

# Thermal Conductivity of Solids

CE 328  
Spring 2006

## **Scenario:**

Your team is part of the Engineering Research Division of CBE Inc. You are being asked to investigate a method for measuring the thermal conductivity of solid materials. You will describe the principles behind the method and compare the thermal conductivities of several materials determined using the technique to literature values.

## **Problem:**

Thermal conductivity is an intensive physical property of a material that relates the heat flow through the material per unit area to temperature gradient across the material. The thermal conductivity of a material is basically a measure of its ability to conduct heat. In a wide variety of applications ranging from building insulation to electronics, it is important to determine a material's thermal conductivity. Typical methods of thermal conductivity measurement can be categorized as either steady-state or non-steady-state. In steady-state techniques, equilibrium heat flux and temperature gradient are measured. In non-steady-state techniques, a variable heat flux is produced and the time-variant temperature gradient is measured. The method you are asked to investigate involves the transient heating of spherical shaped samples.

## **System:**

Solid spheres of known and unknown thermal conductivity are available. The spheres are first cooled to ice temperature and then immersed in a circulating water bath whose temperature is maintained at a uniform (but higher) temperature.

## **Measurements:**

A thermocouple located at the center of each spherical body measures the temperature at that point. This temperature is recorded as a function of time using a data acquisition system controlled by LabView software. *Bring a floppy disk for this lab.*

## **Theory:**

The thermal conductivity of a material can be determined by comparing the measured temperature to that predicted from a mathematical analysis of heat conduction in the solid body. The thermal conductivity is a parameter in the theory that can be varied to match the experimental temperature profile.

Fourier's Law of Heat Conduction states that the heat flux by conduction is proportional to the temperature gradient. (Note that  $q$ , heat flux, has units of heat flow per unit area)

$$q = -k\nabla T \quad (1)$$

where the proportionality constant,  $k$ , is the thermal conductivity .

For the problem of heat conduction in a sphere of diameter  $R$ , application of Fourier's Law of Heat Conduction to the unsteady flow of heat from the object to a surrounding fluid produces the energy balance

$$\frac{\partial T}{\partial t} = \mathbf{a} \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial T}{\partial r} \right) \text{ for } t \geq 0 \text{ and } 0 = r = R \quad (2)$$

where

$T$  = temperature of the solid at radius  $r$  and time  $t$

$t$  = time

$r$  = radial distance from the center

$\mathbf{a} = k/(\mathbf{r}C_p)$  = thermal diffusivity of the solid

$k$  = thermal conductivity of the solid

$\mathbf{r}$  = density of the solid

$C_p$  = heat capacity of the solid

$R$  = radius of the sphere.

For a sphere initially at a uniform temperature,  $T_o$ , suddenly immersed into a bath of constant, uniform temperature,  $T_\infty$ , the initial and boundary conditions for this partial differential equation are:

$$T = T_o \text{ for } t = 0 \text{ and } 0 = r = R \quad (3)$$

$$\frac{\partial T}{\partial r} = 0 \text{ at } r = 0 \text{ for all } t > 0 \quad (4)$$

$$-k \frac{\partial T}{\partial r} = h(T - T_\infty) \text{ at } r = R \text{ for all } t > 0 \quad (5)$$

where  $h$  is the heat transfer coefficient for the transfer of heat from the solid to the fluid. This coefficient must be determined experimentally. (What parameters of the system would affect  $h$ ?)

The solution to the above partial differential equation (eqn. (2)) for the temperature at the center of the sphere with the initial and boundary conditions given in equations (3),(4), and (5) can be written as

$$\frac{T - T_\infty}{T_o - T_\infty} = 2 \sum_{n=1}^{\infty} \frac{(\sin(\mathbf{b}_n) - \mathbf{b}_n \cos(\mathbf{b}_n)) \exp\left(-\mathbf{b}_n^2 \frac{\mathbf{a}t}{R^2}\right)}{(\mathbf{b}_n - \cos(\mathbf{b}_n) \sin(\mathbf{b}_n))} \quad (6)$$

where

$\mathbf{b}_n = n^{th}$  root of the equation

$$Bi - 1 = \frac{-b_n}{\tan(b_n)} \quad (7)$$

$$Bi = \frac{hR}{k} = \text{Biot number (dimensionless)}$$

The solution expressed in Eqn.(6) can be written in terms of dimensionless variables as:

