May 4, 2018 **EE705:VLSI Design Lab** ***Ranjith KT***
Friday **Project:Individual Contribution** ***Roll No:17307R020***

In this Project we are creating a SRAM Controller which requires the basic understanding how a RAM memory is implemented and what are the Architectures it uses. For this along with Adarsh, we have done some literature review on Basic SRAM and DRAM architectures. With this research we got an idea about when pre-charge has to be done, how should be the structure of the controller block. Also we understood how important is refresh in RAM and how often we have to refresh a particular RAM in order to keep the data intact in the memory cells (For this RAM we have to refresh every row in each 64ms). With the DATA sheet we understood the timing sequences to be followed for all the processes to be done in the controller. Also 'Analysis and Design of Digital Integrated Circuits by David Hodges helped us to understand the sequence to be followed for creating read and write sequences. With these overall idea we developed the code for the SDRAM Controller architecture and I was assigned with development of Write Sequence.

In the Idle state, Write sequence can be started by giving the controller a write request. If the $wr\_req = 1$, the Bus will be activated and the next state will be $write\_act$ where the 13 bit ROW address(bits 21 to 9) a well as the two bit back address(MSBs) is passed.Once the RAS is kept high, according to the DATASHEET, there has to be two free cycles till before the pre-charge begins. One of this cycle is used for write CAS. So I we need an NOP in between to make two cylces of the 50MHz clock.($write\_nop1$) that accounts for that NOP cycle.In the next state ($write\_CAS$) the column address(9 LSBs) is passed(CAS =1) after which the pre-charge starts .This state is also followed by some NOPs (corresponding to the CAS Latency). At the end of the last NOP the instruction will be ready(pre-charge is done) which will be acknowledged by READY signal. We have to keep one more NOP to get enough length for the ready signal. After this we go back to the Idle state. This is the state flow of the Write sequence.