Lesson 12
Using VEE Pro for Application Simulations

This lesson will examine how to use VEE Pro for application simulations. It consists of a pre-lab and five labs.

Lesson 12 Pre-lab Summary

The following are described in the Lesson 12 pre-lab. See Appendix E, page E-64.

- The Instrumentation Amplifier
- Understanding a Strain Gauge
- The VEE Pro-provided application programs
- MATLAB® Script
- MATLAB® Function & Object Browser

As noted in Lesson 1, Appendix B includes a cross-reference to each of these items and to all objects and subprograms in the labs of this and later lessons.

Overview

Lab 12.1 – Simulating an Instrumentation Amplifier
This lab will show you how to generate a differential signal with an ideal instrumentation amplifier, examine the ability of an IA to measure a small signal buried in noise, and change the differential gain of the IA by a factor of ten.

Lab 12.2 – Simulating a Strain Gauge
This lab will show you how to construct a four-element simulated strain gauge using simulated strain-gauge resistors and a Wheatstone bridge.

Lab 12.3 – Exploring Four Mechanical Simulations
This lab will show you how to access and explore four application programs embedded within VEE Pro. These applications are the cantilever beam deflection, angular deflection of a round torsion shaft, the natural frequencies of a coil spring, and heat transfer coefficients for several body or surface types.

Lab 12.4 – Exploring a Simulated Manufacturing Test System
This lab will show you how to combine features of VEE Pro to build a manufacturing test system.

Lab 12.5 – Plotting Simulated Vehicle Radiator Temperatures via MATLAB®
This lab will show you how to use MATLAB® to display changes in the Vehicle Radiator temperature distribution.

Lab 12.1 – Simulating an Instrumentation Amplifier

This lab will show you how to generate a differential signal with an ideal instrumentation amplifier, examine the ability of that amplifier to measure a small signal buried in noise, and change the differential gain of the amplifier.

1. Clear your Work Area, deselect the Program Explorer, and maximize Main.

Generating a differential signal with an ideal instrumentation amplifier

2. Select Menu Bar => Device => Virtual Source => Function Generator; place it in the upper left corner of Main.
3. Clone the Function Generator object and place at the lower left corner of Main; reduce the size of both internal Function Generator objects as much as possible.
4. Select Menu Bar => Device =>Virtual Source => Noise Generator; reduce its size to as small as possible and place it between the two Function Generator objects.
5. Select Menu Bar => Device => Formula; place it to the right of the upper Function Generator; add an input terminal by placing the cursor over the input terminal area and pressing Ctrl-a; type A+B in the formula expression.
6. Clone the Formula object and place it to the right of the lower Function Generator object.
7. Rename the Title Bars of the on-screen objects from their Property Menus as follows:
   The Upper Function Generator: V_{in}(+)
   The Noise Generator: Common Mode Noise Voltage (CMV)
   The Lower Function Generator: V_{in}(-)
   The Upper Formula Object: V_{in}(+) + CMV
   The Lower Formula Object: V_{in}(-) + CMV
8. Interconnect the on-screen data pins as follows:
   V_{in}(+) Func data-output pin to V_{in}(+) + CMV data input pin A.
   CMV Func data-output pin to V_{in}(+) + CMV data-input pin B.
   CMV Func data-output pin to V_{in}(-) + CMV data-input pin A.
   V_{in}(-) Func data-output pin to V_{in}(-) + CMV data-input pin B.
   Convert the V_{in}(+) + CMV and V_{in}(-) + CMV objects to icons.
   (See Figure 12.2, page 12.4 for labeling and connection verification.)
9. Select Menu Bar => Device => UserObject and place it in Main; double-click the resulting icon; change the name of the UserObject in the Title Bar to Instrumentation Amplifier.
   Place the cursor in the left margin of the Instrumentation Amplifier UserObject and press Ctrl-A; double-click the terminal and rename it Plus Input (A). Add another terminal using the same procedure and name it Minus Input (B). Add an output terminal to the Instrumentation Amplifier using the same method, and name the terminal:
   Out = K1(A) - K2(B).
11. Select Menu Bar => Device => Formula; place it to the right of, and in line with, Plus Input (A); rename it +10X Gain (K1).
12. Go to its formula expression to: type 10*A.

13. Clone +10X Gain (K1); place it to the right of, and in line with, Minus Input (B); rename it -10X Gain (K2); change its input terminal from A to B.
14. Go to its formula expression to: type -10*B.
15. Connect the input pin of each Formula Object to its associated UserObject input pin.
16. Clone the upper Formula Object; place it to the left of the UserObject output terminal; change its name to Adder K3*(A + B); add another input terminal.
17. Go to its expression; change it to: 1*(A+B).
18. Connect the Adder K3*(A + B) input pins to the respective output pins (Result) of the other two formula objects; connect its output pin to the UserObject output pin. See Figure 12.1.

