

Faculty Name:

Faculty Sign & Date:

K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**

Roll No:

Grade/Marks:



1912060

Course Name:	Hardware Description Language Lab (2UXL401)	Semester:	IV
Date of Performance:	02/02/2021	Batch No:	B2

Experiment No: 2

Title: Study of Multiplexer (Dataflow, Behavioral, Structural)

Prof. Bhargavi Kaslikar

Aim and Objective of the Experiment:

Write a VHDL code for implementing a 4:1 multiplexer

- a) Using when else construct (Data Flow)
- b) Using structural architecture

Write a testbench to verify your results.

Write a VHDL code for 16: 1 mux using 4:1 mux and gates. (Structural)

To study basic types of architectures and to understand the use of concurrent statements.

COs to be achieved:

- **CO 1**: Use basic Concurrent and Sequential statements in VHDL and write codes for simple applications
- **CO 2**: Test a VHDL code and verify the circuit model.
- **CO 3**: Synthesize and Implement the designed circuits on CPLD/ FPGA.

HDL Laboratory Semester: IV Academic Year: 2020-21

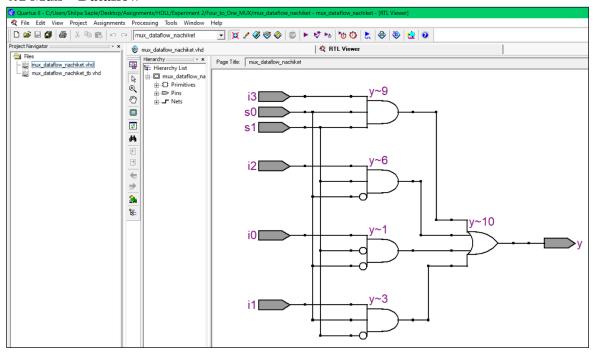


(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**

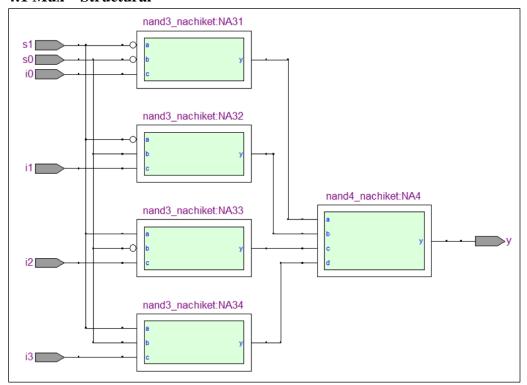


Logic circuits:

4:1 Mux - Dataflow



4:1 Mux – Structural



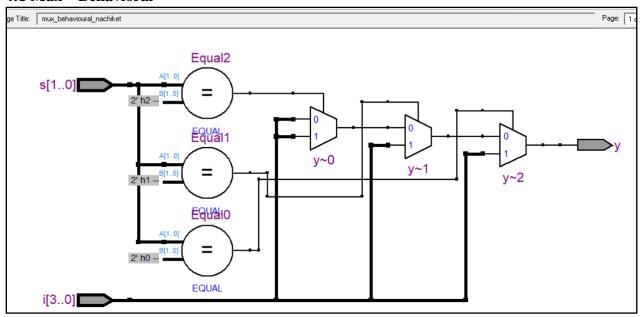
HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



