

Lab 1

Shahed Ahmed

2/16/2023

CDA 3203 Computer Logic Design


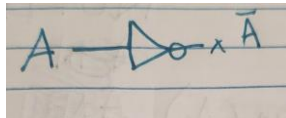
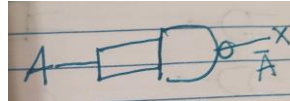
Spring 2023

Dr. Maria Petrie


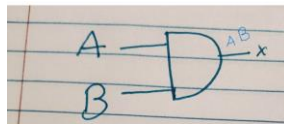
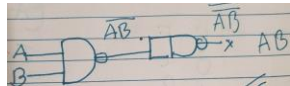
Florida Atlantic University

Part 1: Design 9 circuits by completing below: Draw the symbol for the gate, its Truth Table, its Simplest Sum of Products Expression, draw its NOT-AND-OR Equivalent Circuit, its all-NAND Equivalent Circuit.

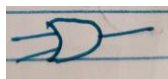
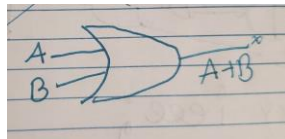
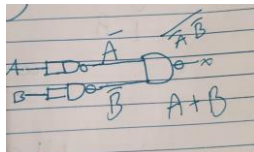
1.1 NOT gate.

Draw NOT gate	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit						
	<table border="1"><tr><th>A</th><th>NOT(A)</th></tr><tr><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td></tr></table> <p>Y1=A'</p>	A	NOT(A)	1	1	0	0		
A	NOT(A)								
1	1								
0	0								


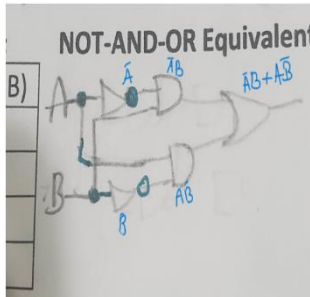
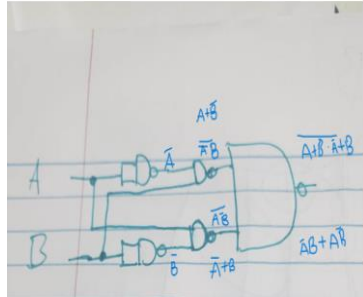
1.2- AND gate

Draw AND gate	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit															
	<table border="1"><thead><tr><th>A</th><th>B</th><th>AND(A,B)</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></tbody></table> <p>Y2= AB</p>	A	B	AND(A,B)	0	0	0	0	1	0	1	0	0	1	1	1		
A	B	AND(A,B)																
0	0	0																
0	1	0																
1	0	0																
1	1	1																


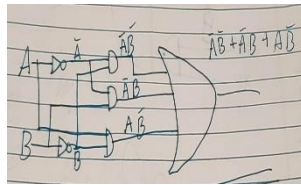
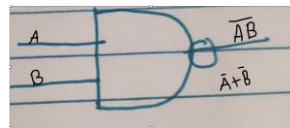
1.3- OR gate

Draw OR gate	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit															
	<table border="1"><thead><tr><th>A</th><th>B</th><th>OR(A,B)</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></tbody></table> <p>Y3=A+B</p>	A	B	OR(A,B)	0	0	0	0	1	1	1	0	1	1	1	1		
A	B	OR(A,B)																
0	0	0																
0	1	1																
1	0	1																
1	1	1																


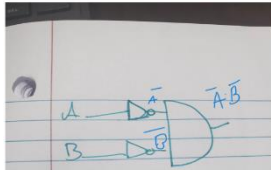
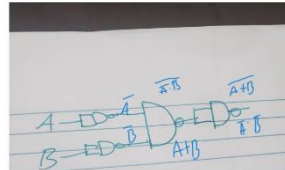
1.4- XOR gate

Draw XOR gate	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit															
	<table border="1"><thead><tr><th>A</th><th>B</th><th>XOR(A,B)</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></tbody></table> <p>$Y_4 = A'B + AB'$</p>	A	B	XOR(A,B)	0	0	0	0	1	1	1	0	1	1	1	0		
A	B	XOR(A,B)																
0	0	0																
0	1	1																
1	0	1																
1	1	0																

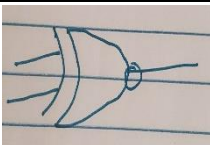
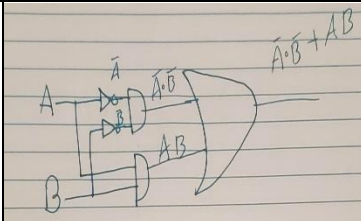
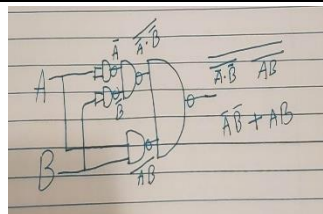
1.5- NAND gate

Draw NAND gate	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit															
	<table><tr><th>A</th><th>B</th><th>NAND(A,B)</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> <p>$Y_5=A'B'+A'B+AB'$</p>	A	B	NAND(A,B)	0	0	1	0	1	1	1	0	1	1	1	0		
A	B	NAND(A,B)																
0	0	1																
0	1	1																
1	0	1																
1	1	0																