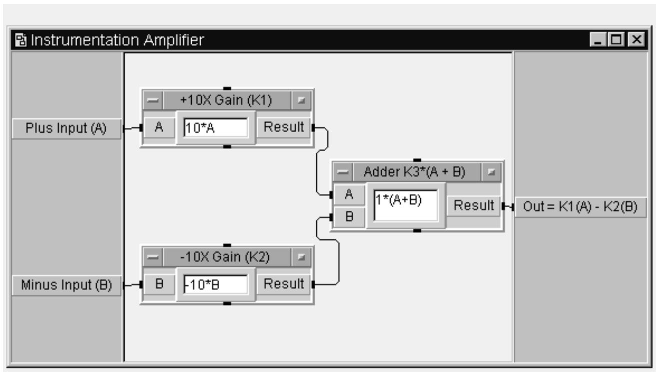

Figure 12.1. The Instrumentation Amplifier UserObject

19. Reduce the Instrumentation Amplifier UserObject to an icon – click on its (_) icon.
20. Connect the V_{in}(+) + CMV output pin to the Instrumentation Ampl upper input pin Plus Input (A).
21. Connect the V_{in}(-)+ CMV output pin to the Instrumentation Ampl lower input pin Minus Input (B).
22. Select Menu Bar => Display => Waveform (Time); place it to the right of Instrumentation Amplifier UserObject; reduce the size of the display so it fits on the screen.
23. Connect the Instrumentation Amplifier output pin to the Waveform (Time) input pin (Trace1).
24. Save your program as LAB12-1.
25. Run this program; your output should be a zero line on the scope (Waveform (Time)).

Examining the ability of an IA to measure a small signal buried in noise

26. Go to the V_{in}(+) amplitude; change it to 10.
27. Go to the V_{in}(-) amplitude; change it to 9.99.
28. Save your program again.
29. Run your program; note the scope amplitude change. See Figure 12.2.
    Note: You are observing the amplified differential-input amplitude on the scope. You are measuring a 10mV differential-input signal in the presence of a common-mode error voltage of 1V peak.

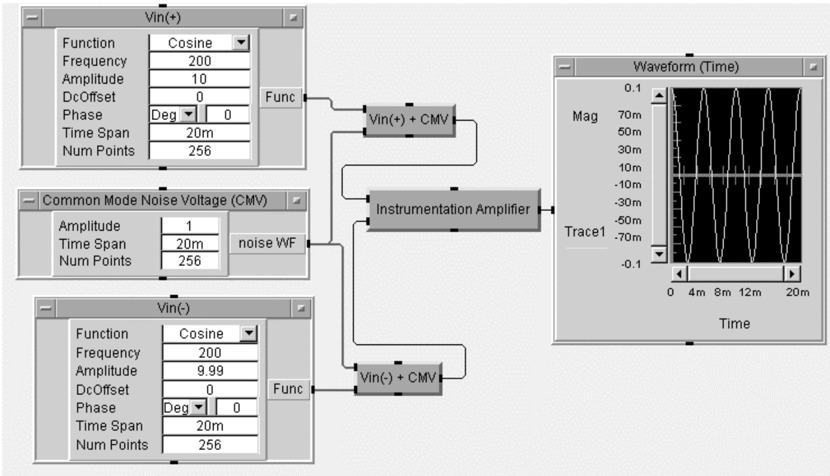

Figure 12.2. Display of a common mode voltage

◊ 30. Re-run this program several times entering differential-input voltages; observe the Waveform (Time) display result; note the absence of common-mode noise.
31. Close your program without saving it; re-open it as LAB12-1.

Changing the differential gain of the instrumentation amplifier by a factor of ten

32. Open the Instrumentation Amplifier UserObject; go to the Adder…; change its formula white-space field from 1*(A+B) to 10*(A+B); close this UserObject.
Note: This gain change will increase the instrumentation-amplifier sensitivity by a factor of ten; it demonstrates that this does not degrade the common-mode rejection ratio so long as the absolute gain value of the preceding two amplifiers exactly match.
33. Save this program as LAB12.1a; run this program.
34. Change the Instrumentation Amplifier Adder… back to 1*(A+B); change the -10X Gain (K2) Formula Object to -(9.99*B); run this program.
◊ 35. Run this program several times with a variety of Common Mode Noise Voltage (CMV) amplitudes. Note the effect of this noise on the scope display.
Note 1: In the real world, operational-amplifier gain-value exact matching is impossible. This portion of the lab demonstrates the effect. Also, external circuit imbalances can cause common-mode error because they will modify the differential inputs unequally.
Note 2: Other (identical) gain values may be applied in steps 12 and 14.
36. Close this program without saving <u>unless</u> you desire to keep the changes.

Lab 12.2 – Simulating a Strain Gauge

This lab will simulate a four-element strain gauge. You will then study its sensitivity and the range of output strain and accompanying voltages by varying five different parameters.

