Getting Started with the LabVIEW™ Robotics Module

Version 2009

The LabVIEW Robotics Module is a software package that allows you to develop and deploy a robotics application using LabVIEW, other National Instruments software, and device drivers.

This document discusses how to develop a robotics project in LabVIEW and deploy the application to a hardware target, such as a real-time controller. Use this manual to access information about robotics programming concepts and to complete exercises to teach you how to set up hardware controllers and build and deploy applications. This document provides references to the *LabVIEW Help* and other National Instruments documents for more information as you create the robotics application.

Contents

Overview of the LabVIEW Robotics Module

In the LabVIEW development system, you build programs, or VIs, that include a user interface, or front panel, and a block diagram that contains graphical code, such as other VIs and structures, that controls the front panel. Wires connect block diagram objects, and data flows through the wires from object to object.

If you are new to LabVIEW, use the *Getting Started with LabVIEW* manual as a tutorial to familiarize yourself with the LabVIEW graphical programming environment. Access this document by navigating to `LV_Getting_Started.pdf` in the `labview\manuals` directory.

Many LabVIEW modules and toolkits and other National Instruments software products add functionality to the LabVIEW development system. The LabVIEW Robotics Module adds the following components to LabVIEW:

- **VIs and Instrument Drivers**—Robotics Module VIs provide algorithms for navigation, steering, and so on. The Robotics Module also provides drivers that allow you to control certain sensors commonly used in robotics systems.

  To access the Robotics VIs and instrument drivers, select **View»Functions Palette** from the block diagram in LabVIEW and navigate to the **Robotics** palette. You also can right-click any blank space on the block diagram to display the **Functions** palette. The items on the **Robotics** palette are divided into subpalettes according to the functionality they provide, as shown in Figure 1.

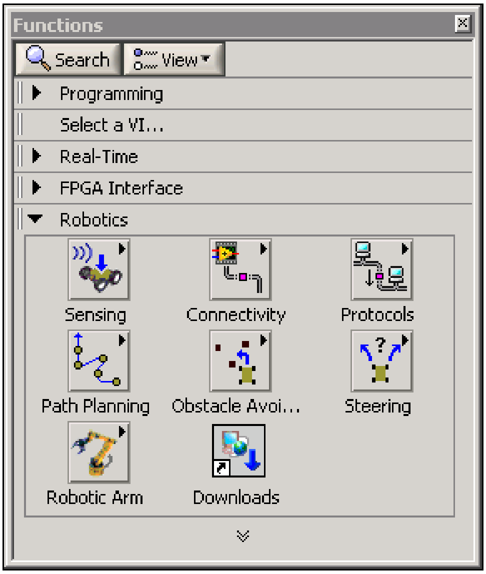


Figure 1. Robotics Palette

- **Robotics Project Wizard**—The Robotics Project Wizard allows you to create a new LabVIEW project configured with your hardware. Refer to the *Creating a Robotics Project* section for information about using the Robotics Project Wizard. Select **View»Getting Started Window** and click the **Robotics Project** link in the **Create New** section to launch the Robotics Project Wizard.
- **Examples**—Example VIs demonstrate robotics concepts, such as instrument control, path planning, and kinematics. You can modify an example VI to fit an application, or you can copy and paste from one or more example VIs into a VI that you create. Use the NI Example Finder, available in LabVIEW by selecting **Help»Find Examples** to browse or search for example VIs. The **Robotics** category opens automatically.

  Note Some robotics examples require additional software. In the NI Example Finder, select an example and refer to the **Requirements** section for a list of software you must install to run the example.

You also can access robotics-specific example VIs from the `labview\examples\robotics` directory. Access examples for certain instrument drivers in the `labview\instr.lib\`*`instrument`* directory, where *instrument* is the name of a device for which you have drivers installed.

Related National Instruments Software

Depending on the LabVIEW Robotics package you purchase, you have access to certain LabVIEW modules and toolkits, other National Instruments software, and device drivers that add functionality useful in robotics programming to the LabVIEW development system. For example, you will complete an exercise later in this manual that requires the LabVIEW Real-Time Module and LabVIEW FPGA Module. The Real-Time Module allows you to use LabVIEW graphical programming to build deterministic applications that run on some real-time operating systems, and the FPGA Module allows you to create VIs that run on National Instruments FPGA targets.

Refer to the readme file on the Robotics DVD for information about the products to which you have access.

Device Drivers

LabVIEW Robotics DVD packages include drivers for a variety of programmable sensors. You can access information about many drivers, such as the device models the drivers support, in readme files located in the `labview\instr.lib\`*`instrument`* directory, where *instrument* is the name of a device for which you have drivers installed. Refer to the National Instruments Web site at `ni.com/info` and enter the info code `ex3mbp` to find and download additional sensor drivers.

Documentation

After you install a LabVIEW add-on such as a module, toolkit, or driver, the documentation for that add-on appears in the *LabVIEW Help* or appears in a separate help system you can access by selecting **Help»*Add-On Help***, where ***Add-On Help*** is the name of the separate help system for the add-on.

For add-on help that appears in the *LabVIEW Help*, you can locate help for the add-on you installed on the **Contents** tab of the *LabVIEW Help*, in one of the following places.

- In a book titled ***Add-On***, where ***Add-On*** is the name of the product you installed
- In the **Toolkits** book, in a book titled ***Add-On***, where ***Add-On*** is the name of the product you installed.

Related Documentation

The following documents contain information that you might find helpful as you read this manual.

