***Name – De Silva A.P.C.***

***Index No – 210098R***

***Group – CSE***

**Assigned lab task -** Design and develop a half adder, full adder and a Ripple carry adder using Vivado. Here, the Full adder should be implemented using two half adders and then the 4 bit ripple adder should be implemented using full adders.

**Truth tables and Boolean expressions simplification –**

*Half adder*

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **Sum** | **Carry** |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

Using sum of the products method: Sum = A’B+AB’ (i.e. A XOR B) , Carry = A.B

*Full adder*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **B** | **Cin** | **Sum** | **Cout** |
| 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 |

Using sum of the products method: Sum = A’B’Cin + A’BC’ + AB’Cin’ + ABCin

= A’(B’Cin + BC’in) + A(B’C’in + BCin )

= A( B XOR Cin) + A(B XOR C)’

= A XOR (B XOR C) = A XOR B XOR C

Then using a Kmap :

BCin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 13 | 2 |
| 1 | 4 | 15 | 17 | 6 |

A

Cout = AB + BCin +ACin

**HA.VHD**

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

-- Company:

-- Engineer:

--

-- Create Date: 03/18/2023 12:32:39 AM

-- Design Name:

-- Module Name: HA - 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 HA is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

S : out STD\_LOGIC;

C : out STD\_LOGIC);

end HA;

architecture Behavioral of HA is

begin

S <= A xor B ;

C <= A and B ;

end Behavioral;

**TB\_HA.VHD**

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

-- Company:

-- Engineer:

--

-- Create Date: 03/18/2023 12:38:30 AM

-- Design Name:

-- Module Name: TB\_HA - 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\_HA is

-- Port ( );

end TB\_HA;

architecture Behavioral of TB\_HA is

COMPONENT HA

PORT(A,B : IN STD\_LOGIC;

S,C : OUT STD\_LOGIC);

END COMPONENT;

SIGNAL A,B : std\_logic;

SIGNAL S,C : std\_logic;

begin

UUT: HA PORT MAP(

A => A,

B => B,

S => S,

C => C

);

process

begin

A <= '0';

B <= '0';

WAIT FOR 100 ns;

B <= '1';

WAIT FOR 100 ns;

A <= '1';

B <= '0';

WAIT FOR 100 ns;

B <= '1';

WAIT;

end process;

end Behavioral;

**FA.VHD**

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

-- Company:

-- Engineer:

--

-- Create Date: 03/18/2023 12:58:43 AM

-- Design Name:

-- Module Name: FA - 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 FA is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C\_in : in STD\_LOGIC;

S : out STD\_LOGIC;

C\_out : out STD\_LOGIC);

end FA;

architecture Behavioral of FA is

component HA

port (

A: in std\_logic;

B: in std\_logic;

S:out std\_logic;

C: out std\_logic);

end component;

signal HA0\_S, HA0\_C,HA1\_S, HA1\_C : std\_logic;

begin

HA\_0 : HA

port map (

A => A,

B => B,

S => HA0\_S,

C => HA0\_C);

HA\_1 : HA

port map (

A => HA0\_S,

B => C\_in,

S => HA1\_S,

C => HA1\_C);

S <= HA1\_S;

C\_out <= HA0\_C or HA1\_C;

end Behavioral;

**TB\_FA.VHD**

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

-- Company:

-- Engineer:

--

-- Create Date: 03/18/2023 01:49:53 AM

-- Design Name:

-- Module Name: TB\_FA - 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\_FA is

-- Port ( );

end TB\_FA;

architecture Behavioral of TB\_FA is

COMPONENT FA

PORT(A,B,C\_in : IN STD\_LOGIC;

S,C\_out : OUT STD\_LOGIC);

END COMPONENT;

SIGNAL A,B,C\_in : std\_logic;

SIGNAL S,C\_out : std\_logic;

begin

UUT: FA PORT MAP(

A => A,

B => B,

S => S,

C\_in => C\_in,

C\_out => C\_out

);

process

begin

A <= '0';

B <= '0';

C\_in <= '0';

WAIT FOR 100 ns;

C\_in <= '1';

WAIT FOR 100 ns;

B <= '1';

C\_in <= '0';

