Getting Started with the LabVIEW™ Robotics Module

Version 2011

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.

NATIONAL INSTRUMENTS™

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 sensor drivers**—Robotics Module VIs provide algorithms for navigation, steering, and so on. The Robotics Module also provides drivers that allow you to acquire data from certain sensors and actuators commonly used in robotics systems.

  To access the Robotics VIs and most sensor drivers, select **View»Functions Palette** from the block diagram in LabVIEW and navigate to the Robotics palette. The items on the Robotics palette are divided into subpalettes according to the functionality they provide, such as steering, simulation, and communicating with sensors and actuators.

  Note The Robotics Module also includes drivers that provide connectivity to sensors on FPGA targets. To browse or search for these drivers, use the NI Example Finder, available by selecting **Help»Find Examples** in LabVIEW. The **Robotics»Sensor Drivers** category contains these drivers.

- **Setup wizards**—The Robotics Module provides the following wizards:
  - **Hardware Setup Wizard**—This wizard configure the basic settings of NI-RIO hardware targets and installs software on the targets. Select **View»Getting Started Window** and click the **Hardware Setup Wizard** link in the **Getting Started** section of the **Getting Started** window to launch this wizard.
  - **Robotics Project Wizard**—This wizard allows you to create a new LabVIEW project configured with your hardware or a simulation project. Refer to the *Creating a Robotics Project* section for information about using the Robotics Project Wizard. Click the **Robotics Project** link in the **Create New** section of the **Getting Started** window to launch the Robotics Project Wizard.
- **Simulator**—A built-in robotics simulator renders 3D simulation scenes and models the motion of robots in a real-world environment. Refer to the **Robotics Module»Simulating Mobile Robots** topic on the **Contents** tab of the *LabVIEW Help* for more information about simulating robots.
- **Examples**—Example VIs demonstrate robotics concepts, such as path planning, kinematics, and acquiring data from sensors. 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 **Information** 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 sensor and actuator drivers in the `labview\instr.lib\`*device* directory, where *device* is the name of a device for which you have drivers installed.

Accessing LabVIEW Robotics Software

The Robotics Module provides a different environment than the LabVIEW Development System to allow you to easily access robotics-specific features and focus on developing robotics applications. To access the LabVIEW Robotics environment, select **Start»All Programs»NI LabVIEW Robotics *x***, where *x* is the LabVIEW version number.

If LabVIEW is already open, select **Tools»Choose Environment** to display the **Choose Environment Settings** dialog box, which you can use to change environments.

Figure 1 shows the **Getting Started** window in the LabVIEW Robotics environment.

Figure 1. Getting Started Window

Related National Instruments Software

Depending on the LabVIEW Robotics package you purchase, you might 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, this manual contains an exercise that requires the LabVIEW Real-Time and LabVIEW FPGA modules, which allow you to use LabVIEW graphical programming to build applications that run on real-time operating systems or NI FPGA targets, respectively.

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

Sensor and Actuator Drivers

The Robotics Module includes drivers for a variety of programmable sensors and actuators. You can access information about many drivers, such as the device models the drivers support, in readme files

located in the `labview\instr.lib\`*device* directory, where *device* is the name of a sensor or actuator for which you have drivers installed.

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 use the Robotics Module.

• *LabVIEW Help*—Access information about LabVIEW programming concepts, step-by-step instructions for using LabVIEW, and reference information about LabVIEW VIs, functions, tools, and so on. Select **Help»Search the LabVIEW Help** in LabVIEW to display the *LabVIEW Help*. 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.

> **Note** The **Contents** tab of the *LabVIEW Help* contains the **Robotics Module** book, which provides information about developing robotics applications.

• **Manuals**—User manuals and other documents, such as the *Getting Started with LabVIEW* manual, are available as PDFs when you install LabVIEW. Access these documents by navigating to the `labview\manuals` directory.