LabVIEW Help

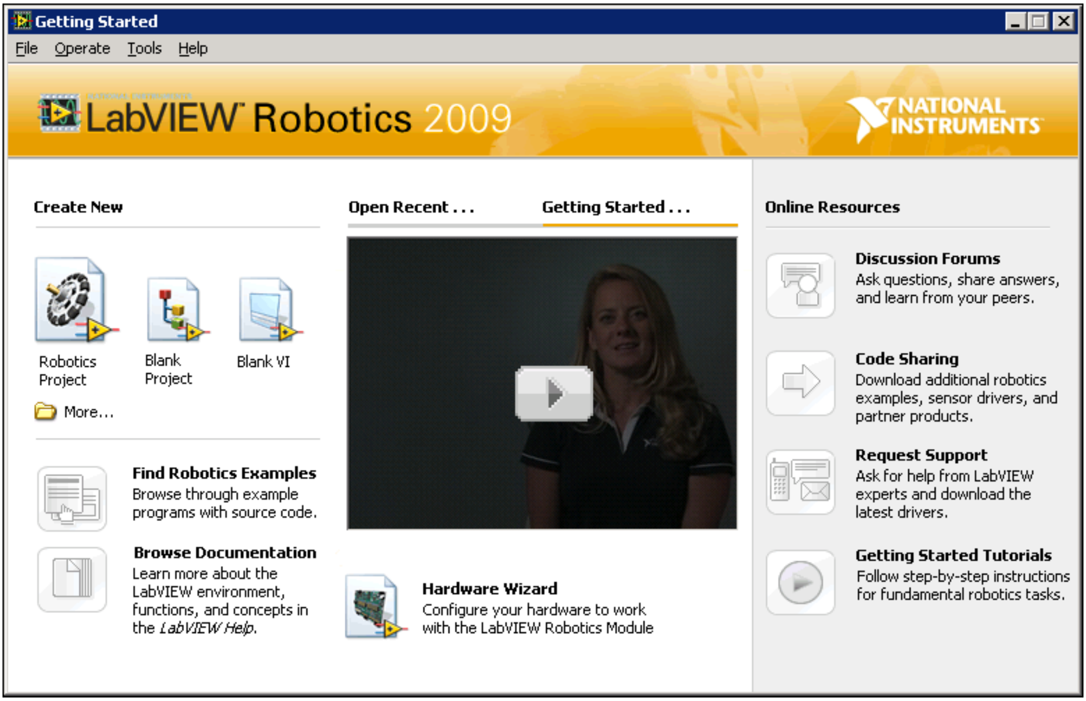
Use the *LabVIEW Help* to access information about LabVIEW programming concepts, step-by-step instructions for using LabVIEW, and reference information about LabVIEW VIs, functions, palettes, menus, and tools. Access the *LabVIEW Help* by selecting **Help»Search the LabVIEW Help** in LabVIEW. Refer to the topics in the **Getting Started with LabVIEW** book on the **Contents** tab of the *LabVIEW Help* to familiarize yourself with the LabVIEW graphical programming environment.

The *LabVIEW Help* contains the **Robotics Module** book, which provides procedural and conceptual information about developing robotics applications. The **Robotics VIs** book includes reference information about using the Robotics Module VIs that appear on the **Functions** palette.

Complete the following steps to access the Robotics Module section of the *LabVIEW Help*.

1. Launch LabVIEW or select **View»Getting Started Window** to display the **Getting Started** window shown in Figure 2. You can use the Getting Started window to create new VIs and robotics projects,

select among the most recently opened LabVIEW files, find examples, and launch the *LabVIEW Help* and other robotics-related resources.



Browse through example programs with source code.

Learn more about the LabVIEW environment, functions, and concepts in the *LabVIEW Help*.

Configure your hardware to work with the LabVIEW Robotics Module

Ask questions, share answers, and learn from your peers.

Download additional robotics examples, sensor drivers, and partner products.

Ask for help from LabVIEW experts and download the latest drivers.

Follow step-by-step instructions for fundamental robotics tasks.

Figure 2. Getting Started Window

2. Select **Browse Documentation** from the **Getting Started** window to launch the *LabVIEW Help*.
3. Expand the **Robotics Module** book on the **Contents** tab of the *LabVIEW Help*.

Note You can search for a specific topic in the *LabVIEW Help* by clicking the **Search** tab and searching for the topic name enclosed in quotes.

PDF Documents

Documents, such as the *LabVIEW Upgrade Notes*, are available as PDFs when you install LabVIEW. Access these documents by navigating to the `labview\manuals` directory.

Robotics Module Readme

The `readme_Robotics.html` file contains additional help resources you can use while working with the Robotics Module, including information on known issues with the module. Access this document by navigating to the `labview\readme` directory and opening `readme_Robotics.html`.

Components of a LabVIEW Robotics System

A robotics system consists of software and hardware components. In a system designed with the LabVIEW Robotics Module, the software components include National Instruments software, such as the LabVIEW projects and VIs you create in LabVIEW. The hardware components of a robotics system might include the following:

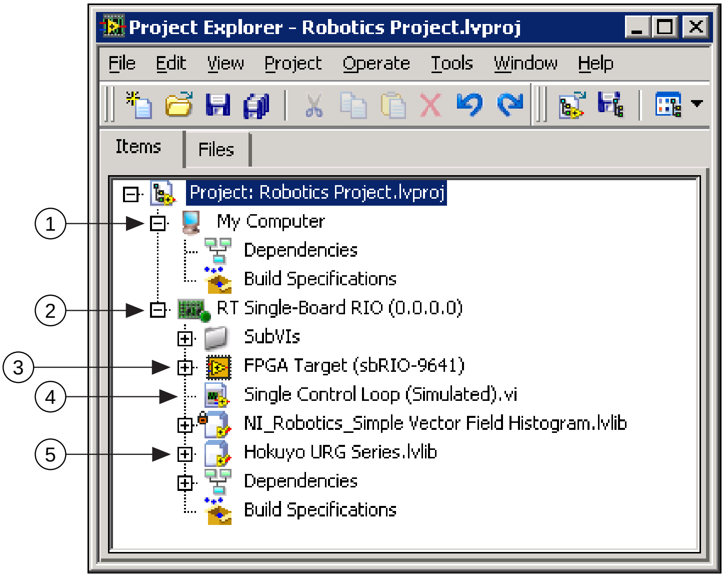
- **Host computer**—Develop and debug robotics applications on a host computer. You also can communicate with the robot and log data it acquires using the host computer. Run applications you develop on the host computer, or deploy them to a Windows-based system or real-time controller.
- **Real-time controller**—Deterministically run the programs you create in LabVIEW and control data input and output with a real-time, or RT, controller, which serves as the "brain" of the system.
- **Sensors and actuators**—Acquire data about the environment of the system with sensors, such as cameras, and control robot motion with actuators.

Host Computer

The host computer is a computer with the LabVIEW development system and LabVIEW modules and toolkits installed on which you develop the VIs for the robotics system. After you develop the robotics VIs, you can download and run the VIs on RT and FPGA targets, such as CompactRIO and Single-Board RIO products. The host computer also can run VIs that communicate with the VIs running on targets to provide a user interface.

LabVIEW Projects

Use LabVIEW projects on the host computer (1) to group LabVIEW files and files not specific to LabVIEW, create stand-alone applications, and deploy or download VIs (4) and other files, such as instrument drivers (5), to RT and FPGA targets (2, 3). When you save a project, LabVIEW creates a project file (`.lvproj`), which can include references to files in the project, configuration information, build information, and deployment information. Figure 3 shows the parts of a typical LabVIEW project.


