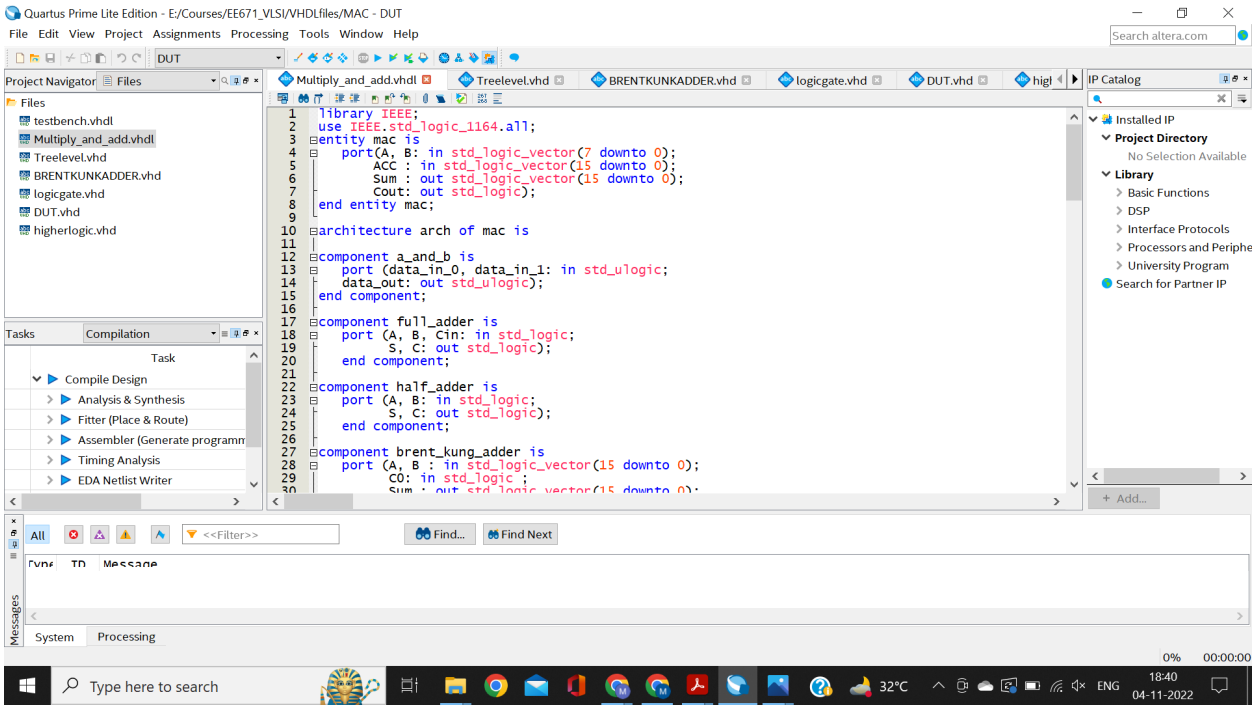


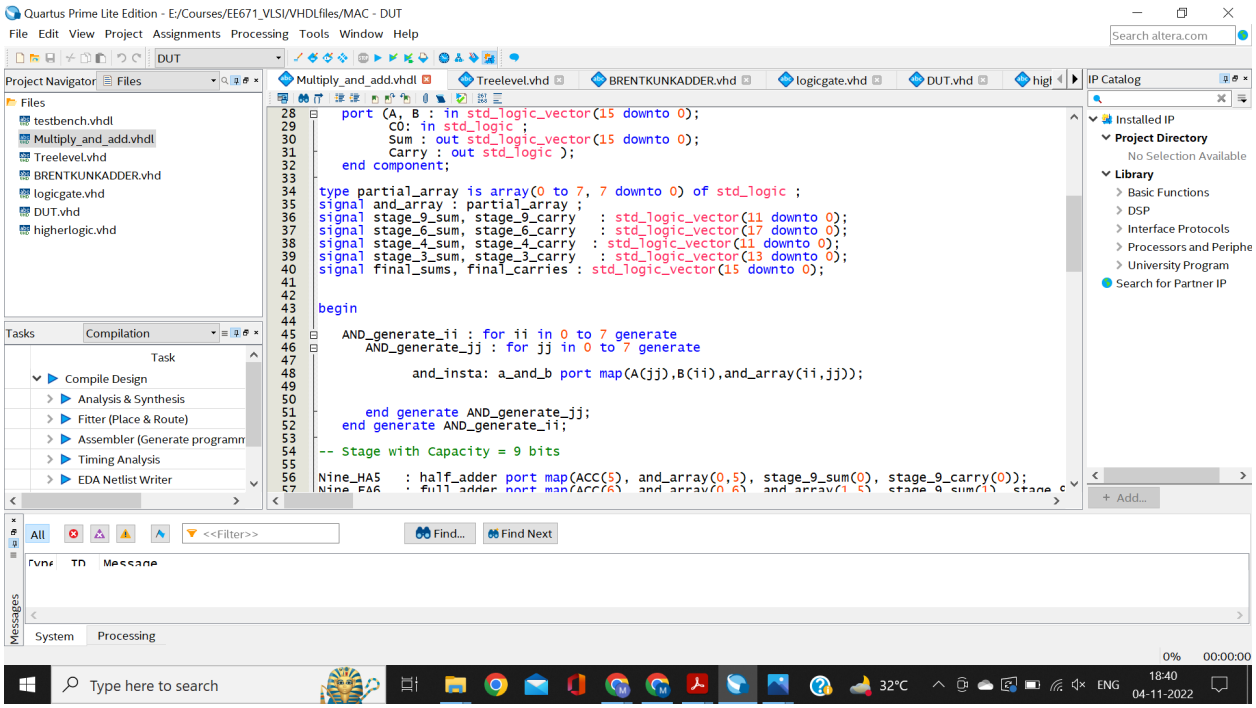
GATES INCLUDING AND , XOR ETC. :