• **Product Readmes**—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 and readmes for other products by navigating to the `labview\readme` directory.

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.

• **Robot control system**—Construct a robot steering frame that includes the following components:

 – **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 robot environment with sensors, such as laser range finders, and control robot motion with actuators.

The LabVIEW Robotics Starter Kit is an example of a robotics system that consists of these components and runs LabVIEW and Robotics Module software. On the Starter Kit robot, a Single-Board RIO controller, mounted on top of the robot base, includes a real-time processor and Field-Programmable Gate Array (FPGA) on a single board. By default, the FPGA has a sensor and motors wired to digital I/O: an ultrasonic distance sensor, a servo motor on which the sensor is mounted, and two DC drive motors.

> **Note** Refer to the **Robotics Module»LabVIEW Robotics Starter Kit** book on the **Contents** tab of the *LabVIEW Help* for more information about the Robotics Starter Kit.
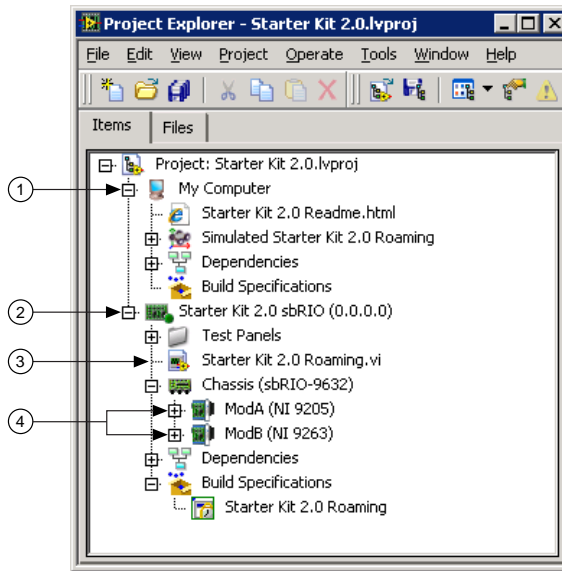
The remainder of this manual uses the Robotics Starter Kit as an example to explain the process of getting started with a typical robotics system and the Robotics Module.

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 (3) and applications to RT (2) and FPGA targets that contain I/O modules (4). 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 2 shows the parts of a typical LabVIEW project.

1.	Host computer
2.	Real-time controller
3.	RT VI
4.	I/O modules

Figure 2. 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 Starter Kit 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 run programs you create in LabVIEW. On the Starter Kit robot, the Single-Board RIO is the real-time controller. Some real-time controllers, such as NI 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. Use the **Project Explorer** window to configure third-party hardware similarly to how you configure 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. 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 configure to match the requirements of a specific system. 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. FPGAs automatically communicate with I/O modules and provides deterministic I/O to the real-time processor. Thus, you typically program the FPGA to perform the lowest-level I/O and control tasks with the most critical level of timing in a robotics application.

On the Starter Kit robot, you might want to program the FPGA to read obstacle data from the distance sensor and adjust the drive motors accordingly to run deterministically so the robot avoids obstacles. You can use LabVIEW programming techniques to develop an FPGA VI on the host computer and download the VI on the target.

I/O Modules

National Instruments offers a variety of I/O modules that directly connect to sensors and actuators. On the Starter Kit robot, built-in I/O connects the FPGA to the sensor and actuators. You typically write an FPGA VI to access I/O, and then use an RT VI to read from and write to the I/O in the FPGA VI.

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. On the Starter Kit robot, the distance sensor detects obstacles in the path of the robot, a servo pans the sensor back and forth, and drive motors propel the robot.

Use sensor and actuator drivers from the Robotics palette in an RT or FPGA VI to communicate with sensors and actuators like the ones on the Starter Kit robot. Drivers reduce program development time by eliminating the need to learn the programming protocol for each device. Refer to the *Where to Go from Here* section of this manual for information about finding sensor drivers.