$$q = f(t, h = 0) = 2 \sum_{n=1}^{\infty} \frac{(\sin(b_n) - b_n \cos(b_n)) \exp(-b_n^2 t)}{b_n - \cos(b_n) \sin(b_n)} \quad (8)$$

where

$$q = \frac{T - T_{\infty}}{T_o - T_{\infty}} = \text{unaccomplished temperature change (dimensionless)}$$

$$t = \frac{at}{R^2} = \text{dimensionless time}$$

$$h = r/R = \text{dimensionless radial position}$$

and again

$$b_n = n^{\text{th}} \text{ root of the equation}$$

$$Bi - 1 = \frac{-b_n}{\tan(b_n)}$$

More details about the solution of the PDE are given in Appendix 1.

Expression (8) has been evaluated and plotted by several authors. Similar solutions for other shapes, e.g., cylinders and flat plates, as well as the solution for a cylinder of finite length can also be found in the literature. Graphical forms of the solutions to transient heat conduction problems are called Heisler charts. In this experiment, you can evaluate Eqn.(8) using a given Excel spreadsheet or you may choose to evaluate the expression using Maple instead.

### **Experimental work:**

Your team will record the thermal history of several samples (initially in thermal equilibrium with ice), immersed in a circulatory water bath that is held at constant temperature. You will monitor the temperature using a thermocouple embedded in the center of the sample. This thermocouple, as well as thermocouples located in the ice bath, and in the circulatory water bath are connected to a data acquisition board controlled using LabView software. A basic LabView program for recording the temperature measurements from these thermocouples is provided. It is described in more detail in Appendix 3. **Bring a floppy disk for this lab.** Record in your lab notebook the dimensions of the spheres and the names of the files containing the recorded data.

The unknown heat transfer coefficient,  $h$ , must first be determined by recording the thermal history of a sample of known shape, thermal conductivity, density and heat capacity. We postulate that this coefficient is independent of the material; it depends only on the shape of the sample and the conditions prevailing in the water bath. Thus, once  $h$  is known for the system, the thermal conductivity of any solid can be determined.

### Calculations

1. The aluminum alloy (Type 6262-T9) sphere is to be used to determine the heat transfer coefficient,  $h$ . This can be done by evaluating the solution (Equation 8) numerically using the known conductivity. The heat transfer coefficient,  $h$ , can then be varied to fit the experimental data for the aluminum sphere. As a starting point in your iteration, use an estimate of  $h$  determined from a “lumped parameter” solution (as discussed in CE308). Use the full temperature vs. time curve to fit the best possible value of the Biot modulus to match all of the data. The heat transfer coefficient can then be calculated from the Biot modulus. The error in  $\theta$  should be calculated by propagation of errors from your experimental data, and compared (after making the appropriate corrections) to the sum of squares error resulting from your curve fit.
2. The remaining unknown materials are believed to be stainless steel (Type 304), "free cutting" brass, and poly-methyl-methacrylate (Lucite® or Plexiglas®). Once the heat transfer coefficient is determined for the system, the unknown conductivities can be determined by fitting the numerical solution to the experimental data for the other spheres by varying the conductivity,  $k$ , assuming that  $h$  remains the same.
3. An Excel file will be given to you that you can use to solve for the heat transfer coefficient of Al, and for the thermal conductivity of the other materials. For a description of the calculation method, see Appendix 4.
4. Plot your experimental data in terms of unaccomplished temperature change versus dimensionless time along with the values calculated from the mathematical model.

### Pre-lab Assignment:

Write a detailed operating procedure. **Be prepared to discuss the calculations/programming needed (Understand what the Excel spreadsheet does!).** Determine what the “sum of square errors” represents. Also, for error determination in the experimental values for  $\theta$ , be prepared to discuss how you will estimate the error in  $T_{\text{center}}$ .

### Some Ideas to Consider in the Report

1. How do the measured values for  $k$  compare with those given in the literature? Remember to cite sources for literature values.

