Lab 22 Temperature 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, Nyquist Theorem, Valvano Postulate,
- Develop a temperature measurement system using a thermistor,
- Design and implement various linear and nonlinear digital filters.

Review
- Operation of the 6812 ADC system in the Technical Data on MC68HC812A4 manual,
- Data sheets on the Texas Instruments TLC2274 four op amp IC,
- Valvano Chapters 11,12,15 on thermistors, analog amplifiers, ADC's, data acquisition systems, and digital filters.

Starter files
- OC3.H, OC3.C, OC3TEST.C, ADTEST.C, DAS.C

Background

Figure 22.1 shows an analog signal. The name refers to the fact that such signals are analogous to physical quantities that they represent.

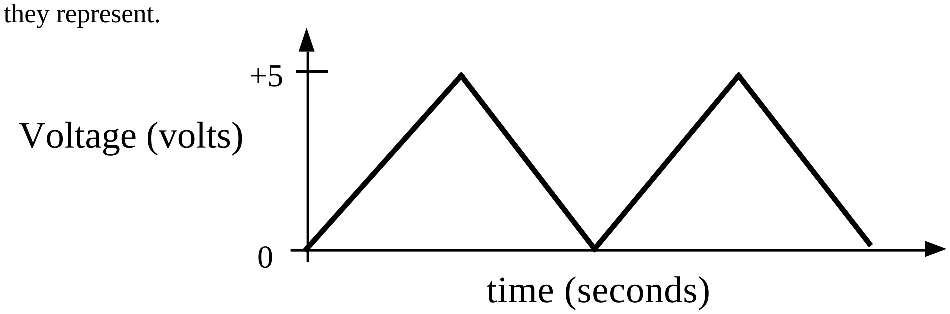

Figure 22.1. A Continuous-Time Analog Waveform.

Figure 22.2 shows a representation of the signal in terms of discrete-time samples. Since the signal of Figure 22.2 is defined only at the sampling times, it is a discrete-time signal. However, if the magnitude of each sample can still take any value in a continuous range, then the samples in Figure 22.2 are still analog.

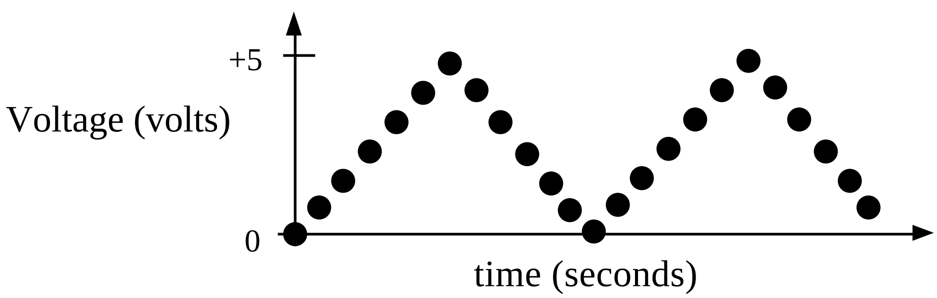

Figure 22.2. A Discrete-Time Analog Waveform.

If the magnitude of each of the signal samples in Figure 22.2 is represented by a finite-precision digital number (in our case, 0 to 255) then the signal is no longer continuous but is instead quantized, discretized, or digitized. The process of converting a sample into a number is called analog-to-digital or ADC conversion. If the continuous signal of Figure 22.1 is sampled at a sufficiently high rate the digitized samples could be output through a digital-to-analog converter to reproduce the waveform in Figure 22.1 as shown in Figure 22.3.

