Laboratory 7 Analog-to-Digital Converter

Last Revised: August 1999

1 Objectives

With the completion of this lab, you should be able to:

- Design and construct a circuit to interface an ADC to the MUN-88 microcomputer.
- Write assembly programs to capture and store samples from an analog signal.

2 Introduction

Analog-to-Digital Converters (ADC) and Digital-to-Analog Converters (DAC) are extremely useful devices which enable us to interface the real world events (which are frequently analog) with microprocessors (which are inexpensive and provide good precision and accuracy when compared to their analog counterparts) for monitoring or control.

There exist many types of ADCs and DACs differing in their internal construction, resolution, precision and accuracy, speed, compatibility with different micros, and other capabilities like power supply requirements, unipolar/bipolar operation and coding. In this lab, you will interface an ADC to the MUN-88 and test its operation.

Caution!

The ADC device that you will be using in this lab is sensitive to electrostatic fields and can be easily damaged. **Do not insert these devices into powered sockets.** Remove power before insertion or removal. Do not touch the pins of these devices with your fingers. It is a good practice to switch off the power supply before you make any change to the hardware.

These devices are expensive – please be careful not to ruin any chips.

Measure the exact value of the resistors or the capacitors with an LCR meter before connecting these devices. This precaution may eliminate hours of frustration and debugging.

Note that, in this lab, a reference to a signal line/pin that is active low will have the * symbol after its name (for example, CS* or IOWR*).

3 Analog-to-Digital Converter

You should have received a copy of the data sheet of AD7574 – if not, it is available on the course web site in a PDF file. Read it fully before designing your circuit. This device is a simple 8-bit ADC based on the successive approximation technique.

3-1 Circuit Diagram

This device should be interfaced to the MUN-88 as an input port; the IOSEL6* and IORD* signals may be used to achieve this. You will be using the AD7574 to obtain **unipolar binary** operation in **ROM interface mode**. Besides the +5V power supply, a -6V power supply should be used as reference voltage V_{ref}; this can be easily achieved by connecting two 1k ohm resistors between the -12V line and the ground, and connecting the middle point to pin 2 of AD7574.

Based on Figures 8a & 8b of the AD7574 data sheet, what will be the input voltage range of the ADC used in this lab?

The AD7574 contains an internal clock oscillator, which is active when the ADC converts the input analog voltage to a digital value. The oscillator requires an external resistor and capacitor. A 100pF (or 130pF) capacitor and a 200k ohm (180k - 220k) resistor can be used as in Figure 6. This combination determines the conversion time of the ADC (Figure 7a). **What is the estimated conversion time?**

Use a 74LS32 to gate IORD* and IOSEL6* signals, and connect the output of this OR gate to the RD* line of the AD7574. Connect the CS* line of the AD7574 to ground, thus permanently enabling the ADC. Notice (*Figure 2*) that the data lines of the AD7574 are always tristated (high Z, that is high impedance) except when RD* is active low. The analog input to the AD7574 may be obtained using a signal generator. Remember to use a 1k ohm resistor in series with this input.

Sketch a circuit diagram showing all the connections required for the ADC device to operate in the required mode.

3-2 Circuit Wiring

Switch OFF all power supplies. Wire the circuit as in your diagram. Do not connect the signal generator output. Make sure that all the ground points are connected together. Your partner should verify the connections.

Set the signal generator to produce a +5V DC. Check the output of the signal generator with a DMM or CRO. Turn ON the power supply ***and then*** connect the signal generator output to the input of the AD7574. Using MUN-88 monitor commands, check if the ADC operates properly. When this device is used in the ROM interface mode, the first ***read*** produces an incorrect result; subsequent ***read*** operations will produce correct results (see Figure 2). **Reduce the input voltage (in steps) from 5V to 0V and find the converted binary value of the input signal in each case.**

3-3 Signal Generation

Remove the connection between the output of the signal generator and the ADC. Set the signal generator to produce a 500Hz sinusoidal waveform with a peak-to-peak voltage of 5V. Adjust the DC offset of the signal generator to ensure that its output never falls below 0V. This is required because you have configured the ADC for unipolar operation. Check your signal on a CRO, and apply this signal to the ADC. Try out the monitor command read the ADC output. You should obtain different numbers.

3-4 Software

1. Write an assembler program that repeatedly reads the ADC output until a switch signal is obtained. Do not attempt to store the results in the memory because the RAM in the MUN-88 is limited. Include a small delay before looping back. This delay is required because the ADC requires several microseconds to perform the conversion.

Run this program and observe the ***Busy**** output line of the ADC on the oscilloscope. Trim the delay routine in your program to obtain the fastest possible conversion rate. Discuss your results.

2. Write an 8088 assembler program that reads the output of the ADC and stores this result in memory. Make this operation repeat 256 times. Remember that the first result is useless, and should not be stored. Also remember to include a small delay before you loop back – if the time to execute the loop is different from the first program, you will have to adjust the delay accordingly. Run this program and observe the stored values. Report on your observations. ***Demonstrate correct operation of this program to a TA, who will sign your list file.***
   Based on the instruction execution time, determine what input sinusoid frequency is necessary so that exactly one period is captured and stored by your program. Set the signal generator appropriately, and report on your success when you run your program.

4 Submission

At the end of the lab submit print-outs of your ***list files***. Include circuit diagrams, answers to all questions in the lab, and test results.

Before leaving the lab, deconstruct your circuit and return your components to the provided kits.