Wifibot Lab

Quick Start Guide

Thank you for choosing the **wifibot xpe** platform for your robotic application.

- Before using the platform, please read with care this manual
- Keep this manual in a safe place for any future reference
- For updated information about this product visit the official site of wifibot http://www.wifibot.com

Index

Package contents

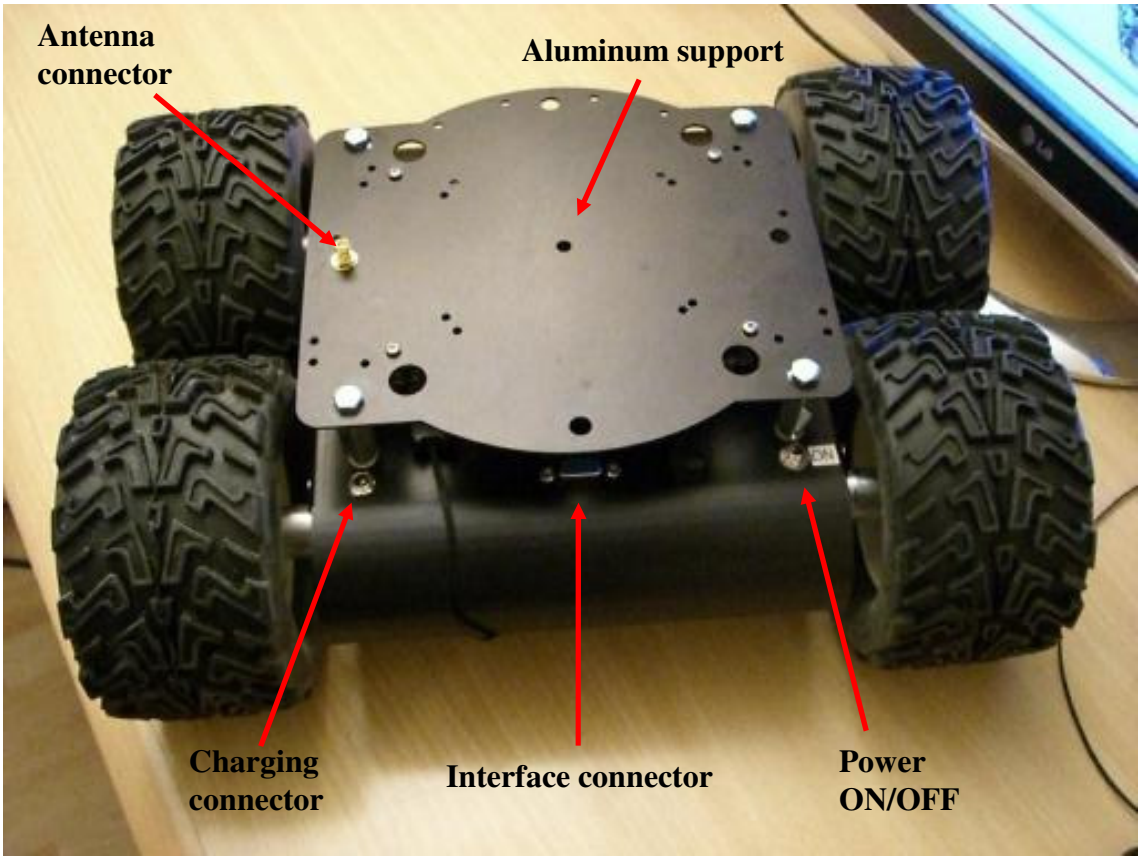
Make sure to be in possession of all the articles mentioned below. If any of them should be missing, contact your reseller as soon as possible.

USB Platform
IP camera (Optional)
Battery charger
Wifibot CDROM
Camera CD-ROM and documentation
1x charging cable
1x RJ45 cable for the IP camera

Quick start

1- Install the simple interface or RTMIX multi-robot interface and reboot your computer if RTMIX.
2- Switch ON the robot or robots.
3- Set you IP settings (see page 12) for example: 192.168.1.25 mask 255.255.255.0
4- Connect to the robots' ad-hoc network (see page 13)
5- (Optional if mesh needed) Launch OLSR, select the appropriate network device and press **start**.
6- Launch the interface

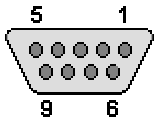
Platform overview



Platform interfaces

The 5V,GND and 9.6V power output:

A 5V output, GND and a 9.6V filtered battery output are accessible at a female DSUB-9 connector located at the front of the robot. Pin 1-2 are 9.6V, pin 3-4-5 are 5V and 6-7-8-9 are GND output respectively. The 5V output can't give more than 6A and a maximum of 6A is recommended too for the 12V output. An incorrect use of this connector beyond those values (short circuit or other) can provoke a malfunction of the platform or of the DC/DC converter and even damage it.



The charging connector:

This connector located on the left at the rear of the robot, presents directly the + and – of the platform batteries. When charging the platform make sure the power switch is OFF as it drives a relay for connecting the connector to the batteries.

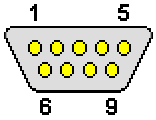
The ON/OFF switch:

The platform is switched ON and OFF by the interruptor located on the right at the back of the platform.

Interface connector:

This DSUB-9 male connector presents a mix of input-output signals. The pinout is the following:

1- 5V	5- I/O or ADC
2- ADC Sharp1	6- SDA (I2C)
3- ADC Sharp2	7-8- GND
4- I/O or ADC	9- SCL (I2C)



The antenna connector:

This is the wi-fi antenna connector. Screw the antenna carefully on the connector till the end.

Charge:

A battery charger is included with the platform. First make sure the platform and the charger are OFF, then connect the plugs of the charging cable and finally switch the charger ON and set the charging current.

Computer and camera installation

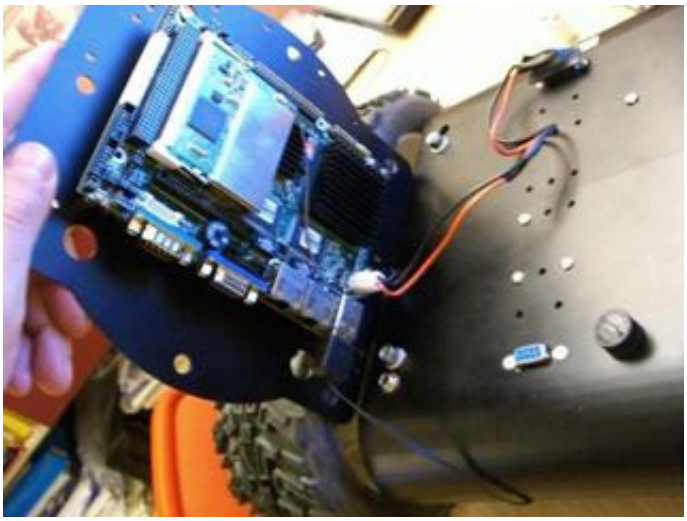
The platform is sold with an IP camera and an embedded computer which model can vary depending of the version. Those are independent elements from the platform which can be replaced by any other model. For more information about your particular camera and embedded computer please refer to their respective manuals included in the CD ROM of the robot. The top aluminum support of the platform has been thought for the fixation of those and other user components.