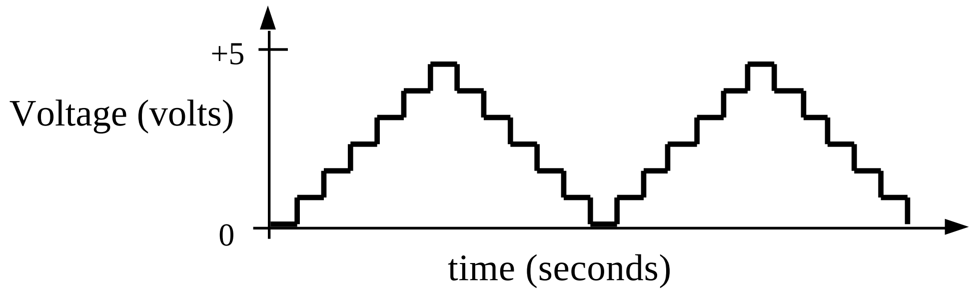

Figure 22.3. DAC conversion of the samples in Figure 22.2.

This experiment will use one of the 8-bit ADC converters on the 6812 to construct a digital thermometer. You are free to choose any the temperature range that suites your experimental conditions (e.g., 20 to 40 ˚C). If the current temperature is above the upper limit in the specified range, a red LED should be turned on. You can test this feature by shorting the thermistor leads together (zero resistance.) If it is below the lower limit of the specified range, a yellow LED should be turned on. Similarly, you can test this feature by disconnecting one wire of the thermistor (infinite resistance.) Otherwise, a green LED will stay on indicating the temperature is within the specified range. The current temperature will be displayed on the LCD (see the example in LCDTEST.C, LCD12.H, LCD12.C) or on the PC terminal port using the serial port routines developed in an earlier lab (your choice). The resolution of temperature measurement will be about 1 ˚C. There are two components of the signal. The temperature component is 0 to 1 Hz. In addition to temperature, we will be processing 60 Hz noise with a digital notch filter. So, the overall frequencies of interest are 0 to 60 Hz.

Figure 22.4 shows a typical multichannel data acquisition system. The transducer converts a signal into a resistance. The bridge converts the resistance into a voltage. The amplifier output matches the full-scale range of the ADC. A low-pass analog filter, with a cutoff frequency of about 1/2 the sampling rate, removes high frequencies that might otherwise cause aliasing. A low-pass analog filter will not be implemented in this lab.

