The Open University of Sri Lanka
Department of Electrical and Computer Engineering
Bachelor of Software Engineering Honors
EEX3373 Communication and Computer Technology

862

Academic Year: 2021/2022
Examination: Final Examination
Duration: 3 hours.
Date: 22nd November 2022

Answer all questions.

Instructions:

- Read all instructions carefully before answering the questions.
- This question paper has four questions in four pages.
- This is a no-book type examination.
- Do not use red colour pens for answering questions.
- You can use non-programmable calculators.

Q1.

I.

a) Explain the function of an ALU.
b) Describe the concept of Accumulator based computer architecture with a block diagram.

(8 Marks)

II. Explain the execution of the AND operation in a microprocessor with the help of diagrams (consider accumulator based computer architecture).

(6 Marks)

III. Consider an array of twenty 8-bit positive integer numbers (Arr[20]). You have to get the average of each consecutive four numbers. The first four numbers start from Arr[0], the second four from Arr[4], the third four from Arr[8], and so forth. The resulting average numbers must be stored in a separate array Avg[ ]. Further, you must find the maximum number in the array Arr[ ] and store it in a different memory location.

Draw a single flowchart to show the algorithm of the assembly program (accumulator-based computer architecture) to perform the above task. Clearly state any assumptions you made.

Important: Use the standard symbols/ shapes to draw the flowchart.

(16 Marks)

IV. Write an assembly program for the above algorithm (Q1, III). Consider the ISA given in **Appendix A** to write the program. Assume that the array Arr[ ] data are stored from the memory address 70_{16}. For the array Avg[ ], you may select an appropriate memory location.

(10 Marks)

Q2.

I. Convert the following decimal numbers to binary. Show the intermediate calculation steps.
   a) 127.23
   b) 23.73

(2 Marks)

II. Convert the following binary numbers to decimal. Show the intermediate calculation steps.
   a) 1011011
   b) 10111.0101

(3 Marks)

III. Perform the following operations.
   a) 1011.0101 / 1101
   b) Use the 2's complement method and do the following calculation in binary.
      21 -- 86 =

(5 Marks)

IV. Convert the following numbers to IEEE 754 Single precision format
   a) 2154.251
   b) -317.67

(10 Marks)

Q3.

I. The Carrier signal and modulating signal are given as $e_c(t) = E_c \cos 2\pi f_c t$ &, $e_m(t) = E_m \cos 2\pi f_m t$.

   a) Write the standard expression of the Amplitude Modulated (AM) signal e(t).
   b) What is the envelope of the Amplitude Modulated (AM) signal $e(t)$
   c) Discuss how the retrieval of the original transmitted signal can be achieved using the signal parameter in amplitude modulation?

(8 Marks)

II. State the main types of modulation and explain what type of modulating signal is used in each main type of modulation. In addition to that, explain how this main type of modulations can be divided further?

(5 Marks)

III. The frequency modulated signal is given as,

$$V_m(t) = 14\cos(8x10^8 t + 10\sin(710t)\cos(710t))$$

Find the following

   a) Modulating Frequency
   b) Carrier frequency

c) Modulation Index
d) Frequency deviation

(7 Marks)

Q4.

I. Explain the main functionality and data used in each layer of OSI reference model using a table.

(4 Marks)

II. Briefly explain the CSMA/CD

(2 Marks)

III. How do you know whether two IP addresses are on the same subnet? Explain using an example.

(4 Marks)

IV. ABC company has 6 branches across different areas. Number of hosts in a branch is given in brackets. Colombo (450), Matara (260), Kandy (120), Kuliyapitiya (18), Bandarawela (60), Matale (20). Ip address of the main IP block is 172.110.132.98/18

a) Find the network address of main IP block.
b) What is the maximum number of usable hosts can accommodate in main IP block without subnetting?
c) Perform subnetting the given network for the branches.

Calculate network address, subnet mask, starting host address, ending host address and the broadcast address of each subnet in the part iii). State your assumptions.

