

# Table of Contents

PREREQUISITE.....	2
Functional specification document.....	2
Requirements document.....	2
Software design document.....	2
Hardware design document.....	2
.....Software repository.....	2
Overview.....	2
DEVELOPMENT GUIDELINES.....	3
LABVIEW USER MANUAL.....	3
LABVIEW COURSES.....	3
Labview Fundamentals.....	3
Basic course I.....	3
Basic course II.....	4
.....Advanced.CERN.Customized.Course.....	4
Getting started.....	4
.....LABVIEW.Real-Time.I.....	4
DATA COMUNICATION.....	5
MEASUREMENTS AND DATA ACQUISITION.....	5
MOTION CONTROL.....	5
TEMPLATES.....	6
Overview.....	6
State Machines.....	6
.....Functional.Global.Variables.....	6
REQUIREMENTS.GATEWAY.....	7



The following page is designed to give guidelines for the National Instrument Labview software. The material is divided by topics.

## **PREREQUISITE**

To obtain an exhaustive project, the following milestones documents and actions should be implemented.

### **Functional specification document**

It is a document that reviews all the function of the software and the related interconnection to other software or hardware devices. Please see the following guideline.

[FunctionalSpecifications.pdf](#)

### **Requirements document**

It is a document that defines all the requirements that the software has to fulfil. Use the NI requirements gateway to tag and cover all the requirements. See the Requirements Gateway section of this web tutorial and the example in the following attachments.

PrsRequirements.docx

MdcRequirements.docx

### **Software design document**

It is a document that describes in technical details the software. Please follow the the IEEE standard for code documentation present in the Code Documentation section of this web tutorial. Examples in the following attachments.

PrsSoftwareDesignDocument.docx

MdcSoftwareDesignDocument.docx

### **Hardware design document**

It is a document that describes the hardware (if present) that is controlled by the software. Please see the example in the following attachment.

LCCS\_HW\_Design\_Specification.docx

### **Software repository**

Use tortoise (<http://tortoisesvn.tigris.org/>) to commit software into the section software repository for labview code. The section repository software is named STI-ECE-SVN located in the CERN web SVN. Please ask Rosario Esposito for access rights.

## **Overview**

Overview goals:

- Become comfortable with the LabVIEW environment and data flow execution

- Ability to use LabVIEW to solve problems
- LabVIEW Concepts :
  - ◆ Acquiring, saving and loading data
  - ◆ Find and use math and complex analysis functions
  - ◆ Work with data types, such as arrays and clusters
  - ◆ Displaying and printing results

[Introduction\\_to\\_LabVIEW-Overview.pdf](#)

## DEVELOPMENT GUIDELINES

The LabVIEW Development Guidelines describe many of the issues that arise when developing large applications. The guidelines are based on the advice of LabVIEW developers and provide a basic survey of software engineering techniques you might find useful when developing your own projects.

This manual also includes a discussion of style for creating VIs. Developers who have used LabVIEW and are comfortable in the LabVIEW environment can use the LabVIEW Development Guidelines to maintain a consistent style in their projects.

[Introduction\\_to\\_LabVIEW-Development\\_Guidelines.pdf](#)

## LABVIEW USER MANUAL

This manual describes the LabVIEW graphical programming environment and techniques for building applications in LabVIEW, such as test and measurement, data acquisition, instrument control, datalogging, measurement analysis, and report generation applications.

[Documentation\\_-\\_UserManual.pdf](#)

## LABVIEW COURSES

### Labview Fundamentals

This manual describes Labview programming concepts, techniques, features, VIs and function you can use to create test and measurement, data acquisition, instrument control, datalogging, measurement analysis and report generation application.

