EECS 388: Computer Systems and Assembly Language

Lab 6

Assembly Language: Loops, Branches, and Subroutines

Objective: To learn to combine loops, branches and subroutines.

Problem 1:

There's an ASCII string of arbitrary length stored in a table in locations starting at $5000. The final byte of the string is $FF. Write a program to convert all the lower-case letters of the string to upper-case, and copy the entire string to a table in locations starting at $6000.

For example, if the following string is stored starts at location $5000:

Here's a string with punctuation!

Then after program execution, the capitalized string will be copied to locations starting at $6000:

HERE'S A STRING WITH PUNCTUATION!

In your program, you are required to write a subroutine. ASCII codes are on page 396. Recall from lab 1 that if you put ASCII hex codes into memory, their ASCII character equivalents are displayed on the right-hand side of the monitor window. Explanation of subroutines is in Section 3.4 of the book.

In order to prepare for the lab, at a minimum you should review loops and branches from the last lab, and sections 2.6, 3.2, and 3.4. Additionally, you should be able to answer the following questions:

1) How can you distinguish lower-case characters from other ASCII characters?
2) How can you convert the lower-case characters to upper case?
3) Since you know the string will always end in $FF, how will you detect the end of string in the loop?

The following program template is for your reference.

Program Template:

; Description: Takes a string terminated in $FF from one memory location, converts the lower- case characters to upper-case, and copies it to another location.

```
                            ;Make table TAB1 start at $5000.
                            ;Make table TAB2 start at $6000.

TEST          EQU           ;This character signals the end of the string.

MAIN                        ;main program
                            ;Initialize stack pointer
                            ;Load table 1 and 2  addresses to X and Y.

                            ;Call COPYTBL subroutine
                            ;Stop program execution.
```

```
COPYTBL                 ;Save A, X, and Y registers on the stack

LOOP                    ;Check to see if we're at the end of the string and leave the loop if so.
                        ;Load current character. If it's a lower-case character, convert it to upper-
                        ;case and copy it to table 2.
                        ;Repeat loop

DONE                    ;restore cpu registers from the stack
                        ;return from subroutine
```

REQUIRED: Demonstrate your program to the GTA using the following procedure:

1. Use block fill to load memory locations $5000 – $6100 with zeros.
2. Load your program onto the board using hyperterminal.
3. Display locations starting at $6000.
4. You will then enter the string given by GTA into memory locations starting at $5000.
5. Execute the program, and display memory locations starting at $6000.

Report Format and Grading:

Following the report format in your syllabus, include the following in your report:

1. Your name, student number, lab project number and title, course number, lab section number, and date.

2. Description of the lab in your own words. What did you learn? If your code did not work in the lab, explain why. (45% of report grade)

3. The source code for your program, containing all three problems. **Use comments to indicate what changes you made to the program template.** (45% of report grade)

4. A short evaluation of the lab. What did you like about the lab? What could be improved? (10% of report grade)