

# IT 209: Lab Manual

## Lab 4

### General Instructions:

- Students are required to maintain a lab notebook for this course.
- At the end of each lab, the students will be graded on the scale of 10.
- Take home assignments has to be submitted to the following e-mail id day before the next lab session.

it209.co@gmail.com

- Keep subject as <student\_id>, Lab#<lab\_number>. Write all program codes in one text file with proper indentation and order. Name text file same as subject of your mail.

### **Target Device: NXP LH75400**

### Tasks:

1. Write an assembly language program to perform dot product of two vectors using subroutine. Pass arguments to the subroutine using both pass-by-stack technique.
2. Write an assembly language program to solve the expression  $x^2+y^2$  using subroutine. Pass arguments to the subroutine using both pass-by-register and pass-by-reference technique.
3. Write an assembly language program to reverse a string without using any other memory location.

### Take-Home Assignments:

4. Write an assembly language program to find GCD of two numbers using subroutine. Pass arguments to the subroutine using pass-by-stack technique.
5. Write an assembly language program to check whether a string is palindrome or not using subroutine. Pass arguments to the subroutine using pass-by-reference technique.

6. Translate each of the following pseudo-code statements into a sequence of ARM assembly language instructions. Assume x and y are integers and x is in R1 and y is in R2.

```
if (x > 9) {  
    x = 0;  
    if (y > 9) {  
        y = 0  
    }  
    else {  
        y = y + 1;  
    }  
}  
else {  
    x = x + 1;  
}
```

7. If R1 = 0x1000 and R4 = 8, what memory location (in hexadecimal) is loaded into R0 and what is the value of R1 (in hexadecimal) after each of the following instructions has been executed.

1. LDR R0, [R1, #8] ; R0 = [0X1008], R1 = 0X1000
2. LDR R0, [R1], #-8
3. LDR R0, [R1, #12]!
4. LDR R0, [R1, R4]
5. LDR R0, [R1], R4
6. LDR R0, [R1, R4]!
7. LDR R0, [R1, R4, LSL #3]
8. LDR R0, [R1], R4, LSR #1
9. LDR R0, [R1, R4, LSL #2]!