Using LabVIEW to Measure Temperature with a Thermistor

C. Briscoe and W. Dufee, University of Minnesota   November, 2009

For resources, see the LabVIEW Resources page on the UMN ME2011 course site.

Before starting this exercise, install LabVIEW and drivers following the instructions in the "LabVIEW Quick Start" document, and connect and test the USB-68008 data acquisition unit following the directions in the "Connecting the USB-68008" document. The screen shots shown in this document are from a Windows VISTA machine. Your setup may have slight differences.

Introduction

The purpose of the exercise is to give you some experience using LabVIEW for automatic, computer-based data collection from an experiment. You will be measuring the internal temperature of a grape or similar object as a function of time as it is plunged into a glass of ice water. Using the collected data, you will determine the time constant of the heat transfer dynamics that model the temperature drop.

What You Will Need

1. Computer running LabVIEW
2. National Instruments (NI) USB-68008 data acquisition unit
3. USB cable
4. Thermistor temperature sensor (NTC, 10K ohms, jameco.com, PN 207037)
5. Digital multi-meter that can read resistance
6. Solderless breadboard
7. 22 g solid wire
8. A 10K ohm resistor
9. A grape, or other small fruit, at room temperature
10. A glass of ice water

USB-6008   Thermistor

Thermistors

Thermistors are resistors that vary with temperature. The resistance R of a thermistor at temperature T can be modeled by

$$R = R_R \exp\left[B\left(\frac{1}{T} - \frac{1}{T_R} \right) \right] \qquad (1)$$

where R_R is the resistance at a reference temperature T_R and B is a constant. In order for Equation 1 to work, absolute temperatures (Kelvin units) must be used. The thermistor used in this project, (jameco.com PN 207037) has a resistance R_R of 10,000 ohms at the reference temperature T_R of 298 K (25 °C) and a B value of 4038, with ±10% tolerance on the resistance.

Try plunging your thermistor into ice water, which is approximately 273 K (0 °C) and measure the resistance across the leads using your DMM. (Hint: squeeze the DMM leads hard against the thermistor leads for an accurate resistance reading.) See how close the measured value of R is to the theoretical value from Equation 1. You may see the resistance measurement drifting due to self-heating from the current through the thermistor provided by the DMM.

Rearranging Equation 1 for temperature as a function of resistance yields

$$T = \left[\frac{1}{T_R} + \frac{1}{B} \ln\left(\frac{R}{R_R} \right) \right]^{-1} \qquad (2)$$

Measuring Circuit

The data acquisition pod can only measure voltage, not resistance. You will construct a voltage divider circuit to measure resistance. The schematic for a voltage divider is shown below
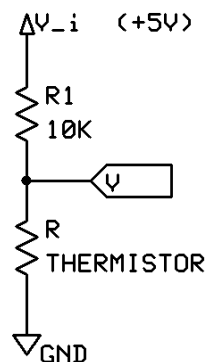
V_i is a fixed supply voltage, which for the experiment will be +5V. R1 is a 10K ohm resistor. R is the thermistor whose resistance varies with temperature. V is the output voltage that will be sampled by the data acquisition unit. The equation for V is

$$V = V_i \frac{R}{R_1 + R} \qquad (3)$$

Rearranging to find R yields

$$R = \frac{R_1 V}{(V_i - V)} \qquad (4)$$

Inserting Equation 4 into Equation 2 gives the equation for temperature as a function of the voltage measured by the data acquisition unit

$$T = \left[\frac{1}{T_R} + \frac{1}{B} \ln\left(\frac{R_1 V}{R_R (V_i - V)} \right) \right]^{-1} \qquad (5)$$

For the setup in your experiment, the values for the constants in this equation are

T_R [deg K]	298
R_R [ohms]	10,000
B [unitless]	4038
R_1 [ohms]	10,000
V_i [volts]	5

Equation 5 gives you the temperature in Kelvin. For convenience, the output should be in Fahrenheit or Celsius. Here are the conversion formulas

$$[\text{deg F}] = [\text{deg K}] * 1.8 - 460$$

$$[\text{deg C}] = [\text{deg K}] - 273$$

Suggestion: Give yourself an engineering challenge and see if you can derive Equations 2-5 on your own. Equation 3 can be derived using Ohm's Law, $V = I*R$.

LabVIEW VI

The first step in the experiment is to program a custom LabVIEW VI to sample the voltage from the voltage divider and convert it into a temperature reading. You will do this in two steps, first creating the conversion section then creating the data acquisition setting. This way you can proceed with a good portion of the experiment setup without needing the data acquisition hardware. It is assumed that you have read the "LabVIEW Quick Start" document that describes the basics of LabVIEW.

Start LabVIEW and create a new blank VI.

On the Front Panel, place two Numeric Indicators and a Gauge.

Change one of the indicators to a Control (R-click > Change to control) and change its label to "Reading [V] (R-click > Properties). The indicator is a placeholder for the voltage reading that later will come from the data acquisition unit. On the other indicator, suppress the label (R-click > Visible Items > Label) and align under the gauge (alignment tool button is at the top of the

window). Using the Properties function, change the gauge label to "Temp [F]" and change its min and max to 20 and 100. Your front panel will look something like this[1]
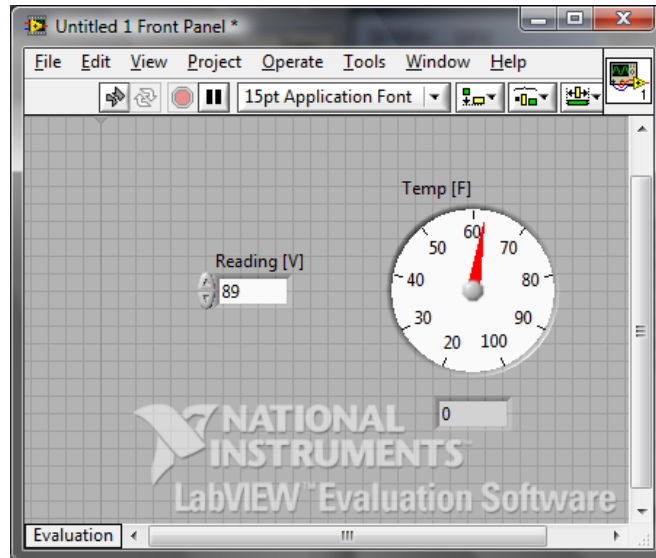
Save your VI.