Project Explorer - Robotics Project.lvproj
File Edit View Project Operate Tools Window Help
Items
Files
Project: Robotics Project.lvproj
1
My Computer
Dependencies
Build Specifications
2
RT Single-Board RIO (0.0.0.0)
SubVIs
3
FPGA Target (sbRIO-9641)
4
Single Control Loop (Simulated).vi
NI_Robotics_Simple Vector Field Histogram.lvlib
5
Hokuyo URG Series.lvlib
Dependencies
Build Specifications

1. Host computer
2. Real-time controller
3. FPGA target
4. RT VI
5. Instrument drivers

Figure 3. Parts of a LabVIEW Project

From the **Getting Started** window, select **Blank Project** in the **Create New** section to create a new LabVIEW project. You also can use the Robotics Project Wizard to create a LabVIEW project from a template. Refer to the *Creating a Project Using the Robotics Project Wizard* section of this manual for information about creating a LabVIEW robotics project using the Robotics Project Wizard.

Real-Time Controller

Real-time controllers deterministically run programs you create in LabVIEW. Some real-time controllers, such as National Instruments CompactRIO and Single-Board RIO products, consist of a real-time processor and operating system and a Field Programmable Gate Array (FPGA). Other NI real-time controllers, such as PXI products, contain only a real-time processor. You also can use certain third-party hardware, such as ARM microcontrollers, to run LabVIEW programs. Work with third-party hardware in the **Project Explorer** window like an NI RT or FPGA target.

Real-Time Target

The real-time processor and operating system, or RT target, serves as the "brain" of the robotics system, running VIs you create using LabVIEW. Robotics applications require many tasks to complete deterministically, or within a guaranteed time period. For example, VIs that perform obstacle avoidance must run cyclically to prevent a robotic vehicle from running into obstacles. The RT Engine, a version of LabVIEW that runs on RT targets, deterministically executes VIs in which you prioritize tasks so that the most critical task can take control of the processor when necessary.

FPGA Target

An FPGA is an embedded chip that you can reconfigure for different applications. Some controllers, such as CompactRIO and Single-Board RIO products, contain integrated FPGA targets, which are directly connected to the I/O modules that access sensor and actuator data. By default, this FPGA automatically communicates with I/O modules and provides deterministic I/O to the real-time processor.

You typically program the FPGA target to perform the lowest-level I/O and control tasks in a robotics application. You can use normal LabVIEW programming techniques to develop FPGA applications on the host computer and download the VI on the target.

I/O Modules

National Instruments offers a variety of analog and digital I/O modules that directly connect to sensors and actuators.

Note The *LabVIEW Help* does not contain hardware-related information about specific networked devices. Refer to the appropriate device documentation for information about the device.

Sensors and Actuators

Robotics systems acquire data about the environment around them with sensors, such as laser range finders, and move through the use of actuators. To help speed your development, National Instruments works with major sensor vendors to provide a library of ready-to-use instrument drivers, which correspond to programmatic operations such as configuring, reading from, and writing to sensors and actuators. Instrument drivers reduce program development time by eliminating the need to learn the programming protocol for each device.

Refer to the *Creating a Project Using the Robotics Project Wizard* section of this manual for information about using the Robotics Project Wizard to detect robotics-related sensors for which you have drivers installed.

Configuring Software and Hardware for a Robotics System

The remainder of this manual contains exercises to teach you how to build and deploy a robotics application. Before you create a LabVIEW project and application to deploy to a hardware target, you must install software on the host computer and configure the target on which you want the application to run.

Installing Software and Device Drivers on the Host Computer

Refer to the readme file on the Robotics DVD for information about the National Instrument products to which you have access, system requirements, and instructions for installing LabVIEW, modules, and toolkits on the host computer.

Note If you purchased only the LabVIEW Robotics Module, rather than the LabVIEW Robotics Bundle, you can use the Robotics Module with only the LabVIEW development system to develop programs that include VIs and sensor drivers on the **Robotics** palette. However, you must install the LabVIEW Real-Time and FPGA Modules and the NI-RIO drivers to complete the exercises in this manual. You can install evaluation versions of these products from the Robotics Module DVD.

Configuring an RT Target

Before you can deploy robotics applications to an RT target, you must configure the basic settings of the target and install software on the target. If you use an NI Single-Board RIO or CompactRIO product, you can use the National Instruments Robotics Hardware Setup wizard, which launches after you install the LabVIEW Robotics software on the host computer, to configure the target. You also can click the **Hardware Wizard** link in the **Getting Started** section of the **Getting Started** window to launch this wizard manually.

Note If you use another type of target, use the National Instruments Measurement & Automation Explorer (MAX) to detect, configure, and test an RT target. MAX provides access to National Instruments devices and systems, so you can use MAX to communicate with networked RT targets located on the same subnet as the host computer. Select **Start»All Programs»National Instruments»Measurement & Automation** to launch MAX.

In the Robotics Hardware Setup wizard, complete the instructions that appear on the screen to perform the following tasks:

- Detect RT targets in the system
- Set the IP address of the target you want to use
- Download software to the target
- Test that the target is configured correctly

The host computer communicates with the remote system over a standard Ethernet connection. If the host computer is already configured on a network, you must configure the remote system on the same network. If neither machine is connected to a network, you must connect the two machines directly using a CAT-5 crossover cable or hub.

When you complete all the pages in the Robotics Hardware Setup wizard, launch LabVIEW and continue this exercise. If you do not use the Robotics Hardware Setup wizard to configure your RT target or you plan to target a Windows-based target instead of an RT target, launch LabVIEW and click the **Robotics Project** link in the **Create New** section of the **Getting Started** window to launch the Robotics Project wizard.

Creating a Robotics Project

With the Robotics Module, you must use a LabVIEW project to build and configure robotics applications and to work with RT and FPGA targets. Use projects to group LabVIEW files and non-LabVIEW files, configure communication between hardware, create real-time build specifications, and deploy files to targets. When you save a project, LabVIEW creates a project file (`.lvproj`), which includes references to files in the project, configuration information, build information, and deployment information.

Refer to the topics in the **Fundamentals»Working with Projects and Targets** book on the **Contents** tab of the *LabVIEW Help* for more information on LabVIEW projects.

For this exercise you will use the Robotics Project Wizard to create an application that performs simulated navigation for a mobile robot vehicle. The application consists of a VI that simulates data acquisition from a sensor, calculates a heading for travel, and then applies the correct amount of velocity to the robot vehicle.