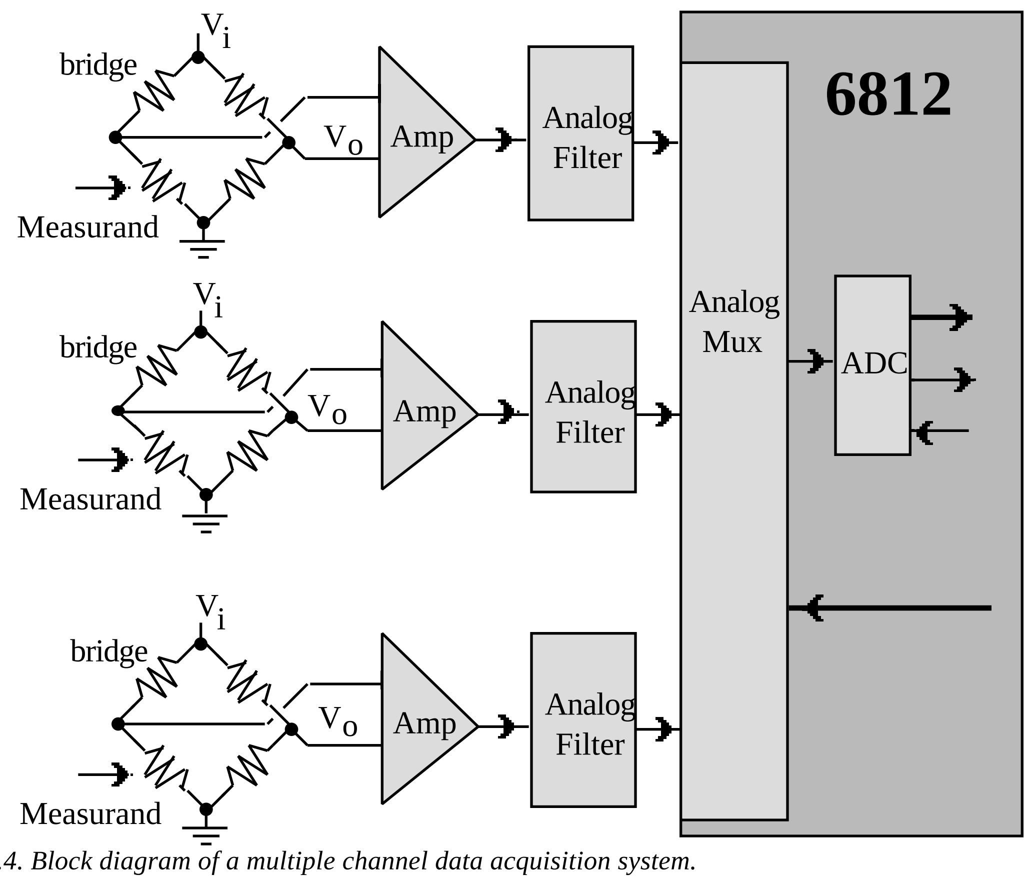

Figure 22.4. Block diagram of a multiple channel data acquisition system.

The 8-bit ADC Converters on the 6812 are successive approximation devices with a short conversion time. The ritual should initialize the ATDCTL2 ATDCTL3 and ATDCTL4 registers. Except for the ADPU bit in the ATDCTL2 register (back ADPU=1 to activate the ADC), you can use the default values for these three registers. Writing to the ADC Control register (ADCTL5) begins a conversion. To sample a single channel 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 possibility is

```
void ADCInit(void){
   ATDCTL2 = 0x80;   // Activate ADC
   ATDCTL3=0;        // no freeze
   ATDCTL4=1;}       // sample time 2 E clocks, /4 clock
unsigned char ADCSample(unsigned char chan){
    ATDCTL5=chan;    // Start ADC
    while ((ATDSTAT & 0x01) == 0){};  // wait for CCF0
    return(ADR0H); }
```

1. Temperature-Resistance Calibration of the Thermistor:

The thermistor resistance varies nonlinearly with its temperature. It is very important to use temperature units of Kelvin in this equation and not ˚C.

$$R = R_0\, e^{+\beta/T} \qquad \text{(where T is temperature in degrees Kelvin)}$$

The thermistors in this lab have a resistance of about 500 kΩ at 25 ˚C. Perform a very crude temperature calibration experiment with three points somewhere in your temperature range (done as part of a previous lab). For example, the skin temperature in the axilla region (arm pit) is about 36˚C=309K, and the room temperature is about 25 ˚C. Use simple algebra to determine R_0 and β from the three calibration points.

