WiRobot™

The Heart of Robot

# WiRobot Getting Start Guide

Dr Robot®

Version: 1.0.4
Feb. 2004

## Table of Contents

# I. Preface

## I.1. Audience

This document is written for robot developers in using WiRobot systems. It provides the initial product information as well as a guide in helping users to understand how to use this system. The developers should have basic knowledge in Microsoft Visual C++ or VB. Detail programming information can be found in WiRobot SDK API Reference Manual and relevant WiRobot User Manuals.

## I.2. Feedback

If you find any problems in this document, please send us your feedback to support@drrobot.com.

## II. WiRobot Overview

WiRobot™ is an integrated electronic and software robotic system extended from Dr Robot's comprehensive humanoid robot, which has demonstrated its interactive capabilities in the public and the media. Each WiRobot development system is designed to provide a user-friendly programming environment for hobbyists, students in robotic areas and researchers to develop their robot programs and applications at an affordable cost.

The power of WiRobot mobile robot system comes from the Dr Robot's Distributed Computation Robotic Architecture and System (DIRAS) technology, which offloads most of the computation and storage intensive tasks to a home PC. Through a digital wireless connection supporting over 100kbps data communication rate, user programs running on PC are virtually connected directly to the WiRobot development system. Data such as image, audio, sensor information, and etc. are available to the user through a set of ActiveX control components (SDK) developed for MS VC++ and VB programming environment. Multiple PC programs are also allowed to access the data information obtained from the sensors simultaneously. High level schemes such as tele-operation, navigation, reasoning, learning, recognition, and image processing routines are programmed and executed on the PC remotely. Multi-robot coordination is also feasible for applications like soccer game. Microsoft Visual Studio programming environment is chosen as the development platform due to its popularity and ease-of-use than the non-user friendly embedded programming interface. Note that communication protocol for WiRobot system is also available for developers/researchers who prefer to use different platform or operating systems to communicate and control the WiRobot system. But this document focuses on how to use the WiRobot system using the SDK under Microsoft platform.

As well, WiRobot system already comes with low-level drivers for all its electronic modules and can provide a flexible way for users to control the robot. For instance, it allows user to control standard servo motors and DC motors by using the built-in commands available in the control command library, which offers several types of DC motor control method including open-loop PWM, closed-loop position control, closed-loop velocity control, and closed-loop current control. Control parameters are also configurable.

## II.1. Software Components

The WiRobot system comes with a CD containing the following software components and documents:

- "WiRobot Gateway" which is used to connect the PC to the robot and show the connection status,
- An ActiveX control, called WiRobot SDK ActiveX Module, with a set of APIs is provided for user to access the robot when developing his/her own applications in MS VC++ or VB,
- Several PC sample applications with source code is provided to demonstrate the capabilities of the WiRobot system, and
- WiRobot documents. (The latest documents can be found on www.drrobot.com)

In the WiRobot system, low level electronic drivers are pre-programmed and embedded in the WiRobot controllers (PMS5005 and PMB5010). Data information such as image, audio, sensor information, and etc. are available to the user via the WiRobot ActiveX control developed for MS VC++ and VB program environment or by using the WiRobot communication protocol. Using this ActiveX control, user can also send various control commands to the robot. A general connection architecture of the WiRobot system is shown as follows.

