基礎ソフト 実験レポート1

C過程 S-15組 1TE20137W 2022/10/19

柳 鷹

問1

(1)

|  |
| --- |
| MOVE.W (%A0)+,(%A1)+ |

　 上記の68000のアセンブラプログラムは A0レジスタの中身が示すメモリアドレスの値をA1レジスタの中身が示すメモリアドレスに転送し、この命令実行後、A0レジスタ・A1レジスタそれぞれに（ワードサイズだから）2加算される。

例えば、はじめのA0レジスタの値が0x1000、A1レジスタの値が0x2000、0x1000番地の値が0x1234、0x２2000番地の値が0x0000のとき、0x2000番地に0x1234が転送され、その後A0レジスタ・A1レジスタそれぞれに2加算されてA0レジスタが0x1002に、A1レジスタに0x2002になる。

(2)

|  |
| --- |
| .equ TOP, 0xFFFC00  .equ MASK, TOP+0x80  MOVE.W #0x07F7, MASK |

上記の68000のアセンブラプログラムは、まず上2行でシンボルとしてTOPを0xFFFC00に、MASKをTOPに0x80を加えた値すなわち0xFFFC80に設定する。そして、「MOVE.W #0x07F7, MASK」でシンボルMASKを0x7F7（ワードサイズ）に再設定する。

(3)

|  |
| --- |
| .dc.b 'a','b','c','d','e',0 |

上記の68000のアセンブラプログラムは、データ列'a','b','c','d','e',0をバイトサイズでメモリ内に配置する。データが文字列の場合は、ASCIIコード（例えば、'a'なら0x61、'b'なら0x61）に変換される。

(4)

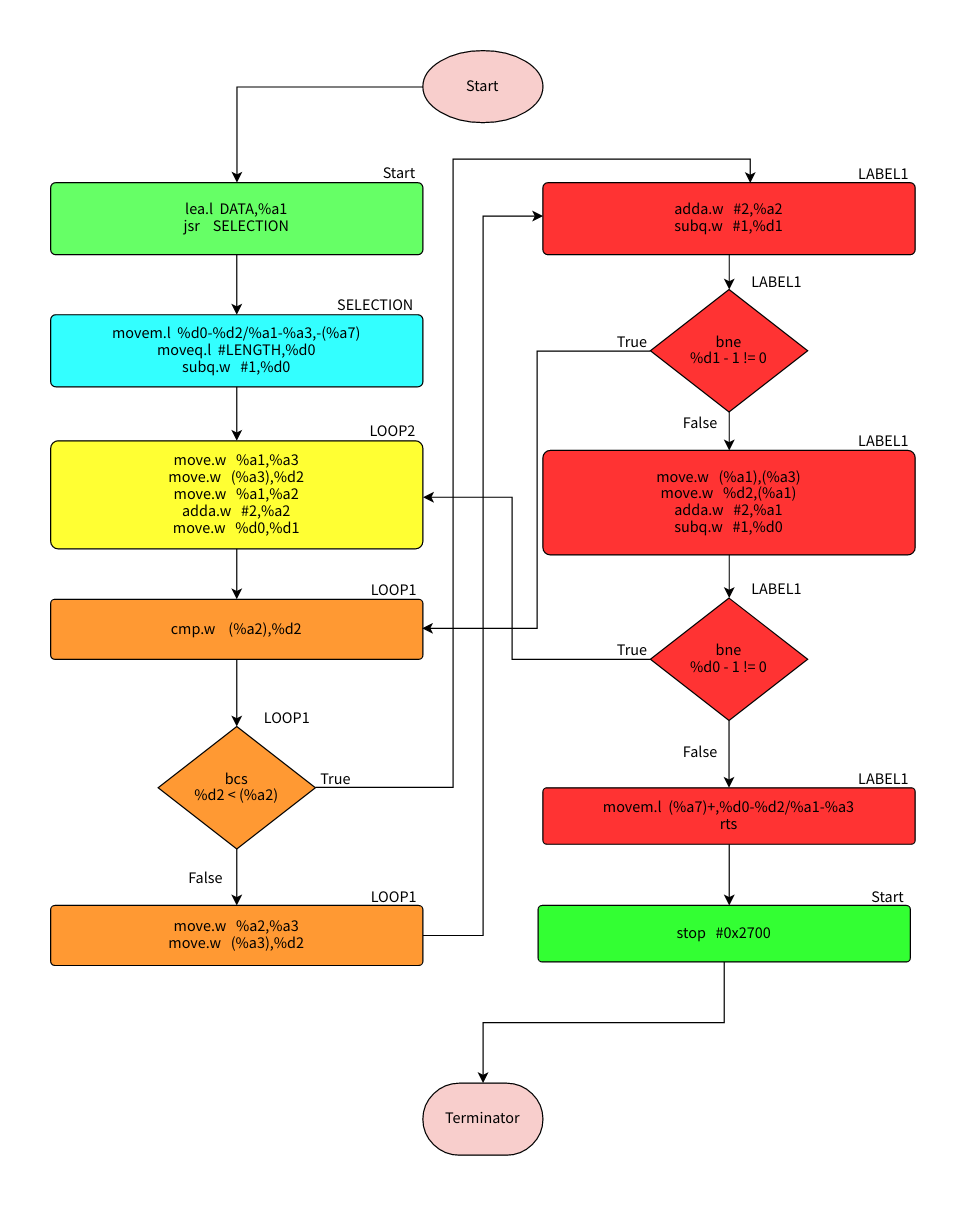
|  |
| --- |
| .equ BOTTOM, 4  MOVE.L %A2,BOTTOM(%A0) |