4:1 Mux - Behavioral



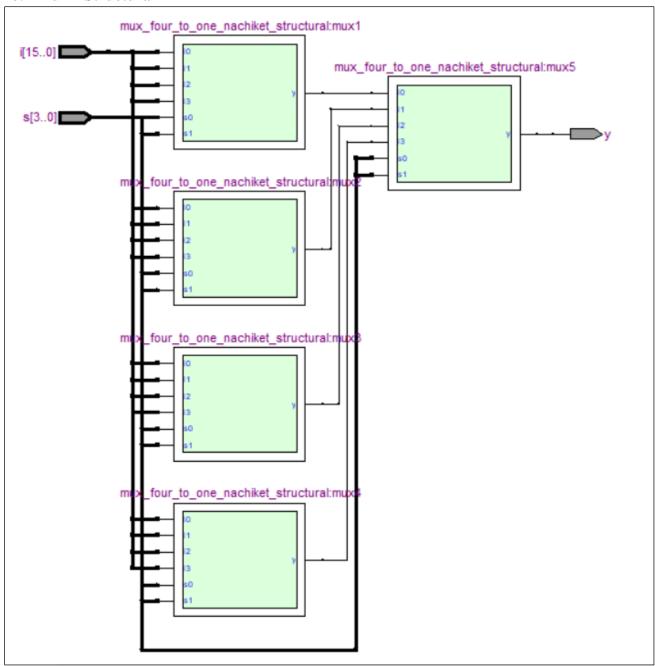
HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



16:1 Mux - Structural



HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



Work to be uploaded

```
Code for 4:1 Mux , Code for 16:1 mux
4:1 Data Flow:
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity mux_dataflow_nachiket is
port(
  i0,i1,i2,i3,s0,s1: In std_logic;
  y: Out std_logic
  );
  end mux_dataflow_nachiket;
architecture mux_dataflow_nachiket_arch of mux_dataflow_nachiket is
signal u1,u0:std_logic;
begin
  u1 \le not s1;
  u0 \le not s0;
  y \le (u1 \text{ and } u0 \text{ and } i0) \text{ or } (u1 \text{ and } s0 \text{ and } i1) \text{ or } (s1 \text{ and } u0 \text{ and } i2) \text{ or } (s1 \text{ and } s0 \text{ and } i3);
end mux_dataflow_nachiket_arch;
```

HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



```
4:1 Structural:
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity mux_four_to_one_nachiket_structural is
       port(
              i0,i1,i2,i3,s0,s1: In std_logic;
              y: Out std logic
end mux_four_to_one_nachiket_structural;
architecture mux four to one nachiket structural arch of mux four to one nachiket structural
is
       component nand3_nachiket is
              port(
                      a,b,c: In std_logic;
                     y: Out std_logic
              ):
  end component;
  component nand4 nachiket is
              port(
                      a,b,c,d: In std_logic;
                     y: Out std_logic
  end component;
signal u1,u0,o0,o1,o2,o3:std_logic;
begin
       u1 \leq not s1;
  u0 \le not s0;
       NA31: nand3_nachiket port map(u1,u0,i0,o0);
       NA32: nand3 nachiket port map(u1,s0,i1,o1);
       NA33: nand3_nachiket port map(s1,u0,i2,o2);
       NA34: nand3 nachiket port map(s1,s0,i3,o3);
       NA4: nand4 nachiket port map(o0,o1,o2,o3,y);
end mux_four_to_one_nachiket_structural_arch;
```

HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



```
4:1 Behavioral:
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity mux_behavioural_nachiket is
       port(
               i: In std_logic_vector(3 downto 0);
               s: In std_logic_vector(1 downto 0);
               y: Out std_logic
               );
       end mux_behavioural_nachiket;
architecture mux behavioural nachiket arch of mux behavioural nachiket is
       begin
               process(s,i)
                      begin
                              if(s="00") then
                                     y \le i(0);
                              elsif(s="01") then
                                     y \le i(1);
                              elsif(s="10") then
                                     y \le i(2);
                              else
                                     y \le i(3);
                              end if;
               end process;
end mux_behavioural_nachiket_arch;
```

HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



```
16:1 Mux using 4:1 Mux – Structural:
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity mux_16_to_one_nachiket is
       port(
              i: In std_logic_vector(15 downto 0);
              s: In std_logic_vector(3 downto 0);
              y: Out std_logic
              ):
end mux 16 to one nachiket;
architecture mux_16_to_one_nachiket_arch of mux_16_to_one_nachiket is
       component mux_four_to_one_nachiket_structural is
                      port(
                             i0,i1,i2,i3,s0,s1: In std_logic;
                             y: Out std_logic
                      );
       end component;
signal o: std_logic_vector(3 downto 0);
begin
       mux1: mux_four_to_one_nachiket_structural port map(i(0),i(1),i(2),i(3),s(0),s(1),o(0));
       mux2: mux_four_to_one_nachiket_structural port map(i(4),i(5),i(6),i(7),s(0),s(1),o(1));
       mux3: mux_four_to_one_nachiket_structural port map(i(8),i(9),i(10),i(11),s(0),s(1),o(2));
       mux4:
                                   mux_four_to_one_nachiket_structural
                                                                                             port
map(i(12),i(13),i(14),i(15),s(0),s(1),o(3));
       mux5: mux four to one nachiket structural port map(o(0),o(1),o(2),o(3),s(2),s(3),y);
end mux_16_to_one_nachiket_arch;
```

HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



```
Test bench for 4:1 mux and 16:1 Mux and simulation waveform of the same.
4:1 Testbench:
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity mux_dataflow_nachiket_tb is
end mux_dataflow_nachiket_tb;
architecture mux_dataflow_nachiket_tb_arch of mux_dataflow_nachiket_tb is
component mux dataflow nachiket is
port(
   i0,i1,i2,i3,s0,s1: In std_logic;
   y: Out std_logic
   );
end component;
signal i0,i1,i2,i3,s0,s1,y: std_logic;
begin
    uut: mux_dataflow_nachiket port map(i0,i1,i2,i3,s0,s1,y);
   process begin
      i0 < = '1';
      i1 <= '0';
      i2 < = '1';
      i3 < = '0';
      s0 < = '0';
      s1 <= '0';
      wait for 20ns;
      s0 < = '0';
      s1 <= '1';
      wait for 20ns;
      s0 <= '1';
      s1 <= '0';
```

HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



```
wait for 20ns;
      s0 < = '1';
      s1 <= '1';
      wait for 20ns;
  end process;
 end;
4:1 Testbench Behavioral:
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity mux_behavioural_nachiket_tb is
end mux_behavioural_nachiket_tb;
architecture mux_behavioural_nachiket_tb_arch of mux_behavioural_nachiket_tb is
       component mux_behavioural_nachiket is
              port(
                      i: In std_logic_vector(3 downto 0);
                      s: In std logic vector(1 downto 0);
                      y: Out std_logic
                      );
       end component;
       signal i: std_logic_vector(3 downto 0);
       signal s: std_logic_vector(1 downto 0);
       signal y: std_logic;
       begin
              uut: mux_behavioural_nachiket port map(i,s,y);
              process begin
                      i \le "1010";
                      s \le "00";
                      wait for 20ns;
```

HDL Laboratory Semester: IV Academic Year: 2020-21

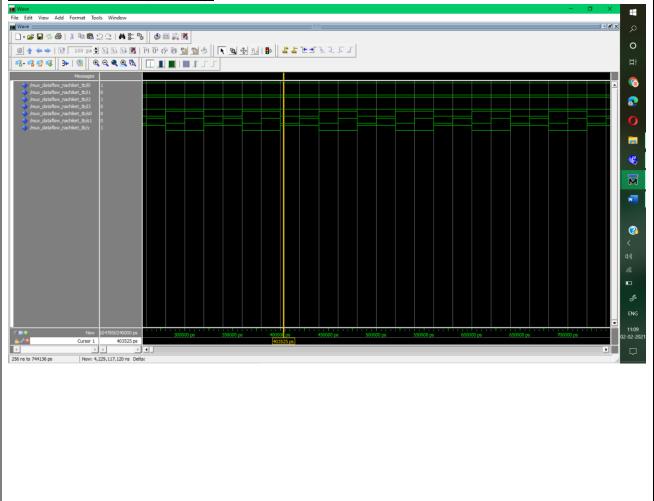


(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



```
s <= "01"; wait for 20ns; s <= "10"; wait for 20ns; s <= "11"; wait for 20ns; end process; end;
```

