B38DF: Computet Architecture and Embedded Systems 2016
Part 1 - Tutorial 1 – Weeks 1-3

For the first three questions in this tutorial (true/false, multiple choice, and terms), the answers can be easily found in the lecture slides. The solutions of the remaining questions are given in this tutorial.

Q1. Indicate whether the following statements are True (**T**) or False (**F**).

Statement	True/False (?)
1) A characteristic feature of every computer is the presence of an ALU in the main memory.	False
2) In a CPU, it is only the Control Unit which accesses the Registers.	False
3) One of the challenges of the study of computer architecture is the aesthetics of the hardware that composes the computing system.	False
4) In von Neumann Architecture, there is only one set of address/data busses between the CPU and memory	True
5) A computer's primitive instrcutions form a programming language.	True
6) Each computer language in different levels is intended to be used by applications programmers in another format.	False
7) C++ and Java are examples for Assembly Languages but not for Machine Languages.	False
8) Program Counter is a special type of register in CPU which counts the number of instructions and in this way keeps track of the sequencing of instrucitons.	False
9) Register-register instructions perform operations on the registers.	True
10) Registers are located somewhere in between the CPU and the Main Memory	False
11) Mnemics are representations of Opcodes and they cannot be interpreted by the computer.	True
12) Pipelining is a method to increase the speed of execution of individual instructions.	False
13) Primary Memory is the location where both data and program are stored.	True
14) A computer with 32-bit word has 4 bytes/word.	True
15) The longest memory words are carried to the cache for easy access by the CPU	False

Q2. Indicate (circle) all and only the correct answer(s) for the following questions. More than one correct answer is possible.

Statement	Answer(s)
1) General term for every kind of set of machine-readable instrcutions that directs a computer's processor to perform specific operations.	a) Instruction b) Software c) Command d) Program e) Operating system

2) The architecture which holds the data and instructions in separate memories.	a) Von Neumann b) Harvard c) Unified d) Split
3) Von Neumann architecture has:	a) Separate data buses for instructions and data b) Separate meories for instructions and data c) Single data bus for instructions and data d) Single memory for instructions and data
4) Datapath is composed of these elements.	a) CPU b) Program counter c) ALU d) Registers e) Connection buses f) Memory
5) On which one does the Next Instrcution Reference of an instruction have a more direct/immediate impact?	a) Program counter b) Instruction register
6) Single Machine Instrcutions might be composed of the following entities.	a) Always only one Opcode b) One or more Opcodes c) Always only one Operand Reference d) One or more Operand References e) Always only one Mnemonic f) One or more Mnemonics
7) How many memory addresses can be referenced by an 16-bit microprocessor with 20 address lines?	a) 1 G b) 1 M c) 66 M d) 32 M

Q3. Write the terms that are described by the following statement.

Statement	Term
1) Collection of physical elements that constitutes a computer system.	Hardware
2) A sequence of instructions describing how to perform a certain task.	Program
3) A program that replaces each instruction in a higher computer language with an equivalent sequence of instructions in a lower level language.	Translator

4) The register that holds the instruction currently being executed.	Instrcution register
5) The smallest addressable unit in a memory.	Cell
6) The term that is used for the combination of the data bits for the memory word and the chcck bits.	Codeword
7) The Hamming distance wetween the codewords 001101001100 and 001001001101.	2
8) The form of cache which allows for parallel access of instructions and data, hense more suitable with pipelining.	Split cache

Q4. Describe the following terms with a few sentences.
a) Virtual Machine
b) Compiler
c) High-level languages
d) Endianness
e) Locality principal

Use this space for your work

a)**Virtual Machine:**
A hypothetical computer whose machine language is L1 and which performs automatically the translation or interpretation operations, without the user of L1 noticing or bothering whether it is translation or interpretation.

b)**Compilers**
Compilers are the translators (occasionally interpreters) that translate (interpret) the high-level languages (Level 5) to lower level languages (Level 3 or Level 4).

c)**High-Level Languages**
Languages designed to be used by applications programmers with problems to solve, such as C, C++, Java, LISP, and Prolog.

d)**Endianness**
When you want to break a large value (such as a word) into several small ones, you must decide on an order to place it in memory. Endianness refers to the order of the bytes, comprising a digital word, in computer memory. It also describes the order of byte transmission over a digital link. Words may be represented in big-endian or little-endian format. With big-endian the most significant byte of a word is stored at a particular memory address and the subsequent bytes are stored in the following higher memory addresses, the least significant byte thus being stored at the highest memory address. Little-endian format reverses the order and stores the least significant byte at the lower memory address with the most significant byte being stored at the highest memory address. (Source: Wikipedia)

e)**Locality Principle:**
The memory references made in any short time interval tend to use only a small fraction of the total memory. If a given memory reference is to address A, it is likely that the next memory reference will be in the general vicinity of A.

