Lab Exercise #14 -- Assembler Processing

A. Complete the exercises below.

1. The second page of this worksheet lists the source lines for a SPARC assembly language program. Fill in each blank in the column on the left of the page with the appropriate hexadecimal value (the value of the location counter before that source line is processed during Pass One).

Note: the assembler maintains a separate location counter for each segment. Use the notation "T+xxxx" or "D+xxxx" (where "T" and "D" refer to the text and data segments, and "xxxx" refers to a hexadecimal offset) to give the value of the location counter.

2. Complete the symbol table shown below, based on the processing which you performed in part (1). For each symbol, indicate its value (a specific constant or a segment plus a hexadecimal offset), whether its value is absolute or relocatable, and whether it is a local or global symbol.

symbol	value (segment+offset)	abs/rel	local/global
__unpack	__T+0000	__rel__	__local
__list	__D+0000	__rel__	__local
__masks	__D+0014	__rel__	__local
__fmt	__D+0020	__rel__	__local
__SIZE	__5	__abs__	__local
__main	__T+0028	__rel__	__global
__loop	__T+0038	__rel__	__local
__endloop	__T+0070	__rel__	__local

3. For each of the indicated source lines, fill in the object code (machine instruction or data value) which would generated during Pass Two and placed in the object code file. Give your answers using hexadecimal notation.

B. Assemble the source code file and check your work using the following commands:

```
<prompt> gcc -c ~cse320/Labs/lab14.source.s
<prompt> dis -d .data -t .text lab14.source.o
```

Alternatively, you can use the following command to generate an assembly listing:

```
<prompt> ~cse320/bin/assem -a ~cse320/Labs/lab14.source.s
```

If any of your responses are incorrect, re-work that section of the worksheet.

```
_____________             .section ".text"
__T+0000____              .align   4
__T+0000____    unpack:
__T+0000____              set      masks, %l7
__T+0008____              ld       [%l7+0], %o2
__T+000C____              ld       [%l7+4], %o3
__T+0010____              ld       [%l7+8], %o4
__T+0014____              and      %o1, %o2, %o2
__T+0018____              and      %o1, %o3, %o3
__T+001C____              and      %o1, %o4, %o4

__T+0020____              retl                        ! __81C3E008_____________
__T+0024____              nop

_____________             .section ".data"
__D+0000____              .align   4
__D+0000____    list:     .single  0r-64.0
__D+0004____              .single  0r+1.625
__D+0008____              .single  0r-1.3e-6
__D+000C____              .single  0r-100.0625
__D+0010____              .single  0r+12.6e+32

__D+0014____    masks:    .word    0x80000000
__D+0018____              .word    0x7f800000
__D+001C____              .word    0x007fffff

__D+0020____    fmt:      .asciz   "Number: %8.8x Fields: %8.8x %8.8x %8.8x\n"
__D+0049____              .align   4

__D+004C____    SIZE      =        5

_____________             .global  main
_____________             .section ".text"
__T+0028____              .align   4
__T+0028____    main:
__T+0028____              save     %sp, -96, %sp

__T+002C____              mov      0, %l0
__T+0030____              set      list, %l2
__T+0038____    loop:
__T+0038____              cmp      %l0, SIZE
__T+003C____              bge      endloop          ! __1680000D_____________
__T+0040____              nop

__T+0044____              sll      %l0, 2, %l1
__T+0048____              ld       [%l2+%l1], %o1
__T+004C____              call     unpack           ! __7FFFFFED_____________
__T+0050____              nop
__T+0054____              set      fmt, %o0
__T+005C____              call     printf           ! __40000000_____________
__T+0060____              nop

__T+0064____              inc      %l0
__T+0068____              ba       loop             ! __10BFFFF4_____________
__T+006C____              nop
__T+0070____    endloop:

__T+0070____              ret                       ! __81C7E008_____________
__T+0074____              restore
```