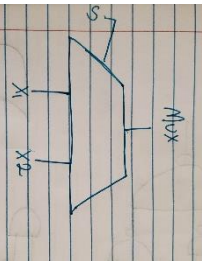
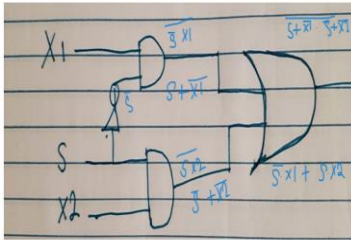
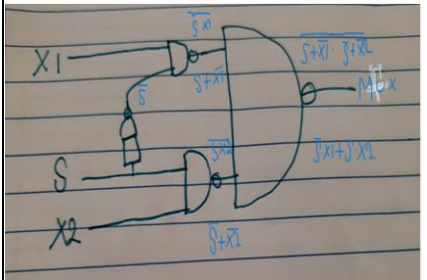
1.6- NOR gate

Draw NOR gate	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit															
	<table border="1"><thead><tr><th>A</th><th>B</th><th>NOR(A,B)</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></tbody></table> <p>$Y_6=A'B'$</p>	A	B	NOR(A,B)	0	0	1	0	1	0	1	0	0	1	1	0		
A	B	NOR(A,B)																
0	0	1																
0	1	0																
1	0	0																
1	1	0																

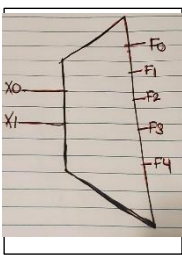
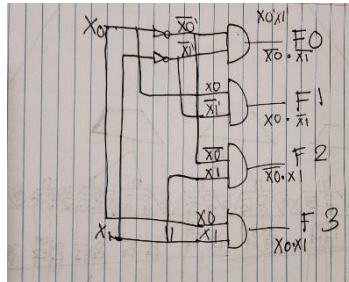
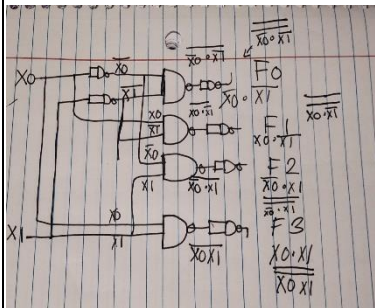
1.7- XNOR gate

Draw XNOR gate	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit															
	<table border="1"><thead><tr><th>A</th><th>B</th><th>XNOR(A,B)</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></tbody></table> <p>Y7=A'B'+AB</p>	A	B	XNOR(A,B)	0	0	1	0	1	0	1	0	0	1	1	1		
A	B	XNOR(A,B)																
0	0	1																
0	1	0																
1	0	0																
1	1	1																

1.8- 2 to 1 Encoder or Multiplexer (Mux)

Draw 2to1 Mux	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit																																				
	<table border="1"> <thead> <tr> <th>s</th><th>X₁</th><th>X₂</th><th>Mux</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr> <td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr> <td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr> <td>0</td><td>1</td><td>1</td><td>1</td></tr> <tr> <td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr> <td>1</td><td>0</td><td>1</td><td>1</td></tr> <tr> <td>1</td><td>1</td><td>0</td><td>0</td></tr> <tr> <td>1</td><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>$Y8 = S'X1 + SX2$</p>	s	X ₁	X ₂	Mux	0	0	0	0	0	0	1	0	0	1	0	1	0	1	1	1	1	0	0	0	1	0	1	1	1	1	0	0	1	1	1	1		
s	X ₁	X ₂	Mux																																				
0	0	0	0																																				
0	0	1	0																																				
0	1	0	1																																				
0	1	1	1																																				
1	0	0	0																																				
1	0	1	1																																				
1	1	0	0																																				
1	1	1	1																																				

1.9- 2 to 4 decoder or Demultiplexer (DMux)

Draw 2to4 DMux	Truth Table and Canonical Sum of Products Equation	NOT-AND-OR Equivalent Circuit	all-NAND Equivalent Circuit																														
	<table><tr><th>X_1</th><th>X_0</th><th>F_3</th><th>F_2</th><th>F_1</th><th>F_0</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr></table> <p>Y9: $F_3 = x_0 x_1$ $F_2 = x_0' x_1$ $F_1 = x_0 x_1'$ $F_0 = x_0' x_1'$</p>	X_1	X_0	F_3	F_2	F_1	F_0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0		
X_1	X_0	F_3	F_2	F_1	F_0																												
0	0	0	0	0	1																												
0	1	0	0	1	0																												
1	0	0	1	0	0																												
1	1	1	0	0	0																												

Part 2. Create a new project in Altera Quartus using VHDL instead of a schematic simulate the Mux2to1 using all-NAND gates.