Note This exercise contains references to creating an application for an RT target. You also can complete the steps in this exercise to create an application for a Windows-based target, but some steps and screenshots will differ slightly from your situation.

Creating a Project Using the Robotics Project Wizard

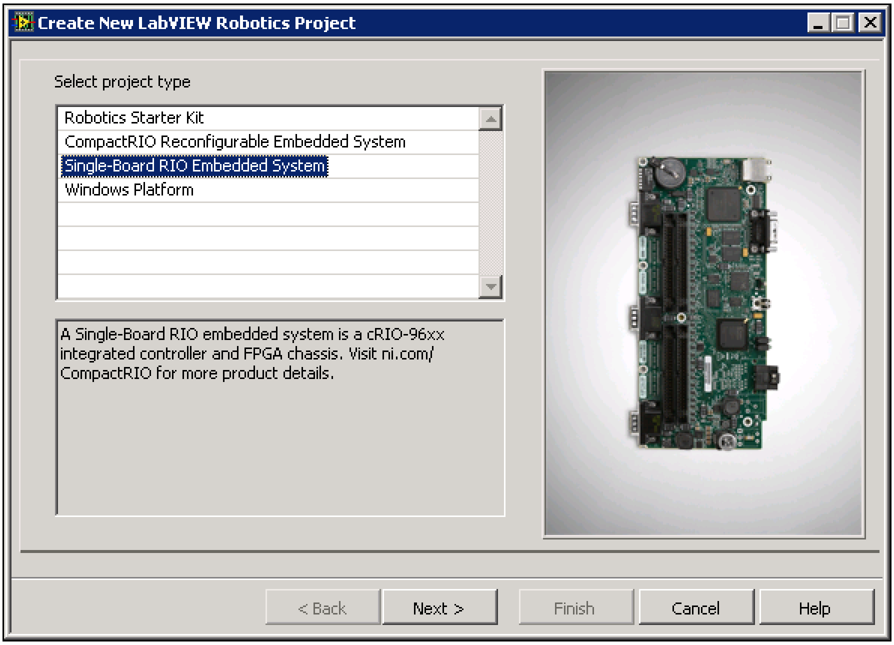
Use the Robotics Project Wizard to create a new project that includes hardware targets, sensors, VIs, and other application files. You can create projects with template VIs and then customize these VIs for your own application.

Complete the following steps to create a project using the Robotics Project Wizard.

1. Click the **Robotics Project** link in the **Create New** section of the **Getting Started** window to launch the Robotics Project Wizard. You also can launch this wizard from the Robotics Hardware Setup wizard or select **File»New** to launch the **New** dialog box and then select **Project»Project from Wizard»Robotics Project**.

   The Robotics Project Wizard displays the **Select project type** page.

2. Select the type of project that corresponds to your system from the **Select project type** list, as shown in Figure 4.



Create New LabVIEW Robotics Project

Select project type

Robotics Starter Kit
CompactRIO Reconfigurable Embedded System
Single-Board RIO Embedded System
Windows Platform

A Single-Board RIO embedded system is a cRIO-96xx integrated controller and FPGA chassis. Visit ni.com/CompactRIO for more product details.

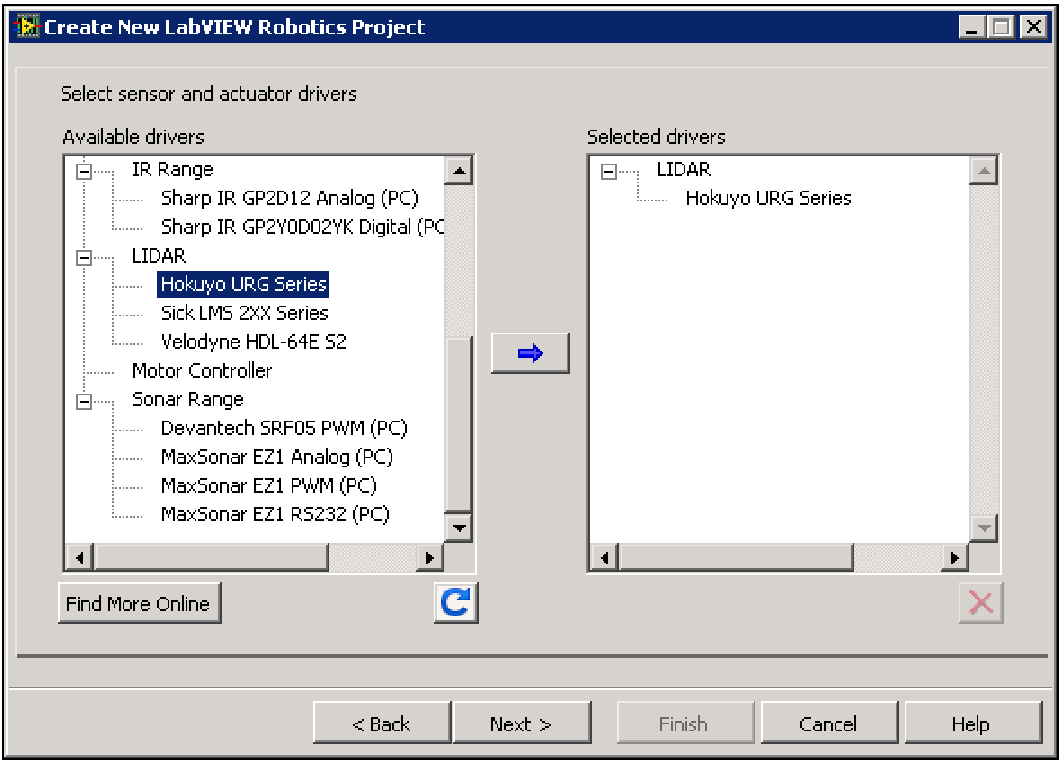
Figure 4. Selecting a Project Type for Your Application

3. Click the **Next** button to accept the settings and continue.
4. The Robotics Project Wizard offers different settings according to the type of system you select. Complete each page for the system you select until you reach the **Select sensor and actuator drivers** page.

   Refer to the *Creating a Project and Application Using the Robotics Project Wizard* topic of the *LabVIEW Help* for more information about using the Robotics Project Wizard.

5. On the **Select sensors and actuator drivers**, page select the **LIDAR»Hokuyo URG Series** item and click the arrow button to add the device to the **Selected drivers** list. The Robotics Project

Wizard displays devices for which you have drivers installed in the `labview\instr.lib\_robotics` directory, as shown in Figure 5.