4:1 MUX waveform (dataflow):



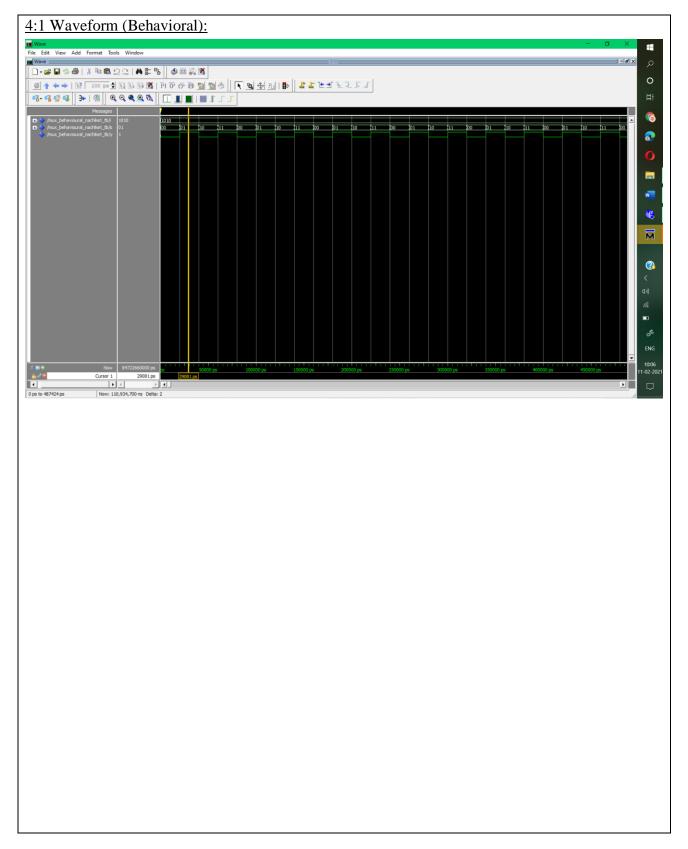
HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University)

Department of Electronics Engineering





HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



```
16:1 Testbench:
library ieee;
use ieee.std logic 1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity mux_16_to_one_nachiket_tb is
end mux_16_to_one_nachiket_tb;
architecture mux 16 to one nachiket tb arch of mux 16 to one nachiket tb is
       component mux_16_to_one_nachiket is
              port(
                     i: In std_logic_vector(15 downto 0);
                      s: In std_logic_vector(3 downto 0);
                      y: Out std_logic
       end component;
       signal i: std_logic_vector(15 downto 0);
       signal s: std_logic_vector(3 downto 0);
       signal y: std_logic;
       begin
               uut: mux_16_to_one_nachiket port map(i,s,y);
               process begin
                      i \le "1011010010101011";
                      s \le "0000";
                      wait for 20ns;
                      s \le "0001";
                      wait for 20ns;
```