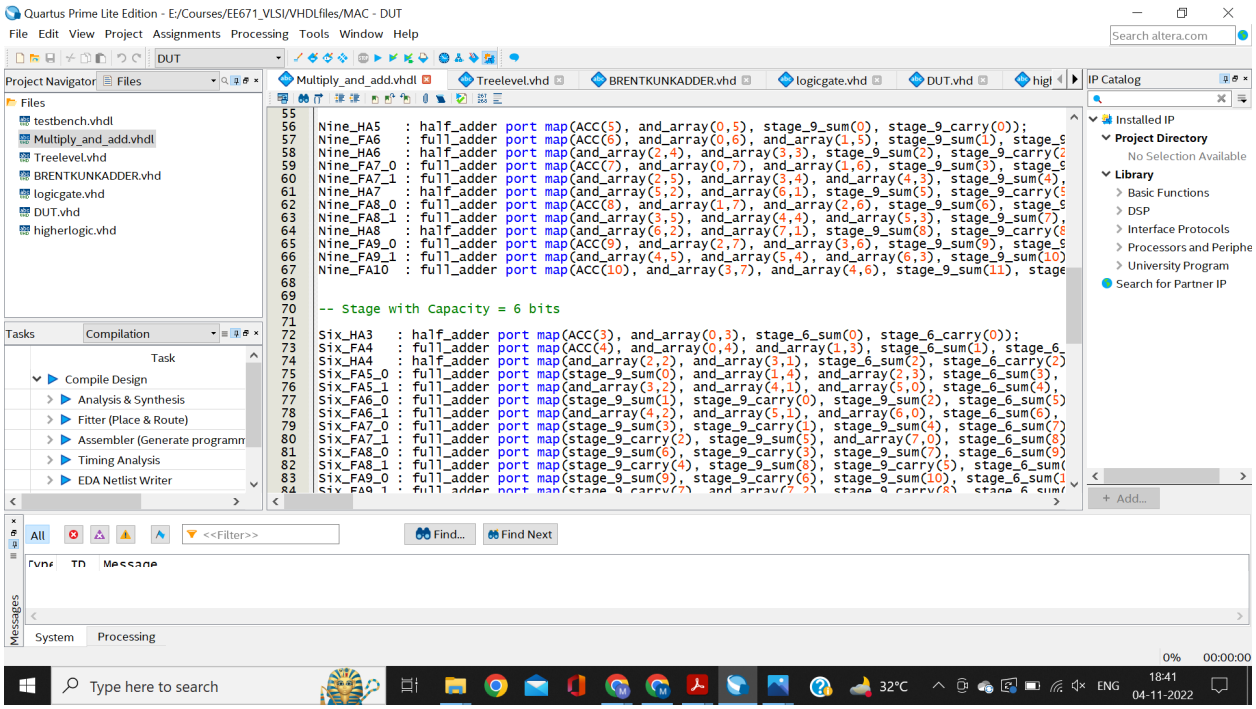


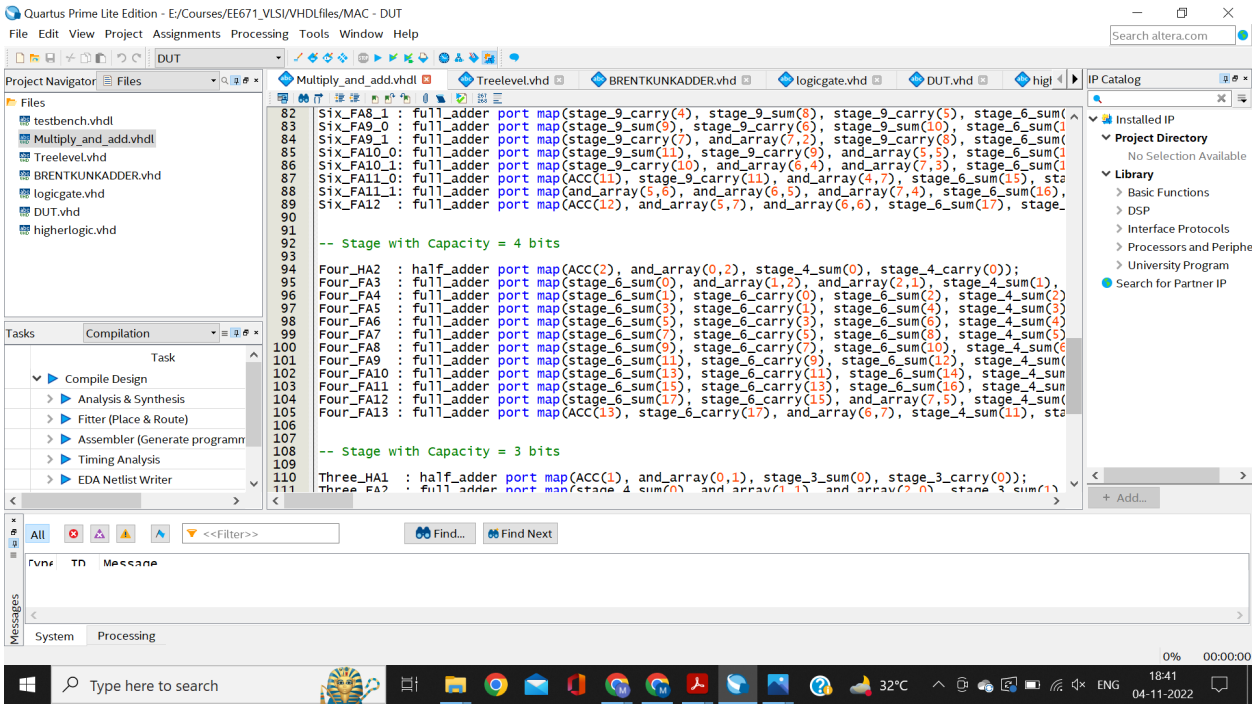


FINAL MAC CIRCUIT :

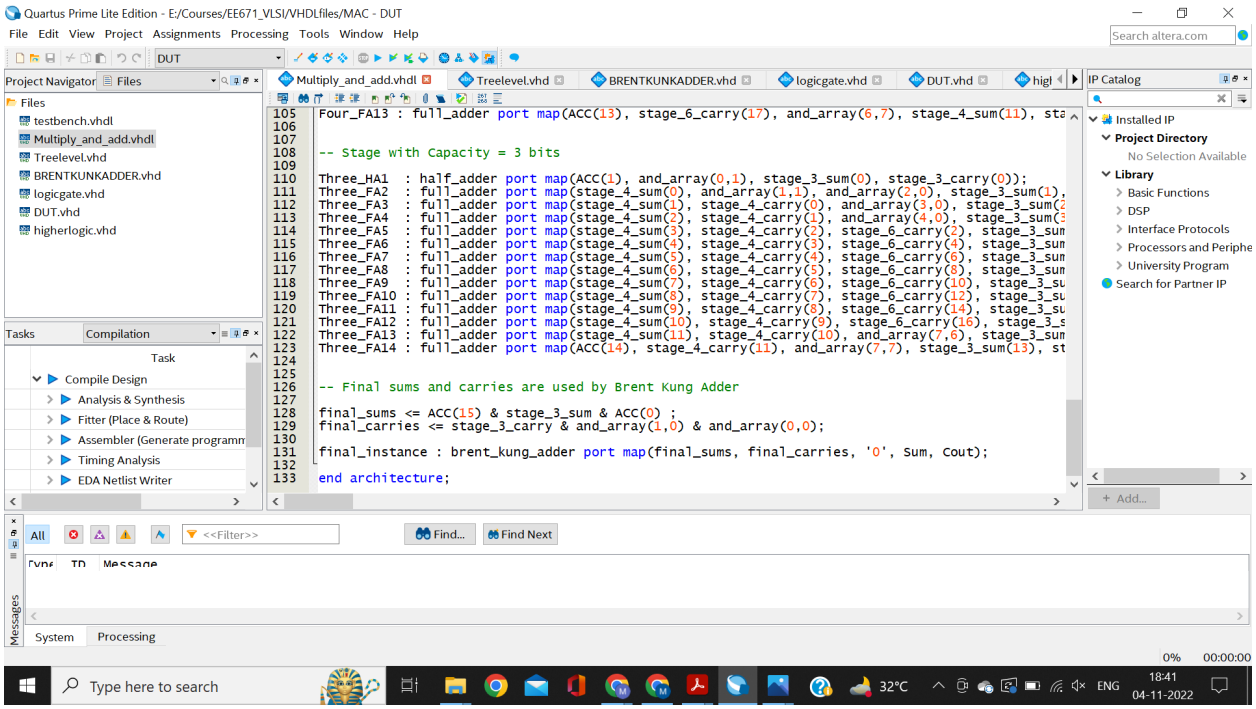



Quartus Prime Lite Edition - E:/Courses/EE671_VLSI/VHDLfiles/MAC - DUT
File Edit View Project Assignments Processing Tools Window Help
Search altera.com
Project Navigator Files
Files
testbench.vhdl
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
higherlogic.vhd
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
IP Catalog
Installed IP
Project Directory
No Selection Available
Library
Basic Functions
DSP
Interface Protocols
Processors and Periphe
University Program
Search for Partner IP
Tasks
Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Assembler (Generate programm
Timing Analysis
EDA Netlist Writer
port (A, B : in std_logic_vector(15 downto 0);
CO: in std_logic ;
Sum : out std_logic_vector(15 downto 0);
Carry : out std_logic);
end component;
type partial_array is array(0 to 7, 7 downto 0) of std_logic ;
signal and_array : partial_array ;
signal stage_9_sum, stage_9_carry : std_logic_vector(11 downto 0);
signal stage_6_sum, stage_6_carry : std_logic_vector(17 downto 0);
signal stage_4_sum, stage_4_carry : std_logic_vector(11 downto 0);
signal stage_3_sum, stage_3_carry : std_logic_vector(13 downto 0);
signal final_sums, final_carries : std_logic_vector(15 downto 0);
begin
AND_generate_ii : for ii in 0 to 7 generate
AND_generate_jj : for jj in 0 to 7 generate
and_insta: a_and_b port map(A(jj),B(ii),and_array(ii,jj));
end generate AND_generate_jj;
end generate AND_generate_ii;
-- Stage with Capacity = 9 bits
Nine_HA5 : half_adder port map(ACC(5), and_array(0,5), stage_9_sum(0), stage_9_carry(0));
All
<<Filter>>
Find...
Find Next
Type
ID
Message
System
Processing
0%
00:00:00
Type here to search
32°C
ENG
18:40
04-11-2022