Activate the Block Diagram (Ctrl-E to switch between Front Panel and Block Diagram) and wire the output of the reading indicator to the inputs of the gauge and the second indicator. Your block diagram will look something like this
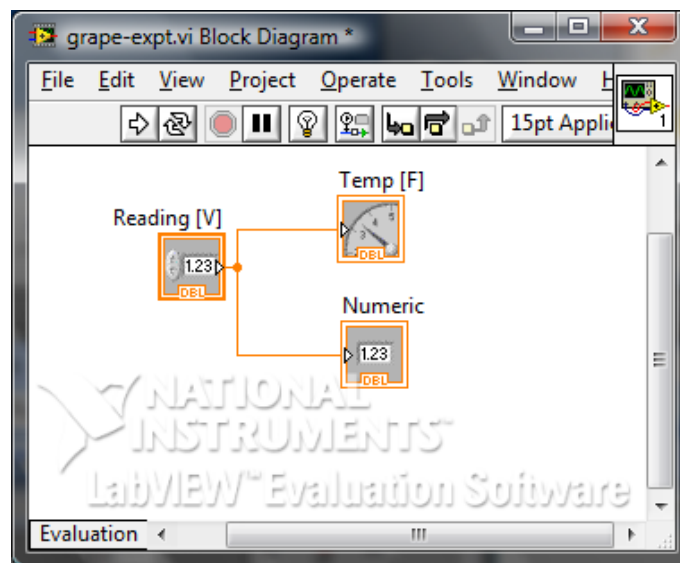
Return to the front panel and run the application with the Run Continuously button. Change the value in the reading indicator and confirm that the gauge and second indicator follow. Stop the application with the Abort button.

---

[1] You might like the Meter or Thermometer numeric indicators more than the Gauge. Feel free to explore and modify the instructions to suit your taste.

The next step is to add the conversion between voltages read by the data acquisition unit and temperature read by the gauge. Return to the block diagram and delete the wire connections. Ctrl-B is useful for clearing dangling wires.

Place a Formula block on the diagram. (Forget how? Consult the "LabVIEW Quick Start" document.) In the Configure Formula dialog box (if it doesn't appear, R-click > Properties), change the label on Input X1 to V.

Starting with Equation 5, substituting the constants from the table, noting that R_1 and R_r cancel in the ln function, adding in the Kelvin to Fahrenheit conversion and expressing in plain text format, the formula for temperature [F] from voltage is

$$((1/298)+(1/4038)*ln(V/(5-V)))**(-1) * 1.8 - 460$$

Type this into the Configure Formula box. If the box to the right of the formula turns green, the formula is syntactically correct. The Configure Formula box will look like this
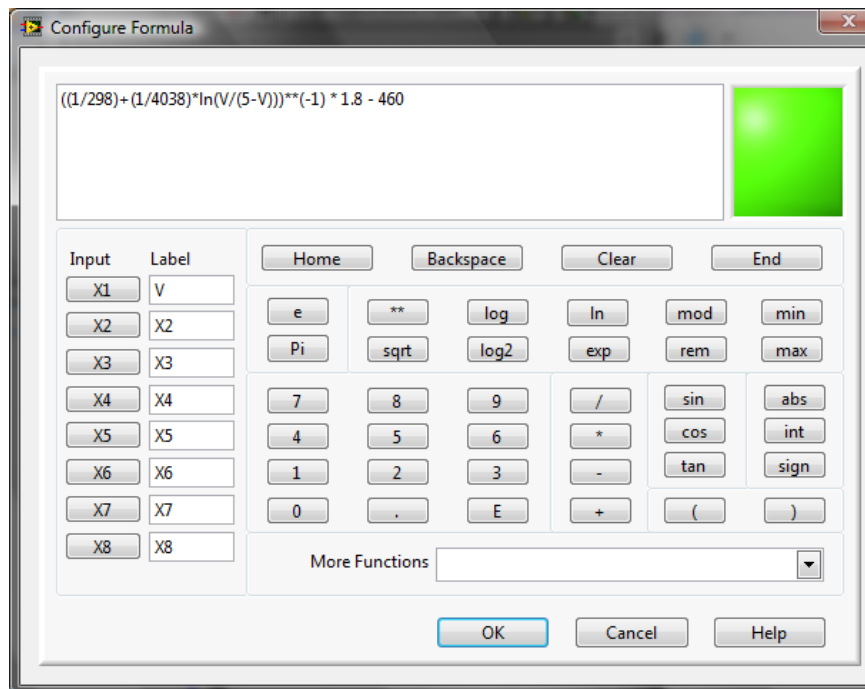
Back on the block diagram, wire the Reading indicator to the input V of the formula block and wire the output Result of the formula block to the gauge and the second indicator blocks. (The Edit > Clean Up Diagram, Ctrl-U, function is useful if the diagram gets messy.) Your Block Diagram will look like this
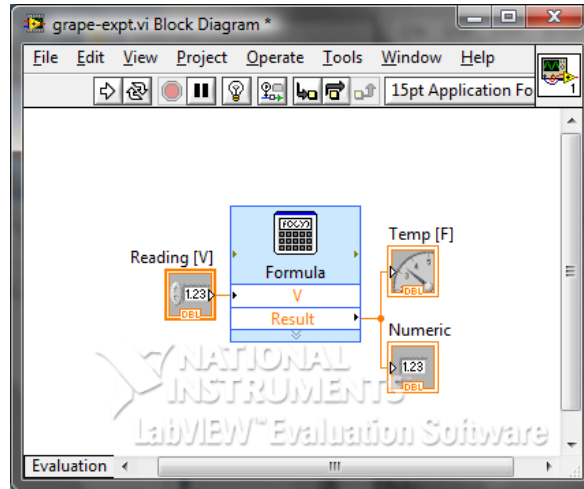
Return to the Front Panel and run the application, entering voltage readings between zero and 5 volts. If everything is correct, you will get these pairs of values from your VI.

| Volts	Temp [deg F]
3.87	31.7
0.31	210.9

For the remainder of this exercise, you will need the USB-6008 unit.

Measuring Voltages

The next step is to use the USB-6008 and modify the VI to collect samples from it.

Connect the USB-6008 to your computer with the USB cable. The green LED on the device will blink.

