Name : Mazhar Saleem

Id 14455

paper Computer Architecture

Semester 4th

Date 24-04-20

Submitted to Sir Amin

1) :- Ibm zenterprise Ec12 core layout, explain Function of each sub-area?

i) IFU :- (Instruction Fish Unit)
IFU gets information from memory.

ii) IDU :- Instruction decode Unit :-
The IDU is fed From the IFU buffers, and is responsible For the parsing and decoding of all z/Architecture operation codes.

iii) LSU :- (Load Store Unit)
The LSU contains the 96 KB L1 data cashet and manages data traffic btw the L2 data cashe and the Functional executional execution Units. Responsible for handling all types of operand accesses.

iv) XU (translation Unit) :-
This Unit translates logical addresses From instructions into physical addresses

in main memory. The XU also contains a translation lookaside buffer (TLB) used to speed up memory access.

v) FXU (Fixed-point Unit):-
The FXU executes fixed-point arithmetic operations.

vi) BFU (Binary Floating-point Unit):-
The BFU handles all binary and hexa-decimal Floating point operations as well as fixed-point multiplication operations.

vii) DFU (decimal Floating point Unit):-
The DFU handles both Fixed point and Floating point operations on numbers that are stored as decimal digits.

viii) RU (Recovery-Unit):-
The RU keeps a copy of the complete state of the system that include all registers, collects hardware fault signals, and manages the hardware recovery actions.

x) cop (dedicated-coprocessor)
The Cop is responsible for data compression and encryption functions for each core.

xi) I-cache:
This is a 64-KB L1 instruction cache, allowing the IFU

to prefetch Instructions before they are needed.

L2-control:-

This is the control logic that manages the traffic through the two L2 caches.

Data-L2:-

A 1-MB L2 data cache for all memory traffic other than instruc

Instr-L2:-

A 1-MB L2 instruction cache.

(12) Discuss the IAS operation?

Ans operations:-

Explain these operations with reference to fig 1.6.

Memory buffer register (MBR):-

Contains a word to be stored in memory or sent to the I/O unit, or is used to receive a word from memory or from the I/O unit.

The IAS operates by respectively performing an instruction cycle. Each instruction cycle consist of two subcycle.

Fetch cycle:-

The opocode of next instruction is loaded into the IR and the address portion is loaded into the MAR. This instruction may be taken from the IBR or it can be obtained from memory by loading a word into the MBR and the down to the IBR, IR and MAR.

Execute cycle:-

The control circuitry interprets the opcode & executes the instructions by sending out the appropriate control signals to cause data to be moved or an operation to be formed by the ALU.

3 What is embedded system? List daily life examples of it?

The term 'embedded system' refers to the use of electronics and software within a product, as opposed to a general-purpose computer such as a laptop or desktop system.

Daily life examples are:-

- cell phones
- Digital camera
- video camera
- calculator
- Microwave ovens
- Home security system
- Washing machine
- Lighting system
- Thermostats
- Printers
- various automative systems.
- tennis rackets
- toothbrushes
- automated systems.

D

Q1 part D

Ans Different desktop applications that require the great power of contemporary microprocessor based system are:

Image processing.
Three dimensional rendering.
Speech recognition
Video conferencing
Multimedia authoring
Voice and video annotation of files.
Simulation modeling.

E

Q1 part E

Ans Pipelining:

Pipelining enables a processor to work simultaneously on multiple instructions by performing a different phase for each of the multiple instructions at the same time.

Branch prediction:

Branch prediction potentially

increased the amount of work available for the processor to execute.

Branch prediction. Superscalar execution:
This is the ability to issue more than one instruction in every processor clock cycle. In effect, multiple parallel pipelines are used

Data flow analysis.
The processor analyzes which instructions are dependant on each other's results. or data to create an optimized schedule of instructions.

speculative execution:-
This enables the process or to keep its execution engines as busy as possible by executing instructions that are likely to be ~~needed~~. needed.

(F) Q. part F

power:
As the density of logic and the clock speed on a chip increase, so the power density increases and also dissipated the heat.