WAIT FOR 100 ns;

C\_in <= '1';

WAIT FOR 100 ns;

A <= '1';

B <= '0';

C\_in <= '0';

WAIT FOR 100 ns;

C\_in <= '1';

WAIT FOR 100 ns;

B <= '1';

C\_in <= '0';

WAIT FOR 100 ns;

C\_in <= '1';

WAIT;

end process;

end Behavioral;

**RCA\_4.VHD**

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

-- Company:

-- Engineer:

--

-- Create Date: 03/18/2023 11:27:04 AM

-- Design Name:

-- Module Name: RCA\_4 - 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 RCA\_4 is

Port ( A0 : in STD\_LOGIC;

A1 : in STD\_LOGIC;

A2 : in STD\_LOGIC;

A3 : in STD\_LOGIC;

B0 : in STD\_LOGIC;

B1 : in STD\_LOGIC;

B2 : in STD\_LOGIC;

B3 : in STD\_LOGIC;

C\_in : in STD\_LOGIC;

S0 : out STD\_LOGIC;

S1 : out STD\_LOGIC;

S2 : out STD\_LOGIC;

S3 : out STD\_LOGIC;

C\_out : out STD\_LOGIC);

end RCA\_4;

architecture Behavioral of RCA\_4 is

component FA

port(

A: in std\_logic;

B: in std\_logic;

C\_in : in std\_logic;

S: out std\_logic;

C\_out : out std\_logic);

end component;

SIGNAL FA0\_S, FA0\_C , FA1\_S,FA1\_C, FA2\_S, FA2\_C, FA3\_S, FA3\_C

: std\_logic;

begin

FA\_0 : FA

port map (

A => A0,

B => B0,

C\_in => '0',

S=> S0,

C\_out => FA0\_C);

FA\_1 : FA

port map (

A => A1,

B => B1,

C\_in => FA0\_C,

S=> S1,

C\_out => FA1\_C);

FA\_2 : FA

port map (

A => A2,

B => B2,

C\_in => FA1\_C,

S=> S2,

C\_out => FA2\_C);

FA\_3 : FA

port map (

A => A3,

B => B3,

C\_in => FA2\_C,

S=> S3,

C\_out => C\_out);

end Behavioral;

**TB\_4\_RCA.VHD**

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

-- Company:

-- Engineer:

--

-- Create Date: 03/18/2023 11:36:31 AM

-- Design Name:

-- Module Name: TB\_4\_RCA - 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\_4\_RCA is

-- Port ( );

end TB\_4\_RCA;

architecture Behavioral of TB\_4\_RCA is

COMPONENT RCA\_4

PORT(A0,A1,A2,A3,B0,B1,B2,B3,C\_in : IN STD\_LOGIC;

S0,S1,S2,S3,C\_out: OUT STD\_LOGIC);

END COMPONENT;

SIGNAL A0,A1,A2,A3,B0,B1,B2,B3,C\_in: std\_logic;

SIGNAL S0,S1,S2,S3,C\_out : std\_logic;

begin

UUT: RCA\_4 PORT MAP(

A0 => A0,

A1 => A1,

A2 => A2,

A3 => A3,

B0 => B0,

B1 => B1,

B2 => B2,

B3 => B3,

C\_in => C\_in,

C\_out => C\_out,

S0 => S0,

S1 => S1,

S2 => S2,

S3 => S3

);

process

begin

A0 <= '0';

A1 <= '1';

A2 <= '0';

A3 <= '0';

B0 <= '1';

B1 <= '0';

B2 <= '1';

B3 <= '1';

C\_in <= '0';

WAIT FOR 100 ns;

A0 <= '0';

A1 <= '1';

A2 <= '0';

A3 <= '1';

B0 <= '1';

B1 <= '1';

B2 <= '0';

B3 <= '0';

C\_in <= '0';

WAIT FOR 100 ns;

A0 <= '0';

A1 <= '1';

A2 <= '1';

A3 <= '1';

B0 <= '1';

B1 <= '0';

B2 <= '1';

B3 <= '1';

C\_in <= '0';

WAIT FOR 100 ns;

A0 <= '1';

A1 <= '1';

A2 <= '1';

A3 <= '1';

