Hands-On Introduction to LabVIEW™ for Scientists and Engineers
Hands-On Introduction to
LabVIEW™
for Scientists and Engineers

Fourth Edition

John Essick

Reed College

New York   Oxford
OXFORD UNIVERSITY PRESS
To my wife, Katie
# Contents

Preface                                      iii  
About the Author                            xviii 

1. **LABVIEW PROGRAM DEVELOPMENT**       1  
   1.1 LabVIEW Programming Environment   1  
   1.2 Blank VI                          2  
   1.3 Front-Panel Editing               3  
   1.4 Block-Diagram Editing             11 
   1.5 Program Execution                 22 
   1.6 Pop-Up Menu and Data-Type Representation 24  
   1.7 Program Storage                   27 
   1.8 Quick Drop                        29  
      Do It Yourself                     32 
      Use It!                           34  
      Problems                          36 

2. **THE WHILE LOOP AND WAVEFORM CHART**  39  
   2.1 Programming Structures and Graphing Modes 39  
   2.2 While Loop Basics                  40 
   2.3 Sine-Wave Plot Using a While Loop and Waveform Chart 42  
   2.4 LabVIEW Help Window                48 
   2.5 Front-Panel Editing                 50 
   2.6 Waveform Chart Pop-Up Menu         53 
   2.7 Finishing the Program              56 
   2.8 Program Execution                  57 
   2.9 Program Improvements               59 
   2.10 Data Types and Automatic Creation Feature 70  
      Do It Yourself                     74 
      Use It!                           75  
      Problems                          77 

3. **THE FOR LOOP AND WAVEFORM GRAPH**    83  
   3.1 For Loop Basics                    83 
   3.2 Sine-Wave Plot Using a For Loop and Waveform Graph 84
3.3 Waveform Graph 85  
3.4 Owned and Free Labels 86  
3.5 Creation of Sine Wave Using a For Loop 87  
3.6 Cloning Block-Diagram Icons 89  
3.7 Auto-Indexing Feature 91  
3.8 Running the VI 94  
3.9 X-Axis Calibration of the Waveform Graph 94  
3.10 Sine-Wave Plot Using a While Loop and Waveform Graph 100  
3.11 Front-Panel Array Indicator 103  
3.12 Debugging with the Probe Watch Window and Error List 108  
Do It Yourself 115  
Use It! 117  
Problems 119

4. THE MATHSCRIPT NODE AND XY GRAPH 127  
4.1 MathScript Node Basics 127  
4.2 Quick MathScript Node Example: Sine-Wave Plot 130  
4.3 Waveform Simulator Using a MathScript Node and XY Graph 137  
4.4 Creating an XY Cluster 142  
4.5 Running the VI 143  
4.6 LabVIEW MathScript Window 144  
4.7 Adding Shape Options Using an Enumerated Type Control 148  
4.8 Finishing the Block Diagram 151  
4.9 Running the VI 155  
4.10 Control and Indicator Clusters 156  
4.11 Creating an Icon Using the Icon Editor 163  
4.12 Icon Design 163  
4.13 Connector Assignment 168  
Do It Yourself 172  
Use It! 173  
Problems 174

5. INTRODUCTION TO DATA ACQUISITION DEVICES USING MAX 179  
5.1 Data Acquisition Hardware 179  
5.2 Measurement & Automation Explorer (MAX) 181  
5.3 Analog Input Modes 185  
5.4 Range and Resolution 187  
5.5 Sampling Frequency and the Aliasing Effect 187  
5.6 Analog Input Operation Using MAX 189  
5.7 Analog Output 193  
5.8 Analog Output Operation Using MAX 193  
5.9 Digital Input/Output 197  
5.10 Digital Input/Output Operation Using Max 198
CONTENTS

Do It Yourself 200
Use It! 202
Problems 204

6. DATA ACQUISITION USING DAQ ASSISTANT 206
   6.1 Data Acquisition VIs 206
   6.2 Simple Analog Input Operation on a DC Voltage 207
   6.3 Digital Oscilloscope 218
   6.4 DC Voltage Storage 228
   6.5 Hardware-Timed Waveform Generator 234
   6.6 Placing a Custom-Made VI on a Block Diagram 237
   6.7 Completing and Executing Waveform Generator (Express) 240
   Do It Yourself 242
   Use It! 243
   Problems 246

