5.0 Preliminary Assignment

1. Completely read all seven sample programs to familiarize yourself with their objectives.

2. Review the 68000’s instructions and addressing modes. The user should read both the “Data Organization and Address Capabilities” section and the “Instruction Set Summary” section in the M68000 Programmer’s Reference Manual.

3. Review the TUTOR Command Set, which is described in the MC68000 Educational Computer Board User’s Manual.

4. Procedure #8 of sample Program 2.6 requires that the student write a short MC68000 Assembly Language program. This student should have this program written before coming to the lab.

\*-----------------------------------------------------------

\* Title : Sorted List

\* Written by : Theo Guidroz

\* Date : 01/29/2020

\* Description: Adding a number to a sorted list

\*-----------------------------------------------------------

ORG $2000

\* Messages and Data

PROMPT: DC.B 'PLEASE INPUT THE NUMBER: '

DC.B 0

START:

MOVEA.L #PROMPT,A1 ; Move address for PROMPT into A1

MOVE.B #13,D0 ; Move #13 into D0 to display the prompt

TRAP #14 ; Execute TRAP

MOVEA.L #$500,A1 ; Test value

MOVEA.L #$700,A0 ; Test value

CLR.W D3 ; Clear D3 to store the word read from keyboard input

MOVE.B #2,D0 ; Move #4 into D0 to read keyboard input

TRAP #14 ; Execute TRAP

ATOH MOVEM.L D1/A0-A1,-(SP) ; Save the original values of D1, A0 and A1 by putting them on stack

ALGO MOVE.B (A1)+,D2 ; Move the first character into D2

CMPI.B #$39,D2 ; Test to see if the character entered is a hex letter (Letters have ASCII values >39)

BGT SEV ; Branch to SEV

ALGO2 ANDI.B #$0F,D2 ; The second character corresponds to the actual hex value, so you must AND it with 0x0F to get only the second character

ADD.W D2,D3 ; Move the character onto data register D3

COUNT SUBI.B #1,D1 ; Subtract 1 from the counter

BNE BIT ; If more characters are left, branch to BIT

BRA END ; Else, branch to END

BIT LSL.W #4,D3 ; Shift the bit to the next position to its left

BRA ALGO ; Branch to ALGO

SEV SUBI.B #$7,D2 ; Subtract 7, because there are 7 characters between 9 and A in ASCII table

BRA ALGO2 ; To get all the possible hex characters next to each other. (30 to 3F). Branch to ALGO2

END MOVE.W D3,-(A0) ; Store word found in D3 to list

MOVEM.L (SP)+,D1/A0-A1 ; Restore registers

BRA SORT ; Go to sorting algorithm

ORG $3000

\* Code below was provided in the lab manual

SORT CMP.W (A0),D2

BCC $300C

MOVE.W (A0),-(A0)

ADDQ #4,A0

CMPA.L A0,A1

BCC $3000

MOVE.W D0,-(A0)

MOVE.B #228,D7

TRAP #14

SIMHALT ; halt simulator

\* Put variables and constants here

END START ; last line of source