1. Clear your Work Area, deselect the Program Explorer, and maximize Main.

Constructing a four-element simulated strain gauge

2. Select Menu Bar => Display => Note Pad. Place it in the upper-right corner of your Main window. Type the following information into its editing area:
   This is a simulation of a four-element strain gauge.
   It is monitoring a mechanical structure.
   This implementation also compensates for the structure's temperature change.
3. Size the Note Pad.
4. Select Menu Bar => Data => Continuous => Real64 Knob. Place it in the upper-left corner of your main window. Go to Properties, change its title to Gauge Excitation Voltage (V_{exc}), its Label Spacing to Every Major Tic, and its maximum value to 10.
5. Clone Gauge Excitation Voltage. Place the new Object to the right of Gauge Excitation Voltage. Change its title to Gauge Full Load (N/sq m), its Label spacing to Every Other Tic, its maximum value to 200, and its minimum value to "2".
6. Select Menu Bar => Data => Continuous => Real64 Knob. Place it below Gauge Excitation Voltage. Change its name to Gauge Sensitivity (mV/V_{exc}), its Label Spacing to Every Other Tic, and its maximum value to 10m.
7. Clone Gauge Sensitivity. Place the new Object to the right of Gauge Sensitivity. Change its title to Gauge Input Load (N/sq m) and its maximum value to 10.
8. Select Menu Bar => Device => Formula, place it between the real knobs.
   Note: This may require that you "size" the other four objects so the Formula Object will fit between them.
9. Add a Data Input Terminal. Change the A data-input terminal name to: Vexc, change the B data-input terminal name to: Sensitivity. Change its formula to read: Vexc*Sensitivity. Change its title bar to Strain Gauge (mV). See Figure 12.3.

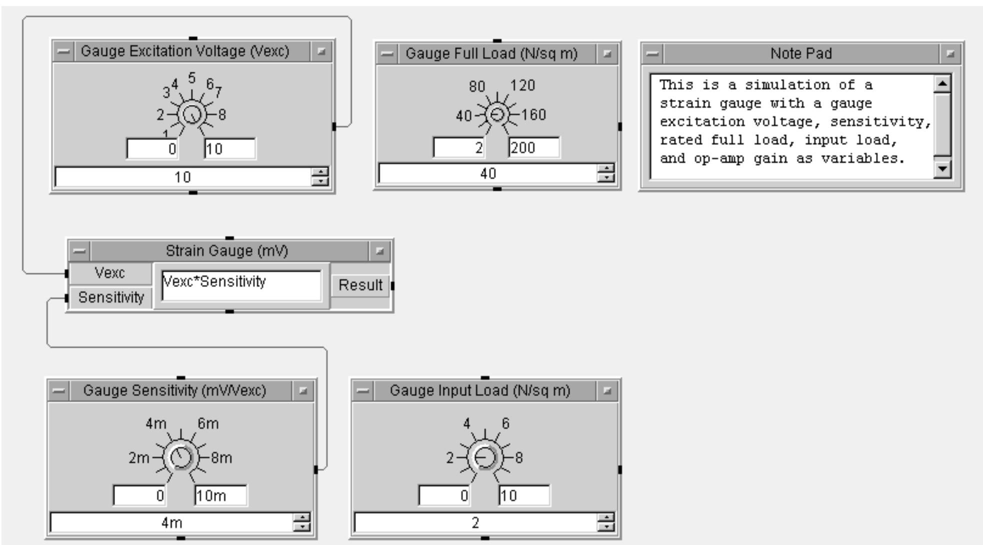

Figure 12.3. Strain Gauge simulation; initial construction

10. Connect the output pin of Gauge Excitation Voltage to the Vexc input pin of Strain Gauge Output; connect the output pin of Gauge Sensitivity to the Sensitivity input pin of Gauge Output. See Figure 12.3 to verify your connections.
11. Select Menu Bar => Device => Formula, place it the right of Strain Gauge Output.
12. Add two Data Input Terminals. Change the A data-input terminal name to: Full_Load, change the B data-input terminal name to Gauge_Output. Change its C data-input terminal name to Input_Load.
13. Change its formula to Gauge_Output*(Input_Load/Full_Load). Change its title bar to Strain Gauge Output (mV).
14. Connect the Gauge Full Load Object output pin to the input pin “Full Load” of the Strain Gauge Output Object.
15. Connect the output pin (Result) of Strain Gauge Output to the input pin “Gauge_Output” of the Strain Gauge Output Formula Object.
16. Connect the output pin of Gauge Input Load to the input pin “Input_Load” of Strain Gauge Output.
17. Select Menu Bar => Device => Formula, place it below Strain Gauge Output. Change its title bar to Op Amp with Gain K.
18. Change its formula to 1*A. Connect the Strain Gauge Output to the Op Amp input.
19. Select Menu Bar => Display => AlphaNumeric, place it below Op Amp. Change its name to Amplified Strain Gauge Output (V). Connect its input to the Op Amp output. Verify your Object positioning and connections using Figure 12.4.

