EXPT NO. : 2

To simulate the operation of a half-adder using basic gates

Theory:

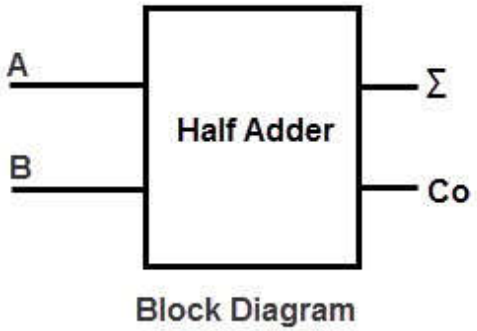
A logic circuit block used for adding two one bit numbers or simply two bits is called as a half adder circuit. This circuit has two inputs which accept the two bits and two outputs, with one producing sum output and other produce carry output.

As we know that binary addition is commonly performed by Ex-OR gate, but for the first three rules, it performs the binary addition and when the two inputs are logic 1, it does not develop any carry.

To accomplish the binary addition with Ex-OR gate, there is need of additional circuitry to perform the carry operation. Hence, a half adder is formed by connecting AND gate to the input terminals of the Ex-OR gate so as to produce the carry.

$$\text{Sum } (\sum) = A \oplus B, \text{ Carry Out } (Co) = A \cdot B$$

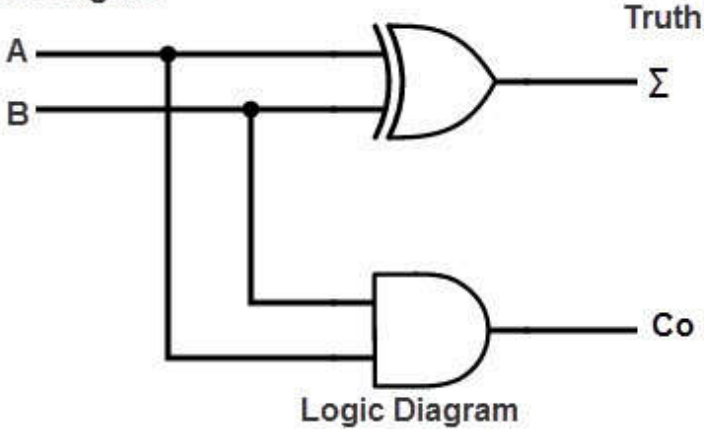
The block diagram, circuit diagram and the truth table for half adder is given below:



Block Diagram

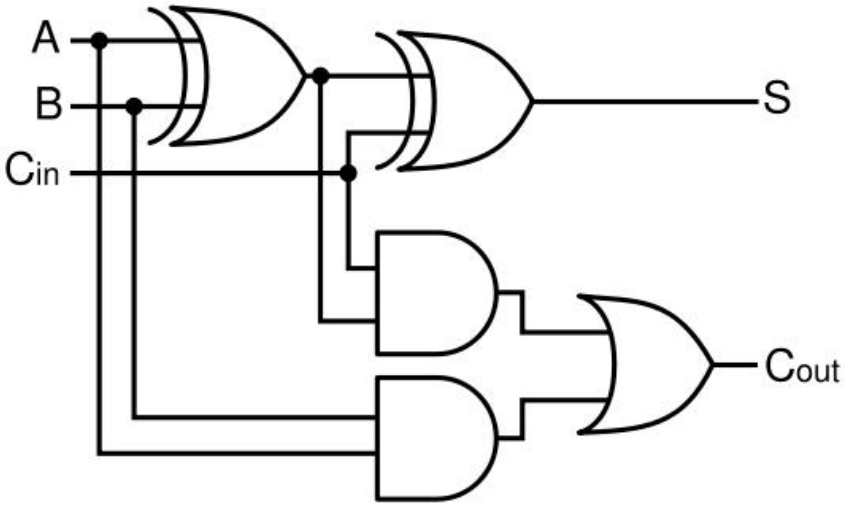
A	B	Sum (∑)	Carry Out (Co)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Truth Table



Logic Diagram

The I/O ports needed to be declared for the formation of half adder is given below:



Port Name	input/output	Bus
A	In	No
B	In	No
Sum	Out	No
Carry	Out	No

NB: Use temporary variable where ever necessary.