7. DATA FILES AND CHARACTER STRINGS 254
   7.1 ASCII Text and Binary Data Files 254
   7.2 Storing Data in a Spreadsheet-Formatted File 256
   7.3 Storing a One-Dimensional Data Array 257
   7.4 Transpose Option 260
   7.5 Storing a Two-Dimensional Data Array 262
   7.6 Controlling the Format of Stored Data 266
   7.7 The Path Constant and Platform Portability 267
   7.8 Fundamental File I/O VIs 269
   7.9 Adding Text Labels to a Spreadsheet File 274
   7.10 Backslash Codes 277
   Do It Yourself 279
   Use It! 282
   Problems 284

8. SHIFT REGISTERS 292
   8.1 Shift Register Basics 292
   8.2 Quick Shift Register Example: Integer Sum 295
   8.3 Noise and Signal Averaging 299
   8.4 Noisy Sine VI 301
   8.5 Moving Average of Four Traces 307
   8.6 Modularity and Automatic SubVI Creation 316
   8.7 Moving Average of Arbitrary Number of Traces 323
   Do It Yourself 337
   Use It! 337
   Problems 340
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.</td>
<td>THE CASE STRUCTURE</td>
<td>348</td>
</tr>
<tr>
<td>9.1</td>
<td>Case Structure Basics</td>
<td>348</td>
</tr>
<tr>
<td>9.2</td>
<td>Quick Case Structure Example: Runtime</td>
<td>351</td>
</tr>
<tr>
<td>9.3</td>
<td>State Machine Architecture: Guessing Game</td>
<td>363</td>
</tr>
<tr>
<td>9.4</td>
<td>State Machine Architecture: Express VI-Based</td>
<td>377</td>
</tr>
<tr>
<td></td>
<td>Digital Oscilloscope</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Do It Yourself</td>
<td>386</td>
</tr>
<tr>
<td></td>
<td>Use It!</td>
<td>387</td>
</tr>
<tr>
<td></td>
<td>Problems</td>
<td>389</td>
</tr>
<tr>
<td>10.</td>
<td>DATA DEPENDENCY AND THE SEQUENCE STRUCTURE</td>
<td>396</td>
</tr>
<tr>
<td>10.1</td>
<td>Data Dependency and Sequence Structure Basics</td>
<td>396</td>
</tr>
<tr>
<td>10.2</td>
<td>Event Timer Using a Sequence Structure</td>
<td>400</td>
</tr>
<tr>
<td>10.3</td>
<td>Event Timer Using Data Dependency</td>
<td>407</td>
</tr>
<tr>
<td>10.4</td>
<td>Highlight Execution</td>
<td>411</td>
</tr>
<tr>
<td></td>
<td>Do It Yourself</td>
<td>413</td>
</tr>
<tr>
<td></td>
<td>Use It!</td>
<td>414</td>
</tr>
<tr>
<td></td>
<td>Problems</td>
<td>417</td>
</tr>
<tr>
<td>11.</td>
<td>ANALYSIS VIs: CURVE FITTING</td>
<td>425</td>
</tr>
<tr>
<td>11.1</td>
<td>Thermistor Resistance-Temperature Data File</td>
<td>425</td>
</tr>
<tr>
<td>11.2</td>
<td>Temperature Measurement Using Thermistors</td>
<td>428</td>
</tr>
<tr>
<td>11.3</td>
<td>The Linear Least-Squares Method</td>
<td>431</td>
</tr>
<tr>
<td>11.4</td>
<td>Inputting Data to a VI Using a Front-Panel Array Control</td>
<td>433</td>
</tr>
<tr>
<td>11.5</td>
<td>Inputting Data to a VI by Reading from a Computer File</td>
<td>436</td>
</tr>
<tr>
<td>11.6</td>
<td>Slicing Up a Multidimensional Array</td>
<td>439</td>
</tr>
<tr>
<td>11.7</td>
<td>Running the VI</td>
<td>443</td>
</tr>
<tr>
<td>11.8</td>
<td>Curve Fitting Using the Linear Least-Squares Method</td>
<td>444</td>
</tr>
<tr>
<td>11.9</td>
<td>Residual Plot</td>
<td>452</td>
</tr>
<tr>
<td>11.10</td>
<td>Curve Fitting Using the Nonlinear Least-Squares Method</td>
<td>454</td>
</tr>
<tr>
<td></td>
<td>Do It Yourself</td>
<td>457</td>
</tr>
<tr>
<td></td>
<td>Use It!</td>
<td>460</td>
</tr>
<tr>
<td></td>
<td>Problems</td>
<td>462</td>
</tr>
<tr>
<td>12.</td>
<td>ANALYSIS VIs: FAST FOURIER TRANSFORM</td>
<td>470</td>
</tr>
<tr>
<td>12.1</td>
<td>Quick Fast Fourier Transform Example</td>
<td>470</td>
</tr>
<tr>
<td>12.2</td>
<td>The Fourier Transform</td>
<td>480</td>
</tr>
<tr>
<td>12.3</td>
<td>Discrete Sampling and the Nyquist Frequency</td>
<td>481</td>
</tr>
<tr>
<td>12.4</td>
<td>The Discrete Fourier Transform</td>
<td>482</td>
</tr>
<tr>
<td>12.5</td>
<td>The Fast Fourier Transform</td>
<td>484</td>
</tr>
<tr>
<td>12.6</td>
<td>Frequency Calculator VI</td>
<td>485</td>
</tr>
</tbody>
</table>
# CONTENTS

