EEE 454 VLSI I Project Group No: 07

Project Title: VLSI Design Project: Configurable Logic Block (CLB)

Prepared By:

Student Name: Shakil Ahmed
Student ID: 1606140

Name of Group Member: Arnob Roy Shaon
Ishraq Tashdid
Student ID of Group Member: 1506093
1606142

Date of Submission: 18.06.2021

INTRODUCTION

"Configurable Logic Block(CLB)" needs to be implemented in this project, more presisely 3:1 CLB. Three components of the CLB is listed below:

1. SRAM : 8 SRAM cells need to be implemented since this a 3: 1 CLB. These are memory cells. This memory array will store the input values of the mux of the LUT i.e output of our desied logic function.
2. LUT : It will perform 3 input NAND operation. A look up table is like a mux which will perform some specified logical operaton to connect desired value stored in SRAM to the output.
3. D Flip-flop: Flip-flop will act like a memory element of the CLB block.
It will enable the CLB to perform sequencing operation.
4. Mux : Mux will basically show outputs based on the input select pins. In this design 2:1 mux will be implemented.

"Configurable Logic Block(CLB)" design required following input & output pins :

1. LUT Input Pins : (2:0) Three select pins will be available as select pins of the mux placed inside the LUT block
2. Clk : This pin will be connected to the D flip-flop.
3. Load : This pin will control the write and store operation of the SRAM.
    Load = 1 data will be written to the SRAM cells.
    Load = 0 data will be stored(previously written data) in the SRAM cells.
4. Data **:** (7:0) Serial input of bit stream to load into the SRAM.
5. Output Pin: There will be a single output pin which will be the output port of the LUT block.

PROJECT DETAILS

Logic Family	Standard CMOS
Logic Function	NAND(A,B,C)

Design of SRAM Cell:

Static random-access memory (static RAM or SRAM) is a type of random-access memory (RAM) that uses latching circuitry (flip-flop) to store each bit. SRAM is volatile memory; data is lost when power is removed. [4]

Here we will adopt the standard 6 transistor (6T) design of SRAM cell.

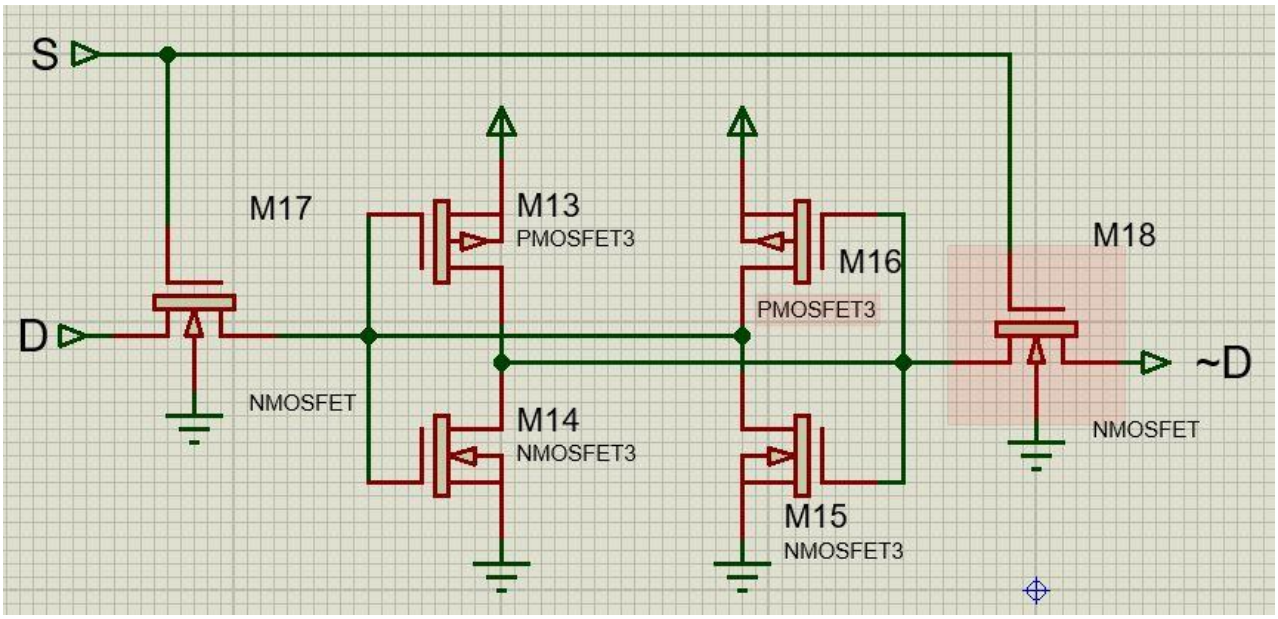

Fig 1: 6T SRAM Cell [4]