To simulate the operation of a full adder using logic gates

Full adder is a digital circuit used to calculate the sum of three binary bits which is the main difference between full adder and half adder. Full adders are complex and difficult to implement when compared to half adders. The additional third bit is carry bit from the previous stage and is called Carry – in generally represented by Cin. It calculates the sum of three bits along with the carry. The output carry is called Carry – out and is represented by Cout.

The circuit diagram of a full adder and its truth table is given below:

Input			Output	
A	B	Cin	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

The i/o ports needed to be declared for the formation of full adder is given below:

Port Name	Input/output	Bus
A	In	No
B	In	No
Cin	In	No
Sum	Out	No
Cout	Out	No

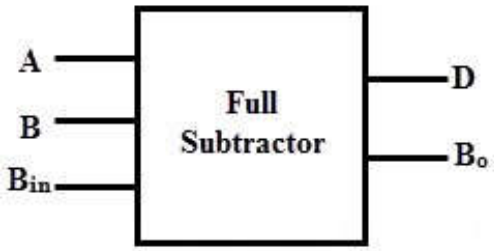
NB: Use temporary variable where ever necessary.

To simulate the operation of a full adder using logic gates

Theory:

A combinational logic circuit performs a subtraction between the two binary bits by considering borrow of the lower significant stage is called as the full subtractor. In this, subtraction of the two digits is performed by taking into consideration whether a 1 has already been borrowed by the previous adjacent lower minuend bit or not.

It has three input terminals in which two terminals corresponds to the two bits to be subtracted (minuend A and subtrahend B), and a borrow bit B_i corresponds to the borrow operation. There are two outputs, one corresponds to the difference D output and other borrow output B_o as shown in figure along with truth table.

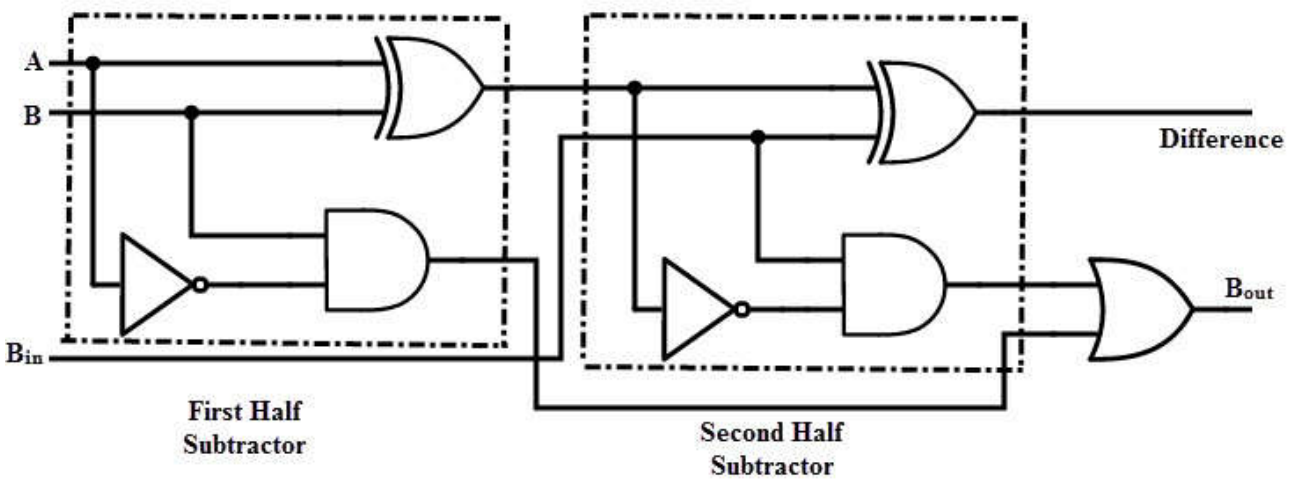


Block Diagram of
Full Subtractor

A	B	B_{in}	D	B_o
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Truth Table

The circuit diagram of full subtractor is given below:



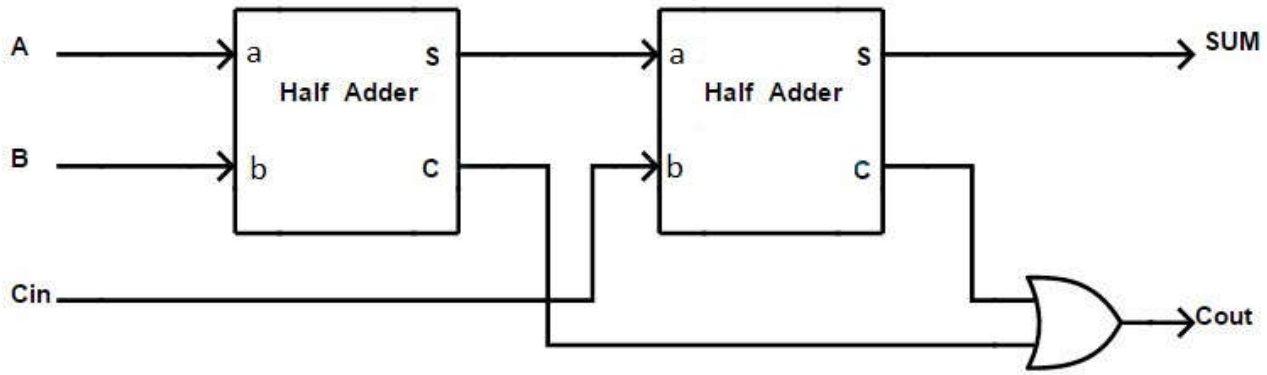
The i/o ports needed to be declared for the formation of full subtractor is given below:

Port Name	Input/output	Bus
A	In	No
B	In	No
Bin	In	No
Diff	Out	No
Bout	Out	No

NB: Use temporary variable where ever necessary.

To simulate the operation of a full adder using half adder as component

The block diagram of a full adder constructed using half adder is given below:



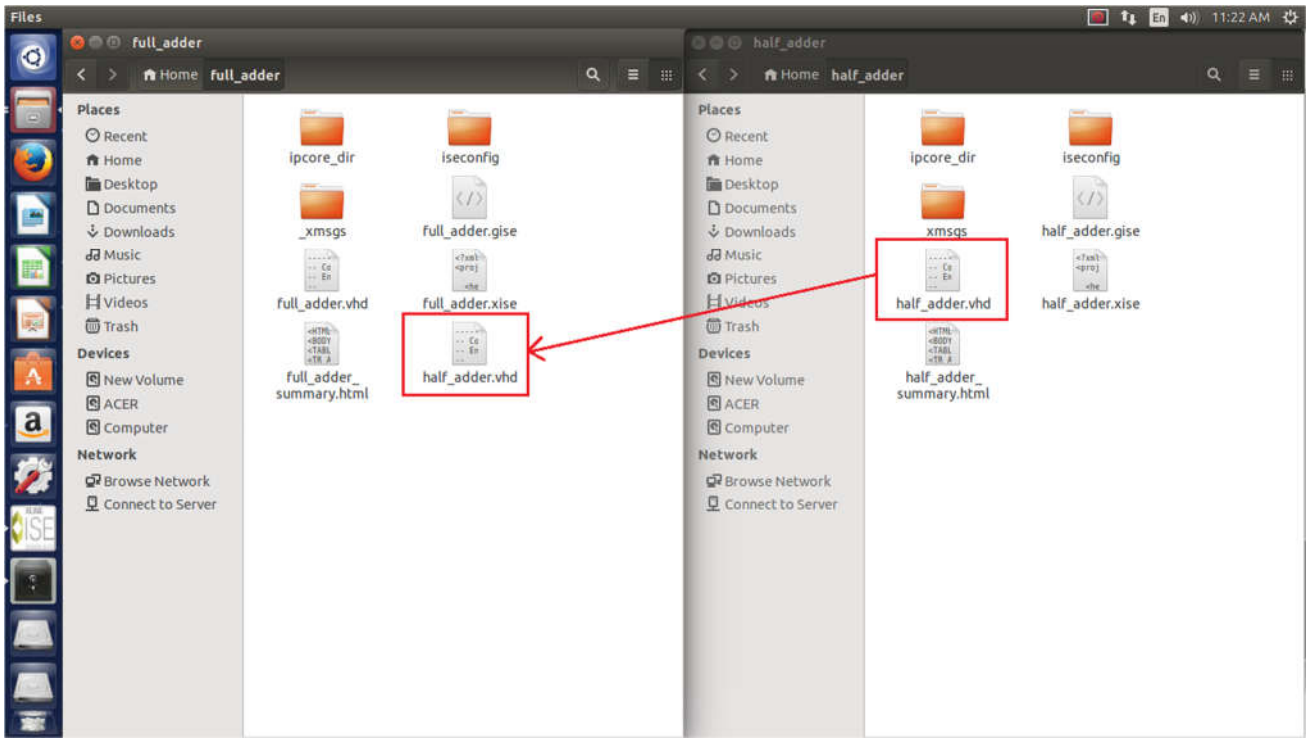
The i/o ports needed to be declared for the formation of full adder is given below:

Port Name	Input/output	Bus
A	In	No
B	In	No
Cin	In	No
Sum	Out	No
Cout	Out	No

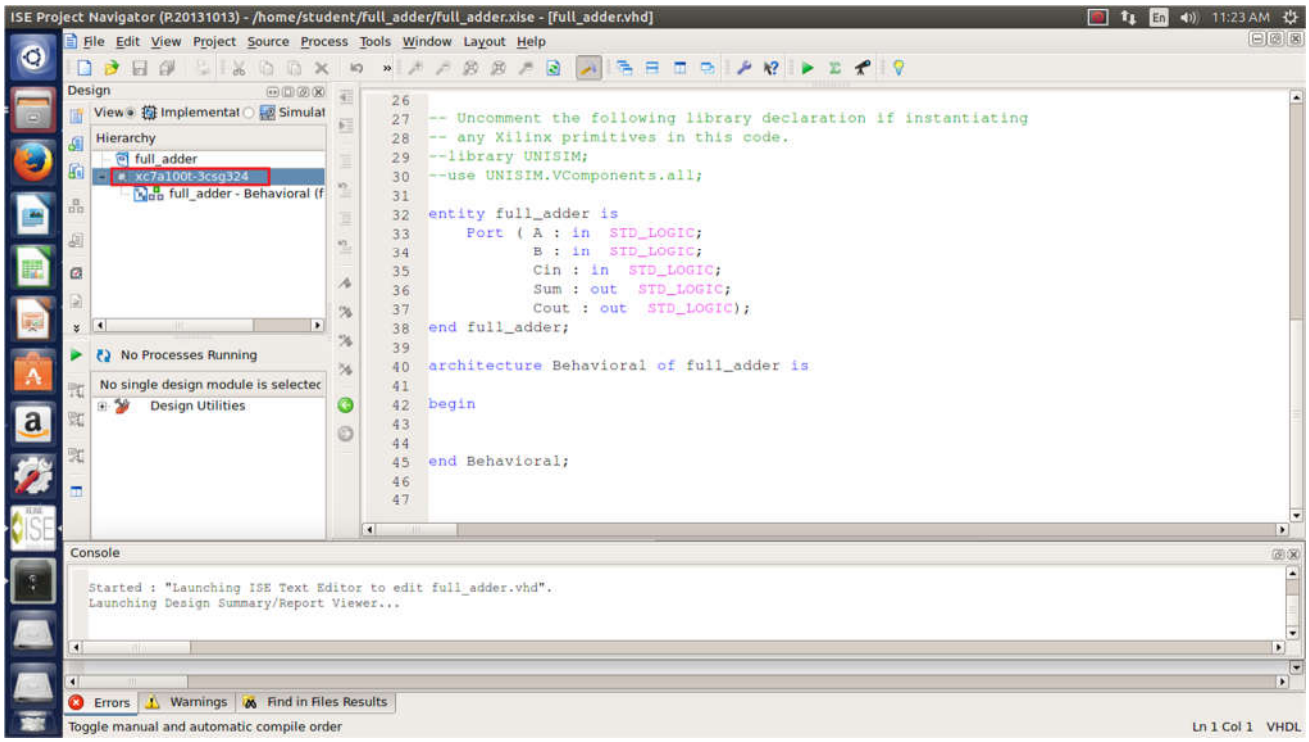
NB: Use temporary variable where ever necessary.

How to add component and map your port from full adder to half adder

To add the half adder component, you can use the half adder **.vhd** file from your previous assignment and copy it into the full adder folder.