HDL Laboratory Semester: IV Academic Year: 2020-21







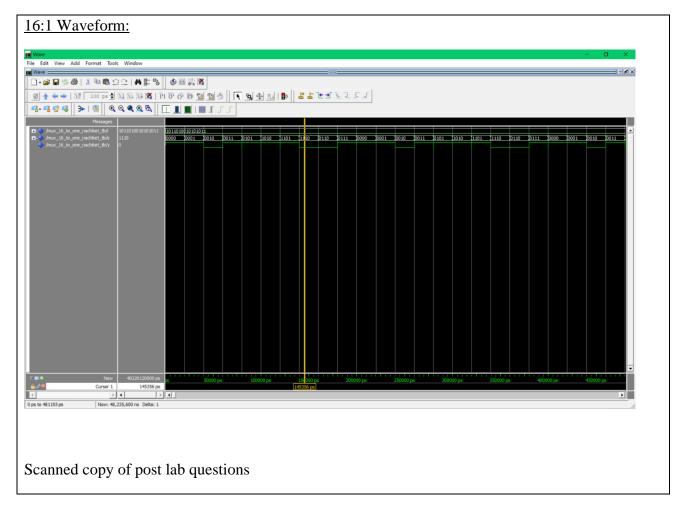
```
s \le "0010";
                       wait for 20ns;
                       s \le "0011";
                       wait for 20ns;
                       s \le "0101";
                       wait for 20ns;
                       s <= "1010";
                       wait for 20ns;
                       s <= "1101";
                       wait for 20ns;
                       s \le "1110";
                       wait for 20ns;
                       s \le "0110";
                       wait for 20ns;
                       s \le "0111";
                       wait for 20ns;
               end process;
end mux_16_to_one_nachiket_tb_arch;
```

HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**





HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



Post Lab Subjective/Objective type Questions:

Upload Answer of following question before coming to next laboratory.

Q1. Analyze the following code and explain its output.

```
library ieee;
use ieee.std logic 1164.all;
entity xyz is
                 port(g1,g2,g3:in std logic;
                        sel: in std logic vector(2 downto 0);
                        q:out std logic vector(7 downto 0)
end xyz;
architecture xyz a of xyz is
           signal q s: std logic vector(7 downto 0);
Begin
    with sel select q s<=
           "01111111" when "000",
           "10111111" when "001",
           "110111111" when "010",
           "11101111" when "011",
           "11110111" when "100",
           "11111011" when "101",
           "11111101" when "110",
           "11111110" when "111",
           "11111111" when others;
     q \le q s when (g1 \text{ and not } g2 \text{ and not } g3) = '1'
     else "111111111";
end xyz a a;
```