RC delay:

The speed at which electrons can flow on a chip between transistors is limited by the resistance and capacitance of the metal wires connecting them; specifically, delay increases as the RC product increases.

Memory latency:

Memory access speed (latency) and transfer speed (through put) lag processor speed.

(9) Q1 part (i)

The speed up using a parallel processor with N processors that fully exploits the parallel portion of the program is as follows:

speed up = Time to execute program on a single processor / time to execute program on N parallel processor

$= T(1-f) + \{TF/T(1-F) + Tf/N = 1(1-7)\downarrow + f/N$

H

Q1 part H

Multicore:

* The use of multiple processors on the same chip provides the potential to increase performance without increasing the clock rate.
* Strategy is to use two simpler processors on the chip rather than one more complex processor.
* With two processors larger caches are justified.
* As caches became larger it made performance sense to create two and then three levels of cashe on a chip.

MIC:

Leap in performance as well as the challenges in developing software to exploit such a larger number of cores.

The multicore and MIC strategy involves a homogeneous collection of general purpose processors on a single chip.

GPUs:-

* Core designed to perform parallel operations on graphics data.
* Traditionally found on a plug-in graphics card, it is used to encode and render 2D and 3D graphics as well as process video.
* Used as vector processors for a variety of applications that require repetitive computations.

Q1 Q1 part I

QpI protocol layer:

In this layer, the packet is defined as the unit of transfer, One key function performed at this level is a cache coherency protocol, which deals with making sure that main memory values held in multiple caches are consistent. A typical data packet payload is a block of data being sent to or from a cache.

Q1 part J

Physical and logical Architecture of pcie:

* A root complex device, also refferred to as a chipset or a host bridge connects the processor, and memory subsystem to the PCI express Switch fabric comprising one or more PCIe and PCIe switch devices.
* PCIe links from the chipset attach to the following kinds of devices that implement pcIe.
* Switch: The switch manages multiple pcIe streams.
* PCIe endpoint: An I/O device or controller that implements PCIe such as a Gigabit ethernet switch a graphics or video controller, disk interface, or a communications controller.
* Legacy endpoint: Legacy endpoint category is intended for existing design that have been migrated to PCIe Express, and it allow legacy behaviors such as use of I/O space and locked transactions.
* PCI bridge: Allow older PCI devices to be connected to PCIe-based systems.

Q: Write note on the following:

A: Ans: Main structural components of computer:
There are four main structural components:

1. Central processing unit (CPU): controls the operation of the computer and performs its data processing functions often simply referred to as processor.
2. Main memory: stores data
3. I/O: Moves data b/w the computer and its external environment
4. System interconnection:
Some mechanism that provides for communication among CPU, main memory and I/O.

B:- Key characteristics of a planned computer family?

* similar or identical instr set: In some cases, the lower end of the family has an instruction set that is a subset of that of the top end of the family, this means that program can move up but not down.

Similar or identical OS:- The same basic O·S is available for all family members.
Increasing speed: The rate of instructions execution increase in going from lower to higher family members.
Increasing no of I/O ports: The number of I/O ports increase in going from lower to higher family members.
Increasing memory size: The size of main memory increases in going from lower to higher family members.
* Increasing cost: At a given point in time, the cost of a system increases in go iu from lower to higher family members.

C Stored program computer?
A fundamental design approach first implement in the IAS computer is known as the stored program concept. This idea is usually attributed to the mathematician John von Neumann.
The first publication of the idea was in a 1945 proposal by von Neumann for a new

computer, The EDVAC (Electronic discrete variable computer)

In 1946, von Neumann and his colleagues began the design of a new stored-program computer, referred to as the IAS computer at the princeton institute for Advanced Studies

It consist of:

* main memory: which stores both data and instructions.
* ALU :- capable of operation on binary data.

D Moore's law?

The famous moore's law which was propounded by gordon Moore, co founder of intel in 1965 [MOOR 65]. Moore observed that the number of transistors that could be put on a single chip was doubling every year.

The pace slowed to a doubling year 18 months in the 1970s but has sustained that rate over since.