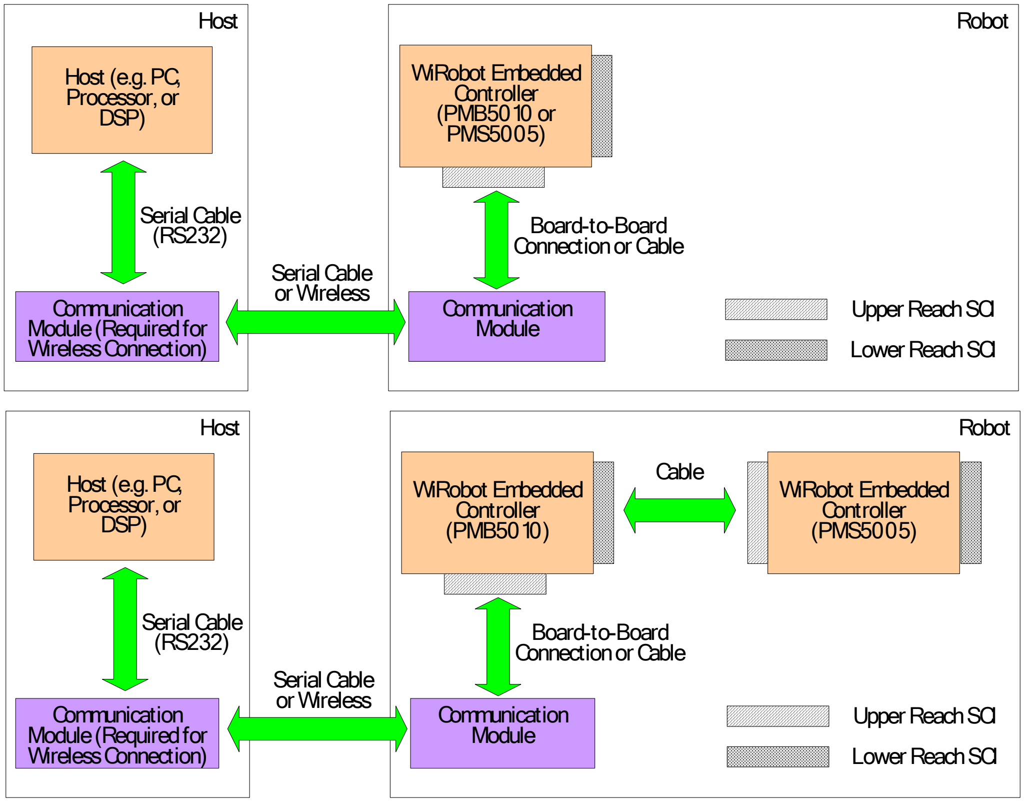

Host
Host (e.g. PC, Processor, or DSP)
Serial Cable (RS232)
Communication Module (Required for Wireless Connection)
Serial Cable or Wireless
Robot
WiRobot Embedded Controller (PMB5010 or PMS5005)
Board-to-Board Connection or Cable
Communication Module
Upper Reach SCI
Lower Reach SCI
Host
Host (e.g. PC, Processor, or DSP)
Serial Cable (RS232)
Communication Module (Required for Wireless Connection)
Serial Cable or Wireless
Robot
WiRobot Embedded Controller (PMB5010)
Cable
WiRobot Embedded Controller (PMS5005)
Board-to-Board Connection or Cable
Communication Module
Upper Reach SCI
Lower Reach SCI

Figure II.1 WiRobot System Communication Architecture

# III. Software (WiRobot SDK) Installation

## III.1. System Requirements

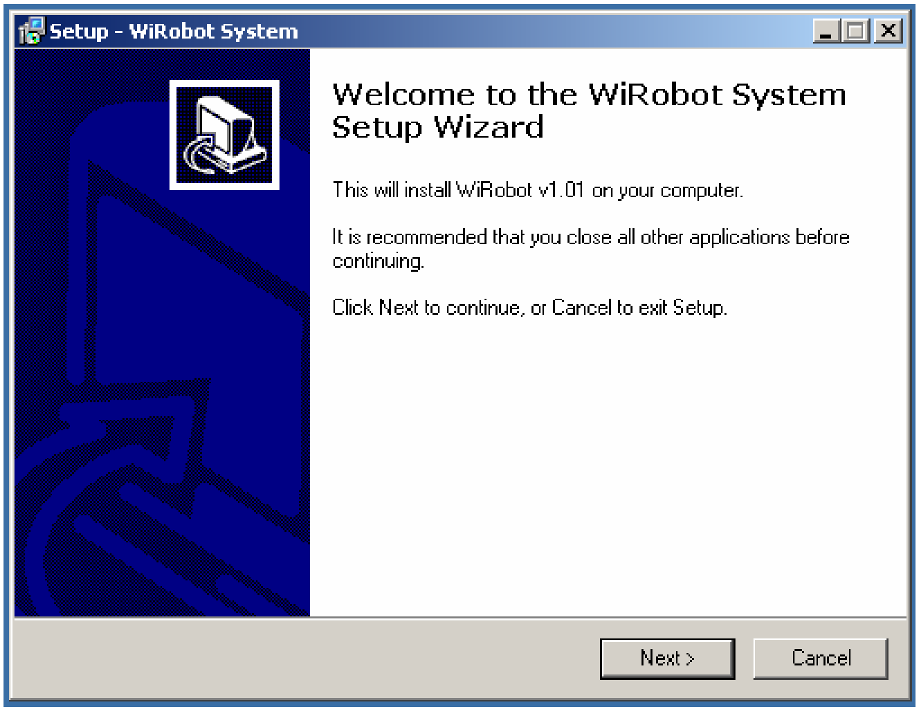
The PC requirements in using the WiRobot system are:

- PIII 550MHz or faster CPU
- 64MB RAM or more
- 20 MB hard disk free space
- Microsoft Windows 2000 Professional or XP Professional operating system

As well, Microsoft Visual VB or VC++ 6.0 (with Service Pack 5) or higher is required for users to develop their own applications.