Create New LabVIEW Robotics Project

Select sensor and actuator drivers

Selected drivers: LIDAR — Hokuyo URG Series

Figure 5. Selecting Sensor Drivers for Your Application

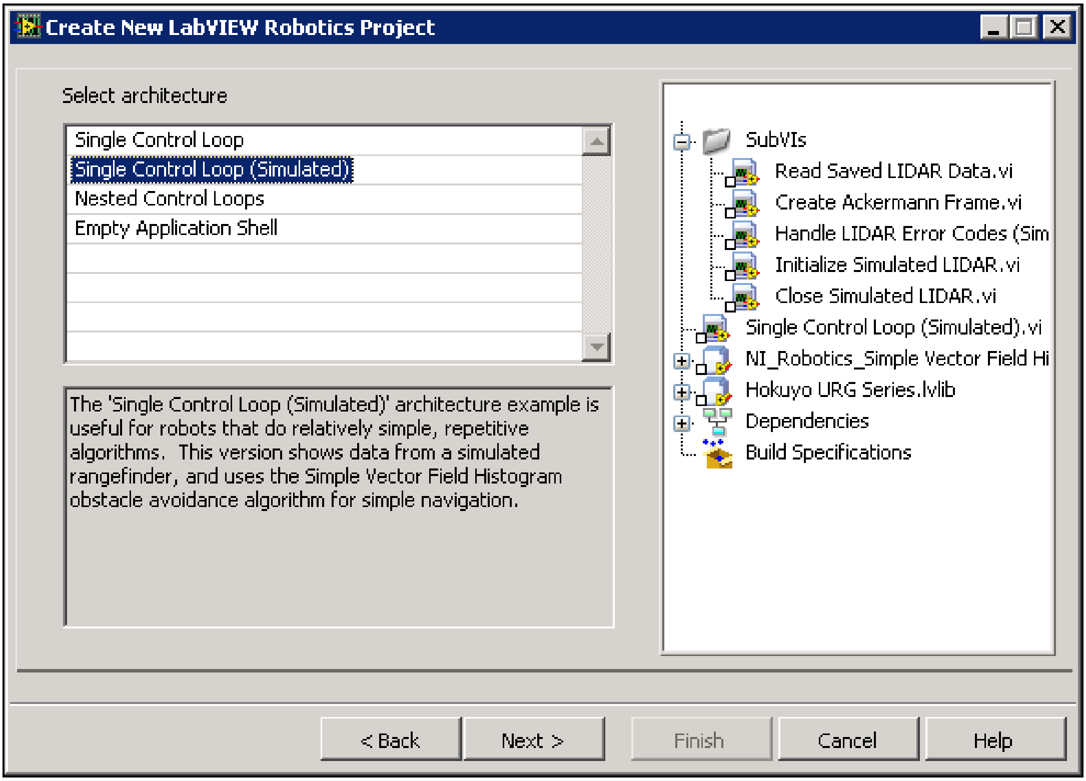
Tip Click the **Find More Online** button to launch the National Instruments Web site in a Web browser, where you can download additional sensor drivers. After you download and install a sensor driver, click the **Refresh** button to update the list of available drivers.

6. Click the **Next** button to accept the settings and continue.

   The Robotics Project Wizard displays the **Select architecture** page.

7. Select the **Single Control Loop (Simulated)** option.

   The Robotics Project Wizard displays a diagram in the right pane that represents the architecture you select, as shown in Figure 6.



The 'Single Control Loop (Simulated)' architecture example is useful for robots that do relatively simple, repetitive algorithms. This version shows data from a simulated rangefinder, and uses the Simple Vector Field Histogram obstacle avoidance algorithm for simple navigation.

Figure 6. Selecting a Project Architecture

8. Click the **Next** button to accept the settings and continue.

   The Robotics Project Wizard displays the **Enter project name and folder** page.

9. Enter `Robotics Project` in the **Project name** text box.
10. Specify the directory where you want to save the project files on the host computer in the **Project folder** text box or click the **Browse** button to browse to the directory in the file system.
11. Click the **Finish** button to exit the Robotics Project Wizard and generate the new project, VIs, and support files for the robotics application. The **Project Explorer** window displays the new project that you create.

Exploring the Project in the Project Explorer Window

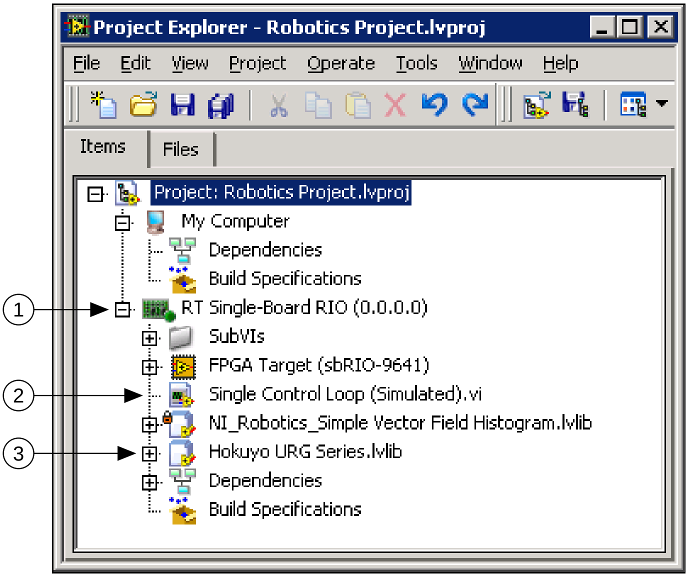
The **Project Explorer** window enables you to manage the hardware targets, VIs, and other support files of the project from one location. The **Project Explorer** window includes the following sections:

- **Project root**—Contains the host computer and the RT targets assigned to the current project.
  - **My Computer**—Represents the host computer in the project.
    - **Dependencies**—Includes items that VIs under the host require.
    - **Build Specifications**—Includes build configurations for source distributions and other types of builds available in LabVIEW toolkits and modules. You can use **Build**

Specifications to configure stand-alone applications, shared libraries, installers, and zip files.

– ***Target***—Represents hardware targets in the project, where *Target*, by default, is the RT or FPGA target name that appears in the Robotics Hardware Setup wizard or MAX. The **Project Explorer** window also includes **Dependencies** and **Build Specifications** sections under each target.

VIs and libraries that you add to a hardware target appear under the target in the **Project Explorer** window. A VI appears under a target to indicate that you can run the VI on that target. Figure 7 shows a project similar to the `Robotics Project.lvproj` file you created with the Robotics Project Wizard.