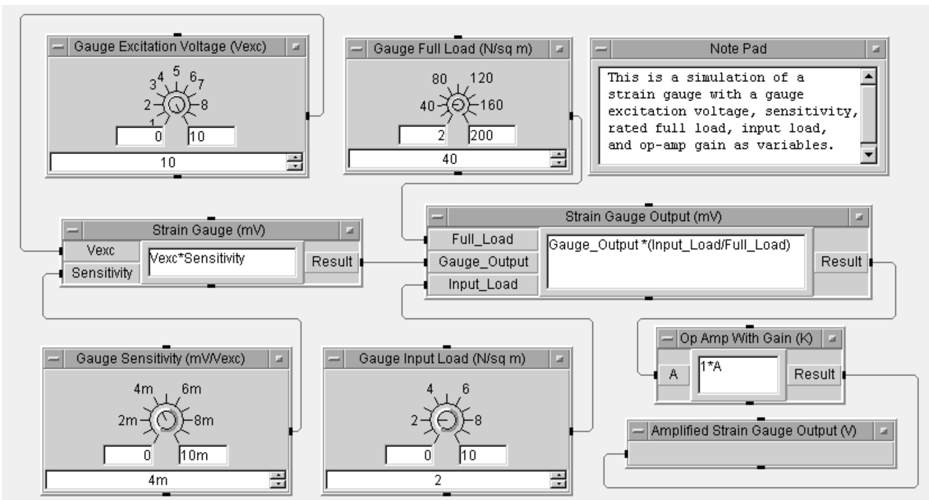

Figure 12.4. Strain Gauge simulation; total construction

20. Set the Gauge Excitation Voltage to its maximum value (10), the Gauge Full Load to 40, the Gauge Sensitivity to 4m and the Gauge Input Load to 2.
Note: Your VEE Pro program may not allow you to set your values at exactly the values given; “close enough” is satisfactory.

21. Save your program as LAB 12-2; run this program; observe the Amplified Strain Gauge Output value, can you justify the small value of the result? Complete the following tables.

Gauge Excitation Voltage	Gauge Sensitivity	Strain Gauge (Result)
Ours: 10V	$4mV/V_{exc}$	40mV = 0.04V
Yours:		

Gauge Full Load	Gauge Input Load	Strain Gauge Output (Result)
Ours: 40 N/sq m	2N/sq m	0.002V
Yours:		

Strain Gauge Output (Result)	Op Amp Gain	Amplified Strain Gauge Output
Ours: 0.002 V	1	0.002 V = 2mV
Yours:		

Note: Are your observed values more understandable?

◊ 22. Vary a number of parameters; record their values in the tables below, including those that are not allowed by the program.

Gauge Excitation Voltage	Gauge Sensitivity	Strain Gauge (Result)

Gauge Full Load	Gauge Input Load	Strain Gauge Output (Result)

Strain Gauge Output (Result)	Op Amp Gain	Amplified Strain Gauge Output

23. What have you learned from these assumptions? Is your op amp output value large enough so you could connect it directly to a multiplexer? What values would you have to choose so it would provide an overall range from 0 to 5V full scale into the multiplexer? Record these values in the following table.

Gauge Excitation Voltage	Gauge Sensitivity	Strain Gauge (Result)
Gauge Full Load	**Gauge Input Load**	**Strain Gauge Output (Result)**
Strain Gauge Output (Result)	**Op Amp Gain**	**Amplified Strain Gauge Output**

24. Save and close this program as LAB12-2.

Lab 12.3 – Exploring Four Mechanical Simulations

This lab will show you how to access and explore four application programs embedded within VEE Pro.

1. Go to the hard-drive location where the program files are stored; open the Program Files folder. (You may use Ctrl F as an option to find the Program Files folder.)
2. Open, in the following sequence, the Agilent folder, the VEE Pro 6.2 folder, the examples folder, and the Applications folder.

Examining a cantilever beam deflection problem

3. Open Beam.vee; change to Detail View; click on the "Info About" icon and study its contents. **Note:** If you desire more information regarding the program content, then click on the third icon on the left side of the title bar. It will display: "To Detail" if you hold your mouse over that icon for a few moments.
4. Go to the Formula object labeled "length(numElem,from,thru)"; note the object Description.
5. Run this program; it should look like Figure 12.5.

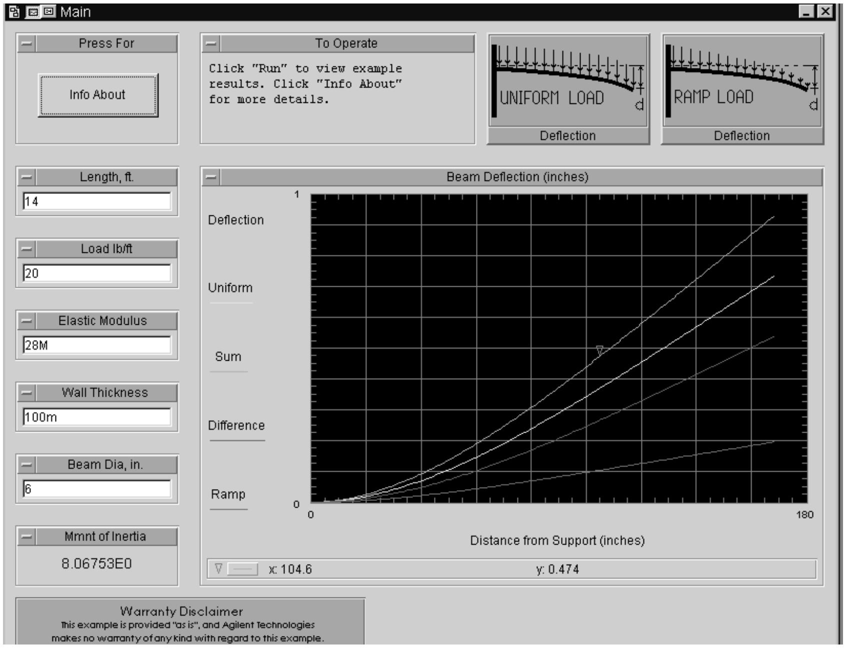

Figure 12.5. Beam.vee program after running

◊ 6. Change the parameter values such as beam length, elastic modulus, load, beam diameter, and wall thickness. Record the results in an Excel™ spreadsheet embedded in a Word™ report (if desired).
7. Close this program; rename and save it if you want your program modifications retained.