Their installation takes place as follows:

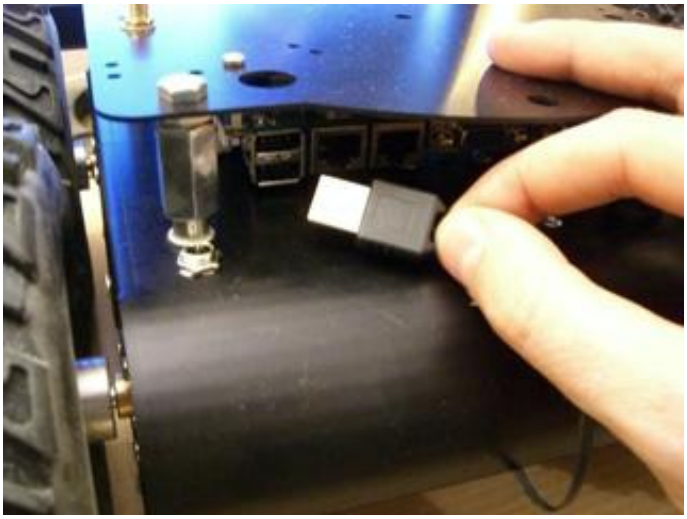
Unscrew the upper aluminum support:

Fix the embedded computer on the down part of the support :

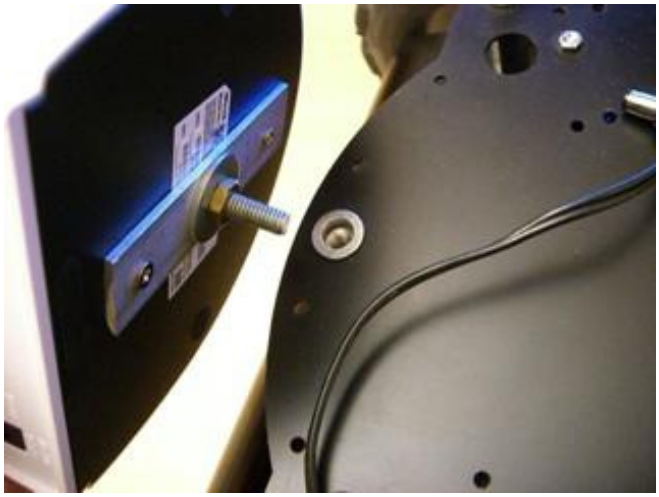
Connect the power cable of the computer to the appropriate connector:



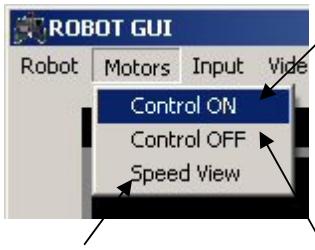
Connect the USB connector to the computer.



Srew back the aluminum support on top of the platform and screw the IP camera on top of the support.



The control software (TCP) :



Motor Control ON: Activates the speed control, Input_Left and Input_Right set on the dialog will be applied.

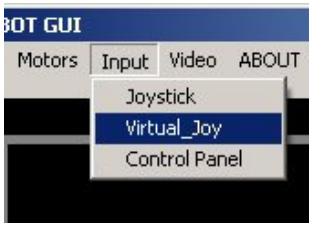
Speed View: Plots in real time the speed signal from the code wheels.

Motor Control OFF: Deactivates the speed control.

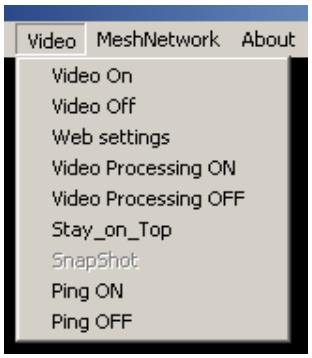
The control software:

The control software can be found in the CDROM in **..\Software\control software**

- Install if necessary the **Video Decoder** present in the same folder.
- Launch the **WifibotGUI** program.
- Click on **Robot** then **Settings**. The **Robot Settings** window appears.
- Set the **Control Server IP** and the **Control Server Port** which by default is **15000**.
- Set the **Camera IP** and the **Camera Port** which for the image is by default **80**.
- Select the proper **Camera Type**.
- Click on **Video**, then select **VideoOn**. The image from the camera will appear.
- Click on **Robot** then **Connect.**
- Click on **Input** then select **Joystick** or **Virtual_joy**. The robot can now be operated.

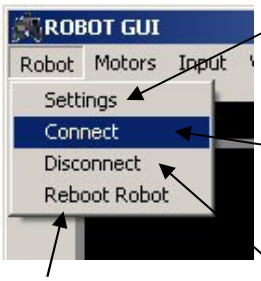


Input Selections (control panel for calibrating the joystick)



Video selections: Allows to configure and control some options of the camera.

The menu options:

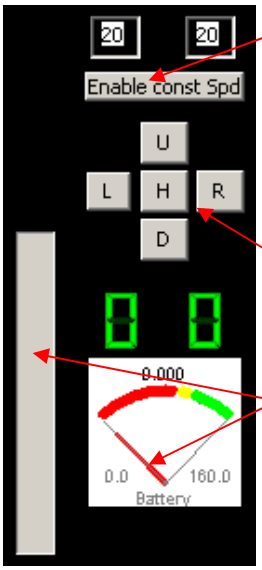


Settings: IP settings of the Control Server and the Camera.

Connect: Starts the communication with the Control Server.

Disconnect: Stops the communication with the Control Server.

Reboot: Reboots the robot's CPU.



Current input: shows the current input or allows to set it manually with keyboard.

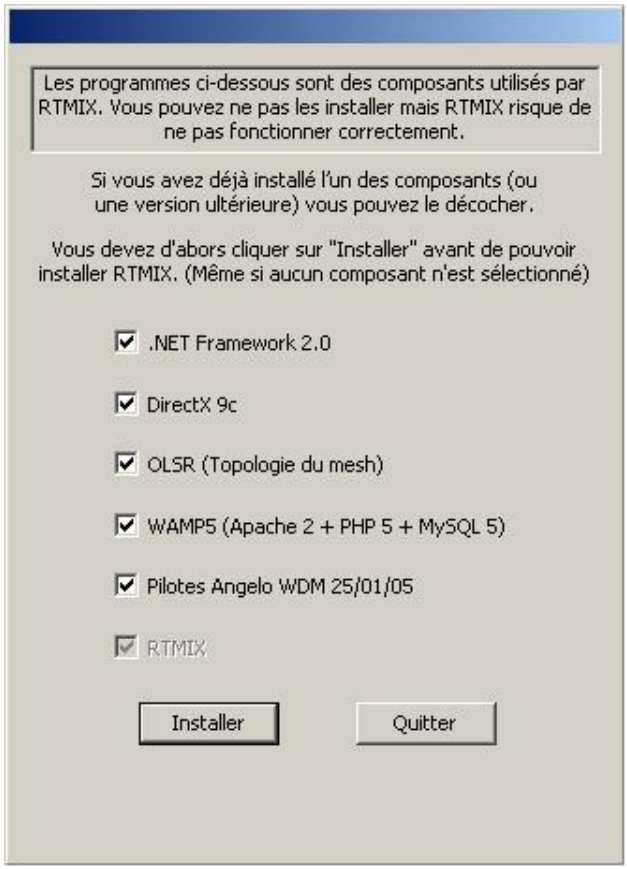
Pan-Tilt camera control: The red button takes the camera to the default position. You can click on the image too for moving the camera.