HDL Laboratory Semester: IV Academic Year: 2020-21



K. J. Somaiya College of Engineering, Mumbai-77 (A Constituent College of Somaiya Vidyavihar University)

Department of Electronics Engineering



NALLIKET NAIK - 1912060 - B2 HOLL - Ext. 2 - Paster B Got) Matthe Strive in the lode: Last line is given as led reyz-a-a' but it should be rey and reyz-a-a' but it should be rey and reyz-a-a' but it should be rey and reyz-a; The given entry reyz has 24 inpets, gl, 92, 93 (std-loyie is 'o' or ') and o set (vector of 3 bits) and one output 9 of 8 bits. To the lode we are also using a signal 9-5 which is a vector of 8 bits The value of 9-5 depends on that of set: Whatever be the value of set, that bit of 9-5 which is at the corresponding downal value index of set will be 0. Rest 7 bits will be 1 Indexing starts from 0. ea: if set = '101" - (5)10 then the 6th lift of 9-5 will be 0 & others will be 1 9-5 = "1111011" This is happens for set in the grange 'O00" to '11" For any other value of set, 9-5 = "11111111"		Page No.:	γουνλ
Attack Error in the code: Last line is given as led xy2-a-a' but it should be the lend xy2-a; The given entity the code to and the content of the lend the state of the lend of the self of the lend	VACHIKET / VAIK - 1912060 - B2	Date:	
Attack Error in the code: Last line is given as led xy2-a-a' but it should be the lend xy2-a; The given entity the code to and the content of the lend the state of the lend of the self of the lend	HOLL - EXP-2 - POSTLAB		*
About the lode is given as lend xyz-a-a' but it should be very lend xyz-a; The given entity xyz has 4 inpute, gl, g2, g3 (std-loyis is 'o' or 'i') and e Set (vector of 3 bits) and one output q of 8 bits. In the lode we are also using a signal q-s which is a vector of 8 bits The value of q-s depends on that of sel. Whatever be the value of sel, that bit of q-s which is at the corresponding desimal value involves of sel will be 1. Indiving starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of q-s will be 0 & others will be 1 q-s = "1111011"			
The given entry xyz has 4 inputs, gl, g2, g3 (std-logic i.e 'o' or 'i') and o sel (vector of 3 bits) and one output q of 8 bits. In the lode we are also using a signal 9-s which is a vector of 8 bits The value of 9-s depends on that of sel. Whatever be the value of sel, that bit of 9-s which is at the corresponding decimal value involven of sel will be 0. Rest 7 bits will be 1 Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-s will be 0 l others will be 1 9-s = "1111011"	Mistake Error in the lode :-		
The given entry xyz has 4 inputs, gl, g2, g3 (std-logic i.e 'o' or 'i') and o sel (vector of 3 bits) and one output q of 8 bits. In the lode we are also using a signal 9-s which is a vector of 8 bits The value of 9-s depends on that of sel. Whatever be the value of sel, that bit of 9-s which is at the corresponding decimal value involven of sel will be 0. Rest 7 bits will be 1 Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-s will be 0 l others will be 1 9-s = "1111011"	Last line is given	as end	nyz-a-a;
The given entry xyz has 4 inputs, gl, g2, g3 (std-logic i.e 'o' or 'i') and o sel (vector of 3 bits) and one output q of 8 bits. In the lode we are also using a signal 9-s which is a vector of 8 bits The value of 9-s depends on that of sel. Whatever be the value of sel, that bit of 9-s which is at the corresponding decimal value involven of sel will be 0. Rest 7 bits will be 1 Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-s will be 0 l others will be 1 9-s = "1111011"	but it should be my end my	12-a;	<u> </u>
To the lode we are also using a Signal 9-s which is a veder of 8 bits "The valve of 9-s depends on that of Sel. "Whateve, be the value of sel, that bit of 9-s which is at the corresponding decimal value indexe of sel will be 0. Rest 7 bits will be 1. Indexing starts from 0. eg: if sel = "101" → (5)10 then the 6th bit of 9-s will be 0 l others will be 1 9-s = "1111011"	V		
To the lode we are also using a Signal 9-s which is a veder of 8 bits "The valve of 9-s depends on that of Sel. "Whateve, be the value of sel, that bit of 9-s which is at the corresponding decimal value indexe of sel will be 0. Rest 7 bits will be 1. Indexing starts from 0. eg: if sel = "101" → (5)10 then the 6th bit of 9-s will be 0 l others will be 1 9-s = "1111011"			