Q5. Consider the different memories given in the rows of the following table. Fill in the empty cells of the table using the given information.

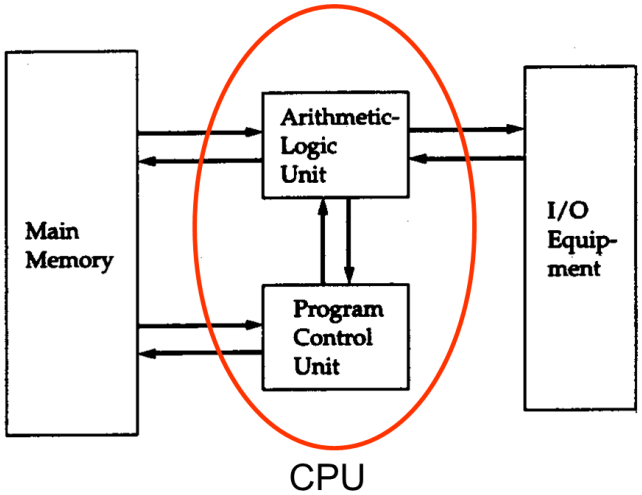
Memory	# of addr	# of data lines	# of addr lines	# of total bytes
1M x 8	1,048,576	8	20	
2M x 4	2,097,152	4		
1K x 4	1024		10	512 B
4M x 32	4,194,304		22	
16K x 64	16,384			

Use this space for your work

Memory	# of addr	# of data lines	# of addr lines	# of total bytes
1M x 8	1,048,576	8	20	1 MB
2M x 4	2,097,152	4	21	1 MB
1K x 4	1024	4	10	512 B
4M x 32	4,194,304	32	22	16 MB
16K x 64	16,384	64	14	128 KB

Q6. Draw the schematic diagram of a standard von Neumann Architecture showing the principal elements and the communication links between them.

Use this space for your work



Q7.

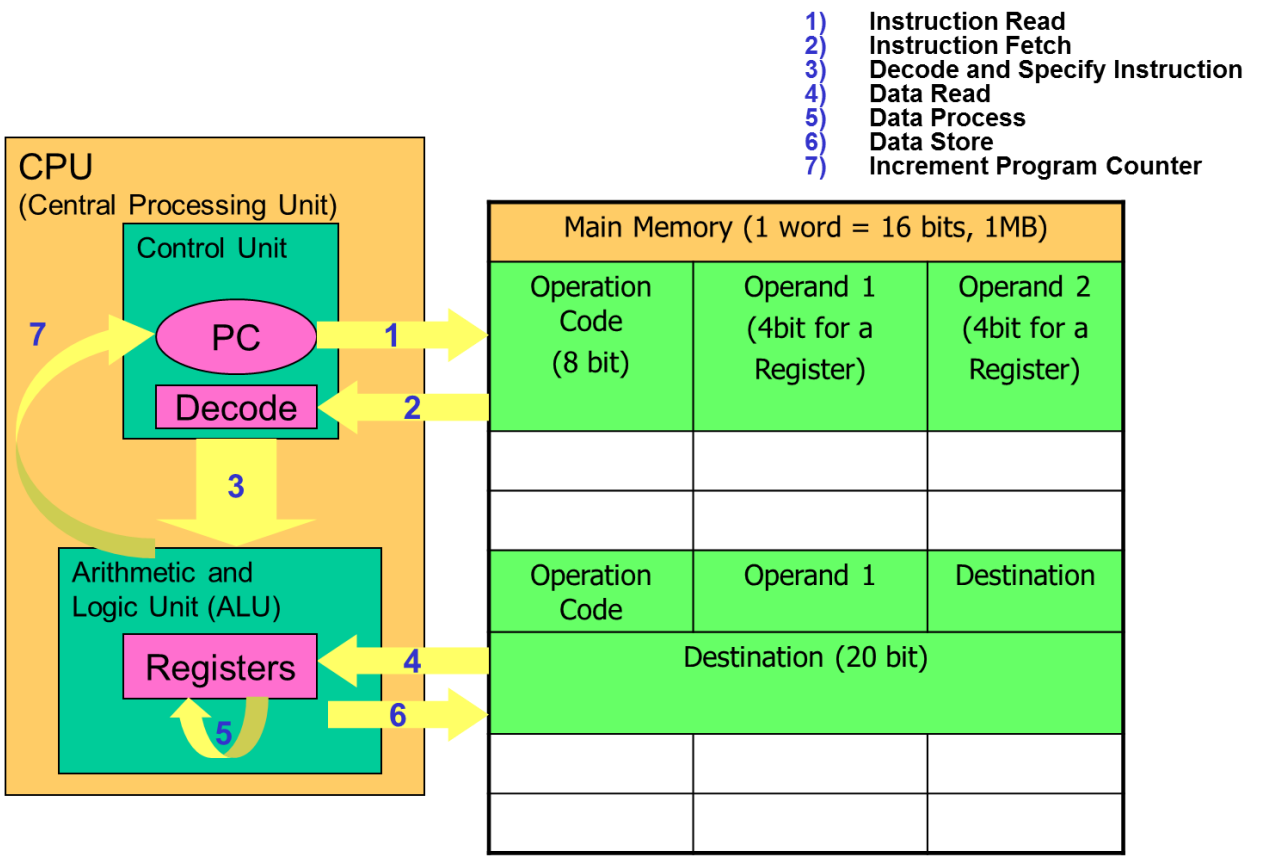
(i) Satate the seven steps of CPU Cycle,