Project Wizard

New Project Wizard: Summary [page 5 of 5]

When you click Finish, the project will be created with the following settings:

Project directory:
Z:/CDA3203-Logic/Lab1_ShahedAhmed/

Project name: Mux2to1_ShahedAhmed

Top-level design entity: Mux2to1_ShahedAhmed

Number of files added: 0

Number of user libraries added: 0

Device assignments:

Family name: MAX3000A

Device: EPM3064ALC44-10

EDA tools:

Design entry/synthesis: <None>

Simulation: <None>

Timing analysis: <None>

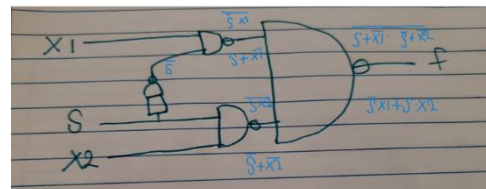
Operating conditions:

Core voltage: 3.3V

Junction temperature range: 0-85 °C

< Back Next > Finish Cancel

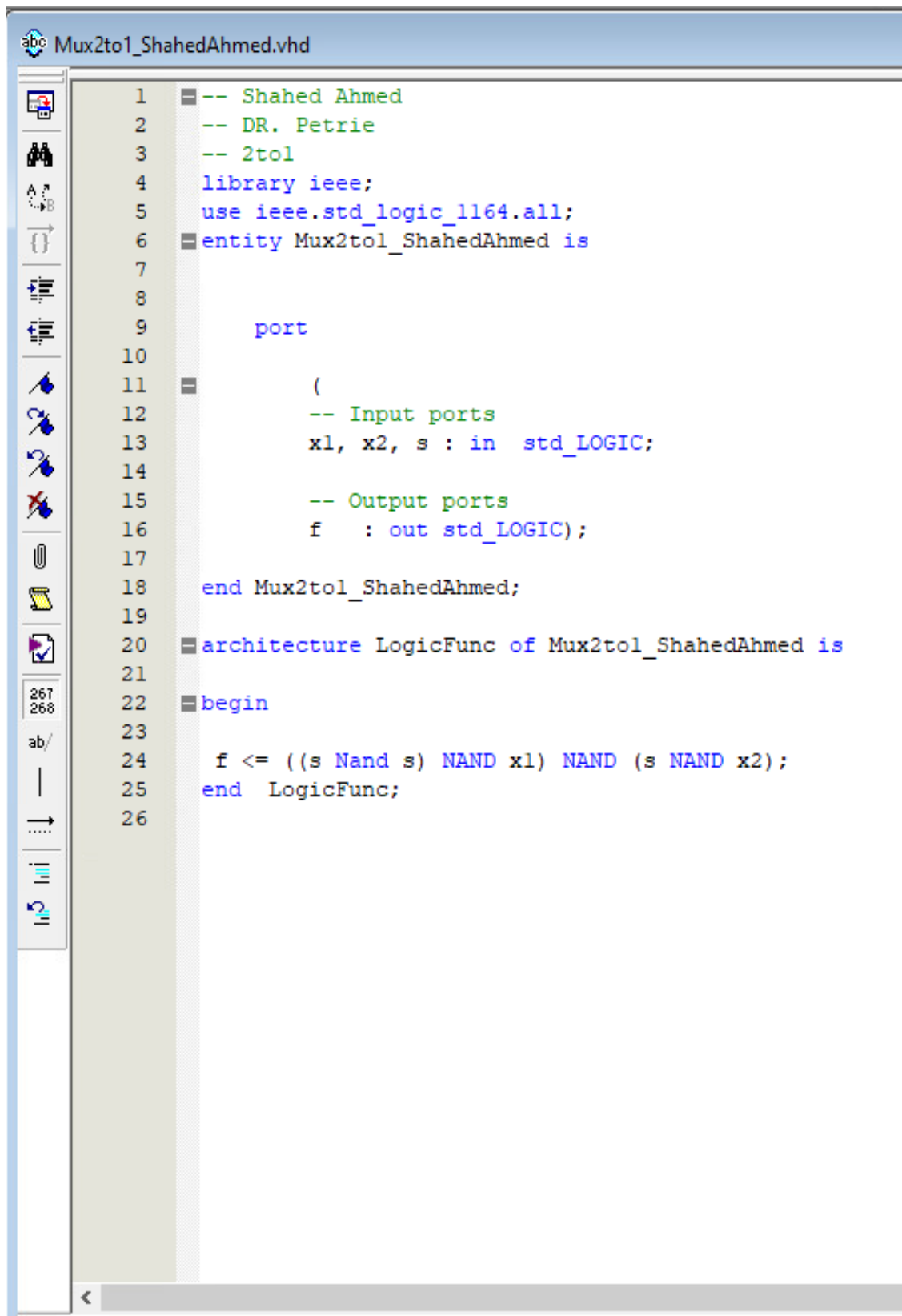
Mux2to1 Diagram



Truth Table

S	X1	X2	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

VHDL code



```
1  -- Shahed Ahmed
2  -- DR. Petrie
3  -- 2to1
4  library ieee;
5  use ieee.std_logic_1164.all;
6  entity Mux2to1_ShahedAhmed is
7
8
9      port
10
11      (
12          -- Input ports
13          x1, x2, s : in  std_LOGIC;
14
15          -- Output ports
16          f      : out std_LOGIC);
17
18  end Mux2to1_ShahedAhmed;
19
20  architecture LogicFunc of Mux2to1_ShahedAhmed is
21
22  begin
23
24      f <= ((s Nand s) NAND x1) NAND (s NAND x2);
25  end LogicFunc;
26
```

