LabWindows/CVI Programming for Beginners, Shahid F. Khalid, Prentice Hall PTR, 2000, 0130165123, 9780130165121, 651 pages. LabWindows/CVI 5.0 is the #1 system for building Windows-based virtual instrumentation with ANSI C -- and this hands-on, project-oriented guide is the fastest way to get results with LabWindows/CVI! Engineering professionals with C experience will master all they need to know to build programs that control instruments and data acquisition hardware -- while still taking advantage of sophisticated, easy-to-use Windows interfaces. Shahid F. Khalid presents never-before-published LabWindows/CVI tips and tricks, plus coverage of every key LabWindows/CVI skill you'll need. Khalid offers step-by-step coverage of the Code Builder, introducing code analysis, event generation, code generation and enhancements, statistical analysis functions, and more. Learn how to create effective graphical user interfaces; work with key libraries such as Advanced Analysis, VXI Utilities, GPIB, RS-232, VISA, TCP, ANSI C and ActiveX Automation; debug code; create standalone executables; and more. Each chapter is organized for maximum clarity and convenience, and the book contains handy appendices covering LabWindows CVI's key formatting and scanning functions..

DOWNLOAD HERE

LabVIEW for LEGO Mindstorms NXT, Michael Gasperi, 2008, LEGO toys, 376 pages.

LabVIEW for automotive, telecommunications, semiconductor, biomedical, and other applications Hall T. Martin, Meg L. Martin, Hall T. Martin, Meg L. Martin, 2000, , 247 pages. Most test and measurement books focus on theory. This one is radically different: it brings together dozens of the best real-world LabVIEW applications in leading test and ....

LabVIEW for Electric Circuits, Machines, Drives, and Laboratories, Nesimi Ertugrul, 2002, Technology & Engineering, 436 pages. This interactive, hands-on guide to AC/DC circuits, machines, devices, and power electronics uses custom-written LabVIEW Virtual Instruments to illuminate every key concept ....

Black Hills, Nora Roberts, Jul 1, 2009, , 544 pages. Lil Chance fell in love with Cooper Sullivan pretty much the first time she saw him, an awkward teenager staying with his grandparents on their cattle ranch in South Dakota ....

Encyclopedia for Windows the one book that belongs beside your personal computer, Eddy Group, 1993, Computers, 1012 pages. An alphabetical list of every Windows command, feature, capability, and concept offers concise definitions and how-to explanations.

LabVIEW 8 Student Edition, Robert H. Bishop, 2007, , 619 pages. The defacto industry standard for test, measurement, and automation software solutions. LabVIEW 8 delivers the graphical programming capabilities that allow users to design ....

Inside SPICE, Ron M. Kielkowski, 1998, , 200 pages. Generate faster, more accurate SPICE simulations! Make your SPICE simulations faster, more accurate - and avoid nonconvergence using
the breakthrough methods packed into the ....

Ridge Waveguides and Passive Microwave Components, J. Helszajn, 2000, Technology & Engineering, 327 pages. The ridge waveguide, which is a rectangular waveguide with one or more metal inserts (ridges), is an important transmission line in microwave engineering, now widely used in ....

Languages And Machines: An Introduction To The Theory Of Computer Science, 3/E , Sudkamp, Sep 1, 2007, , 672 pages. .

Framework Molecular Model Student Kit , George Brumlik, 1965, , . This accurate, reasonably priced molecular model set enables users to represent all atoms having up to 12 electrons in their valence shells -- including those which cannot be ....

Prentice Hall Molecular Model Set for General and Organic Chemistry , Prentice Hall, Pearson, 1997, , . Designed for general chemistry courses that consider a lot of organic examples, or for students who plan to continue in organic chemistry. The Prentice Hall molecular model set ....


Head First Statistics , Dawn Griffiths, Aug 26, 2008, Mathematics, 677 pages. "Head First Statistics" brings a typically difficult subject to life, teaching readers everything they want and need to know about statistics through engaging, interactive, and ....

