Department of Electronic and Telecommunication Engineering

University of Moratuwa

Processor Design Project

Team Halo

Pramuditha A.A.H	-	200476P
Thilakarathne D.L.J	-	200650U
Wijetunga W.L.N.K	-	200733D

This report is submitted as a partial fulfillment of the module

EN2031 – Fundamentals of Computer Organization and Design

1st of March 2023

1. **Question (a)**

   a. **Part (i)**

      I. <u>ALU Instructions</u>

- ADD R_a, R_b
  - Meaning - Add the values in R_a and R_b registers and store the resulting value in the R_a register. R_a and R_b can be any register from R0 to R6.

- ADD_I R_a, y
  - Meaning - Add the value in R_a and y immediate value and store the resulting value in the R_a register. R_a can be any register from R0 to R6.

- SUB R_a, R_b
  - Meaning - Subtract the value in R_b from the value in R_a register and store the resulting value in the R_a register. R_a and R_b can be any register from R0 to R6.

- SUB_I R_a, y
  - Meaning - Subtract the value in R_a from the immediate value and store the resulting value in the R_a register. R_a can be any register from R0 to R6.

- B_AND R_a, R_b
  - Meaning - Do bitwise AND operation for the values in R_a and R_b registers and store the result in R_a register. R_a and R_b can be any register from R0 to R6.

- B_OR R_a, R_b
  - Meaning - Do bitwise OR operation for the values in R_a and R_b registers and store the result in R_a register. R_a and R_b can be any register from R0 to R6.

      II. <u>Stack Operations</u>

- POP R_a
  - Meaning - Get the required value from the stack and store it in R_a register. R_a can be any register from R0 to R6. The Stack Pointer (Here R7) should be decremented.

- PUSH R_a
  - Meaning - Store the value in R_a to the Top of the Stack (TOS).

      III. <u>Load and Store Instructions</u>

- LOAD R_a, R_b
  - Meaning - Load data from the memory location which is stored in R_b register to R_a register. R_a and R_b can be any register from R0 to R6.

- LOAD_I R_a, y
    - Meaning - Load the immediate value to R_a register. R_a can be any register from R0 to R6.

- LOAD_A R_a, R_b
    - Meaning - Load data from the memory location which is stored in R_b register to R_a register. The address register R_a is then incremented. R_a and R_b can be any register from R0 to R6.

- STORE R_a, R_b
    - Meaning - Store the data in the memory location specified in R_b register to R_a register. R_a and R_b can be any register from R0 to R6.

- MOV_R R_a, R
    - Meaning - Transfer data from R to R_a register. R_a and R can be any register from R0 to R6.

IV. <u>Control Flow Instructions</u>

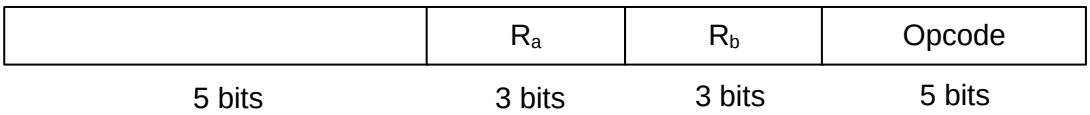
- JUMP y
    - Meaning - An unconditional branch to a memory location which is in the current program counter value by the immediate value y.

- IFEQ R_a, y
    - Meaning - Compare value of the register R_a with the value in the designated register R0. If the values are equal, branch the memory location which is in the current program counter value by the immediate value y. Ra can be any register from R0 to R6.

- IFLEQ R_a, y
    - Meaning - Compare value of the register R_a with the value in the designated register R0. If the value of register R_a is less than or equal the value in register R0, branch the memory location which is in the current program counter value by the immediate value y. R_a can be any register from R0 to R6.
- IFGEQ R_a, y
    - Meaning - Compare value of the register R_a with the value in the designated register R0. If the value of register R_a is greater than or equal the value in register R0, branch the memory location which is in the current program counter value by the immediate value y. R_a can be any register from R0 to R6.

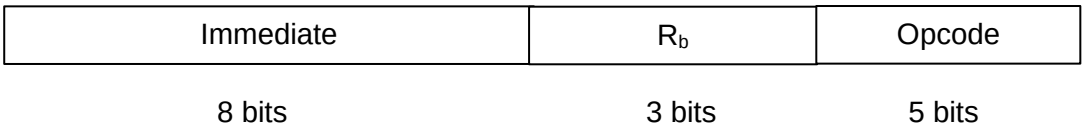
b. Part (ii) - Instruction Formats Required

- ALU Instructions



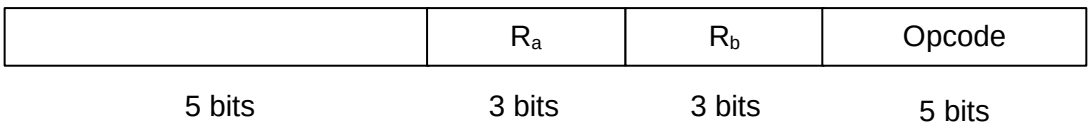
  - Since there are 17 instructions, at least 5 bits are required for the opcode.
  - There are 8 registers which leads to have 3 bits to represent them and, in this format, to represent source and destination registers, 3 bits each is required.
  - Therefore, the remaining 5 bits are extra bits here.