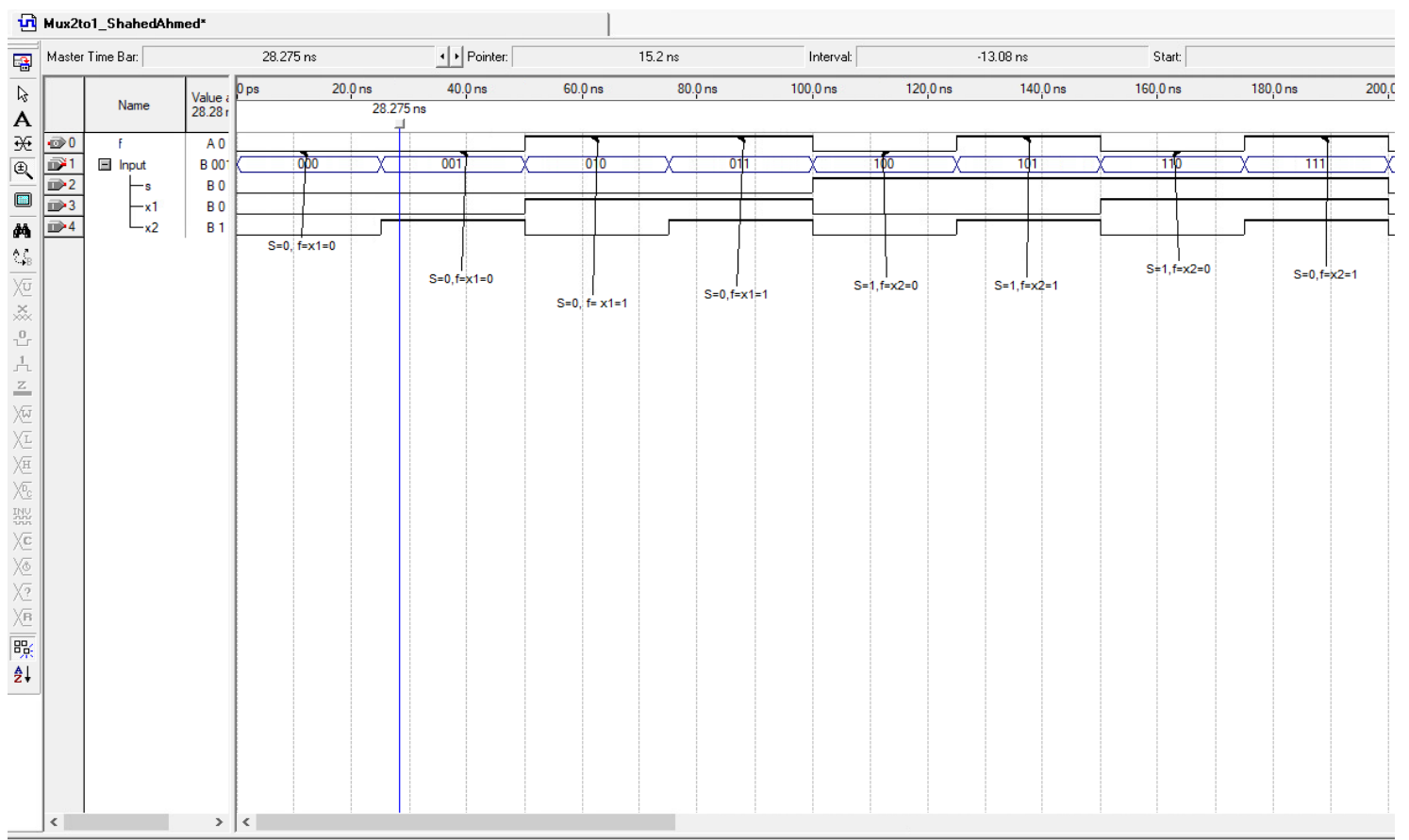
Successful Compilation

The screenshot displays the Quartus II IDE with the VHDL source code for `Mux2to1_ShahedAhmed.vhd` on the left and the `Compiler Tool` window on the right. The code defines a 2-to-1 multiplexer entity with inputs `x1`, `x2`, and `s`, and output `f`. The logic is implemented as a `LogicFunc` architecture using a NAND gate.

```
1  -- Shahed Ahmed
2  -- DR. Petrie
3  -- 2to1
4  library ieee;
5  use ieee.std_logic_1164.all;
6  entity Mux2to1_ShahedAhmed is
7
8
9
10     port
11     (
12         -- Input ports
13         x1, x2, s : in  std_LOGIC;
14
15         -- Output ports
16         f : out std_LOGIC);
17
18 end Mux2to1_ShahedAhmed;
19
20 architecture LogicFunc of Mux2to1_ShahedAhmed is
21
22 begin
23
24     f <= ((s NAND s) NAND x1) NAND (s NAND x2);
25 end LogicFunc;
26
```