Project Explorer - Robotics Project.lvproj
File Edit View Project Operate Tools Window Help
Items
Files
Project: Robotics Project.lvproj
My Computer
Dependencies
Build Specifications
1
RT Single-Board RIO (0.0.0.0)
SubVIs
FPGA Target (sbRIO-9641)
2
Single Control Loop (Simulated).vi
NI_Robotics_Simple Vector Field Histogram.lvlib
3
Hokuyo URG Series.lvlib
Dependencies
Build Specifications

1. Real-time controller, with target name and IP address
2. RT VI from Project Wizard architecture
3. LIDAR sensor drivers

Figure 7. Exploring the Robotics Project in the Project Explorer Window

Under the RT target (1), the project contains the Single Control Loop (Simulated) VI (2) you created with the Robotics Project Wizard. This VI contains a program that simulates a mobile robot vehicle that travels in response to obstacles in the robot environment. The VI uses simulated light detection and ranging (LIDAR) sensor data that describes the robot environment. In an application that acquires real LIDAR data, the project contains sensor drivers (3) you use to control the LIDAR unit. The Simple Vector Field Histogram VI uses the sensor data to calculate paths for travel.

Complete the following steps to explore the VI and support files you created with the Robotics Project Wizard.

1. Double-click `Single Control Loop (Simulated).vi` in the **Project Explorer** window to launch the main project VI.
2. Select **Window»Show Block Diagram** to display the block diagram of the VI, as shown in Figure 8.

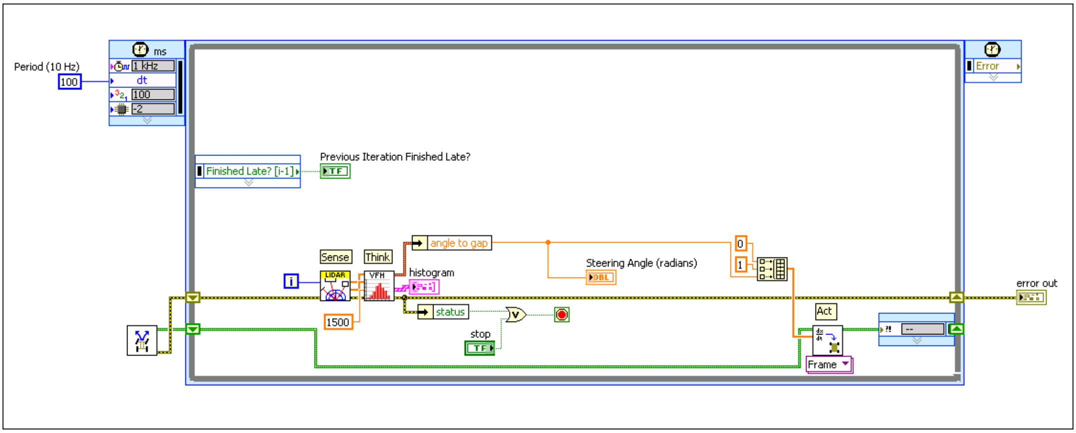


Figure 8. Block Diagram of the Single Control Loop (Simulated) VI

The Timed Loop, configured to run at 10 Hz, controls the code that returns sensor data, calculates the heading to which the vehicle drives, and applies the correct velocity to the vehicle wheels. All of the processing must execute fast enough to keep up with the loop rate.

Tip Select **Help»Show Context Help** to display the **Context Help** window, which displays basic information about LabVIEW objects and the LabVIEW environment when you move the cursor over each object.

3. Select **Window»Show Front Panel** to display the front panel of the VI, as shown in Figure 9.

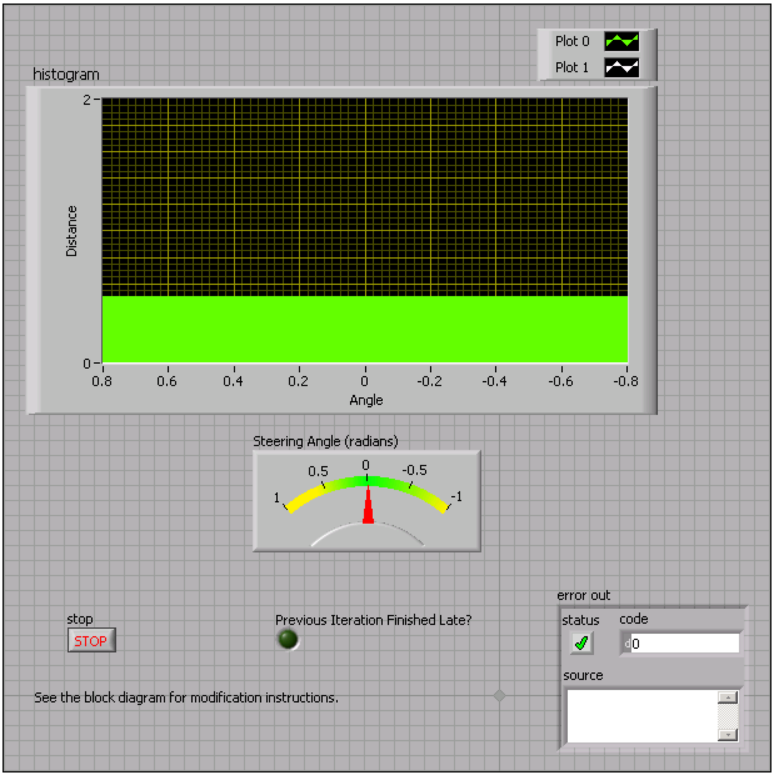


Figure 9. Front Panel of the Single Control Loop (Simulated) VI

When the VI runs, the graph indicator displays data that represents the distances to objects in range of the sensor, arranged by angle and direction, while the meter indicator displays the current steering angle of the robot. Also, the **Previous Iteration Finished Late?** indicator is TRUE if the code in the Timed Loop does not finish running within the specified period.

4. Notice that LabVIEW displays the project and target in which the VI exists at the bottom-left corner of the front panel and block diagram windows. This text indicates the application instance in which the VI exists.
5. Close the front panel of the VI. The block diagram closes automatically.
6. In the **Project Explorer** window, expand the `Hokuyo URG Series.lvlib` project library.

   When you specify in the Robotics Project Wizard that the system includes a sensor or actuator, the LabVIEW project contains a project library, such as `Hokuyo URG Series.lvlib`, that contains driver VIs for the device. Expand the **Public** folder to display the available driver VIs you can drag to the block diagram of a VI for use in the robotics application.
7. Notice that the RT target in the **Project Explorer** window also includes **Dependencies** and **Build Specifications**.