2. Do  $h$  and  $k$  appear to change with unaccomplished temperature? In other words, does the fit of the mathematical model vary over the course of the experiment?
3. How did you determine the "best" value for  $h$  and  $k$  from your work?
4. What are some of the sources of error (both systematic and random) in your work? How can you quantify these sources, and what is their effect on your final results?
5. What is the error that you determined for  $\theta$  using propagation of errors? Did it change for the different materials? How does it compare to the sum of square errors, (SSQ) that you determined for your best fits? (You need to make some calculations to the SSQ errors or to theta to make a comparison.)

### **References**

1. Bird, R.B., Stewart, W.E., and Lightfoot, E.N., Transport Phenomena, New York: John Wiley and Sons, Inc., 1966.
2. Perry's Chemical Engineers' Handbook, John H. Perry, Editor, NY.
3. Foust, A.S., Clump, C.W., Maus, L., and Wenzel, L.A., Principles of Unit Operations, New York: John Wiley and Sons, Inc., 1967.
4. Hull, C. and West, C.J., International Critical Tables of Numerical Data, Physics, Chemistry and Technology. New York: McGraw-Hill Book Co., Inc., 1933.
5. McAdams, W.H., Heat Transmission. New York: McGraw-Hill Book Co., Inc., 1948.
6. Ingersol, et al., Heat Conduction. New York: McGraw-Hill Book Co., Inc., 1948.
7. Gurney and Lurie, *Ind. Eng. Chem.* **15**, 1170 (1923).
8. Parker, J.D., Boggs, J.H. and Blick, E.F., Introduction to Fluid Mechanics and Heat Transfer, Addison-Wesley Publishing Co., 1969.
9. Carslaw, H.S. and Jaeger, J.C., Conduction of Heat in Solids, 2<sup>nd</sup> edition, Oxford Science Publications, 1959.

## Appendix 1: Analysis of transient conduction in a sphere:

$$\frac{\partial T}{\partial t} = a \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial T}{\partial r} \right) \text{ for } t \geq 0, \text{ and } 0 \leq r \leq R$$

with initial and boundary conditions

$$T = T_o \text{ for } t = 0 \text{ and } 0 = r = R$$

$$\frac{\partial T}{\partial r} = 0 \text{ at } r = 0 \text{ for all } t > 0$$

$$-k \frac{\partial T}{\partial r} = h(T - T_\infty) \text{ at } r = R \text{ for all } t > 0$$

This can be made dimensionless using

$$\mathbf{q} = \frac{T - T_\infty}{T_o - T_\infty}, \quad \mathbf{t} = \frac{at}{R^2}, \quad \mathbf{h} = \frac{r}{R}, \quad \text{and } \text{Bi} = \frac{hR}{k}$$

Substituting these into the equations and boundary conditions gives

$$\frac{\partial \mathbf{q}}{\partial \mathbf{t}} = \frac{1}{\mathbf{h}^2} \frac{\partial}{\partial \mathbf{h}} \left( \mathbf{h}^2 \frac{\partial \mathbf{q}}{\partial \mathbf{h}} \right)$$

with initial and boundary conditions

$$\mathbf{q} = 1 \text{ at } \mathbf{t} = 0 \text{ for all } \mathbf{h}$$

$$\frac{\partial \mathbf{q}}{\partial \mathbf{h}} = 0 \text{ at } \mathbf{h} = 0 \text{ for all } \mathbf{t}$$

$$-\frac{\partial \mathbf{q}}{\partial \mathbf{h}} = \text{Bi} \mathbf{q} \text{ at } \mathbf{h} = 1 \text{ for all } \mathbf{t}$$

Making the substitution  $v = \mathbf{h}\mathbf{q}$ , so that

$$\mathbf{q} = \frac{v}{\mathbf{h}}, \quad \frac{\partial \mathbf{q}}{\partial \mathbf{t}} = \frac{1}{\mathbf{h}} \frac{\partial v}{\partial \mathbf{t}}, \quad \frac{\partial \mathbf{q}}{\partial \mathbf{h}} = \frac{1}{\mathbf{h}} \frac{\partial v}{\partial \mathbf{h}} - \frac{v}{\mathbf{h}^2},$$

