

CS3280 Homework 2; Due: Friday, February 10, at beginning of class

1. (30 points) For each of the following questions, assume that the registers are initialized to the following values (i.e., each part is independent of the others). Note: NZVC are the condition codes and ML is short for memory location. Give the contents of the registers after each instruction is executed. All numbers shown are HEX numbers except when noted otherwise. DO NOT leave any box empty (even if the contents don't change) - empty boxes will be counted wrong! Also, do not use a DASH "-" in a box and do not forget the \$-sign for hex numbers.

A. STAA \$A0

	PC	A	X	Y	ML(\$A000)	ML(\$A001)	NZVC
Initial	\$C000	\$80	\$9FFF	\$A000	\$3C	\$CC	0111
After	\$C002	\$80	\$9FFF	\$A000	\$3C	\$CC	1001

B. ADDA 2,X

	PC	A	X	Y	ML(\$A000)	ML(\$A001)	NZVC
Initial	\$C000	\$80	\$9FFF	\$A000	\$3C	\$CC	0111
After	\$C002	\$4C	\$9FFF	\$A000	\$3C	\$CC	0011

C. SUBA 1,Y

	PC	A	X	Y	ML(\$A000)	ML(\$A001)	NZVC
Initial	\$C000	\$80	\$9FFF	\$A000	\$3C	\$CC	0111
After	\$C003	\$B4	\$9FFF	\$A000	\$3C	\$CC	1001

D. STX 1,X

	PC	A	X	Y	ML(\$A000)	ML(\$A001)	NZVC
Initial	\$C000	\$80	\$9FFF	\$A000	\$3C	\$CC	0111
After	\$C002	\$80	\$9FFF	\$A000	\$9F	\$FF	1001

E. BLT -4 (-4 is the decimal branch offset)

	PC	A	X	Y	ML(\$A000)	ML(\$A001)	NZVC
Initial	\$C000	\$80	\$9FFF	\$A000	\$3C	\$CC	0111
After	\$BFFE	\$80	\$9FFF	\$A000	\$3C	\$CC	0111

2(A.) (15 points) Assemble the following code segment (use Appendix A1 in the book). Note that some rows in the machine code column may contain more than one byte, and some rows may have no machine code. Specify the address for each instruction op-code in the LOCATION column. Do not forget the \$-sign for hex numbers. Note: 'ORG \$C000' is an assembler directive that sets the address for the opcode of the instruction immediately following the ORG statement to \$C000. Assembler directives do not produce any machine code.

2(B.) (5 points) Indicate the addressing mode of each instruction in the appropriate column.

ADDRESS MODE	LOCATION	MACHINE CODE			
				ORG	\$C000
DIR	\$C000	\$DE, \$20		LDX	\$20
DIR	\$C002	\$18, \$DE, \$22		LDY	\$22
INDEX	\$C005	\$E6, \$02	LOOP	LDAB	2, X
INDEX	\$C007	\$18, \$A6, 1		LDAA	1, Y
INHER	\$C00A	\$10		SBA	
RELATIVE	\$C00B	\$2B, \$F8		BMI	LOOP

2(C.) (20 points) Hand execute the above program assuming the initial processor conditions below. In each row, please fill in the contents of the registers/memory cell **AFTER** the execution of the corresponding instruction. Assume the following memory model. Do not forget the \$-sign for hex numbers.

LOCATION	DATA
\$0020	\$00
\$0021	\$22
\$0022	\$00
\$0023	\$20
\$0024	\$55

STEP #	PC	X	Y	A	B	NZVC
Before exec.	\$C000	\$FFFF	\$FFFF	00	00	0000
1	\$C002	\$0022	\$FFFF	00	00	0000
2	\$C005	\$0022	\$0020	00	00	0000
3	\$C007	\$0022	\$0020	00	\$55	0000
4	\$C00A	\$0022	\$0020	\$22	\$55	0000
5	\$C00B	\$0022	\$0020	\$CD	\$55	1001
6	\$C005	\$0022	\$0020	\$CD	\$55	1001

3(A.) (15 points) Assume that memory beginning at location \$C000 contains the contents of the first column of the table below. It is the machine code of a program. In the space provided, disassemble this program (use Appendix A2 in the book), i.e., determine the complete instructions (**including operands; e.g., LDAA \$1234**) to be executed (do not try to make sense of the program as it does not do anything useful). Don't forget to put # and \$-signs in the appropriate places. Note: not all rows will require entries - only one entry per instruction.

If the program includes a branch, please put the label 'LABEL' in front of the branch destination instruction and after the branch mnemonic.

3(B.) (5 points) Fill in the address mode next to each source code instruction. Note: not all rows of this table require entries.

Contents	Instruction	Address Mode
\$96	LDAA \$20	DIR
\$20		
\$C6	LDAB #\$10	IMMED
\$10		
\$FE	LABEL LDX \$1234	EXTENDED
\$12		
\$34		
\$EF	STX 1,X	IND X
\$01		
\$08	INX	INHER
\$23	BLS LABEL	RELAT
\$F8		