Wireless Networked Autonomous Mobile Robot

Sentinel 3

Quick Start Guide

WARNINGS

Do NOT power on the robot before reading and fully understanding the operation procedures explained in this manual.

Neither the robot, nor the program is bug free, accidence could happen; you have to make sure that the robot always maintains a safe distance from people during operation.

The robot should be turn off (i.e. the power switch should be on OFF position) when not in used. Battery should be fully charged before storage. Battery pack should be recharged every two weeks while in storage.

Failure to follow these warnings could cause serious injury or death and/or damage to the robot.

Copyright Statement

www.DrRobot.com

Contact

General: info@DrRobot.com

Technical Support: support@DrRobot.com

25 Valleywood Drive, Unit 20
Markham, Ontario, L3R 5L9, Canada
Tel: (905) 943-9572 Fax: (905) 943-9197

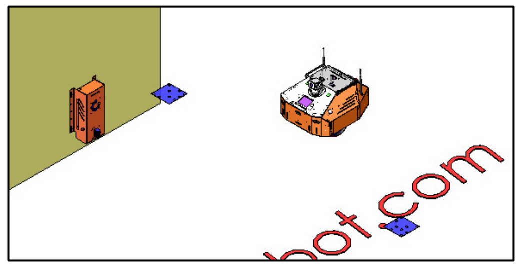
Table of Contents

I. Introduction

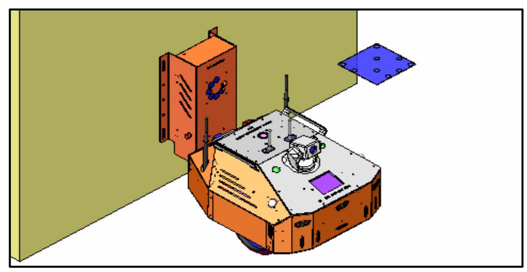
Sentinel^3 is a wireless networked autonomous mobile robot platform, featuring 24x7 collision-free autonomous navigation, vision-landmark localization sensor(indoor GPS), autonomous docking and recharging, high resolution pan-tilt-zoom color camera (max 704x480), two-way audio, tele-operation and remote monitoring. Sentinel^3 is built with durable aluminum chassis; with payload capacity over 15Kg (optional 40Kg) at the body weight under 6kg.

Key Features

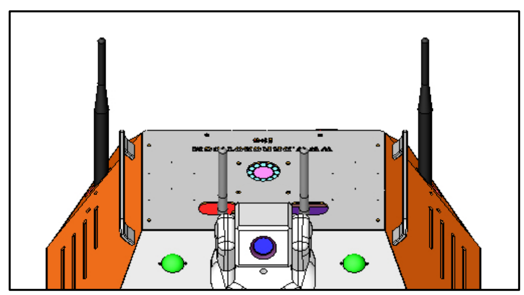
• **Navigation and Localization** providing collision-free point-to-point autonomous navigation. The vision-landmark base indoor localization (indoor GPS, position/orientation) sensor and the landmarks together provides precise position and direction information covering every inch of the floor.



• **Auto-Docking and Recharge Station** offering 24x7 continuous operation and self-docking/recharging capability without human assistance. (2 hour fast charge)

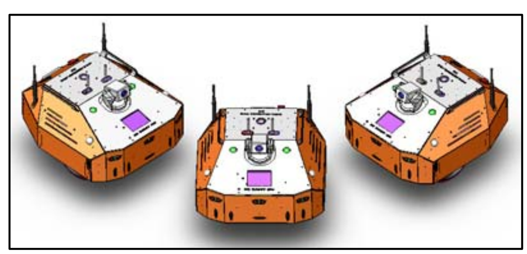


• **Fully Wireless Networked** running on standard WiFi, IEEE 802.11 b/g wireless. (options of 802.11A and 802.11N are available on request)

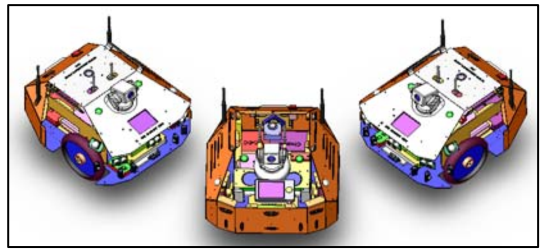


• **OS Independent Application Development Tools** allow operation through simple motion and sensor control command and offer full support of high level application development using raw data packet protocols. High level ActiveX controls and MSRS (Microsoft Robotics Studio) support are also available for rapid development under Windows OS

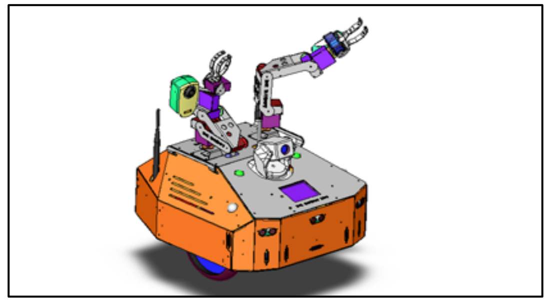
• **Navigation Sensors** a set of indoor collision avoidance sensors including 7 IR, 3 sonar sensors and 2 human sensors. (Laser scanner option is available on request)



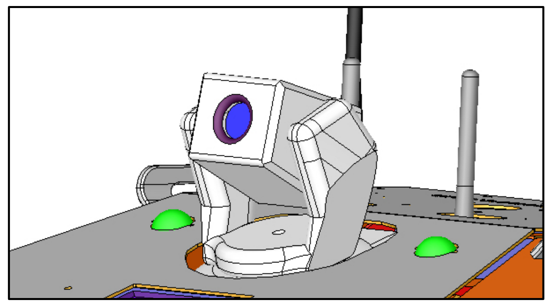
- **Comprehensive Circuit Protection** includes battery over-charge, battery over-heat, motor over-heat and motor over-load protection. Hardware and software watch-dogs provide continuously monitoring of the overall system.



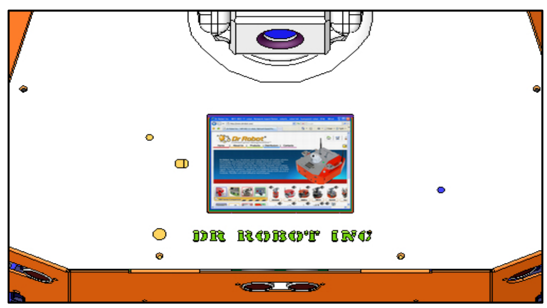
- **Various Upgrades** include 5 DOF servo arms with gripper, camera on arm, dual-camera animated head, laser scanner, multiple cameras and long range RFID reader.



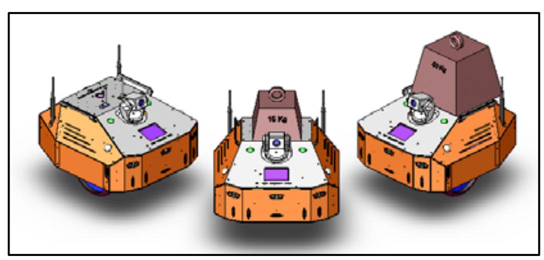
- **High Resolution Pan-Tilt-Zoom Camera** with max 704x480 pixels image, max 30fps, and full-duplex two-way (max 32Kbps) streaming audio. (full image and audio development API is supplied)



- **320x240 Color Touch Screen** is capable of displaying video, playing audio, and running web applications.



- **Lightweight and Large Payload Capacity** Max payload of 15Kg (Optional 40Kg) with body weight under 6Kg.



- **Tele-Operation and Remote Monitoring** via Internet through the Dr Robot Remote Client Program.

- **Extended Operating Time** 2 hour nominal operation time for each recharging (standard version and battery pack). Upgrade options: power and battery systems for 6, 12 hours operation time are available.

II. Sentinel^3 Robot and Upgrade Options

Sentinel^3 Robot Standard Components

i90-ME	i90 Mechanical Chassis	1
AV-PTZ-VH	Pan-Tilt-Zoom Camera with Two-Way Audio	1
SG-NAV	Indoor Localization (GPS) Sensor with 40 landmarks	1
AUTO-CHAR12	"Auto-Recharge" Wall Mounted Charger	1
PDM1950	Color Display with Touch Screen (320x240)	1
WFS802G	WiFi 802.11b/g Wireless Module	2
PMS5005	Robot Sensing and Motion Controller	1
MDM5253	DC Motor Driver Module with Position & Current Feedback	1
DUR5200	Ultrasonic Range Sensor Module	3
DHM5150	Pyroelectric Human Motion Sensor Module	2
GP2Y0A21YK	Sharp IR Distance Measuring Sensor	9
MCR3210	RS232 Interface Module	1
SP-AM-2W	2W Speaker with Amplifier	1
CCR2150	RS232 Cross-Over Serial Cable	1
MCA-50	40kg.cm 12V DC Motor with Integrated 800 Count per Cycle Optical Encoder	2
PMCHR12	Power Management and Recharging System	1
BPN1238	12V Ni-MH 3800mAh Battery Pack (could be upgraded to 9000mAh - BPN1290)	2
IOB-160	IO Interface Board	1
-	Gamepad	1
-	Tele-operation/Monitoring Software	1
-	802.11b/g wireless router	1

Upgrade Options

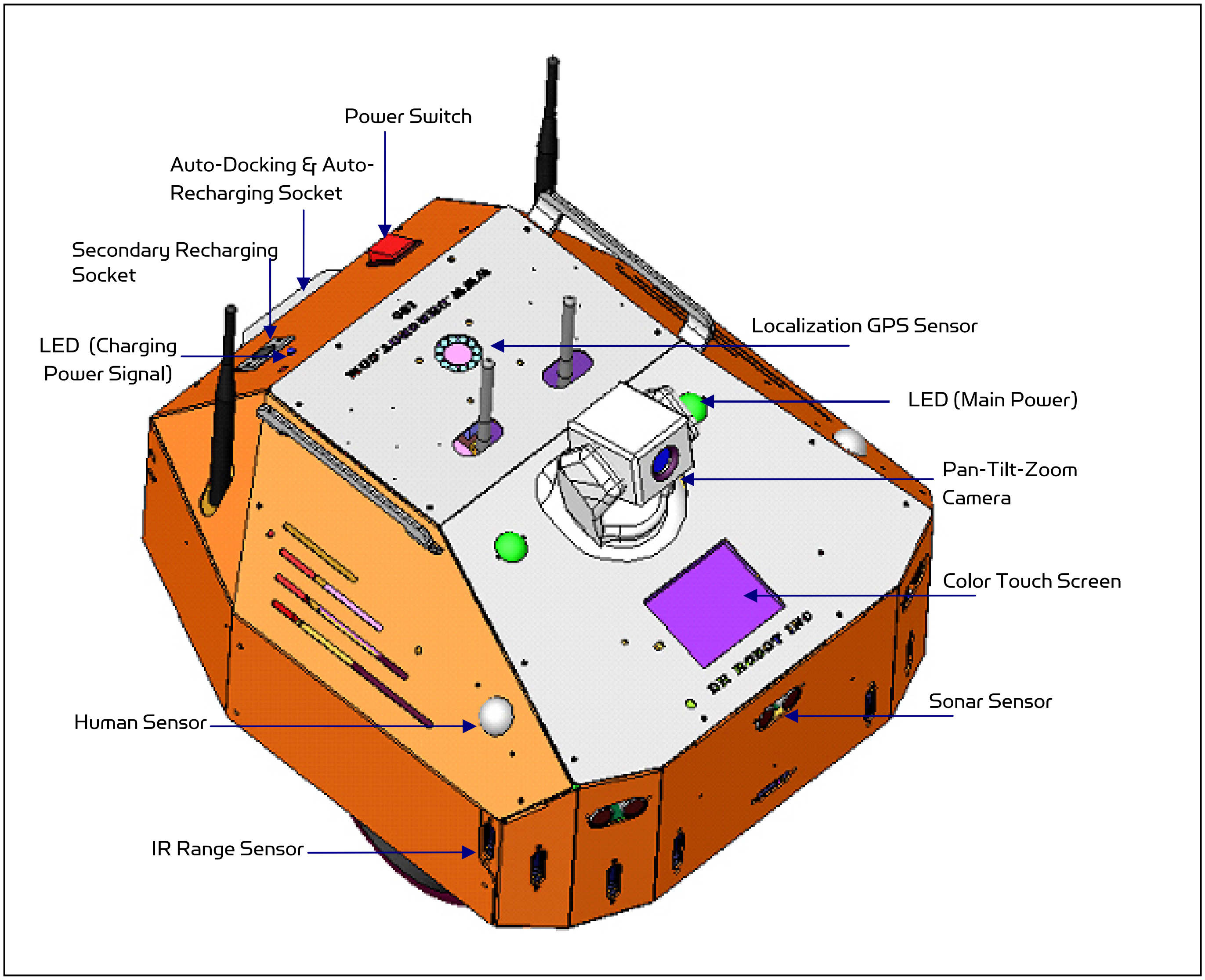
5 DOF servo arm with gripper	SARM-5DOF
Dual-camera animated head	DCAM-HEAD
Laser Scanner	LAS4M
Mini color 640x480 wireless camera with audio	AXCAM
Pan-tilt-zoom (704X480) camera with two-way audio	AV-PTZ-VH
Long range RFID system (reader and tags)	RFID-LR-U
2 X 9000mAH 12V Ni-MH battery	BPN1290

Please contact support@drrobot.com for custom design and integration inquiry.

III. Knowing Your Robot

Sensors and External Components

The figure below illustrates the key functional components you will identify on the outside of the Sentinel[3] robot.



Sentinel[3] Overview

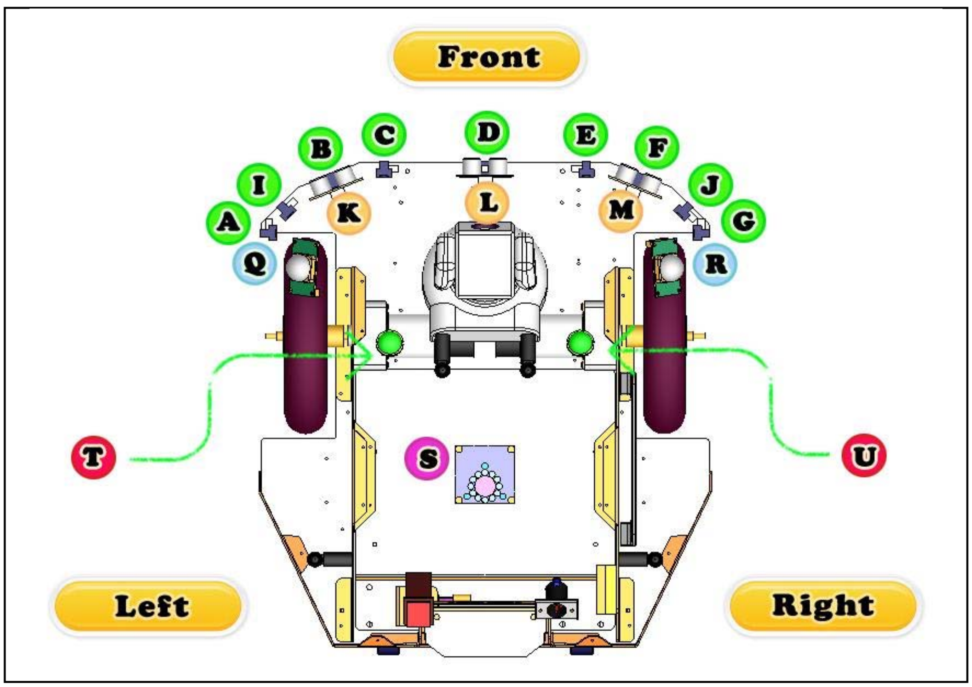
When the maim power switch is on, the main power LED will be lit. When the robot detects input power from the recharging socket, the charging power signal LED will be lit.

The robot comes with 3 sonar and 7 IR range sensors. These range sensors are for environment detection and collision avoidance.

The localization GPS sensor on Sentinel[3] provides precise robot position and direction information for autonomous navigation docking task through working with the ceiling mounted landmarks.

Table below summarizes the numbering of sensor modules.