The consequences of Moore's law are profound:

1. The cost of computer logic and memory circuitry has fallen at a dramatic rate.
2. Because logic and memory elements are placed closer together on more densely packed chips the electrical path length is shortened increasing operating speed.
3. The computer becomes smaller, making it more convenient to place in a variety of environments.
4. There is a reduction in power requirements.
5. With more circuitry on each chip there are fewer interchip connections.

E. Instruction cycle state diagram:

The states in instruction cycle can be described as follows.

Instruction address calculation: Determine the address of the next instruction to be executed. Usually, this involves adding a fixed number to the address of the previous instruction

p.t.o

2

Instruction Fetch (if): Read instruction from its memory location into the processor.

Instruction operation decoding (iod): Analyze instruction to determine type of operation to be performed and operands to be used.

Operand address calculation: If the operation involves reference to an operand in memory or available via I/O, then determine the address of the operand.

Operand Fetch (of) Fetch the operand from memory or read it in from I/O

Data operation: Perform the operation indicated in the instruction

Operand Store (os) Write the result into memory or out to I/O.

F Class interupts?

It is generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine

instruction, or reference outside a User's allowed memory space.

It is generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.

It is generated by an I/O controller, to signal normal completion of an operation request service from the processor, or to signal a variety of error conditions.

It is generated by a failure such as power failure or memory parity error.

Q1c Bus interconnection scheme?

The most common computer interconnections structures are based on the use of one or more system buses.

A system bus consists, typically of from about fifty to hundreds of separate lines. The lines can be classified into 3 functional groups.

P.T.O

a) Data lines:

The ~~data~~ lines provide a path for moving data among system modules, these lines collectively are called the data bus.

b) Address lines

The address line are used to designate the source or destination of the data on the data bus.

The width of the address bus determines the maximum possible memory capacity of the system.

control lines:

The control lines are used to control the access to and the use of the data and address lines. Because the data and address lines are shared by all components, there must be a means of controlling their use. Typically control lines include:

↑ Memory write, memory read, I/O write.

Q3 Differentiate each of the following

A. Computer organization and Architecture?

	Computer Architecture	comp. Organization.
(1)	Computer Architecture is concerned with the way hardware components are connected together to form a computer System.	Computer organization is concerned with the structure and behavior of a computer system as seen by the user.
(2)	It acts as the interface b/w hardware and software.	It deals with the components of a connection in a system.
(3)	While designing a comp. Arch is considered first	An organization is done on the basis of Architecture
	CA deals with high level design issue	CO deals with low level. design issues

(3) B RISC and CISC

Basis for comparison	RISC	CISC
Emphasis on	Software	Hardware
Includes	Single clock	Multi-clock
Instruction set size	small	Large
Instr format	Fixed 32 bit	varying formats (16-64)
clock rate	50-150MHz	33-50 MHz
cpu control	Hardwired without control memory	Microcoded using control memory (CROM)

C

* Microprocessors or Microcontrollers?
Microprocessors is an IC which has only the CPU inside them i.e only the processing powers such as intel pentium 1, 2, 3, 4 core 2 duo etc.
* Micro controller has a CPU, in addition with a fixed amount of RAM, ROM and other peripherals also embedded on a single chip.

D

	Cortex A	Cortex R	Cortex M
①	Application processors	Real time processors	Micro-controllers
②	High performance and efficiency	High performance	low-performance
③	Applications mobiles, tablet	Applications- medical deices car system	Application robotic System
④	It is connected to large amount of memory	—	It is connected to less memory.
⑤	Runs at high clock frequency	It turns on high clock frequency	It runs at clock speed

E Q3 part E

Ans In the interrupt cycle, the processor checks to see if any interrupts have occurred, indicated by the presence of an interrupt signal.

* If no interrupts are pending the processor proceeds to the fetch cycle and fetches the next instruction of the current program.

F Q3 part F

A disabbled interupts simply means that the processor can and will ignore that interrupt request signal. If an interrupt occurs during this time, it generally remains pending and will be checked by the processor after the processor has enabled interrupts.

