UNSOLVED UESTIONS: Question # 3 and 7(a)

Q#1: Answer any 6. (5marks each)

a) Distinguish and explain computer architecture and computer organization? What is its importance in Computer science field?

1st Lecture, 1st Page.

b) State the key concepts of Von Neumann architecture.

2nd Lecture.

c) Clearly explain how the working of cache memory is related to some characteristics of large program.

- **Study of large program reveal that most of the execution time is spend, in the execution of a few routines (sub-sections). When the execution is localized within these routines, a number of instructions are executed repeatedly, this property of programs is known as LOCALITY OF REFERENCE.**
- **Thus, while some localized areas of the program are executed repeatedly, the other areas are executed less frequently.**
- **To reduce the execution, time these most repeated segments may be placed in a fast memory known as CACHE (or Buffer) Memory.**
- **The memory control circuitry is designed to take advantage of the property of LOCALITY OF REFERENCE.**
- **If a word in a block of memory is read, that block is transferred to one of the slots of the cache.**

Taken from Cache Memory handout 1st page.

d) Discuss how the main memory 'Block size' can affect the performance of cache memory?

- **Larger blocks reduce the number of blocks that fits into a cache. Small number of blocks result in data being overwritten shortly after they are fetched.**
- **As a block become larger, each additional word is farther from the requested word and therefore less likely to be needed in near future.**

Taken from Line Size in Cache handout. Not sure if this is the correct answer.

e) What Elements instruction must specify (implicitly or explicitly)?

- **The Processor Architecture is traditionally described by describing the number of addresses contained in an instruction.**
- **Present day computer doesn't need all the FOUR addresses, explicitly.**

- **ONE, TWO or THREE address instruction may be used, the address of the next instruction is provided implicitly by the program counter.**
- **In single address format / machine, many addresses are implied, it can be seen from the example. One address is implied the other address is explicit.**
- **The deficiency of 3rd Address is made up by adding one more instruction. Thus the overall program length has gone up.**
- **Most computers employ a mixture of instruction formats in terms of number of explicit addresses.**

Taken from Instruction Set handout (Instruction representation heading).

f) What is the purpose of dirty bit associated with a cache line?

Dirty bit (also called used bit) is used in "Write back policy" when writing in cache. In write back policy we update only the cache & mark the location with a flag (dirty bit or use bit) and when the block is removed, if dirty bit is set, the corresponding location in Main Memory is updated.

g) What is the purpose of guard bits? Explain their usefulness with the help of a simple addition of two floating point numbers.

Registers in ALU contain additional bits called guard bits for holding implied bit which are used to pad out the right end of the Significand with 0s.
EXAMPLE:

The addition of the 2 numbers is:

0.0256*10^2
2.3400*10^2 +

2.3656*10^2

After padding the second number 2.3400*10^2 with two 0's, the bit after 4 is the guard digit, and the bit after is the round digit. The result after rounding is 2.37 as opposed to 2.36. without the extra bits (guard and round bits), i.e., by considering only0.02+2.34=2.36. The error therefore is 0.01.

i) Why computer today is fitted with a hierarchy of memory? What does it represent as you move down from top?

A typical computer system is fitted with a hierarchy of memory sub systems because no single technology is optimized in satisfying the memory requirements of a computer system.
As we move top to bottom we observe:

- **Increase in capacity.**
- **Increase in Access Time.**
- **Decrease in cost.**

- **Decrease in frequency of use.**

Q#2: a) Compare how signed magnitude and 2s compliment representation differs in their range, expansion of bit length, overflow rule and other features. Which representation is more suitable for the design of ALU and why?

Range of signed magnitude for 8 bit numbers is +/- 2^7-1
Range of 2s compliment representation of 8 bit numbers is -2^{N-1} - 2^{N-1}-1.

Expansion of bit length in sign magnitude is done by adding zeros on the left side of a number. Expansion of bit length in Two's complement representation is done by extending the sign bit to all new positions.

Overflow in sign magnitude occurs when a resulting number is bigger than 8 digits.

Overflow in two's complement is determined by a result with the wrong sign. I.e., if you add two positives and get a negative result, or add two negatives and get a positive.

2s compliment representation is more suitable because in sign magnitude addition and subtraction require consideration of both sig and magnitude of the number. In sign magnitude, there are two representation for '0', whereas in 2s compliment there is only single representation of 0.

b) Using a suitable multiplication algorithm for binary integers multiply 15 by 11. Explain all steps taken. Choose suitable register size with reasons or your choice. (See picture for blurred values)

10100101

Q#3: a) for the 32-bit floating point number as shown below, answer the following question using IEEE representation: 1 10001101 0001 1100 0000 0000 0000 000

- **Is it +ve or –ve exponent? What is the exponent value of the number?**
- **What is the significand value and what is it's sign?**
- **What decimal number is represented by the above bit pattern?**
- **Rewrite the above FP number (32 bit) to represent the smallest +ve number. What is it's value?**

b) Explain clearly how the floating point operation can implemented using fixed point circuitry?