Go to the Front Panel of your VI and delete the reading indicator. Switch to the Block Diagram. Open the Functions Palette. Navigate to Express > Input. Insert the DAQ Assistant block onto the diagram. If you are prompted about licensing restrictions, click the box that says you understand then click Next. The Create New Express Task box is next. Select Acquire Signals > Analog Input > Voltage
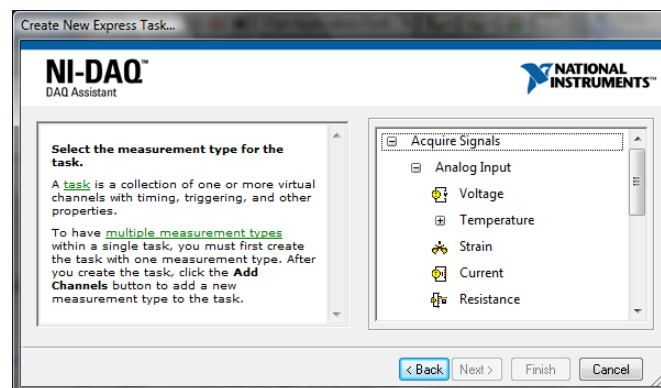
In the next box, select ai0 as the channel to read from. This corresponds to one of the screw terminal connections on the device. Click Finish.
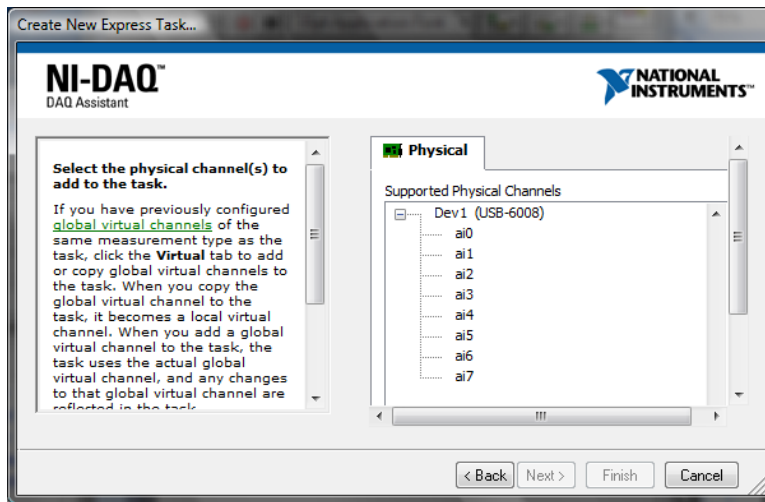
The DAQ Assistant dialog box will open.

Under the Configuration tab, enter these settings
    Terminal Configuration = RSE (to read voltage with respect to GND pin)
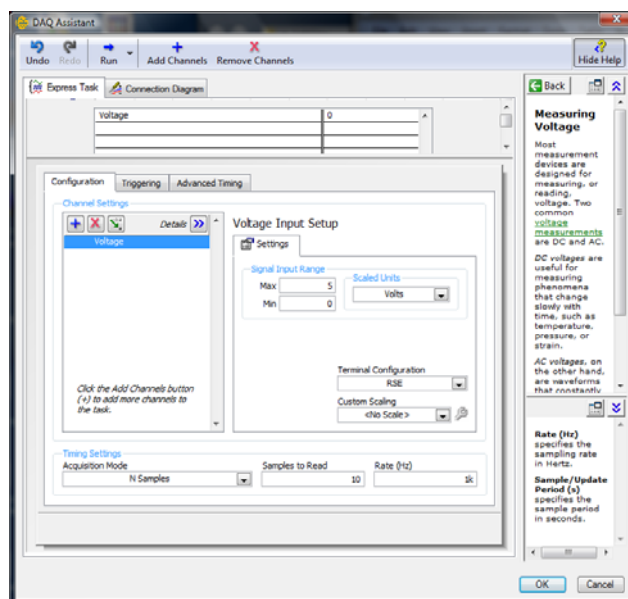   Signal Input Range: Max = 5, Min = 0
    Scaled Units = Volts
    Acquisition Mode = N Samples
   Samples to Read = 10
   Rate (Hz) = 1k

The box will look like this

Click OK to finish the process of inserting the DAQ Assistant into the block diagram.

Add a Numeric Control to the Front Panel, change it to an indicator and change its label to "Volts".

In the block diagram, wire the output of the DAQ Assistant to the volts indicator and to the input of the formula block. It will look like this
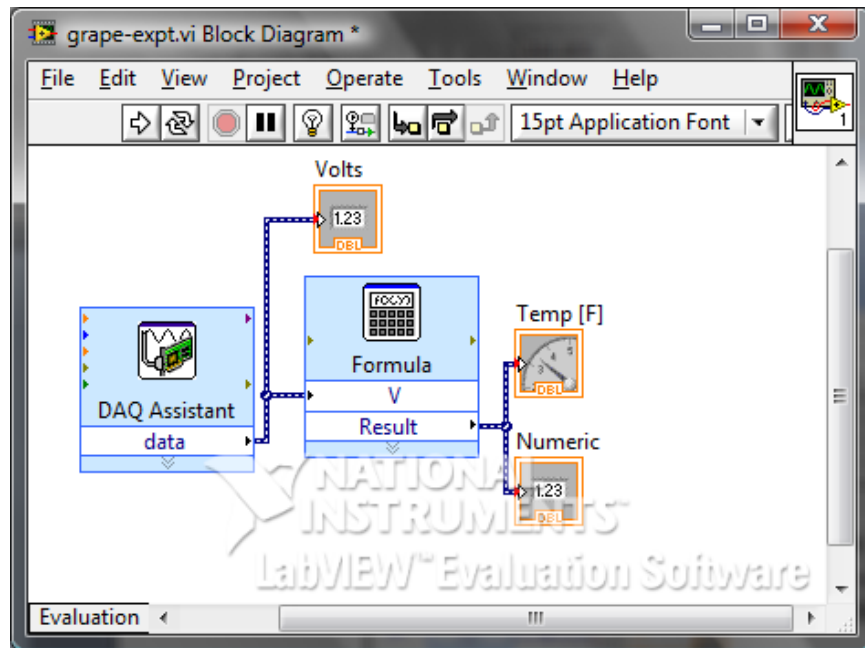
Move to the Front Panel and run the VI, which will look like this
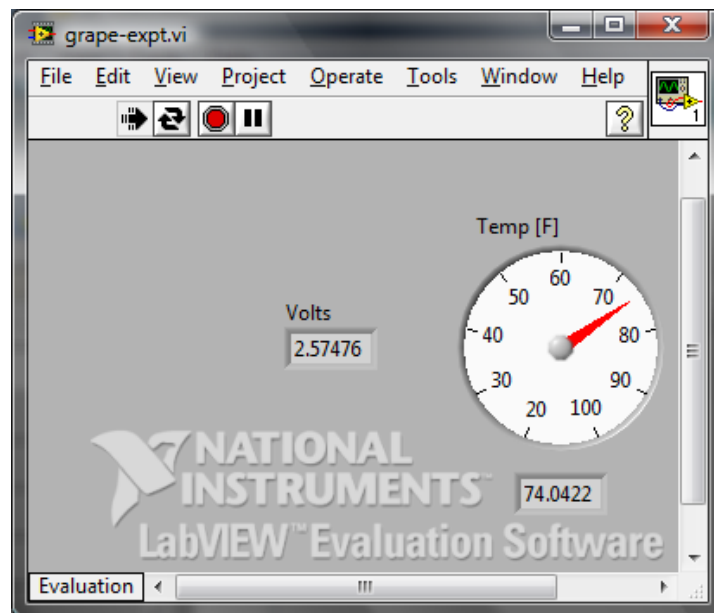
Strip about 3/8" of insulation from both ends of a 22 g solid wire that is about 5" long. Connect one end to terminal AI0 (pin 2) of the USB-6008. To connect a wire, place in the hole and then tighten down the corresponding screw. If you don't have a tiny screwdriver that fits, be inventive.