$$\mathbf{h}^2 \frac{\partial \mathbf{q}}{\partial \mathbf{h}} = \mathbf{h} \frac{\partial v}{\partial \mathbf{h}} - v, \quad \frac{\partial}{\partial \mathbf{h}} \left( \mathbf{h}^2 \frac{\partial \mathbf{q}}{\partial \mathbf{h}} \right) = \mathbf{h} \frac{\partial^2 v}{\partial \mathbf{h}^2}, \text{ and } \frac{1}{\mathbf{h}^2} \frac{\partial}{\partial \mathbf{h}} \left( \mathbf{h}^2 \frac{\partial \mathbf{q}}{\partial \mathbf{h}} \right) = \frac{1}{\mathbf{h}} \frac{\partial^2 v}{\partial \mathbf{h}^2}$$

transforms the equation and initial and boundary conditions to:

$$\frac{\partial v}{\partial \mathbf{t}} = \frac{\partial^2 v}{\partial \mathbf{h}^2}$$

$$v = \mathbf{h} \text{ at } \mathbf{t} \text{ for all } \mathbf{h}$$

$$\lim_{\mathbf{h} \rightarrow 0} \mathbf{h} \left( \frac{1}{\mathbf{h}} \frac{\partial v}{\partial \mathbf{h}} - \frac{v}{\mathbf{h}^2} \right) = 0 \text{ for all } \mathbf{t}$$

$$-\frac{\partial v}{\partial \mathbf{h}} = (\text{Bi} - 1) \text{ at } \mathbf{h} = 1 \text{ for all } \mathbf{t}.$$

A general solution that includes this case is given on pages 126-127 of Carslaw and Jaeger. The solution given there is

$$v = \sum_{n=1}^{\infty} Z_n(\beta_n) \exp(-\beta_n^2 t) \int_0^1 \beta_n Z_n(\beta_n r) dr$$

where

$$Z_n(\beta_n) = \sqrt{\frac{2(\beta_n^2 + (Bi - 1)^2)}{(\beta_n^2 + (Bi - 1)^2) + (Bi - 1)}} \sin(\beta_n)$$

and the  $\beta_n$  are the roots of

$$Bi - 1 = -\beta_n (\cot(\beta_n)) \quad (A)$$

using this substitution and the trig identity for  $\csc^2(\beta_n)$  this can be rearranged to give

$$Z_n(\beta_n) = \sqrt{\frac{2\beta_n}{\beta_n + \cos(\beta_n) \sin(\beta_n)}} \sin(\beta_n)$$

Plugging this into the expression for  $v$  gives

$$v = \sum_{n=1}^{\infty} \frac{2\beta_n \sin(\beta_n)}{\beta_n + \cos(\beta_n) \sin(\beta_n)} \exp(-\beta_n^2 t) \int_0^1 \sin(\beta_n r) dr$$

Evaluating the integral gives

$$v = 2 \sum_{n=1}^{\infty} \frac{(\sin(\beta_n) - \beta_n \cos(\beta_n)) \sin(\beta_n)}{\beta_n^2 - \beta_n \cos(\beta_n) \sin(\beta_n)} \exp(-\beta_n^2 t)$$

The unaccomplished temperature change is then  $q = \frac{v}{h}$ .

We are interested in the temperature at the center of the sphere,  $h = 0$ . So, we take the appropriate limit of the above expression as  $h$  goes to zero to get

$$q(r=0, t) = 2 \sum_{n=1}^{\infty} \frac{(\sin(\beta_n) - \beta_n \cos(\beta_n)) \exp(-\beta_n^2 t)}{\beta_n - \cos(\beta_n) \sin(\beta_n)} \quad (B)$$

## Appendix 2: Evaluation of Infinite Series Solution in Excel®

An Excel Program will be available on the website to evaluate the infinite series solution. Use the correct worksheet for the geometry of the object you are solving for (**Use only the worksheet for spheres**).

