**Index Number - 210098R**

**Name - De Silva APC**

**Lab task –**

Designing and developing a seven-segment display that displays numbers with the use of a lookup table to map which light segments should be on when displaying each number. Further more, the previously developed components were used (previously designed AU+ ) to make it more functional by displaying the number which was added together and outputted by the AU.

**Filled Lookup Table –**

**Table

Description automatically generated with low confidence**

**Screenshots –**

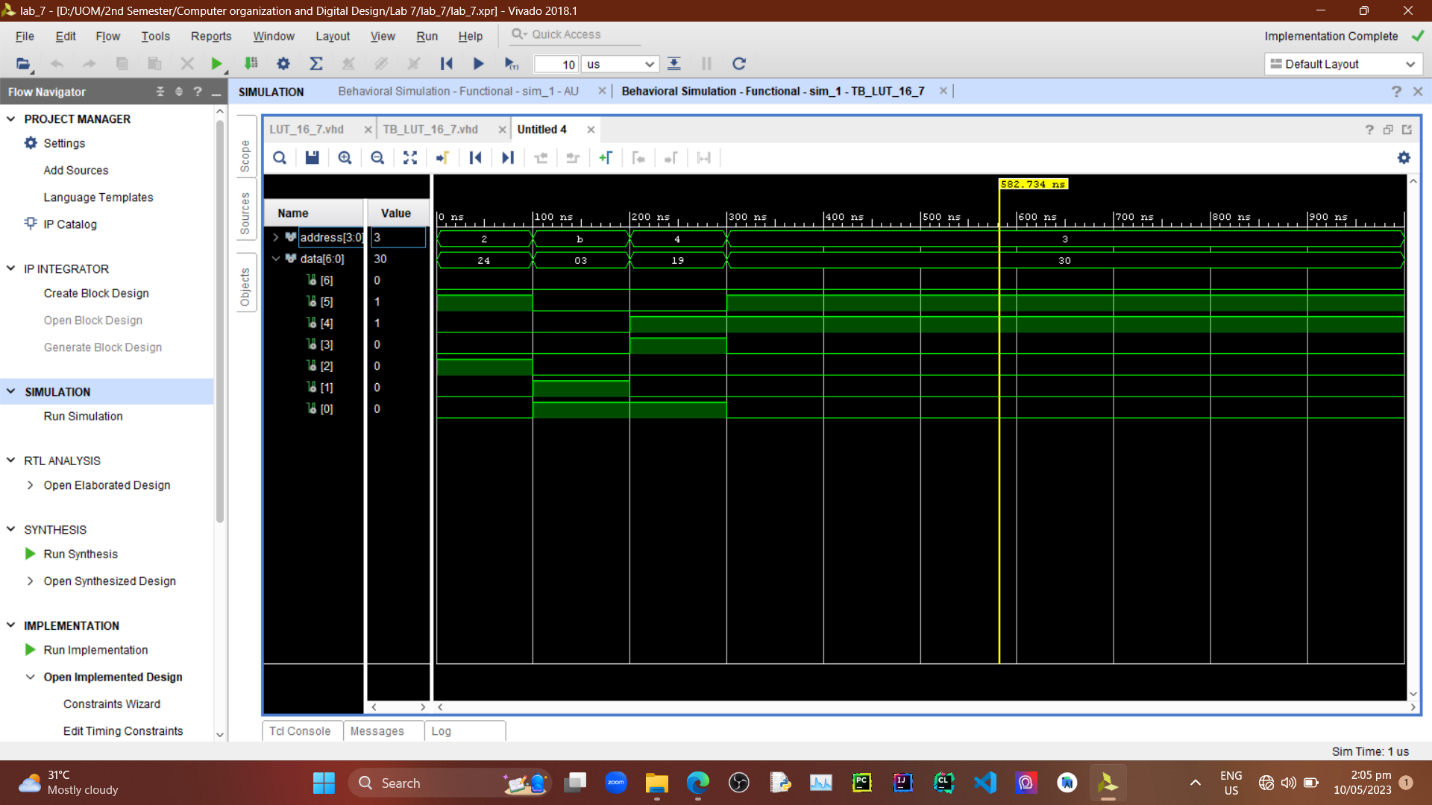
**(Final simulation)**

\*(Segmants of four bits from the end of binary conversion of the index number has been used to test)

**A screenshot of a computer

Description automatically generated**

**(Testing the Lookup Table implementation)**

****

**VHD Files –**

***AU Design file(AU\_7\_Seg.vhd) –***

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 05/10/2023 02:12:58 PM

-- Design Name:

-- Module Name: AU\_7\_seg - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity AU\_7\_seg is

Port ( A : in STD\_LOGIC\_VECTOR (3 downto 0);

Clk : in STD\_LOGIC;

RegSel : in STD\_LOGIC;

S\_LED : out STD\_LOGIC\_VECTOR (3 downto 0);

S\_7Seg : out STD\_LOGIC\_VECTOR (6 downto 0);

Carry : out STD\_LOGIC;

Zero : out STD\_LOGIC);

end AU\_7\_seg;

architecture Behavioral of AU\_7\_seg is

component AU

Port ( A : in STD\_LOGIC\_VECTOR (3 downto 0);

RegSel : in STD\_LOGIC;

Clk : in STD\_LOGIC;

S : out STD\_LOGIC\_VECTOR (3 downto 0);

