

Faculty of Engineering and Technology

Electrical and Computer Engineering Department

Computer Organization and Assembly Language

**Assignment No.1 & Assignment No.2**

Chapter 1 & Chapter 2

**Prepared by:**

|  |  |  |  |
| --- | --- | --- | --- |
| Student Name: | Yuna Nawahda | Student ID: | 1211524 |
| Student Name: | Nour Alaydi | Student ID: | 1222810 |

|  |
| --- |
| 13/12/2023 |

**Date:**

**Assignment No.1 - Chapter 1 (***10th edition***)**

**Review Questions:**

**Question#1 (Q1.3 P42)**

What are the four main functions of a computer?

**Answer#1 (*Yuna Nawahda & Nour Alaydi*):**

**Data processing**:

Data may take a wide variety of forms and the range of processing requirements is broad.

**Data storage:**

Short-term, Long-term

**Data movement:**

Input-output (I/O) - when data are received from or delivered to a device (peripheral) that is directly connected to the computer.

Data communications – when data is moved over longer distances, to or from a remote device.

**Control:**

A control unit manages the computer’s resources and orchestrates the performance of its functional parts in response to instruction.

**Question#2 (Q1.5 P42)**

**List** and **briefly** define the main structural components of a processor.

**Answer#2 (*Yuna Nawahda & Nour Alaydi*):**

**Control Unit:**

It Controls the operations in cpu fetches, decodes, and executes instructions.

**Arithmetic & Logic Unit (ALU):**

The ALU is responsible for performing arithmetic and logical operations such as addition, subtraction, multiplication, division, and comparisons.

**Registers:**

Its fast, on-chip memory inside the CPU, dedicated or general purpose.

**Interconnection:**

It provides for communication among the control unit, ALU as well as registers.

**RAM:**

It provides storage for data and instructions that are currently being used by the Cpu.

**Internal Clock:**

It derived directly or indirectly from system clock.

**Internal Buses:**

it connects the components.

**Logic Gates:**

It is controlling the flow of the instruction information.

**Execution Pipeline:**

The execution pipeline breaks down the instruction execution process into stages to improve overall processing speed.

**Chapter 1 (***10th edition***):**

**Problems:**

**Problem#1 (P1.2 P43)**

1. On the IAS (***Table 1.1 Page 16***), what would the machine code instruction look like to load the contents of memory address 2 to the accumulator?
2. How many trips to memory does the CPU need to make to complete this instruction during the instruction cycle?

**Answer#1 (*Yuna Nawahda & Nour Alaydi*):**

1. Opcode:00000001 - Operand 000000000010.

b.

CPU needs to make **2** separate trips to memory to complete the

instruction:

**Fetch the instruction:** This involves fetching the machine code instruction itself from memory. This is necessary to determine what operation the CPU needs to perform.

**Fetch the data:** CPU needs to fetch the actual data from the specified memory address. This data is then loaded into the accumulator.

**Problem#2 (P1.3 P43)**

On the IAS, describe the process that the CPU must undertake to read a value from memory and to write a value to memory in terms of what is put into the MAR, MBR, address bus, data bus, and control bus.

**Answer#2 (*Yuna Nawahda & Nour Alaydi*):**

**Reading from memory:**

to read a value from memory, the CPU puts the address into the MAR.

Then, it puts the address on the address bus to send it to the memory unit.

After that, the memory unit copies the contents of that memory location passed on the data bus. Finally, the data is transferred to the MBR.

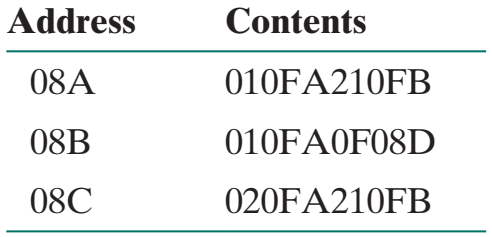
**Writing to memory:**

To write a value to memory, the CPU puts the address into the MBR.

Then, it puts the data into the MAR,sending it by the address bus to the memeory unit. Finally ,it sends a write signal by control bus into the appropriate memory location.

**Problem#3 (P1.7 P43)**

Given the memory contents of the IAS computer shown below,



Show the assembly language **code** for the program, starting at address 08A. Explain what this program **does** (hint: use ***Table 1.1 Page 16***).