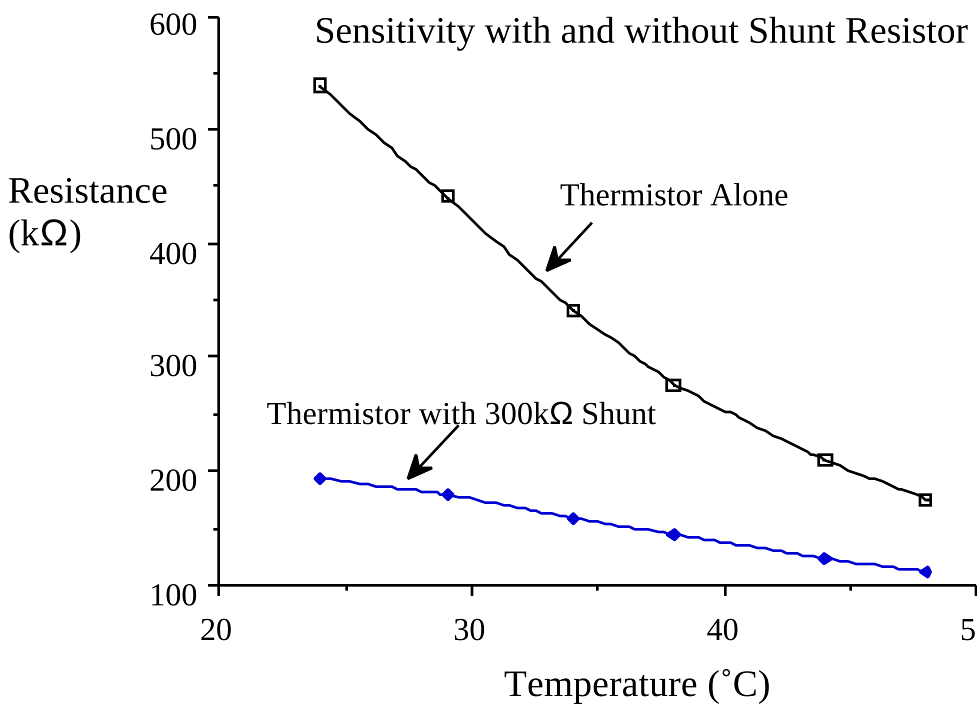

Figure 22.5. Response with and without a 300 kΩ shunt resistor.

One could (but you will not) place a shunt resistor in parallel with the thermistor so that the R versus T response is more linear. A fixed resistor placed in parallel with the thermistor will reduce the sensitivity, but increase the linearity of the combination. The parallel combination of a shunt resistor and a thermistor will have an S-shaped resistance versus temperature response. A typical R versus T response is shown in Figure 22.5. To place the inflection point of the S-curve at the midpoint of the temperature range, let

$$R_p = R_m (\beta - 2\, T_m)/(\beta + 2\, T_m)$$

where T_m is the midpoint temperature (K) (e.g., if 24≤T≤48˚C, then T_m=36˚C=309K),
β is the thermistor value calculated above (K),

R_p is the resistance of parallel shunt resistor (Ω),
R_m is the thermistor resistance at T_m (Ω).

NOTE: IN THIS LAB YOU WILL NOT ADD A SHUNT RESISTOR! This is because one of the lab's objectives is to evaluate software techniques to deal with the nonlinearity.

2. Choose a sampling rate

In this lab we will process temperature signals (0 to 1 Hz) and 60 Hz noise in our digital samples. According to the Nyquist Theorem, we need a sampling rate greater than 120 Hz. In order to implement a *simple* FIR 60 reject filter you will need to choose the sampling rate to be a multiple of 120 Hz. Therefore, you may select 240, 360, 480, or 960 Hz (it doesn't really matter, just pick one.) Output compare interrupts will be used to sample the ADC in a background thread. This high priority interrupt will establish the sampling rate. This selection of sampling frequency will affect the design of the analog and digital filters. So, if you change the sampling rate, you will have to redesign all the filters.

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

$$A + B \sin(2\pi ft + \varphi)$$

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.

3. Hardware Interface

Figure 22.6 shows one possibility for the analog electronics of the digital thermometer. You may consider using the 3-op-amp instrumentation amplifier instead of the single op amp subtractor. YOU DO NOT NEED TO ADD AN ANALOG FILTER IN THIS LAB. The amplifier should convert the entire temperature range into the 0 to +5 V ADC range. You will add a 60 Hz digital notch filter. Because you are using rail-to-rail op amps, the entire system can be powered by a single +5 V supply. PLEASE DO NOT USE +12 OR –12 V SUPPLIES IN THIS LAB.

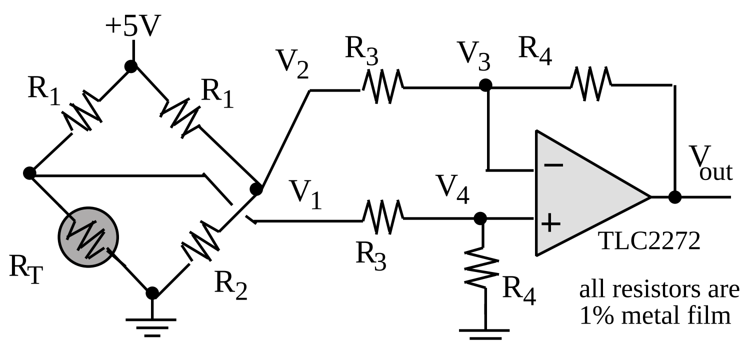

Figure 22.6. Possible thermistor interface.