LabWindows/CVI is the #1 system for building Windows-based virtual instrumentation with ANSI C&&8212;and this hands-on, project-oriented guide is the fastest way to get results with LabWindows/CVI! You'll master all you need to know to build programs that control instruments and data acquisition hardware&&8212;while still taking advantage of an easy-to-use user interface editor for building Windows applications. Shahid F. Khalid presents never-before-published LabWindows/CVI tips and tricks&&8212;plus coverage of every key LabWindows/CVI skill you'll need, including:

Each chapter is organized for maximum clarity and convenience, and you'll find handy appendices explaining the features of LabWindows/CVI environment, formatting and scanning functions, and tutorial on two demo programs. If you're a LabWindows/CVI novice, you'll learn fast&&8212;and once you do, LabWindows/CVI Programming for Beginnerswill serve you well as a reference for years to come.

SHAHID F. KHALID is currently a software engineer at Boeing, Canoga Park, California, where he works on testing the embedded software for the rocket Engine Control Unit on the Delta IV program. In addition, he has created LabWindows/CVI applications for laser control and has written many test software applications for controlling test instrumentation. Mr. Khalid has more than 30 years of experience in software engineering.

Excellent technical book with plenty of sample code to demonstrate the diversity, power and simplicity of this C programming environment. The clarity of this book often surpassed the vendor-supplied documentation, making it possible for me to develop complex software applications within weeks. I wish I could find such good tutorials for the Borland OWL and Microsoft MFC rapid prototyping environments.

Each section uses well thought out examples to demonstrate the features and libraries available to the programmer. The topic areas covered in this book ensure that the user can quickly generate code that will produce professional looking solutions for any application. The GPIB and RS232 tutorials are superb, providing essential background information in both these crucial areas.
This is a really good book for someone who is starting out using Labwindows CVI. I had a little knowledge of C but the book really did a good job of filling in the blanks and refreshing my memory. After only reading the first few chapters I was programming great user interfaces over the GPIB. I have already recommended this book to a number of fellow students.

Take virtual instrumentation to the next level with high-level programming. High-level programming with LabWindows/CVI Live data display via Internet or intranet sources Programmatic creation and control of GUIs Data acquisition and VXI device communication Graph control, table control, function panels, instrument drivers, and Open GL Unleash the true power of LabWindows/CVI when you employ the rich features of this programming environment. In this follow-up to his LabWindows CVI Programming for Beginners, Shahid F. Khalid presents the sophisticated techniques that allow experienced users to make the most of this virtual instrumentation powerhouse. The flexibility of LabWindows/CVI software means that you can build virtual instrumentation using Microsoft Visual Basic and Visual C++ as well as ANSI C. Advanced Topics in LabWindows/CVI focuses on the use of C in an open software architecture. It is a project-oriented guide that will teach you to build applications using the more complex features of this programming environment. Applications include: Live data acquisition via Internet or intranet sources using Data Socket technology GUI controls created and manipulated in real time Advanced features of graph and table controls 3-D data plotting with Open GL Communications with VXI devices using VISA Creating and using function panels and instrument drivers The material is organized to present information with maximum clarity, keeping the reader in mind. For convenience, each chapter concludes with an explanation of the purpose and prototype of the library functions under discussion. Advanced Topics in LabWindows/CVI will give students and working professionals the tools to build and automate sophisticated virtual instrumentation for a world of applications.

arguments explained array callback function char Child_Handle color column command button configuration controlID counter Create cursor DAQ board data types data value DataSocket object DataSocket Server default dialog box dialog window displayed double error codes eventData2 explained in Table function at line Function Input Function Listing function panel function prototype function tree GPIB graph control header file HRESULT instrument driver int CVICALLBACK int event int eventDatal int panel int panelHandle int status integer panel handle label LabWindows/CVI legend control library function loaded in memory menu item metafont mode Name Type Description National Instruments negative value represents OpenGL control Output Name Type panel handle loaded plot prototype is shown pulse Register ring control selected SetCtrlAttribute GUI_Handle shown in Figure signal source code specified string table control taskID tion Type Description Input value represents error VI_NULL VISA VMEbus void callbackData VXIbus waveform

Shahid F. Khalid has over 30 years of experience in software engineering. At Boeing he is presently working on the design and implementation of the Next Generation Automatic Test Equipment using LabWindows/CVI. In the past he has used LabWindows/CVI to test embedded software for the engine control unit on Delta IV rockets, to create applications for laser beam alignment, and to control and diagnose automatic test equipment. He is the author of "LabWindows/CVI Programming for Beginners" (0-13-016512-3), also published by Pearson PTR.

