Implementing a Rudimentary Oscilloscope

Objectives

The purpose of this lab is to become familiar with the 68HC11 on chip Analog-to-Digital converter. This lab builds on the knowledge gained from previous labs to send characters and numbers to the LCD display. In part 1 of the lab, DC input voltages in 0V - +5V range are converted and displayed. In part 2, you will implement an oscilloscope using the 4 lines of the LCD to display DC and very low frequency input signals.

Turn-In Requirements:

1. Pages 2, 3, 4, 5 and answers to the 5 questions as specified in the lab policies handout.

References


Equipment for this lab:

- 68HC11 trainer kit, to include 68HC11 EVBU and prototyping strips.
- IBM compatible PC to connect to the trainer kit via an RS-232 serial cable.
- Agilent 33250A signal generator.
- Agilent 54621D oscilloscope.
In completing this lab, I have abstained from any form of academic dishonesty or deception (e.g. cheating, lying, stealing or plagiarism). This work is entirely of my own origin (beyond the code that was supplied to me by the instructor).

Name: __________________________            Signature: ___________________________

Laboratory Exercise

Notes

• Make sure your development EVBU board is connected to power (green LED on board ON) and the serial port of the EVBU is connected to the serial port of your development PC containing the AXIDE software, configured to the correct port at the correct baud rate, etc.

• Copy the file “oscilloscope.RTF” from the “EE3301 Resources web site” to your floppy disk and modify.

• The lab TA has a copy of the oscilloscope program and you can request a demo.

Part 1

In this part, you will program the A/D converter to read inputs PE0 – PE3 and output the result to the LCD display. The 68HC11 A/D system is an 8 bit successive approximation converter with an input multiplexer. The A/D converter requires two reference voltages, $V_{RL}$ and $V_{RH}$, and on the Axiom board, they are connected to 0V and +5V, respectively. A conversion requires 32-E clock cycles to complete. Before you can use the A/D converter, it must be configured by writing the appropriate values to the two control registers “OPTIONS” and “ADCTL.” The “OPTIONS” register has a bit to turn the power on to the A/D converter and another bit to select the clock input to the converter. Since the clock frequency of the Axiom board is 8MHz, you can choose the E-clock as the source for the A/D converter. The ADCTL register selects the mode of operation, channel control, and channel selection. It also contains a flag (bit) to indicate the end of the conversion and the results of the conversions are in registers ADR1 – ADR4.

You are going to connect an external signal to the A/D converter and display the voltages on the LCD screen. Negative inputs and large inputs driving too much current can damage the A/D converter inputs. Therefore, use the following circuit to protect the inputs from negative voltages and excessive current. Make sure your input signal is positive and doesn’t exceed +5V. Your TA MUST verify inputs signals before you can connect to the A/D inputs. Use the Agilent 33250A signal generator as the external source. To generate DC voltages, push the “Utility” button and then select “DC on” form the display. Make sure the output is in the high impedance state by pushing the “Utility” button and then selecting “Output Setup”, “High Z” and “Done” from the display. If the output is in 50Ω mode, the actual output will be twice the value indicated by the signal generator.

Program the A/D to convert the channels PE0 – PE3 once with results being stored in registers ADR1 – ADR3. Fill in the required bits in the OPTIONS and ADCTL registers shown below.

![Protective Circuit Diagram]

Figure 1: Protective Circuit

Program the A/D to convert the channels PE0 – PE3 once with results being stored in registers ADR1 – ADR3. Fill in the required bits in the OPTIONS and ADCTL registers shown below.
Modify the given program (oscilloscope.RTF) to continuously digitize and display the input channels PE0 – PE3 in the following format.

\[
\begin{align*}
\text{PE0 A/D} &= \text{aaa} = \text{mmmm} \text{ mV} \\
\text{PE1 A/D} &= \text{aaa} = \text{mmmm} \text{ mV} \\
\text{PE2 A/D} &= \text{aaa} = \text{mmmm} \text{ mV} \\
\text{PE3 A/D} &= \text{aaa} = \text{mmmm} \text{ mV}
\end{align*}
\]

aaa is the A/D reading and mmmm is the corresponding voltage in mV. Compute the input voltage (mmmm) from the A/D reading (aaa) for display. Note that A/D produces 00_{10} for 0 V input and 255_{10} for +5V input.

