CS250 Lab 05 (In-Lab)

1 Introduction

In this lab, you will begin using your Raspberry Pi. In the in-lab portion, you will set up your Pi and learn how it can interact with your breadboard. In the take-home portion, you will create an RGB color mixer, explore registers with GDB, and implement two ARM programs.

2 Setting Up Your Pi

Before you do anything, first view the pre-lab that describes how to set up your Pi. It should have been handed out to you alongside this lab. Follow all of the instructions on this pre-lab.

Next, insert your micro-SD card containing NOOBS (an operating system install manager) into the underside of your Pi.

Now power up your Pi by following the wiring diagram on page 2 of the pre-lab.

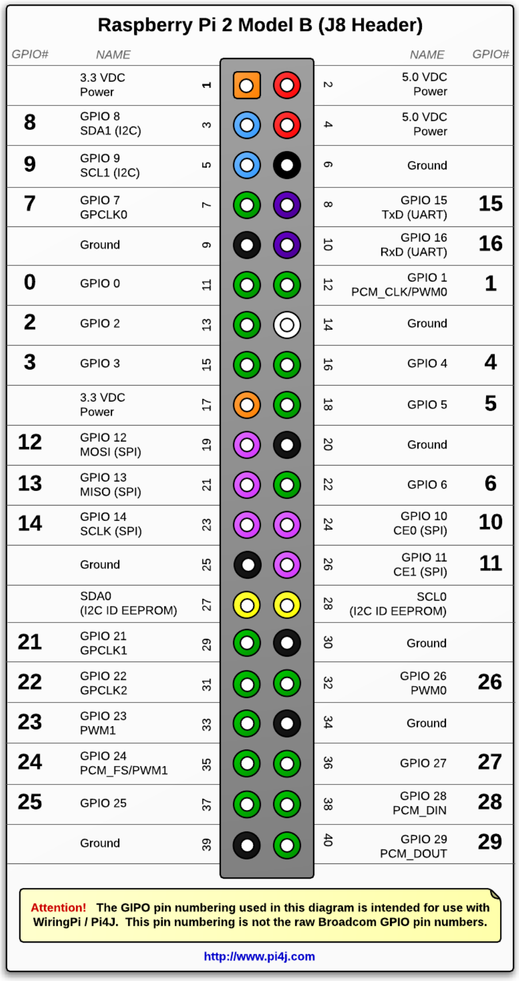
Once your Pi boots, log in with the username "pi" and password "raspberry". Once logged in, type "startx" to start an X session. The TA's will hand out ethernet cords so that you can download the necessary files for this lab from here:
`https://goo.gl/Mne2Pk`.

Finally, you need to install WiringPi, a set of libraries that allow you to use the GPIO pins on your Pi to communicate with your breadboard. In the terminal on your Pi, type:

```
sudo aptitude install git-core
```

Once that is installed, type:

```
git clone git://git.drogon.net/wiringPi
cd wiringPi
./build
```



Raspberry Pi 2 Model B (J8 Header)

GPIO#	NAME			NAME	GPIO#
	3.3 VDC Power	1	2	5.0 VDC Power	
8	GPIO 8 SDA1 (I2C)	3	4	5.0 VDC Power	
9	GPIO 9 SCL1 (I2C)	5	6	Ground	
7	GPIO 7 GPCLK0	7	8	GPIO 15 TxD (UART)	15
	Ground	9	10	GPIO 16 RxD (UART)	16
0	GPIO 0	11	12	GPIO 1 PCM_CLK/PWM0	1
2	GPIO 2	13	14	Ground	
3	GPIO 3	15	16	GPIO 4	4
	3.3 VDC Power	17	18	GPIO 5	5
12	GPIO 12 MOSI (SPI)	19	20	Ground	
13	GPIO 13 MISO (SPI)	21	22	GPIO 6	6
14	GPIO 14 SCLK (SPI)	23	24	GPIO 10 CE0 (SPI)	10
	Ground	25	26	GPIO 11 CE1 (SPI)	11
	SDA0 (I2C ID EEPROM)	27	28	SCL0 (I2C ID EEPROM)	
21	GPIO 21 GPCLK1	29	30	Ground	
22	GPIO 22 GPCLK2	31	32	GPIO 26 PWM0	26
23	GPIO 23 PWM1	33	34	Ground	
24	GPIO 24 PCM_FS/PWM1	35	36	GPIO 27	27
25	GPIO 25	37	38	GPIO 28 PCM_DIN	28
	Ground	39	40	GPIO 29 PCM_DOUT	29

Attention! The GIPO pin numbering used in this diagram is intended for use with WiringPi / Pi4J. This pin numbering is not the raw Broadcom GPIO pin numbers.

http://www.pi4j.com

2.1 Your Favorite Text Editor

The editors that come pre-installed on the Pi are vi and nano. If you usually use another editor, you may want to download it while you still have your ethernet cable. If you would like to download vim, type:

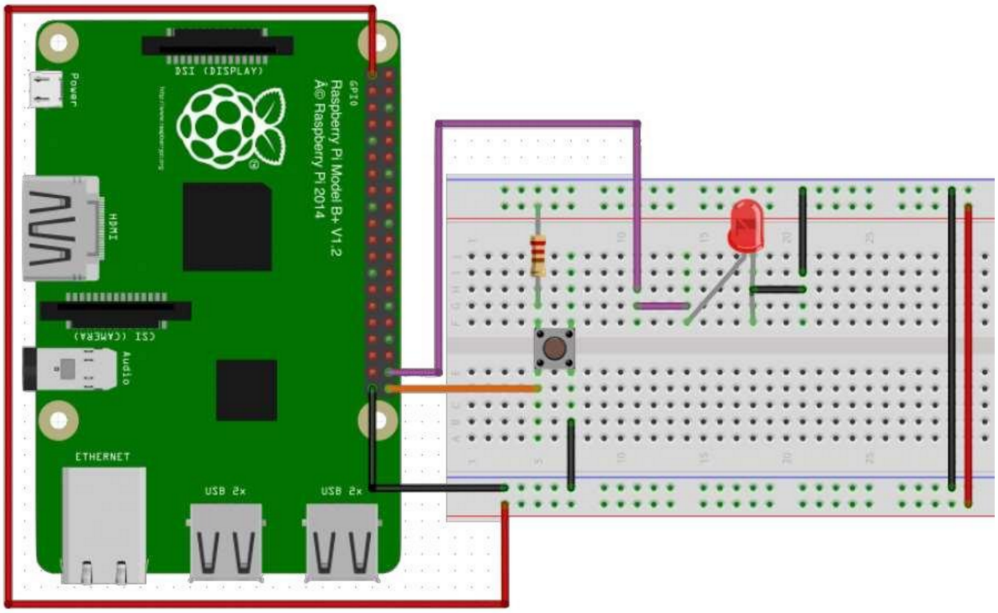
```
sudo apt-get install vim
```

Similarly, for emacs type:

```
sudo apt-get install emacs
```

3 Digital Read/Write

Now that your Pi is set up, wire your Pi and breadboard together as shown below. (Use the jumper cables in your kit to wire the GPIO pins on the Pi to the sockets on the breadboard.)



Once you have this wired correctly, study the program "read_write.c" (it is inside of the pack of files that you downloaded eariler) to understand exactly what it does. The code is well commented. To compile then run the file, type

```
gcc -std=c99 -o read_write read_write.c -lwiringPi
sudo ./read_write
```

This should give you a good idea how to use the pins on your Pi. To safely shut down your Pi, type:

```
sudo shutdown -h now
```

The -h flag stands for "halt" (shut down). Using the -r flag restarts your Pi.