Q#4: a) identify and list types of exchanges that are needed for memory modules, I/O modules and processors.

Lecture 2 Last Page.

b) Identify and discuss the elements that must be considered for a bus design.

Lecture 3 Last Page.

Q#5: a) discuss the significant characteristics of point to point interconnect schemes.

Lecture 4 QPI Characteristics.

b) Compare the working of the following memory access methods. Also identify some of their applications:

*** Sequential**

*** Direct**

*** Associative access**

Lecture 5 First Page.

Q#6: a) explain how the presence of cache improve the system performance. Why do the transfers between main memory and cache memory take place in the form of blocks consisting of few memory locations? What is the purpose of TAG part of cache slot?

- **Study of large program reveal that most of the execution time is spend, in the execution of a few routines (sub-sections). When the execution is localized within these routines, a number of instructions are executed repeatedly, This property of programs is known as LOCALITY OF REFERENCE.**
- **To reduce the execution time these most repeated segments may be placed in a fast memory known as CACHE (or Buffer) Memory which improves system performance.**
- **a tag identifies which particular block is currently being stored in a particular cache line.**
- **Tag is usually a portion of main memory address.**

b) Explain the function of replacement algorithm in the implementation of cache memory. Discuss how LRU is implemented.

REPLACEMENT ALGORITHM

- **When Cache is full, a decision is to be made about which block should be removed to make room for the other block from the main memory.**
- **The Locality of Reference provide a clue to reasonable strategy.**
- **It can be assumed that a block which has recently been referenced will also be referenced in near future.**
- **Thus, the block that has stayed for long without being referenced should be removed. Such a block is known as LEAST RECENTLY USED Block and the algorithm to determine such a block is known as LRU Algorithm.**
- **A counter is used to keep track of LRU Block in Cache. For a 4 slots cache 2 bit counter is used, for a 8 slots cache 3 bits are required.**

FOR A READ OPERATION:

IN CASE OF HIT (Content in Cache):

- **Counter for the block referenced is set to '0'**
- **All other counters with value originally lower than referenced one are incremented by 1, while all others remain unchanged.**

IN CASE OF MISS (Content not in Cache)

IF CACHE IS NOT FULL:

- **The counter associated with the new block loaded form Main Memory is reset to '0'.**
- **Values of all other counters are incremented by 1.**

IF CACHE IS FULL:

- **The block with the largest count is removed.**
- **New block is put in its place.**
- **Its counter is reset to '0'**
- **The remaining counters are incremented by 1**

Other Algorithms are Least Frequently Used (LFU), First In First Out (FIFO) and Random (selecting line to be replaced at random).

Tests have shown that Random technique is best while FIFO has shown poor performance.

Q#7: a) a computer system uses cache memory of 16K words organized in a block set associative manner with a block per set and block size of 8 words. Give the main memory address format as seen by the control circuitry if the main memory size in 2M words.

b) What is understood by access time and cycle time for RAM and non-RAM systems?

Lecture 5, Access Time heading.

Q#8: a) explain how precision of a floating point number can be improved, and how to increase range. Use examples to support your answer.

Precision increase by increasing buts for significand. A large exponent field yield a large range.

b) Explain what is understood by 'Biased exponent' representation? How does it take into account the sign of exponent? What are the advantages of this kind of representation?

In floating point representation, the exponent is stored in biased representation. (A fixed value, called the bias, is subtracted from the filled to get the true exponent value). Typically bias equals 2^{k-1} - 1 where k is the number of bits in binary exponent. (8 bit exponent field yield the number 0 through 255 with bias 2^{8-1} -1 = 127 true exponent value are in range - 127 to + 128).

Advantage of bias representation is that non negative Floating Point Number can be treated as integer for comparison purpose. The leftmost bit store the sign of the number.

Q#9: a) discuss the working, advantages and disadvantages of the following addressing modes with the help of suitable diagram.

- **Immediate**
- **Register**
- **Displacement**

IMMEDIATE ADDRESSING:
The simplest form of addressing is immediate addressing, in which the operand value is present in the instruction

Operand = A

This mode can be used to define and use constants or set initial values of variables.

REGISTER ADDRESSING:

Register addressing is similar to direct addressing. The only difference is that the address field refers to a register rather than a main memory address:

EA = R

o clarify, if the contents of a register address field in an instruction is 5, then register R5 is the intended address, and the operand value is contained in R5.

DISPLACEMENT ADDRESSING:

A very powerful mode of addressing combines the capabilities of direct addressing and register indirect addressing.