If I could pass on some advice that might be more helpful to you in learning CVI: pretend those books do not exist for now. Instead, I think that for learning CVI, there is nothing better than looking at other people's CVI project-level code and really understanding it, inside and out, and then making your own modifications to it to enforce what you just learned from it. There are several places to start looking for project-level CVI code, which I will list in (what I think) are the highest-to-lowest priority of your training.

The first place I recommend is the examples that ship with CVI (located in the "samples" directory under the CVI installation directory). I say this because these are supported by the NI developers and support crew. In fact, back in 1995 when I was learning CVI 3.1, I dove into the "Icon Editor" application, for several reasons: it was a *complete* application (as opposed to the other, mostly point-solution "academic" examples under the samples directory), it showed how to segment your
code into multiple source and header files, and it showed many advanced user interface features (menus, cut-and-paste). There is now an "example finder" to help locate examples that "fit" to what you are currently trying to accomplish. I also use the freeware "grep-like" tool called "Agent Ransack" (URL is below) to hunt down specific CVI functions and identifiers (such as certain EVENTS and how these examples use them).

The second place I recommend is the other "support" locations on the NI web site. There are knowledgebase articles, tutorials, submitted examples, community examples, and also extra "academic" examples from NI engineers that aren't the same as the ones that ship with CVI. There are also whitepapers scattered in there as well, and a few that have hidden sample code that is embedded at the bottom of the article. You really have to hunt for this stuff, and it would be great if the NI folks could try to index this a bit better somehow. I sometimes even resort to using Google's search engine to find them on the NI web site, because the NI search engine may not find them.

The third place I recommend are user-based web sites created by power-users of CVI who like to share their code. Two that I found long ago that remain at the top of the list (in my opinion). One is Guillaume Dargaud's site, where he has lots of CVI examples and freeware. The other is Philippe Baucour's "Rebel" CVI site (but you will need to use an online translator if you don't speak French, though).

Finally, you might find it interesting to know that there is a tremendous repository of C source code scattered all around the Internet, mostly as part of free and open-source software projects, many of which are supported on sites such as SourceForge.net, and many other locations. Many of these places are filled with people (in forums or mailing list servers) who live in the C programming world on a full-time basis, and can help you through some of the advanced techniques. Some of these projects might be targeted to Windows, some use cross-platform techniques, so they have a chance of being applied to your CVI efforts. There is also an entire universe of programming "libraries" and capabilities that you can use to augment your CVI projects as well. All of this because of the open-source revolution, riding on the back of GNU/Linux and the community and corporate entities that are backing it all. Examples that come to mind are Cygwin and MinGW. If you only have the time to just look up these two items on Wikipedia, you will see that this is just the tip of the iceberg of the possibilities that are available to you as a C and CVI programmer on Windows.

To whom it may concern: My alias is also my nickname, I've had it since I was a (very) skinny basketball-playing teen. OK, so I've got a 38 inch waist now, but my hometown friends haven't shaken that appellation for me. I trust that you will someday be OK with that alias, as I have been with that nickname.

I forgot to mention that the Advanced Topics in LabWindows/CVI book by Shahid F. Khalid is available on O'Reilly's Safari Online. Portions of the book are directly available for preview at no cost. Plus, you could read the entire book if you wanted to try out a "Free Trial" subscription to the Safari web site. There are many, many programming books on this web site, by the way. It might be worth the cost to subscribe if you and your co-workers could build an argument to access it regularly.

LabWindows/CVI is an integrated American National Standards Institute (ANSI) C environment (C is a computer programming language) developed by National Instruments Corporation and designed primarily for engineers and scientists creating virtual instrumentation applications. Virtual instrumentation refers to the combination of hardware and software elements that provides you, the user, with complete flexibility in designing and controlling the elements of stand-alone instruments from your computer system. You have the choice of designing the instrument's functionality. LabWindows/CVI helps you leverage the power of the computer to create flexible, reusable, and inexpensive measurement applications that outperform traditional test and measurement methods.

LabWindows/CVI is a programming environment that has been widely adopted throughout industry, academia, and research labs as the standard for data acquisition and instrument control software. It is a powerful and flexible instrumentation software system that embraces the traditional
programming methodologies and enables you to create structured code, and it features an easy-to-use programming environment.

