

Group Members- Apoorv Khattar (2016016), Kshitiz Jain (2016051), Nikhil Sachdeva (2016061)

## Project- **Project Topic 1: Arm Simulator**

Programming Language- **JAVA**

Project Plan-

Objective: Design and implement the function simulator in **JAVA** for a subset of ARM instructions.

Methodology: For the instructions we will be given a .MEM file. Using an instance of BufferedReader we can read each instruction line by line until we encounter 0xEF000011 which is exit instruction (swi 0x11). Each instruction is of the form **0x0 0xE3A0200A** where **0x0** represents the address of the instruction. The hex code **E3A0200A** (in this case it is the MOV instruction) tells us about type of instruction, it's source and destination registers. The problem can be broken down into the following steps-

### 1. **Reading instructions line by line from .MEM file:**

In JAVA, readLine() function from the BufferedReader class can be used to analyse each instruction one by one.

### 2. **Converting hex code into binary:**

The first three characters of the hex code will tell us about the address of the instruction and the remaining code can be converted to binary to find the type of instruction. The binary equivalent of the given instruction can be looked at as



### 3. **Finding op-code, destination and source registers form the instruction:**

The op-code field from bits 24-21 tells us exactly which type of instruction. To find the type of instruction we will need to maintain a list of op-codes for the instructions and then compare the values in this list with that of the given instruction. Bits 19-16 tell us about the first source register. Bits 15-12 tell us destination register and bits 11-0 tell us about the second operand which can either be a register (Bits 11-8) or it can be an immediate value. After decoding it we have store the value in the destination register as it might be used in some other instruction.

### 4. **Printing the output for each stage:**

For any instruction we have to output the work done by a given instruction across the five stages of the datapath namely- FETCH, DECODE, EXECUTE, MEMORY, WRITEBACK. For example for the given instruction- 0x0 0xE3A0200A, the output will be as follows-

Fetch instruction 0xE3A0200A from address 0x0

DECODE: Operation is MOV, First Operand is R0 and immediate second operand is 10.

Destination register is R2.

EXECUTE: MOV 10 in R2.

MEMORY: No memory operation.

WRITEBACK: Write 10 to R2.

After we will have to store the value in R2 in a variable as it might be required in some instruction later on.

# ARM SIMULATOR - TIMELINE