B0 <= '0';

B1 <= '0';

B2 <= '1';

B3 <= '1';

C\_in <= '0';

WAIT FOR 100 ns;

A0 <= '1';

A1 <= '0';

A2 <= '0';

A3 <= '1';

B0 <= '0';

B1 <= '0';

B2 <= '0';

B3 <= '1';

C\_in <= '0';

WAIT FOR 100 ns;

A0 <= '1';

A1 <= '1';

A2 <= '0';

A3 <= '1';

B0 <= '1';

B1 <= '1';

B2 <= '0';

B3 <= '0';

C\_in <= '0';

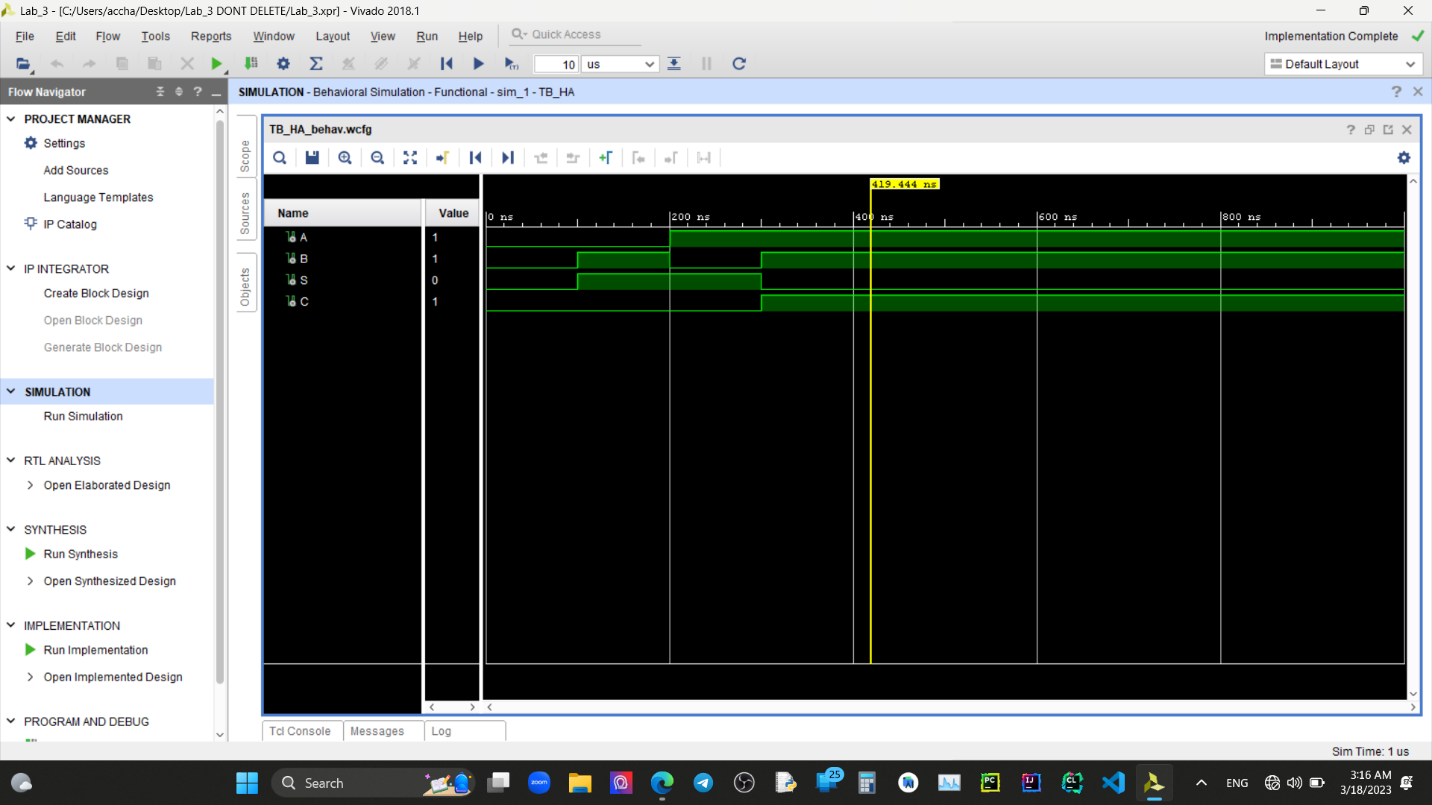
WAIT;

end process;