Put in your time and temperature data in the first two columns. The worksheet can be expanded to fit additional data points. Tau and theta will automatically be calculated.

k (W/(m K))	rho	Cp	R	h	To	Tinf										
172	2700	905	0.0254	1050	2.069	36.4										
Bi							Roots	0.548752	3.871937	7.037648	10.1887	13.33532	16.48004	19.62376		
0.15505814							"Zero"	0.001364	6.62E-06	-5.5E-06	3.61E-06	-1.3E-05	1.38E-05	-1.5E-05		
Time(s)	T(C)	tau	theta(exp)	theta(calc)	error	error^2	term 1	term 2	term 3	term 4	term 5	term 6	term 7			
0.55	2.243	0.060008	0.995211	0.991509	0.003701	1.37E-05	0.505669	-0.01044	0.000534	-1.2E-05	9.26E-09	-2.4E-10	2.06E-13			
2.2	2.546	0.240033	0.986671	0.956667	0.030003	0.000918	0.476906	-0.0007	7.18E-06	-9E-14	1.16E-21	-1.4E-31	1.81E-43			
5	5.016	0.54553	0.918885	0.873762	0.045123	0.002036	0.436888	-7.2E-06	1.92E-14	-1.5E-27	2.95E-45	-1.3E-67	1.3E-94			
7.86	7.813	0.857573	0.841898	0.795411	0.046487	0.002161	0.397706	-6.7E-08	3.73E-21	-1.3E-41	2.34E-69	-2E-104	8.4E-147			
10.61	10.234	1.157615	0.775261	0.726696	0.048566	0.002359	0.363348	-7.4E-10	1.31E-27	-3.9E-55	1.68E-92	-8E-140	5.6E-197			
12.86	12.11	1.403109	0.723624	0.674914	0.048711	0.002379	0.337457	-1.9E-11	6.88E-33	-3.3E-66	1.7E-111	-9E-169	4.9E-238			
15.11	14.161	1.648692	0.667171	0.626821	0.04035	0.001638	0.313411	-4.7E-13	3.61E-38	-2.8E-77	1.9E-130	-1E-197	4.3E-279			
17.36	15.947	1.89408	0.618012	0.582156	0.035857	0.001286	0.291078	-1.2E-14	1.89E-43	-2.4E-88	2.1E-149	-1E-226	0			
19.61	17.646	2.139559	0.571248	0.540679	0.030575	0.000935	0.270336	-3E-16	9.91E-49	-2E-99	2.3E-169	-1E-255	0			
21.26	18.728	2.319694	0.541466	0.512143	0.029323	0.00086	0.256071	-2E-17	1.33E-52	-2E-107	2.8E-162	-7E-277	0			
23.51	20.352	2.585082	0.496786	0.475849	0.021117	0.000446	0.237824	-5.1E-19	6.97E-58	-1E-118	3.1E-201	-8E-308	0			
30.76	24.63	3.356101	0.379015	0.374934	0.004162	1.75E-05	0.187417	-3.6E-24	6.74E-75	-3E-154	2.6E-262	0	0			
35.27	26.761	3.848169	0.32036	0.323211	-0.00285	8.13E-06	0.161806	-2.3E-27	1.78E-85	-2E-176	2.6E-300	0	0			
39.72	28.742	4.333691	0.265834	0.279248	-0.01341	0.00018	0.139624	-1.6E-30	6.32E-96	-2E-198	0	0	0			
44.77	30.709	4.884676	0.211692	0.236556	-0.02466	0.000518	0.118278	-4E-34	8.9E-108	-4E-223	0	0	0			
50.37	32.295	5.49967	0.168038	0.196802	-0.02876	0.000827	0.086401	-4.2E-38	6.4E-121	-1E-250	0	0	0			
63.22	34.822	6.897832	0.098493	0.129026	-0.03054	0.000933	0.064513	-3.2E-47	4.5E-151	0	0	0	0			
69.92	35.729	7.628692	0.073518	0.103532	-0.03001	0.000901	0.051766	-5.5E-52	6.4E-167	0	0	0	0			
76.68	36.379	8.386249	0.055627	0.082912	-0.02728	0.000744	0.041456	-8.7E-57	1.1E-182	0	0	0	0			
85.63	37.034	9.342748	0.037999	0.06179	-0.02419	0.000585	0.030895	-3.8E-63	1.1E-203	0	0	0	0			
94.59	37.783	10.32034	0.018933	0.046033	-0.02905	0.000844	0.023016	-1.6E-69	1.1E-224	0	0	0	0			
						sum sq en	0.020673									

For the AI sphere:

1. Put in k for AI.
2. Guess a value for h.
3. Run the macro to obtain the roots of Eqn (7) ( (Ctrl+Shift+r to run macro)
4. Check the solution to make sure "zero" is near zero, and the roots fall between 0 and pi, pi and 2pi, 2pi and 3 pi, etc. **Make sure the first root is not close to zero. If it is enter a number close to one and run the macro again.**
5. Look at the sum of squares error. (bottom of error^2 column)
6. Repeat steps 1-3 until you have minimized sum of squares error.
7. Now, use this h for other cylinders to determine k.

