

AmigoBot Manual



**By Benjamin Shih
Montgomery Blair High School
bs1212@gmail.com
July/August 2007**

Table of Contents

Introduction	3
Materials	4
Connecting the Cables	6
Turning on the Robot	7
Setting up the Network	8
Programming	9
Resources	10

Introduction

Welcome to the AmigoBot Manual. The purpose of this manual is to assist you in getting started with programming the AmigoBots. The two robots are located at the University of Maryland, inside the Hybrid Networks Laboratory.

The robots were made by ActivMedia Robots. They consist of a polycarbonate body and aluminum chassis. They are both equipped with computers, referred to as “robot computer”, that run Debian Linux. The computers include basic ports, such as the monitor, keyboard and mouse ports. A wireless modem is attached inside each of the computers. They also include iSweet cameras plugged into USB ports. The computers share a battery with the robots that they are connected to.

Contact Information:

Email: bs1212@gmail.com

Please include “AmigoBot” in the subject.

Materials

The materials required differ depending on whether or not you want to use a wireless or wired network. The only materials actually needed for a wireless network are the robot and the robot computer, charger, and computer with Secure Shell (SSH) capabilities. On the other hand, to use a wired network, you will need a robot and the robot computer, charger, and an Ethernet cable. To set up either network, you will also need the mouse, keyboard, VGA cable, and monitor. On the next page you will see pictures of the materials.



Setting up the network.



Charging the robot.



Wireless operation (charger attached).

Monitor



VGA Cable



Mouse



Keyboard



Robot and robot computer

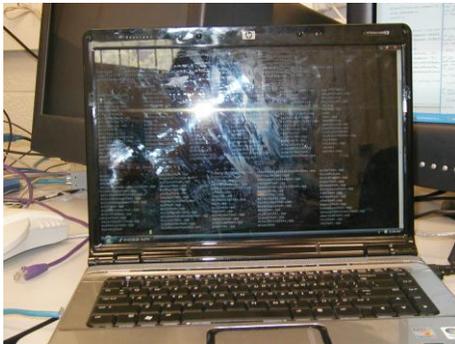


Charger, plugged into the robot computer

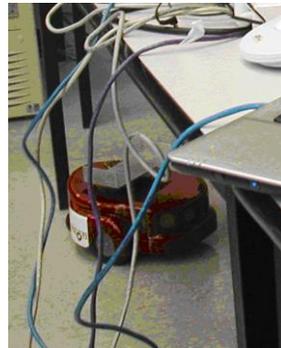


Optional

Computer with Secure Shell (SSH) capabilities



Ethernet Cable



Connecting the Cables

1. Connect the charger to the charging port.



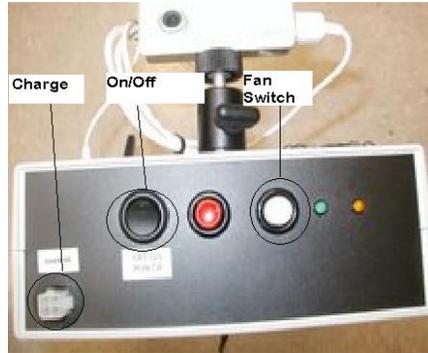
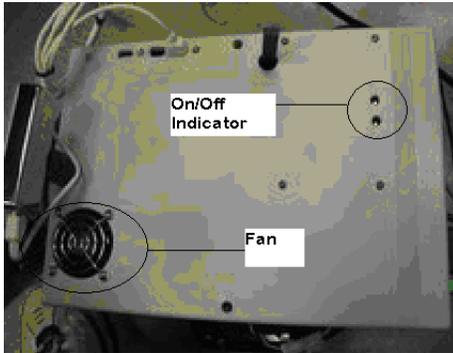
2. Attach the VGA cable, mouse, and keyboard.