## III.2. Step-by-Step Installation

User can perform the installation by running the WiRobotSetup.exe that comes with the CD. The followings are the screen shots taken during the installation process.



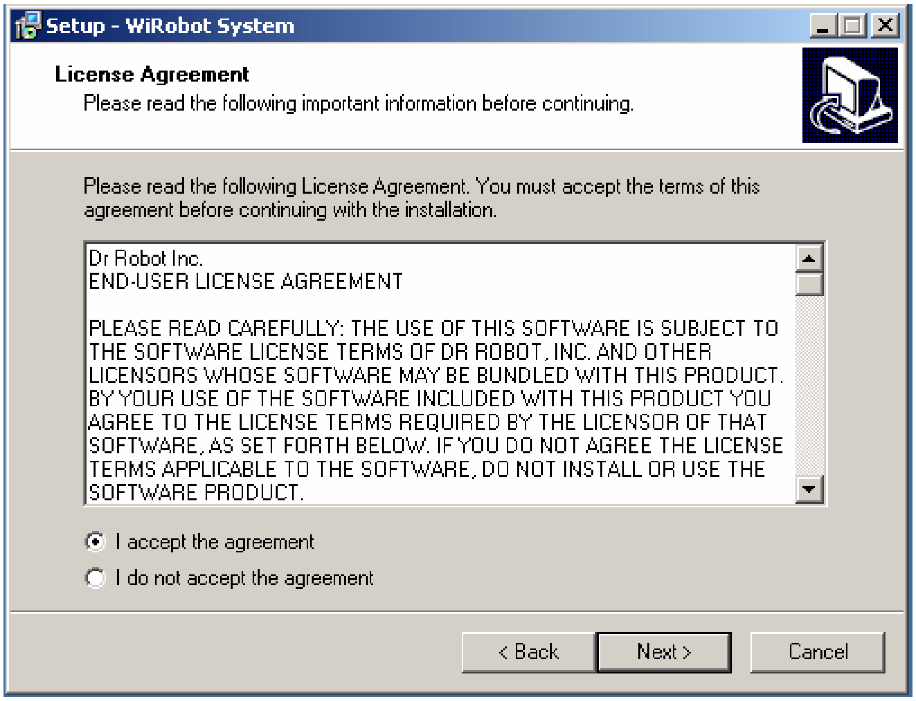
Welcome to the WiRobot System Setup Wizard

This will install WiRobot v1.01 on your computer.

It is recommended that you close all other applications before continuing.

Click Next to continue, or Cancel to exit Setup.

Figure III.1 WiRobot Installation Step 1



License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

Dr Robot Inc.
END-USER LICENSE AGREEMENT

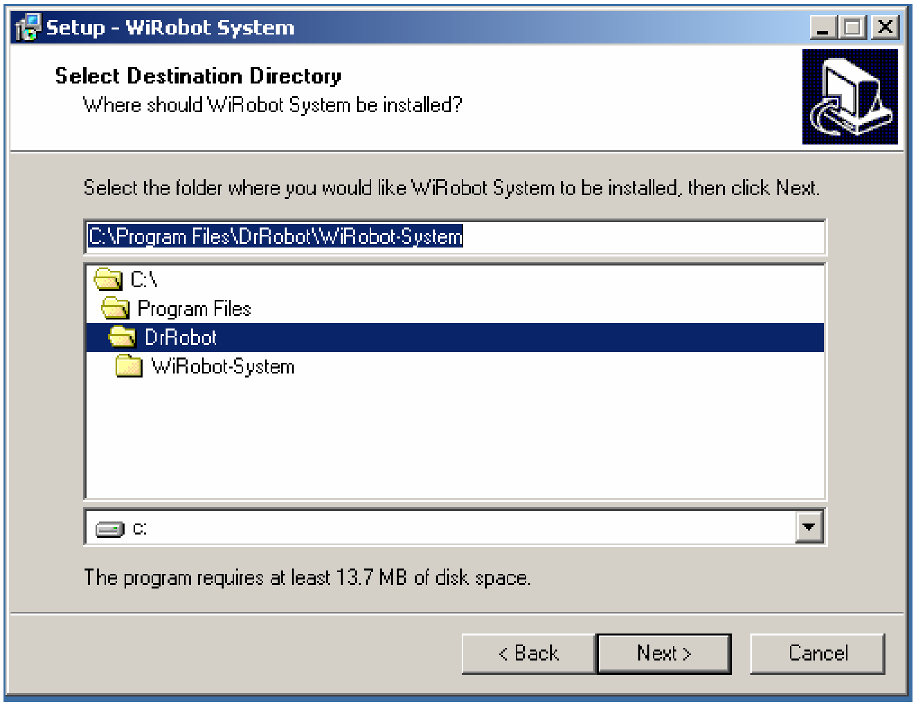
PLEASE READ CAREFULLY: THE USE OF THIS SOFTWARE IS SUBJECT TO THE SOFTWARE LICENSE TERMS OF DR ROBOT, INC. AND OTHER LICENSORS WHOSE SOFTWARE MAY BE BUNDLED WITH THIS PRODUCT. BY YOUR USE OF THE SOFTWARE INCLUDED WITH THIS PRODUCT YOU AGREE TO THE LICENSE TERMS REQUIRED BY THE LICENSOR OF THAT SOFTWARE, AS SET FORTH BELOW. IF YOU DO NOT AGREE THE LICENSE TERMS APPLICABLE TO THE SOFTWARE, DO NOT INSTALL OR USE THE SOFTWARE PRODUCT.