And then right click on the **parent of your full_adder-Behavioral** and select **add source.** The place to click is marked in the image below.

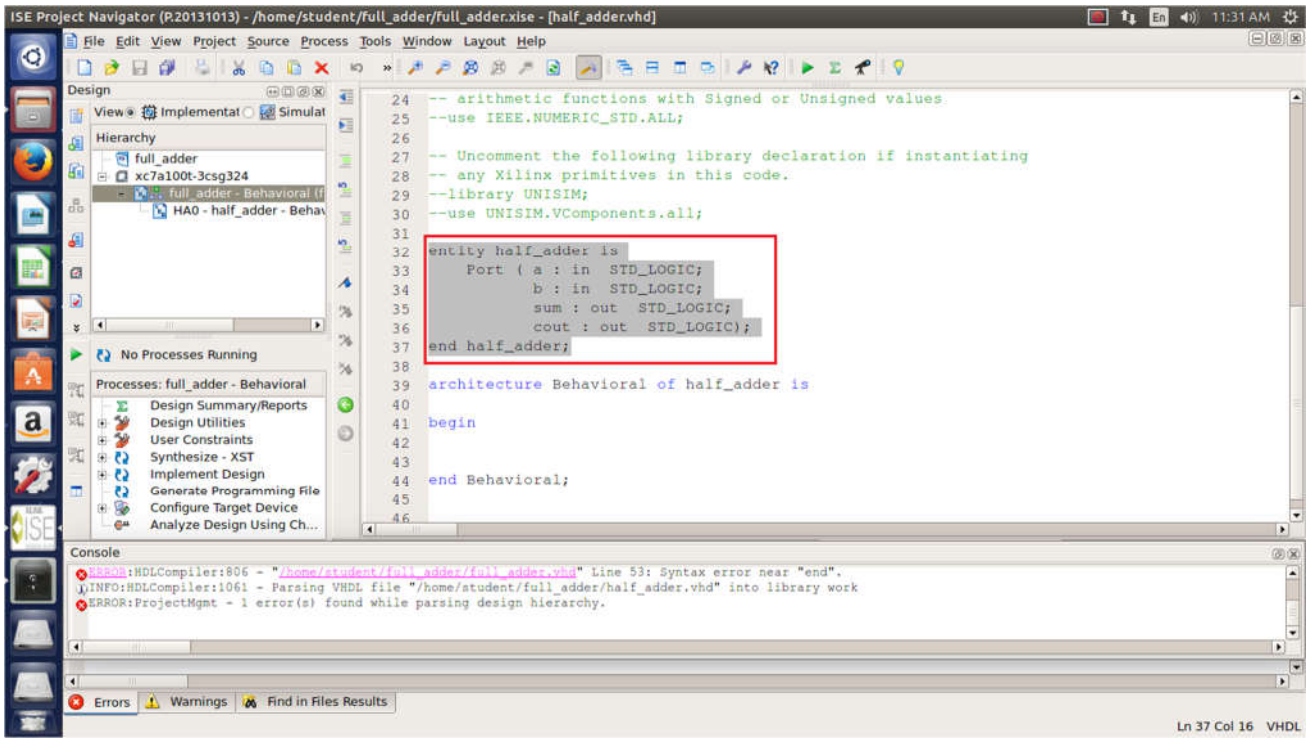


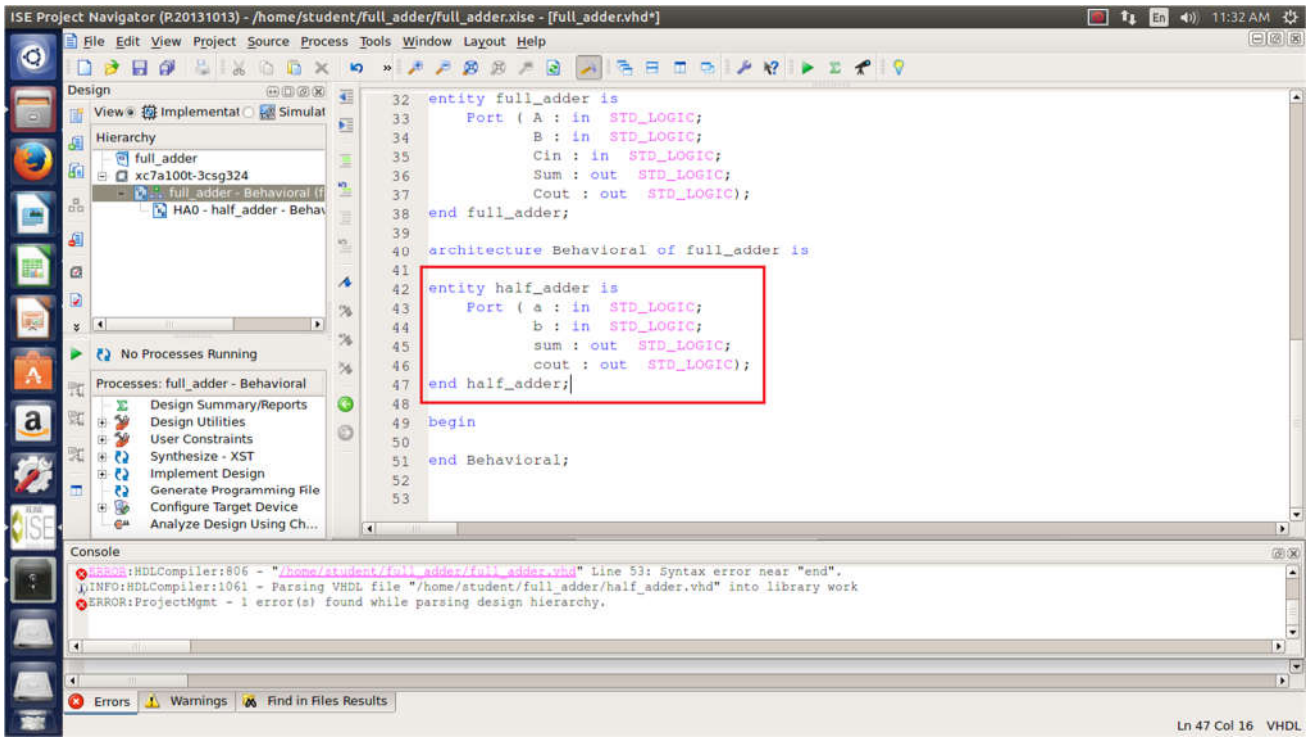
And then browse and select your half adder .vhd that you copied in your full adder folder and click **open**. In the next window click **ok.**