Normally, the R1 resistor in the bridge is chosen large enough to prevent self-heating the thermistor. Assume the dissipation constant to be about 1 mW/˚C. Limit the thermistor power to 0.1 mW so that the self-heating error is below 0.1 ˚C. For our 500 kΩ thermistor and +5V supply any value of R1 around 500 kΩ will be OK. The R2 resistor in the bridge establishes one of the extreme point values of the temperature range. For example, when the thermistor resistance, R, equals R2, then the bridge output voltage is 0. The gain of the differential amplifier, along with the ADC range (0 to +5 V in our case) will determine the temperature range of the system.

4. Software Conversion:

Using the calibration data, the nonlinear thermistor equation, the characteristics of your analog circuit and the response of the 6812's ADC, determine the ADC output sample for each temperature for about 5 to 10 temperature points within your selected temperature range. Show both, a table of figures and a plot of this data. Include appropriate intermediate voltages in the table (e.g., thermistor resistance, bridge output, and analog circuit output.) Design a software conversion routine that calculates temperature from the ADC sample. You should consider various methods:

a) linear equation (don't use because it has errors too large),
b) nonlinear equation,
c) large table lookup (one entry for each ADC value, i.e., 256 entries),
d) small table lookup (≈16 entries) with linear interpolation in between.

The 6812 has special table lookup and interpolation op codes, so you may wish to consider writing the conversion in assembly language. See the TBL.RTF and ETBL.RTF examples on the TExaS simulator.

5. Software:

The program will continuously sample the ADC at a sampling frequency, f_S, selected in part 2. The sampling and digital filter functions of the real time data acquisition software must be implemented using interrupts. The software needs to calculate temperature from the ADC sample. The measured temperature is displayed either on the LCD display or the PC terminal window (using the interrupt driven SCI12A.c driver), and checked against your specified temperature range and appropriate LED's turned on and off. If you output the temperatures to the PC terminal window, sample and calculate temperature at the sampling rate, but only output to the PC once a second. Make sure the PC output rate is slow enough and the TxFifo size is big enough so that the TxFifo never fills. The main or foreground thread executes in a keyboard interpreter. When a key is hit on the keyboard, the main program should interpret the key. The temperature calculation and display processes must be performed in the background using interrupts. The user should be able to choose one of these digital filters from the keyboard.

1. A simple FIR digital filter that eliminates 60 Hz (linear)

$$y(n) = \frac{x(n)+x(n-k)}{2} \qquad \text{where } f_S = 120 \bullet k \text{ Hz and k is an integer} \geq 2$$

2. y(n) = median[x(n), x(n-1), x(n-2), x(n-3), x(n-4)] (nonlinear)
3. A High-Q IIR digital notch filter that eliminates 60 Hz (linear)

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. Choose one of the options as discussed in hardware section and design the appropriate thermistor amplifier. Be prepared during checkout to discuss the reasons for your choice of design. FOR THIS LAB YOU MAY SKIP the low-pass anti-aliasing analog filter. Show name and number of all the pins involved including power. Add bypass capacitors on all chips. Why is it is important to connect bypass capacitors across the power pins for the analog IC components? Label all resistance and capacitance values and types. For example, 1kΩ 5% carbon, or 0.01μF 5% ceramic.

3. Design the simple FIR 60 Hz digital reject filter. The sampling rate was chosen previously. Give the Z transform of the filter. Develop an equation for the gain versus frequency. Create a table showing the expected digital filter response.

4. Design a High-Q 60 Hz IIR digital notch filter. YOU MAY NOT IMPLEMENT A FILTER EXACTLY EQUAL TO ANY EQUATION FROM THE LECTURE NOTES. You are expected to design a similar filter. You may choose a different sampling frequency and/or α value, and work through the design steps. Be sure to add gain correction, so that the DC gain is 1. Include a plot of the expected frequency response of the digital filter.