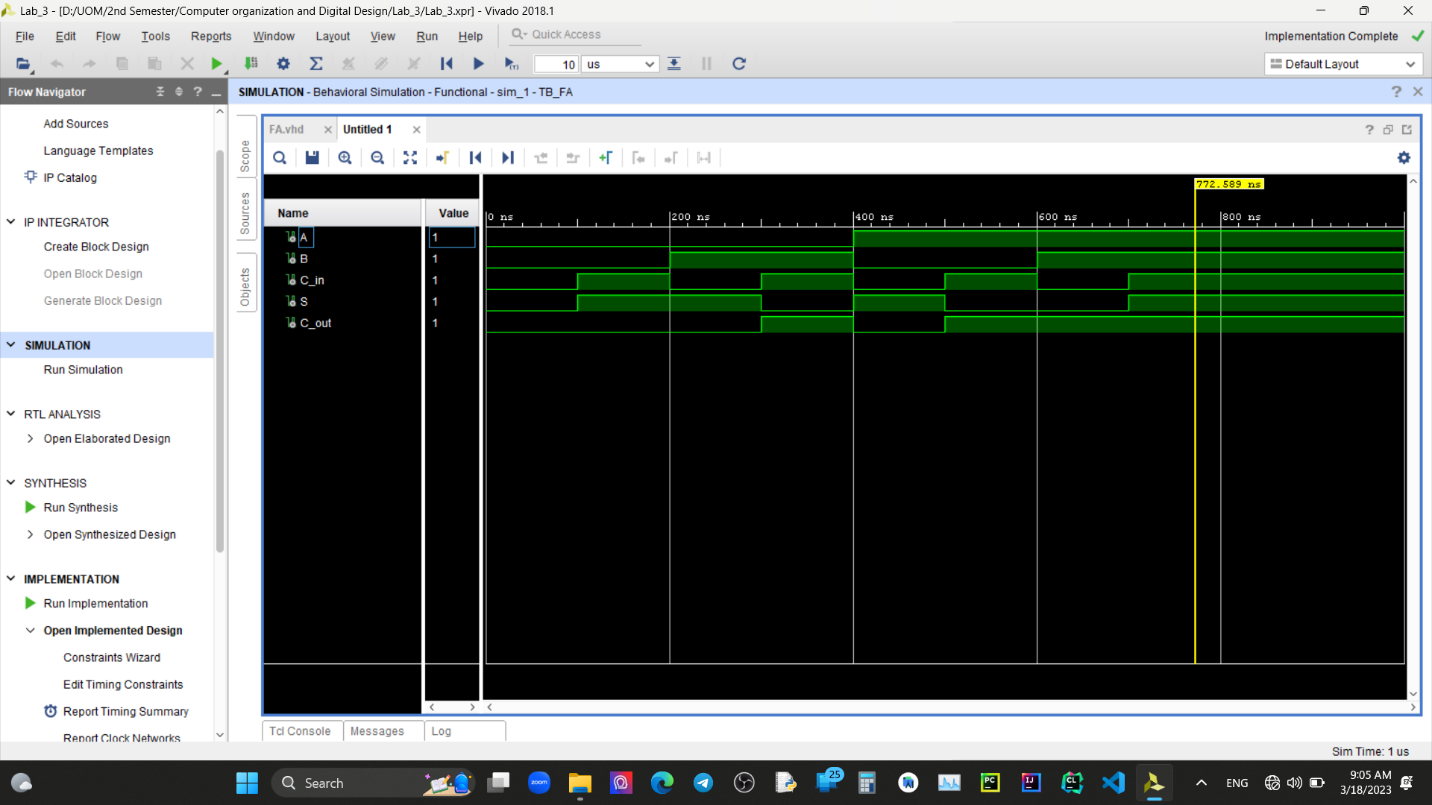
end Behavioral;

