Lab 7d Position Data Acquisition System

This laboratory assignment accompanies the book, Embedded Microcomputer Systems: Real Time Interfacing, by Jonathan W. Valvano, published by Brooks-Cole, copyright © 2000.

Goals • Study ADC conversion,
• Develop a position measurement system using a slide potentiometer.

Review • Operation of the 6812 ADC system in the Technical Data on MC68HC812A4 manual,
• Valvano Section 11.10 on the 6812 ADC, Section 12.1 on DAS parameters,
Subsection 12.2.4 on position sensors.

Starter files • `OC` `OC3` `LCD` and `ADC` projects

Background

You will design a position meter with a range of about 3 cm. The potentiometer converts position into resistance (0<R<9kΩ). You will use an electrical circuit to convert resistance into voltage (V). Since the potentiometer has three leads, one possible solution is shown in Figure 7.1. The 6812 ADC will convert voltage into a digital number (N). Your software will calculate position from the ADC sample. The position measurements will be displayed on a LCD. You should use the LCD interface developed in a previous lab.

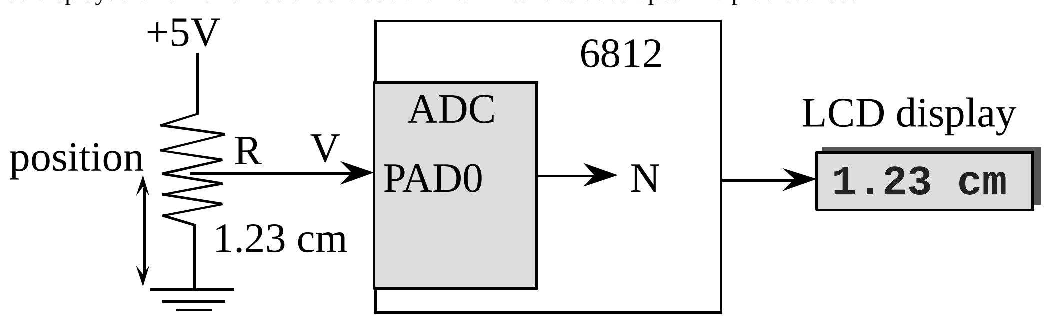

Figure 7.1. Possible interface.

You should make the position resolution and accuracy as good as possible. The position resolution is the smallest change in position that your system can reliably detect. In other words, if the resolution were 0.01 cm and the position were to change from 1.00 to 1.01 cm, then your device would be able to recognize the change. Resolution will depend on the amount of electrical noise, the number of ADC bits, and the resolution of the LCD display. Accuracy is defined as the absolute difference between the true position and the value measured by your device. Accuracy is dependent on the same parameters as resolution, but in addition it is also dependent on the stability of the transducer and the quality of the calibration procedure.

Since the transducer is not linear, you could use a piece-wise linear interpolation to convert the ADC sample to position (Δ of 0.01 cm.) The 6812 assembly language **`tbl`** instruction is an efficient mechanism to perform the interpolation. The **tbl.RTF** assembly program included with TExaS is an example of a piece-wise linear interpolation using the **`tbl`** instruction. There are two small tables `Xtable` and `Ytable`. The `Xtable` contains the ADC results and the `Ytable` contains the corresponding positions. The ADC sample is passed into the lookup function. This function first searches the `Xtable` for two adjacent of points that surround the current ADC sample. Next, the function uses the `tbl` instruction to perform a linear interpolation to find the position that corresponds to the ADC sample. You are free to implement the conversion in any acceptable manner. You are not allowed to use a simple linear equation. You may use C, assembly, or a mixture of C/assembly.

The 8-bit ADC converters on the 6812 are successive approximation devices with a short conversion time. Normally, an ADC ritual should initialize the ATDCTL2 ATDCTL3 and ATDCTL4 registers. As a minimum, you will need to set ADPU bit in the ATDCTL2 register equal to one to activate the ADC. Otherwise, you can use the default values for the other bits in these three registers. Writing to the ADC Control register (ADCTL5) begins a conversion. To sample a single channel four times in sequence, you can make S8CM=SCAN=MULT=0, and set CD,CC,CB,CA select the channel number. The ADC chip clocks itself. After the first sample is complete, CCF0 is set and the result can be read out of ADR0H (0x0070). After all four samples are available (ADR0H,ADR1H,ADR2H,ADR3H), the SCF bit is set. One software possibility is

```
void ADC_Init(void){
  ATDCTL2 = 0x80;   // Activate ADC
  ATDCTL3 = 0;      // no freeze
  ATDCTL4 = 1;      // sample time 2 E clocks, /4 clock
}
unsigned char ADC_Sample(unsigned char chan){
  ATDCTL5 = chan;                    // Start ADC
  while((ATDSTAT & 0x0001) == 0){};  // wait for CCF0
  return(ADR0H);
}
```