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 for the Starter Kit robot, a typical RIO-based LabVIEW robotics system. You can extend the information in this exercise to other, similar robotics systems.

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 installation media 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 Robotics Starter Kit or Robotics Software Bundle, you can use the Robotics Module with only the LabVIEW Development System to develop programs that include VIs and 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 installation media.

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. To configure the Single-Board RIO on the Starter Kit robot, use the National Instruments Robotics Hardware Setup Wizard, which launches after you install the LabVIEW Robotics software on the host computer. You also can click the **Hardware Setup Wizard** link in the **Getting Started** section of the **Getting Started** window to launch this wizard manually.

If you use a CompactRIO or non-Starter Kit Single-Board RIO, you also can use this wizard. However, if you use any other 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 and refer to the MAX help for more information about configuring targets in MAX.

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

- Set the IP address of the target you want to use
- Download software to the target that can run VIs you deploy 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 target 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.

On the final page in the Robotics Hardware Setup Wizard, exit the wizard and launch LabVIEW to continue this exercise. When LabVIEW launches, the Robotics Project Wizard displays.

> **Tip** If you do not have the Robotics Starter Kit, you can 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 (`.lvproj`) 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 build specifications, and deploy files to targets.

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

For this exercise you will use the Robotics Project Wizard to create an application that drives the Starter Kit robot. The application consists of a VI that acquires data from the distance sensor, calculates a heading for travel, and then applies the correct amount of velocity to the drive motors.

> **Note** You can complete some of the same the steps in this exercise even if you do not have the Starter Kit. The process this exercise describes is similar to the process for creating an application for a non-Starter Kit RT target or a Windows-based target.

Creating a Starter Kit Project Using the Robotics Project Wizard

Complete the following steps to create a project for the Starter Kit robot 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**.

2. On the **Select project type** page, select the type of project that corresponds to your system from the **Select project type** list, as shown in Figure 3. For this exercise, select **Robotics Starter Kit 2.0**.
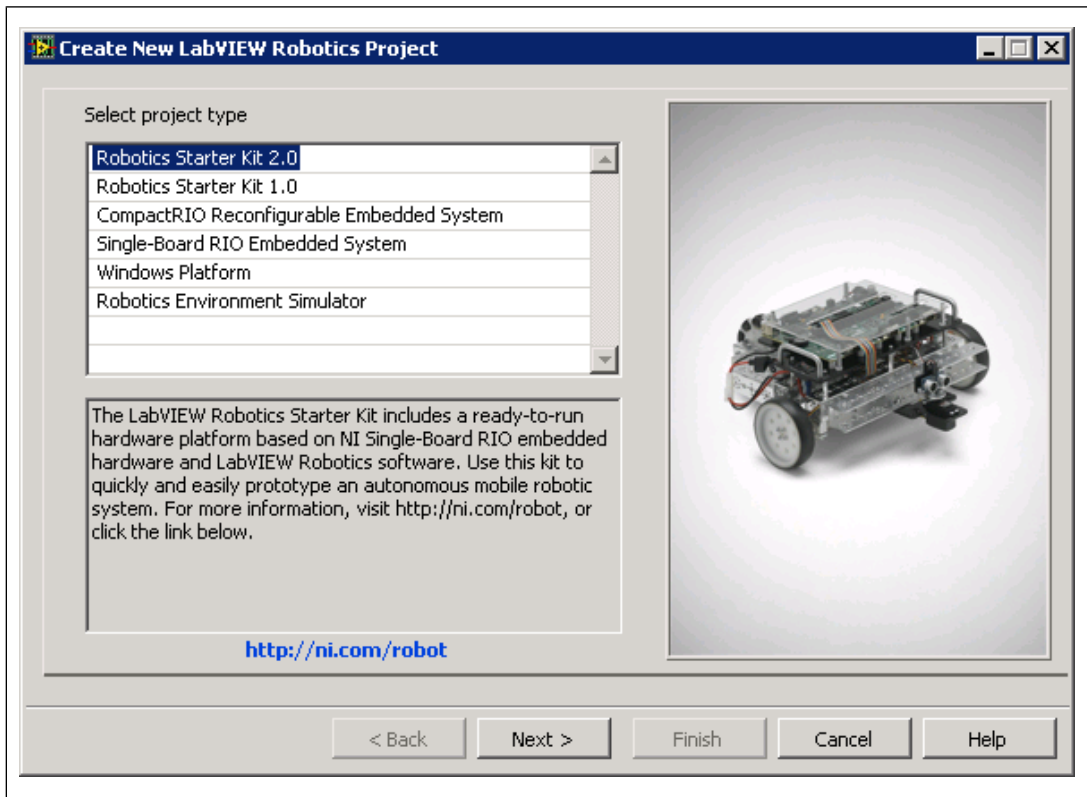