Sensor Module	Location
Ultrasonic #1	K - Left front
Ultrasonic #2	L - Middle front
Ultrasonic #3	M - Right front
Human Sensor #1	Q - Left front
Human Sensor #2	R - Right front
Infrared Range Sensor #1	A – Front left
Infrared Range Sensor #2	B – Front left
Infrared Range Sensor #3	C – Front middle
Infrared Range Sensor #4	D – Front middle
Infrared Range Sensor #5	E – Front middle
Infrared Range Sensor #6	F – Front right
Infrared Range Sensor #7	G – Front right
Infrared Range Sensor #9	I – Front left
Infrared Range Sensor #10	J – Front right
Localization Sensor #1	S – Localization Sensor
DC Motor #1 with quadrature encoder	T - Left , use channel 1
DC Motor #2 with quadrature encoder	U - Right, use channel 2

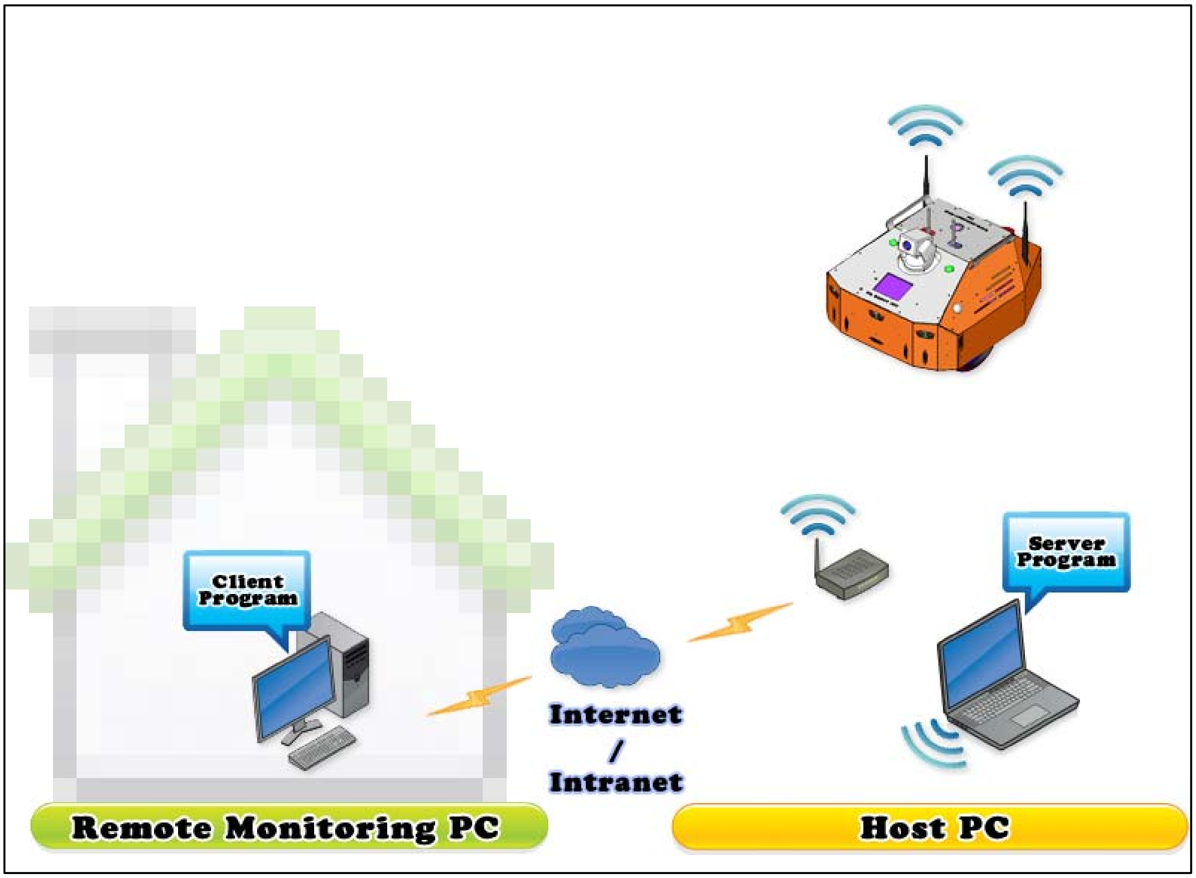


Sensor Module Location (Top View)

Operation Scenario

Diagram below shows the typical operation scenario. The Sentinel[3] is a wireless networked robot. It connects to the wireless AP or router via IEEE 802.11b/g network. The host PC (or called server PC) running the "Sentinel III Control" program could connect to this network via either:

- Network cable – Connect the host PC to one of the LAN ports on the back of the router (DO NOT connect to the WAN port), or
- Wireless – To connect the host PC to the wireless router, configure the host PC's wireless settings using the default wireless configuration settings found in the Network Connection session of this manual.



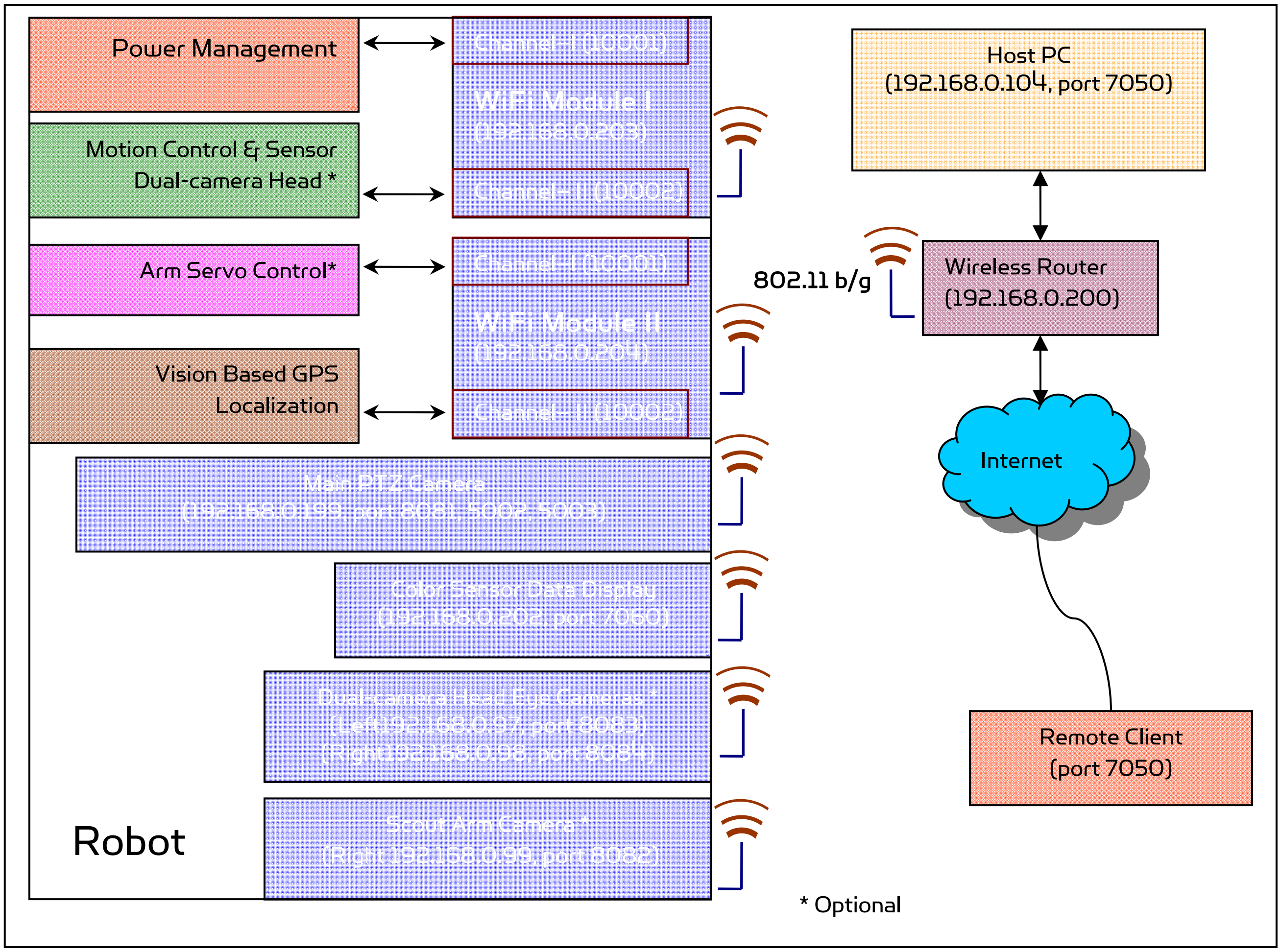
Typical Operation Scenario

Note: The host PC (or called server PC) running the "Sentinel III Control" program could be mounted on the robot instead off the robot if your application requires so.

User could be able to control the robot, see and listen through the robot via the Dr Robot Remote Client program from anywhere around the world with Internet connection.

Network Connections

The diagram bellows shows the network architecture.



Wireless Router Setting

The included pre-configured wireless 802.11 B/G router has the following pre-set settings:

SSID	dri	**Router LAN**	192.168.0.200
WEP	128bits	**Login ID**	admin
KEY	11223344556677889 9AABBCCDD	**Password**	drrobot
Key Type	Open Key		

with virtual server settings as following:

Virtual Server	Port	Protocol	Server IP
Sentinel III Control program	7050	TCP/IP	192.168.0.104** (Host PC IP)
Main PTZ Camera	8081	TCP/IP	192.168.0.199 (Main Camera IP)

Main PTZ Camera	5002	UDP	192.168.0.199(Main Camera IP)
Main PTZ Camera	5003	UDP	192.168.0.199(Main Camera IP)

** These settings have to match your actual host computer's IP address.

Wireless Device Settings

The 802.11X wireless devices used in Sentinel[3] robot have following settings.

Name	**Robot Main IP (Robot WiFi Module 1)**	**IP**	192.168.0.203 *
SSID	dri	**Gateway**	192.168.0.200 (Router IP)
WEP	128bits	**Subnet Mask**	255.255.255.0
KEY	112233445566778899AABBCCDD	**Key Type**	Open Key

Name	**Vision Based GPS sensor IP (Robot WiFi Module 2)**	**IP**	192.168.0.204 *
SSID	dri	**Gateway**	192.168.0.200 (Router IP)
WEP	128bits	**Subnet Mask**	255.255.255.0
KEY	112233445566778899AABBCCDD	**Key Type**	Open Key

Name	**Main PTZ Camera**	**IP (Port)**	192.168.0.199 (8081,5002,5003)*
SSID	dri	**Gateway**	192.168.0.200 (Router IP)
WEP	128bits	**Subnet Mask**	255.255.255.0
KEY	112233445566778899AABBCCDD	**Key Type**	Open Key
Login	root	**Password**	drrobot

Name	**Color LCD Touch Screen**	**IP**	192.168.0.202 *
SSID	dri	**Gateway**	192.168.0.200 (Router IP)
WEP	128bits	**Subnet Mask**	255.255.255.0
KEY	112233445566778899AABBCCDD	**Key Type**	Open Key

*IPs for your robot may be different, they can be found on the bottom of your robot.

In this manual, the Host/Server PC network parameters are assumed to be configured as:

	IP	**Gateway**
Host /Server PC	192.168.0.104	192.168.0.200 (Router IP)

Internet Remote Monitoring/Tele-operation

If Internet remote monitoring/control is required, you need to connect the wireless router WAN port to your current broadband Internet modem (this is optional and is not required for running the robot).

If firewall is in-place in your network, you also need to make sure all the network ports used by the wireless devices (e.g. 8081, 5002, 5003 for the main PTZ camera and other devices in upgrade options) and 7050 on the server and remote client sides are not blocked for the Internet remote monitoring/control tasks to operate properly.

Advanced Network Settings

It's possible to use different network settings (e.g. IP) for the server PC, but the "Virtual Server" settings on the router must also be changed accordingly in order for the Internet remote monitoring feature to work properly.

You could also change the router settings such as IP and SSID etc,. If you need to do so, you are required to change the network settings on the WiFi modules on the robot by following the guidelines as illustrated on the WiFi Module manual.

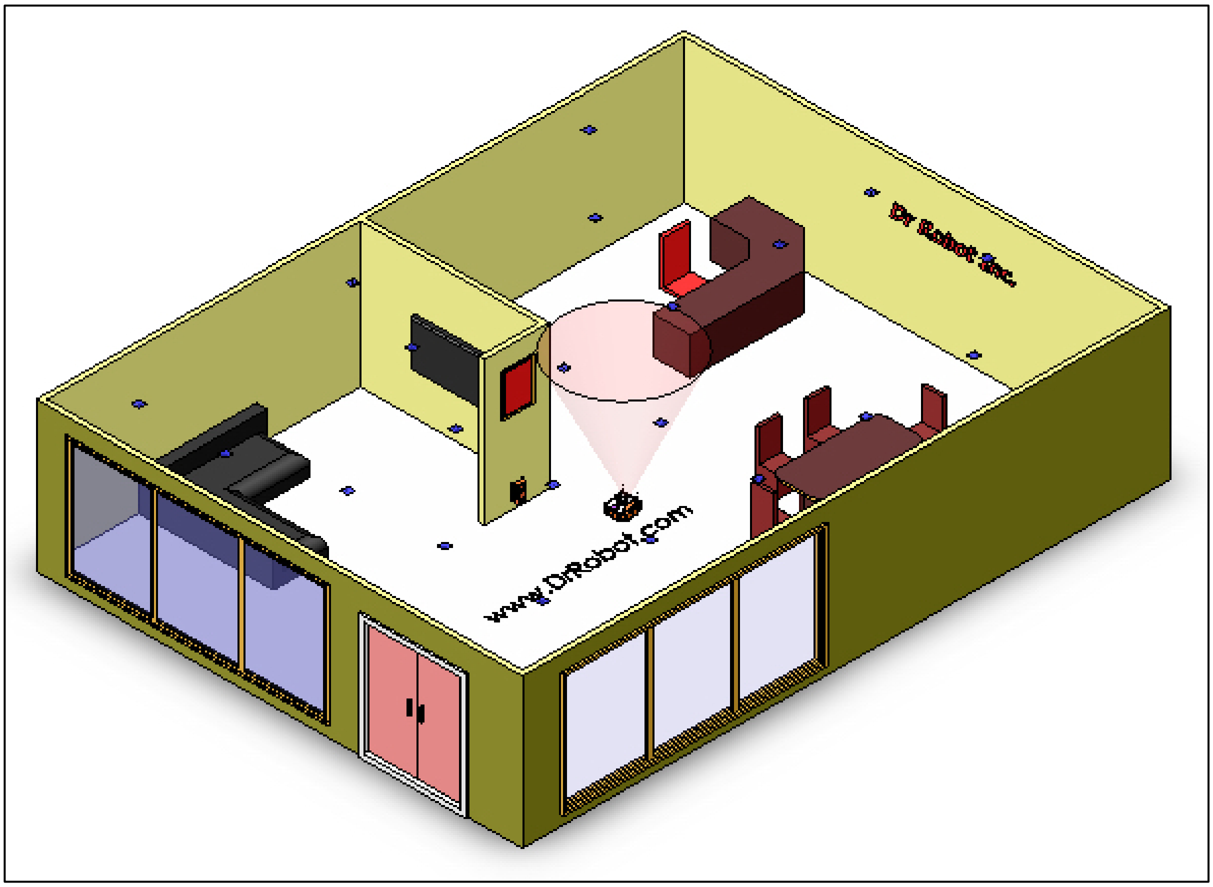
How Localization (indoor GPS) works?

Thanks to the integrated localization GPS sensor, the Sentinel[3] is able to tell where it is in the room. The localization system consists of the localization GPS sensor on the robot and the landmarks mounted on ceiling.

Passive landmarks are attached on ceiling. Landmark is made of a material that strongly reflects infrared ray. Coding on each landmark differentiates landmark one from the other.

The localization (indoor GPS) sensor is composed of both an IR projector and an image processing unit. It analyzes the infrared ray image which is reflected from different ID number-given passive landmarks on the ceiling. The range from the sensor to landmarks on ceiling is also automatically measured and calibrated. The relative position/direction relationship between two landmarks is automatically measured and calibrated through the landmark mapping procedure during landmark setup.

Landmark does not consume any electrical power and easy to install and extend. Each localization sensor can have up to 4,095 landmarks. This sensor is also robust to the disturbances from environmental lighting such as sunshine or fluorescent lights.



Localization Sensor on Robot Providing Precise Position and Orientation Measurement Using Ceiling Mounted Landmarks

More detail regarding the GPS sensor readings, landmark mounting and mapping procedure will be discussed in session "Setting Up the Localization Landmarks" under "Robot Setup".

