Embedded Systems
Assignment 1

Enrico Simetti

ISME - Interuniversity Research Center on Integrated Systems for Marine Enviroment
DIBRIS - Department of Computer Science, Bioengineering, Robotics and System Engineering
University of Genova, Italy
enrico.simetti@unige.it

Assignment 1

1. Simulate an algorithm that needs 7 ms for its execution, and needs to work at 100 Hz.
2. Read characters from UART and display the characters received on the first row of the LCD.
3. When the end of the row has been reached, clear the first row and start writing again from the first row first column
4. Whenever a CR '\r' or LF '\n' character is received, clear the first row
5. On the second row, write "Char Recv: XXX", where XXX is the number of characters received from the UART2. (use sprintf(buffer, "%d", value) to convert an integer to a string to be displayed
6. Whenever button S5 is pressed, send the current number of chars received to UART2
7. Whenever button S6 is pressed, clear the first row and reset the characters received counter

Writing to LCD with SPI

The following power-up sequence should be observed by the user's application firmware when writing characters to the LCD:

1. After any reset operation wait 1000 milliseconds to allow the LCD to begin normal operation. The cursor on the LCD will be positioned at the top row on the left-most column.
2. Configure SPI1 module on your dsPIC30F device to operate in 8-bit Master mode. The serial clock may be set for any frequency up to 1 MHz.
3. To write an ASCII character to the LCD at the location pointed to by the cursor, load the SPIBUF register with the ASCII character byte.
4. After the character is displayed on the LCD, the cursor is automatically relocated to the next position on the LCD.
5. To reposition the cursor to another column on any of the two rows, write the address of the desired location to the SPIBUF register. Addresses in the first row of the LCD range from *0x80* to *0x8F*, while addresses on the second row range from *0xC0* through *0xCF*.
6. After 16 characters are written to the first row on the LCD, it is necessary for the user's application to write the address 0xC0 of the second row to the SPIBUF in order to roll the cursor over to the second row.
7. The user application must wait for a minimum of (8 bits / SPI Frequency) between writing two successive characters or addresses.

Skeleton Solution I

```
void algorithm() {
  tmr_wait_ms(TIMER2, 7);
}

int  main() {

  tmr_setup_period(TIMER1, 10);
  while(1)  {
    algorithm();

    // code to handle the assignment

    tmr_wait_period(TIMER1);
  }
}
```