Lab 4: BIT OPERATIONS

Objectives

1. To be able to use instructions dealing with bits

Preliminary Work

Suppose that you are writing a date handling firmware for the 8-bit MCU in assembler language! And the dates are handled by a subroutine which gets the 16-bit input date data and returns the ASCII string form of it to send the display module.

The date data format is as following:

- 4 bits for month number,
- 5 bits for day number
- 7 bits for year number

Ordered respectively in a 16-bit date data input. The subroutine you wrote should get the address of input, and return the ASCII equivalent string (character array) for that date.

For example, input data address is $D000. The input located at this address is

\[ M[\text{D000,D001}] = \text{A938 : %1010 10010 0111000 (binary)} \]

mm/dd/yy (decimal) : 10 18 56

We have a bias value of 1950 to represent years. So in this example year becomes \(1950 + 56 = 2006\) in decimal format.

The output string for this date is:

OCT 18th, 2006

More examples;

Input sample: $9120 : % 1001 00010 0100000

9 2 32 → SEP 2nd, 1982

$4BB1 : % 0100 10111 0110001

4 23 49 → APR 23rd, 1999

$B580 : % 1011 01011 0000000

11 11 0 → DEC 11th, 1950

You have to be careful about the \(st, nd, rd,\) and \(th\) suffixes since 11, 12, and 13 take \(th\) while 1, 21, 31 takes \(st, 2, 22\) takes \(nd,\) and 3, 23 takes \(rd.\) You should have to find an intelligent method to detect the number’s appropriate suffix.

(Note that all these are ASCII coded character arrays; remember that you can view the ASCII equivalent of the memory content in the monitor program.)

You will write a subroutine of this virtual firmware, which converts 2 Bytes input date data to the human-readable string format which is explained above.

The subroutine will get the address of 16-bit data as input and return the corresponding string (You can use C convention – null (simply 0) character at EndOfString).
1. Write the code for the subroutine described above.
2. Extend your first subroutine to make it able to compare two input dates and evaluate and display the newer one.

Hint: You can use look-up tables for the strings of month names and numbers.

Lab Work

1. Assemble and run the preliminary work programs using JSR instructions.
   The input locations for first & second dates are $D000 & $D002, respectively.
   The starting address of the output string is $D010.
2. Explain to your TA how you decoded the input data.

After Lab

1. Write a report describing what you have done in the lab concisely.
2. Take a look at the listings of the programs generated by AS11 –L option.
   
   \textit{DOS PROMPT> as11 filename.txt –L filename list.txt}
3. Explain what you learned in the lab. What kind of errors did you encounter?

- DO THE ALL WORK YOURSELF!
- DO NOT HESITATE TO ASK QUESTIONS TO YOUR TA.