Use **Dependencies** to view items or subVIs that VIs under the RT target require to execute. Refer to the *LabVIEW Help* for more information about viewing dependencies in a project.

Configuring Target Properties

Use the **Properties** dialog box to configure the project settings of a target from the **Project Explorer** window. You can configure general, VI Server, access, debugging, and miscellaneous target settings, depending on the target you are configuring.

Right-click the target in the **Project Explorer** window and select **Properties** from the shortcut menu to open the target **Properties** dialog box and select an item from the **Category** list to display the configuration options. Select **Project»Save Project** from the **Project Explorer** window to save the project and settings you selected for the target in the target **Properties** dialog box.

You must deploy the target before the settings take effect. Deploying a target refers to applying settings defined in the project to a target.

Note Deploying a target overwrites the current target settings with the latest settings specified in the target **Properties** dialog box.

Using the Architectures in the Robotics Project Wizard

The Single Control Loop robot architecture, which the project from the *Creating a Project Using the Robotics Project Wizard* section uses, serves as a starting point for robots that perform simple, repetitive algorithms. You can insert code for acquiring sensor data, processing data, and controlling the robot inside the Timed Loop.

The Robotics Project Wizard also provides a more advanced Nested Control Loops architecture that uses multiple Timed Loops to handle different aspects of robot control. For example, the Timed Loops in this example perform the following tasks:

- Mission planning—Returns a random goal position to which the robot navigates.
- Path planning—Searches a map of the robot environment to find a path to the goal position.
- Driving—Simulates the robot moving along the path to reach the goal position.

An advanced robot architecture also might include Timed Loops that detect obstacles, control movement, and measure the progress of the robot. Loops can run on different hardware targets, so you must implement communication across the application. For instance, the Nested Control Loops example uses shared variables to communicate data between Timed Loops in different VIs.

Refer to the **Fundamentals»Networking in LabVIEW** book on the **Contents** tab of the *LabVIEW Help* for more information about sending data between VIs.

Note Refer to the documentation resources described in the *Where to Go from Here* section of this manual for specific programming guidelines, procedures, and examples. Also, refer to the robotics example applications in the NI Example Finder by selecting **Help»Find Examples** in LabVIEW.

Deploying and Downloading VIs to a Target

Before you run VIs on an RT or FPGA target, you must deploy RT VIs and any dependencies to the RT target and download files on the FPGA target.

Note In the previous section, if you created the project for a Windows-based target, you do not have to deploy the project files before you run the VI. Instead, you can proceed to the *Building a Robotics Application that Runs at Startup* section.

Deploying VIs to RT Targets

When you run an RT VI under an RT target, LabVIEW deploys RT VIs, all items required by the VIs, and the target settings to the target. You can deploy VIs, libraries, and shared variables to an RT target where you can run or reference the items.

Complete the following steps to deploy and run the robotics application you previously created on the RT target.

1. Right-click the `Single Control Loop (Simulated).vi` under the RT target in the **Project Explorer** window and select **Deploy** from the shortcut menu to download the VI to the RT target.

   LabVIEW deploys the VI to the RT target and opens the front panel to indicate that the VI is in memory on the RT target. Closing the front panel removes the VI from memory on the RT target.

2. Click the **Run** button to run the VI on the target.
3. Click the **Stop** button to stop the execution of the Timed Loop in the RT target VI, which stops the execution of the VI.

Note Deploying an RT VI does not save the VI to disk on the RT target, so the VI does not remain on the target after the target restarts. Refer to the *Building a Robotics Application that Runs at Startup* section for information about building and running a stand-alone application that remains on the RT target and runs when the target powers on.

Downloading VIs on FPGA Targets

The project you created with the Robotics Project Wizard does not contain FPGA VIs. However, when a project does contain an FPGA VI, you must compile the VI before you can download and run the VI on an FPGA target.

Refer to the **Robotics Module»Creating Projects and Applications»Deploying, Downloading, and Running VIs and Applications** topic on the **Contents** tab of the *LabVIEW Help* for information about compiling, downloading, and running FPGA VIs.

Building a Robotics Application that Runs at Startup

Use the LabVIEW Application Builder with the Real-Time Module to create a stand-alone robotics application that runs on an RT target. Stand-alone applications do not require you to run them from within a LabVIEW environment. You can embed a stand-alone application on an RT target and configure the application to launch automatically when you power on the target. Use the Application Builder to create a build specification that defines how you want to build an application.

Note If in the *Creating a Project Using the Robotics Project Wizard* section, you created the project for a Windows-based target, the steps you perform in this section differ as noted.

Creating a Build Specification

A build specification defines the VIs, support files, and settings for a stand-alone application. You must create a build specification and then use the build specification to build the application.

Complete the following steps to create a build specification from the project you previously created.

1. Right-click **Build Specifications** under the RT target in the **Project Explorer** window and select **New»Real-Time Application** from the shortcut menu to open the **Real-Time Application Properties** dialog box.

   Note If you created the project for a Windows-based target, right-click **Build Specifications** under **My Computer** and select **New»Application** from the shortcut menu.

2. Enter `Robotics Application` in the **Build specification name** text box. The build specification name appears under **Build Specifications** for the RT target.
3. Enter `roboapp.rtexe` in the **Target filename** text box to set the file name for the application executable on the RT target. By default, the **Target filename** text box contains `startup.rtexe`.

   Note If you created the project for a Windows-based target, enter `roboapp.exe` in the **Target filename** text box.

4. Click **Source Files** from the **Category** list to display the **Source Files** page.
5. From the **Project Files** tree, select the `Single Control Loop (Simulated).vi`, located under the RT target, and click the right arrow button to move the VI into the **Startup VIs** list, as shown in Figure 10. VIs in the **Startup VIs** list execute when you launch the application.