Figure 3. Selecting a Project Type for Your Application

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

4. On the **Enter the IP address of the controller** page, type the IP address of the RT target from the Starter Kit in the **Controller IP address** text box.

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

6. On the **Enter project name and folder** page, enter a name for the project and the location at which you want to save the .lvproj file.

7. 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 hardware targets, VIs, and other support files from one location. In the project you create, the **Project Explorer** window includes the following sections:

- **Project root**—Contains the host computer, the Starter Kit RT target and I/O, and VIs that control the target.
  - **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. You can use **Build Specifications** to configure stand-alone applications, such as one that launches automatically when you power on the target.
  - **Starter Kit 2.0 sbRIO**—Represents the real-time controller on the Starter Kit robot and contains the I/O through which you access sensors and actuators. The **Project Explorer** window also includes **Dependencies** and **Build Specifications** sections under this target.

VIs and libraries that you add to a hardware target appear under the target to indicate that you can run the VI on that target. Figure 4 shows a project similar to the `Robotics Project.lvproj` file you created with the Robotics Project Wizard.
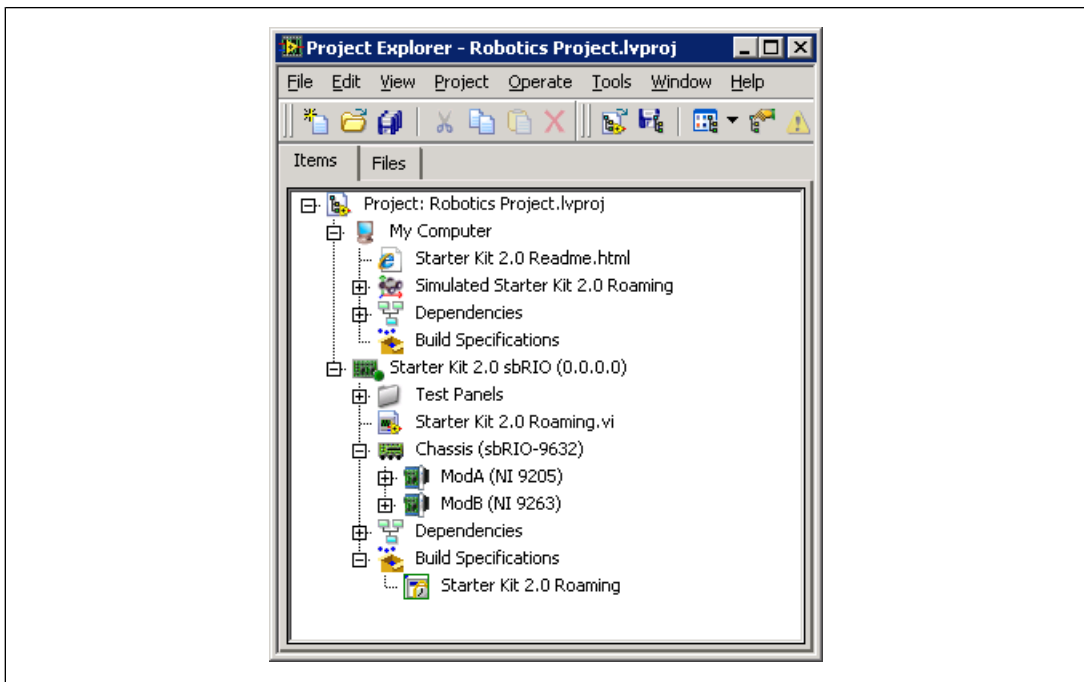