12.7 FFT of Sinusoids 488  
12.8 Applying the FFT to Various Sinusoidal Inputs 491  
12.9 Magnitude of the Complex-Amplitude 494  
12.10 Observing Leakage 500  
12.11 Windowing 503  
12.12 Estimating Frequency and Amplitude 509  
12.13 Aliasing 513  
Do It Yourself 514  
Use It! 515  
Problems 519  

13. DATA ACQUISITION AND GENERATION USING DAQmx VIs 525  
13.1 DAQmx VI Basics 525  
13.2 Simple Analog Input Operation on a DC Voltage 527  
13.3 Digital Oscilloscope 533  
13.4 Express VI Automatic Code Generation 540  
13.5 Limitations of Express VIs 541  
13.6 Improving Digital Oscilloscope Using State Machine Architecture 543  
13.7 Analog Output Operations 556  
13.8 Waveform Generator 557  
Do It Yourself 560  
Use It! 561  
Problems 565  

14. CONTROL OF STAND-ALONE INSTRUMENTS 574  
14.1 Instrument Control Using VISA VIs 574  
14.2 The VISA Session 575  
14.3 The IEEE 488.2 Standard 579  
14.4 Common Commands 579  
14.5 Status Reporting 580  
14.6 Device-Specific Commands 584  
14.7 Specific Hardware Used in This Chapter 586  
14.8 Measurement & Automation Explorer (MAX) 587  
14.9 Simple VISA-Based Query Operation 594  
14.10 Message Termination 598  
14.11 Getting and Setting Communication Properties Using a Property Node 599  
14.12 Performing a Measurement over the Interface Bus 603  
14.13 Synchronization Methods 608  
14.14 Measurement VI Based on the Serial Poll Method 616  
14.15 Measurement VI Based on the Service Request Method 622  
14.16 Creating an Instrument Driver 628
## Contents

14.17 Using the Instrument Driver to Write an Application Program 642  
  Do It Yourself 647  
  Use It! 648  
  Problems 651  

### APPENDIX A. FORMULA NODE PROGRAMMING FOR CHAPTER 4 653  
A.1 Formula Node Basics 653  
A.2 Quick Formula Node Example: Sine-Wave Plot (Section 4.2) 654  
A.3 Formula Node-Based Waveform Simulator (Sections 4.3–4.4) 658  
A.4 Formula Node-Based Waveform Simulator (Section 4.8) 659  
A.5 Formula Node-Based Waveform Simulator (Section 4.10) 660  

