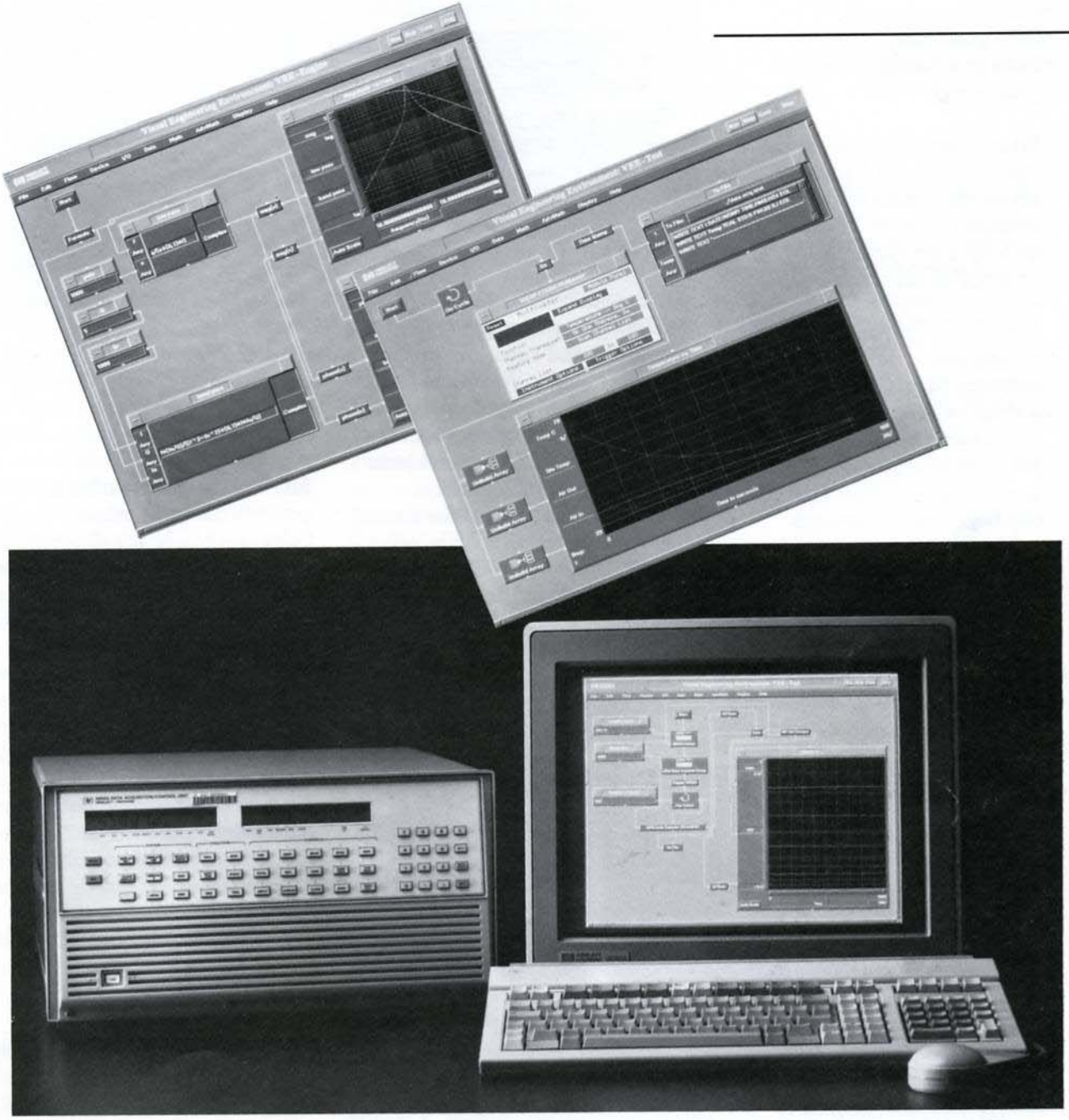
HEWLETT PACKARD

Complete Data Acquisition Solutions with HP VEE-Test

Application Note 1206-01



Data Acquisition Needs

Product research, development, and manufacturing, as well as scientific research, all rely on data acquisition solutions for characterization, monitoring, and control. A complete data acquisition solution usually consists of three areas:

- Instrument control
- Data gathering
- Data processing and report generation

While some sophisticated card cage devices can handle all three areas, most data acquisition solutions require a controller, usually a computer, to accomplish all of the tasks.