Examining the angular deflection (torsion) of a round shaft

8. Open torsion2.vee; change to Detail View; click on the "Info" icon and study its contents.
9. Go to the object labeled "Modulus of Elasticity/Rigidity"; note the description of this UserObject.
10. Repeat step 9 for "Young's Mod(ulus), Shear Mod(ulus), and whatever other icons are of interest.
11. Run this program; it should look like Figure 12.6.

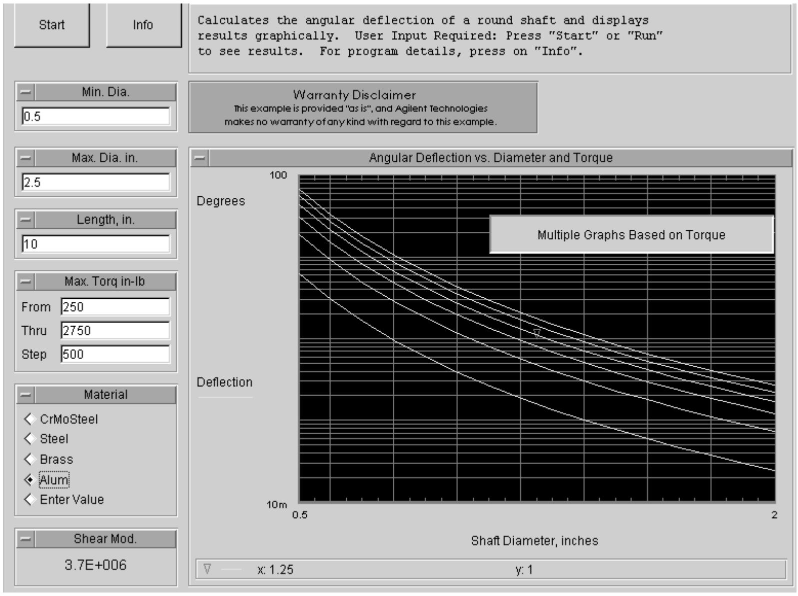

Figure 12.6. Deflection in degrees versus shaft diameter

◊ 12. Change the values of such parameters as pipe diameter (max and min), length, Max torque, and material type. Record the results in a Word™ report if desired.

13. Close this program; rename and save it if you want your program modifications retained.

Examining the natural frequency of a coil spring

14. Open coilspr.vee; change to Detail View; click on the "Info About" icon and study its contents. **Note:** There are no descriptions of interest within any of the icons.
15. Run this program; it should look like Figure 12.7.

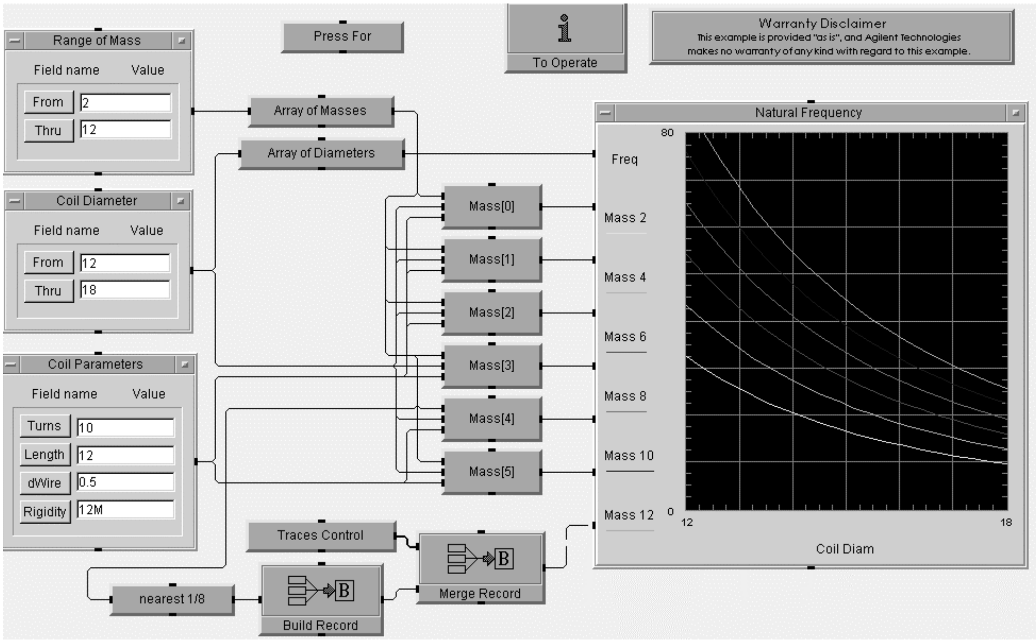

Figure 12.7. Display of coil spring natural frequencies

◊ 16. Change the values of such parameters as Range of Mass, Coil Diameter, and Coil Parameters (Turns, Length, dWire, and Rigidity). Record the results in a Word™ report if desired.