### APPENDIX B. MATHEMATICS OF LEAKAGE AND WINDOWING 661  
B.1 Analytic Description of Leakage 661  
B.2 Description of Leakage Using the Convolution Theorem 665  

### APPENDIX C. PID TEMPERATURE CONTROL PROJECT 670  
C.1 Project Description 670  
C.2 Voltage-Controlled Bidirectional Current Driver for Thermoelectric Device 670  
C.3 PID Temperature Control Algorithm 672  
C.4 PID Temperature Control System 675  
C.5 Construction of Temperature Control System 676  

### INDEX 684
Preface

*Hands-On Introduction to LabVIEW for Scientists and Engineers* provides a learn-by-doing approach to acquiring the computer-based skills used daily in experimental work. This book is not a manual-like presentation of LabVIEW. Rather, *Hands-On Introduction to LabVIEW* leads its readers to mastery of LabVIEW through the process of using this powerful laboratory tool to carry out interesting and relevant projects. Readers, who are assumed to have no prior computer programming or LabVIEW background, begin writing meaningful programs in the first few pages.

*Hands-On Introduction to LabVIEW* can be used as a text in an instructional lab course or for self-study by individual researchers. The book is designed for flexible use so that readers can easily choose the desired depth of coverage. The first six chapters, which form the foundation appropriate for all readers, focus on the fundamentals of LabVIEW programming as well as the basics of computer-based experimentation using a National Instruments data acquisition (DAQ) device. These opening chapters can be used as the basis of a three- or four-week introduction to LabVIEW-based data acquisition. Subsequent chapters have been written as independently as possible so that an instructor or self-learner can fill out their course of study as desired. Those who work through most of the text’s chapters will attain an intermediate skill level in computer-based data acquisition and analysis.

The progression of topics in *Hands-On Introduction to LabVIEW* is as follows:

**Chapters 1–4:** Fundamentals of the LabVIEW Graphical Programming Language. Central features of LabVIEW including its programming environment, control loop structures, graphing modes, mathematical functions, and text-based MathScript (and Formula Node) commands are learned in the course of writing digitized waveform simulation programs.

**Chapter 5:** Introduction to Data Acquisition Devices Using MAX. Features of National Instruments DAQ devices are presented, along with concepts of digitized data such as resolution, sampling frequency, and aliasing. Then, using the Measurement & Automation Explorer (MAX), readers interactively control the full functionality (analog-to-digital, digital-to-analog, digital input/output, and pulse counting) of a National Instruments DAQ device.

**Chapter 6:** Data Acquisition Using DAQ Assistant. Using the high-level DAQ Assistant Express VI, readers write LabVIEW programs that execute
Preface

analog-to-digital, digital-to-analog, and digital input/output tasks on a National Instruments DAQ device. Computer-based instruments constructed include a DC voltmeter, digital oscilloscope, DC voltage source, waveform generator, and blinking LED array.

Chapters 7–10: More LabVIEW Programming Fundamentals. Implementation of data file input/output, local memory, and conditional branching in LabVIEW is investigated while writing several useful programs (e.g., spreadsheet data storage, moving averager) and learning the powerful state machine program architecture. Additionally, LabVIEW's control flow approach to computer programming is studied.

Chapters 11 and 12: Data Analysis. Proper use of LabVIEW's curve fitting and fast Fourier transform (FFT) functions is investigated. Using Express VIs to control a DAQ device, two computer-based instruments—a digital thermometer and a spectrum analyzer—are constructed.

Chapter 13: Data Acquisition Using DAQmx. Programs are written to carry out analog-to-digital, digital-to-analog, and digital counter tasks on a DAQ device using the conventions of DAQmx. This lower-level approach (in comparison to the high-level Express VIs) allows utilization of the full available range of DAQ device features. A DC voltmeter, DC voltage source, waveform generator, and frequency meter are constructed, as well as a sophisticated digital oscilloscope based on the state machine architecture.