Data gathering and control with a computer is generally done by software that sends command strings or programs to the instrument through an I/O path. When data is gathered, data processing and report generation can begin, again through software programmed to handle things like data conversion, signal processing, curve fitting, and graphical display of data as appropriate.

The most widely-used computer environments today perform these tasks at a fairly low level. These environments require explicit knowledge of programming languages, instrument commands, and I/O interface drivers, all involving a lot of syntax. Providing a full, correct solution to a data acquisition problem in these environments requires a great deal of learning, programming, and debugging software, so solutions are time-consuming, costly, and difficult to optimize. Furthermore, current trends in data acquisition towards higher sample rates and user-friendly interfaces increase the investment in time, money, and knowledge even more.

Since creating solutions is so laborious, data processing and report generation are often left out, delegated after the fact to separate software tools. While this may cut development time, it often increases overall time spent getting results. Creating and testing solutions is an iterative process — if analysis of data and results is done later, problems that could be solved by optimizing placement of transducers, for example, will not be spotted until later. The developer will be forced to fix the problem and start the process all over again. A product that addresses *all* aspects of a solution concurrently reduces the time spent fixing problems and improves the quality, efficiency, and cost of the overall process.

Creating Complete Solutions

To create complete, optimum data acquisition solutions with a minimum of time, a software tool is needed that not only reduces development time but provides functionality for carrying out all data acquisition tasks concurrently. HP VEE-Test, HP's visual engineering environment for test and measurement applications, is such a tool. It allows you to build, test, and run complete solutions in a unique, time-saving way.

HP VEE-Test lets you create complete solutions simply by linking visual objects or icons, rather than using traditional programming languages. The direct manipulation of these objects is a highly efficient method for solving problems. The process is similar to creating block diagrams — an intuitive approach that saves considerable time and effort. Complex designs can be built quickly and in a modular fashion.

HP VEE-Test provides a wealth of objects to aid in the collection, analysis, and presentation of data, in addition to features for data storage, flow, modularity, and debugging. There is no syntax to remember; all you need is in the form of modular, functional objects within a powerful user interface. Solutions can be constructed quickly — and modifications can be made quickly. Because it is so visual, any solution created with HP VEE-Test has an implicit user interface — and it can be implemented with custom user interfaces as well. These attributes, combined with in-depth I/O support, make HP VEE-Test a powerful tool for data acquisition.

Data Acquisition Examples

Solutions to data acquisition depend on the physical constraints of the problem. Factors such as sample rate, number of channels, and types and placement of transducers are basic concerns in designing solutions. The number, type and sophistication of the instruments required contribute to the complexity of the solution. The following are three data acquisition problems of increasing complexity solved with HP VEE-Test.

Simple Data Acquisition Using Instrument Drivers

Figure 1 shows a simple data acquisition problem using a digital multimeter. Three thermistors are sampled every three seconds, the data is logged to a file, and then it is displayed.

HP VEE-Test gives you several choices for controlling instruments and getting data from them. Instrument drivers are objects in the form of soft front panels, available for over 170 HP instruments. Direct I/O objects are also available to define I/O transactions interactively and send instrument commands.

In this example, the multimeter is controlled by an instrument driver. The solution (also referred to as a model) consists of the instrument panel through which the multimeter is controlled, as well as data manipulation, storage, and display objects. This solution was constructed in a few minutes simply by pulling down the objects, configuring their options, and connecting them together.

Figure 1 represents a simple data acquisition problem. The number of transducers is small and the sample rate is not demanding. The instrument is relatively simple to use and control. Using an instrument panel within HP VEE-Test is a good match of instrument and software. The panel is configured for the type of measurement and the number of channels. During data acquisition, the multimeter is triggered by the panel's driver to take readings and place them on the interface bus. Through the panel, the data is retrieved as an array of floating-point numbers. Between the time the measurement is taken and when it is read from the interface bus, the multimeter cannot be triggered for another reading. The instrument operates in lock-step fashion with HP VEE-Test.