17. Close this program; rename and save it if you want your program modifications retained.

Examining the natural convection of heat

18. Open Convcoef.vee – The Natural Convection of Heat application; change to Detail View.
19. Click on the "Info About" icon and study its contents.
    Note: If you desire more information regarding the program content, then click on the third icon on the left side of the title bar. It will display: "To Detail" if you hold your mouse over that icon for a few moments.
20. Run this program. It will look like Figure 12.8.
    Note: The Flow Range display indicates that the "Flow is in Laminar Range". It will change to "Flow is in Turbulent Range" for other setting combinations.

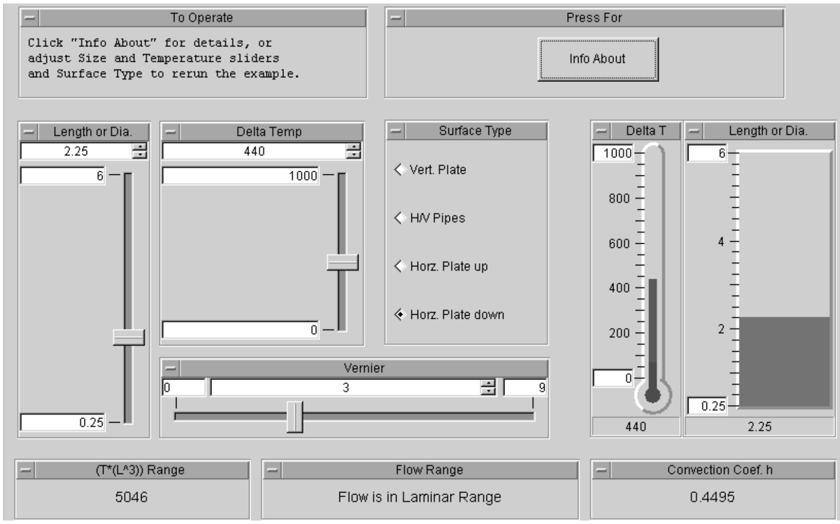

Figure 12.8. Convection Coefficient program after running

◊ 21. Change the values of such parameters as Length or diameter, Delta Temp, Surface Type, and Vernier. Transfer the “run” program to a Word™ report if desired via “Print Screen”.
22. Close this program; rename and save it if you want your program modifications retained.

Lab 12.4 – Exploring a Simulated Manufacturing Test System

This lab will show you how to combine features of VEE Pro to build a manufacturing test system.

Examining a manufacturing test system program

1. Clear your Work Area, deselect the Program Explorer, and maximize the Work Area.
2. Go to the hard-drive location where the program files are stored; open the Program Files folder.
3. Open, in the following sequence, the Agilent folder, the VEE Pro 6.2 folder, the examples folder, and the Applications folder.
4. Open mfgtest.vee; change to Detail View; click on the “Info About” icon and study its contents. The program should look like Figure 12.9.

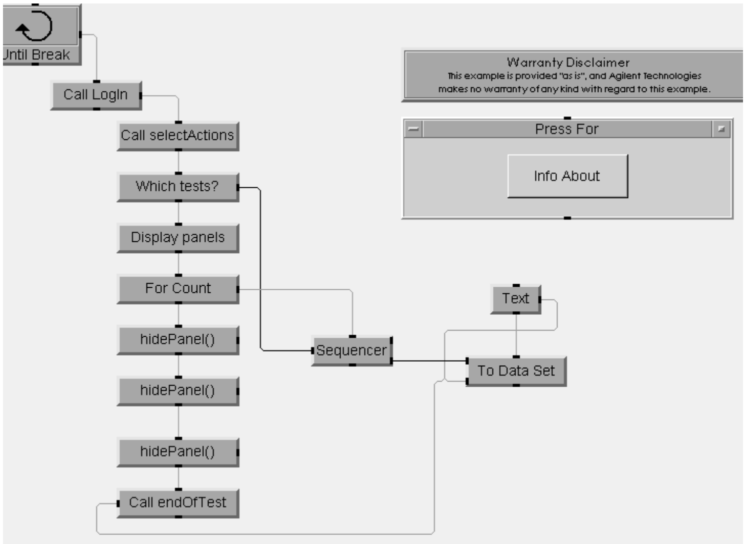

Figure 12.9. Program description for Mfgtest.vee

5. Run this program; select your login name from the list of 4; select the default password.
6. Examine the "run" program first step; it should look like Figure 12.10.