Touch the other end of the jumper to the GND terminal (pin 1) and to the +5V terminal (pin 31) and watch the Volts indicator on your VI. If you run the screw down all the way on those terminals, you can touch the screw heads with the jumper. You may also want to measure between +5V and GND terminals using your DMM to see if it agrees with the reading on the VI.

Wire and Test the Thermistor Circuit

The next step is to wire the thermistor in the voltage divider circuit described previously. Use 22 g solid wire and either solder wires together, or make the connections on a solderless breadboard. You will need to solder 6" wire leads onto each leg of the thermistor so that it can be plunged into a glass if ice water. The images below show a soldered circuit (left) and a circuit using a solderless breadboard (right).
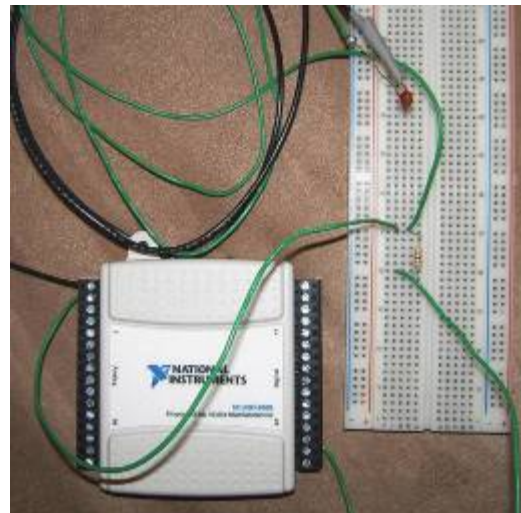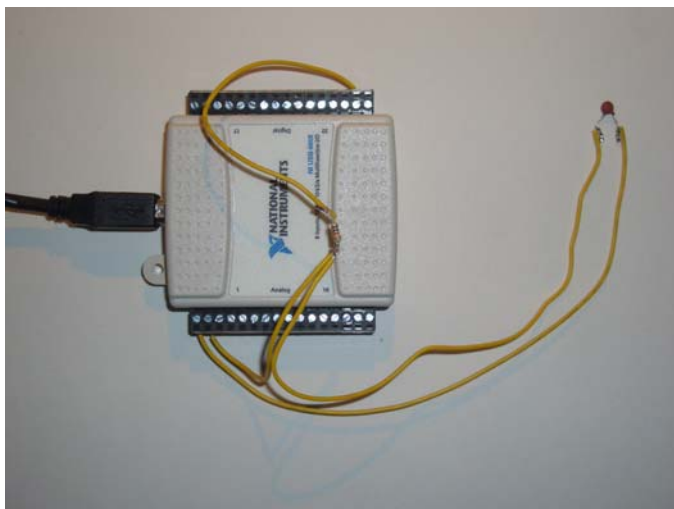

Run the VI. You have a computer-based thermometer. Blow on the thermistor or squeeze between your fingers and watch the gauge go up. Dunk it in ice water and watch it go down.

Automated Data Collection

This next step sets up your VI to collect data at a specified sampling rate and to store the measurements in a data file. Two new objects are needed in the block diagram, a Data Collector and a Write To Measurement File

Find the Data Collector block in the Functions Palette (Express > Sig Manip > Collector) and place on the block diagram. A dialog box will appear. Set the Maximum number of samples to 10,000.