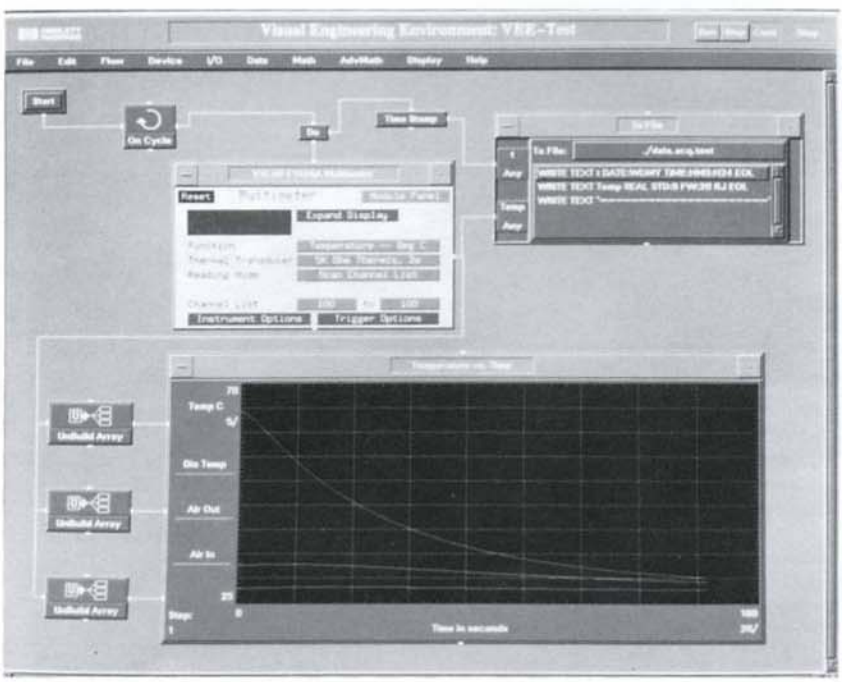


Figure 1. Simple data acquisition made simple.

Multiple readings are taken by using one of HP VEE-Test's control objects, the **On Cycle** iterator. This object, shown as an icon in **Figure 1**,executes the instrument panel at specified intervals, causing the multimeter to trigger. Each iteration causes three temperature readings to be gathered, time-stamped, logged to a file, and plotted on the **Strip Chart** display object. The next cycle of the iterator starts when the specified period is over. The time it takes to execute each iteration sets the sample rate. If the period specified in the **On Cycle** object is less than the execution time of each iteration, the multimeter is triggered to take readings immediately after each completion of the thread. This is not usually desirable, since the sample rate may not be consistent from one iteration to another. Thus, simple instruments that must operate in lock-step fashion will have a limited sample rate when used in an HP VEE-Test solution.

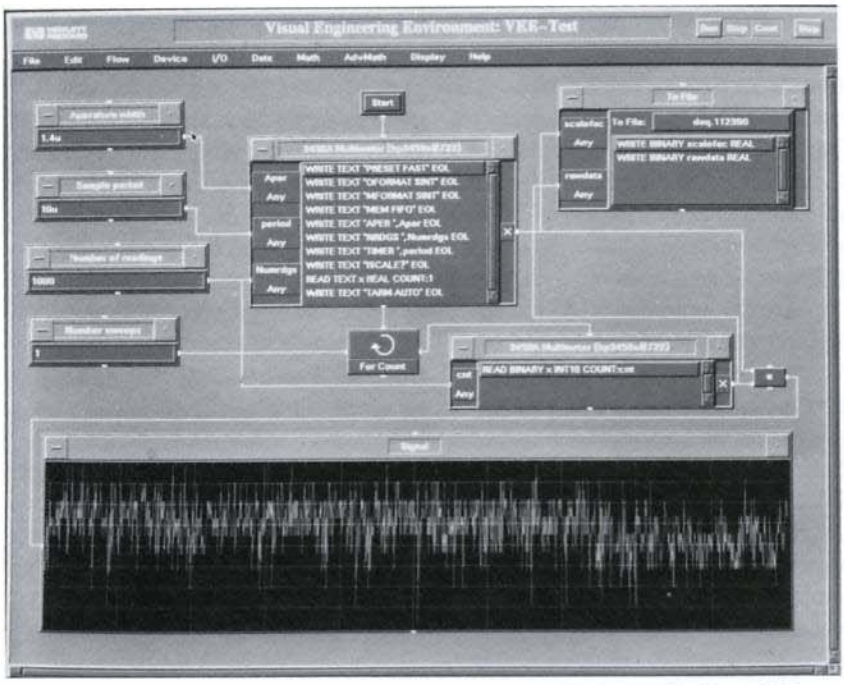


Figure 2. Flexible and powerful instrument control through direct I/O.