IV. Software Installation

Installing Sentinel^3 Programs from CD

On the host/server computer install the following programs from the installation CD:

- *"Sentinel III Control" program*

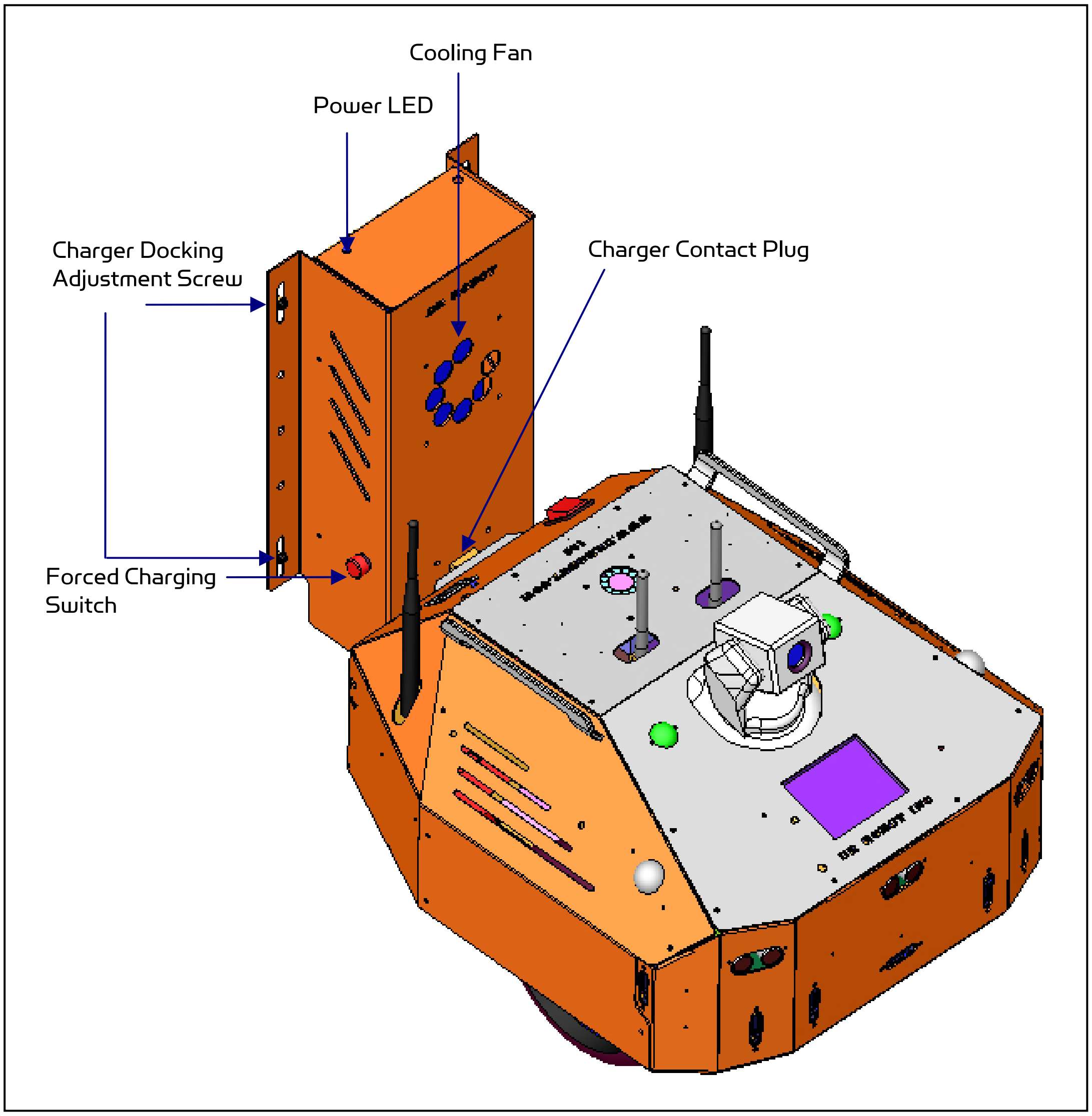
On the remote client computer install the following program from the installation CD (if Internet remote operation and monitoring is to be used):

- *"Dr Robot Sentinel III Remote Control" program*

V. Robot Setup

The robot setup involves setting up the wall-mounted charging station and the localization landmarks.

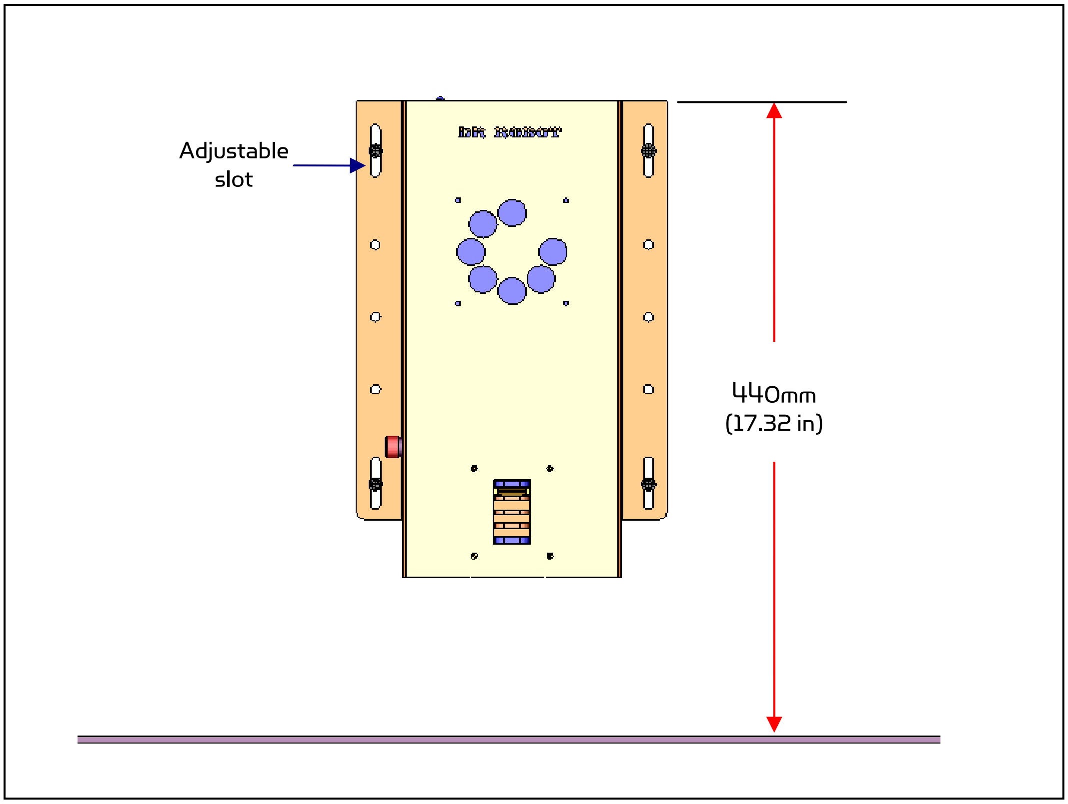
Setting Up the Charging Station



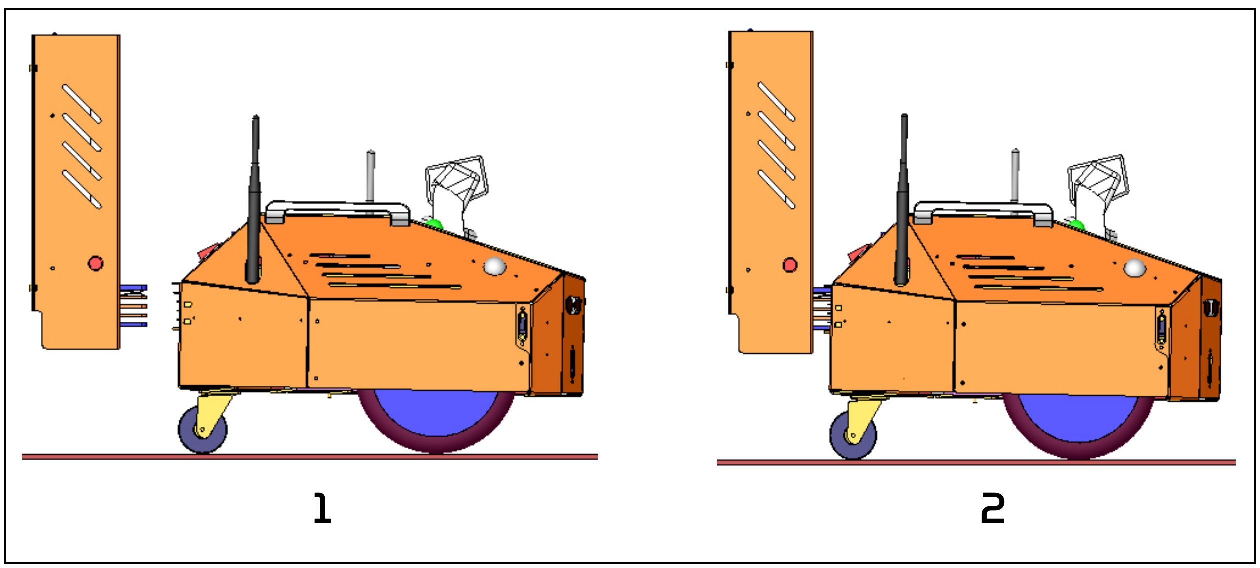
Auto-docking and Auto-recharging Station

Procedure to set up the charging station:

1. Choose a robot easy access spot on the wall to mount the charging station.
2. Mounting the charging station 440mm above the floor measured from the top edge (see figure below) to the floor. Use leveler if needed. Temperately fix the screws on the four adjustable slots (see figure below) for easy height adjusting in next step.



3. Dock the robot (Charging Socket) into the Charger Contact Plug (as shown on figure below). Adjusting the charger height if needed so that the robot could easily plug in and pull out from the charger. Once satisfied, tighten all screws. More screws could be used on the additional 6 mounting holes for stronger mounting.



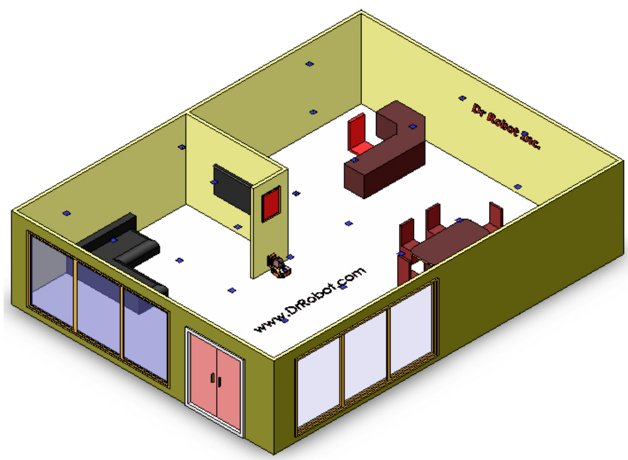
4. Plug the main power cord to the wall power unit.

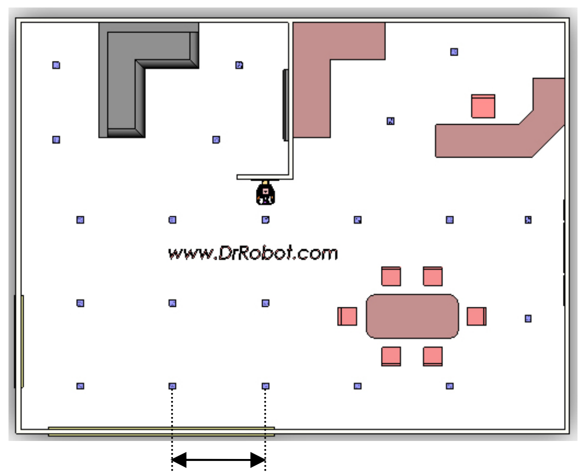
Setting Up the Localization Landmarks

There are 40 localization landmarks (type: HLD2-1) in your Sentinel[3] package. These landmarks are normally enough to cover over 160 m^2 (1700 ft^2) office or home space. This type of landmark (HLD2-1) should be mounted on the ceiling with a height between 1.35 to 3.15m (which is equivalent to 1.1m to 2.9m measured from the localization sensor on robot to the ceiling). For higher ceiling mounting, other types of landmarks, such as HLD2-2 and HLD2-3 are required. Additional landmarks and high ceiling version landmark could be purchased from Dr Robot by contacting sales@drrobot.com.

Placement of Landmarks

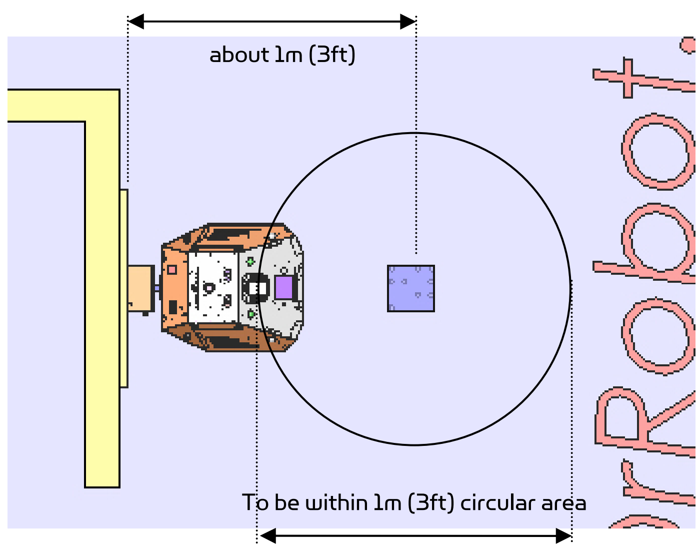
Landmarks should be placed at 2 m intervals for the height of about 2.5 m in order that any dead zone may not occur. Smaller interval should be applied if the ceiling is lower, or bigger interval could be used if ceiling is higher.





For height of 2.5m, we recommend the markers should be placed at about 2m interval to avoid dead zone

For best auto-docking and auto-recharging operation, we recommend that a landmark should be placed above the charging station within an circular area of 1m in diameter, and center at 1m in front of the charging station (see diagram below)



Suggested Charger Landmark Mounting Location

Building the Map of Localization Landmarks

Once landmarks are attached, we could make a map of the landmarks. Once this is done, position/orientation readings from the localization sensors will be under a single coordinate system. The placement of the landmark with the reference ID becomes the origin. In our sample setup, the reference landmark is the one above the charging station. This is the procedure of building the landmark map:

1) Run the "Sentinel^3 Localization/GPS Setup" program.

2) Follow the 4 steps shown on the program as repeated here. **Notice**: After sending out (by clicking on) each command, double check to confirm that the ACK window (on the right) under Received Message section shows the command you sent out to make sure that the sensor is receiving to your command without communication error. If ACK is not returned, you should re-send the command.

Step1: Make connection to the sensor by typing in the correct robot IP address that can be found under your Sentinel^3 and port number "10001" and click the WiFi Connection.

Step2:

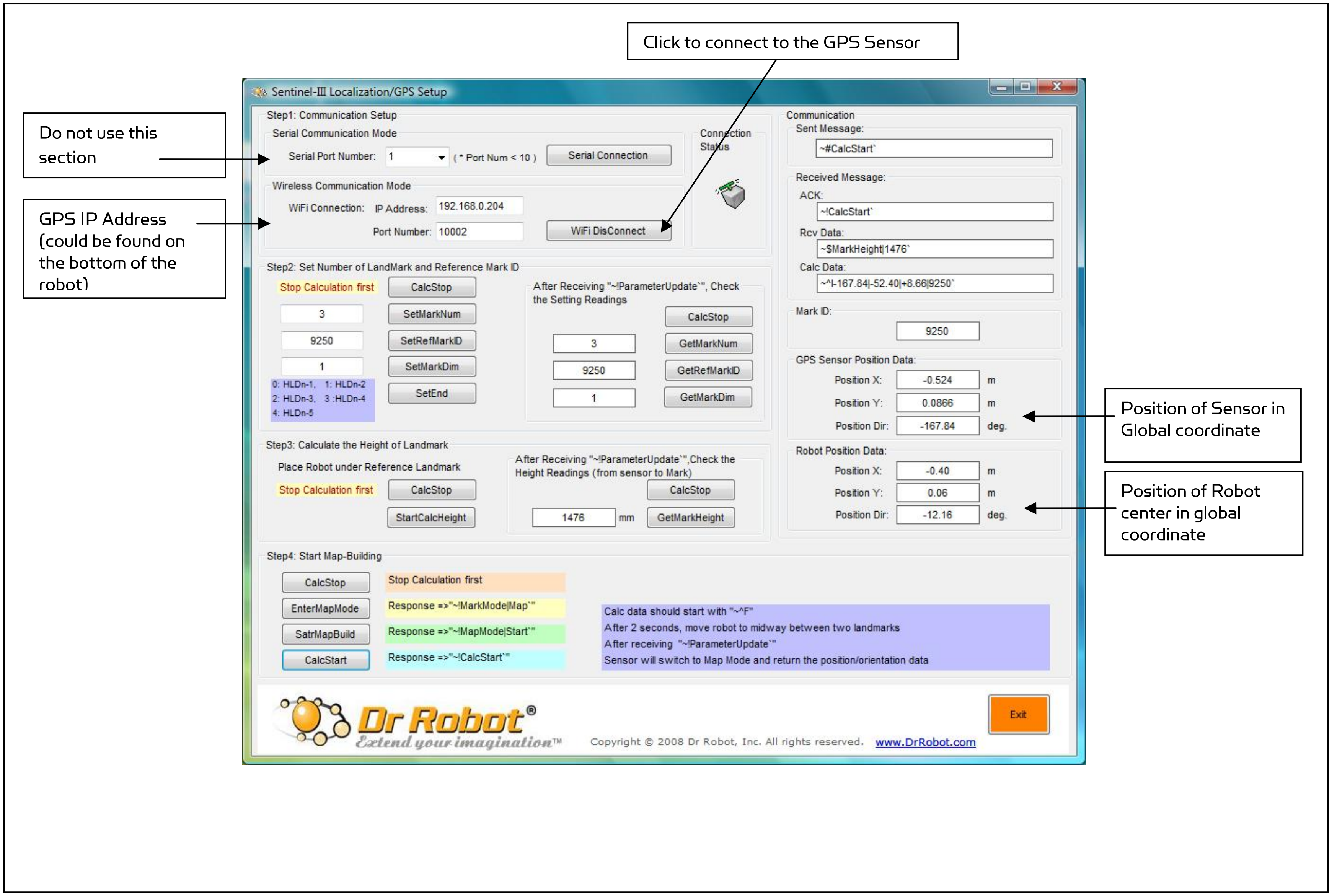
- Press "CalcStop", wait for ACK message return "~!CalsStop";
- Input the total landmark number and press the "SetMarkNum", wait for ACK message;
- Enter the first landmark ID (reference ID) and press "SetRefMarkID, wait for ACK message;
- Set Landmark type to "Office" and press "SetMarkType", wait for ACK message;
- Press "SetEnd", wait for ACK message;
- After receiving "~!ParameterUpdate'" shown in the "ACK" window, using the buttons on the right ("GetMarkNum", "GetRefMarkID", GetMarkType") to double check the parameter settings.