3. Make sure the robot is connected to the robot computer.



Turning on the Robot



1. Using the above images, turn the computer on and press the fan switch. Make sure the fan is running.
2. Turn on the robot by using the switch located on the bottom of it.
3. After starting up, the monitor should look like this:

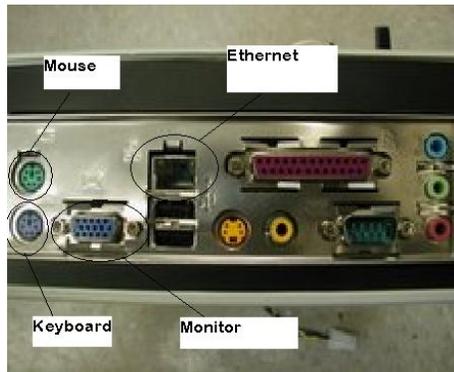


4. Log in as root and create an account. To access the GUI, type in *startx*.



Setting up the Network

1. Turn on the robot computer.
2. Connect to the internet using an Ethernet cable or wireless device. (eth0 and eth1 correspond to what network you are using to connect. They can be altered.)
 - Ethernet cable (eth0)
 - a. Plug in the Ethernet cable.
 - b. Type in `ifdown eth0` to take down the wired network.
 - c. Go to `/etc/network/interfaces`, and edit the settings. For example, choose between dhcp and static IP addresses.
 - d. Type in `ifup eth0` to restore the wired network.
 - e. Wait a little bit to allow the network to establish the connection.



- Wireless device (eth1)
 - a. Type in `ifdown eth1` to take down the wired network.
 - b. Go to `/etc/network/interfaces`, and add the following:

```
iface eth1 inet dhcp           uses dhcp
wireless-essid umd           assigns umd as the ESSID
auto eth1                   loads eth1 at start up
```
 - c. Type in `ifup eth0` to restore the wired network.
 - d. Wait a little bit to allow the network to establish the connection.
 - e. Open a browser and log in to the UMD wireless network.

Programming

1. Turn on the robot computer. If desired, SSH into the robot computer using the IP address found from typing *ifconfig*.
2. Change directories to */usr/local/Aria/*, or wherever the Aria files are located.
3. Program using your preferred text editor. Save the file with a *cpp* extension.
4. Compile the *cpp* file using *make*, *gcc*, or *g++*. (NOTE: There is currently a problem with compiling. One robot cannot compile because of a discrepancy between the libraries, the version of Linux, and the version of *gcc/g++*.)
5. Run the program by typing just the name of the program and pressing enter. (NOTE: One robot encounters the SIGSEGV error, a segmentation fault, which causes the robot to stall and it has to be shut down.)

Sample programs are included in */usr/local/Aria/examples*. Files to be incorporated into programs are located in */usr/local/Aria/src*. Programs used to test the robot are located in */usr/local/Aria/tests*.

Using sonar, the robot can run tests and determine when to stop moving in a direction to avoid running into something. In addition, all of the functions of these programs are indicated in their names. For example, *ArActionDriveDistance.cpp* makes the robot drive a certain distance. The program requires a parameter for distance, and invokes a source file that initiates the motors for turning the wheels. For example, the following drives the robot forward:

```
...
ArActionConstantVelocity constantVelocity("CV", 400);
...
robot.lock();

robot.comInt(ArCommands::ENABLE, 1);
robot.addAction(&constantVelocity, 25);
robot.unlock();

robot.waitForRunExit();
Aria::exit(0);
}
```

To use the java or python wrappers, see the readme in the respective folders.

Resources

Robot:

<http://robots.mobilerobots.com/amigobot/amigofree/AmigoGuide.pdf>

<http://robots.mobilerobots.com/amigobot/originalAmigos.html>

<http://www.activrobots.com/ROBOTS/amigobot.html#Applications>

VI editor:

<http://www.eng.hawaii.edu/Tutor/vi.html>

Debian Linux:

<http://www.ss64.com/bash/>

<http://www.debian.org/doc/manuals/user/ch6.html>

Networking:

<http://www.debian-administration.org/articles/254>

<http://www.debianadmin.com/debian-networking-for-basic-and-advanced-users.html>

<http://qref.sourceforge.net/Debian/reference/ch-gateway.en.html>

Camera:

<http://robots.mobilerobots.com/ACTS/>