As the complexity of data acquisition problems increases, simple instruments give way to more sophisticated ones. To improve performance or in the case where instrument drivers are unavailabile, the direct I/O objects in HP VEE-Test are the best choice.

Data Acquisition Using Direct I/O and I/O Transactions

If a driver for a particular instrument does not exist, or if throughput must be increased, direct I/O objects can be used instead of drivers. These objects send and receive commands and data for any device, using HP-IB, RS-232, and GPIO interfaces.

Figure 2 shows a slightly more complicated solution that uses a high-speed HP 3458A multimeter to digitize a signal. In this example, a direct I/O object was used because it supports the multimeter's scaled 16-bit data type, a smaller type than the 8-byte real supported by the instrument driver. Since the scaled integer is smaller, less data must pass through the bus and overall efficiency is improved.

In this example, we configured a direct I/O object for the multimeter. Once configured, the object was used to send and receive data from the multimeter.

Specifying what data to send or receive from the multimeter is done with the direct I/O object by using I/O transactions. I/O transactions — HP VEE-Test's basic units for I/O control — are statements that specify how and what commands and data should be sent and received. These I/O transactions support a wealth of data formats and encodings. To configure them, the user has a series of dialog boxes by which to choose actions, encodings, formats, and notation. A direct I/O object with its dialog box is shown in **Figure 3**.

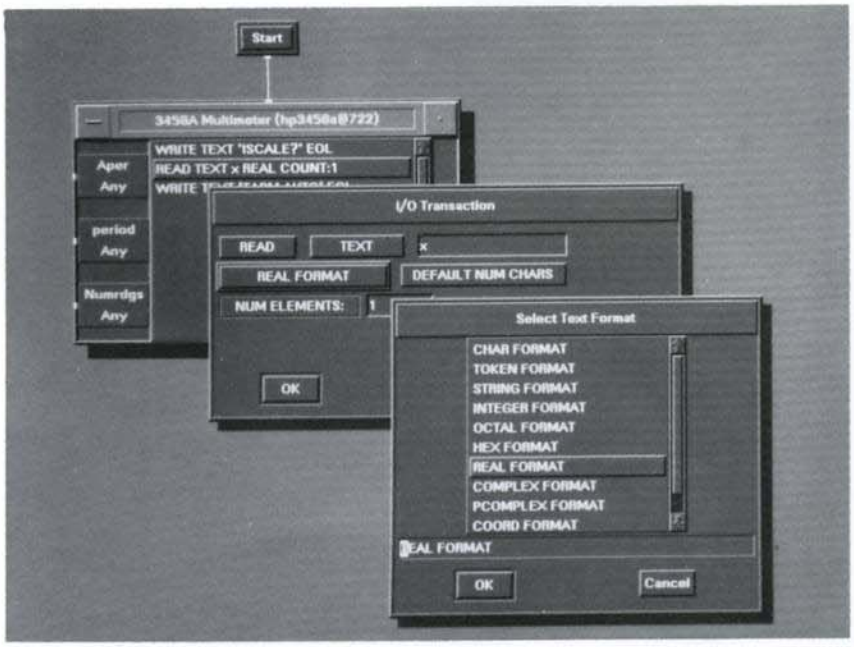


Figure 3. Configuring I/O transactions via dialog boxes.

In this example, the configured direct I/O object for the multimeter was used twice in the solution. In one object, shown again in **Figure 4**, nine transactions write commands that set up the voltmeter, while one reads back a scaling factor. The transaction in the other object, shown again in **Figure 5**, reads large blocks of data from the instrument in 16-bit scaled integer format. Since high sample rates were needed, this object was configured to receive large blocks of data rather than single readings, thus improving throughput.

Direct I/O objects in HP VEE-Test are also highly customizable, allowing you to define inputs and outputs as needed. The object shown in **Figure 5**, for example, allows you to enter parameters which are then sent to the instrument.

