EE 337: SPI and ADC Interface to Weighing Machine Lab 7

Rajbabu Velmurugan and Shankar Balachandran

14th September, 2015

Objective

In this lab sessions you will learn how to interface an ADC (MCP3008) with the 8051 micro-controller using the SPI protocol. SPI (Serial Peripheral Interface) is a synchronous, serial communication protocol used for communicating between microcontrollers and peripherals (or between 2 or more microcontrollers). Once the ADC is interfaced, it is used to measure the output of an electronic weighing machine and determine the weight that has been put atop the weighing machine.

Problem Description

The output provided by the weighing scale is 0.74mV/gm. The task is to interface an ADC with the microcontroller and use the ADC to measure the voltage output from the weighing machine and thus determine the weight put on top of it. The weight measured is then to be displayed on the LCD Screen in the format,

```
Weight: xxxx gm
```

The weight is to be sampled every 250ms using a hardware timer (interrupt based) and updated on the LCD screen so that a consistent and fast updating display is provided to the user. While updating the display, only the actual weight xxxx is to be updated as the rest of the display needs to remain constant throughout the life of the program. The program should take 10 ADC samples every 250ms and take an average of these 10 samples, to compensate for the fluctuations in the output voltage.

You are to perform these tasks.

1. Interface the ADC MCP3008 to the microcontroller in the Pt-51 board

2. Set up one of the hardware timers (Timers 0, 1 or 2) to raise an event every 250ms.

3. Measure the voltage output from the Weighing Scale using the ADC (Take 10 samples of the output every time the timer interrupt is raised and take their average).

4. Convert the measured voltage into weight using the given standard (0.74mV/gm).

5. Display the weight on the LCD.

While displaying the weight, the following is to be kept in mind: The reference voltage provided to the ADC is 3.3V and the ADC has a resolution of 10 bits (Refer to the MCP3008 datasheet for more details). Hence the minimum voltage that can be measured by the ADC is only around 3.3 mV (3.3V/1024). Since the output of the Weighing Scale is around 0.74mV/gm the display is to be approximated to the nearest 10s of grams.

Homework

1. Go through and understand the reference document on embedded C uploaded in Moodle.

2. Read through "SPI-Intro.pdf" file.

3. Read through "ADC Interfacing.pdf" file

Lab Work

1. Write a program to display "Hello" on the first line of the LCD Screen starting from the first position and "World" on the second line of the LCD starting from position 12.

2. Connect a potentiometer between the 3.3 V and ground supplies and use the ADC to measure the voltage at the variable point of the potentiometer. Display the measured voltage on the LCD in the format "Voltage: xxxx mV". Verify that the displayed voltage is correct using a DMM. Display the result to the closest 10s of millivolts.

3. Measure the voltage output from the weighing machine and display the weight on the LCD screen.

Two C code files (`lcd.c` and `adc.c`) are also provided to you along with this assignment. Use the file `lcd.c` to do Part-1. Use the file `adc.c` to perform Part-2. Then create another file `weighmac.c` to do Part-3. The functions and code from Part-1 and Part-2 can be used to do Part-3.

Functions in the `lcd.c` and `adc.c` files:

The file contains the following functions

- `SPI_Init( )` : Initializes the SPI Module and configures it to interface with the ADC.

- `LCD_Init( )` : Initializes the LCD screen using the same commands used in the assembly language programs written earlier, clears the LCD and sets the cursor position to Line 1 Position 1.

- `LCD_DataWrite(char dat)` : Writes a character on the LCD screen. e.g.,

  ```
  LCD_DataWrite(0x38);
  ```

- `LCD_CmdWrite(char cmd)` : Writes a command to the LCD. e.g.,

  ```
  LCD_CmdWrite(0xC6);
  ```

- `LCD_WriteString(char * str, unsigned char len)` : Writes a string on the LCD Screen. e.g.,

  ```
  LCD_WriteString(Hello, 5);
  char str[10];
  LCD_WriteString(str[3], 4);
  ```

  Please note that while displaying a string, the number of characters should match the number of characters in the string. Otherwise incorrect values will be displayed.

- `LCD_Ready()` : Checks if the LCD is ready to receive commands

- `sdelay(int delay)` : Produces a small delay (15 μs for a 24MHz clock)

Documents to Refer to

The following documents from the lab website will be useful for understanding SPI.

1. First read through SPI-Intro.pdf available in the course page.

2. Read through "ADC Interfacing.pdf" file.

3. 89c5131datasheet.pdf file that is in the lab website. Page 93 has the relevant documentation for configuring SPCON.

Read the following documents for programming with embedded C.

1. Embedded C programming for 8051 using Keil.pdf (available in the "Supporting material" section).

2. Programming style sheet `http://wel.ee.iitb.ac.in/teaching_labs/Microprocessor/Developement%20Platforms/Pt-51/programming%20stylesheet.pdf`