LabWindows/CVI includes all the tools necessary for data analysis and presentation of the results on the Graphical User Interface (GUI, pronounced "gooee") on your computer screen. It comes with a complete set of integrated input/output (I/O) libraries, analysis routines, and user interface tools that aid you in rapid application development, reducing the time that would be required by other conventional programming environments.

The integrated environment consists of code generation, prototyping tools, and libraries. There are input/output libraries, user interface libraries, and statistical and mathematical analysis libraries. LabWindows/CVI supports a comprehensive Advanced Analysis Library for time/frequency analysis, curve-fitting, digital filters, integration and differentiation functions, statistical functions, linear equations solutions, and many more. This releases you from seeking an outside application package to perform the analysis. Also included are signal processing, instrument driver creation, and Activex support, multithreading, DataSocket, OpenGL libraries, to name a few.

LabWindows/CVI’s forte lies in building virtual instrumentation systems with General Purpose Interface Bus (GPIB), VME (Versa-Modular Eurocard) extensions for Instrumentation (VXI), PCI (Peripheral Component Interconnect) extensions for instrumentation (PXI), serial interface communication (RS-232), Transmission Control Protocol/Internet Protocol (TCP/IP) based devices with plug-in data acquisition (DAQ) boards without spending too much effort to create the applications. It combines an interactive, easy-to-use development approach with the programming power and flexibility of compiled ANSI C code.

The full potential of LabWindows/CVI is used to automate test systems, bench top experiments, DAQ monitoring projects, verification tests and measurements, process monitoring, and controlling systems. This integrated development environment gives full credibility to the National Instruments® slogan: "The Software is the Instrument" by enabling the user to create virtual instruments on the personal computer and to communicate with the real instruments via the communication interfaces.

The LabWindows/CVI environment supports open software architecture enabling you to reuse existing programs within its environment. If you are programming in C using your preferred environment, LabWindows/CVI complements your existing efforts and streamlines your future development. You can incorporate standard ANSI C source code, object files, and dynamic link libraries (DLLs) within LabWindows/CVI. You also have the flexibility to use the instrumentation libraries from LabWindows/CVI within the C/C++ compilers such as Microsoft Visual C++, Borland C++, Borland C++ Builder, Symantec C, or WATCOM C/C++ with which you may be more familiar.

Why write a book on CVI? The answer is simple. There is not a single book on CVI in the market at the time of this writing. CVI does come with an excellent set of manuals (both bound and on-line), which enables you to get started. Any packaged software in the market always has a couple of books written to give you a different approach from the manuals.

There are about a dozen good books written on LabVIEW®, a graphical programming language, called G, in which you can program in block diagram notation. LabVIEW was invented by National Instruments Corporation®, which is also the creator of CVI. Weeklong courses conducted by National Instruments Corporation teach both these programming tools. I have personally taken the National Instruments sponsored courses for both CVI and LabVIEW, which give the student very useful hands-on experience. During these courses, the students often asked the instructor for the name of some good books on the subject from which they could learn more. The instructor could always recommend a couple of good LabVIEW books but none for CVI.

This is the first book on CVI, and its aim is to teach you how to get started quickly and to create CVI projects. I am assuming you are unfamiliar with using CVI. If you are familiar with CVI, this book can serve as an adequate reference to refresh your memory and to glance over some of the examples.
on a particular topic.

The purpose of LabWindows/CVI Programming for Beginners is to serve as a tutorial to help you, a willing CVI learner, get started quickly with CVI to develop your instrumentation and analysis applications. Almost every aspect of the CVI programming environment has been introduced or pointed out in the CVI manuals. CVI contains many capabilities, and this book does not provide a comprehensive guide. The goal of the book, however, is to give you enough information to provide a foundation on which to build.

This book uses a systematic approach to teaching CVI. Every facet of creating the CVI projects is explained in detail. The chapters are arranged in an order that facilitates learning CVI. Detailed examples are included where necessary. The prototypes of the CVI library functions are explained when introduced for the first time in the project. These prototypes may look very similar to ones given in the manuals. They are included in this book for your convenience, sparing you the task of finding them among the alphabetical listings of innumerable functions spread over half a dozen CVI manuals or searching in Online Help.

http://eduln.org/3886.pdf
http://eduln.org/1322.pdf
http://eduln.org/9033.pdf
http://eduln.org/7442.pdf