Figure 4. Exploring the Robotics Project in the Project Explorer Window

Exploring VIs that Control the Starter Kit Robot

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

1. In the **Project Explorer** window, expand the **Test Panels** folder and double-click `Test Ultrasonic Sensor.vi` in the **Project Explorer** window. `Test Ultrasonic Sensor.vi` is an RT VI that runs on the Single-Board RIO.

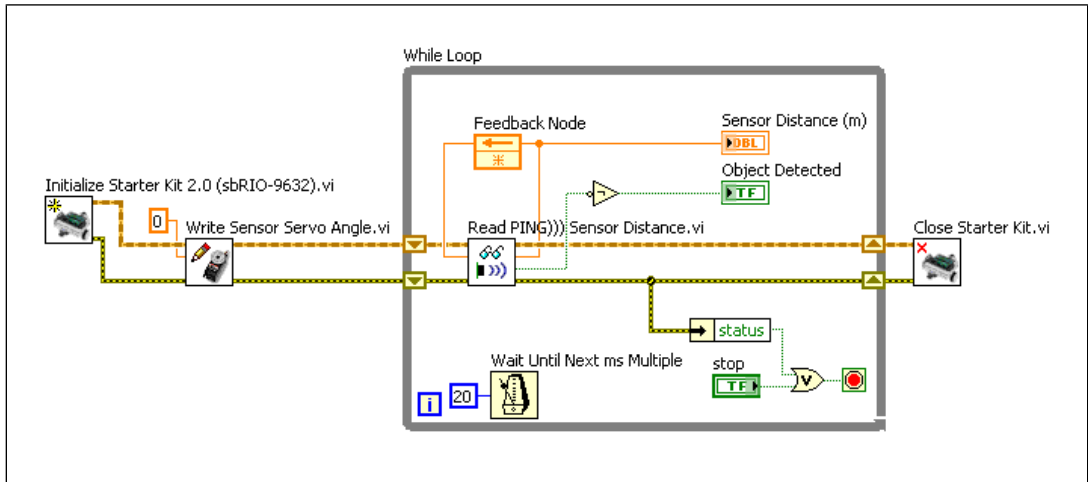2. Select **Window»Show Block Diagram** to display the block diagram of the VI, as shown in Figure 5.

Figure 5. Block Diagram of the Test Ultrasonic Sensor VI

On the previous block diagram, the following VIs from the Starter Kit palette allow you to communicate with the sensor and actuators connected to I/O on the Starter Kit robot. Although this VI is fairly simple, the block diagram illustrates a common workflow for controlling RIO-based mobile robots.

*   The Initialize Starter Kit 2.0 (sbRIO-9632) VI begins a communication session with the FPGA on the robot and returns a reference you use to read from or write to the FPGA.
*   The Write Sensor Servo Angle VI writes an angle to the servo on which the distance sensor is mounted to turn the servo to that angle.
*   The Read PING))) Sensor Distance VI reads and returns the distance between the ultrasonic distance sensor and the nearest obstacle the sensor detects.
*   The Close Starter Kit VI terminates the communication session with the FPGA on the Starter Kit robot.

> 💡 **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.