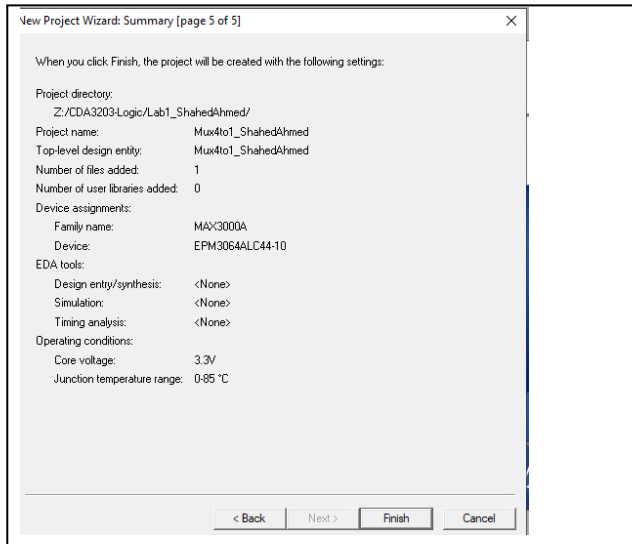
The `Compiler Tool` window shows the compilation progress for Analysis & Synthesis, Filter, Assembler, and Classic Timing Analyzer, all at 100% completion. The Full Compilation bar is also at 100%. A message box titled "Quartus II" confirms: "Full Compilation was successful (1 warning)".

Timing Diagram

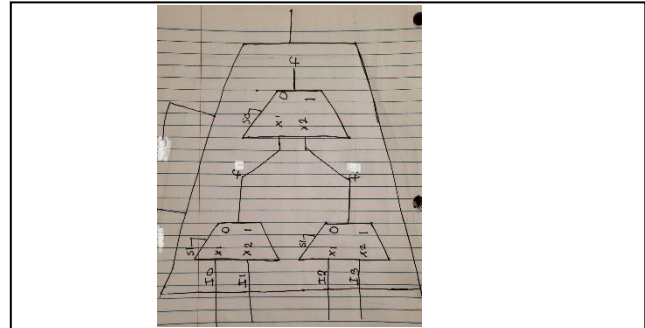


Part 3. Design and simulate a 4-to-1 Multiplexer (Mux) using only Mux2to1 components.

Project Wizard



Mux4to1 Diagram



Truth Table

S1	S0	F
0	0	I0
0	1	I1
1	0	I2
1	1	I3

VHDL code

```
Mux4to1_ShahedAhmed.vhd*
1  --Shahed Ahmed
2  -- DR.Petrie
3  -- 4 to 1
4  library ieee;
5  use ieee.std_logic_1164.all;
6  -- Library Clause(s) (optional)
7  -- Use Clause(s) (optional)
8  entity Mux4to1_ShahedAhmed is
9      port
10         (
11             -- Input ports
12             I0,I1,I2,I3,S1,S0: in  STD_LOGIC;
13             -- Output ports
14             f : out STD_LOGIC);
15
16 end Mux4to1_ShahedAhmed;
17
18
19
20 architecture Structure of Mux4to1_ShahedAhmed is
21
22     Signal f1,f2: STD_LOGIC;
23
24     COMPONENT Mux2to1_ShahedAhmed is
25
26     port
27
28         (
29             -- Input ports
30             x1, x2, s : in  STD_LOGIC;
31             -- Output ports
32             f : out STD_LOGIC);
33     End COMPONENT Mux2to1_ShahedAhmed;
34 Begin
35
36     Mux0: Mux2to1_ShahedAhmed Port MAP (I0,I1,S1,f1);
37     Mux1: Mux2to1_ShahedAhmed Port MAP (I2,I3,S1,f2);
38     Mux2: Mux2to1_ShahedAhmed Port MAP (f1,f2,S0,f);
39
40
41 end Structure;
```


Mux4to1_ShahedAhmed.vhd

Compiler Tool

Mux4to1_ShahedAhmed.vhd

```

1  --Shahed Ahmed
2  -- DR.Petrie
3  -- 4 to 1
4  library ieee;
5  use ieee.std_logic_1164.all;
6  -- Library Clause(s) (optional)
7  -- Use Clause(s) (optional)
8  entity Mux4to1_ShahedAhmed is
9      port
10         (
11             -- Input ports
12             I0,I1,I2,I3,S1,S0: in  STD_LOGIC;
13
14             -- Output ports
15             f : out STD_LOGIC);
16
17 end Mux4to1_ShahedAhmed;
18
19
20 architecture Structure of Mux4to1_ShahedAhmed is
21
22     Signal f1,f2: STD_LOGIC;
23
24     COMPONENT Mux2to1_ShahedAhmed is
25
26     port
27
28     (
29         -- Input ports
30         x1, x2, s : in  STD_LOGIC;
31
32         -- Output ports
33         f : out STD_LOGIC);
34 End COMPONENT Mux2to1_ShahedAhmed;
35 Begin
36
37 Mux0: Mux2to1_ShahedAhmed Port MAP (I0,I1,S1,f1);
38 Mux1: Mux2to1_ShahedAhmed Port MAP (I2,I3,S1,f2);
39 Mux2: Mux2to1_ShahedAhmed Port MAP (f1,f2,S0,f);
40
41 end Structure;

```

Compiler Tool

Analysis & Synthesis

Filter

Assembler

Classic Timing Analyzer

100 %

00:00:03

100 %

00:00:03

100 %

00:00:01

100 %

00:00:01

Full Compilation

100 %

00:00:08

Start

Stop

Report

Quartus II

Full Compilation was successful (1 warning)

OK

Part 4. Design and simulate an 8-to-1 Multiplexer (Mux) using only Mux4to1 and/or Mux2to1 components.