Step3: Calculate the height of the landmark. Place the robot under the first landmark (reference Landmark, in our sample setup, this is the landmark above the charging station):

- Press "CalcStop", wait for ACK message return "~!CalsStop";
- Press "StartCalcHeight", wait for ACK message;
- Using the "GetMarkHeight" to check the Height calculation after receiving "~!ParameterUpdate'" shown in the "ACK" window.

Step4: Start landmark map building, while the robot is still under the first (reference) landmark,

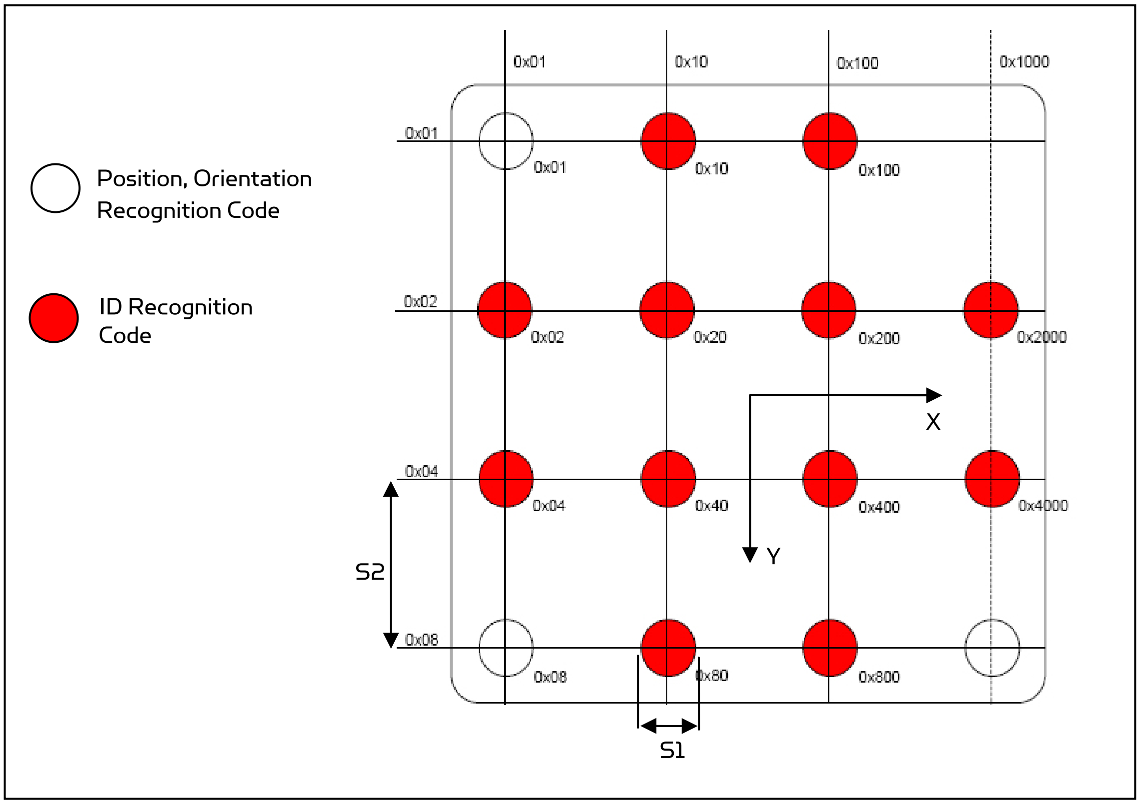
- Press "CalcStop", wait for ACK message return "~!CalsStop";
- Press "EnterMapMode", wait for ACK message;
- Press "StartMapBuild", wait for ACK message;
- Press "CalcStart", wait for ACK message;
- The "Calc Data" window should display data starting with "~^F"
- Move the robot to midway of the 1st and 2nd landmark and wait for few seconds, continue this procedure till finishing with the last landmark.
- When the last landmark is seen, "~!ParameterUpdate'" will be received (this message only stay for a short while)
- The GPS sensor will automatically switch to Map mode and return the position/orientation reading. The "Calc Data" window should display data string starting with "~^I" in normal localization mode.



Sentinel[3] Localization/GPS Setup Program

Landmark ID and Co-ordinations

Each Landmark has a unique identification number (ID). Following diagram illustrates the coding of the ID. (X, Y) coordinate is shown in the diagram below.

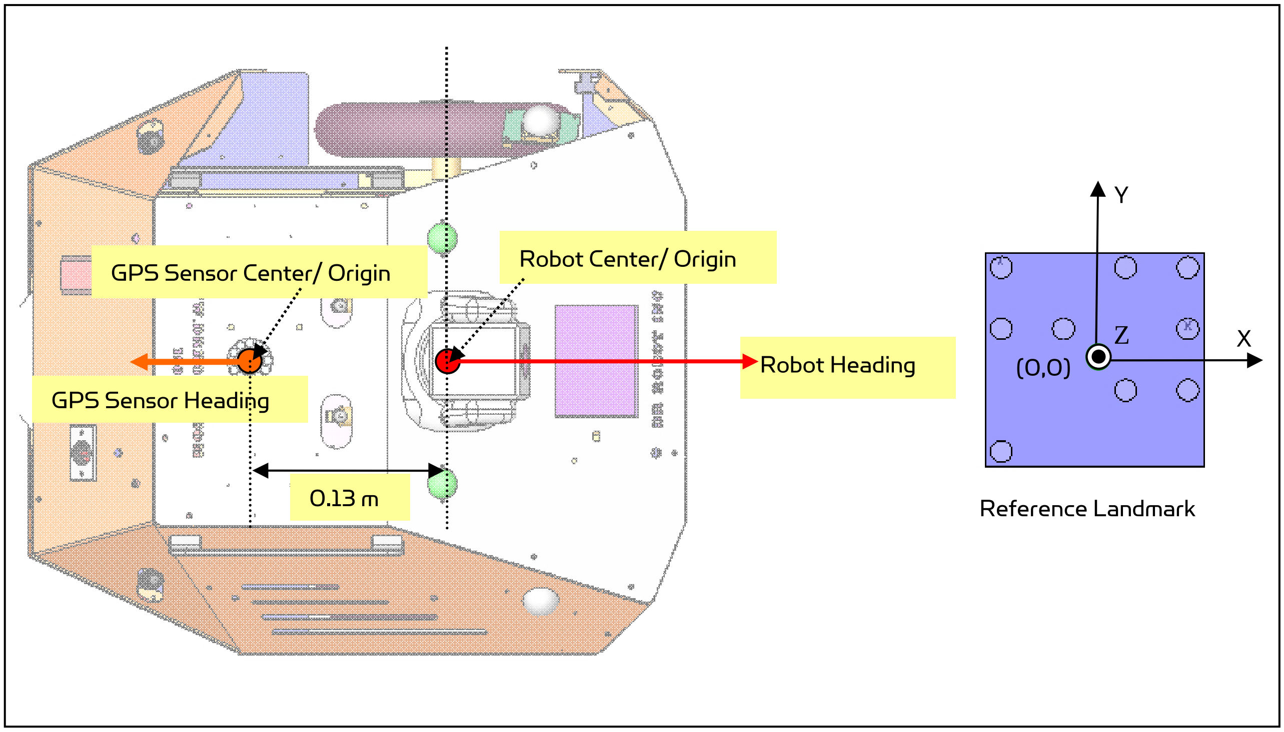


Landmark ID Coding and (X,Y) Coordinate (view facing mark points)

Understanding the Robot Position and Direction information

Once the landmark map is built, position and direction information you are going to use or see in the Sentinel3 programs is referred to the global coordinate which is attached to the reference landmark. In this case, the reference landmark is the one above the charging station.

The position and direction data you will use for path planning are of the center of the robot (as shown in the diagram below) which is the mid point along the wheel axis and with respect to the global coordinate. When the robot heading aligns with the X axis of the global coordinate, we have direction = 0 degree and when the robot heading align with the Y axis of the global coordinate, we have direction = 90 degree. Right-hand coordinate is adopted here.



Coordinates and Heading Direction

When using the "Sentinel[3] GPS Setup" program, you may also notice that there are two sets of position and direction information, one for the GPS sensor and one for the robot.

GPS Sensor Position Data:		
Position X:	0.3251	m
Position Y:	-0.324	m
Position Dir:	87.88	deg.
Robot Position Data:		
Position X:	0.32	m
Position Y:	-0.19	m
Position Dir:	92.12	deg.

As shown in the diagram "Coordinates and Heading Direction"above, the heading direction of the GPS sensor is in opposite of the robot heading and the distance between the center of the GPS sensor and the robot is 0.126m. The positive rotation direction of the GPS sensor is in clockwise around the Z axis, while the positive rotation of the robot is in clockwise around the Z axis, therefore, the relationship between GPS sensor direction θ_{GPS} and θ_{Robot} is:

$$\theta_{Robot} = 180 - \theta_{GPS}, if\ \theta_{GPS} \in (0,180)$$

$$\theta_{Robot} = -180 - \theta_{GPS}, if\ \theta_{GPS} \in (-180,0)$$

$$\begin{bmatrix} X_{Robot} \\ Y_{Robot} \end{bmatrix} = \begin{bmatrix} 0.126 * \cos(\theta_{Robot}) + X_{GPS} \\ 0.126 * \sin(\theta_{Robot}) + Y_{GPS} \end{bmatrix}$$

Landmark Type

The standard landmarks coming with the Sentinel[3] are HLD2-1 type, which is a 4x4 ID coding landmark. The maximum unique number of landmark is 4,095. The mounting height of HLD2-1 (from the GPS sensor to the landmark) is 1.1-2.9 m (3.7 - 9.7 ft) with covering area in radius of 1.6-3.2m (5.3-10.7 ft) depending on the mounting height. Higher the ceiling (no less than 2.9m for HLD2-1 landmark), bigger coverage it will be.

Other types of landmark are available for higher ceiling mounting, e.g. HLD2-2 and HLD2-3. The mounting height of HLD2-2 is 2.9-4.5m while the HLD2-3 is 4.5-6.5m. They also provide larger covering area with each landmark while the landmark size is also larger. Please contact support@drrobot.com if these landmarks are needed.

Model	ID Type	Height Range (cm)	Localization Range (cm)	S1/S2 (mm) (Size/Space of Landmark Points)
HLD2-1*	4x4	110-290	160-320	14/40
HLD2-2**	4x4	290-450	350-440	24/60
HLD2-3**	4x4	450-650	440-630	30/80

* 40 Landmarks are included; ** Available on request.

VI. Robot Operations

Using the "Sentinel III Control" program, Sentinel[3] could patrol on pre-set path continuously and return to charger when needed. During the patrol, remote user could watch, listen and talk through the robot camera, microphone and speakers.

Program the Patrol and Auto-docking/recharging path

"Sentinel III Control" performs the auto patrol and auto-docking/recharging task based on the path described in the "patrol.xml" and "charge.xml" (sample XML files can be found in Appendix I and II)

Path file XML format

Tag Name	Description
TargetX	Target point X coordinate
TargetY	Target point Y coordinate
TargetDir	Robot heading direction at target point (-179, +180)
StopTime	The time you like the robot to stay at target point before moving to next target (second). This will be ignored if NonStop is true.
ForwardSpeed	Maximum forward running speed when heading to this target (m/s)
Forgetable	true: in the case that the robot could not reach the target, robot is allowed to gradually give up and move to next target if TargetTime is passed. false: robot will keep trying to reach the target indefinitely.
NonStop	true: the robot will move to the next target without stopping once the current target is reached. It will ignore the TargetDir and StopTime settings. false: The robot will Stop at the target, and stay for the time specified by StopTime.
FinalPosture	true: after reaching the target, robot will turn and adjust its heading to the direction specified by TargetDir. false: the robot will ignore the direction setting TargetDir.
TargetTime	The time allowing the robot to run to the target, unit: 0.1s. If Forgetable = true, when TargetTime passes, the robot will gradually give up the current task and move to next target.
TargetTolerance	When reaching the circular area centered at the target with radius of TargetTolerance (m), the robot is considered to reach the target.
MaxTurnSpeed	Maximum forward running speed when heading to this target (degree/s)
CAEanble	true: built-in collision avoidance feature will be enabled during the robot movement. false: built-in collision avoidance feature will be disenabled during the robot movement.
ReverseDrive	The robot will drive in reverse direction (backup), e.g. used when backing up into charging station.
TargetDirTolerance	Robot heading direction tolerance when FinalPosture = true (degree)
SeqNo	Point sequence number in path

Here is an example

```
<PointSet xmlns="http://tempuri.org/PointSet.xsd">
    <PointConfigTable>
        <TargetX>2.01</TargetX>
        <TargetY>-0.72</TargetY>
        <TargetDir>90</TargetDir>
        <StopTime>2</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>false</NonStop>
        <FinalPosture>true</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>90</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>2</SeqNo>
    </PointConfigTable>
```

In this example, the robot is commanded to drive to the target (X,Y) = (2.01, -0.72) with MaxForwardSpeed=0.5m/s and MaxTurnSpeed= 90 degree/s. The time allowed to complete this task is 20 second, and when robot reaches the target with a radius of 0.05m and heading direction within TargetDirTolerance of 5 degree, task is considered to complete. During the task, the collision avoidance feature is enabled. The robot will keep trying to reach this target indefinitely and with heading of 90 degree. Once the target is reached, it will stop for 2 seconds before executing next task.

Planning a Patrol Path

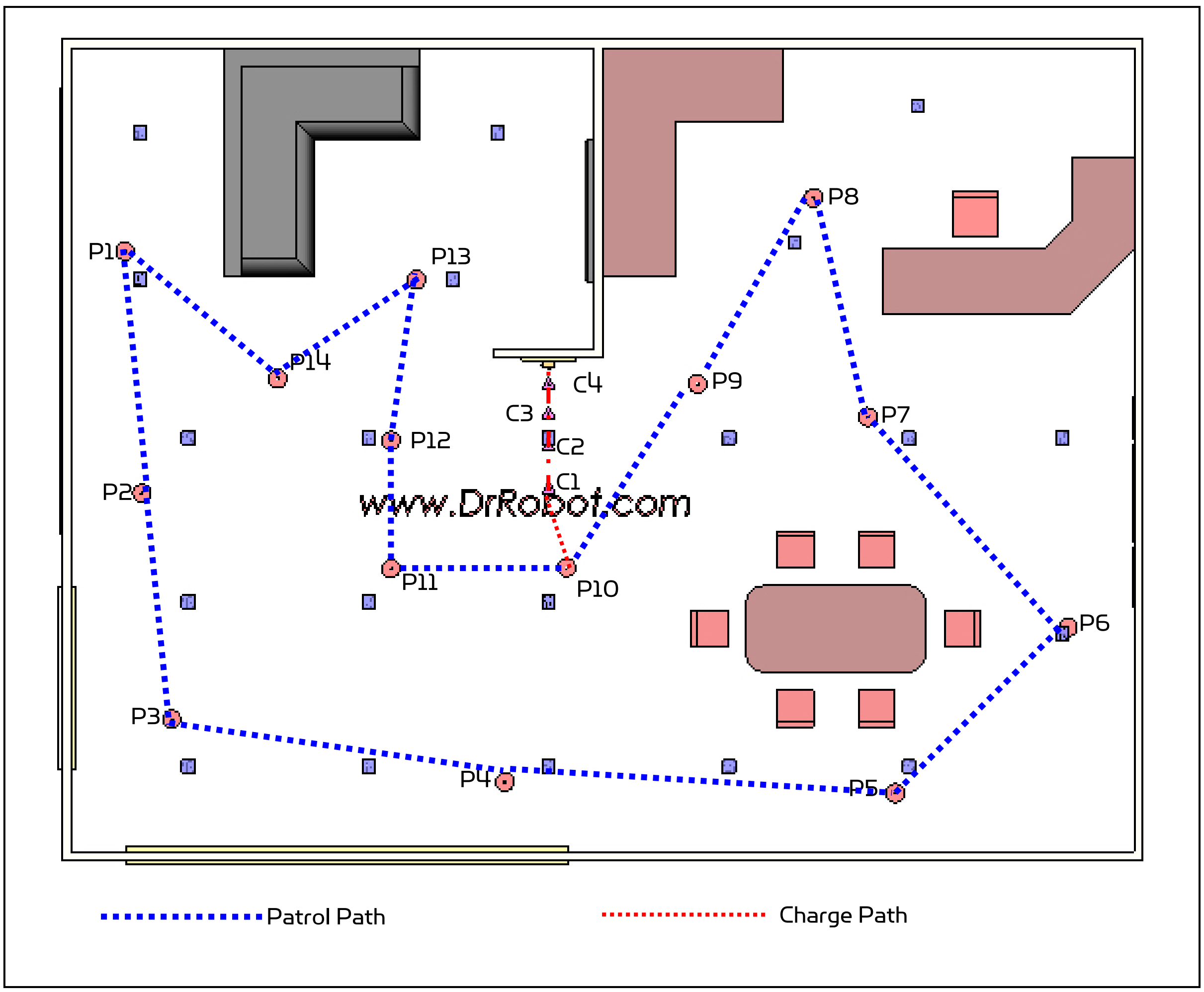
Here is an example on planning a patrol path in the office environment. We will use the landmark above the charging station as reference mark ID, so the origin (0, 0) of the global coordinate is under this landmark. X, Y axes are showed in the diagram below. Here, we assume that the landmark mapping has been carried out using the "Sentinel^3 GPS setup" program.

