T4.1 Label References

This tutorial will demonstrate how to use various resources to get information about labels used in code.

Following is a table of contents for this tutorial.

- T4.1.1 Create new hardware project
- T4.1.2 Get template main code
- T4.1.3 Assemble code and identify assembler listing information
- T4.1.4 Determining the address referenced by a label
- T4.1.5 Using an assembler label with gdb

T4.1.1 Create new hardware project

First, we will create a project, change to the newly created project directory, and switch to hardware mode:

```
Lab04 $ create_project pla
Project: pla created
Lab04 $ cd pla/
Lab04/pla $ make switch
mode changed to hardware
```

T4.1.2 Get template main code

```
Lab04/pla $ cd src
Lab04/pla/src $ wget http://ece.boisestate.edu/ECE330/code/lab04p1a_main.s
... 
Lab04/pla/src $ cd ..
```

Next, edit the `makefile` to look like the following.

```
1 PGM=pla
2 SRCS=lab04pla_main.s
3 include makeopts
```

The `main` function will now be ARM assembly language code.
T4.1.3 Assemble code and identify assembler listing information

Assemble the project, and look at assembler listing using the following:

```
Lab04/p2a $ make all
Lab04/p2a $ less src/lab04pla_main.s.lst
```

Refer to the annotations in the code above to observe where to find the various types of information requested on the following page.
When a label is used on the same line as an assembler data directive, by convention we will associate the type of data of the directive with the label.

```
SL --LC --BYTES-- --- assembly code ------
  28 0004 78563412 AFN: .word 305419896
```

Q1. What is the value of the label AFN?
A1. The value of the label AFN is 4 (0004).

This can be found in the second column at line 30 of the source file. It can also be found in the symbol table, which shows that symbol (label) AFN is at location 4 of the .data section.

Q2. What is the value of the memory referenced by the label AFN?
A2. The value of the memory referenced by the label AFN is 0x78 0x56 0x34 0x12

Since the label AFN is associated with the .word assembler data directive, the size the referenced memory is 4 bytes. Though the third column shows the data in byte order, the value shown as 78563412 is more correctly depicted as the four hex bytes:

```
0x78 0x56 0x34 0x12
```

Q3. What is the value of the data referenced by the label AFN?
A3. The value of the data referenced by the label AFN is 305419896 (decimal), or 0x12345678

Since the label AFN is associated with the .word directive, the size of the data referenced by the label is a 4-byte value. Since the processor is little endian, the hexadecimal representation of 4 bytes from memory is the value 0x12345678.

Q4. What is the address referenced by the label AFN?
A5. The address referenced by the label AFN is not shown in the assembler listing, as it is not known until after the link process.
T4.1.4 Determining the address referenced by a label

objdump can be used to discover the address referenced by the label AFN.

```
Lab04/pla $ arm-none-eabi-objdump -s -t -j .data bin/pla.elf
```

```
bin/pla.elf: file format elf32-littlearm
SYMBOL TABLE:
00000000 l .data 00000000 .data
00000004 l .data 00000000 AFN
00000100 l 0 .data 00000428 impure_data
00000200 g .data 00000000 _data_start_
00000438 g .data 00000000 _data_end_

Contents of section .data:
00000000 bbbbbbb 78563412 eeeeee e0000000 ........xV4........
00000010 0000000 fc620020 64030020 cc038020 ...... d. ...
00000200 0000000 0000000 0000000 0000000 0000000 0000000 ...........
00000300 0000000 0000000 0000000 0000000 0000000 ...........
00000400 0000000 a6400008 0000000 0000000 0000000 ...........
00000500 0000000 0000000 0000000 0000000 0000000 ...........
00000600 0000000 0000000 0000000 0000000 0000000 ...........
00000700 0000000 0000000 0000000 0000000 0000000 ...........
```

This listing shows that the address **0x20000004** has been assigned to the label AFN.
T4.1.5 Using an assembler label with gdb

Yet another way to get answers to a lot of these questions is through use of gdb.

The address referenced by a label can be shown with the following:

(gdb) p /a &AFN
$1 = 0x20000004

Since assembler labels only reference data and have no knowledge of size, the size must be specified by using the gdb casting operator. The data referenced by the label can be seen below.

(gdb) p /x (unsigned int)AFN
$2 = 0x12345678

The data referenced by a label can be displayed using the following command. Note that since AFN is associated with a word, w is the size identifier.

(gdb) x /1xw &AFN
0x20000004: 0x12345678

The actual address can also be used instead of using the address-of operator &.

(gdb) x /1xw 0x20000004
0x20000004: 0x12345678

Since core marks were placed around the data, we can display them by adjusting the starting address and displaying 2 extra words:

(gdb) x /3xw 0x20000000
0x20000000: bbbbbbbbbbb 0x12345678 eeeeee

To show the memory in byte (physical) order, use the following command:

(gdb) x /4xb &AFN
0x20000004: 0x78 0x56 0x34 0x12