A nested is to allow an interrupt of higher priority

to cause a lower priority interrupt handler to be itself interrupted. A User program begins at t=0. At t=10, a printer interrupts occurs. User information is placed on the system stack and execution continues after the printer interrupts service routine While this routine is still executing at t=15, a communication interrupts occurs.

(3) Q part (7) Q3 :-

programing in hardware:

The program is in the form of Hardware and is termed a hardwared program

Suppose we construct a general purpose configuration of arithmetic and logic functions. This set of Hardware will perform various function on data depending on control signals applied to the hardware, An the original case of customized hardware, the system accepts data produces results.

Prog in Software:

The new method of programming which is a sequence of codes or instruction is called Software programing. In this programming is much easier, all we need to do is provide a new sequence of code, Each code is in effect, an instruction, and part of the hardware interprets each instruction and generates control signals.

(4) Solve each of the following:

Ans Contents are divided up into two 5 bit instructions LH and RH

LH instruction = 0 10 F A
opcode = 01
address = 0 FA
RH instruction = 21 0 F B
opcode = 21
address = 0 F B

Since this is in the hexadecimal form you have to convert the numbers to binary form (use the 4AS instruction set).

LH instructions:
01 = 00000001 = Load M(X)
M(X) refers to the memory address location OFA.
The first 5 bits of 08A should read - LOAD M(OFA).
RH instructions:
21 = 00100001 = STOR M(X)
M(X) refers to the memory address location OFB
The second 5 bits of 08A should read - STOR M(OFB)
Finally the assembly language code for 08A 010FA210FB is
LOAD M(OFA)
STOR M(OFB)

2 Here is a simple way to understand this problem:
contents are divided up into two 5 bit instruction LH and RH
LH instruction = 010FA
opcode = 01
address = OFA
RH instruction = 0F08D
opcode = 0F
address = 08D

Since this is in hexadecimal form you have to convert the numbers to binary form: (use 4AS instr set)

LH instruction:

01 = 00000001 = LOAD M(x)

M(x) refers to the memory address location OFA.

The first 5 bits of 08B should read - LOAD M(0FA)

RH instruction:

0F = 00001111 = JUMP + M(x, 0:19)

refers to the memory address location 08D.

The second 5 bits of 08B should read - JUMP + M(08D, 0:19)

Finally the assembly language code for 08B 010FA 0F08D is

LOAD M(0FA)

JUMP + M(08D, 0:19)

3 contents are divided up into two 5 bit instructions LH and RH

LH instruction = 020 FA

opcode = 02

address = 0FA

RH instruction = 210FB

opcode = 21

address = 0FB

Convert hexadecimal to binary form.
LH instructions:
02 = 00000010 = LOAD - M(X)
M(X) refers to the memory address location OFA.
The first 5 bits of 08C should read - LOAD - M(OFA)
RH instructions:
21 = 00100001 = STOR M(X)
M(X) refers to the memory address location OFB.
The second 5 bits of 08C should read - STOR M(OFB)
Finally the assembly language code for 08C 02OFA210FB is
LOAD - M(OFA)
STOR M(OFB)

b Explain what it does?
① An 08A address, the M(OFA) transfer to the accumulator and transfer contents of accumulator to memory location OFB.
② An 08B address the M(OFA) transfer to the accumulator and take next instruction from left half of M(08D).

3 An 08C address the - M(OFA) transfer to the accumulator and transfer contents of accumulator to memory location OFB.

Q4B part B Q 4

a. opcode = 00000001
operand = 00000000010

b. In the beginning, the CPU have to fetch the instruction from the memory, then the instruction will include the address of the data which is required to load. through the execution time the memory will be accessed in that time to load the data contents which is located at that address for a total of two trips to memory.