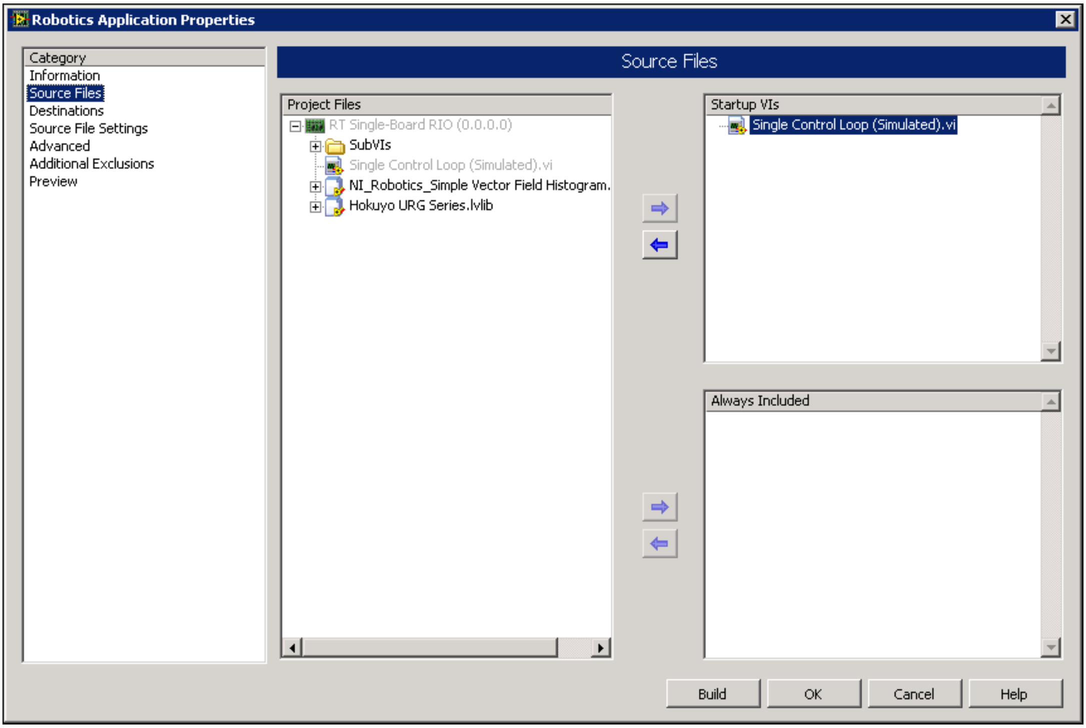


Figure 10. Selecting a Startup VI

6. Click the **OK** button to accept the settings and close the **Real-Time Application Properties** dialog box. The build specification name appears in the **Project Explorer** window under **Build Specifications** for the RT target, as shown in Figure 11.

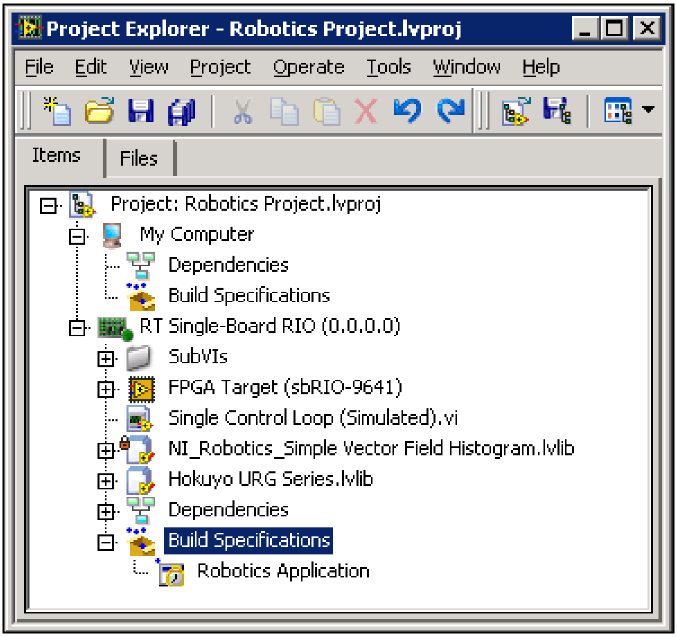


Figure 11. Creating a Build Specification

7. Select **Project»Save Project** from the **Project Explorer** window to save the project and the build specification settings.

Building and Running a Startup Application

After you create a build specification, you can build and run an application on an RT target. You also can set the application to launch when you power on the RT target.

Complete the following steps to build and run an application on an RT target.

1. Right-click the **Robotics Application** build specification and select **Build** from the shortcut menu to build the application. The Application Builder builds the application using with the settings you defined in the build specification. When the build completes, click the **Done** button to close the **Build Status** dialog box.
2. Right-click the build specification and select **Run as startup** from the shortcut menu. The **Run as startup** shortcut menu item sets the application as the startup application, deploys the application to the target, and prompts you to restart the RT target. When you set a build specification as the startup application, LabVIEW displays the build specification with a green border around the specification icon in the **Project Explorer** window.

> **Note** If you created the project for a Windows-based target, right-click the build specification and select **Run** from the shortcut menu. The application launches and displays histogram data that represents the distances to objects in range of the sensor, arranged by angle and direction. You do not need to complete step 3.

3. After you restart the RT target, the application generated from the build specification launches automatically and begins running the simulated navigation program.

Note If you plan to continue editing the RT VI, you might not want the application to run every time you power on the RT target. To unconfigure the application as startup, right-click the build specification and select **Unset as startup** and then **Deploy** from the shortcut menu.

Where to Go from Here

Refer to the following resources for additional information about the LabVIEW Robotics Module.

Robotics Module Examples

Use the NI Example Finder, available in LabVIEW by selecting **Help»Find Examples**, to browse or search for example VIs. The **Robotics** category opens automatically. Refer to the National Instruments Web site at `ni.com/info` and enter the info code `ex4evw` for additional robotics examples.

LabVIEW Help

The **Robotics Module»Robotics VIs** book on the **Contents** tab of the *LabVIEW Help* includes reference information about built-in Robotics Module VIs you can use to develop algorithms and applications. Refer to the Robotics Module VI reference for information about how the VIs work.

The *LabVIEW Help* also contains the following resources for LabVIEW modules that are useful in developing robotics applications:

- The **Real-Time Module** book on the **Contents** tab of the *LabVIEW Help* includes conceptual information about real-time programming techniques and application architectures. Refer to the Real-Time Module help for information about creating a deterministic real-time application.
- The **FPGA Module»Getting Started with the FPGA Module** topic on the **Contents** tab of the *LabVIEW Help* includes links to topics on LabVIEW, FPGA Module, and hardware concepts you should be familiar with before you begin working with the FPGA Module.
- The **Control Design and Simulation Module** book on the **Contents** tab of the *LabVIEW Help* includes information about using LabVIEW to design, analyze, and deploy controllers for dynamic systems.

Sensor Drivers on the Instrument Driver Network

Refer to the National Instruments Web site at `ni.com/info` and enter the info code `ex3mbp` to find and download sensor drivers. You also can use the use the NI Instrument Driver Finder to find sensor drivers. Select **Tools»Instrumentation»Find Instrument Drivers** or **Help»Find Instrument Drivers** to launch the Instrument Driver Finder. You must have Internet access to use the Instrument Driver Finder.