Run the "Sentinel III Control" program, after connecting to the GPS sensor, you will be able to read the robot position data (X,Y, and orientation).

In this example, we will specify a patrol path with 14 via points, named P1 to P14 as shown in the diagram below. You could place the robot at the via points you like it to patrol on, and read the robot position/orientation data using the "Sentinel III Control" program. Then you could write them into the patrol path XML file "patrol.xml".

Please notice that you can set some of the via points with Forgetable as false to make sure the robot will try to reach this via point at most, such as in some scenario like getting out of some corners or dead ends. For example, the P5, P6, P7, P9, P14 could be set with Forgetable = false.

The patrol path should be in a loop for robot to easily move back and continue the patrol indefinitely, i.e. in this example, the last via point P14 should be close to via point P1.



Sample of Patrol and Docking Path Setup

The patrol .xml for this patrol path setup can be found in Appendix I: Patrol.xml.

Planning an Auto-Docking Path

An auto-docking/charging path will bring the robot off the patrol path and dock robot's charging socket (on the back) into the charging plug on the charging station. (see the "Sample Patrol and Docking Path Setup" diagram above for the sample of charging path)

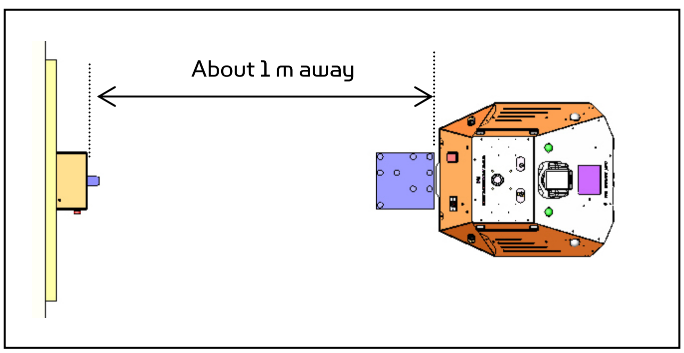
Before you work on the auto-charging/docking path, the GPS sensor must be configured first and landmark mapping should also be completed using the "Sentinel III Localization/GPS Setup" Program. The Sentinel[3] docks into the charging station in backward motion. We recommend that the "charging/docking" path will start from somewhere on the patrol path, or simply share a via point with the patrol path, e.g. the P10 in this sample setup. This will ensure an un-obstructed transaction from the patrol to charging task.

When the robot finds the battery is running low or receiving a "Go charge" command during a patrol task, it will identify the closest via point on the patrol path, and drive over there. After that, it will drive along the patrol path toward the "charging/docking path" assuming the patrol path is in a loop and charging/docking path is accessible from at least one of the via points on this patrol path. In this sample setup, the P10 on patrol path is shared by the charging path.

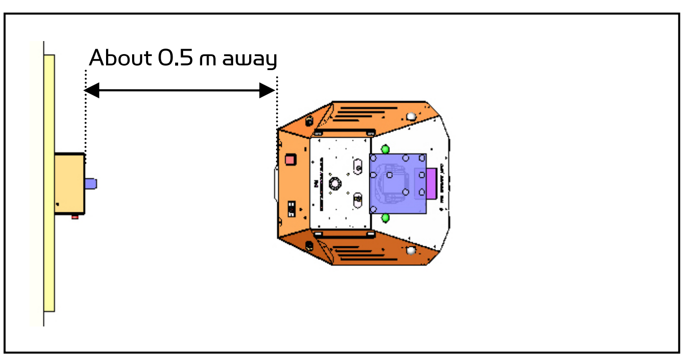
Sample Charging/docking Path

Here we will demonstrate a simple charging path starting with the P10 and 4 other docking via points named C1, C2, C3 and C4. (Referred to the "Sample Patrol and Docking Path Setup" diagram above for the sample of charging path) Please follow four steps below to obtain the position and orientation information of C1-C4 as required for preparing this sample "charge.xml".

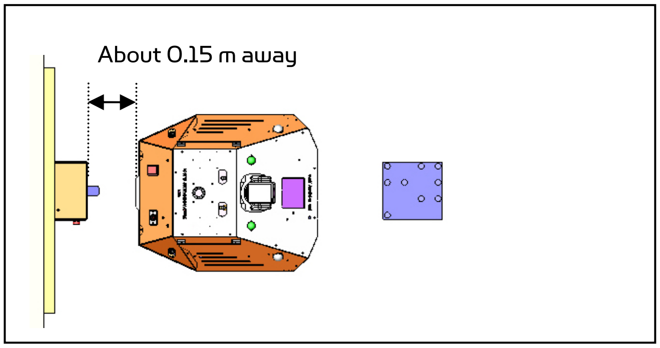
Step 1: Place the robot at about 1 m (between the back of the robot to the front of the charger) in front of the charger; visually align the center of the robot with the center axis of the charger; use the "Sentinel III Control" program to obtain the robot position and orientation. Write this down as C1= (X_1, Y_1, θ_1), and you will need these to enter into the "charge.xml".



Step 2: Move robot closer to the charger at about 0.5 m (between the back of the robot to the front of the charger) in front of the charger; visually align the center of the robot with the center axis of the charger. Using the "Sentinel III Control" program to obtain the robot position and orientation and write this down as C2 = (X_2, Y_2, θ_2).



Step 3: Move robot closer to the charger at about 0.15 m (between the back of the robot to the front of the charger) in front of the charger; visually align the center of the robot with the charger plug. Using the "Sentinel III Control" program to obtain the robot position and orientation and write this down as C3 = (X_3, Y_3, θ_3).



Step 4: The fourth point is a virtual point where the back of the robot is behind the charging station with the same heading direction as the C3. This virtual point C4 = (X_4, Y_4, θ_4) could not be measured by placing the robot, but could be easily estimated/calculated based on C3 by adding about 0.35m along the docking direction.

Table below summarize the settings for these four via points.

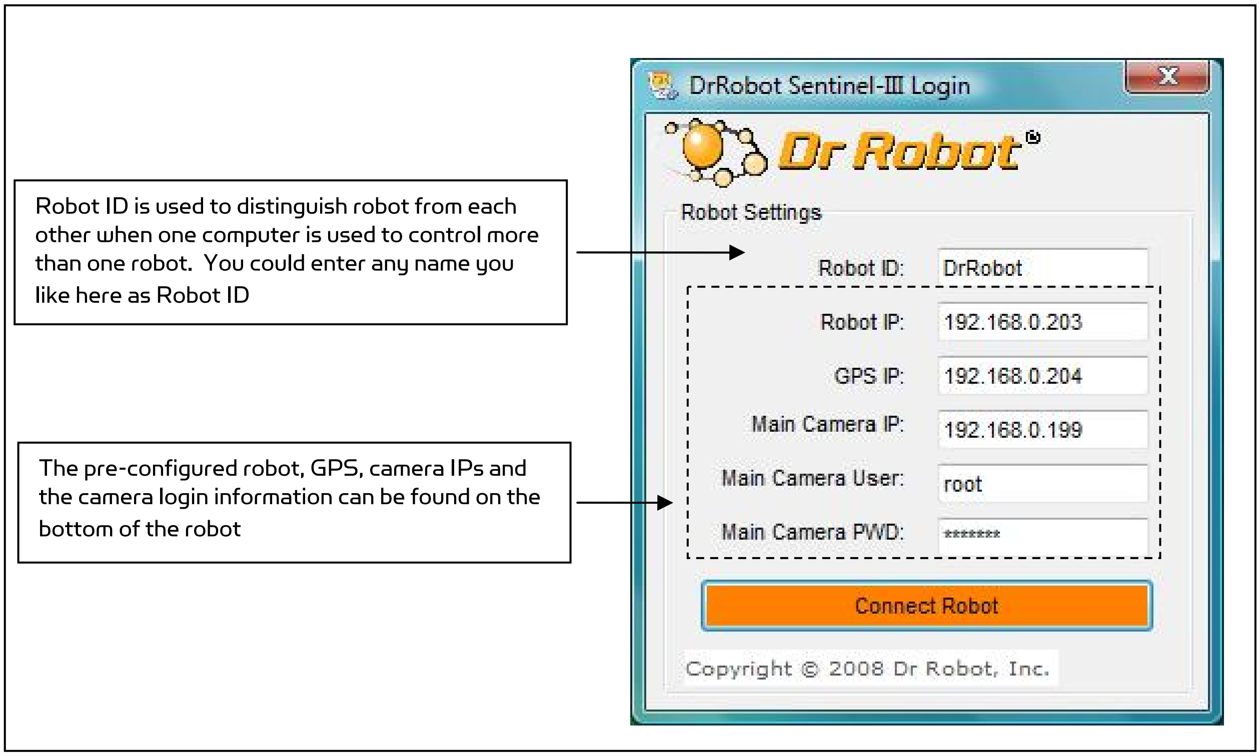
Tag/Point	C1	C2	C3	C4
TargetX	X_1	X_2	X_3	X_4
TargetY	Y_1	Y_2	Y_3	Y_4
TargetDir	θ_1	θ_2	θ_3	θ_4
StopTime	5	2	2	0
ForwardSpeed	0.5	0.5	0.5	0.2
Forgetable	False	false	false	True
NonStop	False	false	false	true
FinalPosture	True	true	true	false
TargetTime	200	200	200	200
Targettolerance	0.05	0.05	0.05	0.05
MaxTurnSpeed	90	90	45	35
CAEnable	True	False	false	false
ReverseDrive	False	True	true	true
TargetDirTolerance	5	5	5	5
SeqNo	1	2	3	4

Notice to application developers: When docking to the charging station, your application program should be monitoring the extended digital input pin 0. When the robot docks successfully into the charging station, this bit will change from "1" to "0", your program should stop the robot first, and set extended digital output pin 0 to "1". This will signal the charging station to activate the charging power (the fan on the charger will run when activated). After that, your program could send charging command to the robot power management system to start charging.

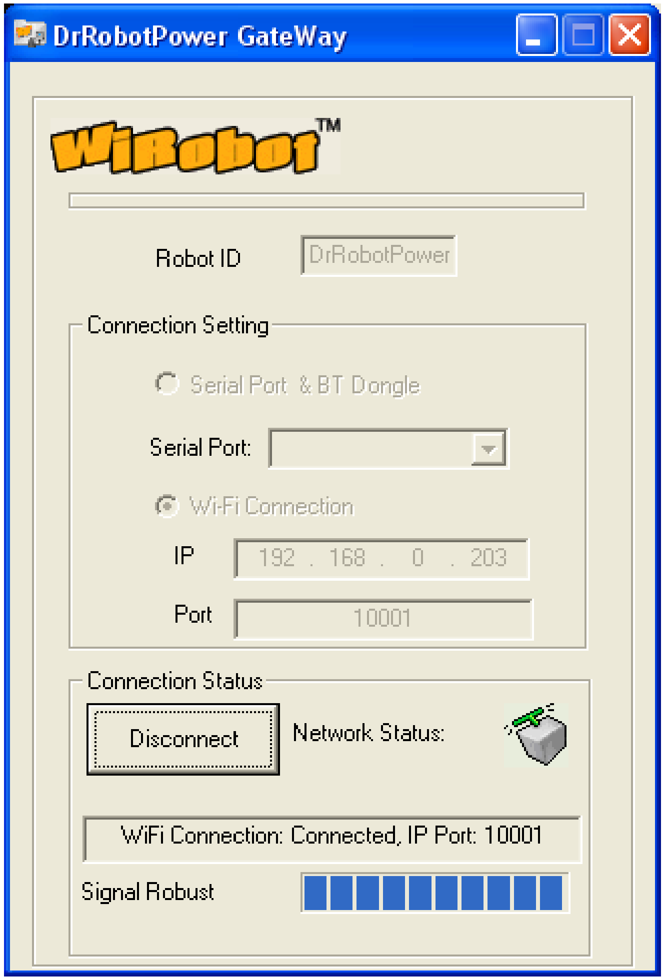
Using the "Sentinel III Control" Program

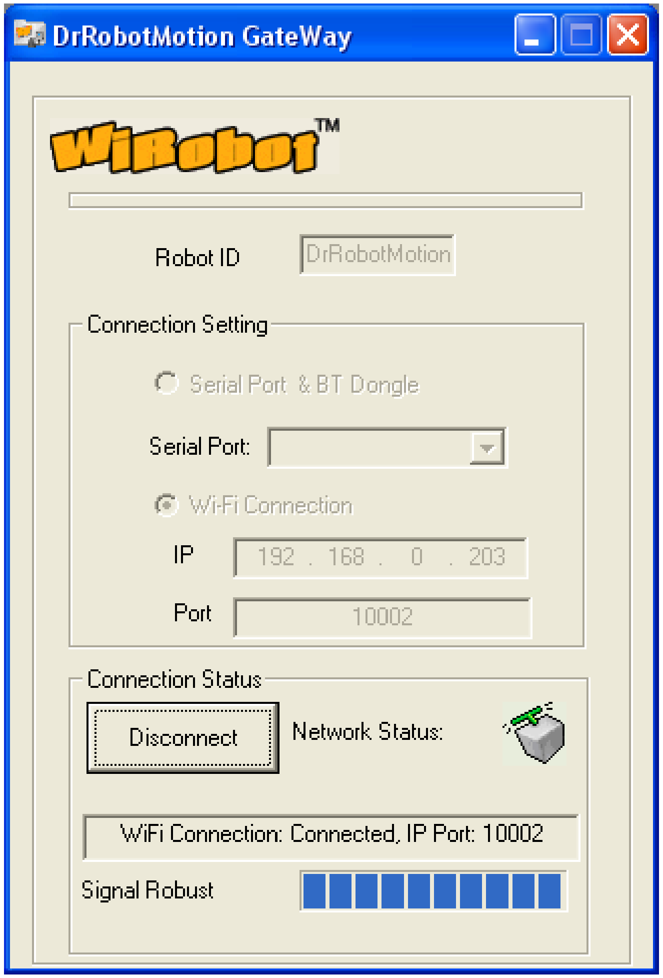
If the "Sentinel III Control" is not installed, install it from the installation CD now, after installation, you should find a shortcut icon ("Sentinel III Control") on your desktop.

Start the "Sentinel III Control" Program, there will be a window allowing you to enter the IP addresses for robot, GPS sensor and camera. The default IP may not match yours. Yours can be found on the bottom of the robot. Click "Connect".