上記の68000のアセンブラプログラムは、「.equ BOTTOM, 4」でシンボルとしてBOTTOMを0xFFFC00に設定し、「MOVE.L %A2,BOTTOM(%A0) 」で[BOTTOMの中身の値]+[A0レジスタの中身]となるメモリ番地にA2レジスタの値を転送する。

問2

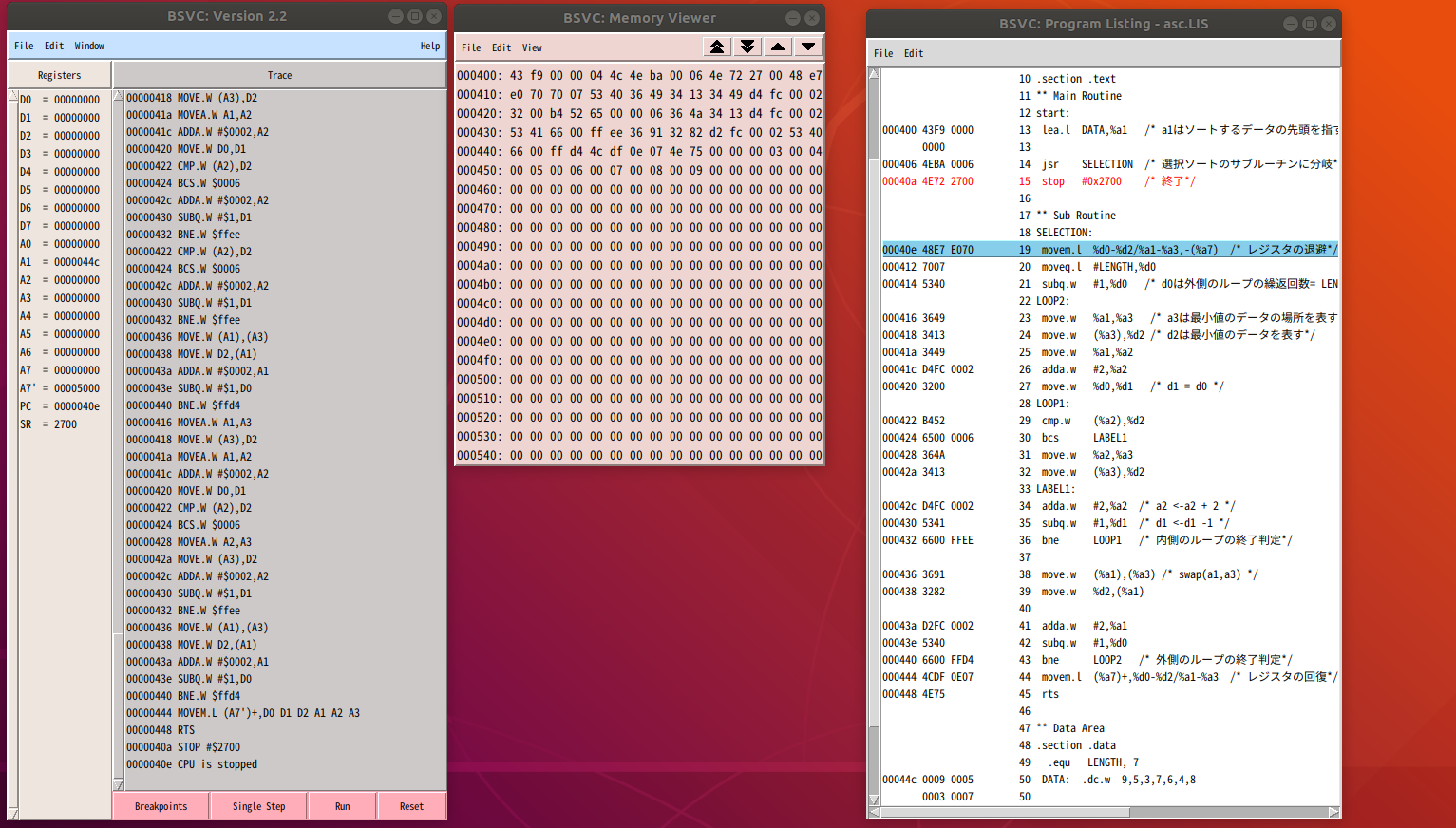
(1)