(10 Marks)

Accumulator Architecture

Mnemonic	Syntax	Operation	Description	Add Mode	Flags affected
Arithmetic instructions					
ADD	ADD AM x	Addition	Acc ← Acc + op	#,&,default	ZF, OF, SF, PF =1
SUB	SUB AM x	Subtraction	Acc ← Acc - op	#,&,default	ZF, OF, SF, CF, PF =1
MUL	MUL AM x	Signed multiplication	Acc (16 bit) ← Acc(8 LSBs) * op(8 LSBs)	#,&,default	ZF, SF, PF =1. OF, CF=0
DIV	DIV AM x	Unsigned division	Acc ← Acc / op2	#,&,default	ZF, SF, PF =1. OF, CF =0
INC	INC	Increment by 1	Acc ← Acc + 1	#,&,default	OF, SF, ZF, PF=1
Logical instructions					
AND	AND AM x	Bit-wise And	Acc ← Acc AND op	#,&,default	OF, CF=0; SF, ZF, PF =1
OR	OR AM x	Bit-wise OR	Acc ← Acc OR op	#,&,default	OF, CF =0; SF, ZF, PF =1
XOR	XOR AM x	Bit-wise XOR	Acc ← Acc XOR op	#,&,default	OF, CF=0; SF, ZF, PF =1
SHL	SHL	Shift left by 1-bit	CF ← Acc (MSB), op ← Acc (14 down to 0) & 0	Implied (A)	CF= bit shifted by Acc. SF, ZF, PF=1.
SHR	SHR	Shift right by 1-bit	CF ← Acc (LSB), Acc ← 0 & Acc (15 down to 1)	Implied (A)	CF= bit shifted by Acc. SF, ZF, PF=1.
ROL	ROL	rotate left by 1-bit	CF ← Acc (MSB), op ← Acc (14 down to 0) & CF	Implied (A)	CF= bit shifted by Acc. SF, ZF, PF=1.
ROR	ROR	rotate right by 1-bit	CF ← Acc (LSB), Acc ← CF & Acc (15 down to 1)	Implied (A)	CF= bit shifted by Acc. SF, ZF, PF=1.
NOT	NOT	One's compliment negation	Acc ← NOT Acc	Implied (A)	none
Control Transfer instructions					
Conditional Branches					
JC	JC d	Jump if carry	If CF =1 then PC ← PC + Operand	PC relative	none
JOF	JOF d	Jump if over-flow	If OF =1 then PC ← PC + Operand	PC relative	none
JS	JS d	Jump if Sign	If SF =1 then PC ← PC + Operand	PC relative	none
JP	JP d	Jump if parity	If PF =1 then PC ← PC + Operand	PC relative	none
JZ	JZ d	Jump if result is zero	If ZF =1 then PC ← PC + Operand	PC relative	none
Unconditional branch					
JUMP	JUMP d	Jump	PC ← PC + Operand	PC relative	none
Loops					
LOOZ	LOOZ d	Loop until zero	Count ← Count – 1, IF Count =0; Loop termination ELSE; PC ← PC + operand	PC relative	none
Calls and Returns					
CALL	CALL d	Procedure call	implied return address ← PC, PC ← immediate address	*	none
RETURN	RETURN	Return from procedure	PC ← Contents of implied return address	**	none
Miscellaneous instructions					
NOP	NOP	No operation		none	none
Data Movement instructions					
LOADacc	LOADacc AM d	Copy the operand to the accumulator	Immediate: Acc ← op; Direct: Acc ← memory (op)	#, default	none
STOREacc	STOREacc AM d	Copies the accumulator to the memory address	Direct: Memory (op) ← Acc Indirect: Memory (memory (op)) ← Acc	default, &	none

*Immediate operand is used as the jumping location.
**Contents of the implied return address is used as the jumping location
x- Memory Address or immediate value, d- Displacement

Addressing Modes -AM

Immediate	#
Direct	default (blank)
Indirect	&