Sensor feedback: shows the data retrieved from the range sensors, the battery level and the speed of the robot.

RTMIX multi-robot interface (Optional)

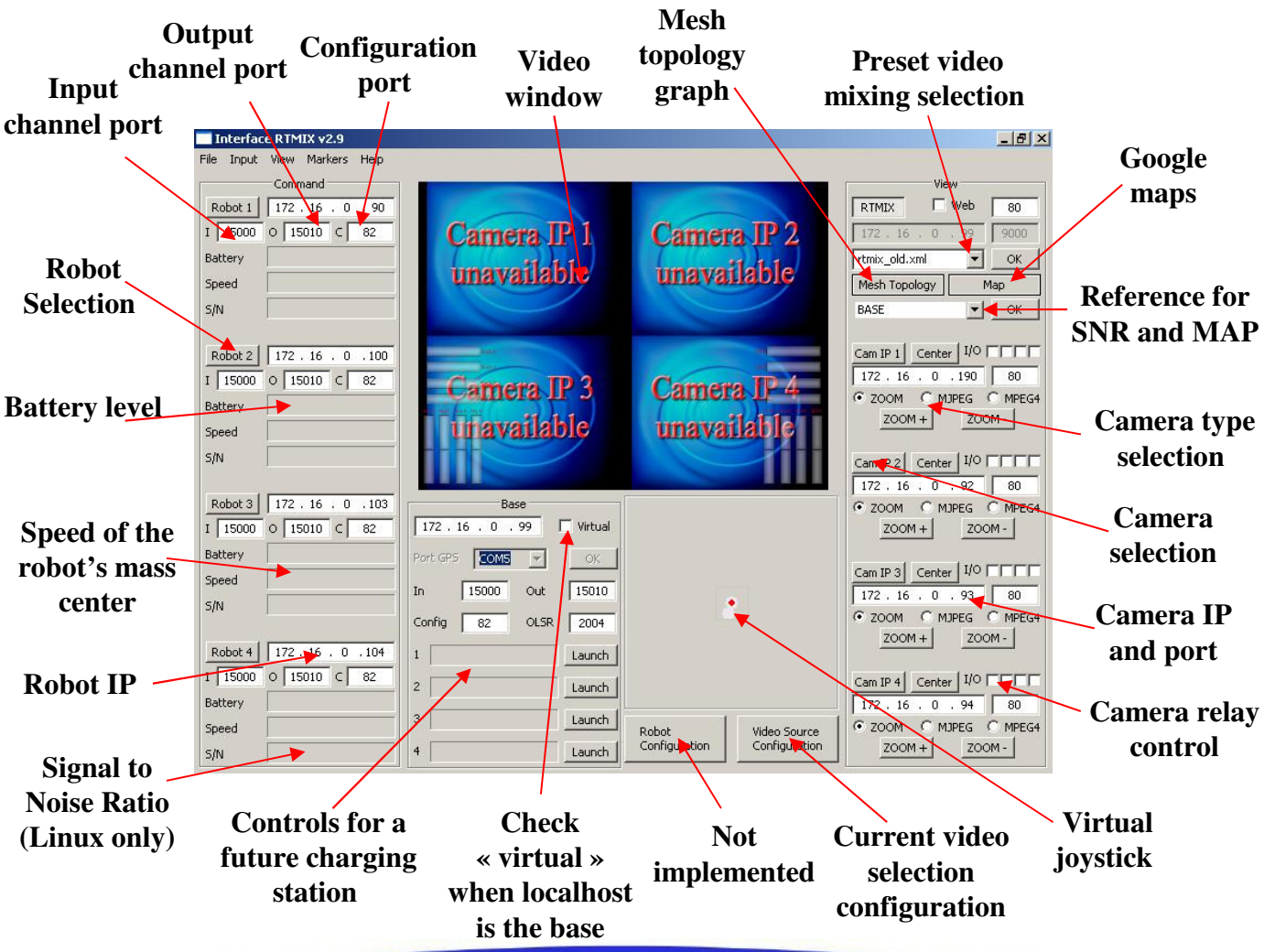
Installation:

1- Unpack the zip file.
2- Double-click on Setup.
3- Unselect from the program list those you may already have installed on your computer.
4- Click on the « install » button.
5- When asked to reboot your computer always say « no ».
6- Once installation is done, create a folder « rtmix » in the web shared folder of the web server installed on your computer. If you have followed the standard installation this would be « C:/wamp/www/ »
10- Copy the containt of «C:/Program Files/RTMIX/monsite/» into the «rtmix» folder you just created, ex:«C:/wamp/www/rtmix/»
11- Reboot your computer.

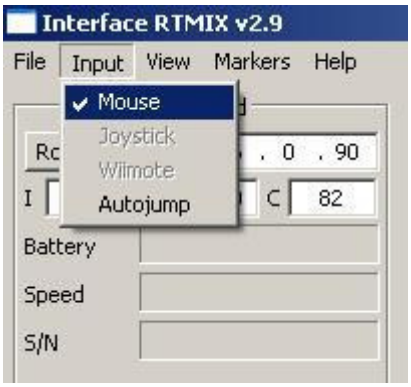


Features:

- The RTMIX multi-robot interface allows the user to control a team of up to four robots.
- The interface is best viewed with a 1024x768 resolution.
- Platform related controls are located on the left while visual related ones are located on the right.
- Before operation the user has to make sure all IP addresses and ports are correctly set.
- Selected robots can be controlled using the virtual joystick, a joystick or a wiimote, selecting more than one robot at a time will have as a result having all of them receiving the same command.
- Cameras can be selected individually or blended in one image with RTMIX. Preset mixing layouts can be selected and new ones added thanks to the "video source configuration" webpage.
- Certain functionalities need a reference which can be selected with a menu located just under the Mesh and Map buttons.
- Certain types of supported cameras have incorporated relays which are set with the I/O checkbox.
- The "base" can be either an external gateway or the control computer itself (check "virtual" for this).
- The Map button needs to have an active internet connection and will show the robot's and the base position on a Google Maps webpage provided those are equipped with a GPS.



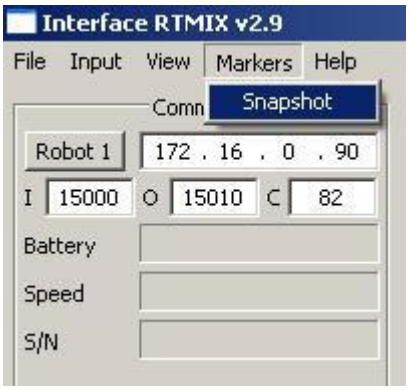
Program menu:

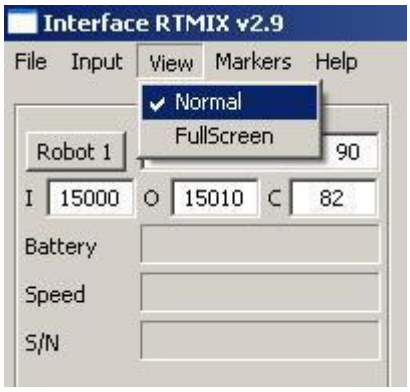


The **input** menu allows to choose between four modes of controlling the robots. Those are:
- mouse (virtual joystick)
- joystick
- wiimote
- autojump (virtual joystick)

This is a mode where the robot advances a certain distance with a set speed and radius.