Quartus Prime Lite Edition - E:/Courses/EE671_VLSI/VHDLfiles/MAC - DUT
File Edit View Project Assignments Processing Tools Window Help
Search altera.com
Project Navigator Files
Files
testbench.vhdl
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
higherlogic.vhd
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
IP Catalog
Installed IP
Project Directory
No Selection Available
Library
Basic Functions
DSP
Interface Protocols
Processors and Periphe
University Program
Search for Partner IP
Tasks
Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Assembler (Generate programm
Timing Analysis
EDA Netlist Writer
Nine_HA5 : half_adder port map(ACC(5), and_array(0,5), stage_9_sum(0), stage_9_carry(0));
Nine_FA6 : full_adder port map(ACC(6), and_array(0,6), and_array(1,5), stage_9_sum(1),
Nine_HA6 : half_adder port map(and_array(2,4), and_array(3,3), stage_9_sum(2),
Nine_FA7_0 : full_adder port map(ACC(7), and_array(0,7), and_array(1,6), stage_9_sum(3),
Nine_FA7_1 : full_adder port map(and_array(2,5), and_array(3,4), and_array(4,3), stage_9_sum(4),
Nine_HA7 : half_adder port map(and_array(5,2), and_array(6,1), stage_9_sum(5),
Nine_FA8_0 : full_adder port map(ACC(8), and_array(1,7), and_array(2,6), stage_9_sum(6),
Nine_FA8_1 : full_adder port map(and_array(3,5), and_array(4,4), and_array(5,3), stage_9_sum(7),
Nine_HA8 : half_adder port map(and_array(6,2), and_array(7,1), stage_9_sum(8),
Nine_FA9_0 : full_adder port map(ACC(9), and_array(2,7), and_array(3,6), stage_9_sum(9),
Nine_FA9_1 : full_adder port map(and_array(4,5), and_array(5,4), and_array(6,3), stage_9_sum(10)
Nine_FA10 : full_adder port map(ACC(10), and_array(3,7), and_array(4,6), stage_9_sum(11),
-- Stage with Capacity = 6 bits
Six_HA3 : half_adder port map(ACC(3), and_array(0,3), stage_6_sum(0), stage_6_carry(0));
Six_FA4 : full_adder port map(ACC(4), and_array(0,4), and_array(1,3), stage_6_sum(1),
Six_HA4 : half_adder port map(and_array(2,2), and_array(3,1), stage_6_sum(2), stage_6_carry(2)
Six_FA5_0 : full_adder port map(stage_9_sum(0), and_array(1,4), and_array(2,3), stage_6_sum(3),
Six_FA5_1 : full_adder port map(and_array(3,2), and_array(4,1), and_array(5,0), stage_6_sum(4),
Six_FA6_0 : full_adder port map(stage_9_sum(1), stage_9_carry(0), stage_9_sum(2), stage_6_sum(5)
Six_FA6_1 : full_adder port map(and_array(4,2), and_array(5,1), and_array(6,0), stage_6_sum(6),
Six_FA7_0 : full_adder port map(stage_9_sum(3), stage_9_carry(1), stage_9_sum(4), stage_6_sum(7)
Six_FA7_1 : full_adder port map(stage_9_carry(2), stage_9_sum(5), and_array(7,0), stage_6_sum(8)
Six_FA8_0 : full_adder port map(stage_9_sum(6), stage_9_carry(3), stage_9_sum(7), stage_6_sum(9)
Six_FA8_1 : full_adder port map(stage_9_carry(4), stage_9_sum(8), stage_9_carry(5), stage_6_sum(
Six_FA9_0 : full_adder port map(stage_9_sum(9), stage_9_carry(6), stage_9_sum(10), stage_6_sum(1
All
<<Filter>>
Find...
Find Next
Type
ID
Message
System
Processing
0%
00:00:00
Type here to search
32°C
ENG
18:41
04-11-2022

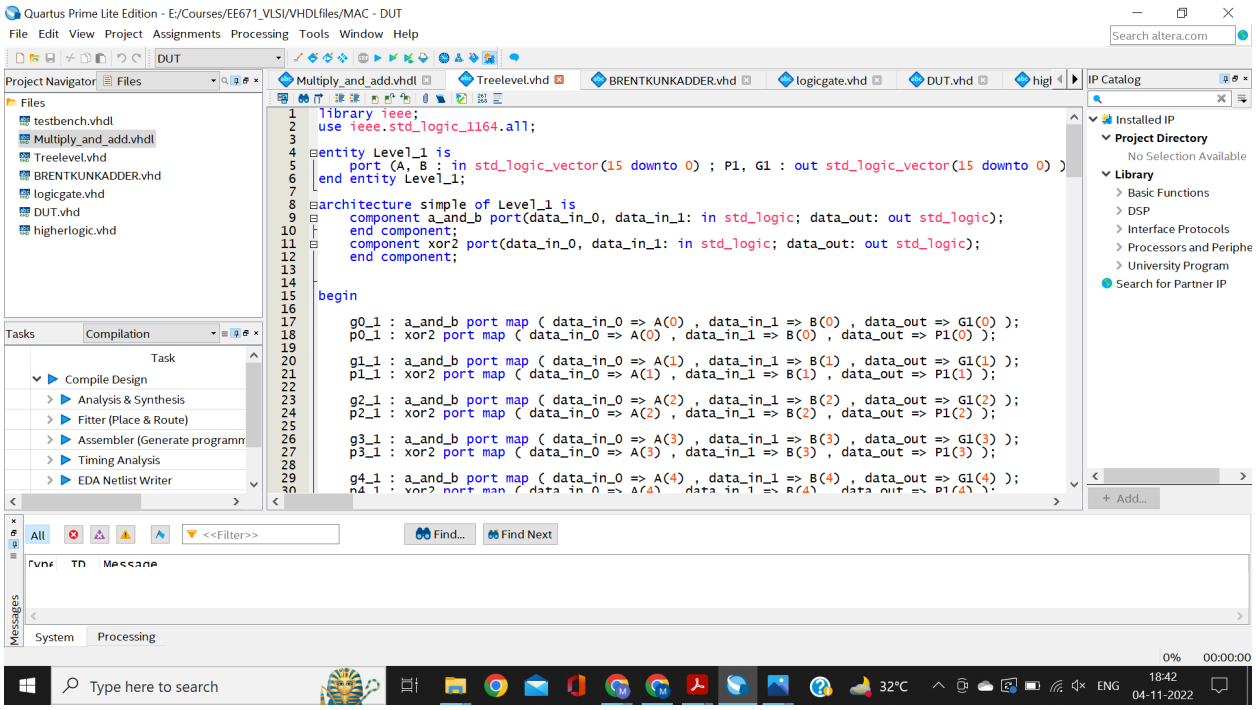



```
82  Six_FA8_1 : full_adder port map(stage_9_carry(4), stage_9_sum(8), stage_9_carry(5), stage_6_sum(
83  Six_FA9_0 : full_adder port map(stage_9_sum(9), stage_9_carry(6), stage_9_sum(10), stage_6_sum(1
84  Six_FA9_1 : full_adder port map(stage_9_carry(7), and_array(7,2), stage_9_carry(8), stage_6_sum(
85  Six_FA10_0: full_adder port map(stage_9_sum(11), stage_9_carry(9), and_array(5,5), stage_6_sum(1
86  Six_FA10_1: full_adder port map(stage_9_carry(10), and_array(6,4), and_array(7,3), stage_6_sum(1
87  Six_FA11_0: full_adder port map(ACC(11), stage_9_carry(11), and_array(4,7), stage_6_sum(15), sta
88  Six_FA11_1: full_adder port map(and_array(5,6), and_array(6,5), and_array(7,4), stage_6_sum(16),
89  Six_FA12  : full_adder port map(ACC(12), and_array(5,7), and_array(6,6), stage_6_sum(17), stage_
90
91
92  -- Stage with Capacity = 4 bits
93
94  Four_HA2  : half_adder port map(ACC(2), and_array(0,2), stage_4_sum(0), stage_4_carry(0));
95  Four_FA3  : full_adder port map(stage_6_sum(0), and_array(1,2), and_array(2,1), stage_4_sum(1),
96  Four_FA4  : full_adder port map(stage_6_sum(1), stage_6_carry(0), stage_6_sum(2), stage_4_sum(2)
97  Four_FA5  : full_adder port map(stage_6_sum(3), stage_6_carry(1), stage_6_sum(4), stage_4_sum(3)
98  Four_FA6  : full_adder port map(stage_6_sum(5), stage_6_carry(3), stage_6_sum(6), stage_4_sum(4)
99  Four_FA7  : full_adder port map(stage_6_sum(7), stage_6_carry(5), stage_6_sum(8), stage_4_sum(5)
100 Four_FA8  : full_adder port map(stage_6_sum(9), stage_6_carry(7), stage_6_sum(10), stage_4_sum(6
101 Four_FA9  : full_adder port map(stage_6_sum(11), stage_6_carry(9), stage_6_sum(12), stage_4_sum(
102 Four_FA10 : full_adder port map(stage_6_sum(13), stage_6_carry(11), stage_6_sum(14), stage_4_sum
103 Four_FA11 : full_adder port map(stage_6_sum(15), stage_6_carry(13), stage_6_sum(16), stage_4_sum
104 Four_FA12 : full_adder port map(stage_6_sum(17), stage_6_carry(15), and_array(7,5), stage_4_sum(
105 Four_FA13 : full_adder port map(ACC(13), stage_6_carry(17), and_array(6,7), stage_4_sum(11), sta
106
107
108 -- Stage with Capacity = 3 bits
109
110 Three_HA1  : half_adder port map(ACC(1), and_array(0,1), stage_3_sum(0), stage_3_carry(0));
111 Three_FA2  : full_adder port map(stage_4_sum(0), and_array(1,1), and_array(2,0), stage_3_sum(1)
```