These VIs and others from the Starter Kit palette allow you to control the Starter Kit FPGA from an RT VI without writing an FPGA VI. To control a non-Starter Kit robot whose real-time controller contains an FPGA target, you typically must write an FPGA VI that runs on the real-time controller and accesses I/O. Figure 6 shows the block diagram for a simple RT VI that communicates with an FPGA VI to perform the same task as the Test Ultrasonic Sensor VI.
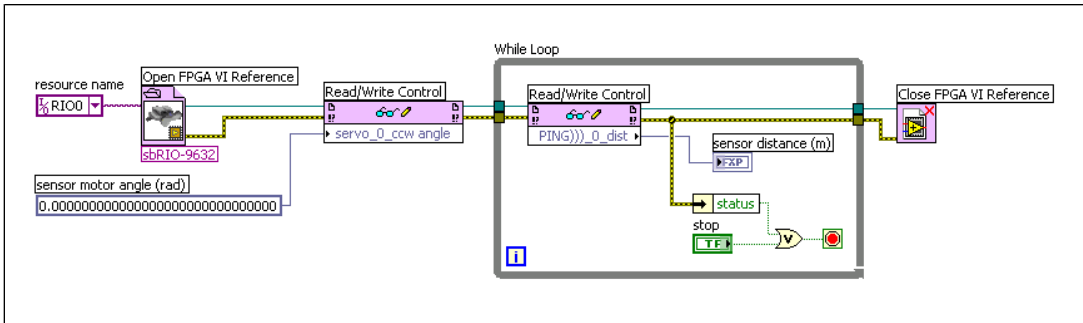
Figure 6. Block Diagram of a VI that Communicates with an FPGA VI

Refer to the **Robotics Module»LabVIEW Robotics Starter Kit»Programming a Starter Kit Robot** topic on the **Contents** tab of the *LabVIEW Help* for more information about using traditional FPGA programming versus the VIs on the Starter Kit palette.

Return to the block diagram for the Test Ultrasonic Sensors VI and complete the following steps to continue exploring the VI.

1. Select **Window»Show Front Panel** to display the front panel of the VI, as shown in Figure 7.
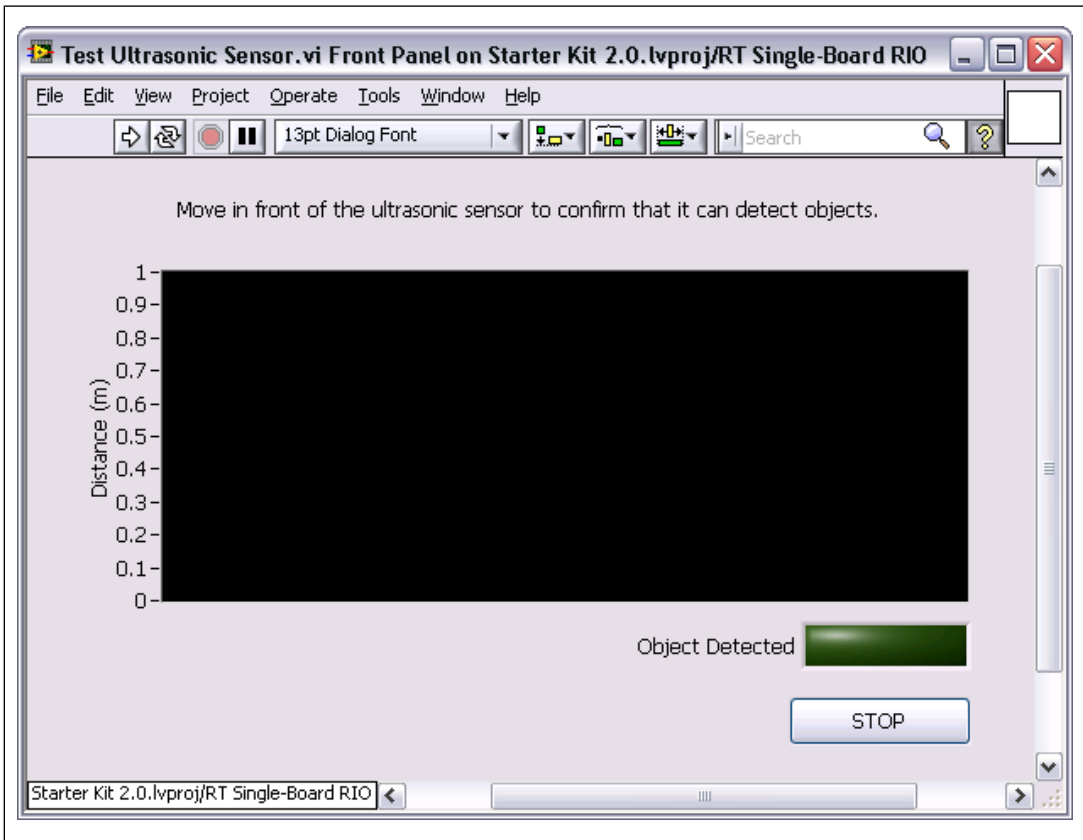