To the lode we are also using a Signal 9-s which is a veder of 8 bits "The valve of 9-s depends on that of Sel. "Whateve, be the value of sel, that bit of 9-s which is at the corresponding decimal value indexe of sel will be 0. Rest 7 bits will be 1. Indexing starts from 0. eg: if sel = "101" → (5)10 then the 6th bit of 9-s will be 0 l others will be 1 9-s = "1111011"	· The given entity xyz has 4	inputs,	
To the lode we are also using a Signal 9-5 which is a veder of 8 bits "The valve of 9-5 depends on that of Sel. "Whatever, be the value of sel, that bit of 9-5 which is at the corresponding decimal value inches of sel will be 0. Rest 7 bils will be 1. Indexing starts from 0. eg: if sel = "101" → (5)10 then the 6th bit of 9-5 will be 0 l others will be 1 9-5 = "1111011"	g1, g2, g3 (std-logic is '0' or 1)	and o	
In the lode we are also using a Signal 9-5 which is a veder of 8 bits "The valve of 9-5 depends on that of Sel. "Whatever, be the value of sel, that bit of 9-5 which is at the corresponding desimal value indexe of sel will be 0. Rest 7 bits will be 1. Indexing starts from 0. eg: if sel = "101" → (5)10 then the 6th lit of 9-5 will be 0 & others will be 1 9-5 = "1111011"	sel (vector of 3 bits) and one	output 9	7
The value of 9-5 depends on that of Sel. " Whatever, be the value of Sel, that bit of 9-5 which is at the remespending desimal value imposes of Sel will be 0. Rest 7 bits will be 1. Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-5 will be 0 & others will be 1. 1. 9-5 = "11111011"	8 bits.		
The value of 9-5 depends on that of Sel. " Whatever, be the value of Sel, that bit of 9-5 which is at the remespending desimal value imposes of Sel will be 0. Rest 7 bits will be 1. Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-5 will be 0 & others will be 1. 1. 9-5 = "11111011"	To the desired of the second o		
The value of 9-5 depends on that of Sel. " Whatever, be the value of sel, that bit of 9-5 which is at the semespending desimal value impleme of sel will be 0. Rest 7 bits will be 1 Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-5 will be 0 & others will be 1 9-5 = "11111011"	in the lode we wie wise w	sing a	lak
· Whatever, be the value of sel, that bit of g-s which is at the serresponding decimal value imposes of sel will be 0. Rest 7 bits will be 1 Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of g-s will be 0 & others will be 1	Signal 9-5 which is a veice	y of o c	
· Whatever, be the value of sel, that bit of g-s which is at the serresponding decimal value imposes of sel will be 0. Rest 7 bits will be 1 Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of g-s will be 0 & others will be 1	" The value of 9-5 depends on	that of	!
· Whatever, be the value of sel, that bit of g-s which is at the serresponding decimal value imposes of sel will be 0. Rest 7 bits will be 1 Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of g-s will be 0 & others will be 1	Sel.		
g-s which is at the serresponding decimal value implies of sel will be 0. Rest 7 bik will be 1. Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-5 will be 0 & others will be 1 9-5 = "1111011"			
g-s which is at the serresponding decimal value implies of sel will be 0. Rest 7 bik will be 1. Indexing starts from 0. eg: if sel = "101" -> (5)10 then the 6th bit of 9-5 will be 0 & others will be 1 9-5 = "1111011"	· Whatever be the value of sel, for	hat bit o	<i>‡</i>
Indexing starts from 0. eg: if $sel = "101" \rightarrow (5)_{10}$ then the 6th bit of $q-s$ will be 0 l others will be 1. $q-s = "1111011"$	g-s which is at the somesp	onding deci	mal
Indexing starts from 0. eg: if $sel = "101" \rightarrow (5)_{10}$ then the 6 th bit of $q-s$ will be 0 l others will be 1 $q-s = "11111011"$	value imposes of sel will be	O. V Rest	7
Indexing starts from 0. eg: if sel = "101" \rightarrow (5)10 then the 6 th bit of 9-5 will be 0 & others will be 1 9-5 = "1111011"	bik will be 1		
eg: if sel = "101" \rightarrow (5)10 then the 6 th bit of 9-5 will be 0 & others will be 1 9-5 = "1111011"	Indexing starts from O.	E = 1.7	
· ·	V		ath
· ·	eg: if sel = "101" \rightarrow (5)10 H	ien the	6
· · · · · · · · · · · · · · · · · · ·	bit of 9-5 will be 0 & of	hers will	be
This is happens for sel in the Hange	2 , 9-5 = 11111011		
"000" to "III"		ha Mara	
000 ,0 111	inis to nappens for set in h	ne range	
For any other value at hel ans = "Illilli"	Francisco de la constante del	= "////////////////////////////////////	
VI UNG OFFICE OF SEC.)	The true of set, t		