Find the Write To Measurement File block in the Functions Palette (Express > Output > Write To Measurement File) and place on the block diagram. A dialog box will appear. In the dialog box, pick a convenient directory for the output file. Choose File Format = Text (LVM), X Value Columns = One column only, Delimiter = Comma, and If a file already exists = Overwrite file.
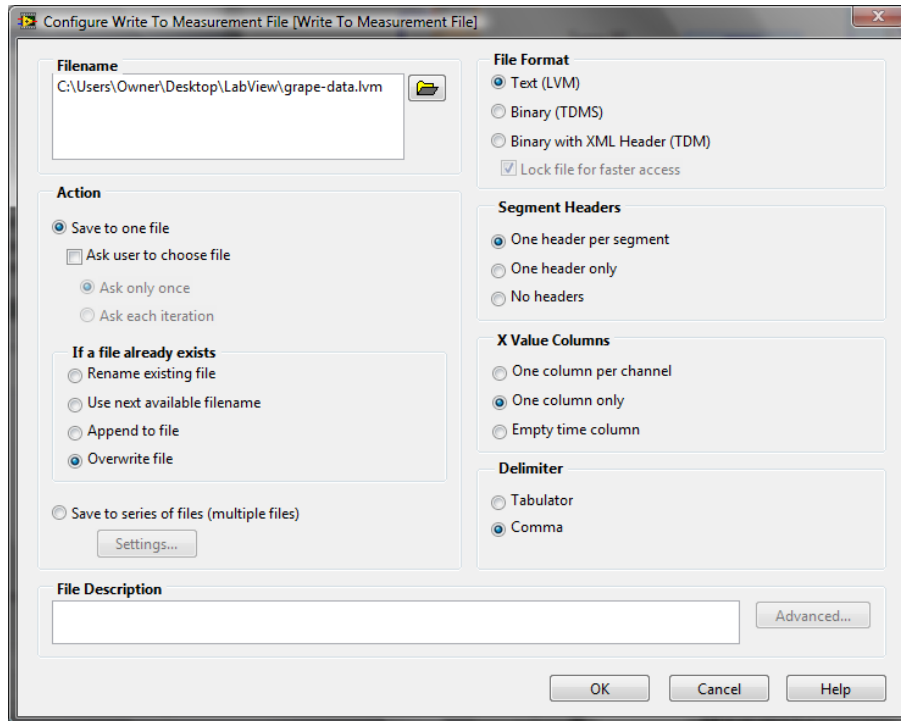
Wire the Result output of the Formula block to the Signals input of the Collector block and wire the Collected Signal output of the Collector block to the Signals input of the Write To Measurement File block.
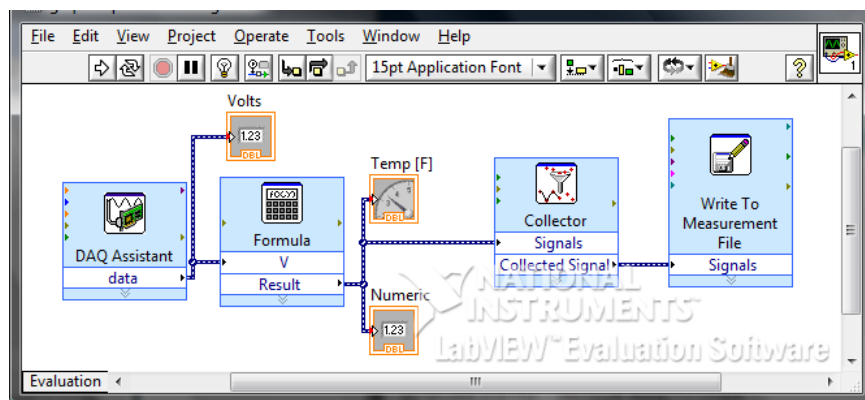
Run the VI for a few seconds then quit. With a plain text editor such as Notepad, open the data file that you specified in the Write To Measurement File dialog. The data file will look something like this

```
grape-data.lvm - Notepad
File  Edit  Format  View  Help
LabVIEW Measurement,
Writer_Version,2
Reader_Version,2
Separator,Comma
Decimal_Separator,.
Multi_Headings,Yes
X_Columns,One
Time_Pref,Absolute
Operator,Owner
Date,2009/10/29
Time,23:03:42.0420000833533440204
***End_of_Header***,
,
Channels,1,
Samples,10,
Date,2009/10/29,
Time,23:03:42.0580000877380371094,
Y_Unit_Label,Volts,
X_Dimension,Time,
X0,0.0000000000000000E+0,
Delta_X,0.001000,
***End_of_Header***,,
X_Value,Voltage (Collected),Comment
0.000000,75.327645
0.001000,75.327645
0.002000,75.327645
0.003000,75.005780
0.004000,75.005780
0.005000,75.005780
0.006000,75.005780
0.007000,75.327645
0.008000,75.327645
0.009000,75.327645
```

The last ten lines are the collected data with the X_Value column being time in seconds and the Voltage column being the temperature data. Because the sample rate in the DAQ Assistant block is set to 1 KHz, the time interval for the data is 0.001 s (1 ms). There are only 10 data points because the Samples to Read setting in the DAQ Assistant is 10.

The sampling rate of the DAQ Assistant block is 1000 Hz, or one sample every millisecond. Because temperature changes slowly and because sampling at 1000 Hz would result in massive and unnecessary amounts of stored data, the sampling rate needs to be lowered to 10 Hz. This will result in a manageable 600 data point set for every minute of data collection.

The current VI runs the DAQ once, saves the data to file, and stops. The VI will be much more useful if it can be run for arbitrary time periods and then save the collected data to the file. The most convenient way to make this happen is to use a While loop.

First, open the properties dialog for the DAQ Assistant on the block diagram. Under the Configuration tab, Timing Settings (you may have to scroll down to reveal), set Acquisition Mode = N Samples, Samples to Read = 10, Rate = 10.
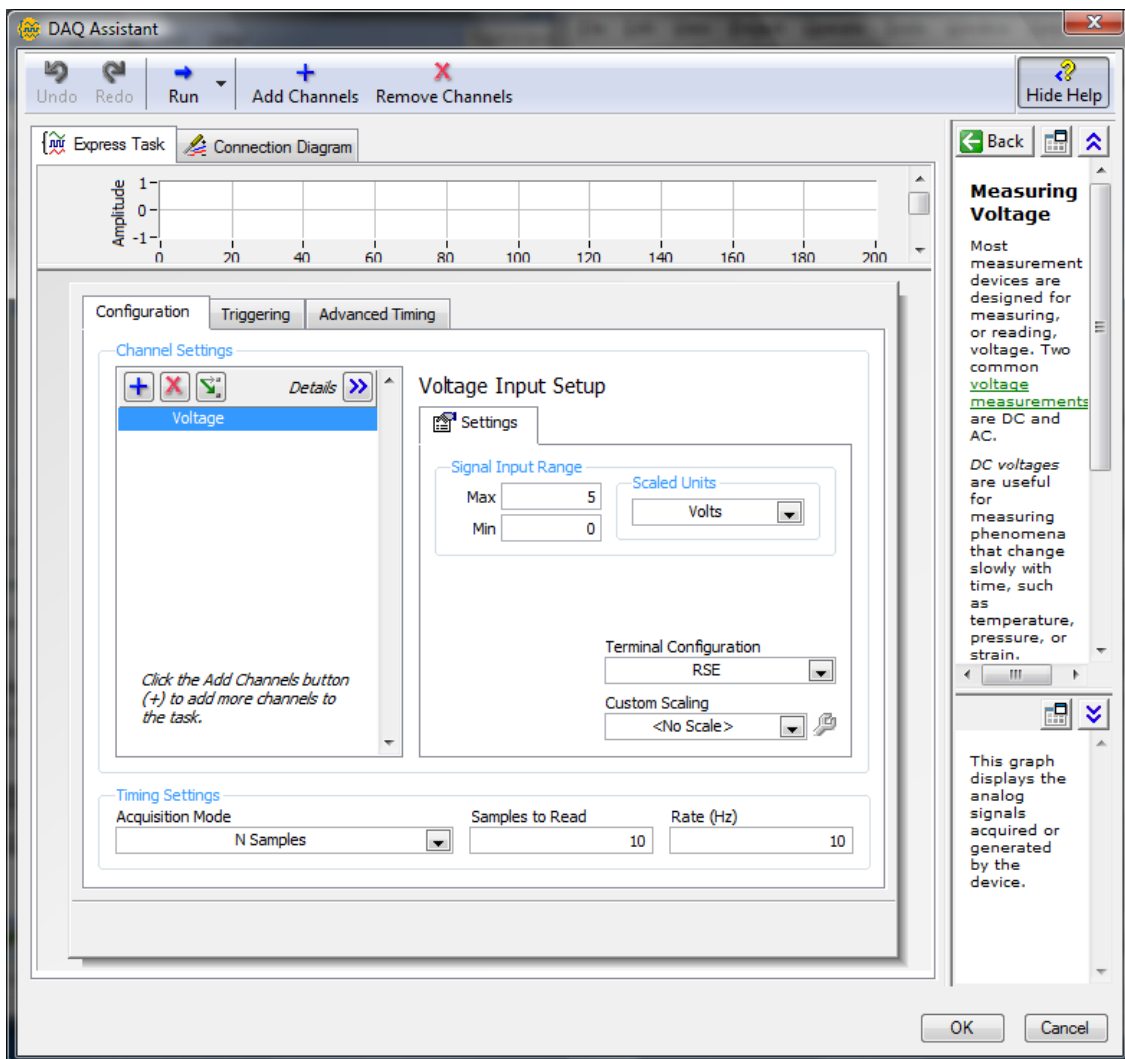
If a dialog box appears asking if you want to create a While Loop, answer no because you will do this manually.