求めるフローチャートは以下である。



(2)

選択ソートのアセンブラプログラムの実行結果は以下である。



上の実行結果からこのプログラムが正しく動いていることを簡単に説明する。

まず、整列するデータ列の先頭アドレスを格納しているA1レジスタは0x00044cが入っており、メモリの0x00044c番地からワードサイズ（16進数4桁）でデータが昇順ソートされているのが確認できる。プログラムの実行前はメモリの0x00044c~0x00044dには0009が格納してあった。

(3)

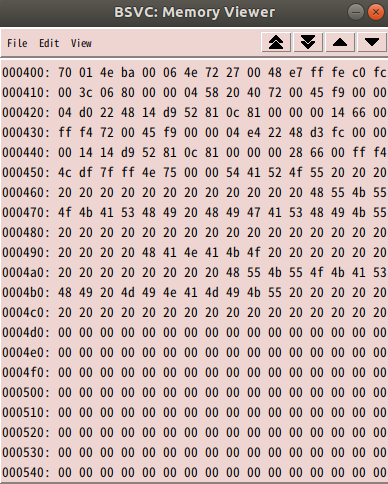
本項での感想・意見を述べる。

本実験までアセンブラプログラムというものにほとんど触れたことがなかった。ただ、最近パチンコやパチスロのメインプログラミング言語がアセンブリ言語であることを知った。それは規則でROMが最大16KBやRAMが最大1024Bとメモリの記憶容量が定められているからであった。そしてこの基礎ソフト実験でメモリを意識したアセンブリ言語を体験することによりその理由がよく分かった。本項の選択ソートを実装・実行して感じたのは、メモリ内のデータ配置を把握することができるということであった。アセンブラプログラムは遊技機のようにメモリが限られており、扱うデータ量がそこまで大きくないとき、ソートにかかる計算量よりもメモリの使用について深く考えられるため、適していると感じる。また、アルゴリズムの講義等で計算量について考察する場面はあったが、メモリ等についてあまり考えていなかったように感じたので、この機会に再度学習したいと思った。