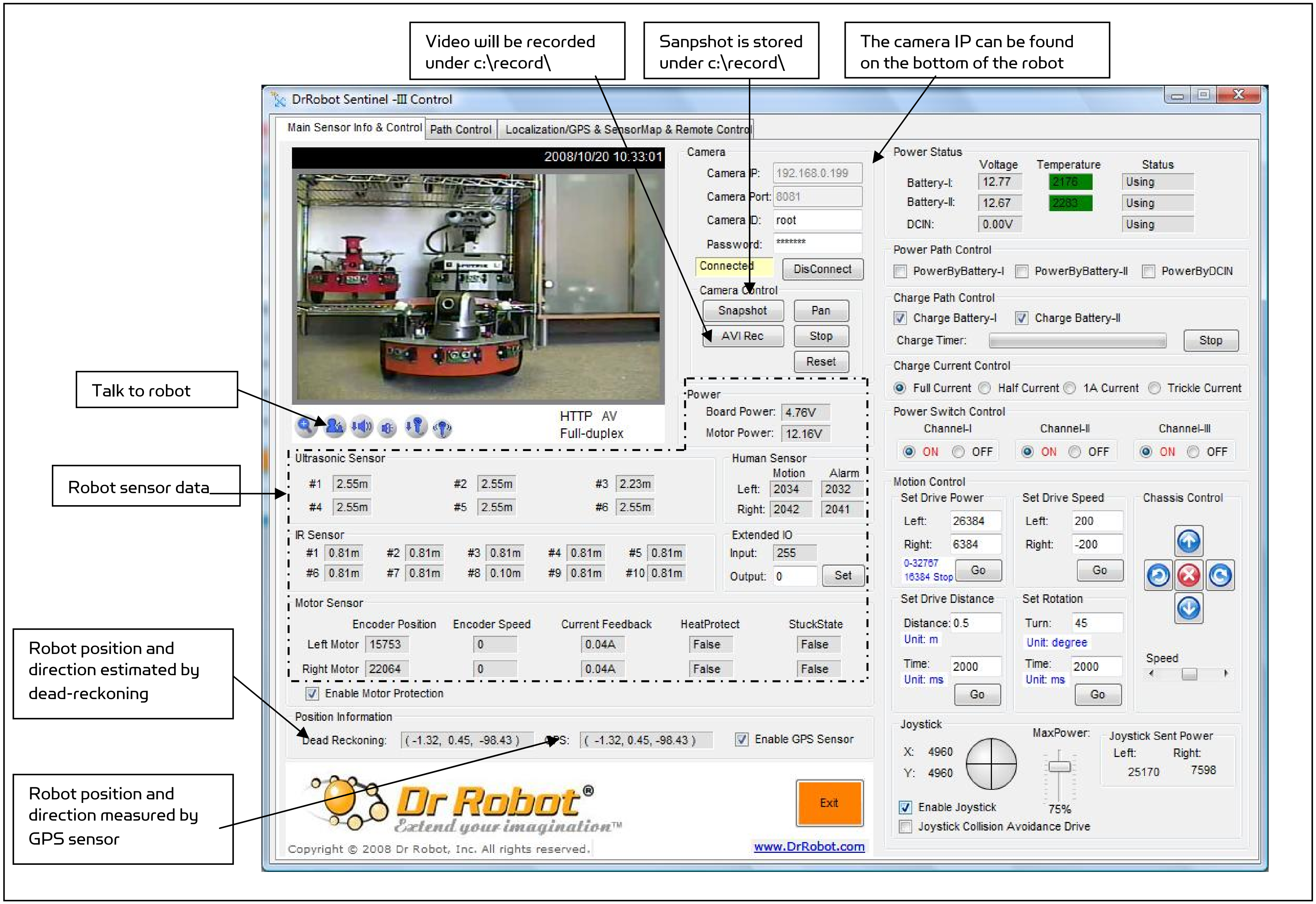


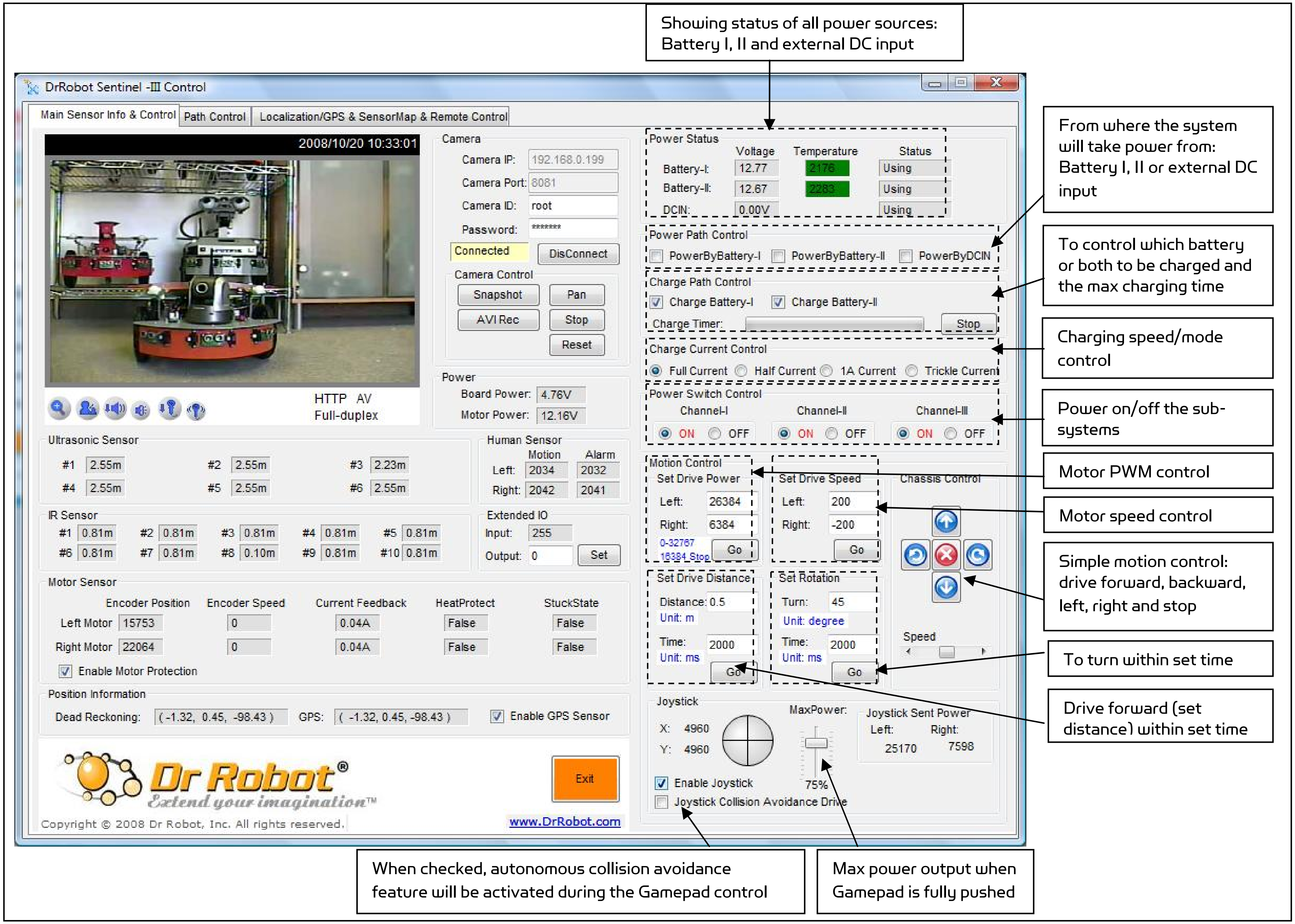
Two copies of "WiRobot Gateway" programs will start and connect using the settings entered in the login screen. After this, the main control program will start.





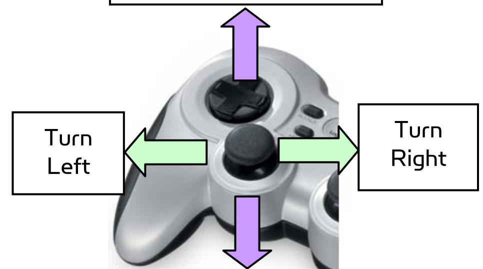
Once the main program starts, there are three GUI tags to choose. The first tag is “Main Sensor Info & Control”



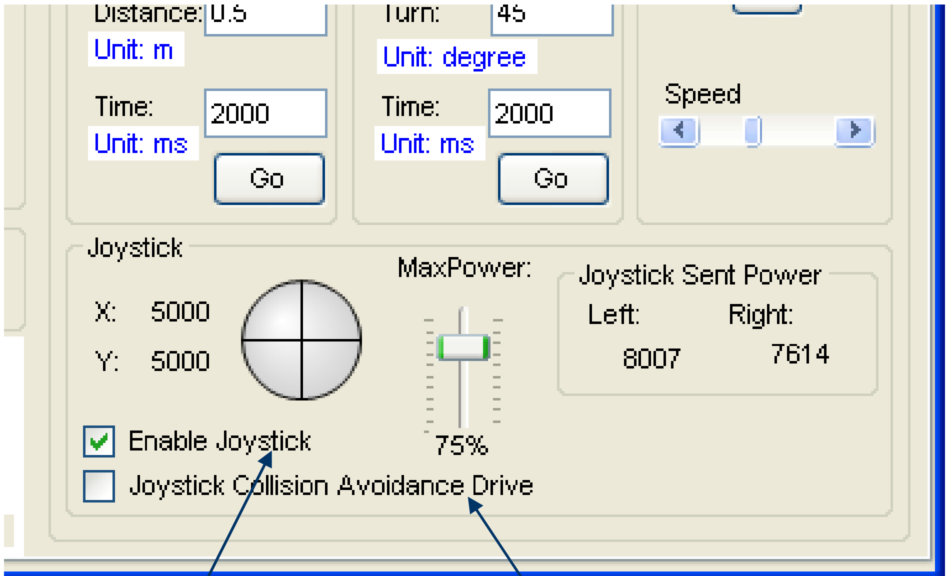

DrRobot Sentinel -III Control
Showing status of all power sources: Battery I, II and external DC input
From where the system will take power from: Battery I, II or external DC input
To control which battery or both to be charged and the max charging time
Charging speed/mode control
Power on/off the sub-systems
Motor PWM control
Motor speed control
Simple motion control: drive forward, backward, left, right and stop
To turn within set time
Drive forward (set distance) within set time
When checked, autonomous collision avoidance feature will be activated during the Gamepad control
Max power output when Gamepad is fully pushed

Driving with Gamepad

Go Forward
More you push, faster the robot goes



Go Backward
More you push, faster the robot goes

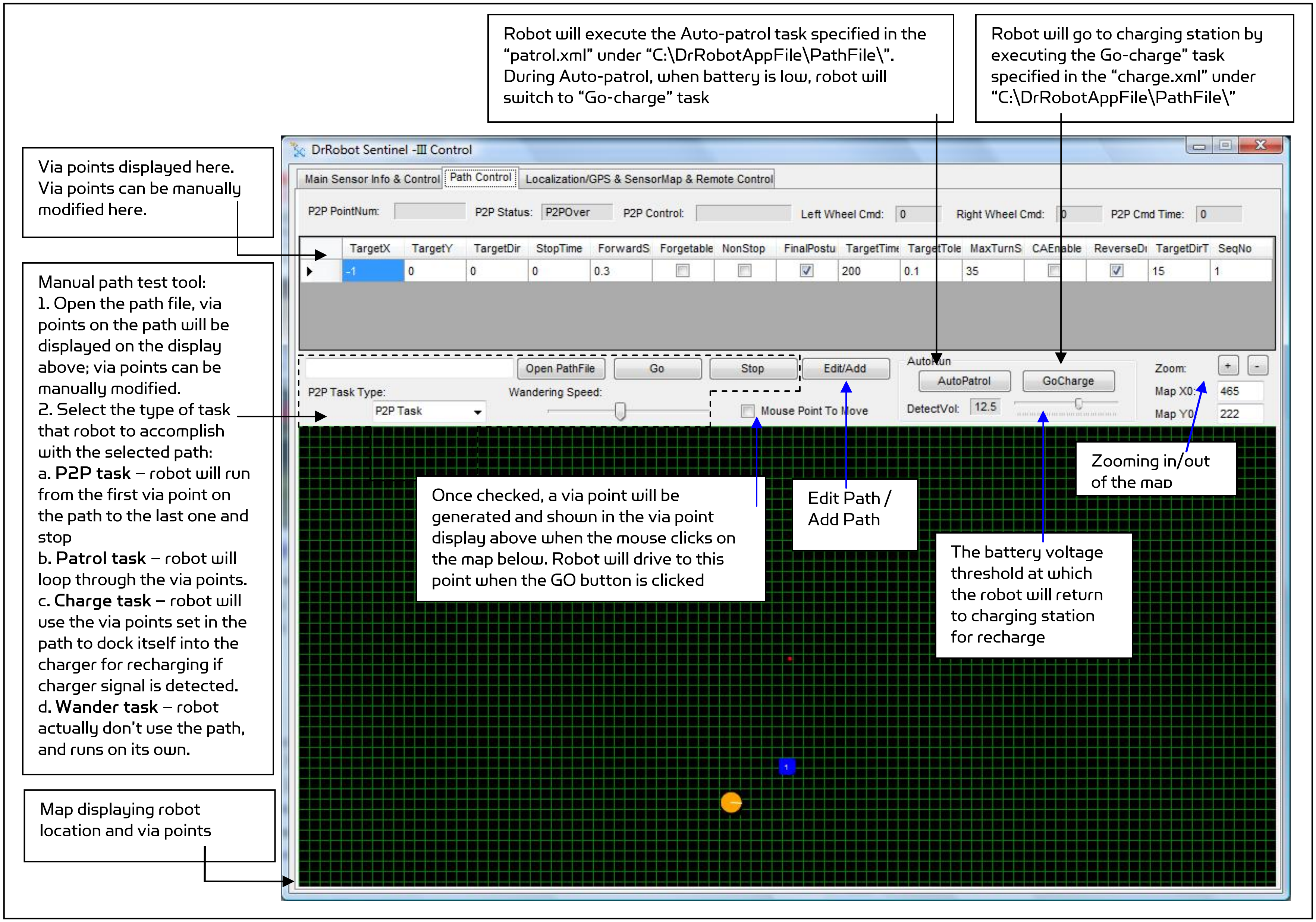


Red Button (#B) Suspend CA

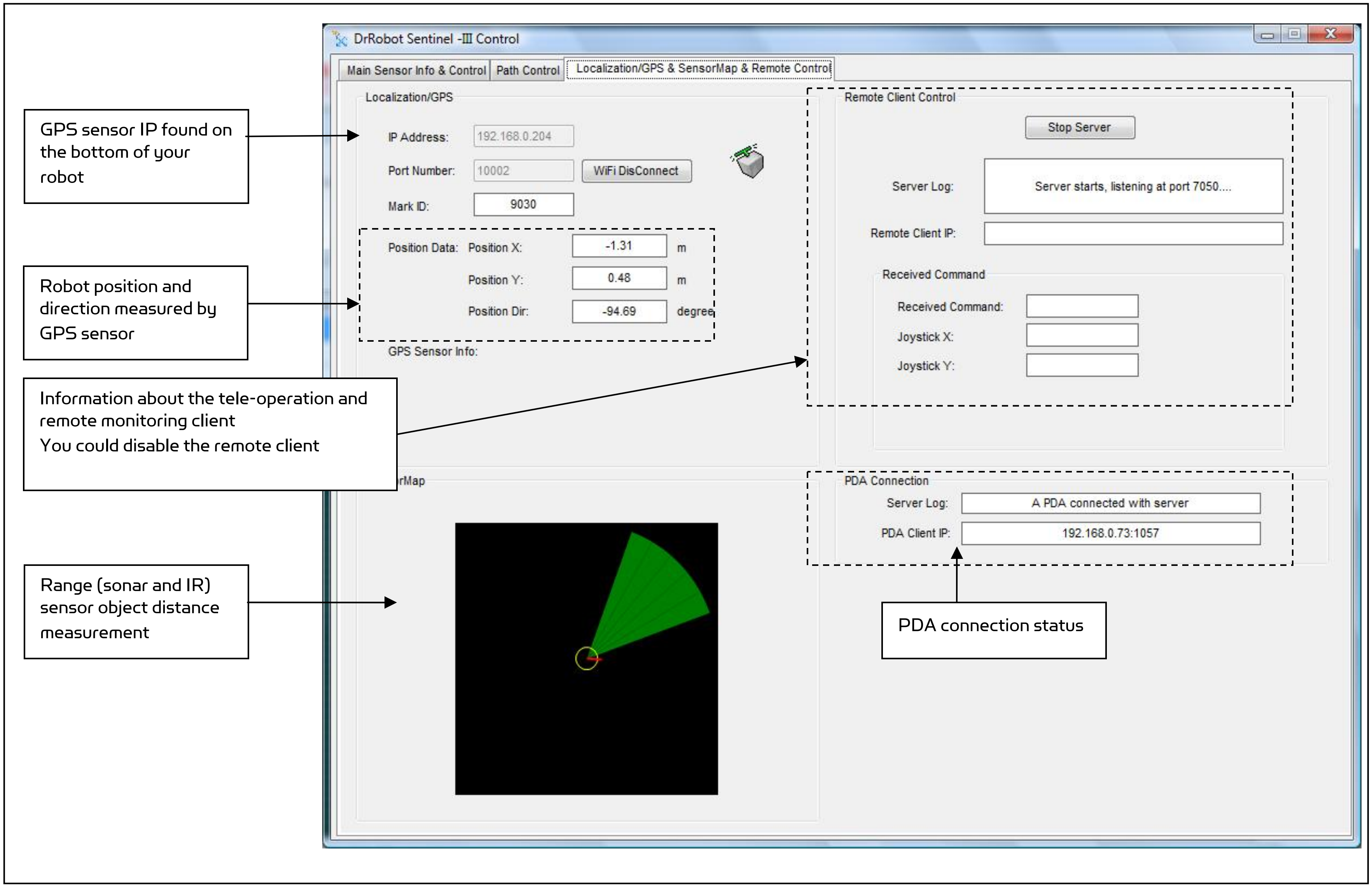
To Enable Gamepad Control

Robot base collision avoidance feature is enabled by default. During Gamepad operation, you could temporally disable this feature by holding the red button (#B) on the Gamepad handle (as shown above)