The **markers** menu allows for now only to take snapshots of the current video window and to place a marker on the map at the location it was taken. The snapshot can then be seen by clicking on the marker.





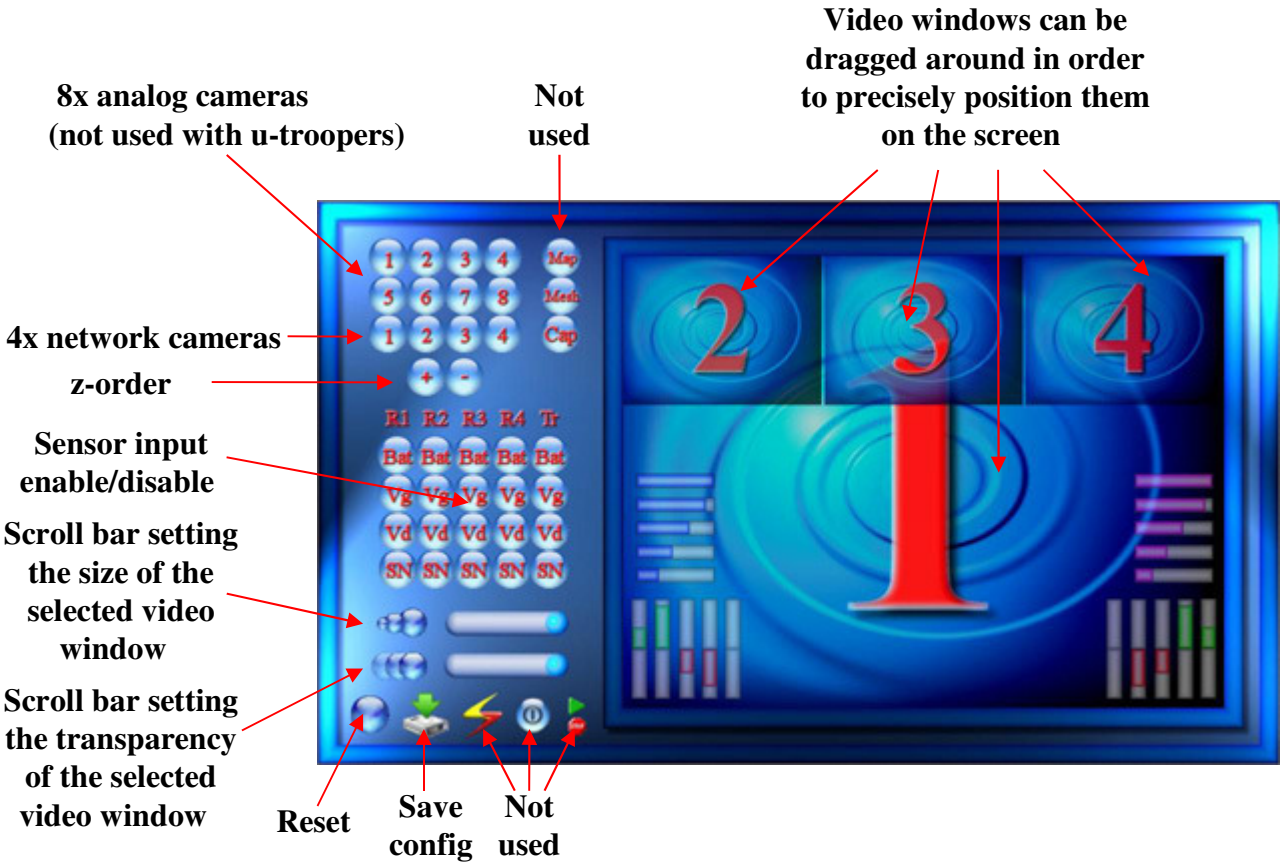
The **view** menu allows to choose between two modes of presentation:
- Normal view (default)
- Fullscreen view

This is a mode where the video windows covers the whole screen. It is particularly useful when using configuration webpages or driving the robots.

Configuration:

The configuration webpage is accessible clicking on the “video source configuration” button when RTMIX is selected.
The video mixing capabilities were originally designed to be embedded in larger robots and controlled remotely. They were only later adapted to the wifibot system, this is the reason why some options such as the analog cameras are not applicable.
To create a custom layout to be added to the interface menu do the following:

- Select the cameras and sensors to be displayed.
- Drag the cameras and the sensors at the desired location.
- Set the size, opacity and z-order of the different cameras.
- Press the save button and introduce a name.



Connecting to the robot:

Configuring your ethernet/wireless adapter:

By default, the robot has been pre-configured with certain IP addresses. Before connecting to the robot you may need to adjust the IP settings of the network adapter of your computer. Make sure all the devices in a same network having to communicate with the robot have the same class of address.
To adjust the TCP/IP settings of the network adapter:

1- Right-click on **My Network Places** in the **Start** menu, then select **Properties** from the pop-up menu. **The Network and Dial-up Connections** window appears.(**Fig1**)

2 - Create a network bridge between your interfaces or else disable all the network adapters except the one you want to use for connecting to the robot (Ethernet or Wi-Fi). Right-click the network adapter, then select **Properties** from the pop-up menu. (in **Fig1**)

3 - Double-click the **Internet Protocol (TCP/IP)** item to display the **Internet Protocol (TCP/IP) Properties** window. (in **Fig2**)

4 - Check the **Use the following IP address** option, then enter the IP address for the network adapter. **Set IP address** depending on the case : (in **Fig3**)

If you are connecting to a robot under Linux with a cable directly to an ethernet port, that is to the LAN, then enter 192.168.0.x (x can be any number between 1 and 254 except 250 and those used by the CPU and the camera of the robot). For example, a Wifibot Serial Number: S/N Y-AXX will have as IP for the CPU 192.168.0.1XX and 192.168.0.XX for the camera, those number are therefore not available. Set the **Subnet Mask** to 255.255.255.0 and leave **Default gateway** and **DNS** empty.

