

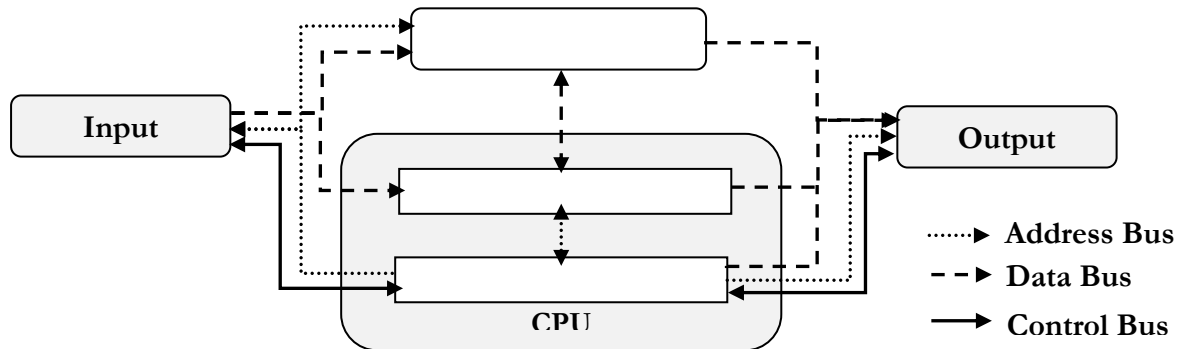
Exercise Series N°02 : Basic Mechanisms of Operating System

Exercise II.1: (Understanding Questions)

- Q1) What is a process?
- Q2) What is a process control block (PCB)?
- Q3) Is a PCB found all operating systems?
- Q4) Is PCB in all operating systems the same?
- Q5) What is an interrupt? When does it occur?
- Q6) What are the main tasks involved in operating system interruption?
- Q7) What is a masked interrupt?

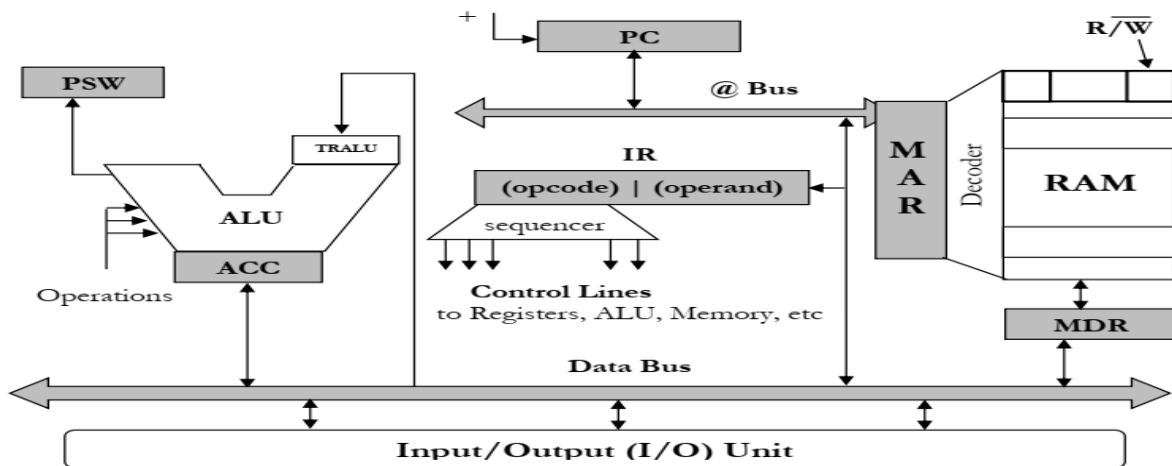
Exercise II.2: (Instruction Cycle)

- Q1) The diagram shows the Von Neumann Architecture. Fill into the boxes the missing components.



- Q2) Explain briefly how the Von Neumann Cycle works (explain the six steps).

Consider the simplified diagram of a computer based on Von Neumann's model shown below:



- Q3) Write the formal microprogram corresponding to the execution of the following Assembler instructions:
- a) CHG 10 ; // Load ACC with value 10
 - b) ADD 10; // Make $ACC \leftarrow ACC + 10$
 - c) SUB @ ; // Make $ACC \leftarrow ACC - [@]$
 - d) JMP @ ; // Jumps to @ address
 - e) MOV AX, [@] ; // $AX \leftarrow [@]$
 - f) ADD AX, [0002] ; // $AX \leftarrow AX + [0002]$

Exercise II.3: (Linker)

It essentially acts as a link editor (linker), calculating the implementation of modules and the values of external definition. A set of modules defined as follows is available:

PROGRAM Module	Size : 332 External Reference :	OPEN READ CLOSE EDIT
LABEL Module	Start address : 133 Size : 128 External Definition :	NAME 0 COMPANY 32 ADDRESS 64 POSTCODE 96 CITY 101
READING Module	Size : 840 External Definition : External Reference :	OPEN 0 READ 340 CLOSE 732 NAME COMPANY ADDRESS POSTCODE CITY
PRINTING Module	Size : 212 External Definition :	PRINT 0
EDITING Module	Size : 642 External Definition : External Reference :	EDIT 0 NAME COMPANY ADDRESS POSTCODE CITY PRINT

- Q1)** We link the modules PROGRAM, LABEL, READ, PRINT, and EDIT using the linker. Briefly justify your answer:
- Module implementation addresses
 - Total size of resulting program
 - Link table after first pass
 - Resulting program start address
- Q2)** We want to put the LABEL, READING, PRINTING and EDITING modules in a library named USE. In what order must they be put, to obtain the correct linker, by providing the PROGRAM module and the USR library to the linker, knowing that the latter sequentially browses the library, to know whether or not to incorporate the successive modules it encounters.
- Explains how this works.**

Exercise II.4: (Interrupt)

Time counting on the PC is performed via an **IRQ 0** hardware interrupt on the **8259 circuit** (timer). **IRQ 0** is triggered **18 times per second**.

- Q1)** Write an algorithm for an **IRQ 0** interrupt routine that displays the current time as: **HH:MM:SS** (Hour, Minute, Second).
- Q2)** What happens if your interrupt routine takes longer than **0.06 seconds**?
- Q3)** What do you suggest in this case?