● I accept the agreement

○ I do not accept the agreement

Figure III.2 WiRobot Installation Step 2

By default, all these components will be installed under the directory “C:\Program Files\DrRobot\WiRobot-System” unless user specifies another location during the installation.



Select Destination Directory

Where should WiRobot System be installed?

Select the folder where you would like WiRobot System to be installed, then click Next.

C:\Program Files\DrRobot\WiRobot-System

The program requires at least 13.7 MB of disk space.

Figure III.3 WiRobot Installation Step 3

## IV. Connecting to WiRobot System Using Gateway Software

In order to use the WiRobot system, user needs to install the WiRobot software on the PC and setup the physical connections. If user has purchased the RS232 wireless module, please connect the WiRobot system as follows:

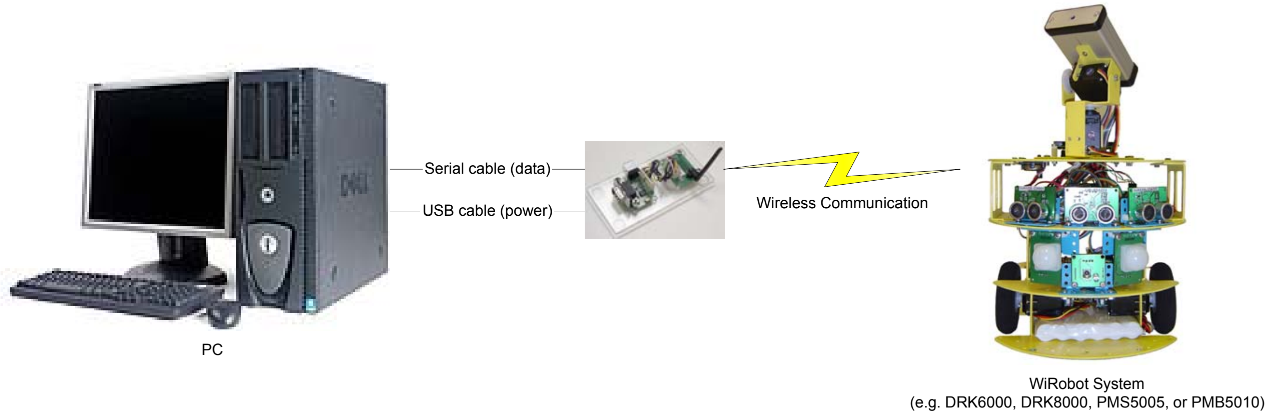


Figure IV.1 WiRobot System Setup with Wireless Module

If the user does not have a RS232 wireless module, he / she can simply connect the WiRobot system to a PC through a null modem cable as follows:

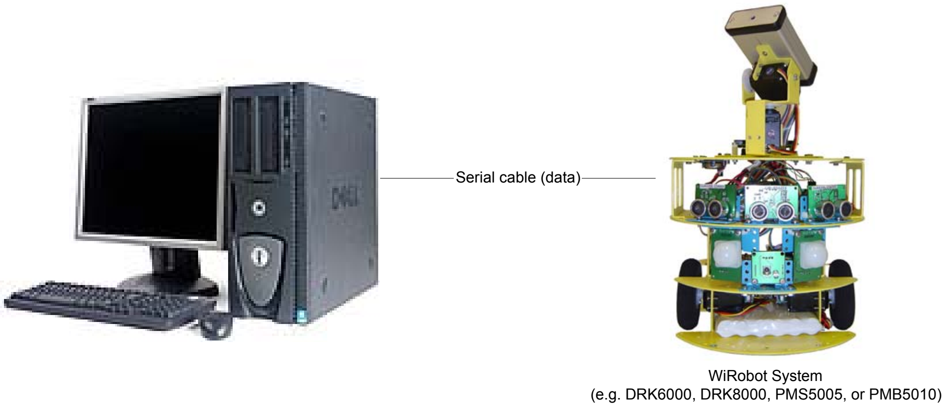
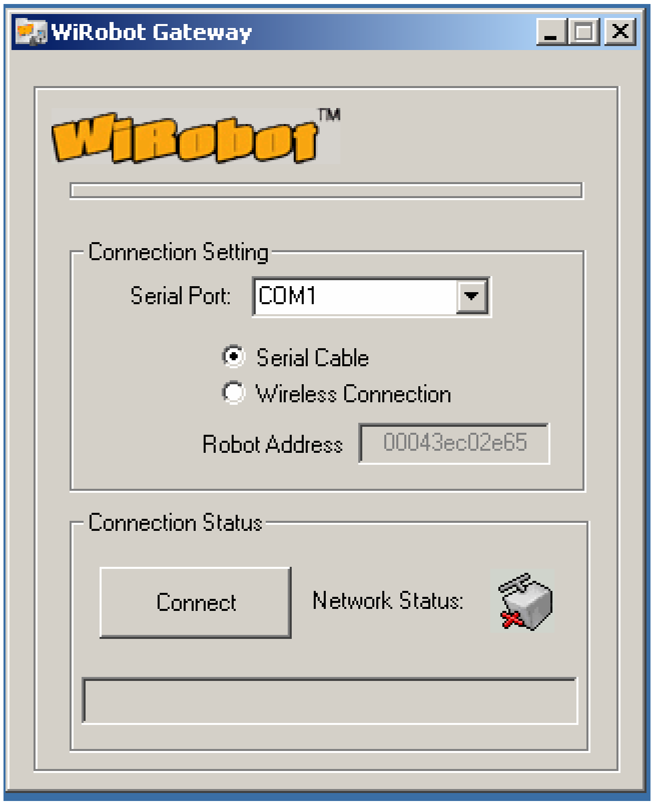


Figure IV.2 WiRobot System Setup without Wireless Connection

User can now plug-in the power source (please refer to the relevant user manual for the power supply requirement), turn on the robot (the WiRobot system) and start the "WiRobot Gateway" software. On the "WiRobot Gateway" interface, user should select the option corresponding to the physical connection:

- Serial Cable: if user uses a null modem serial cable to connect directly to the robot, user should select the "Serial Cable" option on the GUI.
- Wireless connection: if user uses the RS232 wireless module, he/she should select the "Wireless Connection" and type in the wireless address of the robot (The address is labeled at the bottom of the wireless module).



WiRobot Gateway

Connection Setting

Serial Port: COM1

● Serial Cable

○ Wireless Connection

Robot Address 00043ec02e65

Connection Status

Network Status:

Figure IV.3 GUI of WiRobot Gateway on PC

The "WiRobot Gateway" will provide connection status information between the PC and the robot. This program is required to keep running as long as the user wants to access and control the robot through the sample applications or their custom programs. Robot data acquisitions including human sensor, ultrasonic sensor, Infrared distance sensor, tilting sensor, potentiometer, color CMOS image sensor, microphone, and etc. and motion control can be done by making function calls offered by the "WiRobot SDK ActiveX Module". Details of this ActiveX control component can be found in the WiRobot SDK API Reference Manual.

After the connection is established between the robot and the PC, user can start to use the WiRobot system by running the sample applications offered in the WiRobot software package.

# V. Building PC Applications Using SDK

This section will discuss how to program user's applications. Several sample applications with source code are provided to help user kick start in using the WiRobot system. All these source code will be stored in the "SampleApps" folder under the WiRobot SDK installation location.