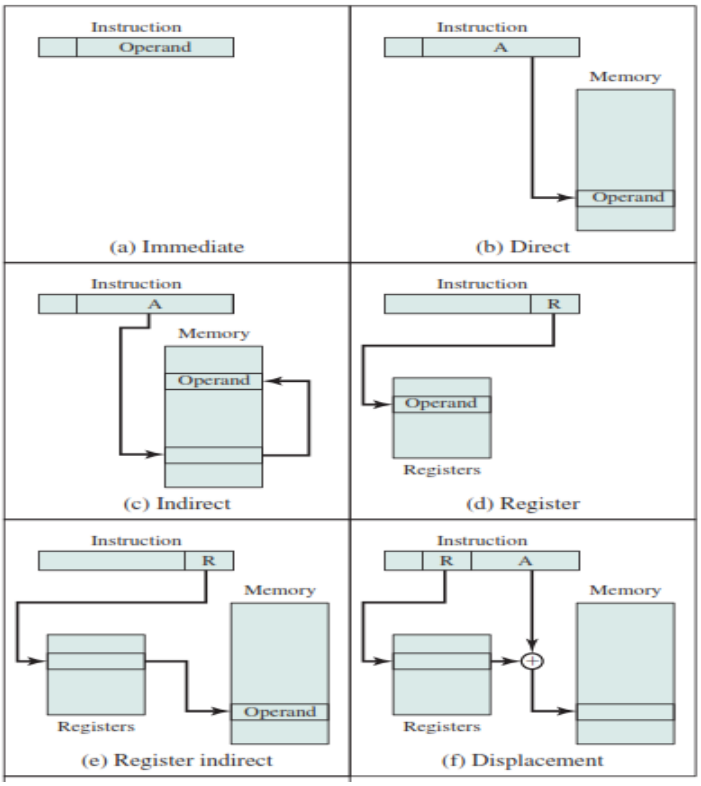
EA = A + (R)

Displacement addressing requires that the instruction have two address fields, at least one of which is explicit.

ADVANTAGES / DISADVANTAGES :

Mode	Algorithm	Principal Advantage	Principal Disadvantage
Immediate	Operand = A	No memory reference	Limited operand magnitude
Direct	EA = A	Simple	Limited address space
Indirect	EA = (A)	Large address space	Multiple memory references
Register	EA = R	No memory reference	Limited address space
Register indirect	EA = (R)	Large address space	Extra memory reference
Displacement	EA = A + (R)	Flexibility	Complexity
Stack	EA = top of stack	No memory reference	Limited applicability

DIAGRAMS:



b) Discuss how number of address in an instruction affect the processor architecture?

The Processor Architecture is traditionally described by describing the number of addresses contained in an instruction. ONE, TWO or THREE address instruction may be used. Most CPU designs involve a variety of Instruction Format.

LESS ADDRESSES MEANS:

- **More Basic Instructions and less complex design of the CPU**
- **Instruction word is shorter**
- **More Instructions means longer program and longer execution time**
- **And the program design is also complex**

Most computers employ a mixture of instruction formats in terms of number of explicit addresses.

Q#10: a) how instruction pipelining enhances system performance and what factors serve to limit the performance enhancement.

Greater performance can be achieved by taking advantage of improvements in technology and organizational enhancement, an organizational approach, which is quite common, is instruction pipelining.

In instruction pipelining, instruction execution cycle is perceived as being divided into a number of stages. As a simple approach, consider subdividing instruction processing into two stages: fetch instruction and execute instruction. There are times during the execution of an instruction when main memory is not being accessed. This time could be used to fetch the next instruction in parallel with the execution of the current one.

FACTORS SERVE TO LIMIT THE PERFORMANCE ENHANCEMENT:

- **If all the stages are not of equal duration, there will be some waiting involved at various pipeline stages.**
- **In case of a conditional branch instruction, several instruction fetches can be invalidated.**
- **An interrupt can also invalidate several instructions fetches.**
- **Some stage may depend on the contents of a register that could be altered by a previous instruction that is still in the pipeline.**

b) Describe the following types of hazards:

*** Structural * data * control**

STRUCTURAL HAZARD:

A structural hazard occurs when two (or more) instructions that are already in the pipeline need the same resource. The result is that the instructions must be executed in serial rather than parallel. A resource hazard is sometime referred to as a resource hazard.

DATA HAZARD:

A data hazard occurs when there is a conflict in the access of an operand location. For example : Two instructions in a program are to be executed in sequence and both access a particular memory or register operand. There are three types of data hazards:

- **read after write**
- **write after write**
- **write after read**

CONTROL HAZARDS:

A control hazard, also known as a branch hazard, occurs when the pipeline makes the wrong decision on a branch prediction and therefore brings instructions into the pipeline that must subsequently be discarded.

Taken from Instruction pipelining handout (last handout) last topic (Pipeline hazards).