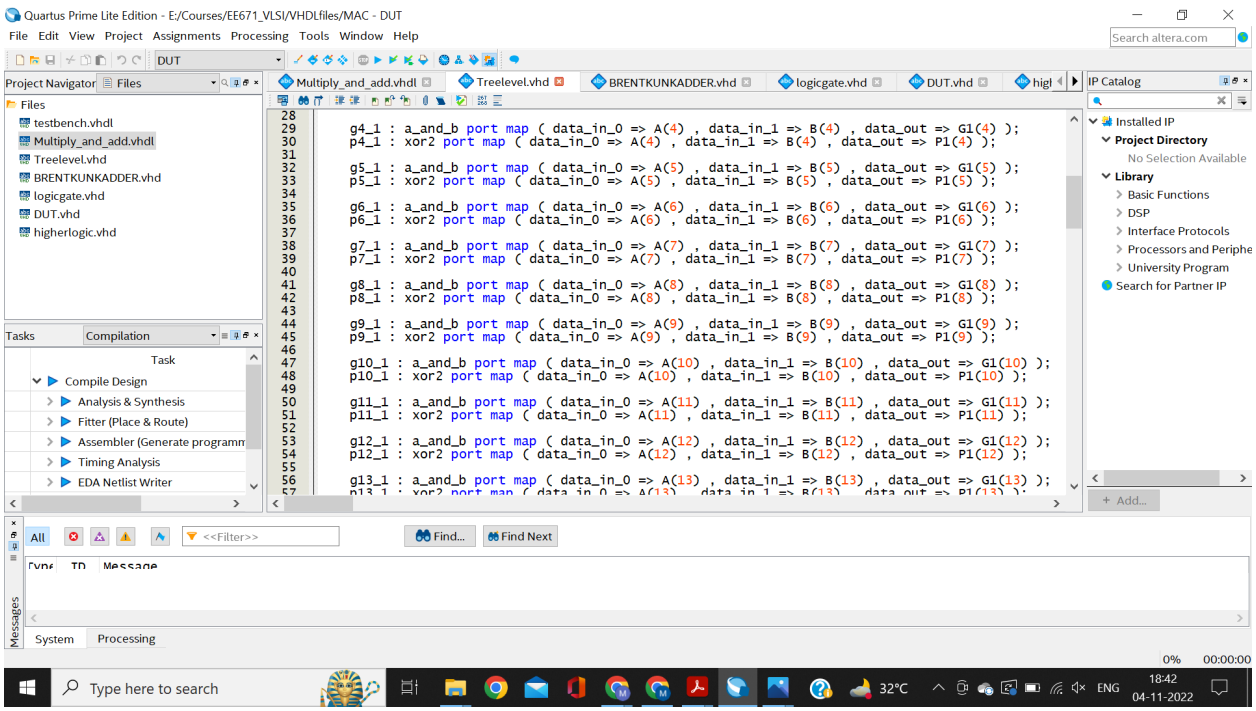



```
105 Four_FA13 : full_adder port map(ACC(13), stage_6_carry(17), and_array(6,7), stage_4_sum(11), sta
106
107
108 -- Stage with Capacity = 3 bits
109
110 Three_HA1  : half_adder port map(ACC(1), and_array(0,1), stage_3_sum(0), stage_3_carry(0));
111 Three_FA2  : full_adder port map(stage_4_sum(0), and_array(1,1), and_array(2,0), stage_3_sum(1),
112 Three_FA3  : full_adder port map(stage_4_sum(1), stage_4_carry(0), and_array(3,0), stage_3_sum(2
113 Three_FA4  : full_adder port map(stage_4_sum(2), stage_4_carry(1), and_array(4,0), stage_3_sum(3
114 Three_FA5  : full_adder port map(stage_4_sum(3), stage_4_carry(2), stage_6_carry(2), stage_3_sum
115 Three_FA6  : full_adder port map(stage_4_sum(4), stage_4_carry(3), stage_6_carry(4), stage_3_sum
116 Three_FA7  : full_adder port map(stage_4_sum(5), stage_4_carry(4), stage_6_carry(6), stage_3_sum
117 Three_FA8  : full_adder port map(stage_4_sum(6), stage_4_carry(5), stage_6_carry(8), stage_3_sum
118 Three_FA9  : full_adder port map(stage_4_sum(7), stage_4_carry(6), stage_6_carry(10), stage_3_su
119 Three_FA10 : full_adder port map(stage_4_sum(8), stage_4_carry(7), stage_6_carry(12), stage_3_su
120 Three_FA11 : full_adder port map(stage_4_sum(9), stage_4_carry(8), stage_6_carry(14), stage_3_su
121 Three_FA12 : full_adder port map(stage_4_sum(10), stage_4_carry(9), stage_6_carry(16), stage_3_s
122 Three_FA13 : full_adder port map(stage_4_sum(11), stage_4_carry(10), and_array(7,6), stage_3_sum
123 Three_FA14 : full_adder port map(ACC(14), stage_4_carry(11), and_array(7,7), stage_3_sum(13), st
124
125
126 -- Final sums and carries are used by Brent Kung Adder
127
128 final_sums <= ACC(15) & stage_3_sum & ACC(0) ;
129 final_carries <= stage_3_carry & and_array(1,0) & and_array(0,0);
130
131 final_instance : brent_kung_adder port map(final_sums, final_carries, '0', Sum, Cout);
132
133 end architecture;
```

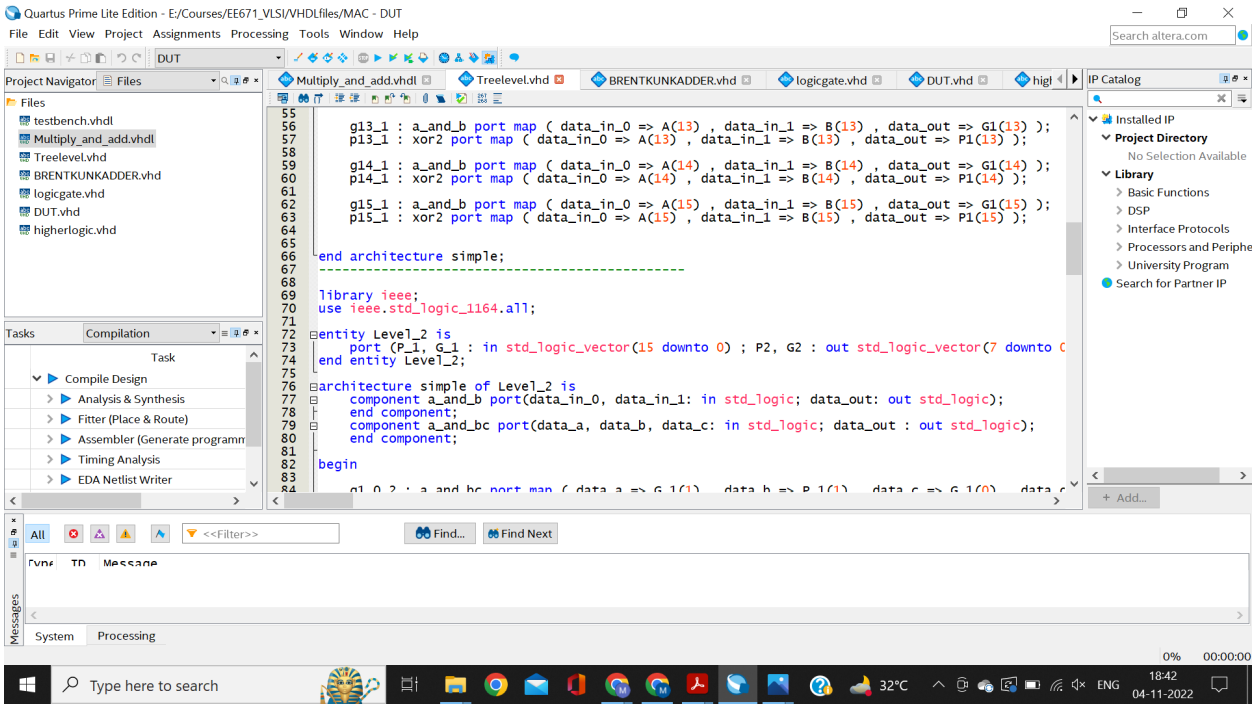
TREE LEVEL :