## V.1. Using WiRobot SDK Component ActiveX Control

When user starts to write an application, he/she first adds the WiRobot SDK Component in your VB or VC++ project.  The ActiveX object is installed during the installation process and the following is a step to step guideline showing how to incorporate the ActiveX Control into a VB 6.0 project:

- Create a new VB project
- Click "Project" in the menu and choose the "Components"

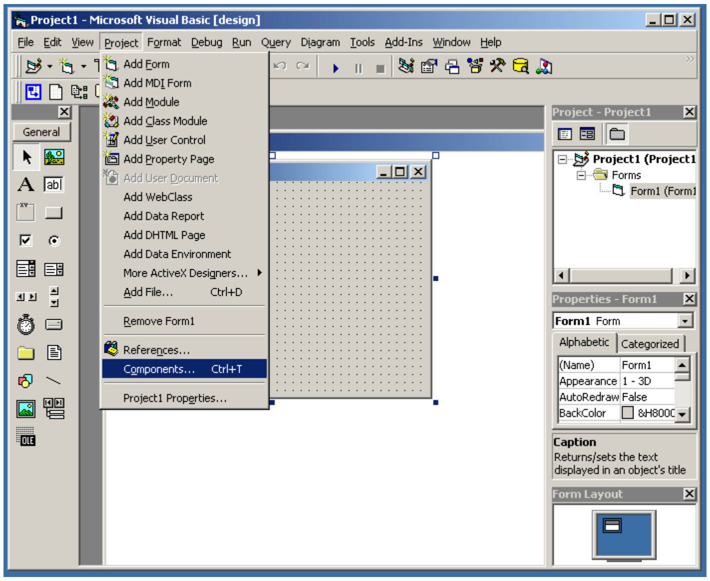
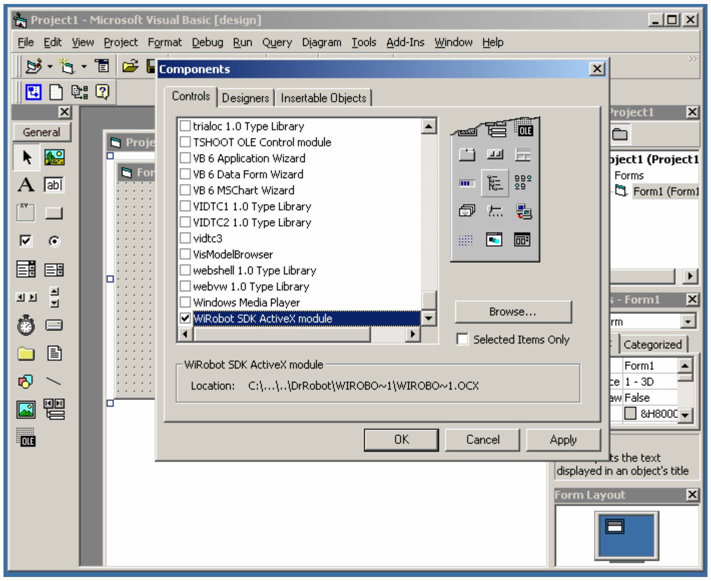


Figure V.1 Using ActiveX module under VB Step 1

- Uncheck the "Selected Items Only" box to show all components, choose the "WiRobot SDK ActiveX Module" and click "OK"



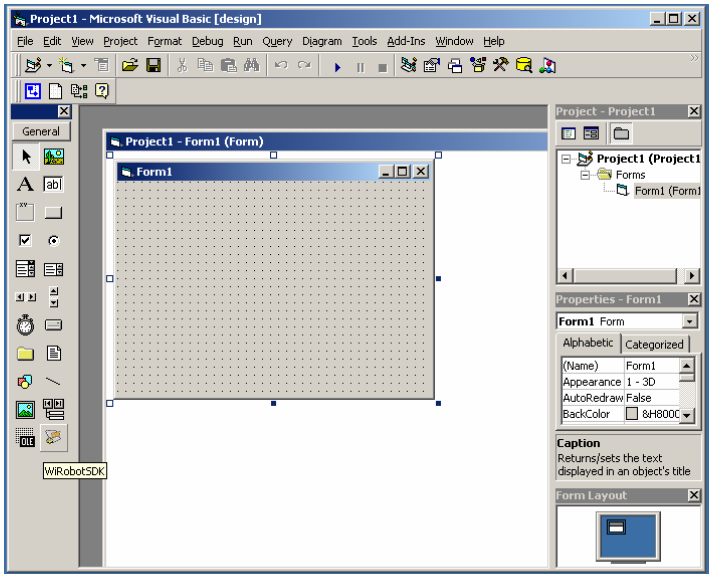
□ trialoc 1.0 Type Library
□ TSHOOT OLE Control module
□ VB 6 Application Wizard
□ VB 6 Data Form Wizard
□ VB 6 MSChart Wizard
□ VIDTC1 1.0 Type Library
□ VIDTC2 1.0 Type Library
□ vidtc3
□ VisModelBrowser
□ webshell 1.0 Type Library
□ webvw 1.0 Type Library
□ Windows Media Player
☑ WiRobot SDK ActiveX module

□ Selected Items Only

WiRobot SDK ActiveX module
Location: C:\...\..\DrRobot\WIROBO~1\WIROBO~1.OCX

Figure V.2 Using ActiveX module under VB Step 2

- A new icon on the left menu bar will appear and user can simply drag and drop this icon to the Project's Form and start using the APIs offered by this ActiveX control. By default the variable name of this component is "WiRobotSDK1".