**Answer#3 (*Yuna Nawahda & Nour Alaydi*):**

The purpose of this program is to store the absolute value of content at memory location 0FA into memory location 0FB.Transfer M(0FA) to the accumulator, then Transfer contents of accumulator to memory location 0FB, If number in the accumulator is nonnegative, take next instruction from left half of M(08D)

**08A** 00000001 🡪 LOAD M(0FA)

21🡪 00100001 🡪 STOR M(0FB)

**08B** 00000001 🡪 LOAD M(0FA)

0F🡪 F🡪15 00001111 🡪 JUMP +M (08D)

**08C** 02 🡪 00000010 🡪LOAD -M (0FA)

21🡪 00100001 🡪 STOR M(0FB)

**Assignment No.2 - Chapter 2 (***10th edition***):**

**Review Questions:**

**Question#1 (Q2.4 P75)**

Briefly characterize Amdahl’s law.

**Answer#1 (*Yuna Nawahda & Nour Alaydi*):**

Deals with the potential speedup of a program using multiple processors compared to a single processor.

It is basically used in Parallel computing scenarios where a part of a program efficiency is improved using multiple processors.

the performance improvement that can be gained through parallel processing is limited by the part of a system that's inherently sequential that is the set of operations that must be run in series.

**Question#2 (Q2.7 P75)**

List and define three methods for calculating a mean value of a set of data values.

**Answer#2 (*Yuna Nawahda & Nour Alaydi*):**

* Arithmetic mean AM = (sum of data values) / (total number of data values)
* Geometric mean GM = (product of data values) ^ (1/n)
* Harmonic mean HM = (total number of data values) / (sum of reciprocals of data values)
* **Arithmetic** works well when the data is in an additive relationship between the numbers, often when the data is in a ‘linear’ relationship which when graphed the numbers either fall on or around a straight line
* **Geometric** mean works well when the data is in an **multiplicative** relationship or in cases where the data is compounded; hence you multiply the numbers rather than add all the numbers to rescale the product back to the range of the dataset
* **Harmonic** mean is used when we want to **average** units such as speed, rates and ratios

**Chapter 2 (***10th edition***):**

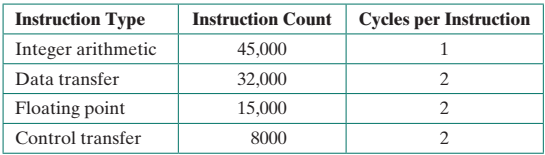
**Problems:**

**Problem#1 (P2.1 P75)**

A benchmark program is run on a 40 MHz processor. The executed program consists

of 100,000 instruction executions, with the following instruction mix and clock cycle

count:



Determine the effective CPI, MIPS rate, and execution time for this program.

**Answer#1 (*Yuna Nawahda & Nour Alaydi*):**

**CPI**= sum(IC\*CPI/ sum of IC)

=( (45000\*1)+(32000\*2)+(15000\*2)+(8000\*2))/(100000)

=155000/100000

=**1.55**

Convert MHZ 🡪 HZ

40 MHZ= 40000000 HZ

**MIPS**= f/CPI \* 10^6

=40000000/(1.55\*1000000)

=40/1.55

=**25.08**

**Execution Time**= IC\*CPI/f

= (100000\*1.55)/(40 \* 1000000)

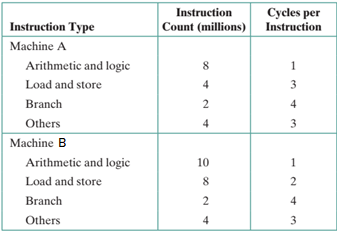
=1.55 / 400

=0.003875 sec

In MS🡪 **3.875 ms**

**Problem#2 (P2.2 P75)**

Consider two different machines, with two different instruction sets, both of which have a clock rate of 200 MHz. The following measurements are recorded on the two machines running a given set of benchmark programs:



Determine the effective CPI, MIPS rate, and execution time for each machine.

**Answer#2 (*Yuna Nawahda & Nour Alaydi*):**

200 MHZ 🡪 200\*(10^6)HZ

T= 1/F = 1/200=5\*10^-3

CPI= sum(IC\*CPI/ sum of IC)

**Machine a:**

**CPI**=(8\*1+4\*3+2\*4+4\*3)/8+4+2+4

=40/18

=**2.22**

**MIPS**=f/(CPI\*10^6)

=200\*10^6/2.22\*10^6

=**90.1**

**Execution Time** =IC\*CPI\*T

=18 \* 2.22\* 5\*10^-3 = 0.1998 = **0.2 sec**

Machine b:

**CPI**=(10\*1+8\*2+2\*4+4\*3)/10+8+2+4

=46/24

=**1.92**

**MIPS**=f/(CPI\*10^6)

=200\*10^6/1.92\*10^6

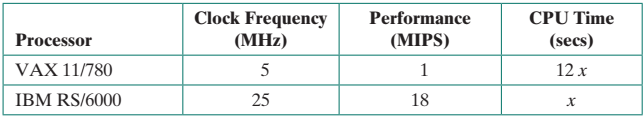
=**104.2**

**Execution Time** =IC\*CPI\*T

**=** 24 \* 1.92\* 5\*10^-3 = 0.1998 = **0.23 sec**

**Problem#3 (P2.3 P76)**

Early examples of CISC and RISC design are the VAX 11/780 and the IBM RS/6000, respectively. Using a typical benchmark program, the following machine characteristics result:



The final column shows that the VAX required 12 times longer than the IBM measured in CPU time.

1. What is the relative size of the instruction count of the machine code for this benchmark program running on the two machines?
2. What are the CPI values for the two machines?

**Answer#3 (*Yuna Nawahda & Nour Alaydi*):**

1. IC = T\* MIPS / 10^6

(x\*18) / (12x\*1) = 18x / 12 x

= 1.5

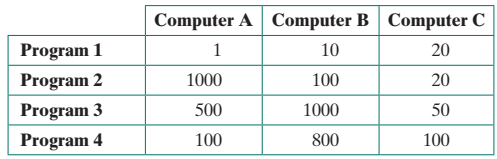
b. CPI = 5 MHz /1 MIPS= 5 for VAX

CPI = 25 MHZ / 18 MIPS

=1.4 for IBM

**Problem#4 (P2.4 P76)**

Four benchmark programs are executed on three computers with the following results:



The table shows the execution time in seconds, with 100,000,000 instructions executed in each of the four programs. Calculate the MIPS values for each computer for each program. Then calculate the arithmetic and harmonic means assuming equal weights for the four programs, and rank the computers based on arithmetic mean and harmonic mean.

**Answer#4 (*Yuna Nawahda & Nour Alaydi*):**

MIPS=IC/ T\*10^6=100000000/ T\*10^6 =100/T

so the MIPS will be🡪

Program 1:

A: 100/1=100 MIPS

B: 100/10=10 MIPS

C: 100/20=5 MIPS

Program 2:

A: 100/1000=0.1 MIPS

B: 100/100=1 MIPS

C: 100/20=5 MIPS

Program 3:

A: 100/500=0.2 MIPS

B: 100/1000=0.1 MIPS

C: 100/50=2 MIPS

Program 4:

A: 100/100=1 MIPS

B: 100/800=0.125 MIPS

C100/100=1 MIPS

The Arithmetic Mean(AM):

A: (100 + 0.1 + 0.2 + 1) / 4 = 25.325 MIPS

B: (10 + 1 + 0.1 + 0.125) / 4 = 2.80625 MIPS

C: (5 + 5 + 2 + 1) / 4 = 3.25 MIPS

The Harmonic Mean(HM):

A: 4 / ((1/100) + (1/ 0.1) + (1/ 0.2) + (1/ 1)) = 0.249 MIPS

B: 4 / ((1/10) + (1/ 1) + (1/ 0.1) + (1/ 0.125) = 0.21 MIPS

C: 4 / ((1/5) + (1/5) + (1/2) + (1/1)) =2.105 MIPS

Based on the arithmetic mean fastest to slowest

A > C > B.

Based on the harmonic mean fastest to slowest

C > B > A

**Problem#5 (P2.13 P79)**

Assume that a benchmark program executes in 480 seconds on a reference machine A. The same program executes on systems B, C, and D in 360, 540, and 210 seconds, respectively.

1. Show the speedup of each of the three systems under test relative to A.
2. Now show the relative speedup of the three systems. Comment on the three ways of comparing machines (execution time, speedup, relative speedup).

**Answer#5 (*Yuna Nawahda & Nour Alaydi*):**

**a.**

Speedup B relative to A (B):

S2B = Execution time on A / Execution time on B

= 480 sec / 360 sec

= 1.33

Speedup C relative to A (C):

S2C = Execution time on A / Execution time on C

= 480 sec / 540 sec

= 0.89

Speedup D relative to A (D):

S2D = Execution time on A / Execution time on D

= 480 sec/ 210 sec

= 2.29

**b.**

execution time fastest to slowest:

D>B>C

Speedup highest to lowest:

D>B>C

relative speedup highest to lowest:

D>B>C

☺ Good luck ☺

Ibrahim A. Nemer @ 2023