Zero : out STD\_LOGIC;

Carry : out STD\_LOGIC);

END component;

component LUT\_16\_7

Port ( address : in STD\_LOGIC\_VECTOR (3 downto 0);

data : out STD\_LOGIC\_VECTOR (6 downto 0));

END component;

signal OutputFeeder : std\_logic\_vector ( 3 downto 0 ):="0000";

begin

AU\_0 : AU

PORT MAP(

A => A,

RegSel =>RegSel,

Clk =>Clk,

Zero => Zero,

Carry => Carry,

S => OutputFeeder

);

S\_LED <= OutputFeeder;

LUT\_16\_7\_0 : LUT\_16\_7

PORT MAP(

address =>Outputfeeder,

data =>S\_7Seg

);

end Behavioral;

***LUT Design file(LUT\_16\_7.vhd) –***

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 05/10/2023 01:22:53 PM

-- Design Name:

-- Module Name: LUT\_16\_7 - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use ieee.numeric\_std.all; -- Importing numeric library

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity LUT\_16\_7 is

Port ( address : in STD\_LOGIC\_VECTOR (3 downto 0);

data : out STD\_LOGIC\_VECTOR (6 downto 0));

end LUT\_16\_7;

architecture Behavioral of LUT\_16\_7 is

type rom\_type is array (0 to 15) of std\_logic\_vector(6 downto 0);

signal sevenSegment\_ROM : rom\_type := (

"1000000", -- 0

"1111001", -- 1

"0100100", -- 2

"0110000", -- 3

"0011001", -- 4

"0010010", -- 5

"0000010", -- 6

"1111000", -- 7

"0000000", -- 8

"0010000", -- 9

"0001000", -- a

"0000011", -- b

"1000110", -- c

"0100001", -- d

"0000110", -- e

"0001110" -- f

);

begin

data <= sevenSegment\_ROM(to\_integer(unsigned(address)));

end Behavioral;

***LUT Testbench file(TB\_LUT\_16\_7.vhd) –***

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 05/10/2023 01:40:13 PM

-- Design Name:

-- Module Name: TB\_LUT\_16\_7 - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity TB\_LUT\_16\_7 is

-- Port ( );

end TB\_LUT\_16\_7;

architecture Behavioral of TB\_LUT\_16\_7 is

COMPONENT LUT\_16\_7

PORT(address : in STD\_LOGIC\_VECTOR (3 downto 0);

data : out STD\_LOGIC\_VECTOR (6 downto 0));

END COMPONENT;

SIGNAL address : STD\_LOGIC\_VECTOR (3 downto 0);

SIGNAL data : STD\_LOGIC\_VECTOR (6 downto 0);

begin

UUT: LUT\_16\_7 PORT MAP(

address => address,

data => data);

process

begin

address <= "0010";

WAIT FOR 100 ns;

address <= "1011";

WAIT FOR 100 ns;

address <= "0100";

WAIT FOR 100 ns;

address <= "0011";

WAIT;

end process;

end Behavioral;

***Final Testbench file(TB\_AU\_7\_seg.vhd) –***

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 05/10/2023 02:32:43 PM

-- Design Name:

-- Module Name: TB\_AU\_7\_seg - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity TB\_AU\_7\_seg is

-- Port ( );

end TB\_AU\_7\_seg;

architecture Behavioral of TB\_AU\_7\_seg is

component AU\_7\_seg

Port ( A : in STD\_LOGIC\_VECTOR (3 downto 0);

Clk : in STD\_LOGIC;

RegSel : in STD\_LOGIC;

S\_LED : out STD\_LOGIC\_VECTOR (3 downto 0);

S\_7Seg : out STD\_LOGIC\_VECTOR (6 downto 0);

Carry : out STD\_LOGIC;

Zero : out STD\_LOGIC);

end component;

signal A,S\_LED : STD\_LOGIC\_VECTOR (3 downto 0) ;

signal S\_7Seg : STD\_LOGIC\_VECTOR (6 downto 0) ;

signal RegSel,Clk,Zero,Carry : STD\_LOGIC :='0' ;

begin

UUT : AU\_7\_seg

PORT MAP(

A => A,

S\_LED =>S\_LED,

Regsel => Regsel,

Clk => Clk,

Zero => Zero,

Carry =>Carry,

S\_7Seg => S\_7Seg

);

process

begin

Clk <= not(Clk);

wait for 2ns;

end process;

process

begin

A<= "0010";

RegSel <= '1';

wait for 100ns;

RegSel <= '0';

A<= "1011";

wait for 100ns;

RegSel <= '1';

A <= "0100";

wait for 100ns;

RegSel <= '0';

A <= "0011";

wait for 100ns;

end process;

end Behavioral;

**How to Display results using a single 7 segment Display –**

Here we are using 4 bit numbers as input. Hence the output would be a 4 bit number and may have another carry bit. Using 4 bits we can display 16 different numbers and that exactly matches with how many different numbers can be displayed using a single 7 segmant display using hexadecimal representations. So by using extra indicator to indicate the carry bit we can display the results without a issue using both indicator and the seven segment display.

**Conclusions –**

Basys 3 board has a common anode 7 segmant displays which can display up to 4 digits.(i.e. four 7 segmant displays in a sequence.)

Using a 7 segmant display we can display every digit used in hexadecimal number system too.

Lookup tables can be used to implement circuits by using it as a memory(ROM).