- Control Flow Instructions



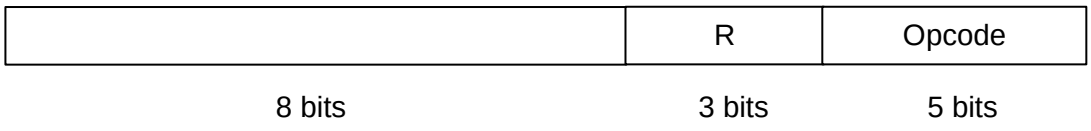
  - Since there are 17 instructions, at least 5 bits are required for the opcode.
  - 3 bits are required to represent the operand and to represent the immediate, 8 bits are required.

- Load and Store Instructions



  - Since there are 17 instructions, at least 5 bits are required for the opcode.
  - There are 8 registers which leads to have 3 bits to represent them and, in this format, to represent source and destination registers, 3 bits each is required.
  - Therefore, the remaining 5 bits are extra bits here.

- Stack Operations

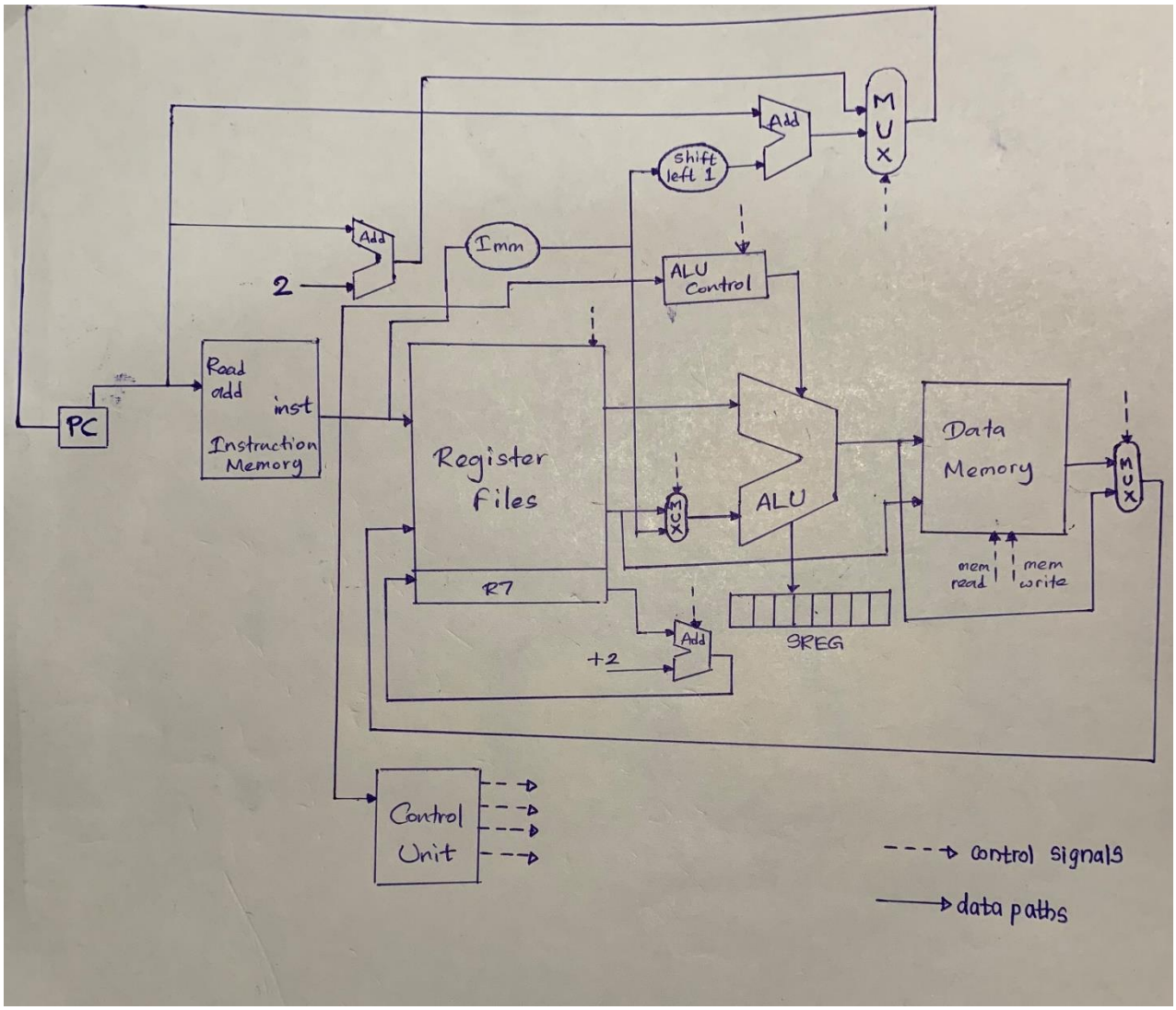


- Since there are 17 instructions, at least 5 bits are required for the opcode.
- There are 8 registers which leads to have 3 bits to represent them and here 3 bits are needed to represent the operand.
- Therefore, the remaining 8 bits are extra bits here.