HDL Laboratory Semester: IV Academic Year: 2020-21



K. J. Somaiya College of Engineering, Mumbai-77 (A Constituent College of Somaiya Vidyavihar University)





1912060	•		•	•		•	•
· Now	then	the the	outp	enpression ut g	on (gl = g-s	and not	g 2 and no
get	the value	(1	11//1/				
· The	expression	ڏر	Irve	only	when	93 = 0	$\frac{1}{2}, \frac{2}{3} = 1$

Q.2 write a test bench for the above code.

1912010
1912060
Q.2) Testbench :-
lih. 1000:
library ieee;
Use ieee-std-logic-1164.all;
entity xyz-tb is
and N area th'
the arch of xyz-th is
architecture xyz-tb-arch of xyz-tb is
component nyz is
0 1
al az 93: in sta-logic,
sel: in std-logic - vector (2 downto 0).
set in starting of downth o)
q: out std-logic-vector (7 downto 0);
);
end component;
1 2 - 2 (+d-10c)
signal 9,92,93. Std-logic-vector (2 downto 0). signal sel: std-logic-vector (7 downto 0);
signal set: sta-logic-vector (7 downto o);
signal q: Std-logic-Vector (
V
begin 92, 93, sel, 9)
begin out: xyz port map (91, 92, 93, sel, 9);
process begin
Process

HDL Laboratory Semester: IV Academic Year: 2020-21



(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



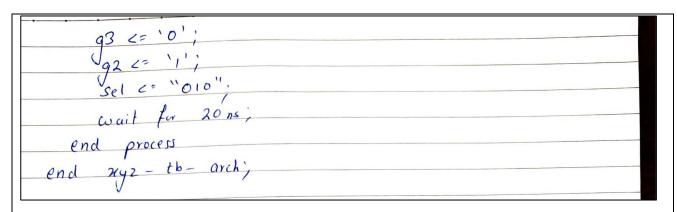
sel <= "000". wait for 20 ns; Sel <= "001" Sel <= "010"; wait for 20 ns; sel' <= " 011"; Sel <= " 110"; wait for 20 ns;

HDL Laboratory Semester: IV Academic Year: 2020-21

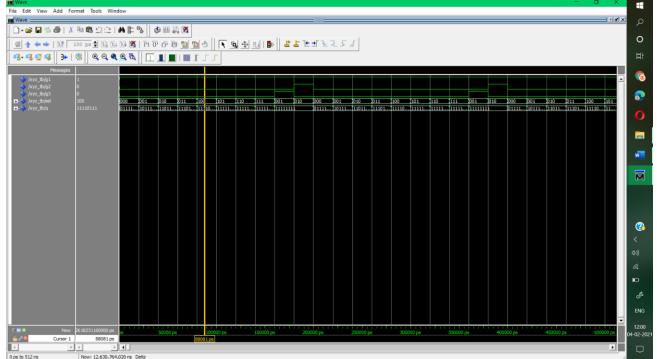


(A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**





Output of the above code:



HDL Laboratory Semester: IV Academic Year: 2020-21



K. J. Somaiya College of Engineering, Mumbai-77 (A Constituent College of Somaiya Vidyavihar University) **Department of Electronics Engineering**



Conclusion:
MACHIKET NAIK - 1912060 - B2
HD11 - EXP2
CONCLUSION
A multiplener or MUX is a circuit that
selects a single output from multiple inputs.
in this experiment we work the work
for 4:1 this by using both data flow and sinchral architecture.
and sinchral architecture.
in dataflow, we specify the functionality
of an entity by defining the
In smotoral architectory
needed are declared first then instances of
lomponents wheated with particular mappings of
signal sorresponds the to the various pins of
lomponents.
4 Hence in dataflow of 4:1 mox, we used
the SOP eauchion where as in smurrar,
we created 2 more components: 3 input & 4 input
NAND gates.
I Lostly, we implemented a 16:1 mux using I y:1 mux (mux tree). Hence we used structural architecture as the circuit made use of predefined components
3, 4:1 mux (mux tree). Hence we used structural
architecture as the circuit made use of predefined components

Signature of faculty in-charge with Date:

HDL Laboratory Semester: IV Academic Year: 2020-21