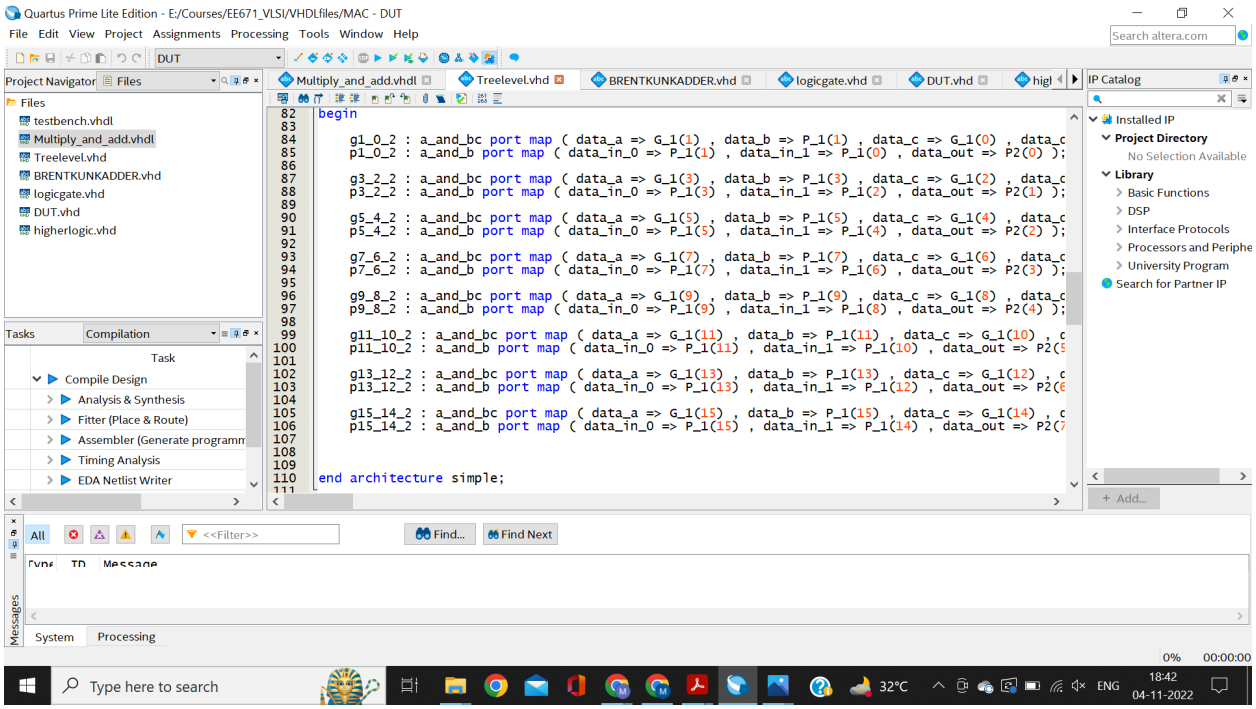


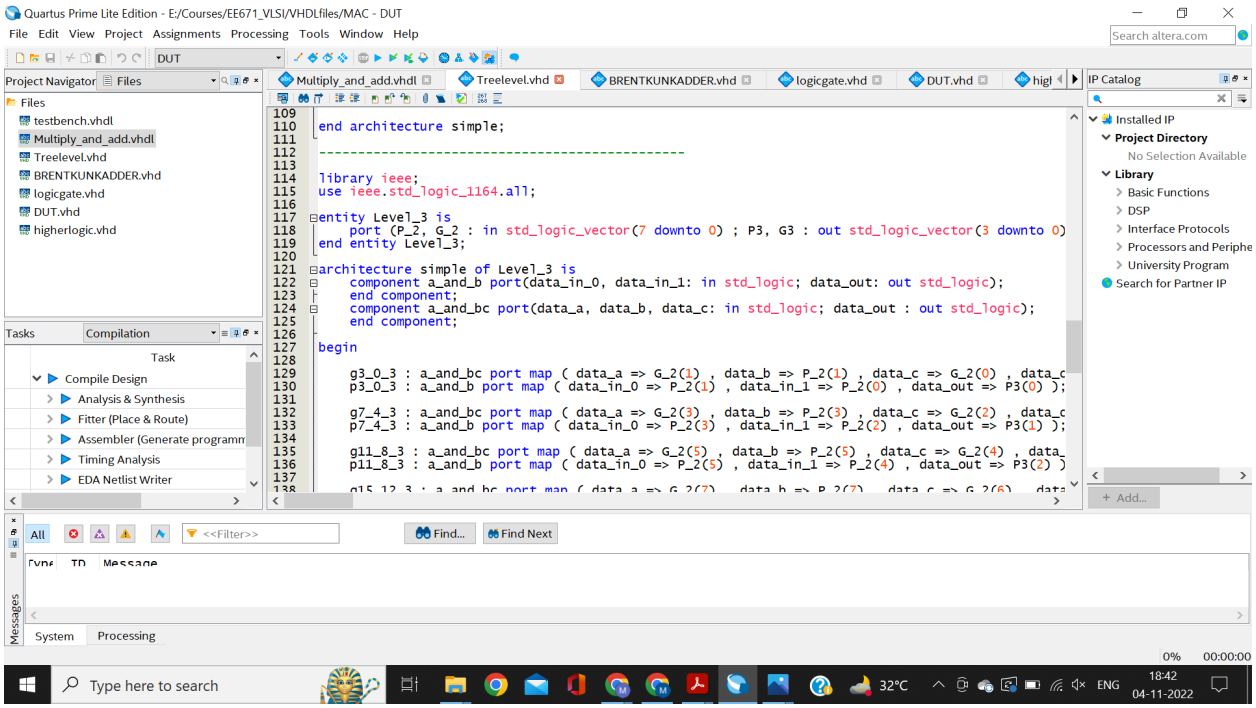

```
library ieee;
use ieee.std_logic_1164.all;

entity Level_1_1 is
    port (A, B : in std_logic_vector(15 downto 0) ; P1, G1 : out std_logic_vector(15 downto 0) );
end entity Level_1;

architecture simple of Level_1_1 is
    component a_and_b port(data_in_0, data_in_1: in std_logic; data_out: out std_logic);
    end component;
    component xor2 port(data_in_0, data_in_1: in std_logic; data_out: out std_logic);
    end component;


begin

    g0_1 : a_and_b port map ( data_in_0 => A(0) , data_in_1 => B(0) , data_out => G1(0) );
    p0_1 : xor2 port map ( data_in_0 => A(0) , data_in_1 => B(0) , data_out => P1(0) );

    g1_1 : a_and_b port map ( data_in_0 => A(1) , data_in_1 => B(1) , data_out => G1(1) );
    p1_1 : xor2 port map ( data_in_0 => A(1) , data_in_1 => B(1) , data_out => P1(1) );

    g2_1 : a_and_b port map ( data_in_0 => A(2) , data_in_1 => B(2) , data_out => G1(2) );
    p2_1 : xor2 port map ( data_in_0 => A(2) , data_in_1 => B(2) , data_out => P1(2) );

    g3_1 : a_and_b port map ( data_in_0 => A(3) , data_in_1 => B(3) , data_out => G1(3) );
    p3_1 : xor2 port map ( data_in_0 => A(3) , data_in_1 => B(3) , data_out => P1(3) );

    g4_1 : a_and_b port map ( data_in_0 => A(4) , data_in_1 => B(4) , data_out => G1(4) );
```