[LabView\\_Fundamentals.pdf](#)

### Basic course I

Use this manual to learn about LabVIEW programming concepts, techniques, features, VIs, and functions you can use to create test and measurement, data acquisition, instrument control, datalogging, measurement analysis, and report generation applications. This course manual assumes that you are familiar with Windows and that you have experience writing algorithms in the form of flowcharts or block diagrams. The course manual is divided into lessons, each covering a topic or a set of topics. Each lesson consists of the following:

- An introduction that describes the purpose of the lesson and what you will learn
- A description of the topics in the lesson
- A set of exercises to reinforce those topics

Some lessons include optional and challenge exercise sections or a set of additional exercises to complete if time permits.

- A summary that outlines important concepts and skills taught in the lesson Several exercises in this manual use one of the following National Instruments hardware products:

- A plug-in multifunction data acquisition (DAQ) device connected to a DAQ Signal Accessory containing a temperature sensor, function generator, and LEDs
- A GPIB interface connected to an NI Instrument Simulator

If you do not have this hardware, you still can complete the exercises. Alternate instructions are provided for completing the exercises without hardware. Exercises that explicitly require hardware are indicated with an icon, shown at left. You also can substitute other hardware for those previously mentioned. For example, you can use a GPIB instrument in place of the NI Instrument Simulator, or another National Instruments DAQ device connected to a signal source, such as a function generato

[Labview\\_Basics\\_I.pdf](#)

## Basic course II

Use this manual to learn about LabVIEW programming concepts, techniques, features, VIs, and functions you can use to create test and measurement, data acquisition, instrument control, datalogging, measurement analysis, and report generation applications.

This course manual assumes that you are familiar with Windows, Macintosh, or UNIX; that you have experience writing algorithms in the form of flowcharts or block diagrams; and that you have taken the **LabVIEW Basics I** or have equivalent experience.

The course manual is divided into lessons, each covering a topic or a set of topics. Each lesson consists of the following:

- An introduction that describes the purpose of the lesson and what you will learn
- A description of the topics in the lesson
- A set of exercises to reinforce those topics
- A set of additional exercises to complete if time permits
- A summary that outlines important concepts and skills taught in the lesson

[Labview\\_Basics\\_II.pdf](#)

## Advanced CERN Customized Course

The custom advanced course contains description on how to make software engineering process, system architecture, best practice developments and validation.

[AdvancedCustomCourse.pdf](#)

## Getting started

This document provides exercises to teach you how to develop a real-time project and VIs from setting up RT targets to building, debugging, and deploying real-time applications. This document provides references to the LabVIEW Help and other Real-Time Module documents for more information as you create the real-time application.

[Labview\\_RealTime\\_Module\\_-\\_getting\\_started\\_.pdf](#)

## LABVIEW Real-Time I

The LabVIEW Real-Time 1 course teaches you to use LabVIEW Real-Time to develop a deterministic and reliable application. Most LabVIEW applications run on a general-purpose operating system (OS) like Windows, Linux, Solaris, or Mac OS. Some applications require deterministic real-time performance that general-purpose operating systems cannot guarantee. The LabVIEW Real-Time Module extends the

capabilities of LabVIEW to address the need for deterministic real-time performance.

This course assumes you have a level of experience with LabVIEW equivalent to completing the material in the LabVIEW Core 1 course. In addition, you should be familiar with the Windows operating system and computer components such as the mouse, keyboard, connection ports and plug-in slots, and have experience writing algorithms in the form of flowcharts or block diagrams. The course and exercise manuals are divided into lessons, described as follows.

In the course manual, each lesson consists of the following:

- An introduction that describes the purpose of the lesson and what you will learn
- A description of the topics in the lesson
- A summary quiz that tests and reinforces important concepts and skills taught in the lesson

In the exercise manual, each lesson consists of the following:

- A set of exercises to reinforce topics
- (Optional) Self-study and challenge exercise sections or additional exercises

Course	Labview_RealTime_Module_-_basic.pdf
Exercise	Labview_RealTime_Module_-_basic_exercise.pdf

Labview online courses from National Instruments are accessible through the following link being registered using the serial key that CERN uses for the software.