You can also create a new vhd file for half adder if you don't have one. For that right click on the full_adder-Behavioral and click on new source and then proceed normally to create a half adder entity.

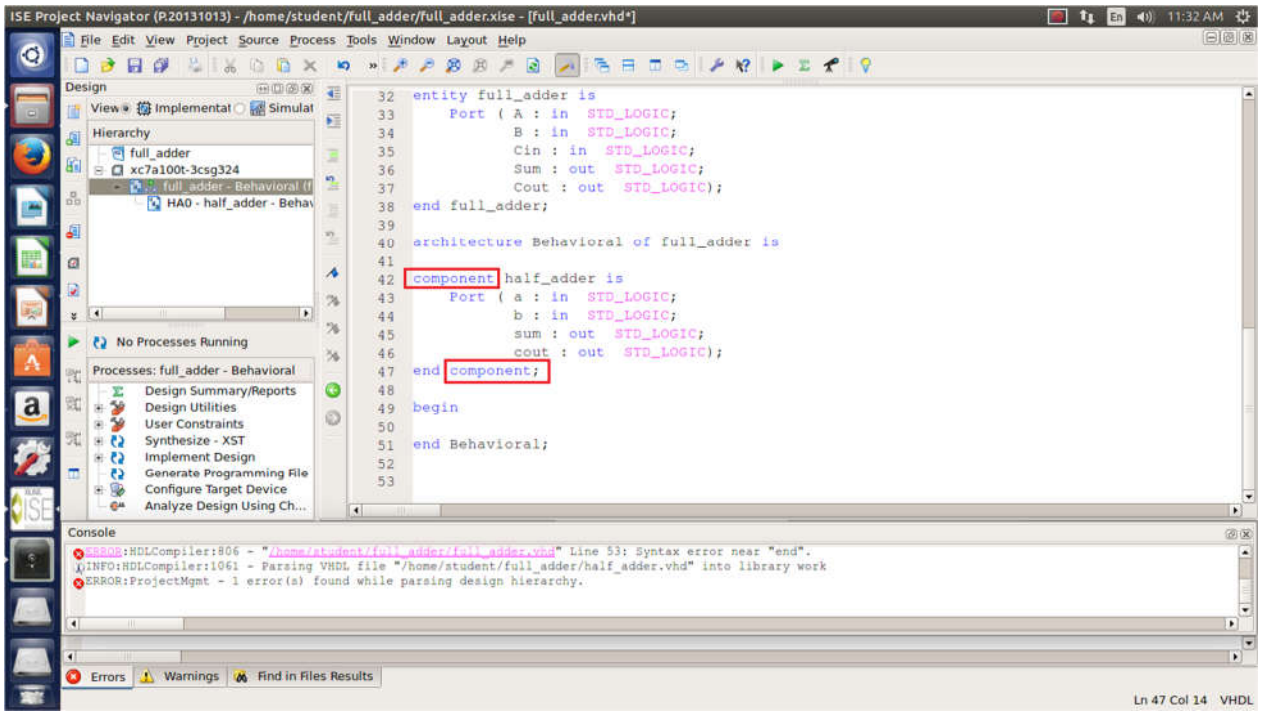
NB – Make sure that the port name for full adder and half adder are different.