Fig 1

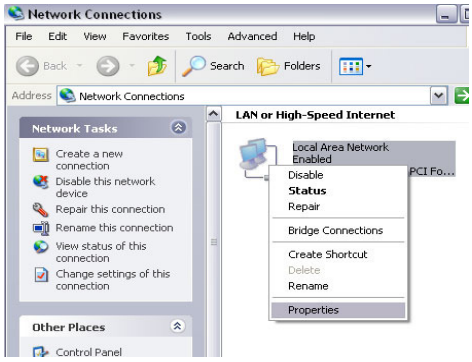


Fig 2

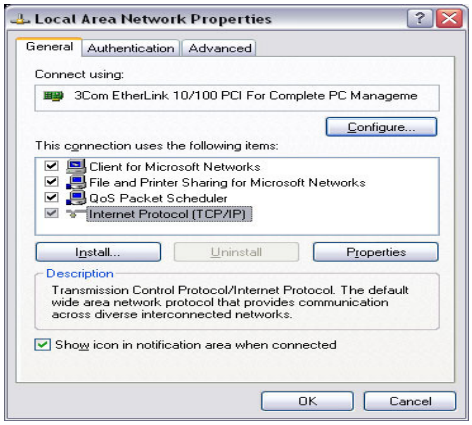
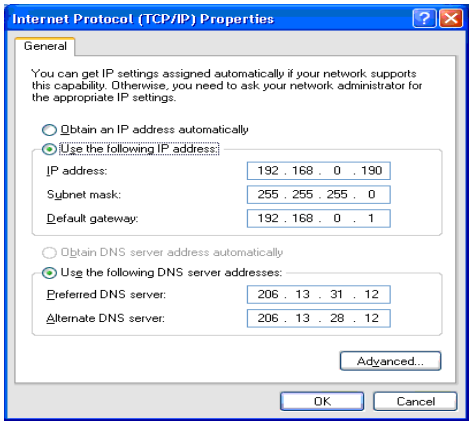


Fig 3



If you are connecting to a robot under windows or wirelessly under Linux, that is to the WLAN, then enter 192.168.1.x (x can be any number between 1 and 254 except those used by robots or other devices). For example, a Wifibot Serial Number: S/N Y-AXX will have as IP 192.168.1.1XX. Set the **Subnet Mask** to 255.255.255.0
5 - Click **OK** when finished.

Connecting your wireless adapter to the robot:

Once you have adjusted the TCP/IP settings, if you are using a cable and providing the robot is switched **ON** then you are already connected but if you are connecting wirelessly you have to make sure your wireless adapter is connected to the robot and not to something else. This can be done through windows or using the software provided with your adapter.

To connect your wireless adapter to the robot using windows, follow these steps:

- Switch **ON** the robot and wait a few seconds.
- In the **Network and Dial-up Connections** window,right-click on the wireless network adapter,then select **Properties** in the pop-up menu. The **Wireless Network Connection Properties** window appears.(**Fig1**)
- Click the **Wireless Networks** tab. A list of wireless access points appears in the **Available networks** box. (in **Fig1**)
- If the **wifibot** network is not listed in the **Available networks**, then click **Refresh** till it is. (in **Fig1**)
- Check the **Use Windows to configure my wireless network settings** option.(in **Fig1**)
- Click **OK** when finished.
- Right-click on the wireless network adapter again, then select **View Available Wireless Networks** from the pop-up menu (in **Fig2**). The **Connect to Wireless Network** dialog box opens with a list of available networks in the **Available networks** box. (in **Fig3**)
- Select the **wifibot** network from the list, then check **Allow me to connect to the selected wireless network, even though it is not secure** option. (in **Fig3**)
- Click **Connect** (in **Fig3**), a pop-up window at the bottom of the screen should appear indicating you are connected. (in **Fig4**)

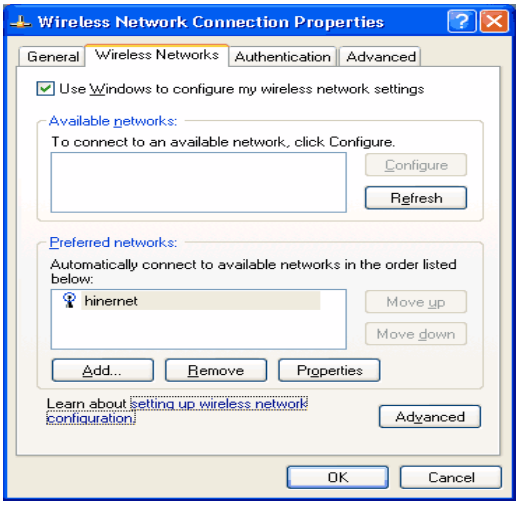


Fig 1

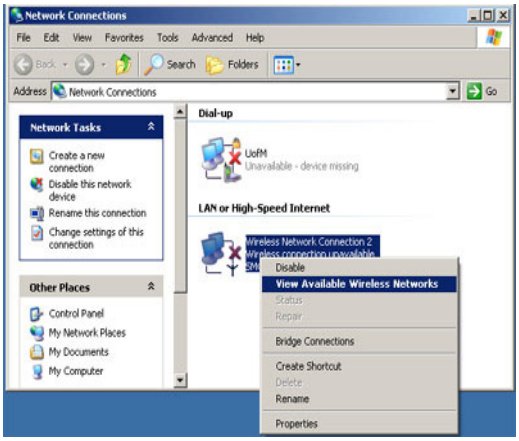


Fig 2

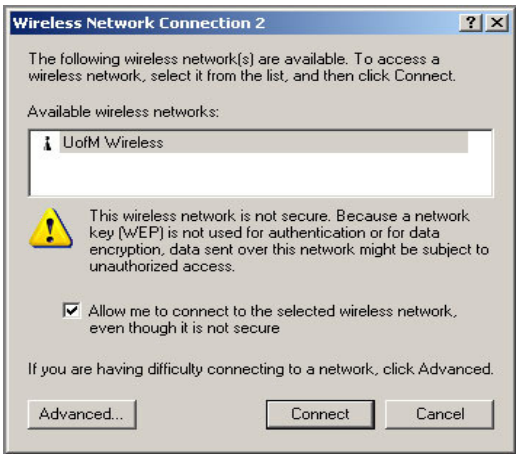


Fig 3

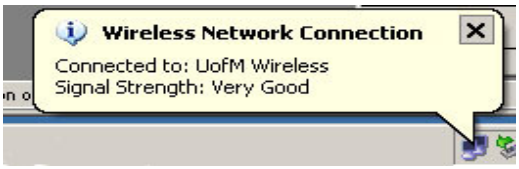


Fig 4

Networking

Network architecture:

In the micro-trooper, the embedded CPU works as a gateway between the internal wired LAN and the external wifi WLAN. The CPU has at least one ethernet card and one wireless card that form two separate networks (LAN/WLAN).The LAN and the WLAN should have in general a different address class and therefore data needs to be routed between them. Depending if you have chosen a robot under Windows or Linux the problem of connecting the two networks has been solved differently. Under Windows this has been done by configuring a bridge between the network interfaces, by doing so the robot's CPU appears to have a unique network interface and uses one single IP address. Under Linux, the interconnection is done through Dynamic NAT (Network Address Translation) and the CPU uses two different IP adresses, one for the internal LAN and one for the WLAN. In both cases, all local components of the robots such the IP camera will have their own IP address within the LAN, but when it comes to accessing them from the WLAN the method will differ. Under Windows as there is in practice no distinction between the WLAN and the LAN, every internal component will be reached using its own IP address (see **Fig1**). Under Linux, only the robot's CPU WLAN IP address can be seen and any internal network element will have to be reached using this single IP. In order to be able to access the separate devices using a single IP, we will need to assign to each of them a separate port (see **Fig2**). This will require to configure the CPU with the proper routing table (see pag 18).

