American International University- Bangladesh
Faculty of Engineering (EEE)
Digital Electronics Laboratory

Title: Implementation of a SRAM cell

Introduction:

A basic 6 transistor based SRAM will be implemented in this lab class. It is used to store logic '0' or '1'. The steps of Reading and Writing a '1' or '0' into a SRAM cell are described here in the lab sheet. Main purpose of this lab experiment is to implement the SRAM cell and study its operation.

Theory and Methodology:

Static Ram or SRAM is the most basic and easy to use type of volatile memory and is found in almost every computer in one form or another. An SRAM device is conceptually easy to understand, consisting of an array of latches along with control and decode logic to resolve the address that is being read or written at any given time. Each latch is a feedback circuit that traps and maintains a particular logic state. A typical SRAM bit implementation is shown in fig.1.

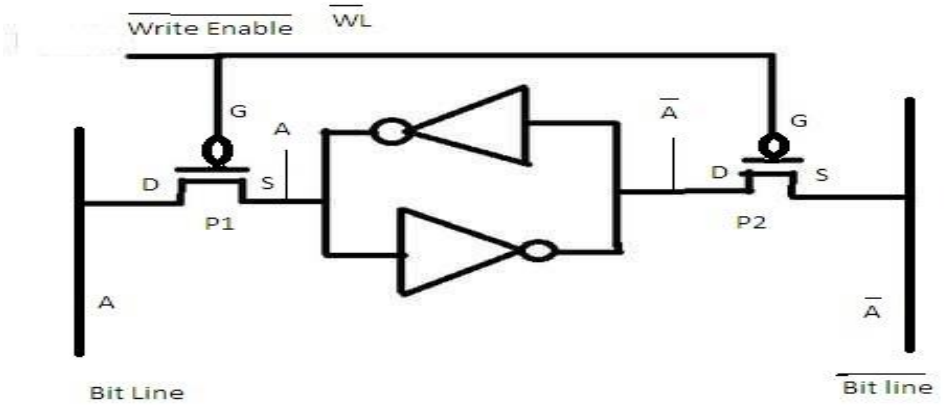

Fig.1: SRAM cell with inverter latch.

An SRAM latch is created by connecting two inverters in a loop. One side of the loop remains stable at the desired logic state, and the other side remains stable at the opposite state. Inverters are used rather than non-inverting buffers, because an inverter is the simplest logic element to construct. The two pass transistors on either side of the latch enable both writing and reading. When writing, the transistors turn on and force each half of the loop to whatever state is driven on the vertical bit lines. When reading, the transistors also turn on, but the bit lines are sensed rather than driven. Typical SRAM implementation requires six transistors for each inverter and the two pass transistors. Fig. 2 shows a six transistor SRAM cell.

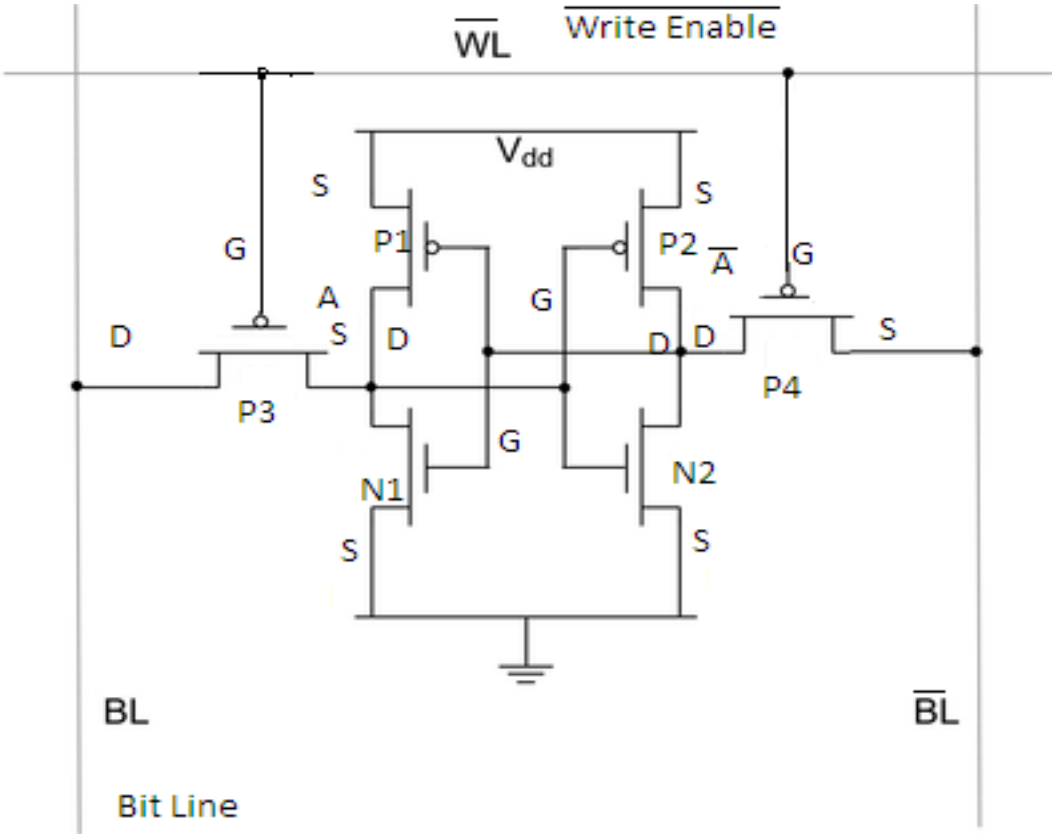

Fig.2: 6-T SRAM cell

An SRAM cell has three different states. It can be in: *standby* (the circuit is idle), *reading* (the data has been requested) and *writing* (updating the contents). The SRAM to operate in read mode and write mode should have "readability" and "write stability" respectively. The three different states work as follows:

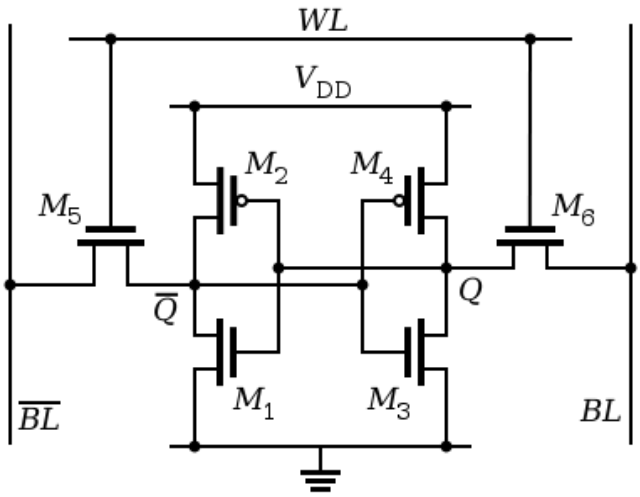

Fig.3: SRAM Operation

Stand by:

If the word line is not asserted, the *access* transistors M_5 and M_6 disconnect the cell from the bit lines. The two cross-coupled inverters formed by $M_1 – M_4$ will continue to reinforce each other as long as they are connected to the supply.

Reading:

Assume that the content of the memory is a **1**, stored at Q. The read cycle is started by precharging both the bit lines to a logical **1**, then asserting the word line WL, enabling both the *access* transistors. The second step occurs when the values stored in Q and Q are transferred to the bit lines by leaving BL at its precharged value and discharging BL through M_1 and M_5 to a logical **0** (i. e. eventually discharging through the transistor M_1 as it is turned on because the Q is logically set to **1**). On the BL side, the transistors M_4 and M_6 pull the bit line toward V_{DD}, a logical **1** (i. e. eventually being charged by the transistor M_4 as it is turned on because Q is logically set to **0**). If the content of the memory was a **0**, the opposite would happen and BL would be pulled toward **1** and BL toward **0**. Then these BL and BL will have a small difference of delta between them and then these lines reach a sense amplifier, which will sense which line has higher voltage and thus will tell whether there was **1** stored or **0**. The higher the sensitivity of sense amplifier, the faster the speed of read operation.

Writing:

The start of a write cycle begins by applying the value to be written to the bit lines. If we wish to write a **0**, we would apply a **0** to the bit lines, i.e. setting BL to **1** and BL to **0**. This is similar to applying a reset pulse to an SR-latch, which causes the flip flop to change state. A **1** is written by inverting the values of the bit lines. WL is then asserted and the value that is to be stored is latched in. Note that the reason this works is that the bit line input-drivers are designed to be much stronger than the relatively weak transistors in the cell itself, so that they can easily override the previous state of the cross-coupled inverters. Careful sizing of the transistors in an SRAM cell is needed to ensure proper operation.

Pre-Lab Homework:

1. Determine the operation of a cross-coupled inverter pair, where an inverter A is driven by another inverter B, while the inverter B also is driven by inverter A.

Students must install PSpice/LTSpice/ Psim software and MUST present the simulation results using transistors to the instructor before the start of the experiment.

Apparatus:

1. PMOS
2. NMOS
3. Inverter (7404)
4. Connecting Wires
5. Trainer board

Precautions:

Have your instructor check all your connections after you are done setting up the circuit and make sure that you apply only enough voltage (within V_{DD}) to turn on the transistors and/or chip, otherwise it may get damaged.

Experimental Procedure:

In the lab, reading and writing of a basic SRAM cell will be observed.
For writing and reading a '1' in the SRAM cell:

1. Connect the circuit as Fig.1 then move to Fig.2 or Fig. 3
2. Give a high value ('1') in the access or pass transistors. As they are PMOS they will remain 'Off'. So, writing is disabled.
3. Now give a high value ('1') in the bit line (A) and low value ('0') in the bit line bar ($\bar{A}$). Normally A will be inverted to $\bar{A}$.
4. Now enable the access transistors by giving a low value ('0'). Then '1' will be written in the cell.
5. Now again disable the access transistors by giving a high value ('1'). And remove the value of the bit lines, A and $\bar{A}$.
6. If you read the value from the latch you will see '1' is saved in the SRAM cell.

For writing and reading a '0' in the SRAM cell:

1. Connect the circuit as Fig.1 or Fig.2.
2. Give a high value ('1') in the access or pass transistors. As they are PMOS they will remain 'Off'. So, writing is disabled.
3. Now give a low value ('0') in the bit line (A) and high value ('1') in the bit line bar ($\bar{A}$). Normally A will be inverted to $\bar{A}$.
4. Now enable the access transistors by giving a low value ('0'). Then '0' will be written in the cell.
5. Now again disable the access transistors by giving a high value ('1'). And remove the value of the bit lines, A and $\bar{A}$.
6. If you read the value from the latch you will see '0' is saved in the SRAM cell.

Results and Discussion:

Students will summarize the experiment and discuss it as a whole. Interpret the data/findings and determine the extent to which the experiment was successful in complying with the goal that was initially set. Discuss any mistake you might have made while conducting the investigation and describe ways the study could have been improved.

Report:

Document the data acquired from the hardware, from simulation (if possible) as well as the expected values for both read and write operation.

Reference:

1. Thomas L. Floyd, *Digital Fundamentals*, 9th Edition, 2006, Prentice Hall.