Q4
(C)

→ EFFECTIVE CPI :

$$CPI = (1 \times 46000) + (2 \times 33000) + 2(16000) + (2 \times 9000) / 100$$

$$= \frac{162000}{100}$$

$$CPI = 1620$$

→ MIPS RATE :

$$= 60MHz / 1620 \times 10^6$$

$$= 60 \times 10^6 Hz / 1620 \times 10^6$$

$$= 60Hz / 1620$$

MIPS rate = 0.037

→ EXECUTION TIME :

$$T = Ic / (MIPS \times 10^6)$$

$$T = 104000 / (0.037 \times 10^6)$$

$$= 2811 / 10^{-3}$$

$$T = 2.811 \text{ sec}.$$

Q4.
(D)

→(a) Since we have same instruction mix it means that additional instruction for each task could be calculated as following.

$$T = IC / (MIPS \times 10^6)$$

$$T_1 = 11 ms.$$

$$T_8 = \frac{2 \times 10^6 / 8 + 0.025}{152 \times 10^6} = T = 1.8 ms.$$

$$Speedup = 11 / 1.8 = (6.11)$$

2. contents divided into LH RH.

LH = 010FA
opcode = 01
Address = 0FA

RH = 0F08D
opcode = 0F
Address = 0F8D

LH = 08B should read LOAD M(0FA)
RH = 0F = 0000 1111 - Jump + M(X, 0:19) refers to
08D 010FA0F08D is
LOAD M(0FA)
JUMP + M(08D . 0:19)

64
(B)

1. In 08A address M(0FA) transfers to accumulator and content to memory location 0FB

2. In 08D address the M(0FA) transfers to the accumulator & takes instructions from left half of M(08D)

3. In 08C address the M(0FA) transfers to the accumulator and transfer contents of accumulator to 0FB.

Q3

(G)

→ Programming in Hardware : The program is in the form of Hardware and is termed as hardware program. Suppose we construct a general purpose config of arethmatic and logic unit. This hardware will perform various functions on data depending on control signal.

→ Programming in Software :

The program here is a software which is loaded into the hardware eg the hardware is coded and it starts to function.

programing in a sequence of code or instruction is called software programming

Q4

(A)

→ Sol to problem : Contents are divided into LH & RH

LH instr = 010FA RH Inst = 210FB

opcode = 01 opcode = 21

address = OFA address = OFB.

→ LH Instruction : 01 = 00000001 LOAD M(X)

The first shift should read 08A Load M(OFA)

→ RH Instruction : 21 = 00100001 = STOR M(X)

The second 5 bit 08A reads STOR M(OFB)

LOAD M(OFA)

STOR M(OFB)

Q3
(D)

→ Cortex A: The cortex A and A50 are application processors intented for mobile devices eg smart phones, ie book readers.
These processors run at higher clock and support MMU.

→ Cortex R: It is designed to support real time apps in which timing of events need to be controlled with rapid response to events. They run as a high clock and have very low response latency.

→ Cortex M: These are developed primirily for micro controlless domain, where the need of fast high management is coptled.

Q3
(F)

→ Disabled Interrupt: This simply means that the processor can and will ignore the interupt request signal. If an interupt occours during this time. It generally remains pending and will be checked by the processor after it has enabled interupt. When a program is executing and interupt occours, it is terminated immidiately.

→ Nested Interupt: It is a interupt of higher priority interupt to a lower priority interupt handler to be interrupted itself. A prog begins at I=0 and at I=0 interupt occours user info is placed on stack and printer continous execution.

Q3
(A)

→ Computer Architecture : It refers to those attributes of a system visible to a programmer or to those that have a direct impact on logical execution of a progsam. A term that is often used interchangeably with CA is ISA.

→ Computer Organization : It refers to operational unit & their interconnections. That realize the architectural specifications.

Q3
(B)

→ CISC : The current x86 represents the result of decades of design effort on CIS.
The x86 incorporates the design principals found on mainframe and supercomputer and serve as best example of SISC.

→ RISC : An alternative approach to processor design is RISC. This architecture is used in a variety of Lembeded system designs and is one of the most powerful design.

Q3
(C)

→ Microprocessors : A microprocessor chip includes a register chip an ALU and control unit or IPL.
As transistor density increases so thus the complexity of the architecture. It includes multiple core & sustainable amount of cache memory.

Q2

(E)

→ Instruction cycle state diagram:

The state in instruction cycle program are as follow.

→ IAC: Determines the address of next instruction to be executed.

→ Instruction fetch (IF): Instructions from its memory loc into the processor.

→ Instruction operation decoding: Analyze instructions to determine type of operation to be performed

→ Operand adress Calculation: If operation involves reference to an operand in memory or available via I/O.

→ Operand Fetch: fetch the operand from memory or read it in form of I/O.

→ Data operand: perform operation on instructions.

Q2

(F)

→ Classes of Interupt:

→ Program: It is generated by some condition that occours as a result of instruction execution.

→ Timer: This allows O.S to perform certain functions on regular basis.

→ I/O: It is generated by an I/O controller to signal normal completion of operation

→ Hardware failure: It is generated by power failure.