(ii) draw a schematic clearly showing the CPU, ALU, specific registers, and Main Memory,

(iii) and indicate the seven steps of the CPU cycle with arrows (projections) on your diagram using number on the arrows.

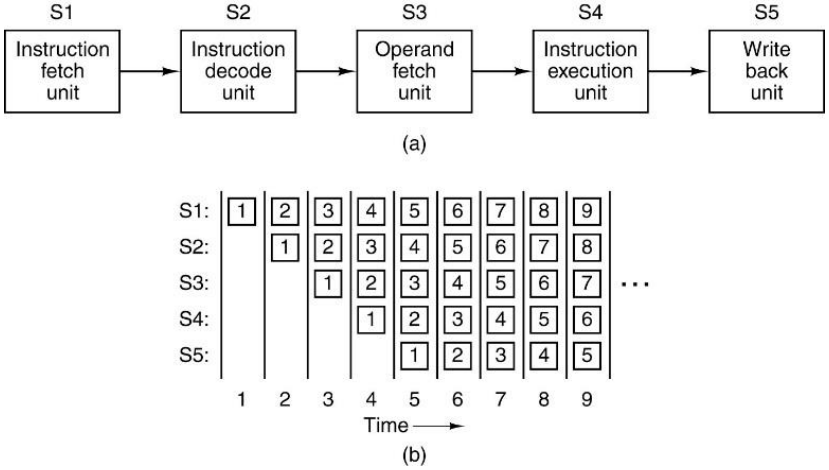
Use this space for your work

1) **Instruction Read**
2) **Instruction Fetch**
3) **Decode and Specify Instruction**
4) **Data Read**
5) **Data Process**
6) **Data Store**
7) **Increment Program Counter**



Q8.

Consider the pipelining as described by the figure below:



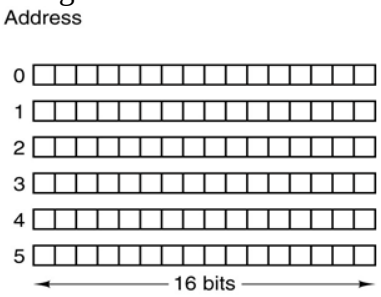
With this pipelining the Processor Badnwith is given to be 500 Millions instruction Per Second (MIPS). Assuming that each stage of the pipelining takes a clock cycle, compute the clock cycle time of this CPU.

Use this space for your work

- ⇨ Processing Rate = #of stages × [1/Latency]
  p = n × 1 / L
- ⇨ Latency: L = n / p = 5 / 500 {unit: 1/MIPS} = 0.01×10^{-6} sec/instruction
- ⇨ = 10×10^{-9} sec/instruction
- ⇨ = 10 nsec
- ⇨ Latency = #of stages × Clock Cycle Time
  L = n × T
- ⇨ Clock Cycle Time: T = L / n = 10 / 5 nsec = 2 nsec

Q9.