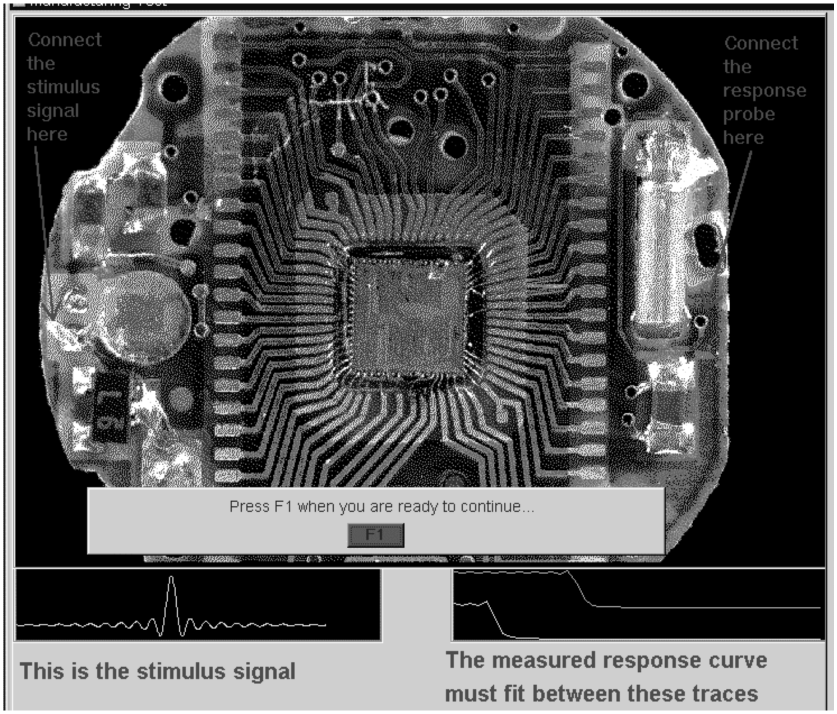

Figure 12.10. Mfgtest.vee illustrating the circuit under test

7. Observe the manufacturing tests cycling; turn off the program.
   Note: These tests are
   Impulse Test
   Power Consumption
   Input Impedance
   Output Impedance
   Crossover Power
   Crossunder Power
   Gain
8. Examine the content and subprograms within the two dialog boxes: LogIn and Which Tests?
9. Close the mfgtest.vee program.

Lab 12.5 – Plotting Simulated Vehicle Radiator Temperatures via MATLAB®

In this lab you are to use MATLAB® to display changes in the Vehicle Radiator temperature distribution.

1. Go to your computer hard drive; type **Ctrl f** (the "find" command); enter Surf3Dplot.vee which will locate the MATLAB® graphing program.
2. Double-click on the icon displayed.
3. Save this program in your lab library as LAB12-5.

Applying MATLAB® example displays to an existing Vehicle Radiator application

4. Change the title of the Function Generator to read: Vehicle Radiator Temperature.
5. Change the following parameters in the Vehicle Radiator Temperature Object:
   Function: Tri
   Frequency: 50
   DcOffset: 190
6. Cut Warranty Disclaimer and Note Pad.
7. Open Alloc Real64; change Dim 1 from 41 to 61.
8. Change the upper For Range parameter Thru from 1 to 15; Step from 50m to 0.5.
9. Change the lower For Range parameter From 0.95 to 14.5; Step from –50m to –0.5
10. Change MATLAB® Script.PLOT as follows:
    axis ([0, 50, -1, 1]) to ([0, 50, 180, 200]) with shown spaces within the brackets.
11. Change MATLAB® Script.SURF as follows:
    axis ([0, 50, 0, 60, -1, 1]) to ([0, 50, 0, 60, 180, 200])
12. Save this program again; arrange the icons as shown in Figure 12.11.

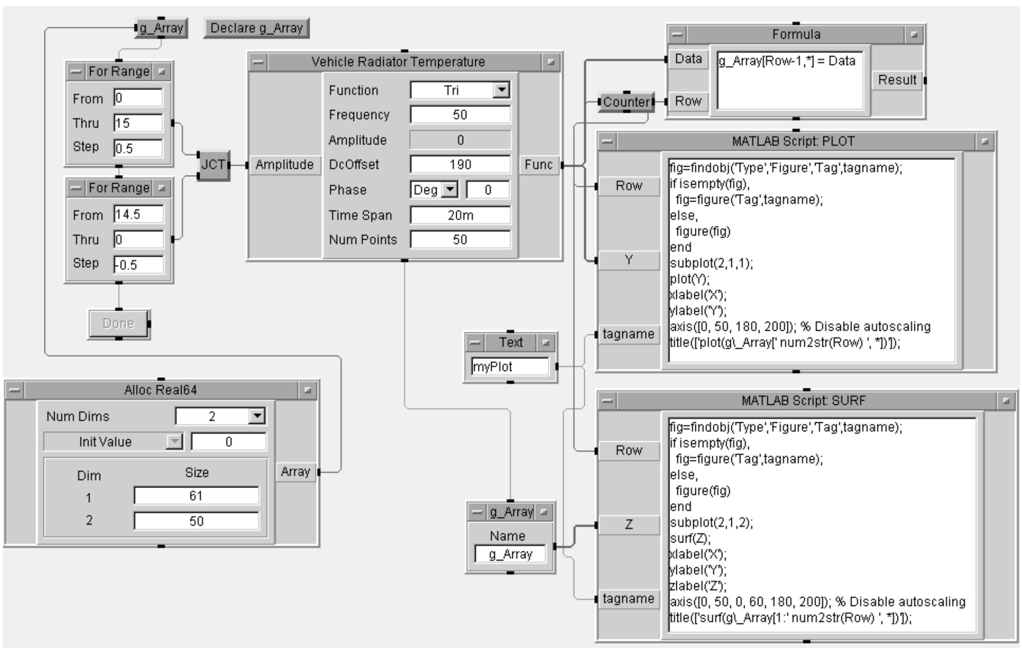

Figure 12.11. Lab 12.5 before running

13. Run this program; the result should look like Figure 12.12.
    Note: First a straight horizontal line will appear at the top. This line gradually becomes a triangular wave that generates a second plot that becomes the figure shown below.