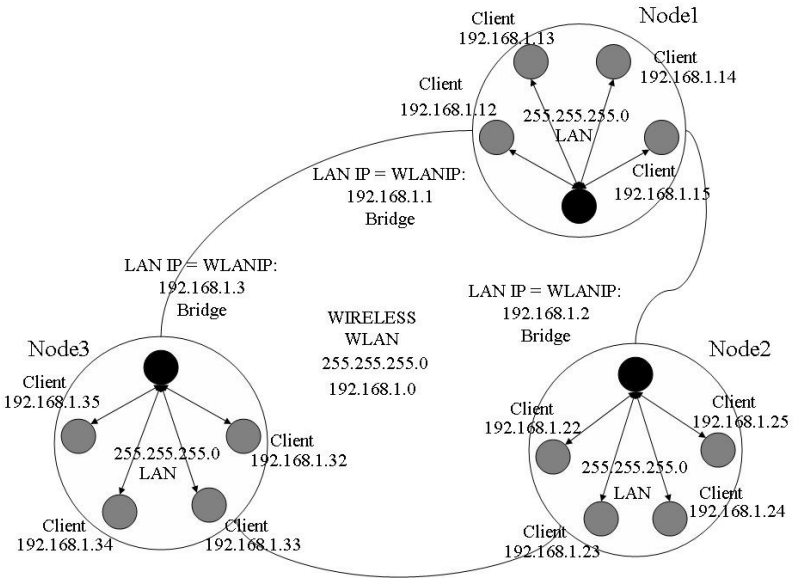


Fig 1

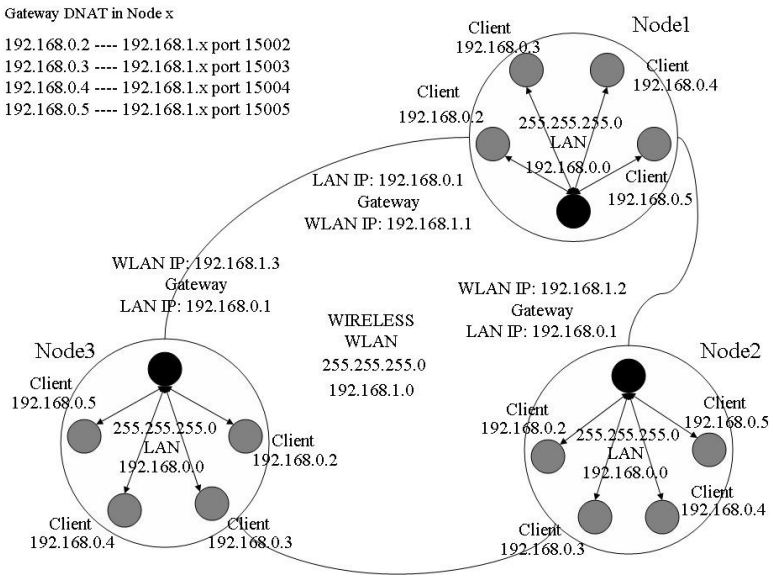


Fig 2

WLAN modes:

Let's have here a quick overview of the different modes Wi-Fi adapters can be configured :

-Infrastructure Master (Access Point)
-Infrastructure Managed (Adapter/Bridge)
-Ad-hoc without routing algorithm
-Ad-hoc with the OLSR routing algorithm
(Mesh Networking)

In infrastructure mode we have a master/slave structure where all the data is centralized in one device called access point (server/master) to which different adapters (clients/slaves/managed) connect. A client cannot talk directly to another but has to pass by the access point which will forward the data to the destination. Several access points can be connected together with cables extending in this way the zone covered by the wireless network. This is the most common setup for a Wi-Fi network (see **Fig1**).

In ad-hoc mode we do not have any central management, each client can talk directly to the other. This mode works fine for networks with few elements. Without any routing algorithm, each element needs to have a direct radio link with the others in order to communicate, no data will be forwarded (see **Fig2**). If a routing algorithm such as OLSR is added, you obtain a self-organizing mesh network in which message forwarding is possible wirelessly between different nodes, connecting in this way devices which are not within direct radio range (see **Fig3**). This allows to extend the zone covered without the need of any cable.The network is completely dynamic, routing tables are rewritten automatically and dynamically as the network changes. If a new OLSR enabled device appears, it will be automatically detected and merged to the routing tables of each node. This is especially useful for mobile networks that can change over time like for example in a multi-robot application.

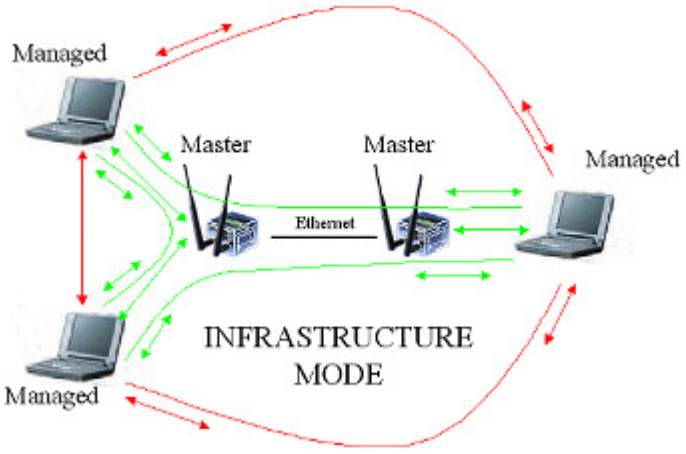


Fig 1

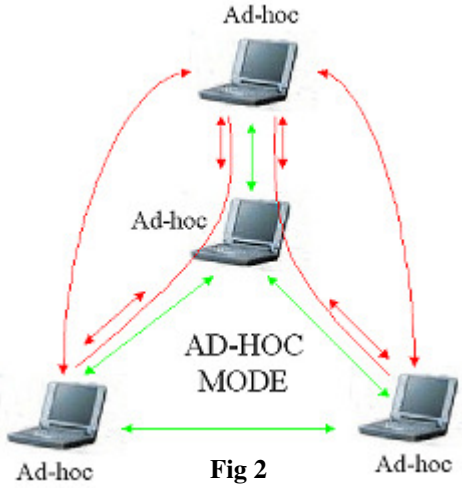


Fig 2

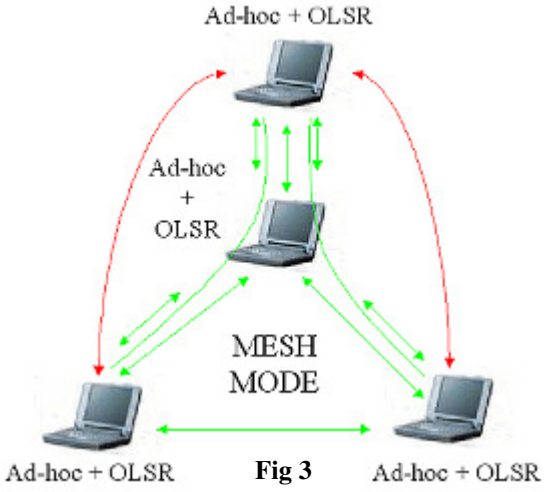
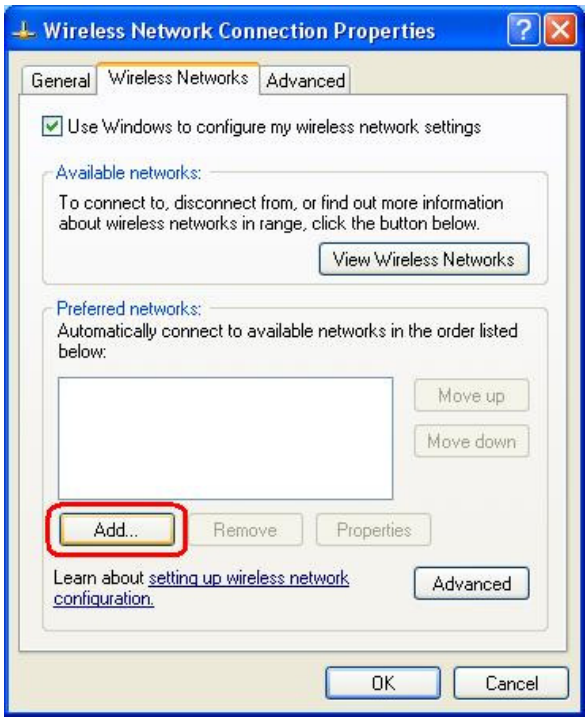