<http://sine.ni.com/myini/self-paced-training/app/main.xhtml>

## DATA COMMUNICATION

An overview of Data Communication in LabView

[Data\\_Communication.pdf](#)

## MEASUREMENTS AND DATA ACQUISITION

An overview measurements and data acquisition in LabView.

[Measurement.pdf](#)  
[Data\\_acquisition.pdf](#)  
[Daqm.pdf](#)

## MOTION CONTROL

An overview of motion control in LabView.

[Motion\\_Control.pdf](#)

# TEMPLATES

## Overview

LabVIEW 2013 continues to expand the in-product templates and sample projects, which provide recommended starting points designed to ensure the quality and scalability of a system. All of the

templates and sample projects are open-source and include extensive documentation designed to clearly indicate how the code works and the best practices for adding or modifying functionality. In

addition to demonstrating recommended architectures, these projects also illustrate best practices for documenting and organizing code.

You can create a new LabVIEW project from an existing template or sample project by selecting "Create Project" from the Getting Started window. The list of available options depends on the

software modules and drivers that you have installed. This document lists the templates and sample projects that National Instruments provides and supports.

[TemplateOverview.pdf](#)

## State Machines

The concept of "State Machines" as they pertain to LabVIEW based application development. Focus will be on design considerations and merits for selecting various State Machines models.

### Topics

- Single loop
- Multiple Loops (Asynchronous)
- Supporting techniques
- Queue
- Functional Globals
- Multi-threading and performance (comments)

Architecture	State_Machine_Architectures.pdf
State Machine and Queue	State_Machines_Queue_based_at_end_.pdf

## Functional Global Variables

In NI LabVIEW software, the order of execution is controlled by the flow of data (data flow) through wires rather than the sequential order of commands. This allows you to create a block diagram with simultaneous (parallel) operations. When you have parallel loop structures, you cannot use wires to communicate data between the two loops because data flow prevents parallel operation. To overcome this, you must use variables. With variables, you can circumvent normal data flow by passing data from one place to another without connecting the two places with a wire. In LabVIEW, variables take many forms.

[LocalGlobalVariablesRaceConditions.pdf](#)  
[Functional\\_Global\\_Variables.pdf](#)

The following document has to be use as templates for the software design document.

[Software\\_Design\\_Document\\_Reference.pdf](#)

In the following document the Labview error codes are explained.

[Documentation\\_-\\_Error\\_Codes.pdf](#)

## REQUIREMENTS GATEWAY

The code implementation could be driven by a requirement document. To link the requirements to code modules, NI Requirements Gateway could be use.

[Requirements\\_Gateway.pdf](#)

In the following there is a list of various examples and exercises.

[Introduction\\_to\\_LabVIEW.zip](#)  
[Motion\\_State\\_Machine.zip](#)  
[Functional\\_Global\\_Variables.zip](#)  
[DMA\\_Best\\_Practices.zip](#)  
[Bioreactor\\_Scan\\_Engine.zip](#)  
[Bioreactor\\_FPGA.zip](#)  
[LabviewRealTimeExercise.rar](#)  
[Network\\_Streams\\_Example.zip](#)  
[yED\\_Basic\\_cRIO\\_Architectures.zip](#)

[cla\\_exam\\_prep\\_guide.pdf](#)  
[cla\\_sample\\_atm\\_solution.rar](#)  
[cla\\_sample\\_exam\\_atm\\_machine.pdf](#)

- Member list (comma-separated list):
  - Set GROUP = MarioDiCastro, AlessandroMasi
- Persons/group who can change the list:
  - Set ALLOWTOPICCHANGE = EnStiEceLabViewGroup

Twiki Web page created on 28.10.2013

-- MarioDiCastro - 28-Oct-2013

Added CLA preparation guide and example exercises.

-- MarioDiCastro - 31 Oct 2013

---

This topic: Main > IntroductionToLabview

Topic revision: r5 - 2013-12-12 - MarioDiCastro



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.  
Ideas, requests, problems regarding TWiki? Send feedback