EG 212 Computer Architecture – Processor design
Assignment – 1: IAS processor design

Guidelines:

- To be done individually or in groups of maximum 3 students (not more than 3)
- You can choose to do the design in any language you wish, C, C++, python, Verilog etc
- **What to submit on LMS:**
  - Submit your report (in pdf format) which contains roll numbers and names of the students in the group, some explanation of the code, the program you chose, the extra instructions you designed, snapshots of result etc. Upload codes separately.
  - When you submit the code, rename the filename of your code to <roll_numbers>_filename.< >
- The submission will be followed by a viva/demo
- All codes will run through a plagiarism check. Files found similar will get a 0 for the assignment. Repeat offence will attract Grade penalty on the overall grade
- Submit by Jan 26, 2024, 11:59pm on LMS under Assignment 1
- Marks: 30 code + 15 for Demo

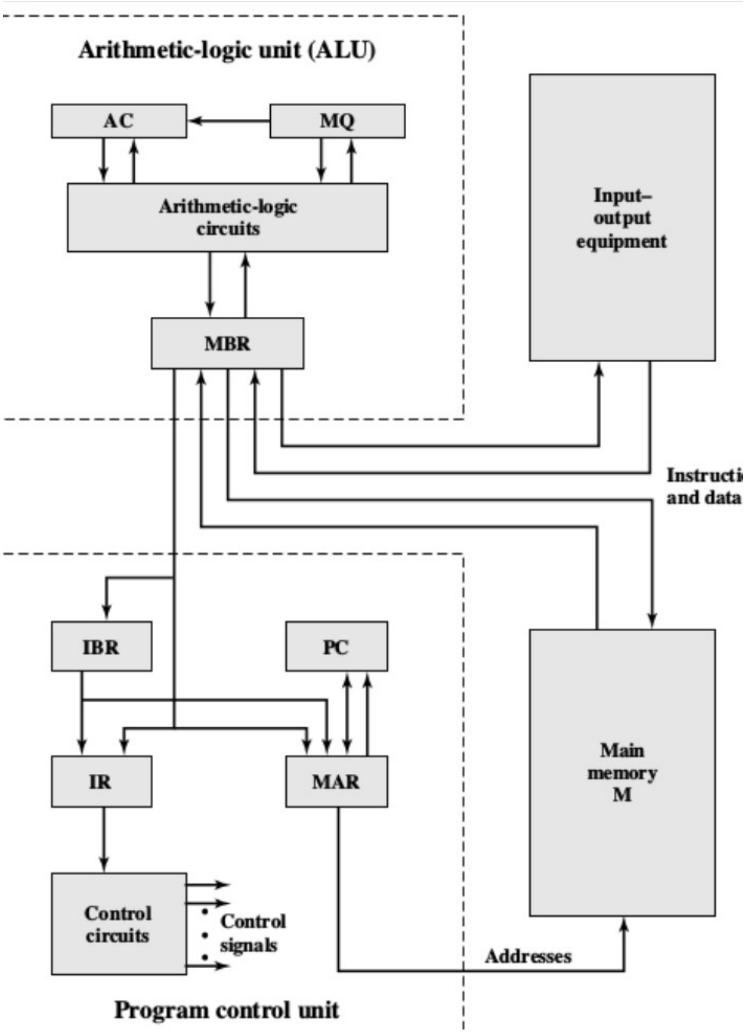
1. C and Assembly programming
   a. Indicate the C version of the program that you wish to implement. For example, matrix multiplication, array sorting, factorial calculation, simple encoding and decoding etc.
   b. Write an IAS assembly program for the C program (5 marks)
   c. The program should be chosen such that it covers at least 10 instructions in the IAS ISA, and it must definitely include Load, Store and Jump instructions after assembling.
   d. Introduce at least 2 new instructions of your own**,** in the ISA, assign it a new opcode, describe its operation. Use it in your assembly program.

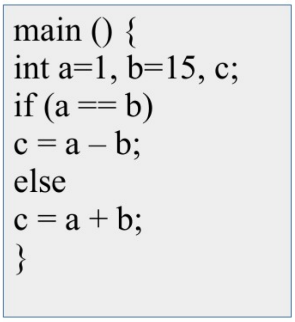
2. Assembler
   - Write an assembler (in any language) that can read in the IAS assembly program of question – 1. (10 marks)
   - The assembler should read in the assembly code and generate a machine code. Verify the machine code generated by your assembler with the IAS instruction set

   Note:
   - You are free to use any programming language
   - Instruction and Data memory can be declared as arrays of limited size (need not be 2^{40})

3. Design a single IAS processor with all the building blocks shown in the diagram (the arrows are only for illustrational purpose). For example AC, MQ etc can be declared as registers or int of a certain width. (15 marks)

- Implement the IAS computer in terms of fetch the machine instruction one after the other, decode and execute. Assume that Instruction fetch, decode and execute complete in one instruction cycle.
- Implement the assembled code you got from step2. That is, save the machine code in an array (presenting memory), fetch, decode and execute.
- You can test more than one program if you wish to do more rigorous testing of the design.





```
main () {
int a=1, b=15, c;
if (a == b)
c = a – b;
else
c = a + b;
}
```

Sample code

How will the processor code/simulation model look like:

- It should start with an instruction fetch, decode and execute of the program you store in memory/array.
- It should mimic the architecture of the IAS machine. For example, if you want to implement Load M(X), you should be able to show the following:
    - Fetch: MAR ← PC. Memory ←MAR, MBR ← M[MAR], AC ← MBR step by step
    - You should not directly code, AC ← M(X), although that is the final step
    - Each internal operation should be shown
    - You can use a control circuitry to generate MemRead, MemWrite signals.
    - Gate level design is not needed.
- Feel free to make your own assumptions regarding any concept no mentioned here.