Figure 7. Front Panel of the Test Ultrasonic Sensor VI

When the VI runs, the graph indicator displays data that represents the distances to objects in range of the sensor. Also, the **Object Detected** indicator is TRUE when the sensor detects an object in range.

2. 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.

3. Close the front panel of the VI. The block diagram closes automatically.

4. In the **Project Explorer** window, double-click the `Starter Kit 2.0 Roaming.vi`.

5. Select **Window»Show Block Diagram** to display the block diagram of the VI, as shown in Figure 8.
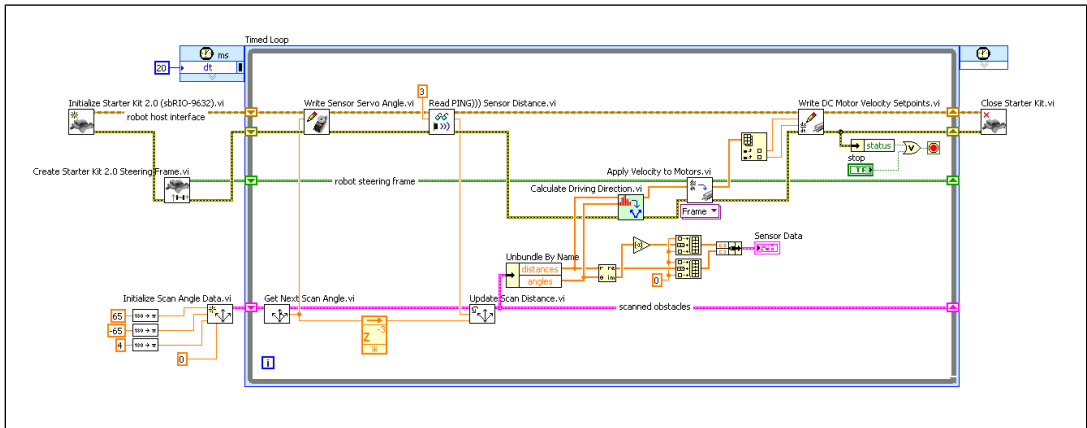

Figure 8. Block Diagram of the Starter Kit Roaming 2.0 VI

This VI shows a more complex Starter Kit application. The Timed Loop controls the code that acquires obstacle data from the distance sensor, calculates the heading to which the vehicle drives, and applies the correct velocity to the drive motors.

6. Close the Starter Kit 2.0 Roaming VI and return to the project.

Configuring Target Properties

Use the **Properties** dialog box to configure the project settings of the Single-Board RIO 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 **Starter Kit 2.0 sbRIO** 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.

Overview of Robotics Application Architectures

The Starter Kit 2.0 Roaming VI serves as an example of a robot that performs simple, repetitive algorithms. You can insert code for acquiring sensor data, processing data, and controlling the robot inside the Timed Loop.

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 between processes on different targets as well as between different processes that run on the same target.

