



Quiz 4 / Fall 2014

Odd

Q1. (10 points) The second column of the following table contains a list of instructions in machine code and related operands that are stored in the memory unit of the basic computer starting with memory location at address (B00)₁₆. All numbers listed below are in hexadecimal.

- Convert this program from Machine Language to Assembly Language, using the annexed instruction list and write the corresponding symbolic codes of these instructions in the 3rd column of the table (“Assembly Language”)
- The computer executes the instructions starting from address B00. Show the content of the AC (in hexadecimal) after each instruction is executed.
- What is the content of the memory word at address B08 when the computer halts?
- Describe in a sentence what is this program doing?

Memory Address	Machine Code	Assembly Language (a = 6 pts)	AC (b = 4.5)
B00	2B06	LDA B06 (1 pt)	(1 pt) 0013 ₁₆ =19 ₁₀
B01	1B07	ADD B07 (1 pt)	(1 pt) 002A ₁₆ =42 ₁₀
B02	7400	CLE (1 pt)	(0.5 pt) 002A ₁₆ =42 ₁₀
B03	7080	CIR (1 pt)	(1 pt) 0015 ₁₆ =21 ₁₀
B04	3B08	STA B08 (1 pt)	(0.5 pt) 0015 ₁₆ =21 ₁₀
B05	7001	HLT (1 pt)	(0.5 pt) 0015 ₁₆ =21 ₁₀
B06	0013	0013 ₁₆ =19	
B07	0017	0017 ₁₆ =23	
B08	0000	0015 (c) (1.5 pt)	

(d) The program calculates the average of 2 numbers stored at addresses B06 and B07 and stores the result at B08. (2 pts – equivalent statements would be OK; give partial marks)

Bonus: 1 pt. for comments or explanations

Q2. (10 points) Write a program loop in assembly language, using a pointer and a counter, which writes $(3333)_{16}$ to 128 memory locations starting at address F00. Your program should be stored in the memory of the basic computer starting with address AA0.

NOTE: All numbers in the above text are in hexadecimal.

Solution:

⇒ The counter CTR should be initialised with $-128_{10} = FF80_{16}$

Assembly Language	Machine Language			10
ORG AA0	Addr	.Code		.1.
LDA ADS	AA0	2AAA		1
STA PTR	AA1	3AAB	/Initialize pointer to the first memory location to be modified	
LDA NBR	AA2	2AAC		1
STA CTR	AA3	3AAD	/Initialize counter to -128	
LDA PAT	AA4	2AAE	/Prepare pattern 3333 in Accumulator to be written to the memory	1
LOP, STA PTR I	AA5	BAAB	/Load pattern 3333 to the memory location indicated by PTR	1
ISZ PTR	AA6	6AAB		1
ISZ CTR	AA7	6AAD	/ Loop mechanism	2
BUN LOP	AA8	4AA5		
HLT	AA9	7001		1
ADS, HEX F00	AAA	0F00	/ 1 st memory location to be written with the PAT pattern	
PTR, HEX 0	AAB	0000	/will be used as pointer to the current memory location to be written	
NBR, DEC -128	AAC	FF80	/or HEX FF80 = number of loops to be performed	
CTR, HEX 0	AAD	0000	/ will be used as counter of loops until => 0	
PAT, HEX 3333	AAE	3333	/ the pattern to be written in the memory	
END				

1 pt for comments

1pt BONUS for good comments

Accept any of the Assembly Language or the Machine Language, or both.

Symbol	Hex code	Description
AND	0 or 8	AND M to AC
ADD	1 or 9	Add M to AC, carry to E
LDA	2 or A	Load AC from M
STA	3 or B	Store AC in M
BUN	4 or C	Branch unconditionally to m
BSA	5 or D	Save return address in m and branch to m + 1
ISZ	6 or E	Increment M and skip if zero
CLA	7800	Clear AC
CLE	7400	Clear E
CMA	7200	Complement AC
CME	7100	Complement E
CIR	7080	Circulate right E and AC
CIL	7040	Circulate left E and AC
INC	7020	Increment AC
SPA	7010	Skip if AC is positive
SNA	7008	Skip if AC is negative
SZA	7004	Skip if AC is zero
SZE	7002	Skip if E is zero
HLT	7001	Halt computer
INP	F800	Input information and clear flag
OUT	F400	Output information & clear flag
SKI	F200	Skip if input flag is on
SKO	F200	Skip if output flag is on
ION	F080	Turn interrupt on
IOF	F040	Turn interrupt off