Project Wizard

New Project Wizard: Summary [page 5 of 5]

When you click Finish, the project will be created with the following settings:

Project directory:
Z:/CDA3203-Logic/Lab1_ShahedAhmed/

Project name: Mux8to1_ShahedAhmed

Top-level design entity: Mux8to1_ShahedAhmed

Number of files added: 2

Number of user libraries added: 0

Device assignments:

Family name: MAX3000A

Device: EPM3064ALC44-10

EDA tools:

Design entry/synthesis: <None>

Simulation: <None>

Timing analysis: <None>

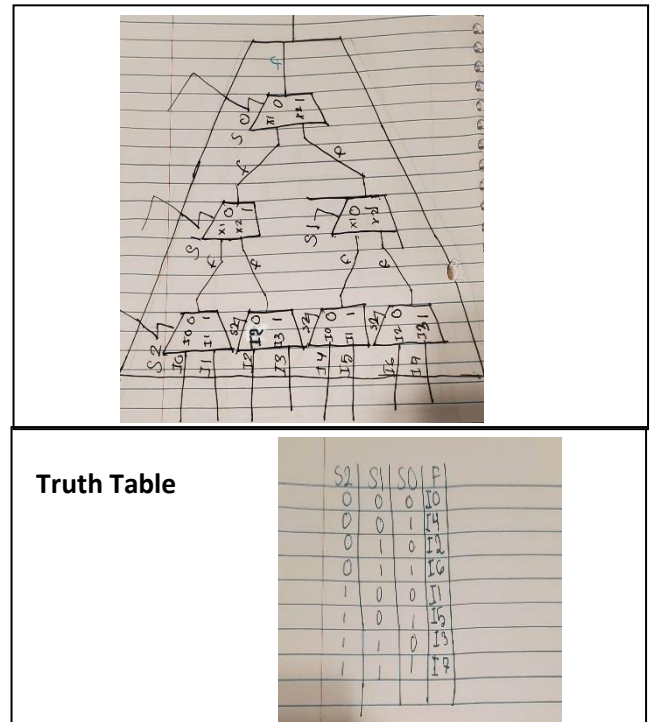
Operating conditions:

Core voltage: 3.3V

Junction temperature range: 0-85 °C

< Back Next > Finish Cancel

Mux8to1 Diagram



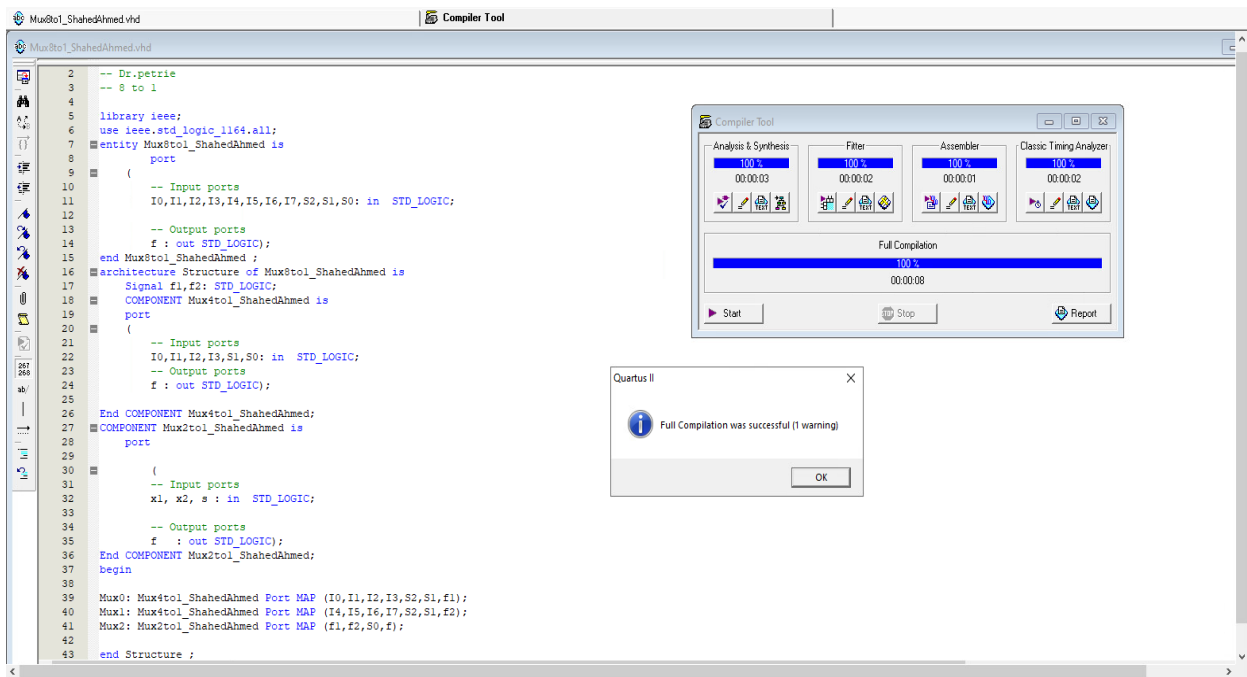
VHDL code

```

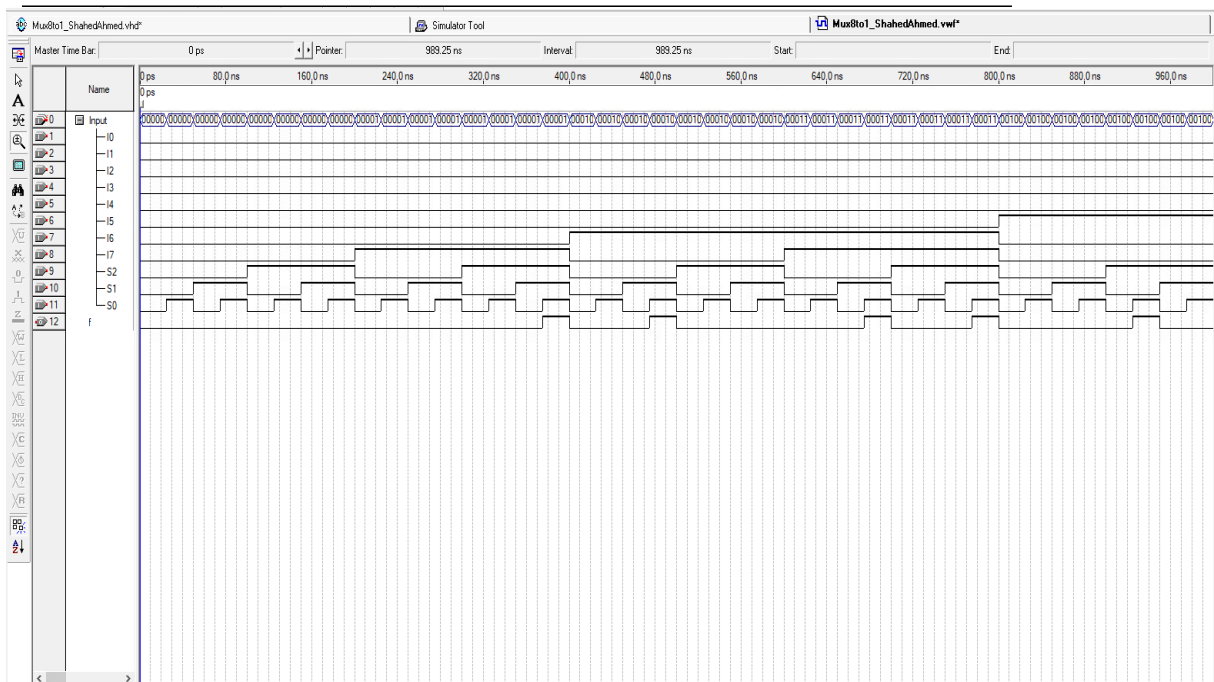
1  --ShahedAhmed
2  -- Dr.petrie
3  -- 8 to 1
4
5  library ieee;
6  use ieee.std_logic_1164.all;
7  entity Mux8to1_ShahedAhmed is
8      port
9      (
10         I0,I1,I2,I3,I4,I5,I6,I7,S2,S1,S0: in  STD_LOGIC;
11
12         f : out STD_LOGIC);
13  end Mux8to1_ShahedAhmed ;
14  architecture Structure of Mux8to1_ShahedAhmed is
15      signal f1,f2: STD_LOGIC;
16      component Mux4to1_ShahedAhmed is
17          port
18          (
19              -- Input ports
20              I0,I1,I2,I3,S1,S0: in  STD_LOGIC;
21              -- Output ports
22              f : out STD_LOGIC);
23      end component Mux4to1_ShahedAhmed;
24      component Mux2to1_ShahedAhmed is
25          port
26          (
27              -- Input ports
28              x1, x2, s : in  STD_LOGIC;
29              -- Output ports
30              f : out STD_LOGIC);
31      end component Mux2to1_ShahedAhmed;
32  begin
33      Mux0: Mux4to1_ShahedAhmed Port MAP (I0,I1,I2,I3,S2,S1,f1);
34      Mux1: Mux4to1_ShahedAhmed Port MAP (I4,I5,I6,I7,S2,S1,f2);
35      Mux2: Mux2to1_ShahedAhmed Port MAP (f1,f2,S0,f);
36
37  end Structure ;

```

Successful Compilation



Timing Diagram



Part 5. Design and simulate a 16-to-1 Multiplexer (Mux) using only Mux8to1, Mux4to1 and/or Mux2to1.