In the Functions Palette, find the While Loop (Express > Exec Control > While Loop). To place on the block diagram, click once in the upper left corner, and then again in the lower right, enclosing all elements on the block diagram except the Write To Measurement File Block, like this
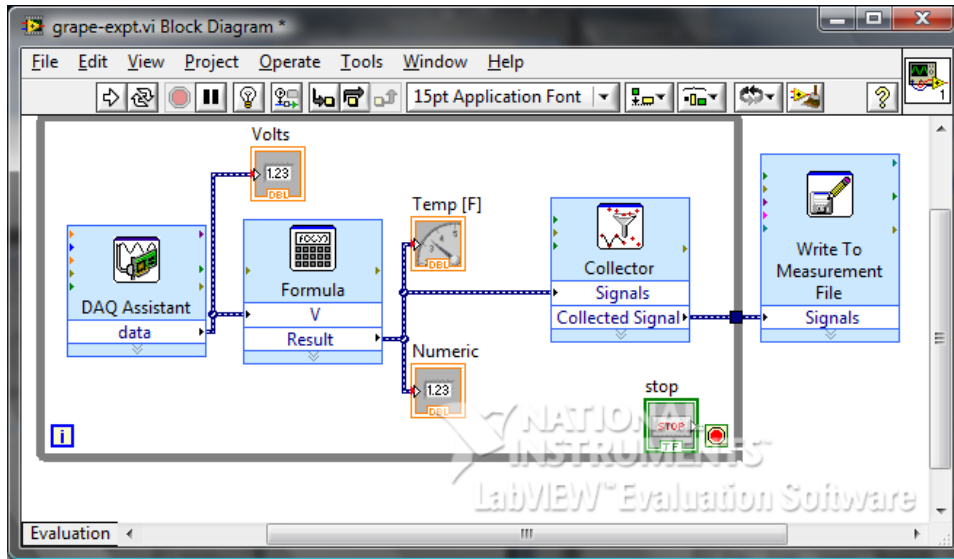
Notice that a STOP button was automatically placed on the front panel. Use it to stop the loop, which will stop data collection and write the data to the file.
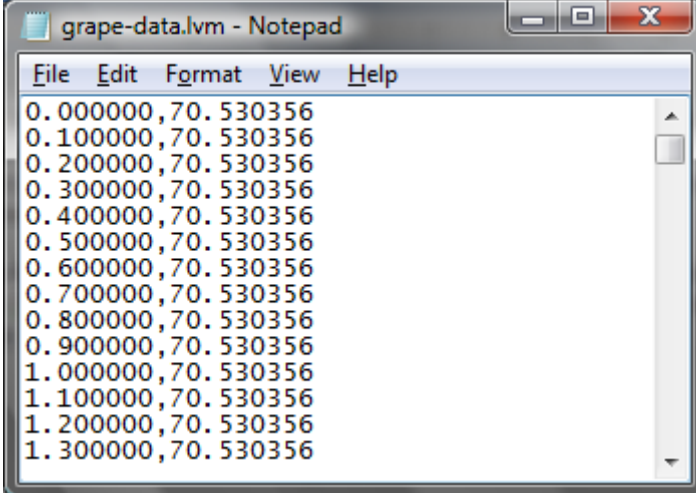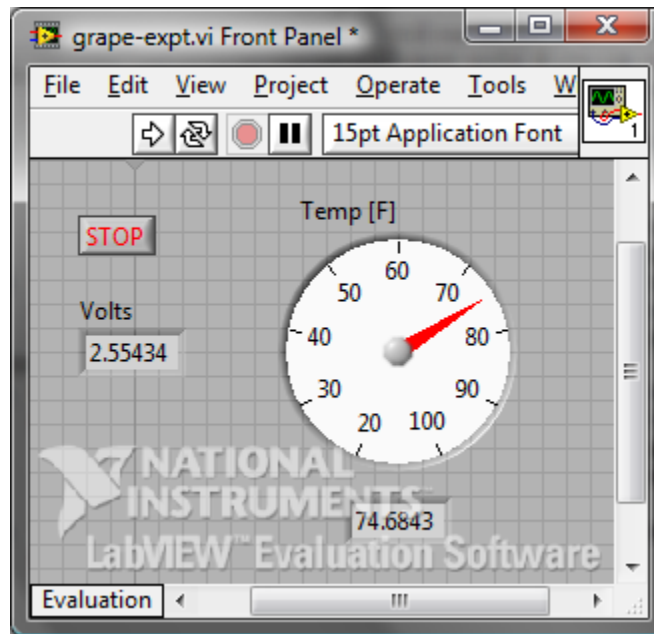

Run the VI. Wait about 10 seconds and then click STOP. Examine the data file. It will have about 100 data points (10 seconds of data collected at 10 Hz). If you click STOP again, new data will be stored, which is the data between successive STOPs. To abort operation, click the Abort button as usual.