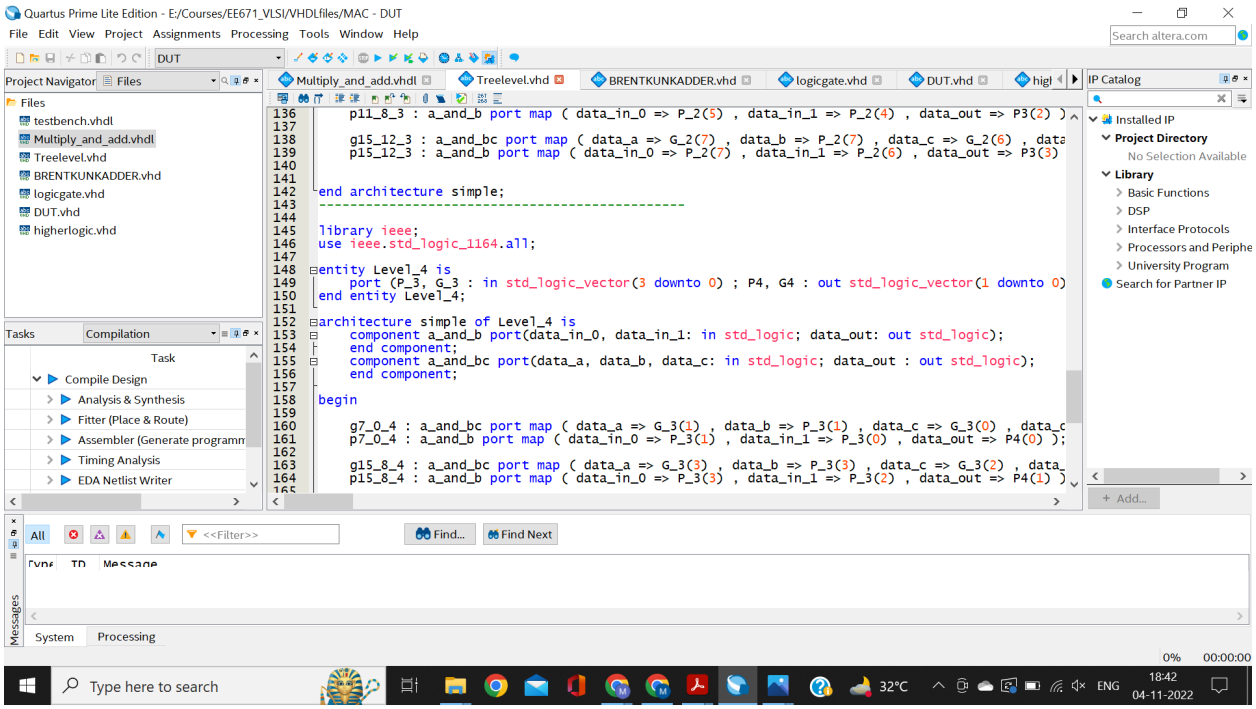


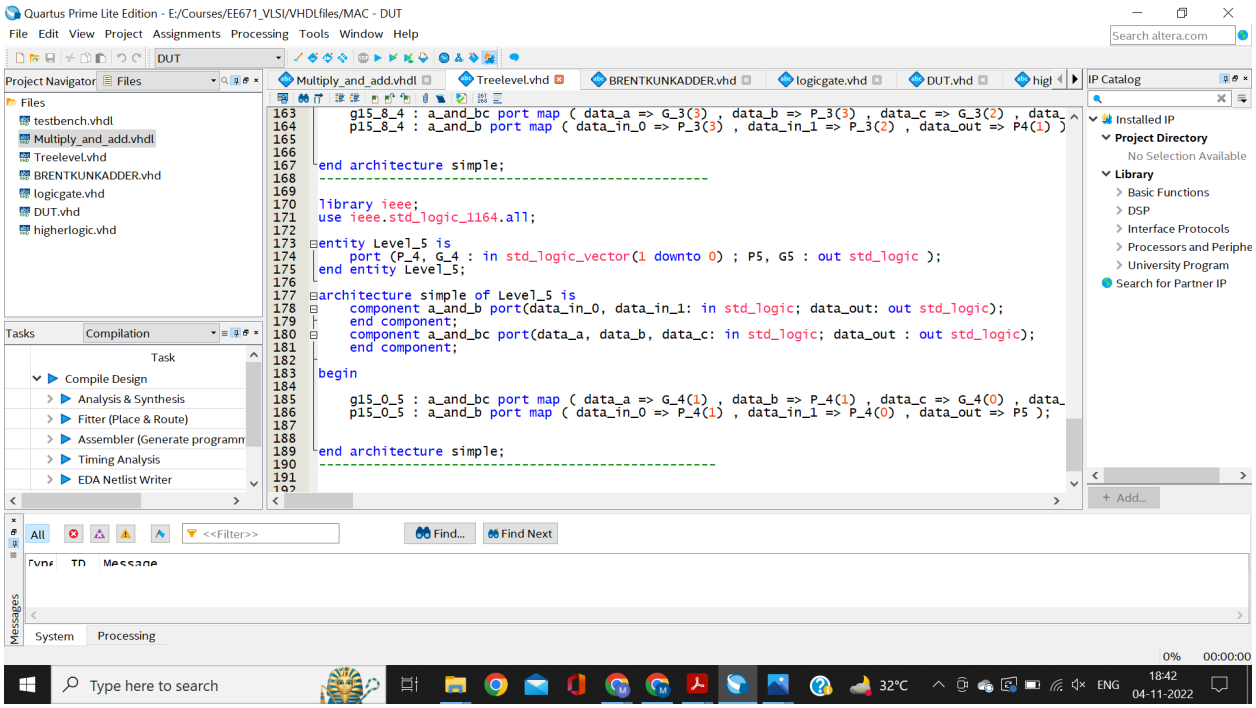

```
    g4_1 : a_and_b port map ( data_in_0 => A(4) , data_in_1 => B(4) , data_out => G1(4) );
    p4_1 : xor2 port map ( data_in_0 => A(4) , data_in_1 => B(4) , data_out => P1(4) );

    g5_1 : a_and_b port map ( data_in_0 => A(5) , data_in_1 => B(5) , data_out => G1(5) );
    p5_1 : xor2 port map ( data_in_0 => A(5) , data_in_1 => B(5) , data_out => P1(5) );

    g6_1 : a_and_b port map ( data_in_0 => A(6) , data_in_1 => B(6) , data_out => G1(6) );
    p6_1 : xor2 port map ( data_in_0 => A(6) , data_in_1 => B(6) , data_out => P1(6) );

    g7_1 : a_and_b port map ( data_in_0 => A(7) , data_in_1 => B(7) , data_out => G1(7) );
    p7_1 : xor2 port map ( data_in_0 => A(7) , data_in_1 => B(7) , data_out => P1(7) );

    g8_1 : a_and_b port map ( data_in_0 => A(8) , data_in_1 => B(8) , data_out => G1(8) );
    p8_1 : xor2 port map ( data_in_0 => A(8) , data_in_1 => B(8) , data_out => P1(8) );

    g9_1 : a_and_b port map ( data_in_0 => A(9) , data_in_1 => B(9) , data_out => G1(9) );
    p9_1 : xor2 port map ( data_in_0 => A(9) , data_in_1 => B(9) , data_out => P1(9) );

    g10_1 : a_and_b port map ( data_in_0 => A(10) , data_in_1 => B(10) , data_out => G1(10) );
    p10_1 : xor2 port map ( data_in_0 => A(10) , data_in_1 => B(10) , data_out => P1(10) );

    g11_1 : a_and_b port map ( data_in_0 => A(11) , data_in_1 => B(11) , data_out => G1(11) );
    p11_1 : xor2 port map ( data_in_0 => A(11) , data_in_1 => B(11) , data_out => P1(11) );

    g12_1 : a_and_b port map ( data_in_0 => A(12) , data_in_1 => B(12) , data_out => G1(12) );
    p12_1 : xor2 port map ( data_in_0 => A(12) , data_in_1 => B(12) , data_out => P1(12) );

    g13_1 : a_and_b port map ( data_in_0 => A(13) , data_in_1 => B(13) , data_out => G1(13) );
```