In this lab you will be measuring the position of the armature on the slide potentiometer. This signal has very few frequency components (0 to 1 Hz.) According to the Nyquist Theorem, we need a sampling rate greater than 2 Hz. You may choose any sampling rate in the range of 2 to 50 Hz. You will sample the ADC at that rate and calculate position using decimal fixed-point with Δ of 0.01 cm. You should display the results on the LCD, including units. No floating point is allowed. Output compare interrupts will be used to sample the ADC in a background thread. This high priority interrupt will establish the sampling rate.

> **Nyquist Theorem**: If f_{max} is the largest frequency component of the analog signal, then you must sample more than twice f_{max} in order to faithfully represent the signal in the digital samples. For example, if the analog signal is $A + B \sin(2\pi ft + \phi)$ and the sampling rate is greater than 2f, you will be able to determine A, B, f, and ϕ from the digital samples.
>
> **Valvano Postulate**: If f_{max} is the largest frequency component of the analog signal, then you must sample more than ten times f_{max} in order for the reconstructed digital samples to look like the original signal when plotted on a voltage versus time graph.

Preparation (do this before your lab period)

1. Review the technical information on the ADC system of 6812. What initiates the conversion process? What are two ways of knowing that the conversion process has been completed?

2. Firmly attach the frame (the fixed part) of the potentiometer to a solid object. Place a metric ruler on the frame but near the armature (the movable part) of the sensor. You could Xerox a metric ruler and tape the paper onto the frame. Attach or draw a hair-line to the armature, which will define the position measurement. Solder three solid wires to the slide potentiometer. If you do not know how to solder, ask your TA for a lesson.

3. Design and build the electrical circuit that interfaces the transducer to the ADC. If you use external devices be careful not to produce voltages above +5 or below 0 volts.

4. Load and run the **`ADC`** project. You will use it to perform a calibration to determine the ADC sample for 6 positions. You will determine the true position using the metric ruler and hairline you built in step 2. You can measure the analog voltage using a DVM. The **`ADC`** project will print out the results of the analog to digital conversion. Make a three-column six-row table for this data. As an option, you could also disconnect the potentiometer from the circuit and measure the resistance using an ohmmeter for each position as well. Plot the position as a function of ADC sample. Does the response have a 1-1 correspondence? Is the response linear? These six measurements could become the data for the `Xtable` `Ytable` method described above.

5. Write the DAS software required for this lab. In particular review the procedure sections. A "syntax-error-free" hardcopy listing for the software is required as preparation. The TA will check off your listing at the beginning of the lab period. You are required to do your editing before lab. The debugging will be done during lab. Document clearly the operation of the routines.

Procedure (do this during your lab period)

1. *System checkout*: Run your hardware/software system to verify operation.

2. *Accuracy determination*: Collect 10 data points equally spaced throughout the range. Again the true position will be determined using the metric ruler and hairline. The measured position is defined as the value on the LCD display. Calculate average accuracy, maximum error, standard error, average accuracy of reading, average accuracy of full scale, maximum accuracy of reading, and maximum accuracy of full scale.

3. *Reproducibility determination*: Place the slide potentiometer at the center position and use your system to measure position on three separate days. Calculate the drift as the maximum difference between these three measurements

Deliverables (exact components of the lab report)
A) Objectives (1/2 page maximum)
B) Hardware Design
 Detailed circuit diagram all external circuits interfaced to the microcomputer
C) Software Design (no software printout in the report)
 Draw a data flow graph of the software system
 Draw a call graph of the software system
D) Measurement Data
 Give the accuracy measurements including details of how the data was obtained (procedure 2)
 Give the reproducibility measurements including details of how the data was obtained (procedure 3)
E) Analysis and Discussion (1 page maximum)

Checkout (show this to the TA)

You should be able to demonstrate the proper operation of position measurement instrument. Connect a DVM to the analog signal and be prepared to discuss with the TA the various aspects of the mechanical, electrical and software design.

Your software files will be copied onto the TA's zip drive during checkout.

Extra credit

For 10 points extra credit, design a new LCD device driver that implements real-time graphics. The vertical resolution will only be 7 pixels. Plot position on the vertical axis and time on the horizontal axis. Collect position data at 1 Hz in real time as a background thread. Pass the data through a FIFO to the foreground where it is displayed graphically on the LCD. You can choose to implement either sweep or scroll mode. This extra credit can be performed at time before the last class day.