→ Data lines: The data lines provide a path for moving data among system modules

→ Control lines: The control lines are used to control the access to and the use of data and address lines.

Q2
(C)

→ Stored Program Computer: A fundemental design approach in IAS computer is known as stored program computer. This idea is attributed to mathematician John Von Neumann.

The first publication of this idea was in 1945 by Von Neumann for a new computer the EDVAC.

A main memo which stores both data and instructions

An arithmatic and logic unit ALU operating on binary data.

Q2
(D)

→ MOORE'S LAW:

The Moores law was proposed by the co-founder of Intel, Gorden Moore in 1965.

→ He observed that number of transistors to be put on a chip were doubling every year

→ Consequences:-

→ The cost of computer logic & memory circuitry fell at a dramatic rate.

→ The computer becomes smaller.

→ A reduction in power requirement.

→ The interconnection on integrated circuits are much more reliable than solder connections

Q2
(A)

→ Structural components of computer:
There are 4 main structural components of computer.

1. CPU: It controls the operation of the computer and perform its data processing function.
2. MAIN MEMO: It stores data.
3. I/O : It moves data b/w computer and external components.
4. SYSTEM INTERCONNECTION : Some mechanism that provide for communication among CPU main memory & I/O.

Q2
(B)

→ The charecteristics of computer family are as follow :-

1. Similar Instruction set : In some cases lower end of the family has an instruction set that is subset of higher end of fam. so program can move up not down.
2. Similar OS : The same basic OS is available for all family members.
3. Increasing Speed : The rate of instruction execution increases in going from lower to higher family members.
4. Incresed I/O ports : number of I/O ports increase in going from lower to higher family members.
5. Incredsing memory size : The size of main memory increases in going from lower to higher family members.

Q1.
(H)

→ MULTICORES:

The use of multiple processors on the same chip, also reffered to as multiple cores. provides to increase performance without increasing clock rate. If the software can support use of multiple processors then it also doubles performance. 2 core chips were simply followed by 4 core and so on.

→ MIC :-

The leaps in performance as well as the challenge in developing software to exploit such a large num of cores has led to a new term "Many integrated core" (MIC)
The MIC strategy involves a homogeneous collection of general purpose processors on a chip.

→ GPU:

A GPU is a core designed to perform parallel op's on graphic data. It is found on plugin graphic card. It is used to encode and render 2D and 3D videos.

Q1
(I)

→ QPI PROTOCOL LAYER:

→ In this layer packet is defined as unit of transfer.
→ One key function performed at this layer as a cache coherency protocol, which deals with making sure that the main memory values held in multiple cache are consistent.

Q 1
(F)

→

The problems created due to increase in clock speed & logic density are :-

1. POWER : As with increase in clock speed and logic density the dissipation of heat energy becomes difficult at high density, which lead chips to a serious design issue.

2. RC DELAY : Due to the increased resistance the flow of electrons on a chip b/w transistors is limited thus resulting a delay in RC products.

Q 1
(G)

→

Consider a program running on a single processor such that function 1-f of execution time involves code that is sequential and a function of that involves code that is infinitely parallelgible with no scheduling overhead.
Let T be total execution time. Then the speedup using a parallel processor that fully exploit the paralled portion as follow :-

$$\text{Speedup} = \frac{\text{Time to execute program on a single proces}}{\text{Time to execute on N parallel processors}}$$

$$\text{Speedup} = \frac{T(1-f) + T \cdot f}{T(1-f) + Tf/N} = \frac{1}{(1-f) + f/N}$$

Q1
(D)

→ Desktop applications that require great power of contemporary microprocessor based system are :-

1. speech recognition.
2. Image processing
3. Multimedia authoring.
4. Video compensing
5. Simulation modeling.

Q1
(E)

→ The techniques used to contemporary processing to increase speed are :-

1) PIPE LINING : pipelining enables a processor to work simultaneously as multiple instructions by performing a different phase for multiple instruction at same time.

2) SUPERSCALAR EQUITION : To issue more than one instruction in every processor clock cycle. In effect multiple parallel pipelines are used.

3) BRANCH PREDICTION :- Branch prediction potentially increases the amount of work for processor to execute.

4) DATA FLOW ANALYSIS : The processor analyzes which instructions are dependent on each others result, to create an optimized schedule of instructions.

Q1
(B)

IAS OPERATION:

The IAS operates by repetiterly performing an instruction cycle. each instruction cycle consists of 2 sub cycles

1. FETCH CYCLE: during fetch cycle the code of the next instruction is loaded into IR and the address portion is loaded into MAR. These instructions are taken from IBR.
2. EXECUTE CYCLE:

This cycle circuitry interprets the opcode and executes the instructions by sending out the appropriate control signals.

Q1
(C)

EMBEDED SYSTEM:

It reffers to the use of electronics and software within a product as opposed to a general purpose computer such as a laptop or desktop.

