EE333 Introduction to Microcontrollers

Lab#6 Analog Measurement

Allan A. Douglas, Associate Professor

**Lab Overview**

In this lab, students will learn how to use microcontroller to measure external analog signals. Students will write an Arduino sketch to measure analog voltage from a potentiometer and from the function generator as well as temperature from a temperature sensor. Students will display their measurements on the Serial Monitor and on the LCD.

**Required Materials**

|  |  |
| --- | --- |
| Breadboard | Thermal Sensor (TMP36) |
| Arduino UNO (or Spark Fun RedBoard equivalent) | Potentiometer |
| ATmega328P Microcontroller Datasheet | Oscilloscope |
| Power Resistor | Arduino ISE Software |
| DC Power Supply | Function Generator |
| Voltmeter | LCD Display |
| Power Resistor | DMM with Temperature Probe |

**Required Work**

The areas of **bold** text below are items required in your Lab Report. Make sure you include the following sections in your Lab Report: Introduction, Results, Discussion, and Conclusion.

Part 1: DC Voltage Measurement

The goal of this part of the lab is to measure DC voltage. Connect a potentiometer to a 5V DC Power supply. Connect the center wire of the potentiometer to the ADC0 input on the Arduino as shown in the figure to the right.

Write an Arduino sketch to measure the ADC0 input, scale your ADC measurement so that full range corresponds to 5V (set your reference to Vcc), and display the voltage on your LCD and on the Serial Monitor. Determine how many digits you should display beyond the decimal point based on the ADC accuracy.

You may use analogRead() to test the functionality, but do not use analogRead() in your final code. You need to control the ADC using the internal registers directly. You may use delay() in your sketch to control the ADC sampling rate.

**Include a copy of your sketch in your Lab Report.**

**Include a picture of your circuit with the LCD and voltmeter in your Lab Report.**

Create a table of your LCD displayed voltages versus the voltage on the center pin of the potentiometer when measured with a voltmeter. Vary the potentiometer between 0V and 5V in 0.5V steps.

**Include your table in your Lab Report. Explain your results.**

Part 2: Sinusoidal AC Voltage Measurement

Setup the function generator to generate a 10Hz sine wave with an amplitude of 2.5V with a +2.5V offset. Check the waveform with the oscilloscope. Your sine wave should vary between 0V and 5V. Make sure the oscilloscope is DC coupled. If using an Analog Discovery for the function generator, you will not need the 50 Ohm resistor.

Write an Arduino sketch to measure the ADC1 input. Determine how many digits you should display beyond the decimal point based on the ADC accuracy. Set your sampling rate to 125kHz and use Free Running mode. Write your code to determine the minimum voltage, the maximum voltage, and the period. Display your results on the Serial Monitor every second.

You may use analogRead() to test the functionality, but do not use analogRead() in your final code. You need to control the ADC using the internal registers directly.

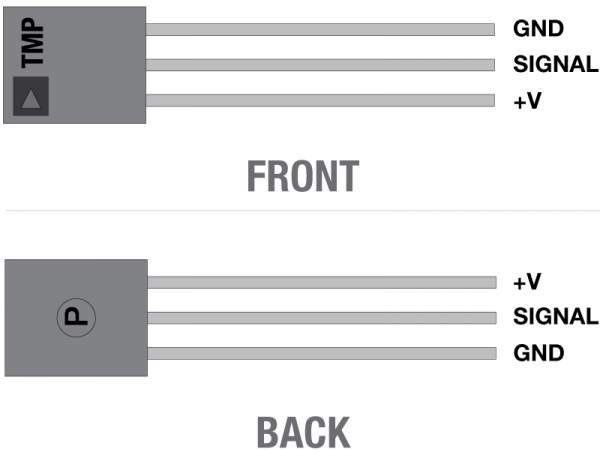
**Include a copy of your sketch in your Lab Report.**

**Include a picture of your Serial Monitor output and the oscilloscope output in your Lab Report.**

Increase the function Generator frequency to 100Hz, 1kHz, and 10kHz.

**Include a picture of your Serial Monitor output and the oscilloscope output for each of these frequencies in your Lab Report.**

Part 3: Temperature Measurement



For this part of the lab you will measure the internal chip temperature and the external temperature with a thermal sensor.

Replace the potentiometer with the three-terminal temperature sensor (TMP36). Connect +V on the sensor to +5V on the Arduino and connect the GND pin to ground.

Write a sketch to measure two temperatures per second. You may use the delay() function for this.

Your sketch should set the ADC input MUX to the internal temperature reference, start a measurement, and wait for the conversion complete interrupt. Then read the ADC data registers, set the ADC input MUX to the pin ADC0, start a measurement, and wait for the conversion complete interrupt.

Now, convert both temperatures to degrees C. Determine how many digits you should display beyond the decimal point based on the ADC accuracy. Display both measurements on the Serial Monitor.

**Include a copy of your sketch in your Lab Report.**

Next, take a power resistor and connect it to a DC power supply. Note the power rating of your resistor. Remember, P = V2/R.

Apply voltage to the power resistor, until it starts to feel warm. Do not exceed the rating of your power resistor.

Place your temperature sensor in contact with the power resistor.

Attach the thermal problem from the DMM to the power resistor. Measure the power resistor temperature.

**Include pictures of your Serial Monitor output, the DMM output, and your circuit in your Lab Report.**

Create a table of your Arduino measured temperatures versus the DMM temperature. Vary the temperature from room temp to 55C in approximately 5C steps. Do the measurements match?

**Include your table in your Lab Report. Explain your results.**