Chapter 14: Control of Stand-Alone Instruments. Using LabVIEW's VISA communication driver, control of a stand-alone instrument over the General Purpose Interface Bus (GPIB) as well as the Universal Serial Bus (USB) is studied. A Keysight/Agilent 34410A Multimeter is used to demonstrate the central concepts of interface bus communication between a PC and stand-alone instrument.

Appendix A: Formula Node Supplement. After a brief introduction to the Formula Node, instructions are given for carrying out Chapter 4 exercises using the Formula Node (rather than the MathScript Node).

Appendix B: FFT Supplement. A mathematical description of the leakage and windowing effects associated with fast Fourier transform analysis is presented.

Appendix C: Temperature Control Project. The LabVIEW skills acquired throughout the book are used to construct a Proportional-Integral-Derivative (PID) temperature control system. A design for the hardware required for this project is included.

Key features of Hands-On Introduction to LabVIEW include its emphasis on real-world problem solving, its early introduction and routine use of data acquisition hardware, its Do It Yourself projects and Use It! examples at the end of each chapter, and its healthy offering of back-of-the-chapter homework problems.

Real-World Problem Solving: Chapter topics and exercises provide examples of how commonly encountered problems are solved by scientists and engineers in the lab. LabVIEW features, along with relevant mathematical background, are
introduced in the course of solving these problems. The “best practice” strategies presented (such as modularity and data dependency) equip readers to optimize their use of LabVIEW.

**Data Acquisition Usage Throughout:** LabVIEW’s Express VIs allow exercises involving DAQ hardware to appear early and then routinely in *Hands-On Introduction to LabVIEW*. Express VIs package common measurement tasks into a single graphical icon and so allow the user to write a program with minimal effort. Of particular note, following the book’s first four software-only chapters that teach the fundamentals of the LabVIEW programming language, data acquisition using a DAQ device is covered in Chapters 5 and 6. For a professor or self-learner who wishes to devote only three or four weeks to instruction in computer-based data acquisition, Chapters 1 through 6 will provide the needed instructional materials. For those planning a more comprehensive study of LabVIEW, the Express VIs allow construction of a state-machine digital oscilloscope, digital thermometer, and spectrum analyzer in Chapters 9, 11, and 12, respectively. In Chapter 13, the control of a DAQ device via the more advanced programming DAQmx icons is covered. In contrast to the Express VIs, the DAQmx icons enable a user to utilize the full available range of the DAQ-device features. In Chapter 14, data are acquired remotely from a stand-alone instrument using the GPIB and/or USB interface bus and, in Appendix C, interested readers can use a DAQ device to precisely control the temperature of an aluminum block. Additionally, commonly used interfacing circuits consisting of low-cost integrated circuits are presented. Circuits include an anti-aliasing filter, thermocouple signal conditioner, and digital potentiometer that communicates via the Serial Peripheral Interface (SPI).

**Do It Yourself Projects:** To allow readers to gauge their understanding of the presented material, each chapter of *Hands-On Introduction to LabVIEW* concludes with a Do It Yourself project. Each of these projects poses an interesting problem and (loosely) directs readers in applying the chapter’s material to find a solution. In some chapters, this project involves writing a program that functions as a stopwatch (Chapter 2) or determines a person’s reaction time (Chapter 10); in other chapters the reader constructs a computer-based instrument including a digital thermometer (Chapter 11), a spectrum analyzer (Chapter 12), and a frequency meter (Chapter 13).

**Use It! Examples:** Ready-to-use example programs, which carry out common tasks encountered in laboratory work, are presented at the end of each chapter. Some of these examples involve programming solutions, for example, showing how to input parameters at the beginning of a data run, save and plot data during runtime, and apply a criterion to a sequence of values to selectively build a data array. Others examples are low-cost hardware solutions, including anti-aliasing through the use of an eighth-order Butterworth low-pass filter, amplification and cold-junction compensation for a thermocouple temperature measurement, control
of integrated circuits using SPI communication, and construction of an Arduino-based voltmeter and digital oscilloscope.

