Final Exam

Computer Architecture

Computer and Syst. Dept.
Time Allowed: 3 Hrs.
3rd Year Students.
Total Marks: 100
2012 – 2013

Solve the following Questions:

- يسمح باستخدام القلم الرصاص (شرط وضوح الخط).
- الرجاء وضوح الرسم قدر المستطاع (ليس شرطا استخدام المسطرة)
- الامتحان في ورقتين.

(1) What is the purpose of the BUS system, Draw the general structure of m-bit Bus, and then show (in details) the structure of 6-bit Bus using 4 registers each of 6-bit size. **(6 marks)**

(2) For the basic computer, derive the control logic gates for:

- SC (clear and count).
- Flip flop E.

(8 marks)

(3) Consider the following memory contents:

Address	Contents
20	7040
21	A222
22	7020
23	7002
24	4026
25	7001
26	1555
27	7200
28	C557
29	4025
30	7008
31	7200
222	0556
555	0002
556	1234
557	0029

(10 marks)

Where: PC=20, AC=A334, E=0. Show the contents of **AC** and **E** when the program stops.

(4) In the basic computer, assuming M[7FF]=6A9F, M[A9F]=FFFF, M[800]=7800, M[801]=7400. The content of PC now is 7FFand the contents of AC is 76AB. Show the contents of PC, AR, DR, IR, SC, AC at each timing signal of the execution of the next two instructions (describing the fetch-decode- and execute for each instruction). **(7 marks)**

(5) For the basic computer, write the required steps for the following instructions, starting from T_4, the adder and logic can perform the XOR, but cannot perform subtraction directly. Note:

- AC must not change unless the instruction specifies a change in its value.
- Multiplication by 2 is a shift left by zero الضرب في 2 هو إزاحة ناحية اليسار بصفر.

(12 marks)

Symbol	OpCode	Function
AAA	000	M[AR] ← AC + 4*M[AR]
BBB	001	if (AC=$\overline{M[AR]}$) Then PC ← AR
CCC	010	M[AR] ← 0
DDD	011	If (AC>M[AR]) then PC ← PC+1

اقلب الورقة

(6) For the basic computer:

(a) Write the interrupt service routine.
(b) Write the assembly program to calculate: C(i)=A(i)+100*B(i)-2*M, the three arrays A, B, and C start at addresses; 500, 700, 900 respectively with length=100 elements.
(c) Write a program to fill an array (starting at address 500 and length=100 words) with characters (each word can store two characters).

(12 marks)

(7) Write the equivalent assembly code for:

```
If (A>B<C)
F=A+B+C;
Else if (A=B>C)
F=A-2B-C;
Else
F=0;
```

(5 marks)

(8) Use assembly language (for the basic computer) to write **subroutines** to:

(a) Calculate the value of the following **logic** operation: F=X'Z'Y + Y'Z + M.
(b) Subtract the value M to each element of an array starting from address 500 with length 100. The value of M should be passed to the subroutine from the main program.
(c) Multiply two numbers by repeated addition, the two numbers should be passed from the main program to the subroutine (**for example, to calculate 5X4, the numbers 5 and 4 are passed to the subroutine**).

(12 marks)

(9) Compare between hardwired and micro-programmed control units, when to use each type of them?

(5 marks)

(10) Write the symbolic micro-program for the:

- Fetch Routine (starts at address 64).
- Indirect Subroutine (start at address 67).

(3 marks)

(11) For the basic computer, write the symbolic micro-program (Routine) for the following instructions (Note: AC must not change unless the instruction specifies a change in its value, also the Fetch routine and Indirect subroutine starts at 64 and 67 respectively).

Symbol	OpCode	Function
AAA	0100	M[AR] ← AC + 4*M[AR]
BBB	0101	if (AC=$\overline{M[AR]}$) Then PC ← AR else PC ← PC+1
CCC	0110	M[AR] ← 0
DDD	0111	If (AC>M[AR]) then PC ← PC+1 **(Note: here AC should be greater than and not equal M[AR])**
EEE	1000	AC← $\overline{(AC/2 \wedge M[AR])}$

(15 marks)

---------- End Of Questions ----------

With Best Wishes
Dr: Ahmed Saleh

PLZ, send your comments about the exam to:
aisaleh@yahoo.com

TABLE 7-1 Symbols and Binary Code for Microinstruction Fields

F1	Microoperation	Symbol
000	None	NOP
001	$AC \leftarrow AC + DR$	ADD
010	$AC \leftarrow 0$	CLRAC
011	$AC \leftarrow AC + 1$	INCAC
100	$AC \leftarrow DR$	DRTAC
101	$AR \leftarrow DR(0\text{–}10)$	DRTAR
110	$AR \leftarrow PC$	PCTAR
111	$M[AR] \leftarrow DR$	WRITE

F2	Microoperation	Symbol
000	None	NOP
001	$AC \leftarrow AC - DR$	SUB
010	$AC \leftarrow AC \vee DR$	OR
011	$AC \leftarrow AC \wedge DR$	AND
100	$DR \leftarrow M[AR]$	READ
101	$DR \leftarrow AC$	ACTDR
110	$DR \leftarrow DR + 1$	INCDR
111	$DR(0\text{-}10) \leftarrow PC$	PCTDR