2. Question (b)

- First the type of instruction should be identified. Therefore, to do that, 2 bits will be used because there are only 4 types of instructions available in the processor.
- Inside a specific instruction type, 3 bits will be used to represent set of instructions that belong to the specific instruction type.
- The reason to use 3 bits is that under ALU instructions, there are 6 instructions which has the maximum number of instructions for a type. To represent them, at least 3 bits will be required.

3. Question (c)



4. Question (d)

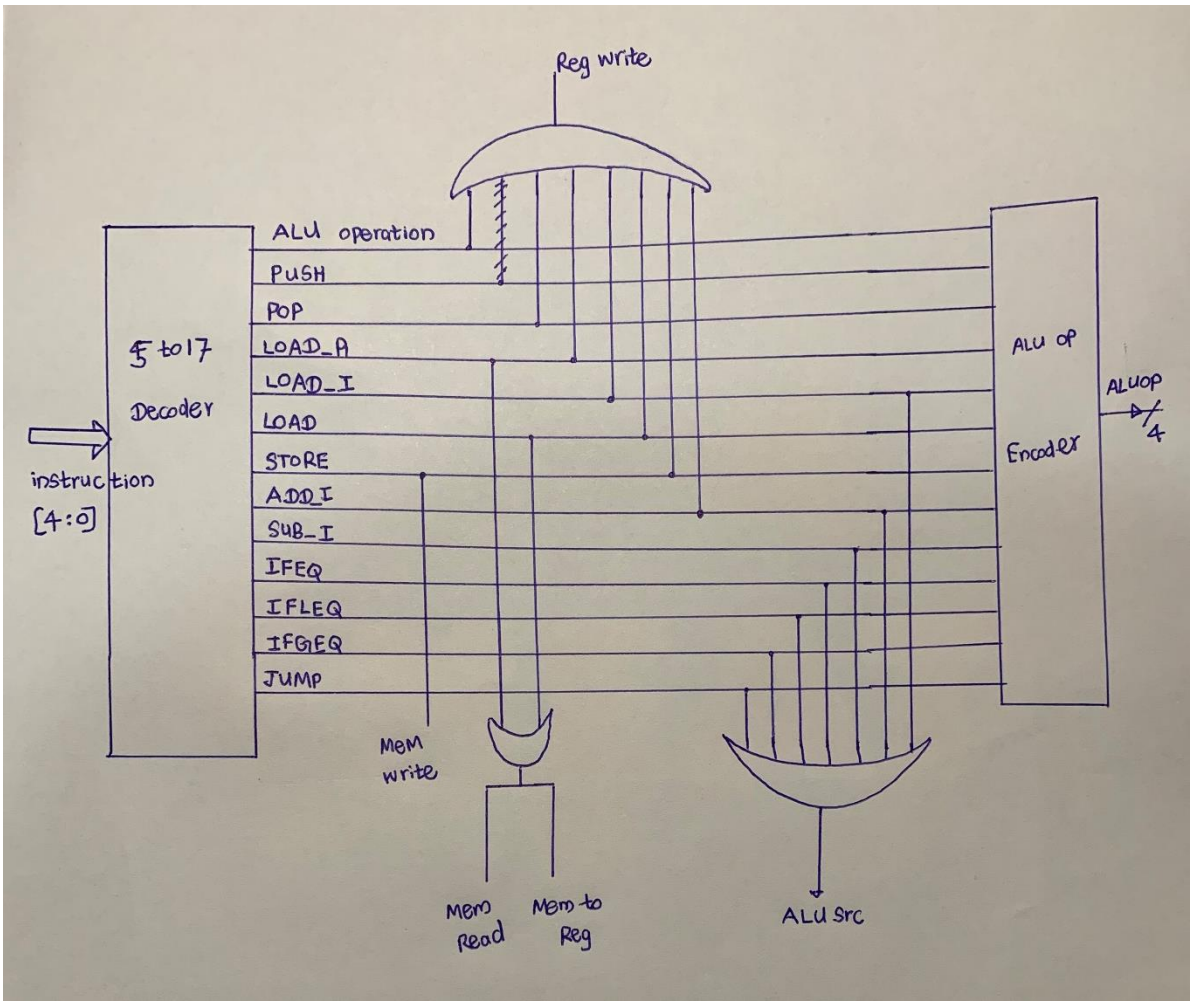
a. Part (i) - Design Approach for the Microarchitecture

Hardwired

Reason -

The processor is a part of System on Chip (SoC) to be placed in an embedded system and it is not expected to run an operating system. Additionally, a custom compiler will convert C language program to assembly which will be loaded to program memory. Therefore, this processor is an application specific processor. So, new instructions will not be added in the future. Therefore, the best design approach for the microstructure will be hardwired approach.

b. Part (ii)



The End.