LAB 2: Digital Design

1) Electronic Voting Machine:

A company has 3 (Equal Ownership) partners say A, B, C. During their Daily business operation they have to take a lot of decisions. Since they have different views and approaches to solve the same problem, they often have conflict while making decisions. This decisions making (Criticizing one's idea etc.) is ruining their friendships. They have come to you for help. They want you to make a device that allows them to vote electronically.

- Each partner will have a switch that can produce a one or zero
- The device has a LED that will glow if and only if when 2 or more votes are at Logic One.

Design a digital logic machine that implements their requirement.

2) Rock Paper Scissor Game:

The Rock Paper Scissors is a guessing game played between two people, Player 1 (P1) and Player 2 (P2). Each player has three buttons namely rock, paper and scissors. The player makes a choice among the three buttons and chooses one. The other player does the same with his set of buttons at the same time.

Depending on the choices they make the game ends with one of the players winning or a tie.

The Rules:

P1 wins if

1. P1 chooses rock and P2 chooses scissors or

2. P1 chooses paper and P2 chooses rock or

3. P1 chooses scissors and P2 chooses paper.

P2 wins if

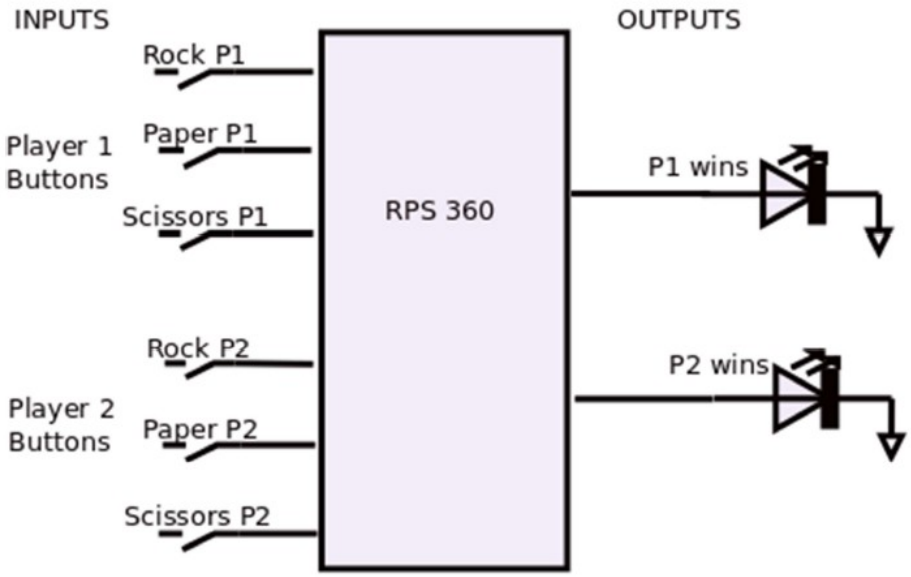
1. P1 chooses rock and P2 chooses paper or

2. P1 chooses paper and P2 chooses scissors or

3. P1 chooses scissors and P2 chooses rock.

Every other case is a tie.

Design a circuit that implements the game. Each player gets 3 buttons. There are 2 outputs indicating which player has won. Connect the outputs to LEDs. The LED corresponding to the winner glows. If there is a tie, both LEDs are off. Your design can assume that each player will choose exactly one button. Binary value of 1 means pressing a button.



