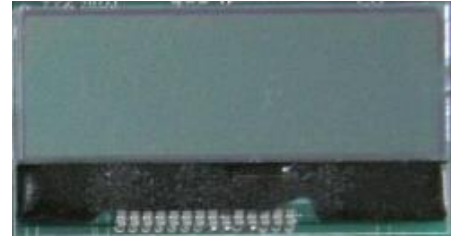


# Lab#3 (LCD, Timers & Interrupts)

---

In this lab, you will be programming a real time clock (displaying hours:minutes:seconds) incorporating an interrupt driven timer and an LCD display.

The **data** pins of the LCD display, **0-7**, are connected to Port H. The **control** lines – Read/nWrite ( $R/\bar{W}$ ), Enable (E) and Register Select (RS) – are connected to Port E (RE2, RE1 and RE0, respectively.)



You can base your program on the **LCDDTest18F.asm** or other programs found the course web site.



All SW2 DIP switches should be off for this program.

## Programming Tasks:

- Configuring the timer (**TMRO**) for interrupts
- An interrupt service routine (ISR) to keep track of hh:mm:ss
- Math for comparison and addition (in the ISR)
- Numeric conversions (e.g. to ASCII characters)
- Basic LCD display code
- Use of table pointers

Begin by programming a interrupt driven timer which keeps track of seconds:minutes:hours. Set the timer to interrupt every second then increment seconds -- the easiest way to handle this is to use two registers (SecondsOnes and SecondsTens). When SecondsOnes exceeds 9, increment SecondsTens and reset SecondsOnes to zero. When the final count exceeds 59, increment minutes and then hours in turn. All of this is done inside the ISR using the twelve variables (two each for seconds, minutes, and hours.)

While the ISR handles the real time clock, the main program displays all data to the LCD.

The program must begin by displaying the your name on the top line of the display for five seconds after which the LCD will continually display the time on the top line of the display as "HH:MM:SS" and the time in WORDS on the bottom line of the display. The bottom line will display a different message every ten minutes, for example "Eight Twenty" between 8:20 and 8:29 and "Eleven O'Clock" between 11:00 and 11:09, etc. These messages are generated by using strings stored in program memory (i.e., tables).

In summary, the LCD would display

B	O	B		S	M	I	T	H							

when starting, then

1	1	:	4	4	:	1	5								
E	L	E	V	E	N	F	O	R	T	Y					

Messages for the bottom line are from two strings: the first string being "One" through "Twelve" and the second string "O'Clock" through "Fifty". If the total string exceeds 16 characters, simply let it run off the display (i.e. truncate to 16 characters).

Download the string tables and **StringOut** routine from the lab web site.

### Optional Bonus Assignment

Add six pushbutton keys to the system (on the matrix keypad), representing "Hours Up", "Hours Down", "Min Up", etc. to allow the time to be set (without this, the clock starts at zero every time). The keypad must be scanned to do this (polling each column) -- five I/O lines are required to do this (with the six pushbuttons configured as a 2\*3 matrix).

By completing this part, it is possible to achieve a mark higher than 100% on this lab.

Demonstrate the lab working to the professor by using the debugger to set the time to 11:59:45 and running the program (the LCD will then show the roll over from "Eleven Fifty" to "Twelve O'Clock").

*You may work individually or in pairs (**no** groups of three or more) on this lab. In any case, you or you **and** your partner will have to answer questions about **your** code as part of the demonstration.*

**Due Date: Friday, April 13, 2018**

*Late labs lose 10% per day late, and receive a zero after 5 days late (Apr. 19)*