Consider the memory structure in the figure below:



How many bits do you need (minimum) to reference the cells in this memory?
What is the address of the cell numner 3?

Use this space for your work

- Number of cells : 6
- Addresses: 0 to 5
- How many bits?
  → **3** ($2^2 < 5 < 2^{\underline{3}}$)
- Address of cell 3 ?
  → 0 1 1

Q10.

Consider the memory structure a part of which is shown in the figure below with its contents. (No hexadecimal coding is assumed, ages and numbers are coded as binary values in each word.)

Cell 0	J	A	C	K	**Cell 3**
Cell 4		B	R	O	**Cell 7**
Cell 8	W	N	0	0	**Cell 11**
Cell 12	A	G	E	:	**Cell 15**
Cell 16	0	0	0	28	**Cell 19**
Cell 20	N	U	M	B	**Cell 23**
Cell 24	E	R	:		**Cell 27**
Cell 30	0	0	2	35	**Cell 33**

Byte

Word

a) How many bits are there in each word?
b) Is this a Big Endian or Little Endian byte ordering?
c) What is the age of this person?
d) What is the number of this person?
e) Which ever byte ordering this is, draw the similar table for the other type of byte ordering and indicate the number of cells on the right and left sides similarly.
f) Considering a cell-to-cell direct transformation; fill in the entries of the table you have drawn.
g) If the computer reads the data in your table according to the byte ordering of your table, which information will be erroneous? How will they be read?

Use this space for your work

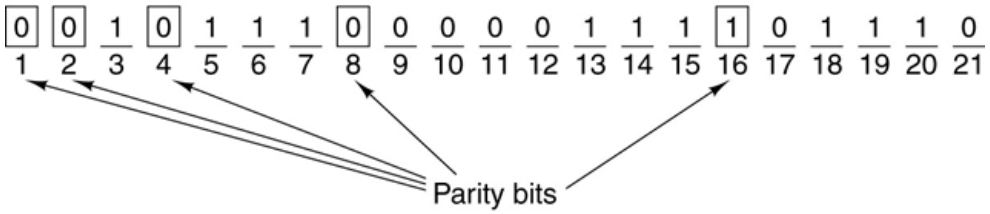
a) Word: 32 bits
b) Big Endian
c) 28
d) 547
e) Little Endian

Cell 0	K	C	A	J	**Cell 3**
Cell 4	O	R	B		**Cell 7**
Cell 8	0	0	N	W	**Cell 11**
Cell 12	:	E	G	A	**Cell 15**
Cell 16	28	0	0	0	**Cell 19**
Cell 20	B	M	U	N	**Cell 23**
Cell 24		:	R	E	**Cell 27**
Cell 30	35	2	0	0	**Cell 33**

f) The age and number will be erroneous. Age will be read as 28×2^{24} and number will be read as $35 \times 2^{24} + 2 \times 2^{16}$.

Q11.

Consider the codeword in Hamming code form as indicated below, together with the data and parity bits.



Remmeber that in Hamming's Algorithm, each parity bit chack specific bit positions as given below:

- Bit **1** checks bits 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21
- Bit **2** checks bits 2, 3, 6, 7, 10, 11, 14, 15, 18, 19
- Bit **4** checks bits 4, 5, 6, 7, 12, 13, 14, 15, 20, 21
- Bit **8** checks bits 8, 9, 10, 11, 12, 13, 14, 15
- Bit **16** checks bits 16, 17, 18, 19, 20, 21

Based on that rule:

i) Determine whether there is an error or not in any bits of this codeword.
ii) If there is an error identify which bit(s) is(are) eeroneous.
iii) If there is any error, write the corrected codeword by clearly indicating the corrected bit(s).

Use this space for your work

- Parity bit **1** correct (1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21 contain six 1s)
- Parity bit **2** incorrect (2, 3, 6, 7, 10, 11, 14, 15, 18, 19 contain seven 1s)
- Parity bit **4** incorrect (4, 5, 6, 7, 12, 13, 14, 15, 20, 21 contains seven 1s)
- Parity bit **8** incorrect (8, 9, 10, 11, 12, 13, 14, 15 contain three 1s)
- Parity bit **16** correct (16, 17, 18, 19, 20, 21 contain four 1s)

i) There is an error
ii) The error is at bit number= 2 + 4 + 8 = 14
iii) Correct codewords is 00101110000010**1**101110