**Timing diagrams –**

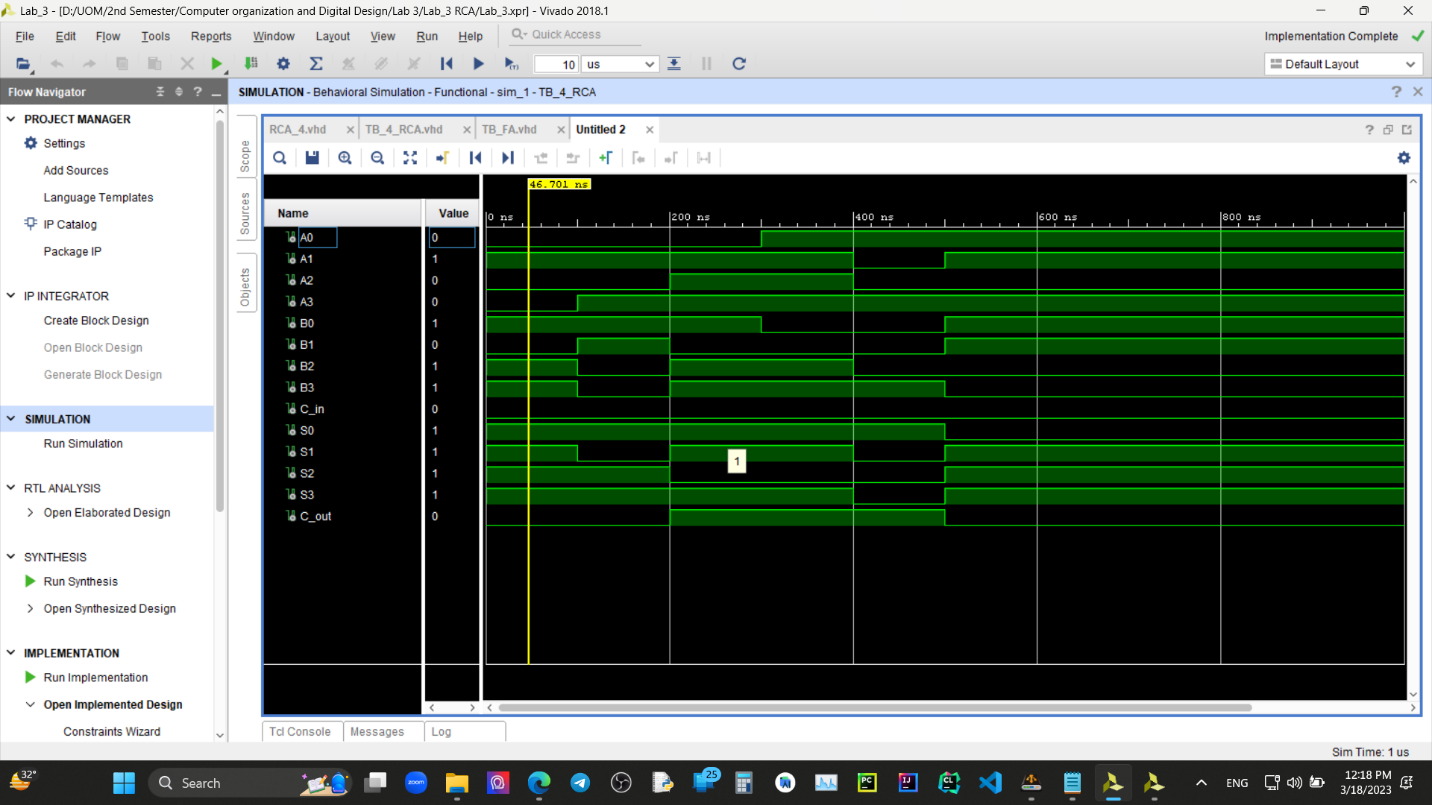
**Half adder –**

****

**Full Adder –**

****

**RCA –**

****

**Discussion –** After an operation if there is an overflow there is no way to represent that using only three LEDs. For an example if we add 1000 and 1011, this will output only 0011 without notifying about the remaining carry bit. To indicate about the carry bit we need that extra LED.

**Conclusion -** We can use basic components to build more complex components. Here we build a half adder using basic logic, then built a full adder using half adders and then build RCA using the full adders.