Form1 Form

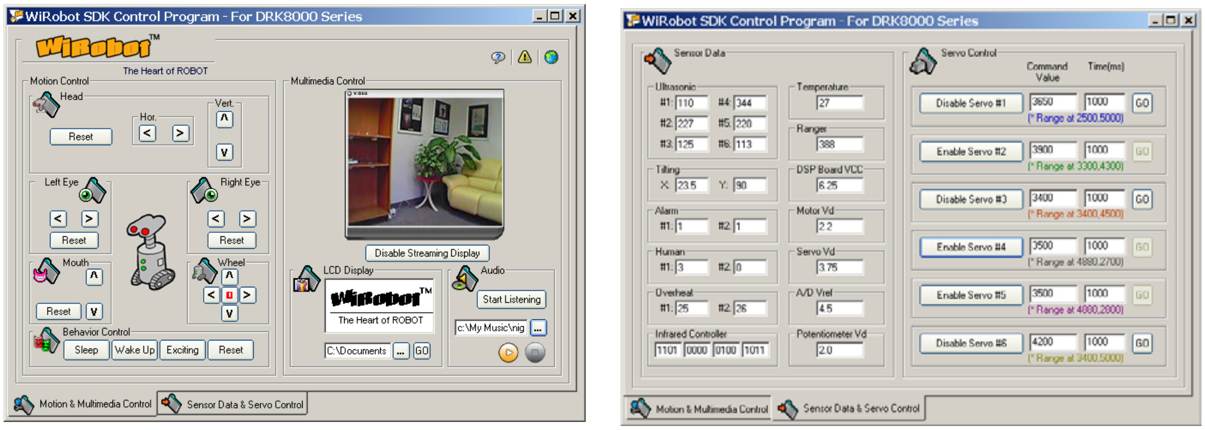
| | |
|---|---|
| (Name) | Form1 |
| Appearance | 1 - 3D |
| AutoRedraw | False |
| BackColor | &H8000 |

Caption
Returns/sets the text displayed in an object's title

Figure V.3 Using ActiveX module under VB Step 3

## V.2. Sample Application 1 - WiRobot DRK8000 Controller (VB)

This sample application demonstrates the basic capabilities of the WiRobot DRK8000 using Microsoft VB with source code provided. This program can read sensor data, obtain image and audio, play wave file, set the LCD display image, and control the robot movement. The GUI of this program is shown as follows:



LCD Display: C:\Documents
Audio: c:\My Music\nig

Sensor Data

Ultrasonic #1: 110 #2: 227 #3: 125 #4: 344 #5: 220 #6: 113

Tilting X: 23.5 Y: 90

Alarm #1: 1 #2: 1

Human #1: 3 #2: 0

Overheat #1: 25 #2: 26

Infrared Controller 1101 0000 0100 1011

Temperature 27

Ranger 388

DSP Board VCC 6.25

Motor Vd 2.2

Servo Vd 3.75

A/D Vref 4.5

Potentiometer Vd 2.0

Servo Control

| | Command Value | Time(ms) | |
|---|---|---|---|
| Disable Servo #1 | 3650<br>(* Range at 2500,5000) | 1000 | GO |
| Enable Servo #2 | 3900<br>(* Range at 3300,4300) | 1000 | GO |
| Disable Servo #3 | 3400<br>(* Range at 3400,4500) | 1000 | GO |
| Enable Servo #4 | 3500<br>(* Range at 4880,2700) | 1000 | GO |
| Enable Servo #5 | 3500<br>(* Range at 4880,2800) | 1000 | GO |
| Disable Servo #6 | 4200<br>(* Range at 3400,5000) | 1000 | GO |

Figure V.4 GUI of the WiRobot DRK8000 Controller (VB)

User can click the button on the interface to control the robot as long as the "WiRobot Gateway" is connected to the robot.

The following are some functions used in this sample application for controlling the servos mounted on the head:

```
WiRobotSDK1.EnableServo 0
WiRobotSDK1.ServoTimeCtr 0, 3800, 1000
```

The WiRobotSDK1is the WiRobot SDK Component ActiveX control. The first line will enable the channel 0 servo. The second line will control the servo to take 1000ms to the target position of 3800.