Q4 C

Effective CPI:

CPi = (1*4600) + (2*3300) + (2*1600) + (2*900) / 100

CPi = 162000/100

CPi = 1620

MiPS Rate:

MiPS rate = 60 MHZ / 1620 * 106

MiPS rate = 60 * 106 / 1620 * 106

MiPS rate = 60 / 1620

MiPS rate = 0.037

Execution Time:

T = IC / (MiPS * 106)

T = 104000 / (0.037 * 106)

T = 104000 / 37 * 103

T =  2811 * 10-3

T = 2.811 Sec

Q4

part D :-

A. Since we have the same instruction mix, that means the additional instructions for each task could be allocted appropriately between the instruction types. Therefore, the following table be gotten:

Instruction type	CPI	Instruction Mix
Arithmetic and logic	1	60 %
Load/st are with cashe hit	2	18 %
Branch	4	12 %
Memor reference with cashe miss	12	10 %

The average CPi = (1x0.6) + (2x0.18) + (4x0.12) + (12x0.1) = 2.64 Therefore, The CPi has been increased Since The time For memory access is also increased.

b MiPS = 400/2.64 = 152. There is a corresponding drop in The MiPS rate.

C. The speedup Factor equals To the ratio of The execution Times. The execution Time is calculated as The following: T = ic/(MiPS*106) For The one Processor, T1 = (2*106)/(178*106) = 11ms.

For The 8 Processors, each Processor executes 1/8 of the 2million instructions Plus The 25,000

T8 = 2*106 ÷ 8 + 0.025*106 / 152*106

T8 = 1.8 ms

Therefore we have

SpeedUP = Time To execute Program on a Single Processor/

Time To execute Program on N Parallel Processor

SpeedUP = 11/1.8

SpeedUP = 6.11

d. BY depending on The information given, It is not obvious how to quantify This EFFect in Amdahl's

Equation. Therefore, if it is supposed that the Fartion of code, which is parallelizable, is F = 1, Then Amdahl's law decreases To

SpeedUP = N = 8.
Therefore, The actual Speedup is only about 75% of the Theoretical Speedup.

Q4 part E

1. a. The Pc contains 300, The address of the First instruction. This value is Loaded into The Mar.b. The value in Location 300 (which is The instruction with the value 1940 in hexadecimal) is Loaded into The MBR and The Pc is incremented. These Two Steps Can be done in Parallel.C. The value in The MBR is Loaded into The IR.
2. a. The address Prtion of The IR (940) is Loaded into the MAR.b. The value in Location 940 is Loaded in To the MBR.C. The value in The MBR Loaded into the AC.
3. a The value in The Pc (301) is Loaded in To The MAR.b. The value in Location 301 (which is The instruction with the value 5941) is Loaded in To The MBR, and The Pc is incremented.C. The value in The MBR is Loaded in To The IR.
   a. The address portion of the IR (941) is loaded into the MAR

b. the value in location 941 (~~which~~ is loaded into the MBR.
c. the old value of the AC and the value of location MBR are added and the result is stored in the AC.

5) a. The value in the PC (302) is loaded in to the MAR. B. The value in location 302 (which is the instruction with the value 2941) is loaded into the MBR, and the PC is increment. C. the value in the MBR is loaded into the IR.

6) a. The address portion of the IR (941) is loaded into the MAR.
b. the value in the AC is loaded into the MBR. c. the value in the MBR is stored in location 941.

Q4 F :-

a. $2^{24} = 16$ MBytes

b. (1) If the local address bus is 32 bits, the whole address can be transferred at once and decoded in memory. However because the data bus is only 16 bit instruction or operand.

(2) The 16 bits of the address placed on the address bus can't access the whole memory.

Thus a more complex memory interface control is needed to latch the first part of the address and then the second part.

c. The program counter must be atleast 24 bits. typically a 32-bit microprocessor will have a 32-bit external address bus and a 32-bit program counter, unless on chip segmeud registers are used that may work with a smaller program counter. If the instruction register is to contain the whole instruction it will have to be 32-bits long; If it will contain only the op code (called the op code register then it will have to be 8 bits long.

Q4 part 9:- 34

A bus cycle takes 0.25 μs So a memory cycle takes 1 μs if both operands are even aligned. It takes 2 μs to fetch the two operands. If one is odd aligned the time required is 3 μs, If both are odd aligned the time required is 4 μs.