A typical SRAM cell is made up of six MOSFETs. Each bit in an SRAM is stored on four transistors (M1, M2, M3, M4) that form two cross-coupled inverters. This storage cell has two stable states which are used to denote 0 and 1. Two additional access transistors serve to control the access to a storage cell during read and write operations. For our design, S = load pin.

S = 1 ; the write cycle begins by applying the value to be written to the bit lines S = 0 ; the read cycle is started.

~D = bit lines ; through this line expected bit will be sent to the SRAM cell.

Schematic of D Flip-flop Using CMOS:

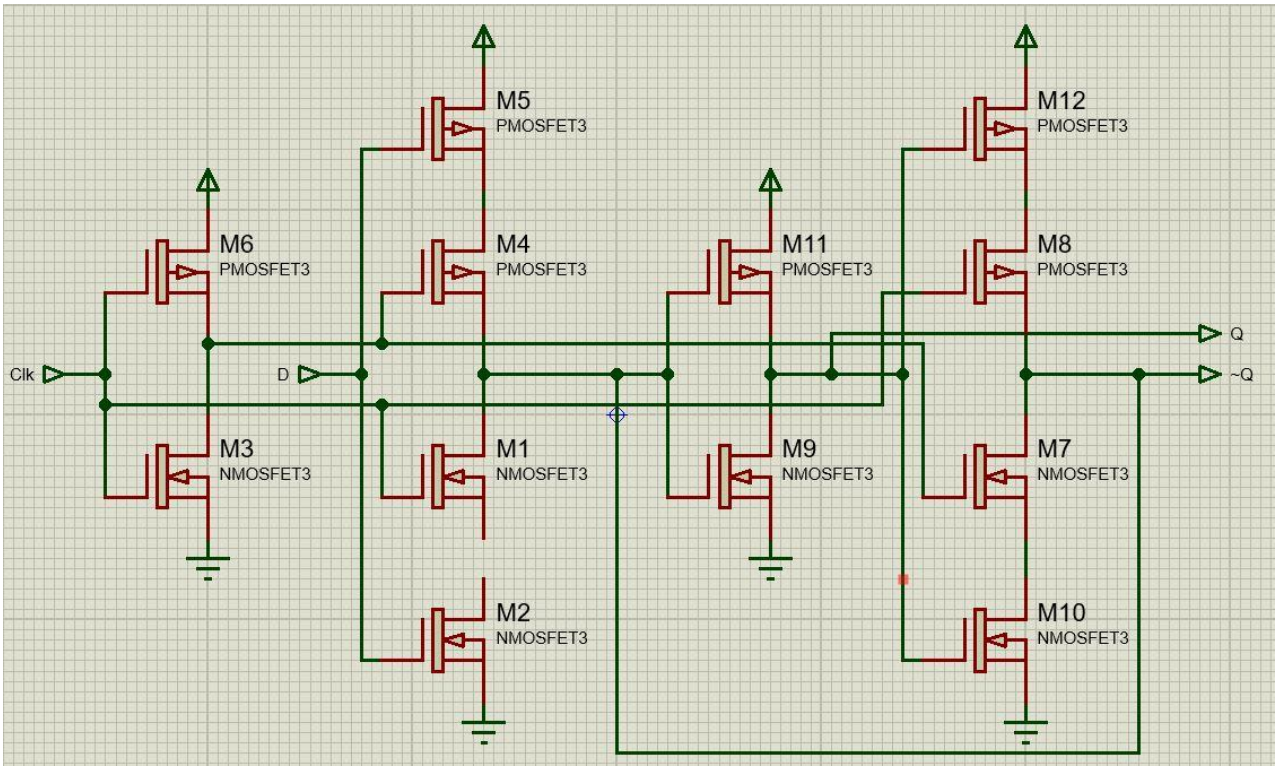

Fig 2: D flip-flop implemented using pass transistor

Here D = data input pin of the flip-flop
Clk :clock pin of the flip-flop.
Q = ouput pin of the flip-flop
~Q = inverted output of the flip-flop