Quartus Prime Lite Edition - E:/Courses/EE671_VLSI/VHDLfiles/MAC - DUT
File Edit View Project Assignments Processing Tools Window Help
Search altera.com
Project Navigator Files
Files
testbench.vhdl
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
higherlogic.vhd
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
IP Catalog
Installed IP
Project Directory
No Selection Available
Library
Basic Functions
DSP
Interface Protocols
Processors and Peripher
University Program
Search for Partner IP
Tasks Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Assembler (Generate programm
Timing Analysis
EDA Netlist Writer
g13_1 : a_and_b port map (data_in_0 => A(13) , data_in_1 => B(13) , data_out => G1(13));
p13_1 : xor2 port map (data_in_0 => A(13) , data_in_1 => B(13) , data_out => P1(13));
g14_1 : a_and_b port map (data_in_0 => A(14) , data_in_1 => B(14) , data_out => G1(14));
p14_1 : xor2 port map (data_in_0 => A(14) , data_in_1 => B(14) , data_out => P1(14));
g15_1 : a_and_b port map (data_in_0 => A(15) , data_in_1 => B(15) , data_out => G1(15));
p15_1 : xor2 port map (data_in_0 => A(15) , data_in_1 => B(15) , data_out => P1(15));
end architecture simple;
library ieee;
use ieee.std_logic_1164.all;
entity Level_2 is
port (P_1, G_1 : in std_logic_vector(15 downto 0) ; P2, G2 : out std_logic_vector(7 downto 0
end entity Level_2;
architecture simple of Level_2 is
component a_and_b port(data_in_0, data_in_1: in std_logic; data_out: out std_logic);
end component;
component a_and_bc port(data_a, data_b, data_c: in std_logic; data_out : out std_logic);
end component;
begin
All
<<Filter>>
Find...
Find Next
Type
ID
Message
System
Processing
0%
00:00:00
Type here to search
32°C
ENG
18:42
04-11-2022


Quartus Prime Lite Edition - E:/Courses/EE671_VLSI/VHDLfiles/MAC - DUT
File Edit View Project Assignments Processing Tools Window Help
Search altera.com
Project Navigator Files
Files
testbench.vhdl
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
higherlogic.vhd
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
IP Catalog
Installed IP
Project Directory
No Selection Available
Library
Basic Functions
DSP
Interface Protocols
Processors and Peripher
University Program
Search for Partner IP
Tasks Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Assembler (Generate programm
Timing Analysis
EDA Netlist Writer
begin
g1_0_2 : a_and_bc port map (data_a => G_1(1) , data_b => P_1(1) , data_c => G_1(0) , data_c
p1_0_2 : a_and_b port map (data_in_0 => P_1(1) , data_in_1 => P_1(0) , data_out => P2(0));
g3_2_2 : a_and_bc port map (data_a => G_1(3) , data_b => P_1(3) , data_c => G_1(2) , data_c
p3_2_2 : a_and_b port map (data_in_0 => P_1(3) , data_in_1 => P_1(2) , data_out => P2(1));
g5_4_2 : a_and_bc port map (data_a => G_1(5) , data_b => P_1(5) , data_c => G_1(4) , data_c
p5_4_2 : a_and_b port map (data_in_0 => P_1(5) , data_in_1 => P_1(4) , data_out => P2(2));
g7_6_2 : a_and_bc port map (data_a => G_1(7) , data_b => P_1(7) , data_c => G_1(6) , data_c
p7_6_2 : a_and_b port map (data_in_0 => P_1(7) , data_in_1 => P_1(6) , data_out => P2(3));
g9_8_2 : a_and_bc port map (data_a => G_1(9) , data_b => P_1(9) , data_c => G_1(8) , data_c
p9_8_2 : a_and_b port map (data_in_0 => P_1(9) , data_in_1 => P_1(8) , data_out => P2(4));
g11_10_2 : a_and_bc port map (data_a => G_1(11) , data_b => P_1(11) , data_c => G_1(10) , c
p11_10_2 : a_and_b port map (data_in_0 => P_1(11) , data_in_1 => P_1(10) , data_out => P2(5
g13_12_2 : a_and_bc port map (data_a => G_1(13) , data_b => P_1(13) , data_c => G_1(12) , c
p13_12_2 : a_and_b port map (data_in_0 => P_1(13) , data_in_1 => P_1(12) , data_out => P2(6
g15_14_2 : a_and_bc port map (data_a => G_1(15) , data_b => P_1(15) , data_c => G_1(14) , c
p15_14_2 : a_and_b port map (data_in_0 => P_1(15) , data_in_1 => P_1(14) , data_out => P2(7
end architecture simple;
All
<<Filter>>
Find...
Find Next
Type
ID
Message
System
Processing
0%
00:00:00
Type here to search
32°C
ENG
18:42
04-11-2022