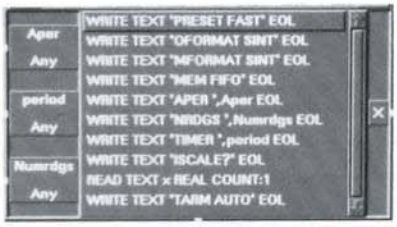


Figure 4. Direct I/O used for instrument configuration.

Figure 5. Direct I/O receiving blocks of data.

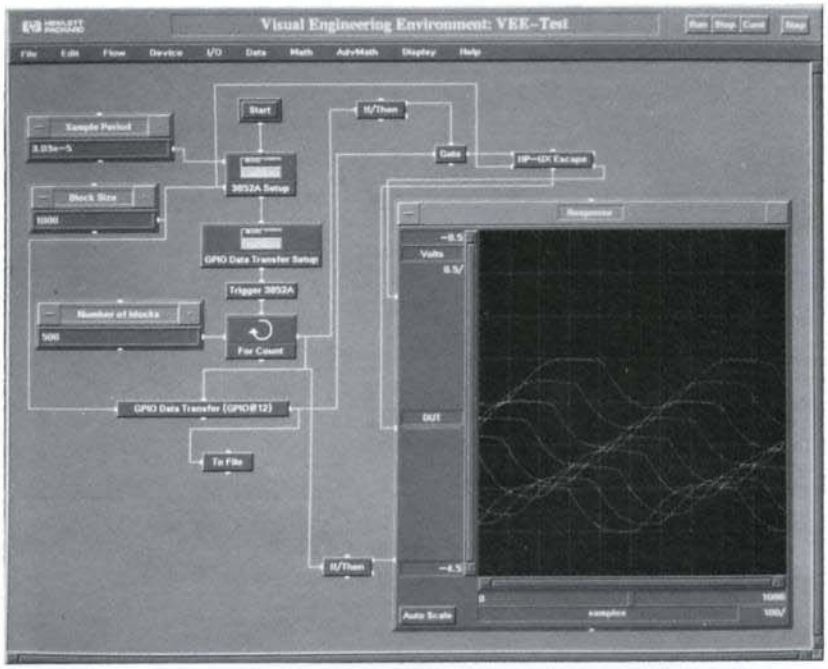


Figure 6. Countinuous, high-speed acquisition.

Data Acquisition at High Rates

When continuous data acquisition at high sample rates is needed, HP VEE-Test's direct I/O objects, combined with its data processing and analysis capabilities, can be used to create complete, optimized solutions.

Generally, only sophisticated instruments that allow distributed processing will support high rates continuously. After commands are sent to the instrument — usually in the form of downloadable subroutines — the instrument acquires data continuously and asynchronously without further direction. Usually the instrument contains internal memory which acts as a first-in/first-out buffer for the data transfer path. To keep up with the instrument, the computer must maintain a high rate of transfer from the interface bus.

Figure 6 shows an HP VEE-Test solution for a continuous, high-rate data acquisition problem. In this example, an HP 3852 data acquisition and control unit is used to continuously read 6 channels measuring the strain of a mechanical structure undergoing vibrational stress. The effective rate of acquisition is 30,000 samples per second. The high-speed digital voltmeter in the HP 3852 is set up and triggered by the HP-IB bus, while the data is gathered through its GPIO interface for faster throughput.

In **Figure 6**, two direct I/O objects are used to set up and trigger the voltmeter. Two other objects handle the initialization

and transfer of large blocks of measurements via GPIO. For HP-IB and GPIO, HP VEE-Test can read the bus, allocate memory, and continue with the processing and display at 190,000 and 420,000 bytes per second, on a HP 370 workstation respectively. Since most high-speed data acquisition uses compact data formats (16-bit scaled or packed) for transfer on a bus, and HP VEE-Test supports these formats, data can be logged to a file at 100,000 readings per second — the rate supported by many instruments.

A more complete solution may require data processing and display. In **Figure 6**, data processing consists of unpacking the readings. HP VEE-Test does this with analysis objects. In this example, however, we have chosen to use to use a separate process — executing in parallel — to obtain maximum throughput. In **Figure 6**, packed readings are sent to a separate process via the HP-UX Escape object. This object initiates and communicates with a program written in C, using the same I/O transactions as the direct I/O objects.