Schematic of Mux Using CMOS:

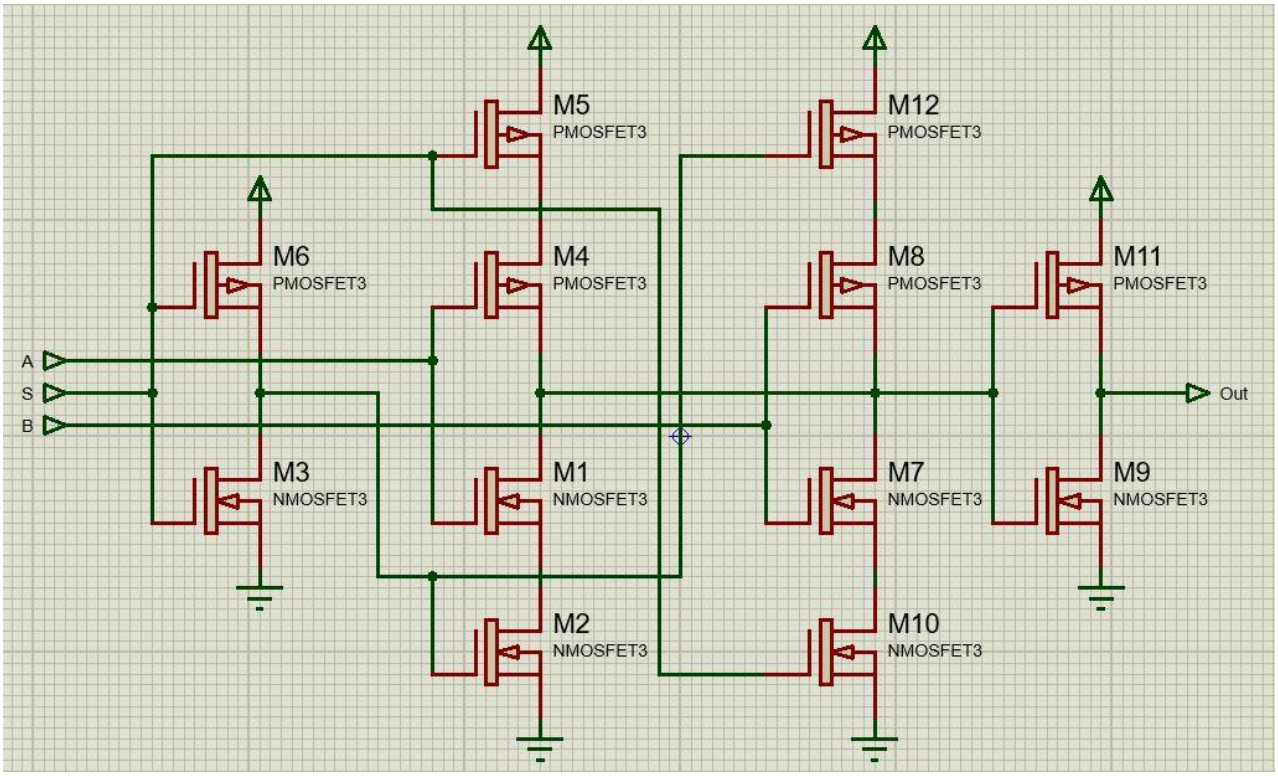

Fig 3: 2:1 mux using pass transistor.

Here S = Select pin of the mux

A & B = These are the input pins of the mux which will connect with Out based on the values of select pin. In our design A & B will come from two different SRAM cells.