問3

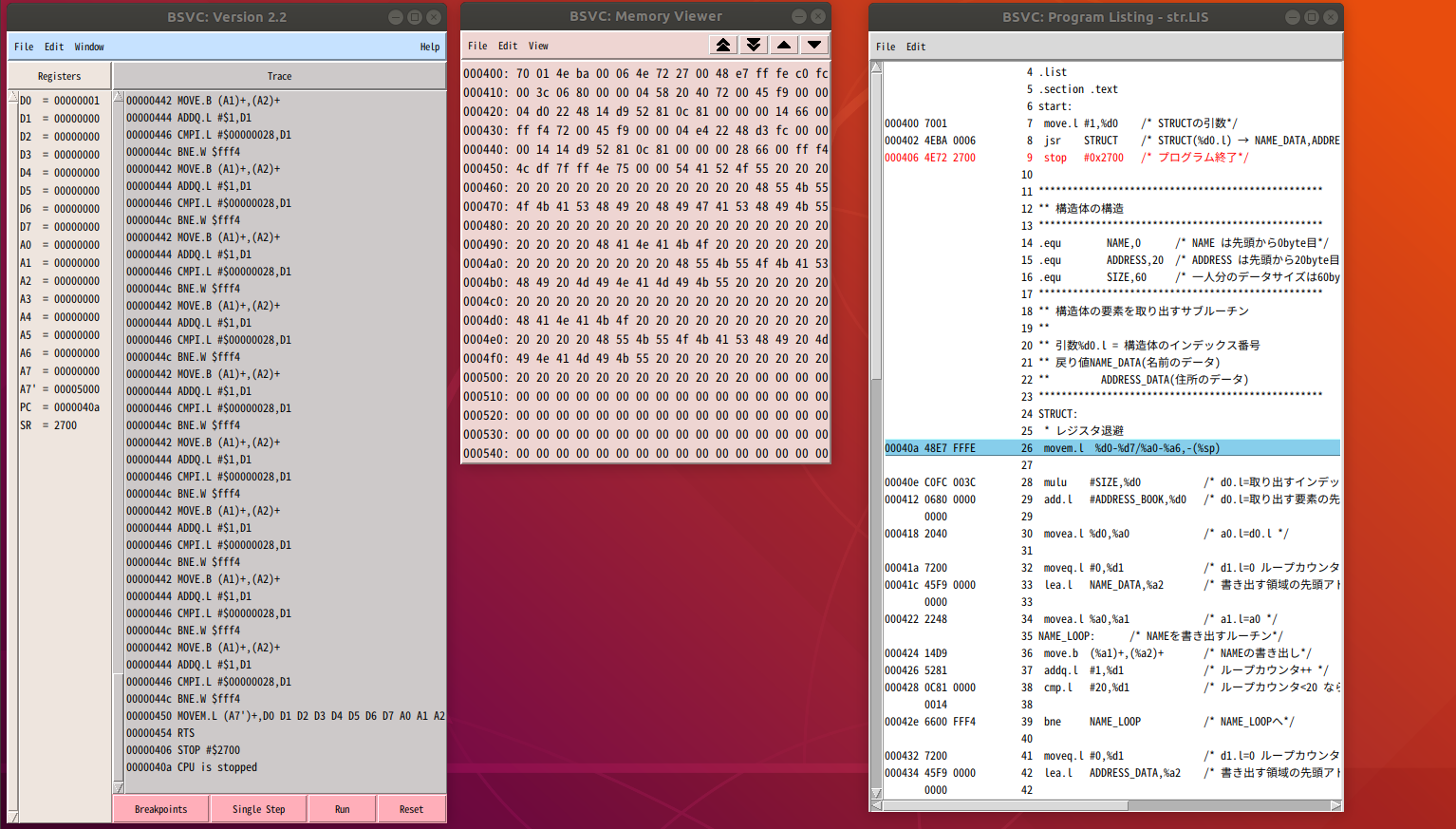
(1)

このプログラムでの、メモリ内のデータ配置は以下である。



プログラム実行前のメモリ内のデータ配置を簡単に考察すると、メモリの0x000400番地から0x000455まではテキスト領域で、0x000458番地から0x0004cfまでデータ領域である。

(2)

 構造体の要素を取り出すアセンブラプログラムの実行結果は以下である。

上の実行結果からこのプログラムが正しく動いていることを簡単に説明する。

STRUCTの出力として、NAMEデータが0x0004d0番地から0x0004e3番地に、ADDRESSデータが0x0004e4番地から0x0004eb番地に出力されている。例えば、0x0004d0番地の48は"H"を示している。

(3)

本項での感想・意見を述べる。

本項では、サブルーチンプログラムやA1・A2レジスタで用いたポストインクリメント付きアドレスレジスタ間接形式について模範コードをトレースしながら学べた。また、C言語等で聞いたことはあるがあまり意識していなかったメモリのテキスト領域とデータ領域などもMemory Viewerなどで視覚的に認識しながら学ぶことが出来てよかった。