Lab 4g Position Data Acquisition System and Interrupting SCI


Goals

- Study ADC conversion,
- Develop an interrupt-driven SCI device driver,
- Develop a real-time position measurement system using a slide potentiometer.

Review

- Operation of the 6812 ADC system in the Technical Data on 9S12C32 manual,
- Valvano Section 11.10 on the 6812 ADC, Section 12.1 on DAS parameters,
  Subsection 12.2.4 on position sensors.

Starter files

- RTI SCIA and ADC projects

Background

You will design a position meter with a range of about 3 cm. A linear slide potentiometer converts position into resistance (0<R<R max). 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 4.1. The 6812 ADC will convert voltage into a 10-bit digital number. Your software will calculate position from the ADC sample as a decimal fixed-point number. The position measurements will be displayed on the PC via the SCI-COM HyperTerminal interface. You will use interrupting I/O for the SCI 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 output display software. 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 0.001 cm.) The 6812 assembly language etbl instruction is an efficient mechanism to perform the interpolation. The etbl.RTF assembly program included with TExaS is an example of a piece-wise linear interpolation using the etbl 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 etbl 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, with the exception that you are not allowed to use a simple linear equation. You may use C, assembly, or a mixture of C/assembly.

The 10-bit ADC converters on the 9S12C32 are successive approximation devices with a short conversion time. You need to enable the ADC in ATDCTL2. You can define the number of ADC conversions (1 to 8) in a sequence using ATDCTL3. You specify the ADC clock in ATDCTL4, which will determine the time to perform an ADC conversion. Writing to the ADC Control register (ADCTL5) begins a conversion. The ADC chip clocks itself.

Jonathan W. Valvano
After the first sample is complete, \texttt{CCF0} is set and the result can be read out of the first result register, \texttt{ATDDR0}. After the entire sequence has been converted, the \texttt{SCF} bit is set.

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 \( \Delta \) of 0.001 cm. You should display the results in the HyperTerminal window on the PC, including units. No floating point is allowed. RTI interrupts will be used to sample the ADC in a background thread. This high priority interrupt will establish the sampling rate.

\textbf{Nyquist Theorem}: If \( f_{\text{max}} \) is the largest frequency component of the analog signal, then you must sample more than twice \( f_{\text{max}} \) 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 \( \phi \) from the digital samples.

\textbf{Valvano Postulate}: If \( f_{\text{max}} \) is the largest frequency component of the analog signal, then you must sample more than ten times \( f_{\text{max}} \) in order for the reconstructed digital samples to look like the original signal when plotted on a voltage versus time graph.

\textbf{Preparation (do this before your lab period)}

1. Review the technical information on the ADC system of 6812. 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. Typically, all that is required here is soldering three wires to the potentiometer, and connecting the three wires to +5V, ADC input, and ground. Make sure the +5V and ground connections go to the two potentiometer leads that are the fixed resistance. The center variable lead is connected to the ADC input, as shown in Figure 3.1.

4. Load and run the \texttt{ADC} project. You can visualize the \texttt{Data} result using the debugger, without having to connect the LCD display that this project normally uses. 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 \texttt{ADC} project will set the \texttt{Data} variable to the results of the analog to digital conversion. Make a three-column (position, voltage, and ADC sample) 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 \texttt{Xtable Ytable} method described above.

5. Load and run the \texttt{SCIA} project. Learn how to use HyperTerminal together with the 9S12C32 board. This system uses interrupt synchronization for both input and output. Rename the project \texttt{SCIB} and modify the driver software (\texttt{SCIB.c} and \texttt{SCIB.h}) and the main program so they implement just SCI output. i.e., remove all the SCI input routines. You should also remove the \texttt{RxFifo} module.

6. Combine the \texttt{RTI ADC} and \texttt{SCIB} projects to create the real-time DAS software required for this lab. Within the RTI interrupt handler, you will start the ADC, wait for the ADC sample, convert the sample to fixed-point position, convert the fixed-point to an ASCII string (Lab 1), and output the string using \texttt{SCI_OutString}. Include units in the output. You will have to modify \texttt{SCI_OutChar} so it does not crash when the \texttt{TxFifo} is full. In addition, you will also have to slow down the ADC sampling rate and/or increase the SCI baud rate so the \texttt{TxFifo} never becomes full. 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. Figure 4.2 shows the data flow graph of the position data acquisition system.

Jonathan W. Valvano
Figure 4.2. Data flows from the timer and the sensor through the SCI port to the PC.

Figure 4.3 shows a possible call graph of the system. Dividing the system into modules allows for concurrent development and eases the reuse of code.

Procedure (do this during your lab period)
1. **System checkout**: Run your hardware/software system to verify operation.
2. **Real-time**: One of the critical factors when designing a real-time system is the maximum time the software runs with interrupts disabled. Use one of the techniques you learned in Lab 2 to measure the worst case (longest) execution time of the RTI handler.
3. **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 PC HyperTerminal window. 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.
4. **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)
   If you organized the system different than Figure 4.2 and 4.3, then draw its data flow and call graphs
D) Measurement Data
   Give the software execution times of the RTI interrupt handler (procedure 2)
   Give the accuracy measurements including details of how the data was obtained (procedure 3)

Jonathan W. Valvano
Give the reproducibility measurements including details of how the data was obtained (procedure 4)

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.