ELEC4403 Lab Prep: EyeBot 7 - Usage

September 2016 - Marcus Pham

Flashing the SD Card image

1. Download the compressed image from:
    a. http://robotics.ee.uwa.edu.au/courses/des/rasp/images-pi1/
2. Unzip file using your desired unzipped – warning, image is 8GB
3. Insert SD Card into computer using either an SD card reader or inbuilt SD Card slot
4. Flash image using:
    a. OSX: ApplePi Baker - http://robotics.ee.uwa.edu.au/courses/des/rasp/bin-mac/
    b. Windows: Win32DiskImager - http://robotics.ee.uwa.edu.au/courses/des/rasp/bin-win/
5. This process will take ~10mins depending on the SD card speed

Booting the Raspberry Pi

You will now have the image for the Raspberry Pi with the EyeBot7 libraries.

Connecting to the Raspberry Pi

This will by default boot to the LCD screen where you can view the SSID broadcasted by the Raspberry Pi (provided the Wifi dongle is attached). The default SSID will be in the form **Pi_********* where ******* is the serial number of the Raspberry Pi
The default password for the wireless network is: **raspberry**
The IP of the Raspberry Pi: **10.1.1.1**

Alternatively you can connect to the EyeBot via Ethernet cable. If you wish to connect directly to your computer, the IP: **10.0.0.1.** Otherwise if you have hooked up the Raspberry Pi to an external router, the router will determine the IP.

Logging into the Raspberry Pi

Now that we are connected to the Raspberry Pi, we can connect to the Raspberry Pi using a variety of methods. All methods use the same login:
The default username for login is: **pi**
With the default login password: **rasp**

1. **SSH**. You can ssh into the Raspberry Pi through either terminal (OSX) or and suitable command line with networking capabilities, eg. Putty (Windows – http://robotics.ee.uwa.edu.au/courses/des/rasp/bin-win/)
    a. From here you can access the files on the Raspberry Pi using the command line and edit/compile here

2. **FTP**. You can use file transfer protocol to transfer and copy files from your computer to the Raspberry Pi. A free GUI version of FTP is FileZilla : again available from the bin-win or bin-osx directories.
3. **Remote Desktop**. To be able to access the Raspberry Pi's XWindows on your computer, you will have to remote desktop login to the Raspberry Pi. This will allow you to run programs that use any XLib functions (ie. All the LCD functions, as these will not work when running from a command line). Remote desktop is by default available with Microsoft Office otherwise you should be able to download from:
    a. **OSX -** https://itunes.apple.com/au/app/microsoft-remote-desktop/id715768417?mt=12
    b. **Windows** - https://www.microsoft.com/en-us/store/apps/remote-desktop/9wzdncrfj3ps

Changing Settings

There may be times in which you may wish to change some settings on the EyeBot, eg. Turning off the GUI on the LCD on boot, in order to speed boot time/save resources, booting to a monitor via HDMI, changing the SSID/password for the Raspberry Pi.
These can all be changed after logging in and modifying the HDT file, located at **/home/pi/eyebot/bin/hdt.txt**, and rebooting the GUI.

The EyeBot7 API

After connecting to the network, you will be able to access the EyeBot7 API by simply typing the IP address of the Raspberry Pi into any web browser, eg. 10.1.1.1. This will pop up the EyeBot7 API, as created by doxygen, and you can simply browse through the webpage links to find more information about the functions. Additionally you can use the search function on the top right to directly find the function/type that you are using.

Alternatively, the API is available at: http://robotics.ee.uwa.edu.au/eyebot7/doxygen/html/

Using the EyeBot7 functions

To use the EyeBot7 library you will have to include the header, "**eyebot.h**", to your C source file.
To compile, you must compile using **gccarm**, followed by the standard c compiler options, eg –o <filename>, -Werror, -Wall, -pedantic.
You however **will not** have to specify the standard (eg. std=c99/std=gnu99).

Running the compiled program is as simple as typing: **./<programname>** in the folder containing the program.

Basic command line usage

a. **cd <directory>** – change directory
b. **ls** – list contents of directory
c. **cp <filename> <location>** - copy file to desired location
d. **mv <filename> <location>** - move/overwrite file
e. **nano <filename>** - a basic text editor for command line, to exit/save <ctrl+x>
f. **pkill <processname>** - kills a process/program
g. **<ctrl+c>** - kills the currently running process on that command line terminal
h. **mkdir <dirname>** - makes a new directory