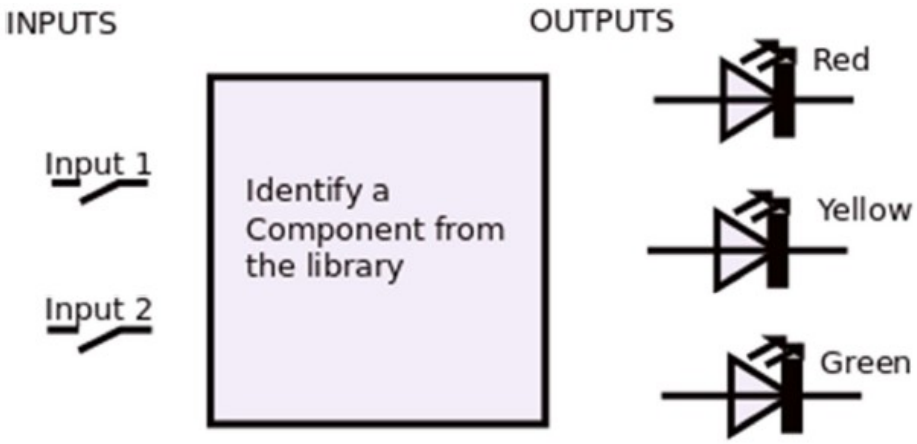
3) Traffic Light Controller:

Use a component from the library to design a traffic controller unit. The Inputs to the component is 2 input switches and outputs are the three traffic lights. The requirement is as follows. When the inputs are

. 00 - Red light must glow

. 01 - Yellow light must glow

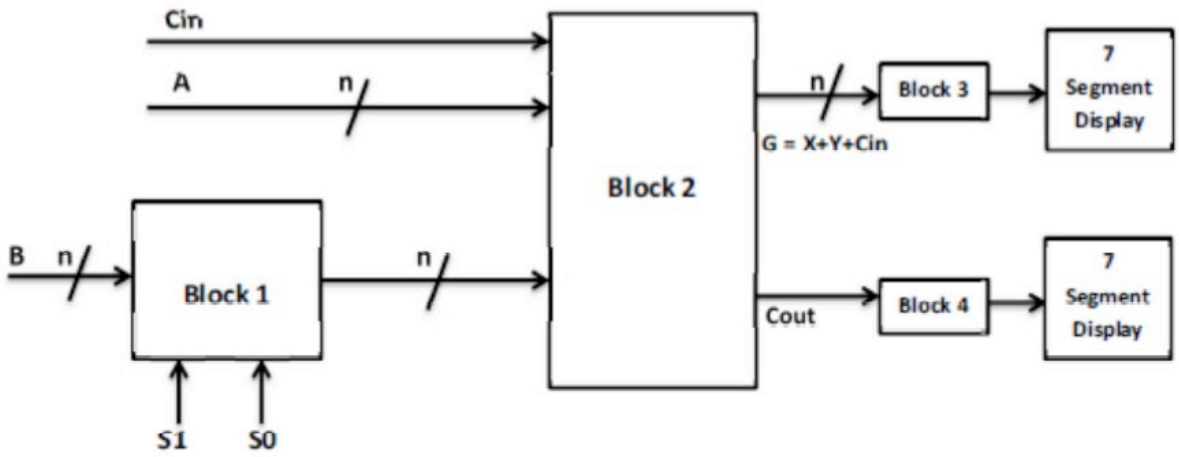
10 - Green light must glow 11 – All the lights glow



4. **Arithmetic Logic Unit (ALU)** (BONUS QUESTION)

Description: Design and Test a simple 8 function 4---bit two's complement ALU. A digital circuit that carries out arithmetic and logical operations is called as an ALU. It is the basic block of a central processing unit (CPU) of a computer.

Block Diagram:



(Design for n =4)

Figure 3: Block Diagram of ALU

S1	S0	Y	Cin=0	Cin=1
0	0	B	G=A+B (Add)	G=A+B+1
0	1	B'	G=A+B'	G=A+B'+1
1	0	All 0's	G=A (Transfer)	G=A+1(Increment)
1	1	All 1's	G=A---1 (Decrement)	G=A (Transfer)

Table 1: ALU Functionality

B' denotes bitwise complement of B.

B' denotes bitwise complement of B.

Blocks Description:

Block 1: This is a digital circuit which *selects* one of the several inputs and has some extra logic which implements part of the functionality of the ALU as given in Table 1.