Apply a DC signal from the Agilent 33250A function generator to each of the inputs PE0 – PE3, one at a time, and demonstrate to the lab TA that your program is working. Fill in the columns for the A/D reading and the displayed voltage in the following table for the given input signals.

<table>
<thead>
<tr>
<th>SIGNAL GENERATOR READING</th>
<th>A/D READING</th>
<th>DISPLAYED VOLTAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>200 mV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1250 mV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2000 mV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3500 mV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4500 mV</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

T. A. Initials:______________________

Part 2

In this part you will display the input signal applied to one of the A/D inputs in the form of a bar graph to form a rudimentary oscilloscope.

Program the A/D converter for continuous conversion of PE0 input channel. Fill in the bits of the ADCTL register shown below.

\[
\begin{array}{cccccccc}
\text{CCF} & - & \text{SCAN} & \text{MULT} & \text{CD} & \text{CC} & \text{CB} & \text{CA} \\
X & X & & & & & &
\end{array}
\]

The following (inefficient) algorithm is suggested for your program. However, you are free to use any other algorithm.
• Allocate 4 memory locations to store 4 consecutive A/D conversion results. Let these locations be AD1, AD2, AD3 and AD4.

• Step 1: Read a new A/D conversion result.

• Copy the content of location AD2 to AD1, AD3 to AD2, AD4 to AD3 and copy the new A/D reading into location AD4. Note that the content of location AD1 is lost.

• Display the value of location AD1 – AD4 to LCD display line 1 – line 4. Use a properly scaled bar to display each A/D reading. See the note below.

• Go back to step 1.

Note: The length of the bar is proportional to the A/D reading. A +5V input should fill the 20 characters in a line. Use the character “black block” ($ff) to construct the bar. The rest of the line should be filled with spaces. For example, +2.5V DC should produce a display similar to the following.

![Bar Graph Example]

The snapshot of the LCD for a ramp input should look like the following:

![LCD Display Example]

Add a suitable delay between printing each line, so that the shape of the waveform is captured. If the display is updated too fast or too slow, you may not be able to see the shape of the input signal.

Do the following:

1. Apply a DC signal, and adjust the voltage from 0V – +5V. Your program should display a bar graph proportional to the magnitude of the input. Demonstrate to the T. A. that you program is running.

   T. A. Initials: _______________________

2. Generate a 0.5Hz, +1V – +4V square waveform and apply it to input PE0. You can use the Agilent 54621D oscilloscope to monitor the input waveform. Verify that the program is displaying the input waveform correctly. You may have to vary the frequency to get a good display. Demonstrate to the T. A. that you program is running.

   T. A. Initials: _______________________

3. Generate a 0.5Hz, +1V – +4V 50% triangular waveform and apply it to input PE0. You can use the Agilent 54621D oscilloscope to monitor the input waveform. Verify that the program is displaying the input waveform correctly. You may have to vary the frequency to get a good display. Demonstrate to the T. A. that you program is running.
Do the Following to document your program (Note: comments, structure, and readability counts!!)

1. Sketch the waveform you observed on LCD in parts 2 and 3 above. One sketch for each waveform is sufficient.

2. Hand-in a complete *Flowchart* and the entire program for parts 1 and 2.

3. Hand in the listing of the complete, well-commented source-code that *matches the flowchart*. Remember, if the TA can not read your code with “reasonable” effort, then it is wrong!

When you turn in the lab report, please answer the following questions. Provide flow charts and/or assembly language programs as necessary to explain your answers.

Questions

1. Why is the 68HC11 A/D converter called a “successive approximation” converter? How many external inputs can be connected to it?

2. If a 0 - +5V signal is applied to
   (i) 8 bit A/D converter
   (ii)12 bit A/D converter
   (iii)16 bit A/D converter
   what are the smallest voltage step sizes that can be detected?

3. Which bit selects the clock source for the A/D converter? What is the reason for switching to an RC oscillator if the system clock frequency is too low. What are the disadvantages of using the RC oscillator?

4. In part 1, there are discrepancies between the signal generator reading and the displayed voltage. What are the possible sources of error?

5. The algorithm of part 2 to implement the oscilloscope is very inefficient if the number of LCD lines is large. Why is it inefficient? Give an efficient algorithm. Hint: think of the display as a window of a circular buffer.