1.2 Using Graphical Programming Throughout the Development Cycle with NI LabVIEW

Bridging Validation and Test with Common Software Technology
Once you move beyond taking interactive measurements, the next step is typically to automate your test measurements.
### Challenges in Automated Measurements

- Faster time-to-market means less time to create applications
- Need to integrate data acquisition from diverse hardware I/O types
- Test systems may involve non-Windows OS or non-PC hardware

However, there are several challenges associated with measurement automation. There is always limited time to bring your products to market, constraining the amount of time that can be spent programming and debugging your measurement applications. You may have to incorporate measurements from many different types of hardware from multiple vendors into the same application. And you also may need to extend your measurement system beyond the limitations of the PC, which usually requires different tools than those used on Windows, along with training on how to use those tools.

However, there are software tools you can use to overcome these challenges, such as NI LabVIEW.
LabVIEW is a highly productive graphical development environment with the performance and flexibility of a programming language, as well as high-level functionality and configuration utilities designed specifically for measurement and automation applications.

In general-purpose programming languages, the code is as much of a concern as the application. You must pay close attention to the syntax (commas, periods, semicolons, square brackets, curly brackets, round brackets, and so on). In contrast, with LabVIEW you use icons to represent functions, and you wire them together to determine the flow of data through your program, similar to creating flowcharts. It has all the breadth and depth of a general-purpose programming language, but it is easy to use, increasing your productivity by decreasing the time required to develop your applications.

You can easily divide measurement and automation applications into three main parts: acquisition, analysis, and presentation of data. LabVIEW provides a seamless way to acquire your data, perform necessary analysis on that data, and present the information in a chosen format. Throughout the seminar, we touch upon each of these three components of a measurement and automation application.

Each program in LabVIEW is called a virtual instrument, or VI. The VI serves as the primary building block of a LabVIEW application, and you can use it to modularize your code for efficient design, clear and concise documentation, and simplified maintenance.
LabVIEW for Automated Test

• Increase productivity with graphical programming
• Automate manual tests and control processes
• Extend systems to Real-Time, FPGA, and PDA targets with high code reuse

In our discussion today, we will discuss how you can be more productive when creating your automated test systems. We will cover reducing development time with LabVIEW programming; automating steps that were previously manual, interactive measurements, and the extension of test systems beyond just Windows with LabVIEW Real-Time, FPGA, and PDA.
First, I would like to start by pointing out that the LabVIEW code you see on the block diagram is compiled directly to machine code, and is supplying your computer’s processor with the same type of code provided by a typical C program.
Let's look at a comparison of LabVIEW and C for a simple data acquisition application. All we are doing is acquiring 100 points from a data acquisition board in a While Loop. In the LabVIEW code, we have also added plotting our acquired data to a graph on our front panel (that is, user interface). We have not added any UI code in the C example, so it is still relatively simple for a C expert. At this level of development, the choice really comes down to personal preference.
To illustrate the value of system abstraction in LabVIEW, we can add more complexity to our earlier example. If we wanted to acquire data simultaneously from a second data acquisition board using a multi-threaded architecture, all we need to do is add a second While Loop with a second data acquisition VI.

As soon as we start adding complexity, the productivity benefits of LabVIEW’s automatic abstraction begin to emerge:

- Analysis
- Parallel loop
- Hardware timing

LabVIEW abstracts much of the system complexity by automatically handling memory and multithreading issues in the background. Although the user can have visibility into these, for the majority of users the automatic abstraction of LabVIEW makes development significantly faster and easier. At the same time, the user does not sacrifice performance for this productivity gain—as a compiled language, LabVIEW still provides execution speeds and performance needed for today’s applications.
In comparison, the C programmer would have to add the code for the additional data acquisition and program the necessary code to handle the parallel, now multi-threaded, nature of the application. The lines of C code begin to drastically increase, and the C programmer also must create each individual measurement function—all of which are included with LabVIEW, which is designed specifically for engineers and scientists.
More sophistication can be added by incorporating hardware timing. In LabVIEW, you can use the Timed Loop structure—just point and click to set OS priorities, delays, loop rates, and so on.
To add this same functionality into our equivalent C program would far exceed the bounds of a single slide.
Demo – Acquire, Analyze, Present in LabVIEW