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. *Basic understanding*: The purpose of this section is to verify the **Nyquist Theorem** and the **Valvano Postulate**. Generate a continuous waveform (0 to +5V) with an adjustable frequency from 10 Hz to 10 kHz. Consider using a function generator, or the capacitor voltage of a 555 timer. First connect the analog waveform to a scope and verify the voltage range is between 0 and +5V. VOLTAGES OUTSIDE THIS RANGE WILL DAMAGE THE 6812. Next connect the signal to 6812 port ATD1 as shown in Fig. 22.7. In this part, we will not be using

the thermistor or analog amplifier. The MAX549A DAC will be interfaced to the SPI port like in the previous lab. Connect the DAC output to the other scope channel. We will test the **Nyquist Theorem** and the **Valvano Postulate** using the following program. Assume the function `MAX549Open()` initializes the DAC interface, and `MAX549DOUTA` outputs an 8-bit value to channel A of the MAX549A. Be careful not to violate the 0 to +5V 6812 input range.

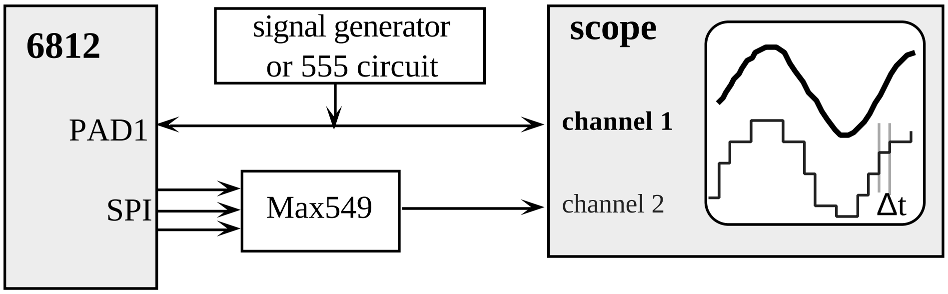

Figure 22.7. Connect a signal generator to the 6812 ADC channel 1, and to one channel of a scope. Connect the DAC output to the other scope channel.

```
#define C5 0x20
#define rate 5556
#pragma interrupt_handler TOC5handler()
void TOC5handler(void){ unsigned char data;
  TFLG1=C5;       // ack C5F
  TC5=TC5+rate;   // fs=360Hz (change this to match your sampling rate)
  data = ADCSample(1);  // new data from ADC channel 1
// put digital filter here for procedure part 2
  MAX549DOUTA(data);}
void OpenClock(void) {
asm(" sei");     // make atomic
  TIOS|=C5;      // enable OC5
  TSCR|=0x80;    // enable
  TMSK2=0x32;    // 500 ns clock
  TMSK1|=OC5;    // Arm output compare 5
  TFLG1=C5;      // Initially clear C5F
  TC5=TCNT+rate;
asm(" cli"); }
void main(void){   // MAX549A DAC interface from last lab, Input to 6812 ATD1
   MAX549Open();   // software from a previous lab
   ADCInit();      // turn on ADC
   OpenClock();    // start background thread
   while(1){};     // copy ADC to DAC in background
}
```

Sketch the two waveforms in your lab notebook. Use the logic analyzer (or a regular scope) to verify the ADC sampling rate. In particular, measure the Δt time step in the DAC output waveform, as shown in Fig. 22.7. Describe what happens as the input waveform frequency goes from 10 Hz to 10kHz. Be prepared to explain your results during checkout.