The throughput rates for complete processing and display in HP VEE-Test depend on the time it takes the acquisition segment to execute and the size of the blocks of data read by the I/O transactions. In **Figure 6**, this is seen as the total time it takes for a block of readings to be unpacked and displayed. The acquisition time and the block size determine the virtual sample rate in the example.

If this virtual sample rate is equal to the sample rate of the instrument, the instrument's memory will not fill up and block. If the virtual sample rate is larger than the instrument's memory, however, some readings could end up waiting to be uploaded. In this case, gathering, processing, and display of data would lag by a time proportional to the number of readings in the instrument's output buffer. The solution in **Figure 6** shortens this time by displaying only part of the readings as multiple curves. In this case, every tenth block of 1,000 readings is displayed; after 10 curves are displayed, the plot is cleared. In this way, the overhead of maintaining the solution is kept to a minimum. Using these techniques, complete solutions can be designed that maintain 30,000-35,000 readings per second.

Although previous discussions have emphasized the I/O capabilities of HP VEE-Test, its display and data processing strengths have also been used to devise complete solutions.

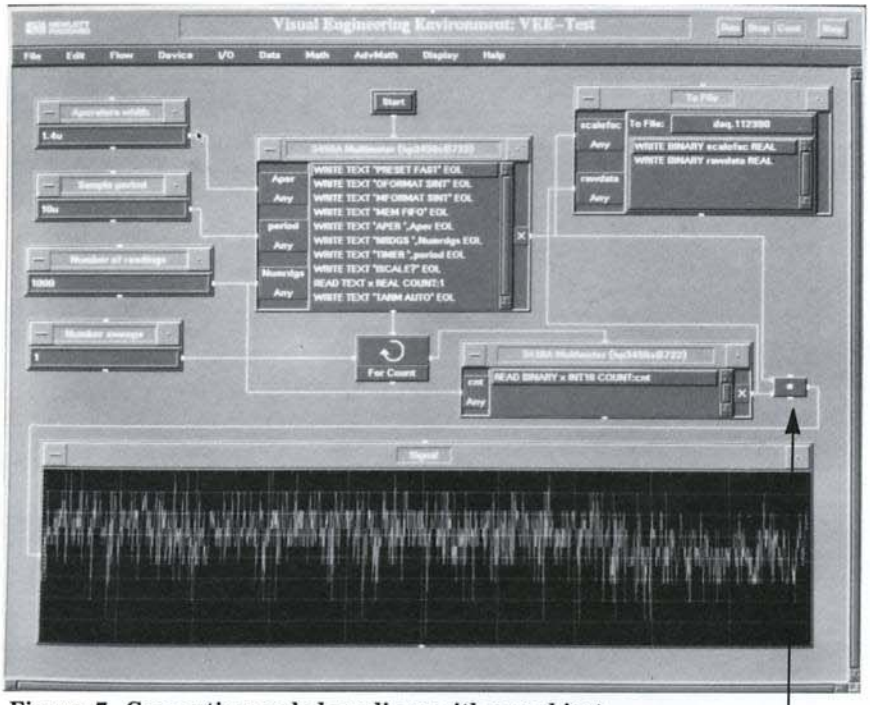


Figure 7. Converting scaled readings with one object.

Data Processing

In **Figure 7** the data acquired by the digital voltmeter is received as a block (or array) of readings in a scaled format. The data must be multiplied by the scaling factor. This is done by using the multiply object to scale each reading in the array. HP VEE allows all mathematical operations to function independent of the type and structure of the operands. In this case, the multiply object's operands are an array of 10,000 scaled readings in a 16-bit format, plus a single scaling factor in an ASCII format. The user does not have to consider the different data types or structures — the array is correctly multiplied by the scaling factor. This is done with one object rather than several lines of code needed in traditional data acquisition environments. All HP VEE objects automatically adapt to the type and structure of the operands.

Data Display

Presenting data, whether during or after acquisition, is the last and most important aspect of a solution. Most data processing is done to ready the data for display. HP VEE-Test supports a myriad of display objects, each configurable for full flexibility. The three displays in the solutions shown in **Figures 1, 2, and 6** are the same object, which has been reconfigured to meet the needs of each solution.

The display shown in **Figure 8**, is from the solution in Figure 1. This display was configured to accept three traces, while the other solutions accept only one. Each display object was sized to better present the data.