For other spheres:

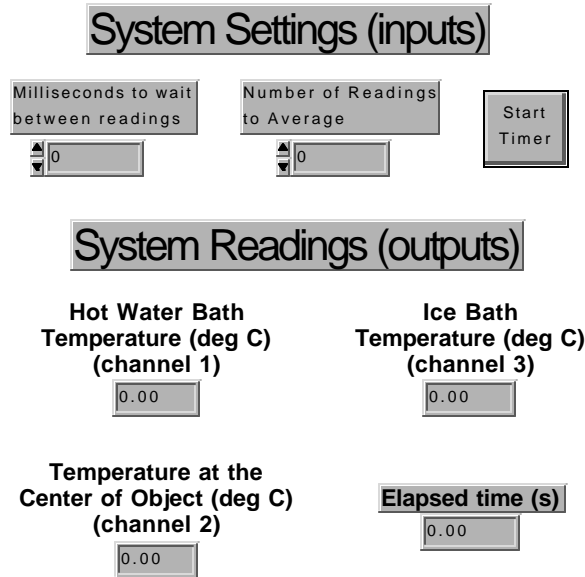
1. Put in h you determined for AI.
2. Guess a value for k.
3. Run the macro to obtain the roots of Eqn (7) (Ctrl+Shift+r)



4. Check the solution to make sure “zero” is near zero, and the roots fall between 0 and  $\pi$ ,  $\pi$  and  $2\pi$ ,  $2\pi$  and  $3\pi$ , etc. **Make sure the first root is not close to zero. If it is enter a number close to one and run the macro again.**
5. Look at the sum of squares error. (bottom of error<sup>2</sup> column)
6. Repeat steps 1-3 until you have minimized sum of squares error.
7. Now, you have  $k$  for your “unknown” sphere.

## Appendix 5: LabViewä program for recording temperatures

The front panel of the LabView virtual instrument (data acquisition program) for this experiment is shown below:

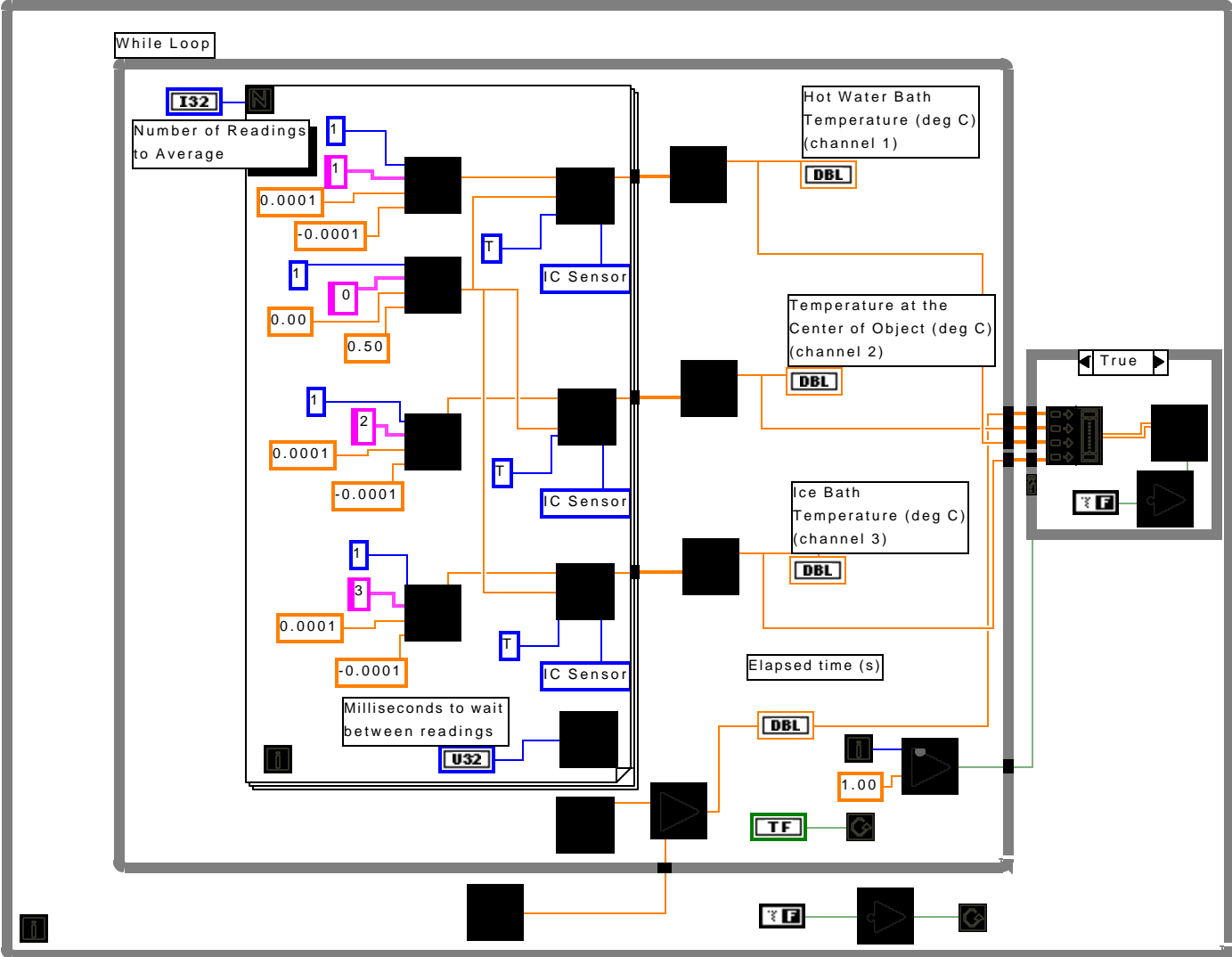


The controls listed under system settings are items that you enter by typing numbers in the boxes or by clicking on the 'start timer' button. You enter a time to wait between temperature readings (the minimum is around 5 milliseconds) and a number of temperature readings to average before displaying the result. The time between data points that are displayed is the product of the waiting time and the number of points. For example, if you wait 10 milliseconds between readings and average 100 readings, you will record one data point about every 1000 milliseconds, or 1 data point per second. The four indicators under system readings show the temperatures in the water bath, at the center of the object that is connected to the data acquisition box, and in the ice bath. The time elapsed since the 'start timer' button was pushed is also displayed. These will be displayed continuously while the program is running, but will not be saved for output to a file until you press the 'start timer' button. After you click that button, it will change to a 'stop timer' button, and each time the readings are displayed, they are also added to a table of results that is recorded in a text file when you push the 'stop timer' button. So, to record the temperatures as a function of time, you provide values for the milliseconds to wait between readings and for the number of readings to average. Then, as you put the object into the water bath, you click the 'start timer' button. Data will then be collected at the specified rate. When the temperature at the center of the object reaches the hot water bath temperature, then you click the 'stop timer' button. You will then be asked for a name for the data file to which the results will be recorded. The data will be saved, and you can then open that file in Excel or any other program that can read a tab delimited text file.

Now that we have seen how to use the virtual instrument, we will try to understand the diagram that was used to build this virtual instrument. It is shown on the following page. We see that this program contains three levels of loops. The outer 'while loop' is an infinite loop. That is, it will keep looping

forever, or until the stop button on the front panel is pushed. The conditional for continuing the loop is connected to the boolean constant 'false' through the boolean operator 'not'. This is equivalent to saying (do while (not.false) ...) The 'not false' always produces the result 'true', so the condition for repeating the loop is always satisfied. Infinite loops are generally bad programming practice, and this structure might be one of the first things to eliminate if one was to try to improve this program.

Within the infinite while loop is another while loop. Wired to its conditional for continuing the loop is



the boolean control from the front screen that is labeled 'start timer' or 'stop timer'. Clicking on that control toggles this input between 'true' and 'false'. The default state for this control is 'false', and in this state the button on the front panel displays 'start timer'. In this state, the while loop thus executes once, then exits without repeating, since its conditional for repeating is false. When the control button is clicked it is set to 'true', and it displays 'stop timer' on the button. The while loop is then repeated until the button is clicked again, setting it back to its 'false' state.