Refer to the **Robotics Module»Communicating between Processes in a Robotics Application** book on the **Contents** tab of the *LabVIEW Help* for more information about sending data between VIs.

Note Select **Help»Find Examples** in LabVIEW to launch the NI Example Finder, where you can find specific example architectures. The **Robotics** category opens automatically.

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.

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 created with the Robotics Project Wizard.

1. Right-click the `Starter Kit 2.0 Roaming.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 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 Robotics Applications»Deploying, Downloading, and Running VIs and Applications** topic on the **Contents** tab of the *LabVIEW Help* to access 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.

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.
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`.
4. Click **Source Files** from the **Category** list to display the **Source Files** page.
5. From the **Project Files** tree, select the `Starter Kit 2.0 Roaming.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 9. VIs in the **Startup VIs** list execute when you launch the application.
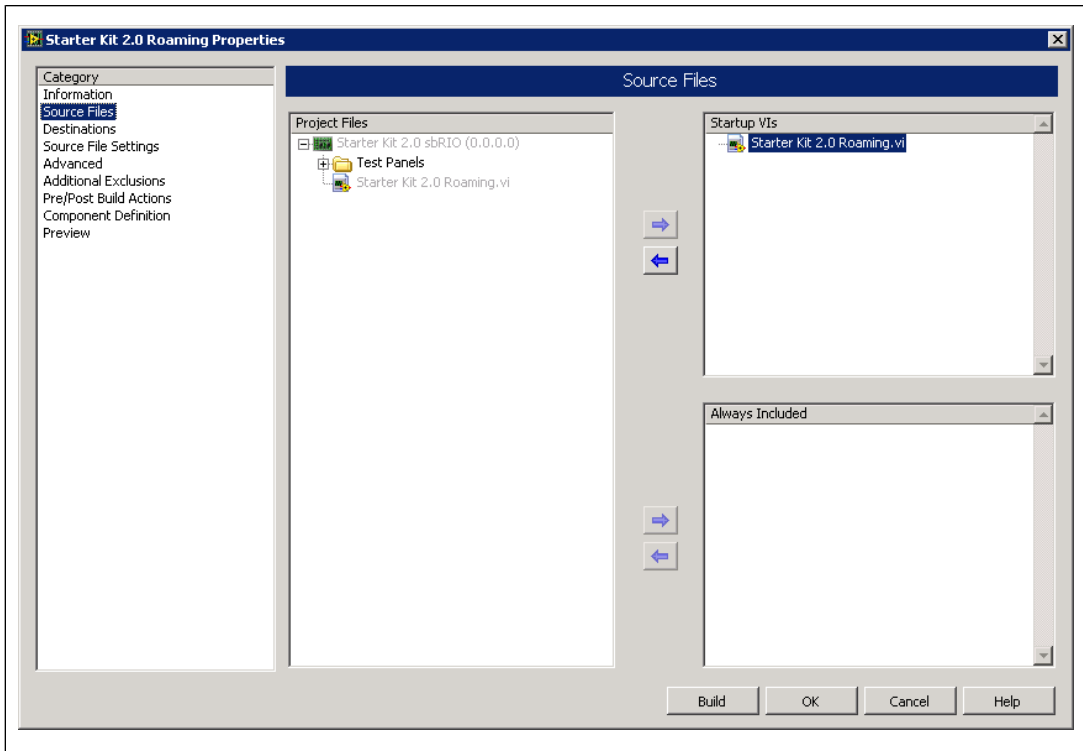
Figure 9. Selecting a Startup VI

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.

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.

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, including more examples for the Robotics Starter Kit.

LabVIEW Help

The **Robotics Module** book on the **Contents** tab of the *LabVIEW Help* includes procedural and conceptual information about developing robotics applications and reference information about built-in VIs on the Robotics palette.

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

National Instruments works with major sensor vendors to provide a library of ready-to-use drivers. 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.

375286B-01   June11