Data display within HP VEE is a flexible, easily modifiable way to provide the visual support needed in a complete solution.

The display of data is done automatically with absolutely no coding. Providing this degree of sophistication, concurrent with acquisition, is difficult in a normal data acquisition environment. With HP VEE, it is easy and transparent.

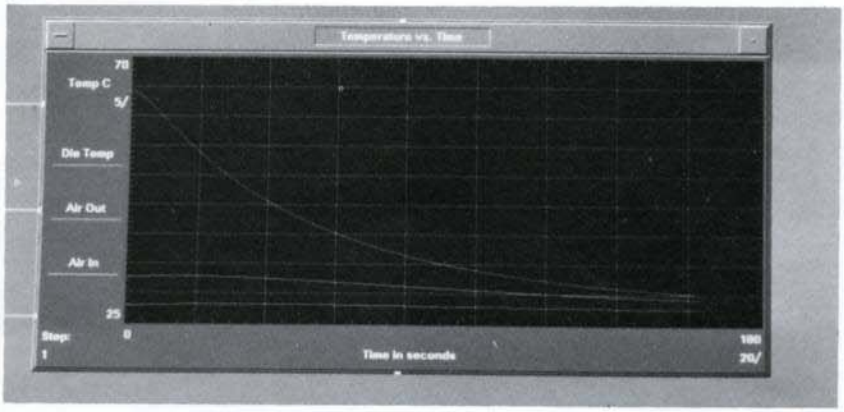


Figure 8. Multiple traces are one of the many display options available.

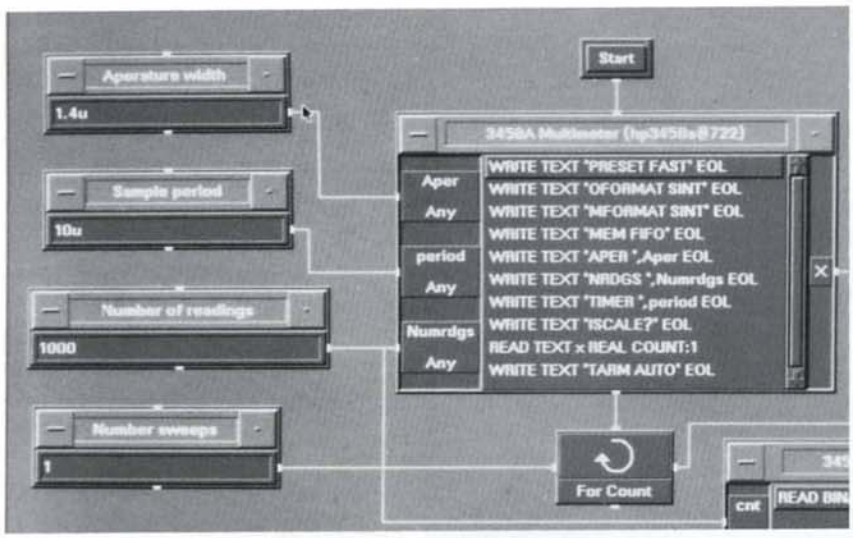


Figure 9. The user interface is implicit in the solution.

Creating User Interfaces

The solution in **Figure 9** shows a direct I/O object which takes user inputs through data objects. This means that an implicit user interface is available. HP VEE-Test also lets you build explicit user interfaces by choosing alternate views of the model, views wherein details of implementation are hidden. Designers of data acquisition solutions can build their own dialog boxes and other user interface objects. This ability is important for data acquisition in production testing, or when someone other than the developer is using the models. Providing this functionality in normal data acquisition environments is time-consuming, to say the least. With HP VEE-Test, it takes only a few minutes.

HP VEE-Test: The Next Generation

HP VEE-Test represents the next step in the human/computer interface. High-level support for I/O, the use of instrument drivers, direct I/O, and I/O transactions, all mean that less time is spent on I/O implementation and more time on getting results. Direct manipulation of visual objects allows you to build solutions quickly and intuitively. Programming skills and long development times will no longer be necessary. Hence, the time and cost of developing complete, integrated data acquisition solutions will decrease dramatically.

Now, *This* is Engineering!

For more information, call your local HP sales office listed in your telephone directory or an HP regional office

Data Subject to Change
Printed in U.S.A. 4/91

5091-1139E