EXAMPLES:

Cameras mobile phones, microwave ovens, home security systems, printers etc

7. BFU : It handles all binary and hexadecimal floating point operations as well as fixed point multiplication.

8. DFU : It handles both fixed point & floating point operations on numbers

9. RU : RU keeps a copy of complete state of the system that includes all registers.

10. COP : It is responsible for data compression & encryption function for each core.

11. I-CACHE : It is a 64-bit instruction cache, that allows IFU to pre fetch instructions

12. L2 CONTROL : This is control logic that manages traffic through 2 L2 caches.

13. DATA L2 : A 1MB L2 Data cache for all memory traffic other than instructions.

14. INSTR L2 : 1MB L2 cache for instructions.

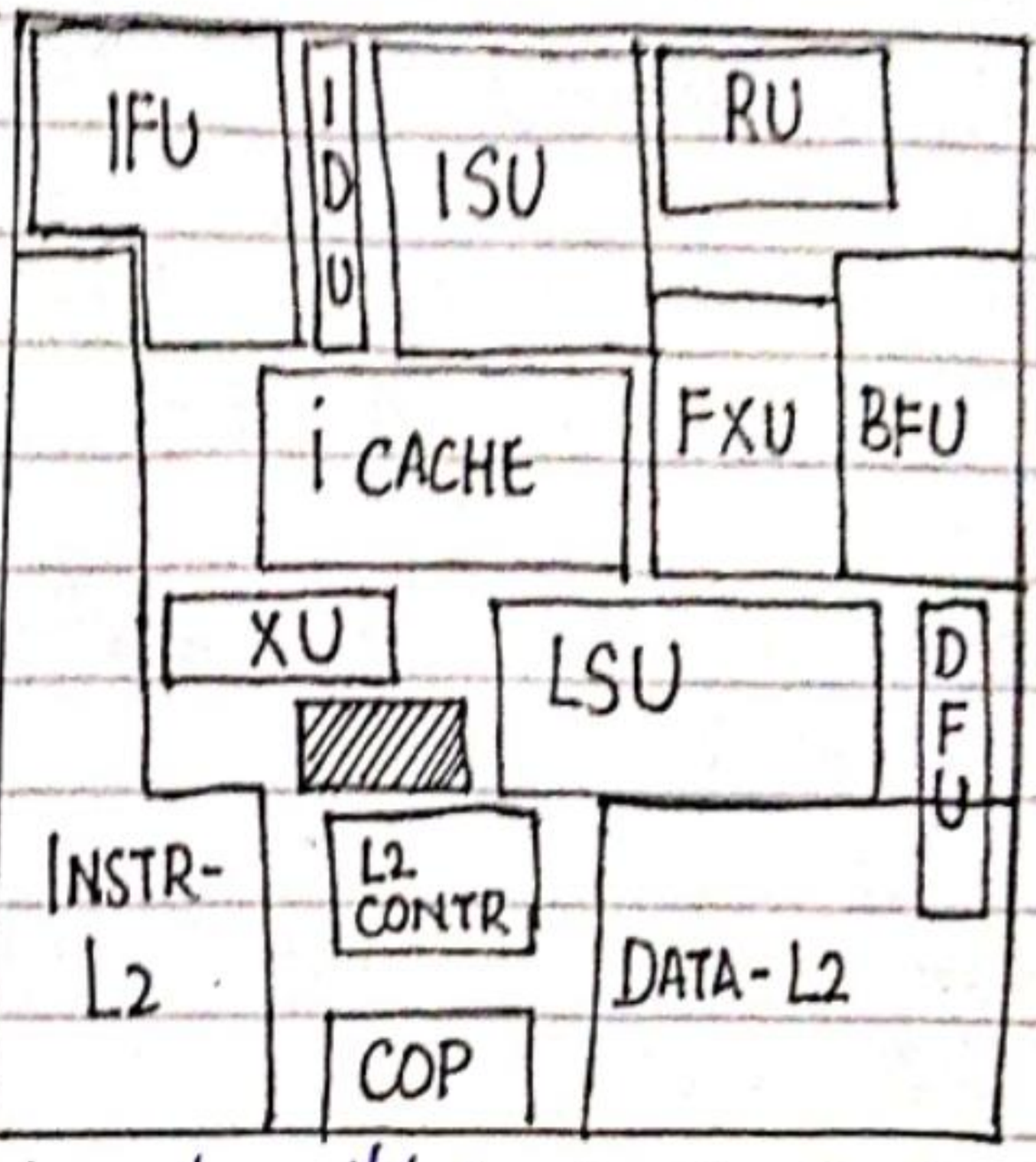
· MID ASSIGNMENT ·

MAZIN KAMRAN
14890 BS(CS) 4th
SIR. AMIN

Q1

(A)

1· ISU : INSTR SEQ UNIT :
Determines the sequence in which instructions are executed what is reffered to a superscalar architecture.



2· IFU : INSTR FETCH UNIT :
Logic for fetching instructions

3· IDU : INSTR DECODE UNIT :
IDU is fed from IFU buffers & is responsible for passing & decoding z/Architecture operations.

4· LSU : LOAD-STORE UNIT : Responsible for handling all types of operand access of all lengths, modes & formats.

5· XU : This unit translates logical address from instruction into physical address in main memory.

6· FXU : It executes fixed point arethmatic operations.