2. *Digital Filter Performance*: Experimentally determine the gain versus frequency responses of your digital filters. Perform the digital filter on the ADC samples, output the filter results to the DAC. Run the ADC, digital filter, DAC at your chosen sampling rate. Just like part 1, put both the ADC input voltage and the DAC output voltage on a dual channel scope, i.e., Fig. 22.7. Vary the input frequency from 0 to 1/2 the sampling rate and plot the gain versus frequency response of your three digital filters (be careful not to violate the 0 +5V 6812 input range). In particular, test your 60 Hz notch filters by connecting input at frequencies near 60 Hz, e.g., 50,55,59,60,61,65,70 Hz. Test the filters first with a sine wave and then with a square wave. Compare the results. Compare to the expected results. Consider using a software scheme to calculate the amplitudes of the input and output waves. E.g.

```
unsigned char MaxX,MaxX,MinY,MinY,X,Y;
unsigned short Count;
void InitGain(void){
  MaxX=MaxY=0;
  MinX=MinY=255;
```

```
  Count=0;}
void MeasureGain(unsigned char x, unsigned char y){
  if(x>MaxX) MaxX=x;
  if(x<MinX) MinX=x;
  if(y>MaxY) MaxY=y;
  if(y<MinY) MinY=y;
  Count++;        // change 360 to match your sampling rate
  if(Count==360){ // calculate once a second
    X=MaxX-MinX;  // amplitude of filter input
    Y=MaxY-MinY;  // amplitude of filter output
    InitGain();   // setup for next time
  }
}
void TOC5handler(void){ unsigned char data;
  TFLG1=C5;       // ack C5F
  TC5=TC5+rate;   // fs=360Hz (change rate to match your sampling rate)
  data = ADCSample(1);  // new data from ADC channel 1
// put your digital filter here
  MeasureGain(x[0],y[0]);
  MAX549DOUTA(data);}
```

3. *Static analog circuit test*: Perform these tests before connecting the circuit to the 6812. Construct and evaluate the thermistor circuit. Use 4 regular resistors that have resistances within the typical range of your thermistor. One should be the resistance at maximum temperature, and another should be the resistance at minimum temperature. Record the voltage values at strategic places in your analog circuit. What voltage output do you get when the thermistor is disconnected? What voltage output do you get when the thermistor wires are shorted?

4. (SKIP THIS PART) *Dynamic analog circuit test*: Again, perform these tests before connecting the circuit to the 6812. Disconnect the thermistor, and connect a sine-wave signal generator in its place. Make sure the voltage level of the signal generator is within range, so that the inputs and outputs of your analog circuit are not saturated. Record the sine-wave amplitudes of the input and output voltages. Start at about 10 Hz and collect measurements ten different frequencies. Make sure you choose frequencies large enough to see the gain roll off. Calculate the gain at each frequency. Plot the gain versus frequency response of your circuit.

5. *Analog-to-Digital Conversion*: Before connecting the input signal to ADC it should be verified that the range of the input signal is between +0 and +5 volts. Check the analog system with an open (infinite resistance) and short (zero resistance) thermistor connection. Connect the output of your thermistor amplifier to the input of the 6812 ADC system. Use your four fixed resistors and collect digital samples and software calculations of resistance and temperature. Add these three columns of data to the data collected in part 4.

6. *System checkout*: Run your hardware/software system to verify operation. Connect the output of your thermistor amplifier to the input of the 6812 ADC system and demonstrate the operation of the digital thermometer and LED's. Your temperature resolution should be about 1 degree C. For a chosen temperature range, a red LED should light up if the temperature is above the upper limit, a yellow LED should light up if the temperature is below the lower limit, and a green LED should light up if the temperature is within the temperature range.

Checkout (show this to the TA)

You should be able to demonstrate the proper operation of digital thermometer and LED's as described in the introduction. You can use a light bulb to heat the thermistor above 65˚C, but I suggest you stay away from ice or fire (it can get messy.)

Hints

1) This is a long lab with many parts, so start early.
2) Don't try to complete the experiment in one full swoop. Run the hardware test program given above before testing your software. Debug the system in an analytical, step-wise manner.
3) See Valvano Chapter 15 for information about digital filters.
4) The 6811 manual suggests a 10 kΩ series resistor to connect analog signals to the ADC. Please try this on our 6812's. Please do not use regular op amps that require ±12 V supplies. An advantage of the single supply op amp is that the output of your analog amplifier will remain within the 0 to +5V ADC range even when the thermistor is disconnected.