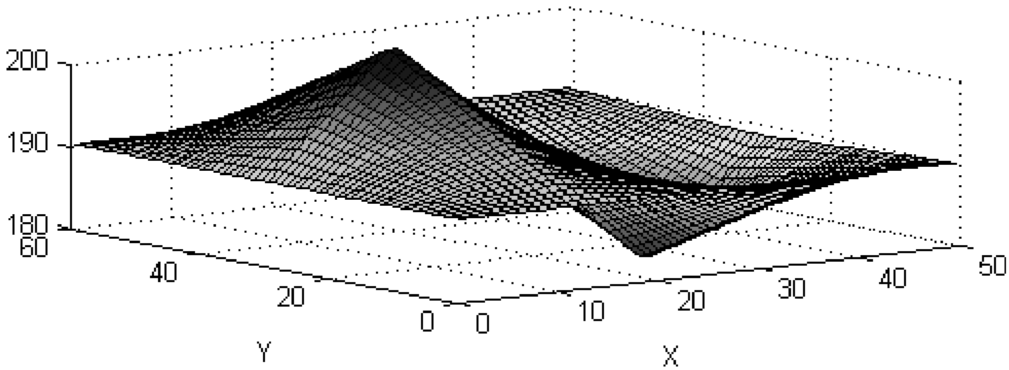

Figure 12.12. A three-dimensional plot of Vehicle Radiator data

14. Go to the upper-right corner of the Figure #1 Tool Bar; click on: Rotate 3D.
15. Place the cursor on the right corner of the 3D plot; turn it in small steps until the peaks are beyond the 180 and 200 Z scale. (This indicates that the scale of the plot is greater than the scale of the display.)
    Note: Changing parameters within this program requires knowledge of both MATLAB® and VEE.
16. Turn the plot until it becomes "two dimensional". Two white spaces will appear; they indicate that the data exceeds the scale of the plot. See Figure 12.13.

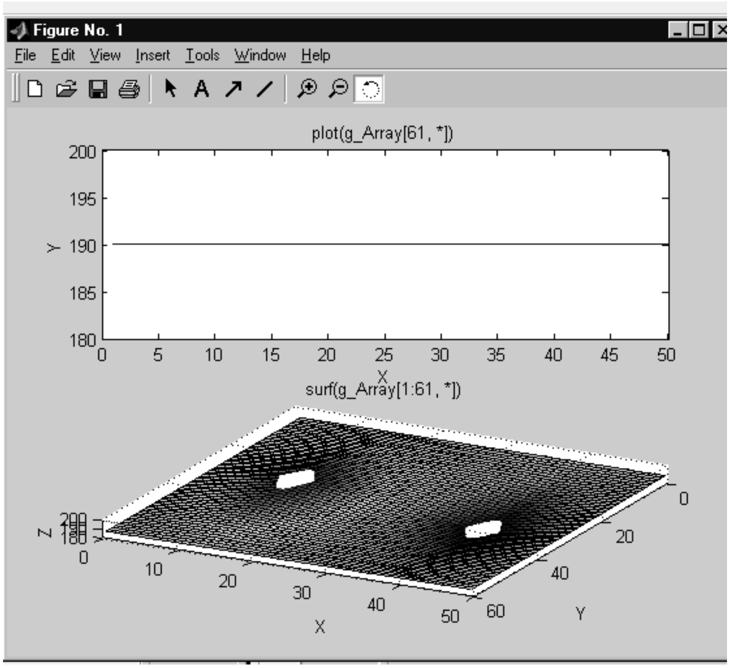

Figure 12.13. Vehicle Radiator temperature rotated 3D graph

17. Change MATLAB® Script.PLOT as follows:
    axis ([0, 50, -1, 1]) to ([0, 50, 170, 210]) with shown spaces within the brackets.
18. Change MATLAB® Script.SURF as follows:
    axis ([0, 50, 0, 60, -1, 1]) to ([0, 50, 0, 60, 170, 210])
19. Rotate the plot to verify that all points are within the graph.
20. Save this program as LAB12-5a; close this program.

Lesson 12 Summary

You have used VEE Pro to simulate a variety of applications. The first two of these labs focused upon unique, derived simulations. The second two of these labs focused upon VEE Pro-provided applications.

Lab 12.1 showed you how to generate a differential signal with an ideal instrumentation amplifier, examine the ability of an IA to measure a small signal buried in noise, and change the differential gain of the IA by a factor of ten.

Lab 12.2 showed you how to construct a four-element simulated strain gauge using simulated strain-gauge resistors and a Wheatstone bridge.

Lab 12.3 showed you how to access and explore four application programs embedded within VEE Pro. These applications are the cantilever beam deflection, angular deflection of a round torsion shaft, the natural frequencies of a coil spring, and heat transfer coefficients for several body or surface types.

Lab 12.4 showed you how to combine features of VEE Pro to build a manufacturing test system.
Lab 12.5 showed you how to use MATLAB® to display changes in the Vehicle Radiator temperature distribution on a three-dimensional plot.

You are now ready to simulate virtual functions and relations, and explore a VEE-prepared program for a filter.