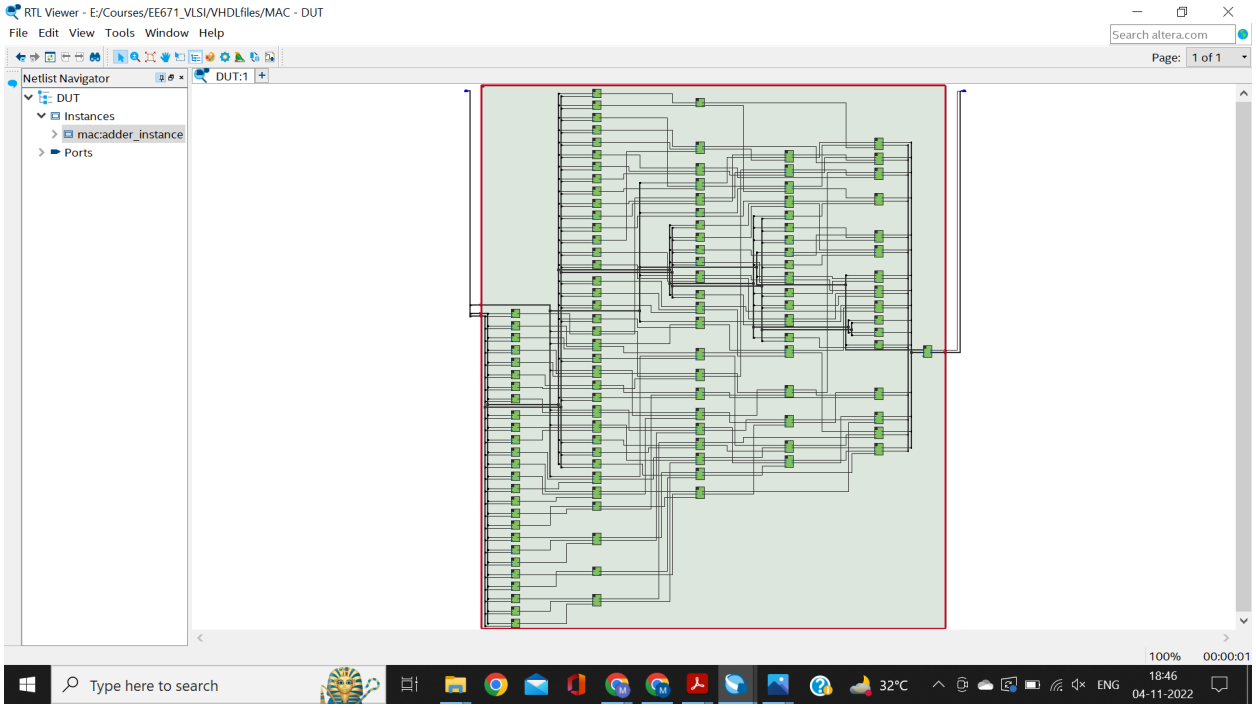

Quartus Prime Lite Edition - E:/Courses/EE671_VLSI/VHDLfiles/MAC - DUT
File Edit View Project Assignments Processing Tools Window Help
Search altera.com
Project Navigator Files
Files
testbench.vhdl
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
higherlogic.vhd
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
IP Catalog
Installed IP
Project Directory
No Selection Available
Library
Basic Functions
DSP
Interface Protocols
Processors and Periphe
University Program
Search for Partner IP
Tasks
Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Assembler (Generate programm
Timing Analysis
EDA Netlist Writer
end architecture simple;
library ieee;
use ieee.std_logic_1164.all;
entity Level_3 is
port (P_2, G_2 : in std_logic_vector(7 downto 0) ; P3, G3 : out std_logic_vector(3 downto 0)
end entity Level_3;
architecture simple of Level_3 is
component a_and_b port(data_in_0, data_in_1: in std_logic; data_out: out std_logic);
end component;
component a_and_bc port(data_a, data_b, data_c: in std_logic; data_out : out std_logic);
end component;
begin
g3_0_3 : a_and_bc port map (data_a => G_2(1) , data_b => P_2(1) , data_c => G_2(0) , data_c
p3_0_3 : a_and_b port map (data_in_0 => P_2(1) , data_in_1 => P_2(0) , data_out => P3(0));
g7_4_3 : a_and_bc port map (data_a => G_2(3) , data_b => P_2(3) , data_c => G_2(2) , data_c
p7_4_3 : a_and_b port map (data_in_0 => P_2(3) , data_in_1 => P_2(2) , data_out => P3(1));
g11_8_3 : a_and_bc port map (data_a => G_2(5) , data_b => P_2(5) , data_c => G_2(4) , data_
p11_8_3 : a_and_b port map (data_in_0 => P_2(5) , data_in_1 => P_2(4) , data_out => P3(2))
All
<<Filter>>
Find...
Find Next
Type
ID
Message
System
Processing
0%
00:00:00
Type here to search
32°C
ENG
18:42
04-11-2022


Quartus Prime Lite Edition - E:/Courses/EE671_VLSI/VHDLfiles/MAC - DUT
File Edit View Project Assignments Processing Tools Window Help
Search altera.com
Project Navigator Files
Files
testbench.vhdl
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
higherlogic.vhd
Multiply_and_add.vhdl
Treelevel.vhd
BRENTKUNKADDER.vhd
logicgate.vhd
DUT.vhd
IP Catalog
Installed IP
Project Directory
No Selection Available
Library
Basic Functions
DSP
Interface Protocols
Processors and Periphe
University Program
Search for Partner IP
Tasks
Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Assembler (Generate programm
Timing Analysis
EDA Netlist Writer
p11_8_3 : a_and_b port map (data_in_0 => P_2(5) , data_in_1 => P_2(4) , data_out => P3(2))
g15_12_3 : a_and_bc port map (data_a => G_2(7) , data_b => P_2(7) , data_c => G_2(6) , data
p15_12_3 : a_and_b port map (data_in_0 => P_2(7) , data_in_1 => P_2(6) , data_out => P3(3)
end architecture simple;
library ieee;
use ieee.std_logic_1164.all;
entity Level_4 is
port (P_3, G_3 : in std_logic_vector(3 downto 0) ; P4, G4 : out std_logic_vector(1 downto 0)
end entity Level_4;
architecture simple of Level_4 is
component a_and_b port(data_in_0, data_in_1: in std_logic; data_out: out std_logic);
end component;
component a_and_bc port(data_a, data_b, data_c: in std_logic; data_out : out std_logic);
end component;
begin
g7_0_4 : a_and_bc port map (data_a => G_3(1) , data_b => P_3(1) , data_c => G_3(0) , data_c
p7_0_4 : a_and_b port map (data_in_0 => P_3(1) , data_in_1 => P_3(0) , data_out => P4(0));
g15_8_4 : a_and_bc port map (data_a => G_3(3) , data_b => P_3(3) , data_c => G_3(2) , data_
p15_8_4 : a_and_b port map (data_in_0 => P_3(3) , data_in_1 => P_3(2) , data_out => P4(1))
All
<<Filter>>
Find...
Find Next
Type
ID
Message
System
Processing
0%
00:00:00
Type here to search
32°C
ENG
18:42
04-11-2022

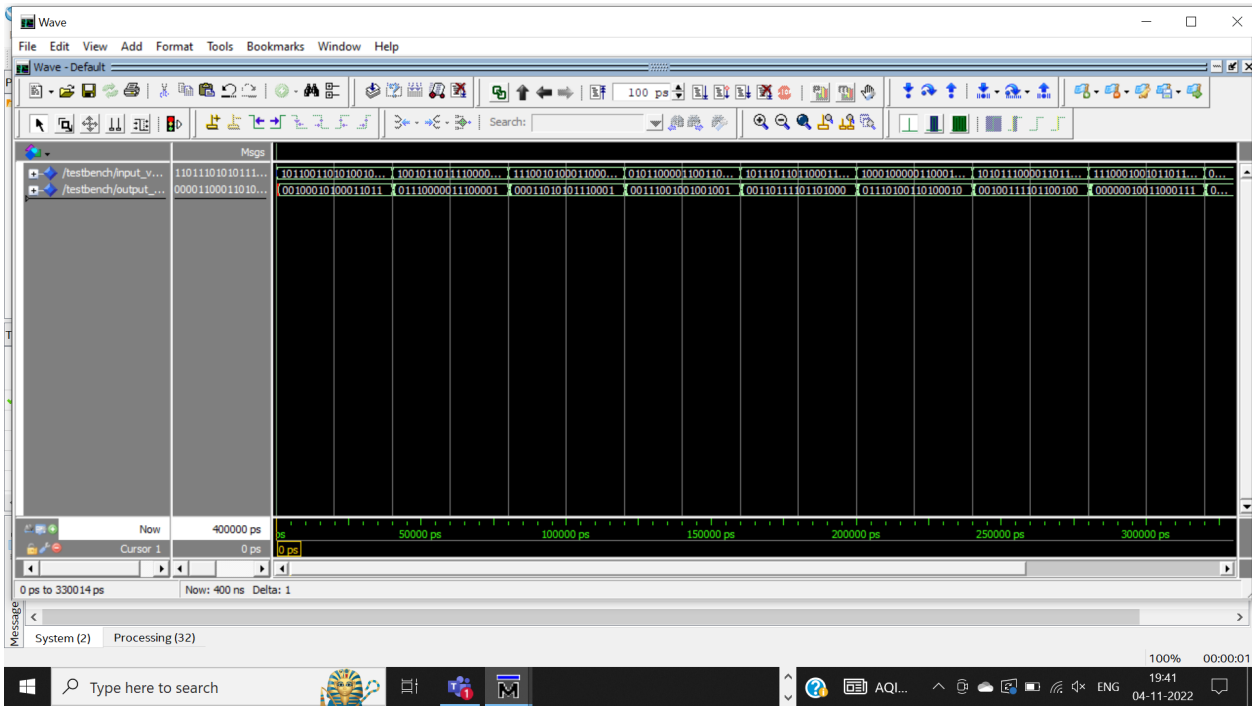



```
    g15_8_4 : a_and_bc port map ( data_a => G_3(3) , data_b => P_3(3) , data_c => G_3(2) , data_
    p15_8_4 : a_and_b port map ( data_in_0 => P_3(3) , data_in_1 => P_3(2) , data_out => P4(1) )


end architecture simple;
-------------------------------------------------

library ieee;
use ieee.std_logic_1164.all;

entity Level_5 is
    port (P_4, G_4 : in std_logic_vector(1 downto 0) ; P5, G5 : out std_logic );
end entity Level_5;

architecture simple of Level_5 is
    component a_and_b port(data_in_0, data_in_1: in std_logic; data_out: out std_logic);
    end component;
    component a_and_bc port(data_a, data_b, data_c: in std_logic; data_out : out std_logic);
    end component;

begin

    g15_0_5 : a_and_bc port map ( data_a => G_4(1) , data_b => P_4(1) , data_c => G_4(0) , data_
    p15_0_5 : a_and_b port map ( data_in_0 => P_4(1) , data_in_1 => P_4(0) , data_out => P5 );


end architecture simple;
-------------------------------------------------
```

RTL VIEW OF THE CIRCUIT :



RTL SIMULATION RESULT :



GATE LEVEL SIMULATION RESULT :