Note that while running, the indicators and dial gauge update once per second. This is because of the Samples To Read = 10 and Rate = 10 settings in the DAQ Assistant. The indicators are only updated after 10 samples are read.
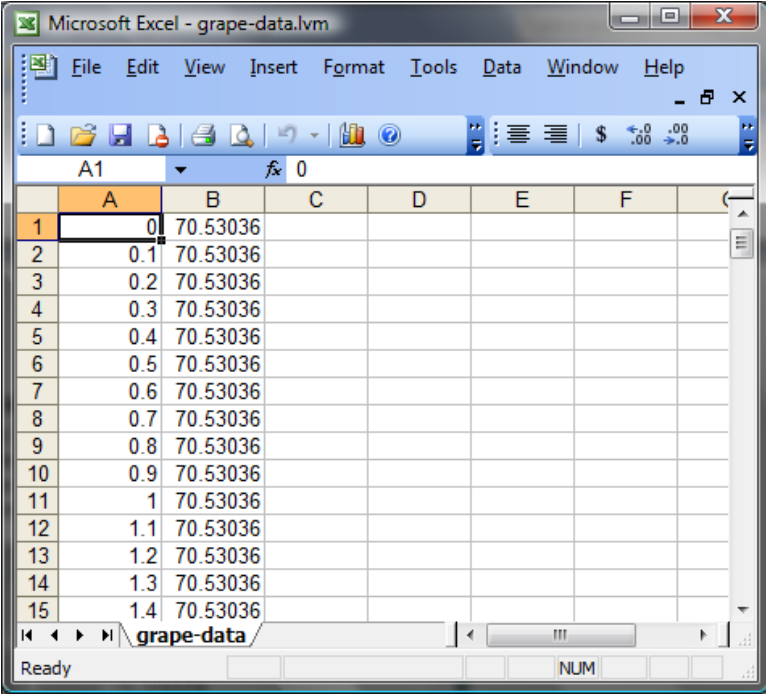
Exporting Data to Excel

Collect data for about 10 s so that there is a data file with about 100 points. During the collection, squeeze the thermistor, hold it next to a lamp, blow on it or do something else to change its temperature. Open the data file and delete all the header lines before the first row of data. Save the file. Here's how the top of the file now looks


Open Excel and then open the file. Depending on your version of Excel, the Text Import Wizard may appear. In successive dialog boxes, select Delimited, to indicate data is in columns and select that the delimiter is a comma. When successful your spreadsheet will have the data in two columns, like this

Hint: On a Windows computer, another way to achieve the same goal is to rename your data file from grape-data.lvm to grape-data.csv. Double-click the file and it will open directly in Excel.

From here, you can create a nice temperature-time plot like this, which shows what happens to temperature when the thermistor is brought up close to the bulb of a halogen desk lamp.
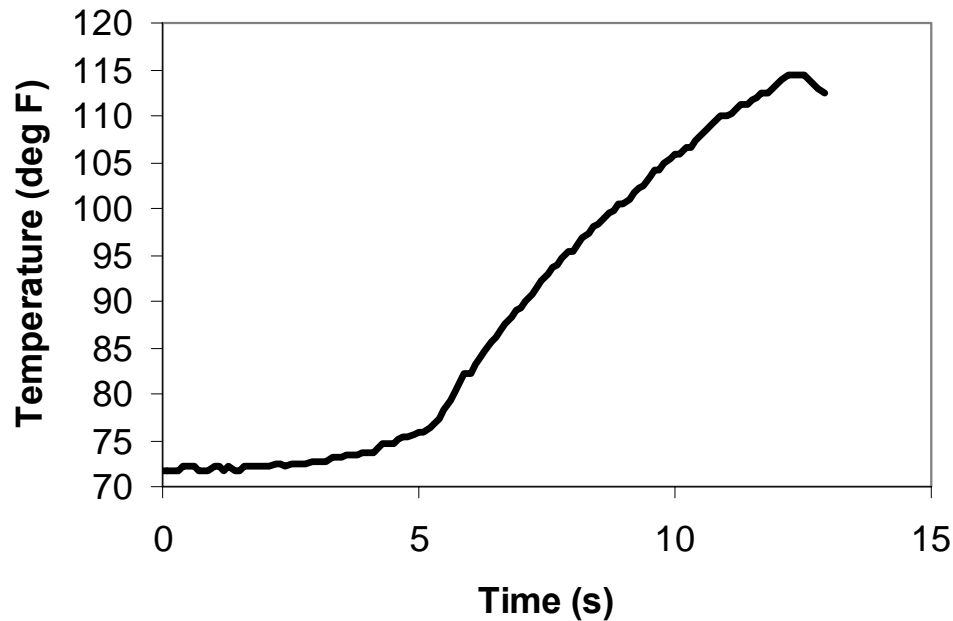
For continuous time plots where there are many data points (100 for every 10 s in this plot), show the line that connects the data points but do not show markers at the points.

Now you are all ready to do the grape experiment, but first a little theory is needed.

The Time Constant of the Temperature Change in a Solid Body

The background physics needed to understand the experiment is lumped transient conduction[2]. Consider a small mass at a uniform initial temperature T_0 placed suddenly in a fluid. Assume the fluid has a constant uniform temperature T_∞. Assume the mass remains at uniform temperature T, i.e. neglect conduction through the mass itself. The temperature at time t is then given by

$$\frac{T - T_\infty}{T_0 - T_\infty} = e^{-(hA/\rho cV)t} \tag{1}$$

where A and V are the surface area and volume of the object, ρ and c are the density and specific heat, and h is the convection coefficient of the fluid. We will not be determining all the material properties in this experiment; therefore, for our purposes, Equation (1) can be re-written as

---

[2] The theory presented in this section is adapted from: Holman, J.P. 2002. *Heat Transfer, Ninth Edition*. New York, McGraw-Hill.

$$T - T_\infty = (T_0 - T_\infty)e^{-t/\tau}$$

<div align="right">(2)</div>

The quantity $\tau = \rho c V/hA$ is known as the **time constant** of the system. The time constant is a measure of how quickly the system responds to changes in temperature. In this case, the difference between T and T_∞ decreases by a factor of e^{-1} (~63%) every time constant. After 4 time constants, the difference is decreased by 98% from its original value. Exponential decay is typically considered complete after 4–5 time constants.