Project Wizard

New Project Wizard: Summary [page 5 of 5]

When you click Finish, the project will be created with the following settings:

Project directory:
Z:/CDA3203-Logic/Lab1_ShahedAhmed/

Project name: Mux16to1_ShahedAhmed

Top-level design entity: Mux16to1_ShahedAhmed

Number of files added: 3

Number of user libraries added: 0

Device assignments:

Family name: MAX3000A

Device: EPM3064ALC44-10

EDA tools:

Design entry/synthesis: <None>

Simulation: <None>

Timing analysis: <None>

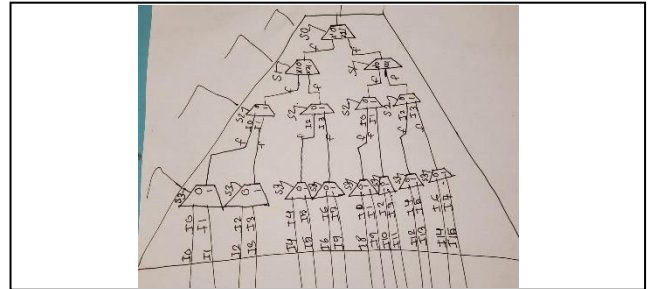
Operating conditions:

Core voltage: 3.3V

Junction temperature range: 0-85 °C

< Back Next > Finish Cancel

Mux16to1 Diagram



Truth Table

S3	S2	S1	S0	F
0	0	0	0	I0
0	0	0	1	I1
0	0	1	0	I2
0	0	1	1	I3
0	1	0	0	I4
0	1	0	1	I5
0	1	1	0	I6
0	1	1	1	I7
1	0	0	0	I8
1	0	0	1	I9
1	0	1	0	I10
1	0	1	1	I11
1	1	0	0	I12
1	1	0	1	I13
1	1	1	0	I14
1	1	1	1	I15

VHDL code

```

1  -- ShahedAhmed
2  -- Dr.Petrie
3  -- Mux16to1
4  library ieee;
5  use ieee.std_logic_1164.all;
6  entity Mux16to1_ShahedAhmed is
7      port
8      (
9          I0,I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,S3,S2,S1,S0: in  STD_LOGIC;
10         f : out STD_LOGIC);
11  end Mux16to1_ShahedAhmed;
12
13  architecture Structure of Mux16to1_ShahedAhmed is
14      signal f1,f2: STD_LOGIC;
15      component Mux8to1_ShahedAhmed is
16          port
17          (
18              I0,I1,I2,I3,I4,I5,I6,I7,S2,S1,S0: in  STD_LOGIC;
19              f : out STD_LOGIC);
20      end component Mux8to1_ShahedAhmed;
21
22      component Mux2to1_ShahedAhmed is
23          port
24          (
25              -- Input ports
26              x1, x2, s : in  STD_LOGIC;
27              -- Output ports
28              f : out STD_LOGIC);
29      end component Mux2to1_ShahedAhmed;
30
31  begin
32
33      Mux0: Mux8to1_ShahedAhmed Port MAP (I0,I1,I2,I3,I4,I5,I6,I7,S3,S2,S1,f1);
34      Mux1: Mux8to1_ShahedAhmed Port MAP (I8,I9,I10,I11,I12,I13,I14,I15,I3,S2,S1,f2);
35      Mux2: Mux2to1_ShahedAhmed Port MAP (f1,f2,S0,f);
36
37  end Structure;
38
39
40
41

```

Mux16to1_ShahedAhmed.vhd

Compiler Tool

```

1  -- ShahedAhmed
2  -- Dr.Petrie
3  -- Mux16to1
4  library ieee;
5  use ieee.std_logic_1164.all;
6  entity Mux16to1_ShahedAhmed is
7      port
8      (
9          I0,I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,S3,S2,S1,S0: in STD_LOGIC;
10         f : out STD_LOGIC);
11 end Mux16to1_ShahedAhmed;
12
13 architecture Structure of Mux16to1_ShahedAhmed is
14     signal f1,f2: STD_LOGIC;
15     component Mux8to1_ShahedAhmed is
16         port
17         (
18             I0,I1,I2,I3,I4,I5,I6,I7,S2,S1,S0: in STD_LOGIC;
19             f : out STD_LOGIC);
20
21     end component Mux8to1_ShahedAhmed;
22
23     component Mux2to1_ShahedAhmed is
24         port
25         (
26             -- Input ports
27             x1, x2, s : in STD_LOGIC;
28
29             -- Output ports
30             f : out STD_LOGIC);
31     end component Mux2to1_ShahedAhmed;
32
33     begin
34
35         Mux0: Mux8to1_ShahedAhmed Port MAP (I0,I1,I2,I3,I4,I5,I6,I7,S3,S2,S1,f1);
36         Mux1: Mux8to1_ShahedAhmed Port MAP (I8,I9,I10,I11,I12,I13,I14,I15,I3,S2,S1,f2);
37         Mux2: Mux2to1_ShahedAhmed Port MAP (f1,f2,S0,f);
38
39     end Structure;
40
41

```

Compiler Tool

Analysis & Synthesis: 100 % 00:00:04

Filter: 100 % 00:00:02

Assembler: 100 % 00:00:01

Classic Timing Analyzer: 100 % 00:00:01

Full Compilation: 100 % 00:00:08

Start Stop Report

Quartus II

Full Compilation was successful (1 warning)

OK

Timing Diagram