The second tag is "Path Control"



The third tag is "Localization/GPS & SensorMap & Remote Control"



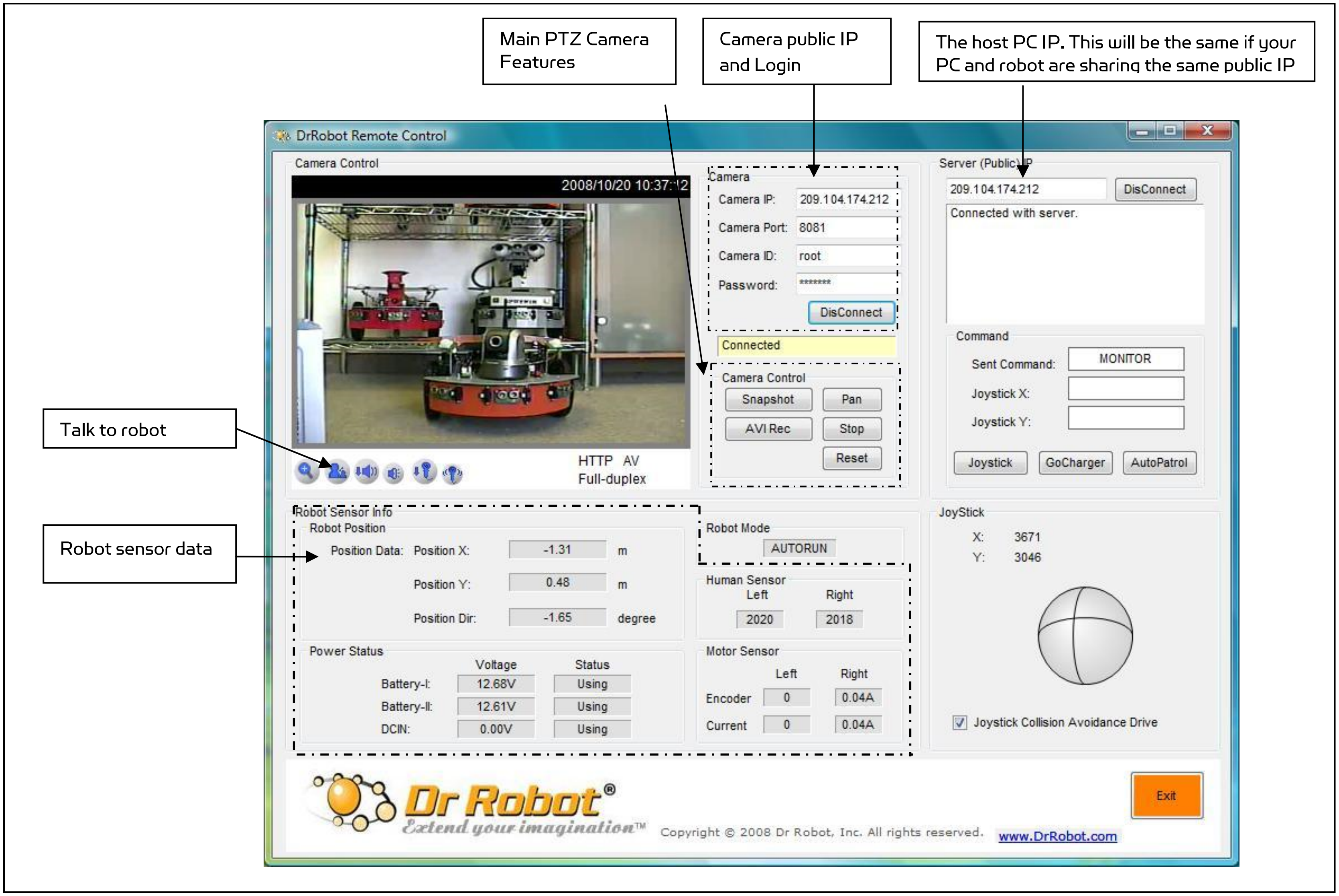
Tele-operation and Remote Control

If Internet remote monitoring/control is required, you need to connect the wireless router WAN port to your broadband Internet modem. You need to find out the public IP assigned by your ISP. (you should be able to find this information from the router status page) This IP will be used by the remote client to connect to the host PC and the devices on the robot.

If firewall is in-place in your network, you also need to make sure all the network ports used by the wireless devices (e.g. 8081, 5002, 5003 for the main PTZ camera and other devices in upgrade options) and 7050 on the server and remote client sides are not blocked for the Internet remote monitoring/control tasks to operate properly.

"Dr Robot Remote Control" program allows you to remotely control the robot, obtain main robot sensor information, view, listen and talk through robot.

Please contact support@DrRobot.com if you need further support on configuring your network for remote monitoring and tele-operation task.


Main PTZ Camera Features
Camera public IP and Login
The host PC IP. This will be the same if your PC and robot are sharing the same public IP
Talk to robot
Robot sensor data
DrRobot Remote Control
Camera Control
2008/10/20 10:37:12
HTTP AV
Full-duplex
Camera
Camera IP: 209.104.174.212
Camera Port: 8081
Camera ID: root
Password: *******
DisConnect
Connected
Camera Control
Snapshot
Pan
AVI Rec
Stop
Reset
Server (Public IP
209.104.174.212
DisConnect
Connected with server.
Command
Sent Command: MONITOR
Joystick X:
Joystick Y:
Joystick
GoCharger
AutoPatrol
Robot Sensor Info
Robot Position
Position Data: Position X: -1.31 m
Position Y: 0.48 m
Position Dir: -1.65 degree
Power Status
Voltage
Status
Battery-I: 12.68V Using
Battery-II: 12.61V Using
DCIN: 0.00V Using
Robot Mode
AUTORUN
Human Sensor
Left
Right
2020
2018
Motor Sensor
Left
Right
Encoder 0 0.04A
Current 0 0.04A
JoyStick
X: 3671
Y: 3046
Joystick Collision Avoidance Drive
Dr Robot®
Extend your imagination™
Copyright © 2008 Dr Robot, Inc. All rights reserved.
www.DrRobot.com
Exit

VII. Recharging

Autonomous Recharging

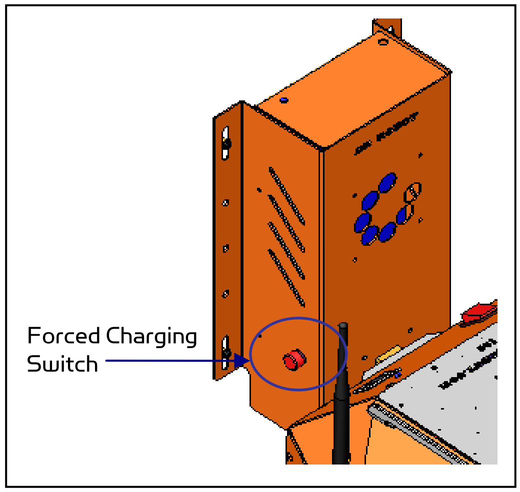
When the robot is running at auto Patrol mode, when the batteries are running low, robot will automatically return to the charger and charge.

Manual Command of Self-Docking Recharging

You could use the "GoCharge" button found on the "Sentinel III Control" or Remote Client software to dock the robot into the charging station and recharge.

Manual Forced Recharging

When manual recharge is required, you could manually dock the robot into the charging station or plug the charging cable from the charging station onto the secondary recharging socket on the back of the robot, with robot turned on, toggle on the "Forced Charging Switch" found on the lower right side of the charging unit as shown below (when on, the fan on the charger will run). The charging LED light on the back of the robot will light up. When charging is completed, the re-charging process will stop automatically. You need to manually switch off the "Forced Charging Switch" after use.



VIII. Further Development & Programming

The Sentinel Control program is written with C# program with Visual Studio 2008 express under .Net 3.5 framework. You could download the development tools (Visual Studio 2008 express under .Net 3.5 framework) free from Microsoft. Please refer to the "Dr Robot Application Development Notes on C# Programming For Robot Control" for further information.

The control program uses the supporting components and libraries that should have been installed when you install the control program from the installation CD:

1. **DRROBOTSentinelCONTROL.OCX:** Please refer to "WiRobot SDK API Reference Manual.pdf" for detail.
2. **WiRobotGateway.exe**
3. **DrRobotSensorMapBuilder.dll:** This dll file provides functions to build the environmental map for collision avoidance feature.
4. **DrRobotP2PSpeedDrive.dll:** This dll file provides functions to drive a robot form one specific point to another.
5. **DrRobotConstellation.dll**: Sentinel–III robot uses the sonar based Constellation indoor GPS localization system. This dll file provides the functions to locate the robot position with the Constellation system.
6. **DrRobotGPS.dll** Sentinel–III use the vision-landmark based indoor GPS localization system. This dll file provides the functions to locate the robot position with vision based GPS system.
7. **VitaminControl.dll** This is the camera control component for the Pan-Tilt-Zoom camera (P/N: AV-PTZ-VH) used for i90 robots such as Sentinel –II and Sentinel-III. Please refer to "PTZ Camera ActiveX Control Reference Manual.pdf" for detail.
8. **AXIS Media Control Library Set** These are the camera control component for the AXIS Mini Camera (P/N: AXCAM) used for the Dual-camera Head and Scout arm. Please refer to "AXIS Media Control SDK Help" for detail.

For support on development using Microsoft Robotics Studio, operation system other than MS Windows, or raw communication protocol, please contact support@DrRobot.com.

IX. Appendix I: “Patrol.xml”


```
<PointSet xmlns="http://tempuri.org/PointSet.xsd">

    <PointConfigTable>
        <TargetX>-1.963</TargetX>
        <TargetY>-4.783</TargetY>
        <TargetDir>180</TargetDir>
        <StopTime>5</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>false</NonStop>
        <FinalPosture>true</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>1</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>0.688</TargetX>
        <TargetY>-4.597</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>true</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
```


(continued...)


```
        <SeqNo>2</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>3.152</TargetX>
        <TargetY>-4.253</TargetY>
        <TargetDir>-90</TargetDir>
        <StopTime>5</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>false</NonStop>
        <FinalPosture>true</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>3</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>3.848</TargetX>
        <TargetY>-0.566</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>true</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
```

(continued...)


```
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
       <SeqNo>4</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>3.970</TargetX>
        <TargetY>3.768</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
       <SeqNo>5</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>2.157</TargetX>
        <TargetY>5.691</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
```


(continued...)


```
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
       <SeqNo>6</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>-0.149</TargetX>
        <TargetY>3.480</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
       <SeqNo>7</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>-2.571</TargetX>
        <TargetY>2.880</TargetY>
        <TargetDir>135</TargetDir>
        <StopTime>5</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>false</NonStop>
        <FinalPosture>true</FinalPosture>
        <TargetTime>200</TargetTime>
```

(continued...)


```
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>8</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>-0.685</TargetX>
        <TargetY>1.542</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>9</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>1.502</TargetX>
        <TargetY>0.12</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>true</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
```


(continued...)


```
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>10</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>1.512</TargetX>
        <TargetY>-1.830</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>true</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>11</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>0.175</TargetX>
        <TargetY>-1.821</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>true</NonStop>
```

(continued…)


```
            <FinalPosture>false</FinalPosture>
            <TargetTime>200</TargetTime>
            <TargetTolerance>0.05</TargetTolerance>
            <MaxTurnSpeed>75</MaxTurnSpeed>
            <CAEnable>true</CAEnable>
            <ReverseDrive>false</ReverseDrive>
            <TargetDirTolerance>5</TargetDirTolerance>
            <SeqNo>12</SeqNo>
      </PointConfigTable>

      <PointConfigTable>
            <TargetX>-1.651</TargetX>
            <TargetY>-1.559</TargetY>
            <TargetDir>135</TargetDir>
            <StopTime>5</StopTime>
            <ForwardSpeed>0.5</ForwardSpeed>
            <Forgetable>false</Forgetable>
            <NonStop>false</NonStop>
            <FinalPosture>true</FinalPosture>
            <TargetTime>200</TargetTime>
            <TargetTolerance>0.05</TargetTolerance>
            <MaxTurnSpeed>75</MaxTurnSpeed>
            <CAEnable>true</CAEnable>
            <ReverseDrive>false</ReverseDrive>
            <TargetDirTolerance>5</TargetDirTolerance>
            <SeqNo>13</SeqNo>
      </PointConfigTable>

      <PointConfigTable>
            <TargetX>-0.567</TargetX>
            <TargetY>-3.068</TargetY>
            <TargetDir>0</TargetDir>
            <StopTime>0</StopTime>
            <ForwardSpeed>0.5</ForwardSpeed>
```


(continued…)


```
            <Forgetable>false</Forgetable>
            <NonStop>false</NonStop>
            <FinalPosture>false</FinalPosture>
            <TargetTime>200</TargetTime>
            <TargetTolerance>0.05</TargetTolerance>
            <MaxTurnSpeed>75</MaxTurnSpeed>
            <CAEnable>true</CAEnable>
            <ReverseDrive>false</ReverseDrive>
            <TargetDirTolerance>5</TargetDirTolerance>
            <SeqNo>14</SeqNo>
      </PointConfigTable>
</PointSet>
```

X. Appendix II: “Charge.xml”


```xml
<PointSet xmlns="http://tempuri.org/PointSet.xsd">

    <PointConfigTable>
        <TargetX>1.502</TargetX>
        <TargetY>0.12</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>true</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>75</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>1</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>0.644</TargetX>
        <TargetY>-0.067</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>2</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>false</NonStop>
        <FinalPosture>true</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>90</MaxTurnSpeed>
        <CAEnable>true</CAEnable>
        <ReverseDrive>false</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
```


(continued…)


```xml
        <SeqNo>2</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>0.175</TargetX>
        <TargetY>-0.062</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>2</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>false</NonStop>
        <FinalPosture>true</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>90</MaxTurnSpeed>
        <CAEnable>false</CAEnable>
        <ReverseDrive>true</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>3</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>-0.179</TargetX>
        <TargetY>-0.063</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>2</StopTime>
        <ForwardSpeed>0.5</ForwardSpeed>
        <Forgetable>false</Forgetable>
        <NonStop>false</NonStop>
        <FinalPosture>true</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>45</MaxTurnSpeed>
```

(continued...)

```
        <CAEnable>false</CAEnable>
        <ReverseDrive>true</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>4</SeqNo>
    </PointConfigTable>

    <PointConfigTable>
        <TargetX>-0.502</TargetX>
        <TargetY>-0.063</TargetY>
        <TargetDir>0</TargetDir>
        <StopTime>0</StopTime>
        <ForwardSpeed>0.2</ForwardSpeed>
        <Forgetable>true</Forgetable>
        <NonStop>true</NonStop>
        <FinalPosture>false</FinalPosture>
        <TargetTime>200</TargetTime>
        <TargetTolerance>0.05</TargetTolerance>
        <MaxTurnSpeed>35</MaxTurnSpeed>
        <CAEnable>false</CAEnable>
        <ReverseDrive>true</ReverseDrive>
        <TargetDirTolerance>5</TargetDirTolerance>
        <SeqNo>5</SeqNo>
    </PointConfigTable>

</PointSet>
```