• Control instruments (arbs, scopes, DMMs, and so on.)
• Develop using intuitive and configurable interfaces

This demonstration will show how to use LabVIEW to quickly acquire data using NI modular instrumentation using LabVIEW Express VIs. Next, we will analyze the data by performing an FFT on the acquired signal. Finally we will save our acquired and processed signal to an ASCII file that can be opened in Excel or Word.
We have talked about how LabVIEW can help you develop test applications faster and easier. I also want to mention the different options you have for targeting non-Windows platforms, which enable you to design embedded and hand-held applications.

LabVIEW continues to bring powerful measurement application design to all engineers through graphical programming. By expanding throughout the range of development platforms and deployment targets, LabVIEW continues to provide solutions for all types of test and measurement challenges, from manufacturing test to distributed monitoring systems.
LabVIEW Real-Time Module

- Develop robust, reliable applications faster with graphical programming
- Implement precise, deterministic performance and validate performance
- Eliminate time spent integrating diverse I/O

In 1999 National Instruments introduced the LabVIEW Real-Time Module and the first real-time hardware targets to run embedded, deterministic LabVIEW code. With LabVIEW Real-Time, you develop your LabVIEW Real-Time application on a host computer with graphical programming and then download the application to run on an independent hardware target with a real-time operating system for deterministic execution and the highest reliability. With LabVIEW 7 Express, NI introduced new software features such as automatic code generation to make it much easier to accomplish difficult real-time development tasks in less time.

The increased reliability of a real-time platform is critical for expensive-to-recreate tests such as those that destroy prototypes, execute for long periods of time, or require the use of expensive resources. Real-time operating systems impart a higher level of reliability because they are streamlined to use fewer resources and eliminate the fragilities of standard operation systems. LabVIEW Real-Time delivers the benefits of real-time operating systems without requiring you to know the low-level details of real-time development. Instead, use the familiar graphical development environment of LabVIEW to build your application and then download it to a dedicated real-time target to realize the benefits of the real-time platform.
Another major development increasing the productivity of engineers is the LabVIEW FPGA Module, a powerful software tool that extends the LabVIEW graphical development environment to Field-Programmable Gate Arrays (FPGAs). Now, with the LabVIEW FPGA Module all test and measurement engineers can take advantage of the advanced timing and triggering of FPGAs without having to learn VHDL.

With the LabVIEW FPGA Module, you develop your FPGA application on a host computer running Windows, then compile and download it to the FPGA on the NI PXI-7831R reconfigurable I/O device. Because LabVIEW FPGA executes logic in hardware, applications have direct access to the eight analog input, eight analog output, and 96 digital input/output signals on the hardware. You can configure the I/O lines to operate independently or synchronize them for more precise measurement. In addition, you can implement integer-based calculations and digital processing within the FPGA to create user-defined I/O such as counter operations, PWM channels, custom digital protocols, and unique triggering.

Applications:
- Pulse width modulation and custom digital protocol I/O
- HIL simulation and rapid prototyping
- Flexible motion control
- Sensor simulation
- Custom counter and encoder interface
- Custom triggering and synchronization of signals
- Discrete control
LabVIEW PDA Module

- Graphical development of custom PALM and Pocket PC applications
- Handheld data acquisition and remote monitoring
- Communication with external devices through 802.11b, IrDA, and RS232 Serial protocols

Personal digital assistants (PDAs) have gained widespread use and satisfy an increasing demand for reducing equipment size while increasing system mobility and modularity. Now you can take advantage of the benefits of this technology by using National Instruments LabVIEW. By adding the LabVIEW PDA Module to LabVIEW development systems, you can run VIs on Microsoft Pocket PC and Palm OS PDA devices. With the new LabVIEW PDA Module engineers and scientists can build many applications including field test systems, remote control and monitoring systems, and portable data acquisition systems.
Summary

- Graphical programming speeds development and increases productivity through interactive configuration
- LabVIEW offers tight integration of real-world I/O, measurement analysis, and data presentation
- Deploy development from the desktop to handheld and embedded devices with no additional code