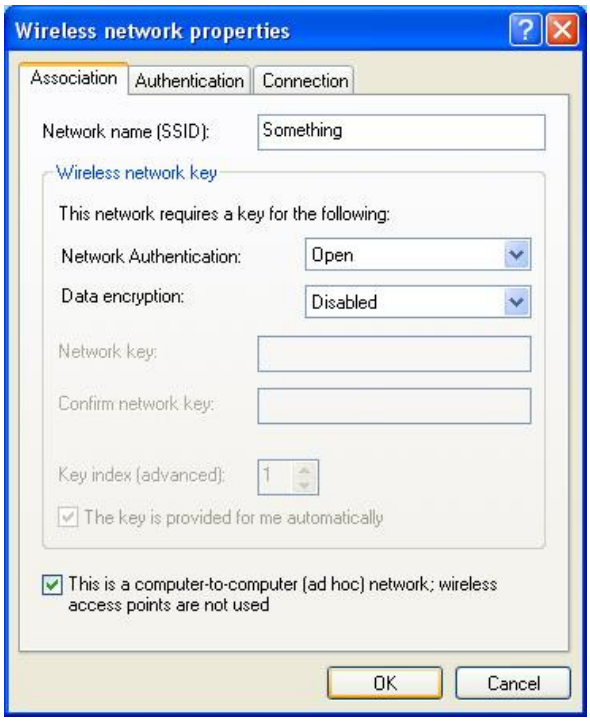
Fig 3

Network configuration

Robot under Windows XP

By default all robots come already configured and ready to work. Information is given here for those users willing to make changes in the network configuration.
When working under Windows the wifibot can only be configured in the managed and ad-hoc modes. For configuring the IP settings in managed mode or connecting the wifibot to an AP or an already created ad-hoc network please follow the steps detailed in the "connecting to the robot" section. In addition to those steps, it is recommended to create from the robot itself the ad-hoc network to be used:





1. Open **Network Connections**, Select your Wireless card right click on it and select **Properties.**
2. Click the **Wireless Networks** tab.
3. Enable **Use Microsoft Windows to configure my wireless network settings**
4. Click **Add…**
5. For **Network name (SSID)** type: **wifibot**
6. For **Data encryption** select **Disabled**
7. Enable **This a computer-to-computer (ad hoc) network**
8. Click **Ok** to close the '**Wireless network properties**' window
9. Click **Ok** to close the '**Wireless Network Connection Properties**' window
10. Using your test computer wireless adapter, view the available wireless networks, check the list and validate that you can see your newly configured **wifibot** network. If it is configured, try to connect to it. If you cannot find your new network verify the settings are correct.

Remote access

Remote access to the desktop of a robot working under Windows XP:

When working with the wifibot, it is always possible to attach a screen, a mouse and a keyboard directly into the embedded computer but it is often more convenient to have access to the robot remotely over the network. If the robot works under Windows follow these steps:

1- Click Start, point to All Programs, and then point to Accessories.
2- In the Accessories menu,point to communications and then click Remote Desktop Connection.
3- In the Computer box, type the IP address or the name of the robot you want to connect to (**Fig 1**).
4- Click Connect.
5- When the Log On to Windows dialog box appears type **root** as the user name and **wifibot** as the password, and then click OK (**Fig 2**).

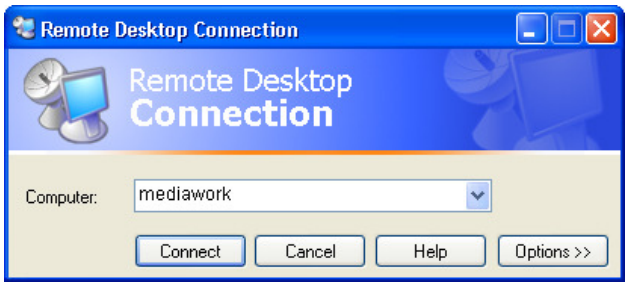


Fig 1

Fig 2

The Remote Desktop window opens, and you see the desktop settings, files, and programs that are on the robot. Your robot remains locked, and nobody can access it without a password. In addition, no one will be able to see the work you are doing remotely.

To end your Remote Desktop session:

1. Click Start, and then click Log Off at the bottom of the Start menu.
2. When prompted, click Log Off (**Fig 3**).

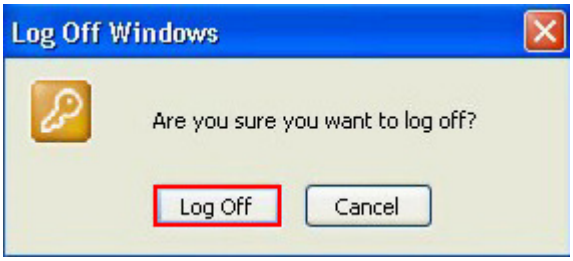


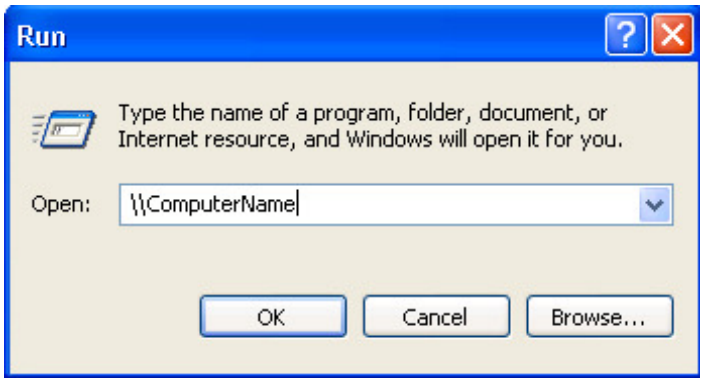
Fig 3

File transfer

Transferring files to a robot working under Windows XP:

For transferring files we will make use of the file sharing capabilities of Windows XP. From your computer in order to connect to the robot and transfer files do the following:

1. Click **Start > Run.**
2. In the **Open** field type **\\Robotname** or **\\IP address**
3. In the window that appears, type in the username **root** and password **wifibot**
4. Click **OK**



Communication protocols TCP/IP

The wifibot has been thought as an open robotic tool and as such it allows the user to have an absolute control over the higher layers. The user can work at two different levels with a wifibot, on one side it is possible to program on the embedded computer your own autonomous behaviors and on the other you can consider the robot as a network peripheral and program all processes on a remote computer to finally only send the final commands to the robot or robots to perform.