**Back-of-the-Chapter Homework Problems:** A selection of homework-style problems is included at the end of each chapter so that interested readers can further develop their LabVIEW-based skills. In some of these problems, readers test their understanding by applying the chapter topics to new applications (e.g., Bode magnitude plot); in others, readers use programs written within the chapter to explore important experimental issues (e.g., frequency resolution of a fast Fourier transform). Finally, a number of problems introduce readers to features of LabVIEW relevant to, but not included in, the chapter’s text (e.g., data storage in binary format).

**Improvements to the Fourth Edition:** This new edition includes the following improvements:

- New chapter interactively introduces all features of National Instruments DAQ devices using the Measurement & Automation Explorer (MAX). [Chapter 5]
- New Use It! examples at the end of each chapter present ready-to-use programs that carry out common tasks encountered in laboratory work.
- Commonly used, low-cost integrated circuits (for example, eighth-order Butterworth low-pass filter, thermocouple signal conditioner) highlighted in end-of-the-chapter problems and Use It! examples.
- LabVIEW control of an Arduino is demonstrated through construction of Arduino-based voltmeter and digital oscilloscope. [Chapter 14]
- All chapters are fully updated to the latest version of LabVIEW. DAQ hardware now commonly used in instructional laboratories and self-learning is highlighted.
- 14 new end-of-the-chapter problems appear throughout the book.

**Hands-On Introduction to LabVIEW** is fully compatible with the Full Development System, Professional Development System, and Student Edition of LabVIEW. In addition, all chapters may be carried out by Base Development System owners, with the exception of Chapters 11 and 12 (since the Base Development System does not include curve fitting and fast Fourier transform functionality). An instructor might consider having students purchase personal copies of the low-cost Student Edition software (the Student Edition can now be purchased by itself at a very affordable price; that is, it is no longer necessary to buy an expensive bundled book/software package). With their own LabVIEW software, students can perform non-hardware-related chapter sections and/or back-of-the-chapter problems as homework on their own computers.

To aid readers in creating their LabVIEW programs, the following conventions are used throughout the book: **Bold** text designates the features such as graphical icons, palettes, pull-down menus, and menu selections that are to be manipulated...
in the course of constructing a program. The descriptive names that label controls,
indicators, custom-made icons, programs, disk files, and directories (or folders)
are given the straight font. Italic text highlights character strings that the program-
mer must enter using the keyboard and also signals the first-time use of important
terms and concepts.

Any suggestions or corrections are gladly welcomed and can be sent to John
Essick, Reed College, 3203 SE Woodstock Boulevard, Portland, OR 97202, USA,
or jessick@reed.edu.

Updates, answers to frequently asked questions, and ancillary materials for
Hands-On Introduction to LabVIEW are available at http://academic.reed.edu/
physics/faculty/essick.

Additionally, solutions to the even-numbered back-of-the-chapter problems
can be downloaded at www.oup.com/us/essick. Instructors who adopt this book
for a course can obtain a password-protected link to the solution set for every prob-
lem from Oxford University Press.

For their advice and assistance in preparing this revision of Hands-On Intro-
duction to LabVIEW, I thank Dan Kaveney, Megan Carlson, and Claudia Duke-
shire of Oxford University Press. For their helpful comments and suggestions, I
express my appreciation to the reviewers.

• Prathap Basappa, Norfolk State University
• Armando Carrasco, Austin Community College
• James Doyle, Macalester College
• Hector Gutierrez, Florida Institute of Technology
• Aubri Hanson, Chipola College
• Robert Haring-Kaye, Ohio Wesleyan University
• Saliman Isa, South Carolina State University
• Robert Muratore, Hofstra University
• Robert Polak, Loyola University Chicago
• John Viator, Duquesne University
• Zifeng Yang, Wright State University

Finally, to my family: Thank you for your love and support while I worked on this
project.

John Essick
Portland, Oregon
About the Author

John Essick is a professor at Reed College with research interests in the optoelectronic properties of semiconductors. Since 1993, he has taught computer-based experimentation using LabVIEW as part of Reed’s junior-level Advanced Laboratory and used LabVIEW to carry out many research projects.