Within the inner 'while' loop is a 'for loop'. It is within this loop that the data acquisition actually takes place. The number of times to repeat this loop before exiting is given by the digital control that is

labeled 'Number of Readings to Average'. It is wired to the input that specifies the number of times to go through the loop. Each time through the loop, the program reads the three temperatures from the board, and then waits a specified number of milliseconds. To read a temperature, the program uses the 'AI ONE PT' vi to get a voltage reading from the data acquisition board to which the thermocouples are connected. The inputs to this vi, from top to bottom, are the device number to read from, the channel of that device to read from, the maximum value expected and the minimum value expected. Thus, the top 'AI ONE PT' shown above reads channel 1 of device 1, and expects a reading of between -0.0001 and 0.0001 volts. The thermocouple that is placed in the hot water bath is physically connected to the input terminals of this channel. Likewise, the next 'AI ONE PT' shown reads channel 0 of device 1, and expects a reading between 0.00 and 0.50 volts. Channel zero is connected to an integrated circuit temperature sensor on the data acquisition board. This temperature sensor provides the known temperature that is required for us to use 'cold junction compensation' to obtain a temperature from the thermocouple voltage. This is discussed in more detail in the lecture on thermocouples. The voltage reading from each of the three thermocouples is sent to a vi called 'THERMOLINEAR'. This is a vi, provided with LabView, that converts a thermocouple voltage to a temperature. As inputs it requires the thermocouple voltage, a voltage from the temperature sensor that is being used for cold junction compensation, a specification of the thermocouple type, and a specification of the type of temperature sensor being used for cold junction compensation. The voltage inputs are wired to the appropriate outputs of the 'AI ONE PT' vi's. The other inputs are specified as constants - we are using type T thermocouples for the temperatures we want, and an IC Sensor for the reference junction temperature. The outputs from the 'THERMOLINEAR' vi's are the measured temperatures. These are sent out of the for loop. You will notice that when the wires leave the loop, they become thicker. This is because the loop has 'auto-indexing' enabled. This means that it takes the value at each iteration and adds it to a list (a one-dimensional matrix, or vector), so that when the loop finishes, the output is a list of numbers whose size corresponds to the number of iterations that were completed. Each list of measured temperatures goes to a 'MEAN' vi. This is another vi that is provided with LabView. This vi simply takes the average of the list of numbers. Its output is therefore a single number, as indicated by the narrow line leaving it. The outputs from the 'MEAN' vi's are sent to the digital indicators that are shown on the front panel. They are also sent out of the 'while' loop for later storage in the data file. After each completion of the 'for' loop, the time is read, and the time before entering the 'while' loop is subtracted from it. The result is the time elapsed since the 'while' loop was entered (the time elapsed since the 'start timer' button was clicked). This is also displayed on the screen and sent out of the while loop for later storage in the data file. Also within the while loop, there is a logical check to see if the iteration number is greater than 1. This is used to decide whether an output file should be written. If the while loop only executed once, then the 'start timer' button has not been pushed, and we do not want to record a data file. If the while loop has executed multiple times, then we have lists of data to record.

When the data leaves the inner while loop, auto-indexing is used again. The heavy lines leaving the loop indicate that we have lists of numbers (arrays of dimension 1). These are the times and the temperatures read at each time. They enter a 'case' structure. The 'true' case is shown above. The 'false' case is empty (if the control variable is false, nothing is done). The control input of the case structure is connected to the output of the check on whether the number of times through the inner while loop was greater than one. If it was greater than one, then the program will execute the 'true' case, if not, then it

will execute the 'false' case, which does nothing. The 'true' case writes the lists of times and temperatures to a file. The first vi that the data enter is the 'build array' vi. It simply takes the one-dimensional arrays of numbers and puts them into a 2-dimensional array. This 2-dimensional array (indicated by a double line) is sent to the 'write to spreadsheet' vi. This vi opens a dialog to ask the user for a name for the file to write, and then writes the 2-dimensional array of numbers to the file, separated with tabs and carriage returns. The resulting file can be read by Excel and many other programs.

This discussion is intended to give you some idea how the data acquisition program (or virtual instrument) works. To understand it in detail, you will have to work through the diagram itself. All of the vi's have online help that explains what their inputs and outputs are, what they do, and how they do it.