That's all for the theory, now it is on to the experiment.

The Temperature Change in a Grape When Plunged Into Ice Water

You will put your program to the test by collecting and analyzing some temperature data. You will record the temperature at the center of a small, round object as it is subjected to a sudden temperature change (i.e. submerged in ice water). You will export the data to an Excel spreadsheet and perform a curve fit of your data to Equation (2) to determine the time constant τ. You will not need to collect data for the full 4 time constants, but you do need to collect enough data to be able to get an accurate estimate. The more data points, the better. Because temperature changes slowly, You will have to run your test and collect data over a period of 2–5 minutes (or more, depending on what object you test).

Find an object to test. The object should be small, approximately round, and amenable to inserting the thermistor into the middle. Grapes or other small fruit are ideal, but you can choose something else if you have it.
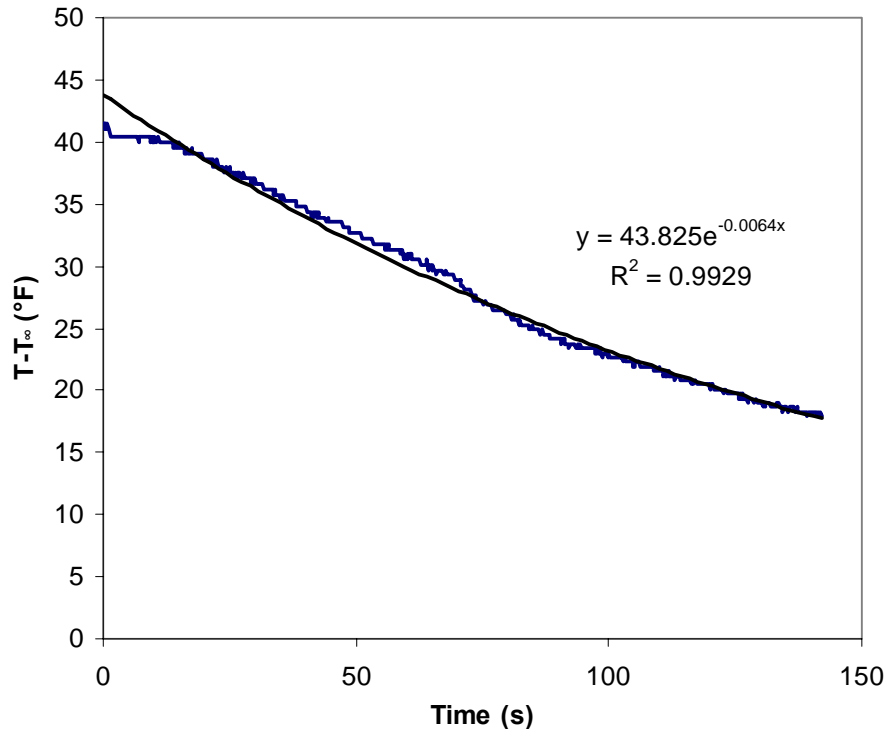
Prepare the apparatus and collect data

1. Prepare an ice bath.
    a. Place a generous amount of ice in a small glass of water.
    b. Give the bath some time to reach equilibrium temperature. You can use your thermistor to monitor the water temperature. Due to variations in the thermistor constants, the temperature reading may not go all the way down to 32°F.
2. Insert the thermistor into the center of the object, being careful not to let the thermistor leads short together.
    a. If you find it difficult to prevent the leads from touching, you can wrap one lead with electrical or office tape to provide insulation.
    b. Verify that the object is approximately at room temperature before starting. The higher the initial temperature, the better your results will be.
3. Plunge the object into the ice bath. Make sure to start recording data as soon as possible (within one second) after doing so.
4. Record data for at least 2 minutes. Do not stop until you notice that the rate of temperature change has become extremely slow.
    a. Have patience. Small grapes have time constants on the order of 2.5 minutes. Larger objects may have even longer time constants. Therefore, don't be surprised if the temperature changes very slowly.
5. Measure the temperature of the ice bath at the end of the experiment, using the thermistor.

6. Import your data file into Excel.

Process the Data

1. Create a new data column, T – Tinf, which is the difference between the recorded temperature T and the temperature of the water bath Tinf. Tinf is "T infinity" and is the temperature that the grape will reach if left in the bath for infinite time.
2. Plot the T – Tinf data versus time using a scatter plot.
3. Right click on the plot line and click **Add Trendline**.
   a. In the **Type** tab choose **Exponential**.
   b. In the **Options** tab, select the options for **Display equation on chart** and **Display R-squared value on chart**.
   c. Three things will appear on your plot: the curve fit to your data (trendline), the equation that defines the curve fit, and the R^2 value for the fit.

The plot shows T-T∞ (°F) on the vertical axis (0 to 50) versus Time (s) on the horizontal axis (0 to 150), with the fitted equation $y = 43.825e^{-0.0064x}$ and $R^2 = 0.9929$.

   d. The R^2 value is a measure of the goodness of fit: a value of 1 represents a perfect fit. Values less than about 0.9 indicate a poor fit. Reasons for low R^2 values include excessive noise and improper modeling, i.e. the system is not accurately described by the exponential model.
4. Increase the number of decimal places on the curve fit equation. Right-click on the equation and select Format Data Labels. Select the Number tab. Increase the number of decimal places to four. Click OK. The equation will now display four decimal places.
5. Use the curve fit equation to determine the time constant of your system. In the plot above, the coefficient of the exponent is given as 0.0064. The time constant is

therefore 1/0.0064 = 156 seconds. Your time constant will be different, perhaps by a lot, because your fruit will be larger or smaller or a different material than the grape used in the example.

If you wish, repeat the process using a different grape, one that is a bit larger or smaller than the first. How does the time constant of the second grape compare to that of the first? Larger? Smaller? The Same? If it is different, what are some possible reasons for the difference?

Lab Report

Prepare a lab report using the standard Introduction, Methods, Results, Discussion format. Two or three pages should be sufficient, if you write and format with care.