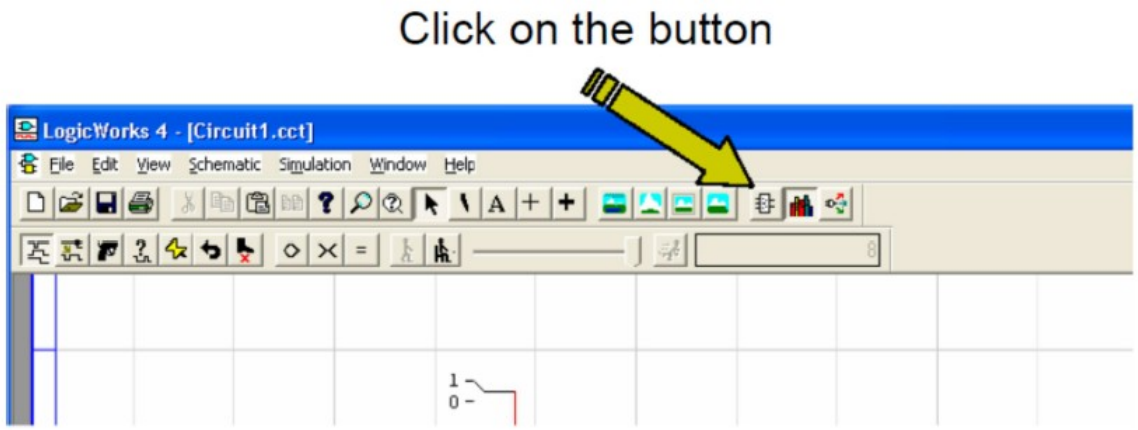
Block 2: This is a 4---bit Adder (n=4). You can find the block in the 'Parts Palette'.

Block 3 & 4: These two blocks are PROM. PROM: Programmable Read Only Memory, it a digital memory which stores binary values (0's and 1's).

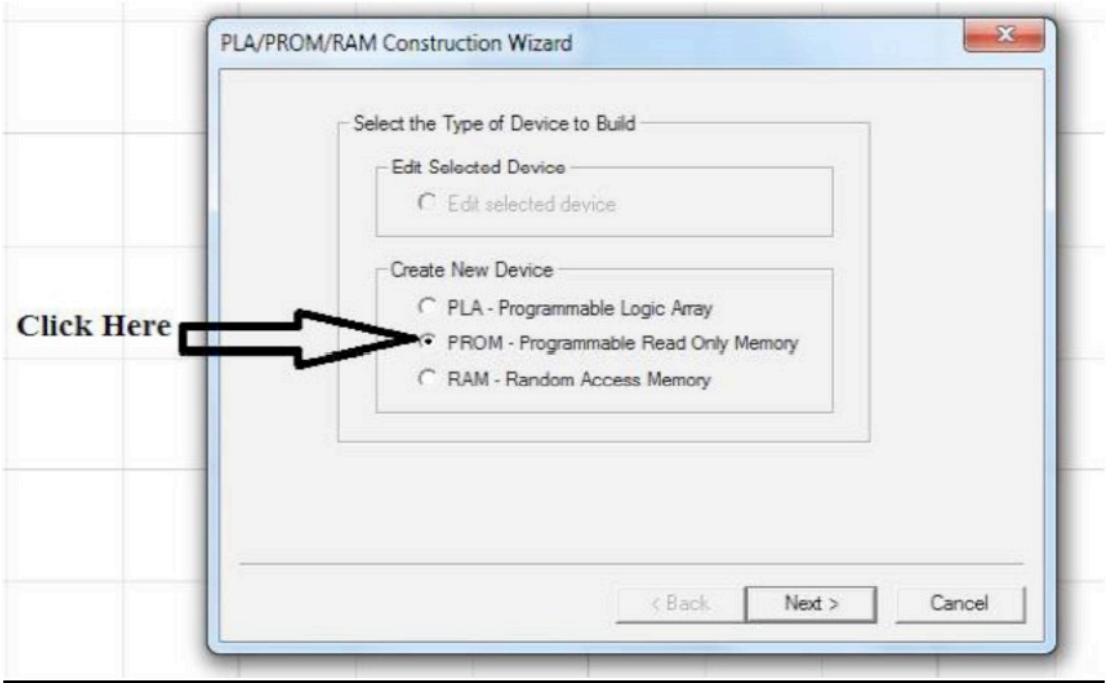
7 Segment Display: These are display devices used to display binary, Hexadecimal, Octal and Decimal numbers. These are used in digital clocks, electronic meters and other electronic devices. For both '7 Segment Display' blocks use **7---Seg Disp Inv---red,** search this component in the parts palette.