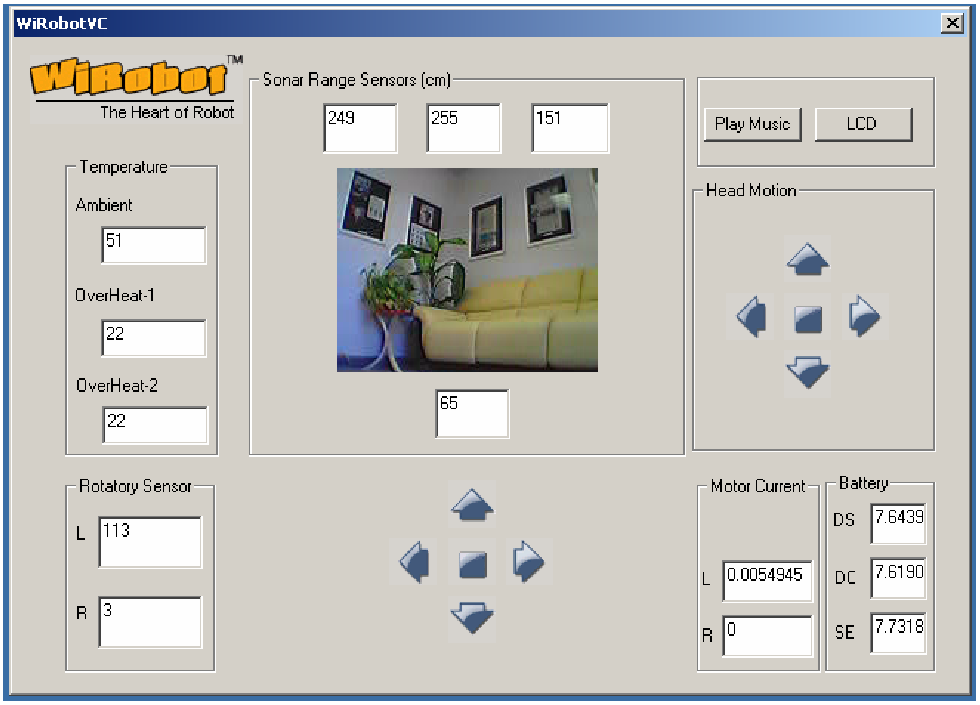
User can also obtain and control the multimedia information of the robot simply by calling the following functions

```
WiRobotSDK1.LcdDisplayPMS sourceFileName
WiRobotSDK1.TakePhoto
WiRobotSDK1.SavePhotoAsBMP destinationfileName
```

The first function will change the LCD display on the robot to the image stored in the "sourceFileName" which is in bitmap format. The second function will request the robot to take a picture and an "ImageEvent" will be triggered when this image is ready for pickup. The third function will save the image to the file with "destinationfileName" in bitmap format.

## V.3. Sample Application 2 - WiRobot DRK6000/8000 Controller (VC++)

The second sample application demonstrates how to program a VC++ application using the WiRobot system. The GUI of this program interface is shown as follows:



Sonar Range Sensors (cm): 249, 255, 151, 65

Temperature: Ambient 51; OverHeat-1 22; OverHeat-2 22

Rotatory Sensor: L 113; R 3

Motor Current: L 0.0054945; R 0

Battery: DS 7.6439; DC 7.6190; SE 7.7318

Figure V.5 GUI of the WiRobot DRK6000/8000 Controller (VC++)

To obtain an image from the robot, user can call the following function where m_ctlSDK is a member variable of the class CWiRobotSDK imported from the "WiRobot SDK ActiveX Module"

```
m_ctlSDK.TakePhoto ();
```

To control the robot to move forward continuously, user can call the following functions

```
m_ctlSDK.SetDcMotorControlMode (0, 0);
m_ctlSDK.SetDcMotorControlMode (1, 0);
m_ctlSDK.DcMotorPwmTimeCtrAll (32066, 32066, -32768, -32768, -32768, -32768, 800);
```

To stop the robot, the following commands can be used:

```
m_ctlSDK.SuspendDcMotor (0);
m_ctlSDK.SuspendDcMotor (1);
```

For details on how to control the robot using Microsoft Visual C++, please refer to the sample C++ source code and the WiRobot SDK API Reference Manual.

## V.4. Other Sample Applications

More sample applications are available in the “SampleApps” folder for reference.

# VI. Miscellaneous

## VI.1. System Update

Dr Robot will provide software update for both the PC software as well as the DSP software to enhance the existing features. User can visit www.DrRobot.com to check for any new updates available for the existing system.