F3	Microoperation	Symbol
000	None	NOP
001	$AC \leftarrow AC \oplus DR$	XOR
010	$AC \leftarrow \overline{AC}$	COM
011	$AC \leftarrow$ shl AC	SHL
100	$AC \leftarrow$ shr AC	SHR
101	$PC \leftarrow PC + 1$	INCPC
110	$PC \leftarrow AR$	ARTPC
111	Reserved	

CD	Condition	Symbol	Comments
00	Always = 1	U	Unconditional branch
01	$DR(15)$	I	Indirect address bit
10	$AC(15)$	S	Sign bit of AC
11	$AC = 0$	Z	Zero value in AC

BR	Symbol	Function
00	JMP	$CAR \leftarrow AD$ if condition = 1
		$CAR \leftarrow CAR + 1$ if condition = 0
01	CALL	$CAR \leftarrow AD$, $SBR \leftarrow CAR + 1$ if condition = 1
		$CAR \leftarrow CAR + 1$ if condition = 0
10	RET	$CAR \leftarrow SBR$ (Return from subroutine)
11	MAP	$CAR(2\text{–}5) \leftarrow DR(11\text{–}14)$, $CAR(0,1,6) \leftarrow 0$

Control Functions and Microoperations for the Basic Computer

Fetch	$R'T_0$:	$AR \leftarrow PC$
	$R'T_1$:	$IR \leftarrow M[AR]$, $PC \leftarrow PC + 1$
Decode	$R'T_2$:	$D_0, \ldots, D_7 \leftarrow$ Decode $IR(12\text{–}14)$
		$AR \leftarrow IR(0\text{–}11)$, $I \leftarrow IR(15)$
Indirect	D_7IT_3:	$AR \leftarrow M[AR]$
Interrupt:		
	$T_0\overline{T}_1\overline{T}_2(IEN)(FGI + FGO)$:	$R \leftarrow 1$
	RT_0:	$AR \leftarrow 0$, $TR \leftarrow PC$
	RT_1:	$M[AR] \leftarrow TR$, $PC \leftarrow 0$
	RT_2:	$PC \leftarrow PC + 1$, $IEN \leftarrow 0$, $R \leftarrow 0$, $SC \leftarrow 0$
Memory-reference:		
AND	D_0T_4:	$DR \leftarrow M[AR]$
	D_0T_5:	$AC \leftarrow AC \wedge DR$, $SC \leftarrow 0$
ADD	D_1T_4:	$DR \leftarrow M[AR]$
	D_1T_5:	$AC \leftarrow AC + DR$, $E \leftarrow C_{out}$, $SC \leftarrow 0$
LDA	D_2T_4:	$DR \leftarrow M[AR]$
	D_2T_5:	$AC \leftarrow DR$, $SC \leftarrow 0$
STA	D_3T_4:	$M[AR] \leftarrow AC$, $SC \leftarrow 0$
BUN	D_4T_4:	$PC \leftarrow AR$, $SC \leftarrow 0$
BSA	D_5T_4:	$M[AR] \leftarrow PC$, $AR \leftarrow AR + 1$
	D_5T_5:	$PC \leftarrow AR$, $SC \leftarrow 0$
ISZ	D_6T_4:	$DR \leftarrow M[AR]$
	D_6T_5:	$DR \leftarrow DR + 1$
	D_6T_6:	$M[AR] \leftarrow DR$, if $(DR = 0)$ then $(PC \leftarrow PC + 1)$, $SC \leftarrow 0$
Register-reference:		
		$D_7I'T_3 = r$ (common to all register-reference instructions)
		$IR(i) = B_i$ $(i = 0, 1, 2, \ldots, 11)$
	r:	$SC \leftarrow 0$
CLA	rB_{11}:	$AC \leftarrow 0$
CLE	rB_{10}:	$E \leftarrow 0$
CMA	rB_9:	$AC \leftarrow \overline{AC}$
CME	rB_8:	$E \leftarrow \overline{E}$
CIR	rB_7:	$AC \leftarrow$ shr AC, $AC(15) \leftarrow E$, $E \leftarrow AC(0)$
CIL	rB_6:	$AC \leftarrow$ shl AC, $AC(0) \leftarrow E$, $E \leftarrow AC(15)$
INC	rB_5:	$AC \leftarrow AC + 1$
SPA	rB_4:	If $(AC(15) = 0)$ then $(PC \leftarrow PC + 1)$
SNA	rB_3:	If $(AC(15) = 1)$ then $(PC \leftarrow PC + 1)$
SZA	rB_2:	If $(AC = 0)$ then $PC \leftarrow PC + 1$
SZE	rB_1:	If $(E = 0)$ then $(PC \leftarrow PC + 1)$
HLT	rB_0:	$S \leftarrow 0$
Input-output:		
		$D_7IT_3 = p$ (common to all input–output instructions)
		$IR(i) = B_i$ $(i = 6, 7, 8, 9, 10, 11)$
	p:	$SC \leftarrow 0$
INP	pB_{11}:	$AC(0\text{–}7) \leftarrow INPR$, $FGI \leftarrow 0$
OUT	pB_{10}:	$OUTR \leftarrow AC(0\text{–}7)$, $FGO \leftarrow 0$
SKI	pB_9:	If $(FGI = 1)$ then $(PC \leftarrow PC + 1)$
SKO	pB_8:	If $(FGO = 1)$ then $(PC \leftarrow PC + 1)$
ION	pB_7:	$IEN \leftarrow 1$
IOF	pB_6:	$IEN \leftarrow 0$