Schematic of LUT using the CMOS mux:

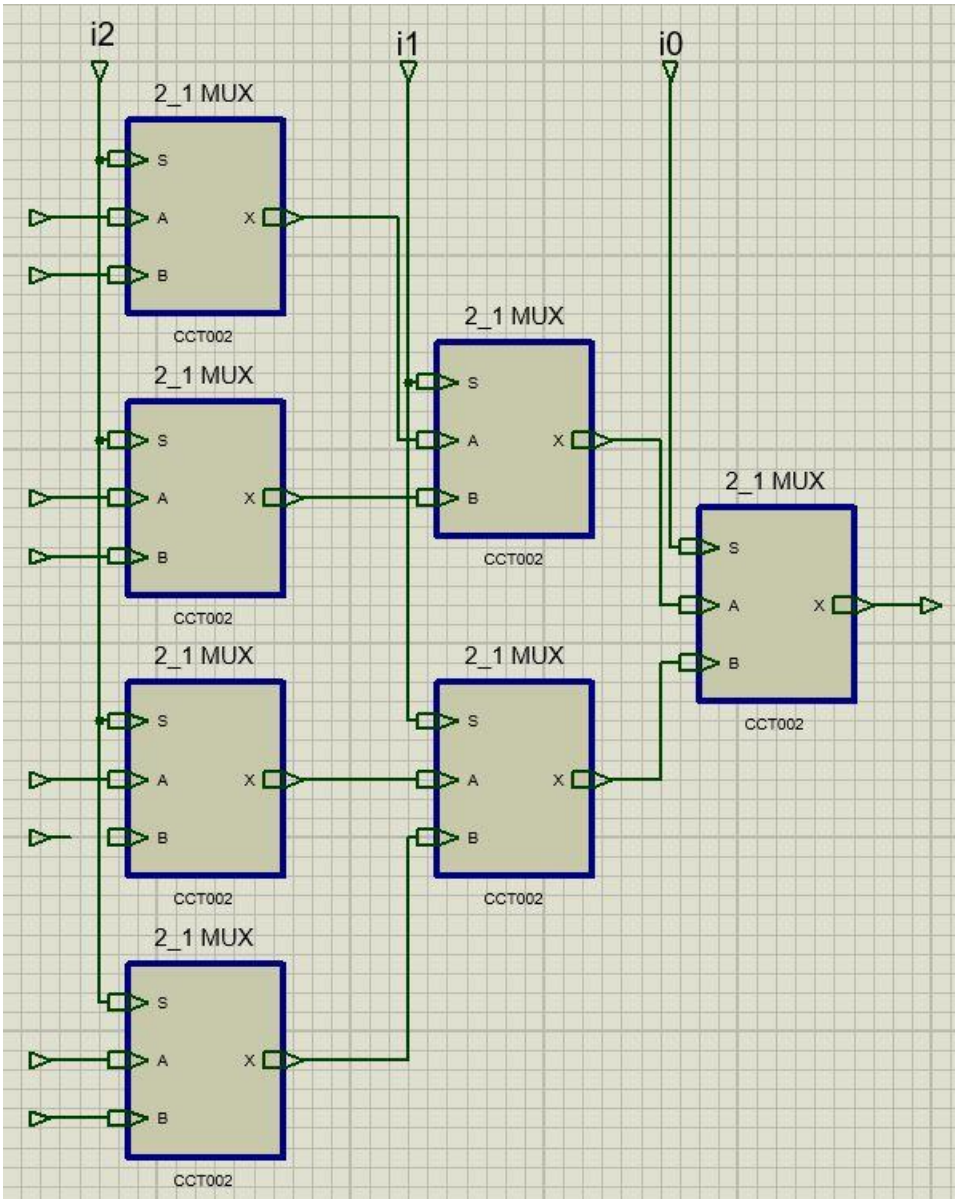

Fig 4 : 3:1 Look-up table

Here we will design the 2:1 mux using the pass transistor logic and the corresponding schematic is attached above. This will be basically an 8:1 mux which will be implemented with five 2:1 mux.

WORK DIVISION

Members Name	Assigned Work
Shakil Ahmed(1606140)	1, 2, 3
Ishraq Tashdid(1606142)	4, 5, 6
Arnob Roy Shaon(1506093)	7, 8

REFERENCES

[1] "D Flip Flop PLC Ladder Logic," [Online]. Available: https://instrumentationtools.com/plc-programto-implement-d-flip-flop-in-plc/.
[2] "Static random-access memory," [Online]. Available: https://en.wikipedia.org/wiki/Static_randomaccess_memory#/media/File:SRAM_Cell_(6_Transistors).svg.
[3] "Shakshat Virtual Lab," [Online]. Available: https://vlsi-iitg.vlabs.ac.in/Multiplexer_theory.html.
[4] "FPGAKey," [Online]. Available: https://www.fpgakey.com/wiki/details/51.