After adding the half adder vhd file copy the half adder entity from the half adder's vhd file and paste it in the behavioral block of the full adder vhd, as shown in the below pictures.


ISE Project Navigator (P.20131013) - /home/student/full_adder/full_adder.xise - [half_adder.vhd]
File Edit View Project Source Process Tools Window Layout Help
Design
Hierarchy
full_adder
xc7a100t-3csg324
full_adder - Behavioral (f
HA0 - half_adder - Behav
No Processes Running
Processes: full_adder - Behavioral
Design Summary/Reports
Design Utilities
User Constraints
Synthesize - XST
Implement Design
Generate Programming File
Configure Target Device
Analyze Design Using Ch...
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
entity half_adder is
Port (a : in STD_LOGIC;
b : in STD_LOGIC;
sum : out STD_LOGIC;
cout : out STD_LOGIC);
end half_adder;
architecture Behavioral of half_adder is
begin
end Behavioral;
Console
ERROR:HDLCompiler:806 - "/home/student/full_adder/full_adder.vhd" Line 53: Syntax error near "end".
INFO:HDLCompiler:1061 - Parsing VHDL file "/home/student/full_adder/half_adder.vhd" into library work
ERROR:ProjectMgmt - 1 error(s) found while parsing design hierarchy.
Errors
Warnings
Find in Files Results
Ln 37 Col 16 VHDL


ISE Project Navigator (P.20131013) - /home/student/full_adder/full_adder.xise - [full_adder.vhd*]
File Edit View Project Source Process Tools Window Layout Help
Design
Hierarchy
full_adder
xc7a100t-3csg324
full_adder - Behavioral (f
HA0 - half_adder - Behav
No Processes Running
Processes: full_adder - Behavioral
Design Summary/Reports
Design Utilities
User Constraints
Synthesize - XST
Implement Design
Generate Programming File
Configure Target Device
Analyze Design Using Ch...
entity full_adder is
Port (A : in STD_LOGIC;
B : in STD_LOGIC;
Cin : in STD_LOGIC;
Sum : out STD_LOGIC;
Cout : out STD_LOGIC);
end full_adder;
architecture Behavioral of full_adder is
entity half_adder is
Port (a : in STD_LOGIC;
b : in STD_LOGIC;
sum : out STD_LOGIC;
cout : out STD_LOGIC);
end half_adder;
begin
end Behavioral;
Console
ERROR:HDLCompiler:806 - "/home/student/full_adder/full_adder.vhd" Line 53: Syntax error near "end".
INFO:HDLCompiler:1061 - Parsing VHDL file "/home/student/full_adder/half_adder.vhd" into library work
ERROR:ProjectMgmt - 1 error(s) found while parsing design hierarchy.
Errors
Warnings
Find in Files Results
Ln 47 Col 16 VHDL

After copying the entity block make the changes marked in the image below.



Now we need to map the ports from full adder to half adder as shown in the picture.