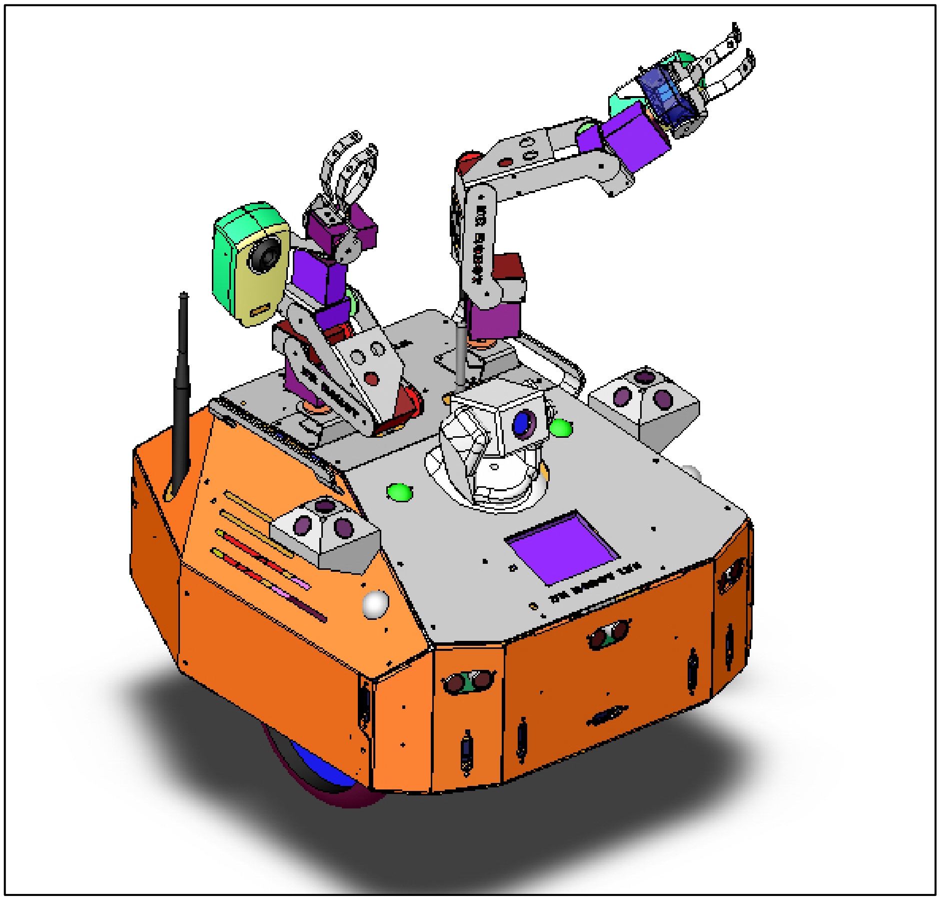
XI. Appendix III: WiFi Module Settings and Connection

There are two WiFi modules on the Sentinel robot. Each WiFi module connects to two serial devices through channel I and II (TCP/IP port 10001 and 10002 respectively). They are pre-configured as below:

Name	**Robot WiFi Module 1**	**IP**	192.168.0.203
SSID **WEP** **KEY**	dri 128bits 112233445566778899AABBCCDD	**Gateway** **Subnet Mask** **Key Type**	192.168.0.200 (Router IP) 255.255.255.0 Open Key
Channel-I (10001)	115200, 8,N,1, no flow control, UDP, Datagram 01, remote IP:0.0.0.0	**Channel-II (10002)**	115200, 8,N,1, no flow control, UDP, Datagram 01, remote IP:0.0.0.0

Name	**Robot WiFi Module 2**	**IP**	192.168.0.204
SSID **WEP** **KEY**	dri 128bits 112233445566778899AABBCCDD	**Gateway** **Subnet Mask** **Key Type**	192.168.0.200 (Router IP) 255.255.255.0 Open Key
Channel-I (10001)	115200, 8,N,1, no flow control, TCP, remote IP:0.0.0.0	**Channel-II (10002)**	115200, 8,N,1, no flow control, TCP, remote IP:0.0.0.0

XII. Appendix IV: Scout Arm (optional)



Network Settings

Name	**On-arm Camera (right)**	**IP (Port)**	192.168.0.99 (8082)*
SSID	Dri	**Gateway**	192.168.0.200 (Router IP)
WEP	128bits	**Subnet Mask**	255.255.255.0
KEY	112233445566778899AABBCCDD	**Key Type**	Open Key
Login	root	**Password**	drrobot

Name	**Arm IP**	**IP (Port)**	192.168.0.204 (10001)*
SSID	Dri	**Gateway**	192.168.0.200 (Router IP)
WEP	128bits	**Subnet Mask**	255.255.255.0
KEY	112233445566778899AABBCCDD	**Key Type**	Open Key

*IPs for your robot may be different, they can be found on the bottom of your robot.

Router and Firewall Settings for Internet Remote Monitoring/Tele-operation

If Internet remote monitoring/control is required, you need to connect the wireless router WAN port to your current broadband Internet modem (this is optional and is not required for running the robot).

If firewall is in-place in your network, you also need to make sure all the network ports used by the wireless on-arm camera (i.e. 8082) on the server and remote client sides are not blocked for the Internet remote monitoring/control tasks to operate properly.

You also need to configure the virtual server settings in your router to include the on-arm camera:

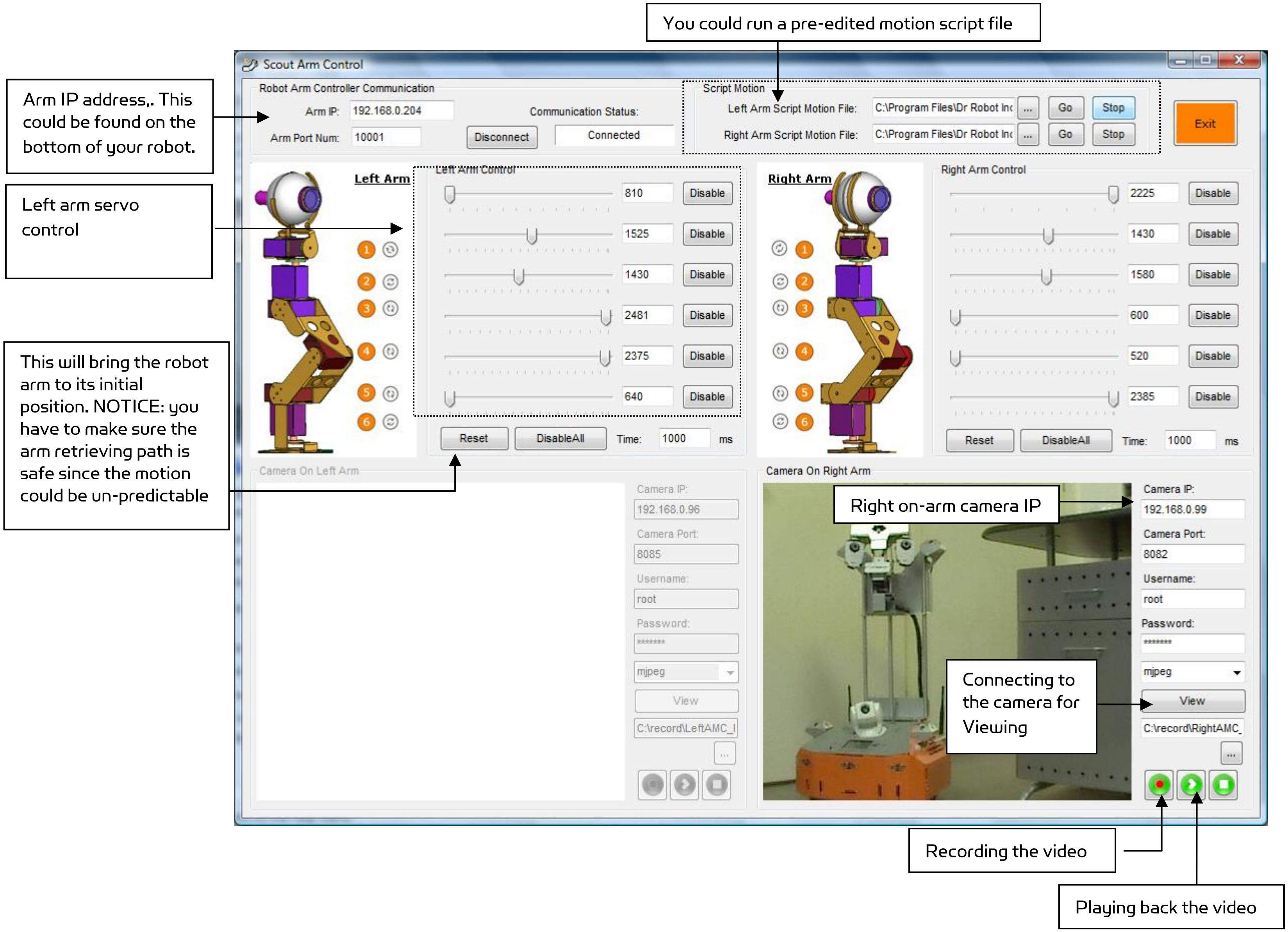
Virtual Server	Port	Protocol	Server IP
Right Arm Camera	8082	TCP/IP	192.168.0.99 (Right Arm Camera IP)

Servo Control

Each Scout arm has five joints and 1 gripper. On-arm camera could be mounted on each arm's wrist. Scout arms use RC servo motors. RC Servo motor position is defined by the pulse width from the control signal line. The pulse width value in milliseconds for specific position (e.g. 0°, 90° and 180°) is servo manufacturer and model depending.

The conversion between the servo command in the "Scout Arm Control" program and the pulse width is: The servo command = 2250 * Pulse Width in millisecond.

In the "Scout Arm Control" program as shown below, the description file, ArmServoConfig.xml defines the max, min and initial servo position (in servo command).


You could run a pre-edited motion script file
Arm IP address,. This could be found on the bottom of your robot.
Left arm servo control
This will bring the robot arm to its initial position. NOTICE: you have to make sure the arm retrieving path is safe since the motion could be un-predictable
Right on-arm camera IP
Connecting to the camera for Viewing
Recording the video
Playing back the video
Scout Arm Control
Robot Arm Controller Communication
Arm IP: 192.168.0.204
Arm Port Num: 10001
Disconnect
Communication Status:
Connected
Script Motion
Left Arm Script Motion File:
Right Arm Script Motion File:
Go
Stop
Exit
Left Arm
Left Arm Control
Right Arm
Right Arm Control
Reset
DisableAll
Time:
1000
ms
Camera On Left Arm
Camera On Right Arm
Camera IP:
192.168.0.96
192.168.0.99
Camera Port:
8085
8082
Username:
root
Password:
mjpeg
View

Sample left arm motion script file:

```
<ServoMotionTable
xmlns="http://tempuri.org/ServoMotionTable.xsd">
    <ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>1490</Servo3>
        <Servo4>2500</Servo4>
        <Servo5>2410</Servo5>
        <Servo6>640</Servo6>
    </ServoMotionStep>
    <ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>1403</Servo3>
        <Servo4>2215</Servo4>
        <Servo5>2046</Servo5>
        <Servo6>640</Servo6>
    </ServoMotionStep>
    <ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>1403</Servo3>
        <Servo4>2215</Servo4>
        <Servo5>2046</Servo5>
        <Servo6>1509</Servo6>
    </ServoMotionStep>
```

(Continued...)

```
<ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>1403</Servo3>
        <Servo4>2120</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>1509</Servo6>
    </ServoMotionStep>
    <ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>1403</Servo3>
        <Servo4>1815</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>1509</Servo6>
    </ServoMotionStep>
    <ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>1403</Servo3>
        <Servo4>1815</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>2401</Servo6>
    </ServoMotionStep>
```

(Continued...)

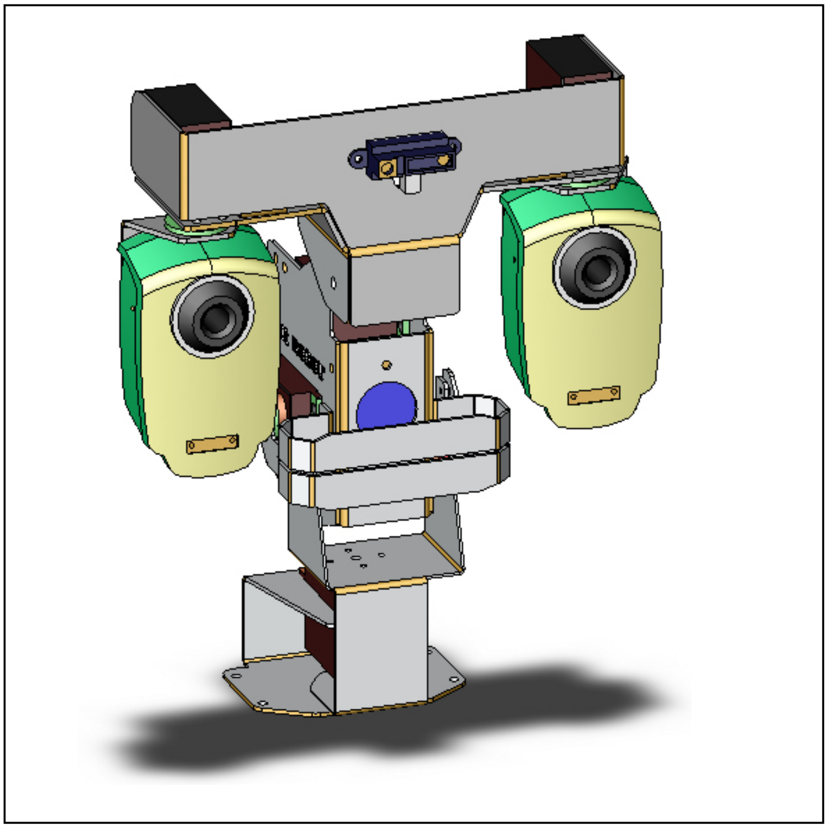
```
<ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>2102</Servo3>
        <Servo4>1815</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>2401</Servo6>
    </ServoMotionStep>
    <ServoMotionStep>
        <Servo1>1450</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>2102</Servo3>
        <Servo4>1815</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>2401</Servo6>
    </ServoMotionStep>
    <ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>2102</Servo3>
        <Servo4>1815</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>2401</Servo6>
    </ServoMotionStep>
```

(Continued...)

```
<ServoMotionStep>
        <Servo1>1450</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>2102</Servo3>
        <Servo4>1815</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>2401</Servo6>
    </ServoMotionStep>
    <ServoMotionStep>
        <Servo1>810</Servo1>
        <Servo2>1525</Servo2>
        <Servo3>2102</Servo3>
        <Servo4>1815</Servo4>
        <Servo5>1568</Servo5>
        <Servo6>2401</Servo6>
    </ServoMotionStep>
</ServoMotionTable>
```

XIII. Appendix V: Dual-camera Animated Head (optional)

This optional head comes with dual high resolution camera and a 6 degree head.



Network Settings

Name	Left Eye Camera	IP (Port)	192.168.0.97 (8083)*
SSID WEP KEY	dri 128bits 112233445566778899AABBCCDD	Gateway Subnet Mask Key Type	192.168.0.200 (Router IP) 255.255.255.0 Open Key
Login	root	Password	drrobot

Name	Right Eye Camera	IP (Port)	192.168.0.98 (8084)*
SSID WEP KEY	dri 128bits 112233445566778899AABBCCDD	Gateway Subnet Mask Key Type	192.168.0.200 (Router IP) 255.255.255.0 Open Key
Login	root	Password	drrobot

*IPs for your robot may be different, they can be found on the bottom of your robot.

Router and Firewall Settings for Internet Remote Monitoring/Tele-operation

If Internet remote monitoring/control is required, you need to connect the wireless router WAN port to your current broadband Internet modem (this is optional and is not required for running the robot).

If firewall is in-place in your network, you also need to make sure all the network ports used by the wireless eye camera devices (i.e. 8083 and 8084) on the server and remote client sides are not blocked for the Internet remote monitoring/control tasks to operate properly.

You also need to configure the virtual server settings in your router to include the eye cameras:

Virtual Server	Port	Protocol	Server IP
Left Eye Camera	8083	TCP/IP	192.168.0.97 (Left Eye Camera IP)
Right Eye Camera	8084	TCP/IP	192.168.0.98(Right Eye Camera IP)

Servo Control

Animated head uses RC servo motors. RC Servo motor position is defined by the pulse width from the control signal line. The pulse width value in milliseconds for specific position (e.g. 0°, 90° and 180°) is servo manufacturer and model depending.

The conversion between the servo command in the "Dual-camera Head Control" program and the pulse width is: The servo command = 2250 * Pulse Width in millisecond.

In the "Dual-camera Head Control" program as shown below, the description file, HeadServoConfig.xml defines the max, min and initial servo position (in servo command).


Dual Camera Head Control
Head Control
Servo Control
Command Value
Time(ms)
Head Tilt
Head Pan
Mouth
Eye Tilt
Left Eye Pan
Right Eye Pan
3200
3250
2470
4100
2700
4250
1000
Disable
Exit
DisableAll
Reset
Motion Script File:
Left Eye Camera
Right Eye Camera
Camera IP:
192.168.0.97
192.168.0.98
Camera Port:
8083
8084
Username:
root
Password:
mjpeg
View
C:\record\LeftAMC_F
C:\record\RightAMC_
Head servo control
You could run a pre-edited motion script file
This will bring the head to its initial position. NOTICE: you have to make sure the head retrieving path is safe since the motion could be un-predictable
Left eye camera IP
Viewing window for the left eye camera
Connecting to the camera for Viewing
Viewing window for the left eye camera
Recording the video
Playing back the video