The only level that is not accessible to the user is the microcontroller that takes in charge the motor control. The programmer will have therefore to stick with the established communication protocol between the embedded computer and the microcontroller. For a description of this protocol please see the dedicated annex and the API. Users are free though to make use of the two existing protocols between the embedded computer and the provided control interfaces for their own projects or to create a new one altogether that would suit better their application needs.

The computer- RTMIX control interface protocol:

Here the communication takes place through two channels of UDP sockets. The first channel (robot's **port 15000**) is for sending commands to the robot while second (robot's **port 15010**) is to request data from it. Communication is done by the use of different and specific messages that the robot separately processes in order to take the proper action. We will present here only the basic set of messages necessary for your program to deal with the control interface, please refer to the source code for more detailed information.

Messages to port **15000**:

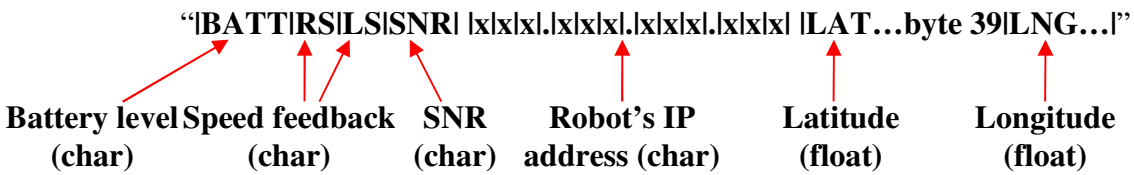
"**|i|n|i|t|**" this has to be answered with "**|o|k|**" to let the control interface know we are really there

"**|s|p|e|e|d| |RS| |LS|**" this needs no answer. RS stands for right speed and LS for left speed.

Messages to port **15010**:

"**|i|n|i|t|**" this has to be answered with "**|o|k|**" to let the control interface know we are really there

"**|d|a|t|a| |x|x|x|.|x|x|x|.|x|x|x|.|x|x|x|**" this is a request for sensor data, the message includes the IP address of the reference for calculating the SNR. This needs to be answered with:



Communication protocols USB/I2C

The wifibot has been thought as an open robotic tool and as such it allows the user to have an absolute control over the USB/I2C lower layers. The user can work at two different levels with a wifibot, on one side it is possible to program on the embedded computer your own autonomous behaviors and on the other you can consider the robot as a network peripheral and program all processes on a remote computer to finally only send the final commands to the robot or robots to perform.

For the low level any program who can send RS232 over USB frame can contrôle the robot.
An internal USB to I2C convert USB frame to I2C to send command to the both DSPIC 30F2012 that control the motors.
http://www.robot-electronics.co.uk/htm/usb_i2c_tech.htm
You can use our embedded computer or a simple laptop to control the wifibot.

```
//BYTE add : DSPIC Left 0x51 DSPIC Right 0x52
struct SensorData GetI2cMotor(HANDLE hUSB, BYTE add)
{
DWORD n;
BYTE sbuf[100];
struct SensorData data;

sbuf[0] = I2C_CMD;                          // send sonar read command
sbuf[1] = add+1;
sbuf[2] = 0x00;
sbuf[3] = 0x03;
WriteFile(hUSB, &sbuf, 4, &n, NULL);
ReadFile(hUSB, &sbuf, 3, &n, NULL);
data.SpeedFront=sbuf[0];
data.SpeedRear=sbuf[1];
data.IR=sbuf[2];
return data;
}

int SetI2cMotor(HANDLE hUSB, BYTE add,BYTE speed)
{
DWORD n;
BYTE sbuf[100];

sbuf[0] = I2C_CMD;                          // send gain limit
sbuf[1] = add;
sbuf[2] = 0x00;
sbuf[3] = 0x01;
sbuf[4] = speed;
WriteFile(hUSB, &sbuf, 5, &n, NULL);
ReadFile(hUSB, &sbuf, 1, &n, NULL);
return sbuf[0];
}

int GetADC(HANDLE hUSB, BYTE add)
{
DWORD n;
BYTE sbuf[100];

sbuf[0] = I2C_CMD;
sbuf[1] = add;
sbuf[2] = 0x00;
sbuf[3] = 0x02;
sbuf[4] = 0x40;
sbuf[5] = 250;
WriteFile(hUSB, &sbuf, 6, &n, NULL);
ReadFile(hUSB, &sbuf, 1, &n, NULL);

sbuf[0] = I2C_CMD;
sbuf[1] = add+1;
sbuf[2] = 0x00;
sbuf[3] = 0x01;
WriteFile(hUSB, &sbuf, 4, &n, NULL);
ReadFile(hUSB, &sbuf, 1, &n, NULL);
return sbuf[0];
}

int stop(HANDLE hUSB)
{
SetI2cMotor(hUSB,0xA2,0x00);
SetI2cMotor(hUSB,0xA4,0x00);
return 0;
}
```

The computer-WIFIBOT control interface protocol:

Here the communication takes place through one TCP socket. The robot is the server and the interface the client. The communication channel is through robot's **port 15020** and unlike the previous protocol it is used for both sending and receiving data.

Messages to port **15020**:

"**|comg|comd|**" this is simply the two speed command for the motor where "**comg**" is the speed for the wheels on the left and "**comd**" is the speed for the wheels on the right.

The information contained in the sent speed bytes is codified over the bits in the following manner:

Speed value range (6bits)

0-40 with closed loop

0-60 with open loop

1: Closed loop
0: Open loop

Speed byte: CTRL DIR SPEED5 SPEED4 SPEED3 SPEED2 SPEED1 SPEED0

1: Forward
0: Backwards

In closed loop the speed value is the number of encoder tics per control period which is 10ms.

In open loop the speed value is simply multiplied by 4 and sent to the PWM registers that feed the motor drivers.

Messages from port **15020**:

The robot sends us a buffer with sensor values. Note: if the robot is the two encoder version, then the rear speeds values are a copy of the front encoders values.

"|BATT|FLS|RLS|FRS|RRS|AN1|AN2|"

Battery level (char)
Front Left Speed (char)
Rear Left Speed (char)
Front Right Speed (char)
Rear Right Speed (char)
ADC on pin 2 of the interface connector (char)
ADC on pin 3 of the interface connector (char)

The CDROM

The CDROM included with the robot contains the documentation and sample programs for the robot. Its contains three folders:

Documentation:

In this folder you will find the original documentation of the platform (this file), the embedded computer and the camera.

Software:

This folder contains the simple robot control software or the RTMIX Rremote multi-robot control software installation program, a copy of the robot's embedded server, as well as several tools necessary to manage the robot. The location in the robot for the embedded server (wifibot_server.exe) is the "data" folder of the Compact Flash, the user is free to replace this file with his or her own program for giving the robot autonomous behaviors etc.

Code Samples:

Here you can find the source code of the robot's server for Windows and Linux as well as samples for the programming of remote control applications on computers under both Windows and Linux.

Simple 3D Dynamic Simulator