5) To design the 60Hz notch digital filter, similar but not identical to the filters in Chapter 15:
   1) read section on digital filters that contains equations 41-59,
   2) choose a sampling rate f_s, then calculate θ,
   3) choose an α different from 7/8 (less than 1),
   4) plug θ, α into equation 48, you should get an equation similar to equation 54,
   5) follow steps similar to equations 55-59,
   6) software implementation should be similar to the programs in this section.

6) The TLC2274 operates rail-to-rail, which means its output can swing all the way from 0 to +5 V. The TLC2274 can operate on a single +5V supply. *In fact, if you connect it up to the usual +12 –12 V supplies, you will damage the device.*

7) Provide a detailed schematic showing thermistor bridge, amplifier, LED driver, LEDs, and 6812. Show name and number of all pins including power, use bypass capacitors on all chips, label all resistor and capacitor values, types, and tolerances.

8) Show all equations and design calculations for all your component values.

9) Choose a sampling rate and show your design calculations for your IIR 60 Hz notch filter. YOU CAN'T USE THE EXACT FILTER FROM THE BOOK, use a different sampling rate or a different α. Give the z-transform for the FIR and IIR filters. Be sure to add gain correction so that the DC gain is 1. For the simple FIR filter, develop an equation for the gain versus. frequency and generate a table and plot for the equation.

10) For part 1 of the procedure in the lab manual, it is important that you sample the ADC and output to the DAC at the sampling rate that you will be using in the rest of the lab. This part does not use any analog or digital filters. Show both the raw input signal and the output from the DAC on the oscilloscope.

Design of a high precision thermometer (informational purposes only)

A precision thermometer, an ohmmeter, and a water bath are required to calibrate thermistor probes. The following empirical equation yields an accurate fit over a wide range of temperature:

$$T = \frac{1}{H_0 + H_1 \ln(R) + H_3 [\ln(R)]^3} - 273.15$$

where T is the temperature in ˚C, and R is the thermistor resistance in ohms. The cubic term was added in order to improve accuracy. It is preferable to use the ohmmeter function of the eventual instrument for calibration purposes so that influences of the resistance measurement hardware and software are incorporated into the calibration process.

The first step in the calibration process is to collect temperature (measured by a precision thermometer) and resistance data (measured by the ohmmeter process of the instrument). The thermistor(s) to be calibrated should be placed as close to the sensing element of the precision thermometer as possible. The water bath creates a stable yet controllable environment in which to perform the calibration. About 10 to 20 data points should be collected throughout the temperature range of interest.

The second step is to use nonlinear regression to determine H_0, H_1, and H_3 from the collected data. The following nonlinear transforms will linearize the problem

$$x = \ln R, \quad y = (\ln R)^3, \qquad z = \frac{1}{T+273.15}$$

where T is in ˚C. The problem is then transformed into the following linear equation

$$z = H_0 + H_1 \bullet x + H_3 \bullet y$$

Let n be the number of data points. Linear regression is used to determine the unknown coefficients from the x,y,z data. The following equations are used to calculate the coefficients H_0,H_1,H_3 from the calibration data.

$$A_{11} = \sum x^2 - \frac{(\sum x)^2}{n}$$

$$A_{22} = \sum y^2 - \frac{(\sum y)^2}{n}$$

$$A_{12} = \sum xy - \frac{\sum x \sum y}{n}$$

$$\Delta = A_{11} \bullet A_{22} - \left(A_{12}\right)^2$$

$$C_1 = \sum xz - \frac{\sum x \sum z}{n}$$

$$C_2 = \sum yz - \frac{\sum y \sum z}{n}$$

$$H_1 = \frac{A_{22} \bullet C_1 - A_{12} \bullet C_2}{\Delta}$$

$$H_3 = \frac{A_{11} \bullet C_2 - A_{12} \bullet C_1}{\Delta}$$

$$H_0 = \frac{\sum z}{n} - H_1 \bullet \frac{\sum x}{n} - H_3 \bullet \frac{\sum y}{n}$$