CSCI 592

LAB ASSIGNMENT – 2

Written by

DINESH SEVETI

Date: 02-08-2025

**OBJECTIVE**

Rewrite the copy program in Lab 1.C so that it will create your name beginning at memory location $02100 and social security number beginning at location $02160.

**TECHNOLOGY USED**

* Easy68K Assembler software to run the code.

**PROCEDURE**

* Initialize memory locations with uppercase letters, lowercase letters, and digits.
* Load these memory locations into registers A2, A3, and A4 using LEA instructions.
* Set up the destination address in A1 where the extracted characters will be stored.
* Extract and store characters sequentially using MOVE.B with different addressing modes.
* Store the full name "Dinesh Seveti" at $02100.
* Store the SSN "123-45-6789" at $02160.
* Halt the program execution with the SIMHALT instruction.

**OPERATIONS**

* Load Effective Address (LEA)
* Move Byte (MOVE.B)
* Immediate Data Handling (32, #45 for spaces and dashes)
* Indexed Addressing (Offset calculation from base address)
* Memory Storage and Manipulation
* Address Register Direct (An)
* Address Register Indirect (An)
* Post-increment Addressing (An)+
* Pre-decrement Addressing -(An)
* Displacement Addressing d(An)

**ALGORITHM**

* Define the memory locations containing uppercase letters, lowercase letters, and digits.
* Load the base addresses into registers (A2, A3, A4).
* Set the destination addresses for storing the name and SSN in A1.
* Extract required characters using indexed addressing and store them sequentially at the desired locations.
* Use immediate values to insert spaces and dashes where required.
* Halt the execution after storing the values.

**CODE LISTING**

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

\* Title :Lab Assignment 2

\* Written by :DINESH SEVETI

\* Date :02/08/2025

\* Description:The Program prints my name and the SSN number

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

ORG $2000

DC.B 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'

ORG $2020

DC.B 'abcdefghijklmnopqrstuvwxyz'

ORG $2040

DC.B '0123456789'

START:

LEA.L $002000,A2 ; Load uppercase letters

LEA.L $002020,A3 ; Load lowercase letters

LEA.L $002040,A4 ; Load digits

LEA.L $002100,A1 ; Destination for name

; Store "Dinesh Seveti" at $02100

MOVE.B 3(A2),(A1)+ ; D

MOVE.B 8(A3),(A1)+ ; i

MOVE.B 13(A3),(A1)+ ; n

MOVE.B 4(A3),(A1)+ ; e

MOVE.B 18(A3),(A1)+ ; s

MOVE.B 7(A3),(A1)+ ; h

MOVE.B #32,(A1)+ ; Space

MOVE.B 18(A2),(A1)+ ; S

MOVE.B 4(A3),(A1)+ ; e

MOVE.B 21(A3),(A1)+ ; v

MOVE.B 4(A3),(A1)+ ; e

MOVE.B 19(A3),(A1)+ ; t

MOVE.B 8(A3),(A1)+ ; i

LEA.L $002160,A1 ; Destination for SSN

; Store "123-45-6789" at $02160

MOVE.B (A4),(A1)+ ; 1

MOVE.B 1(A4),(A1)+ ; 2

MOVE.B 2(A4),(A1)+ ; 3

MOVE.B #45,(A1)+ ; -

MOVE.B 4(A4),(A1)+ ; 4

MOVE.B 5(A4),(A1)+ ; 5

MOVE.B #45,(A1)+ ; -

MOVE.B 6(A4),(A1)+ ; 6

MOVE.B 7(A4),(A1)+ ; 7

MOVE.B 8(A4),(A1)+ ; 8

MOVE.B 9(A4),(A1)+ ; 9

SIMHALT ; halt simulator

END START ; last line of source

**DESCRIPTION**

The program initializes three memory locations containing upper-case alphabets, lowercase alphabets, and numeric digits. It then loads these memory locations into registers and systematically extracts required characters to form the name "Dinesh Seveti" at memory location $02100 and the SSN "123-45-6789" at $02160. The extracted characters are stored using the MOVE.B instruction with indexed addressing. Finally, the program halts execution with the SIMHALT command.

**OBSERVATIONS**

* The program correctly stores the name "Dinesh Seveti" at a memory location $02100.
* The social security number "123-45-6789" is stored at $02160.
* Indexed addressing allows efficient character extraction and storage.
* Immediate values are used for spaces and dashes, making the format accurate.

**RESULTS**

**CONCLUSIONS**

The program effectively demonstrates memory manipulation using assembly language. By leveraging indexed addressing and immediate values, the program successfully constructs a name and an SSN in specified locations.