Steps for building a PROM:

Step 1:



Click on the button pointed by the arrow as shown in the above figure Step 2:



Step 3A: This step is to create PROM for Block 3

Select the following for Block 3:

Address Lines: 4 Bits per Word: 8 Data Entry Method: Enter hex data manually.

Hex data for block 3 as follows:

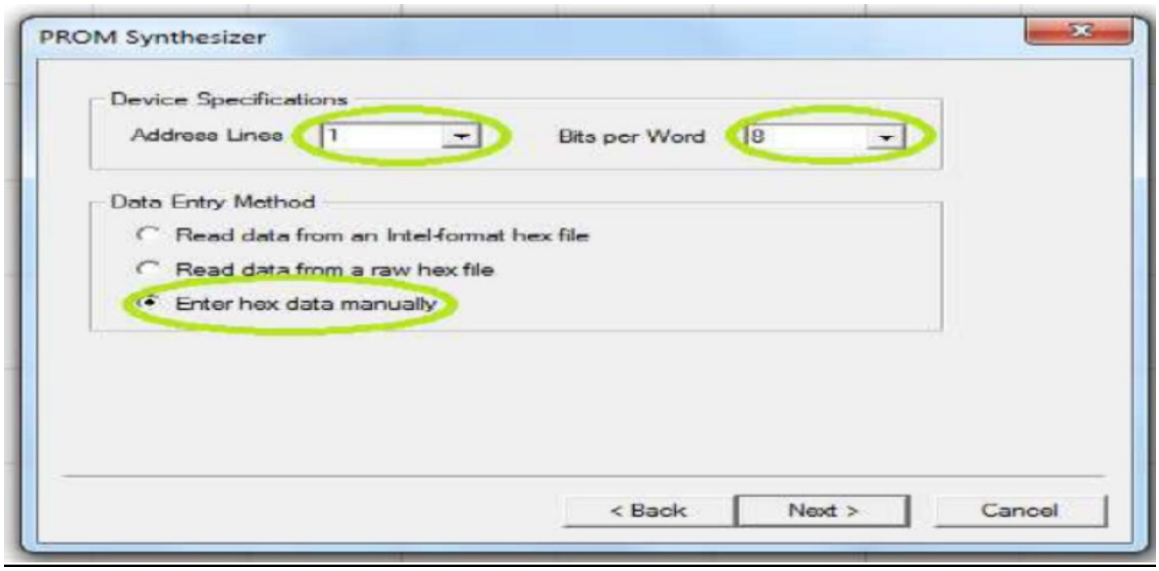
81 F3 49 61

33 25 05 F1

01 21 10 00

8C 80 0C 1C

Step 3B: This step is to create PROM from Block 4



Select the following for Block 4:

Address Lines: 1 Bits per Word: 8 Data Entry Method: Enter hex data manually.

Hex data for block 3 as follows:

81 F3

Note:

Please refer the textbook 'Digital Design' by Morris Mano for basic gates that were taught in Lab1.

Deliverables:

1. Submit your LogicWorks (*.cct) circuit file.
2. Submit a report that shows for each assignment:
    a. Compact truth table that relates the inputs and outputs (for the parts in which you weren't provided with one).
    b. Why and how you used the gates/blocks that you used
    c. The verification patterns (applied from the inputs and expected values at the outputs) that you used to validate your design along with simulation screen shots.
    d. Problems you faced in designing the block and your approach of solving them.

Due Date: Nov 21st 2017, 10:00 am