



Colorado Hack-a-Thon February 22nd and 23rd



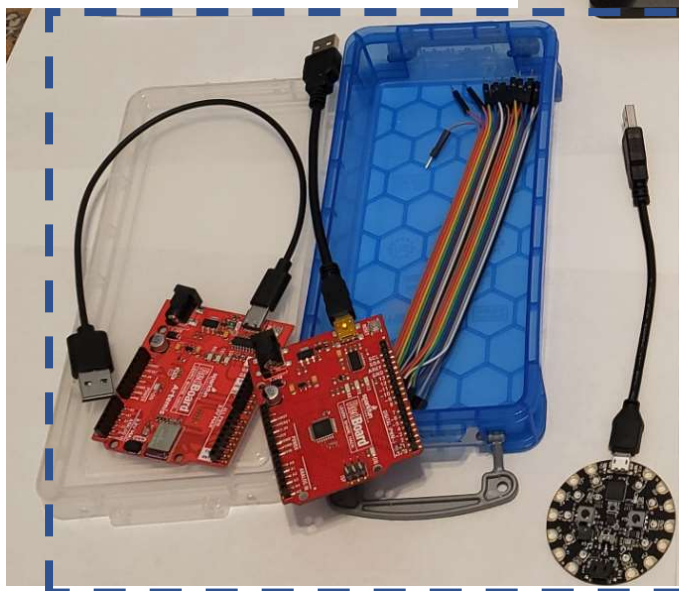
KHP

Keysight Hacking Platform

KHP

Raspberry Pi + Smarti-Pi Case

Standard kit and Circuit explorer kit.



Standard Kit
Circuit Playground
SparkFun Redboard
SparkFun Artemis
20 wire bundle Male-Male



Circuit Explorer Kit
Proto-board wire kit
Pi to Protoboard adapter
MLP3115A – I2C pressure sensor
CSC811 – Air Quality Sensor
TMP36 – Analog Temperature
2 buttons

Wireless Access Point

SSID: PiNet

Passkey: PiNet3295

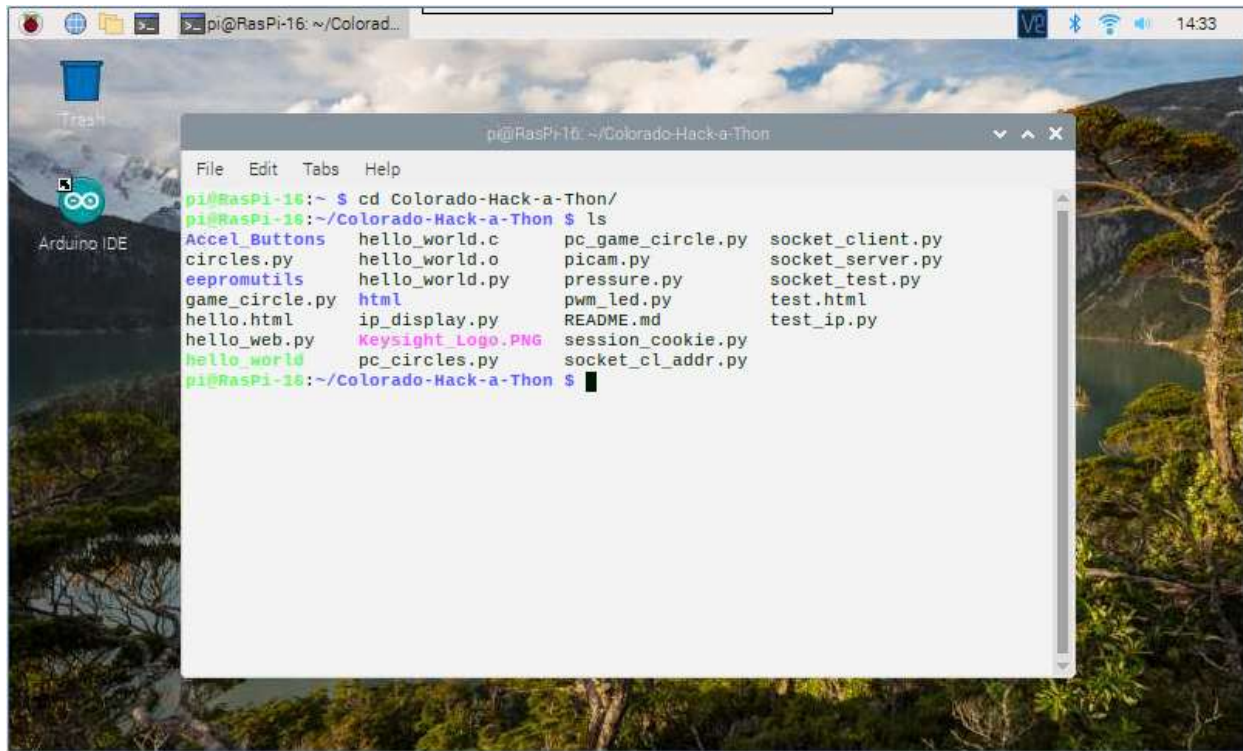
Basic Tour of the Raspberry Pi

On power up:



This will show the Host name and IP address once it is connected to a network. This can be closed at any time

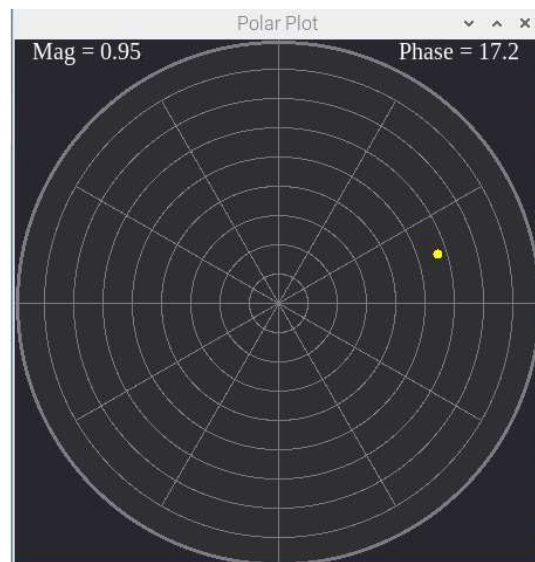
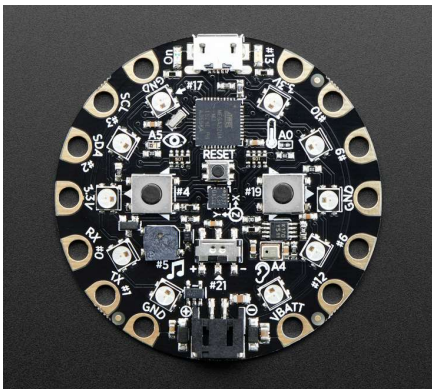
Tour open terminal



Change directory to the Colorado-Hack-a-Thon folder

Type: **python picam.py** - The camera should start up

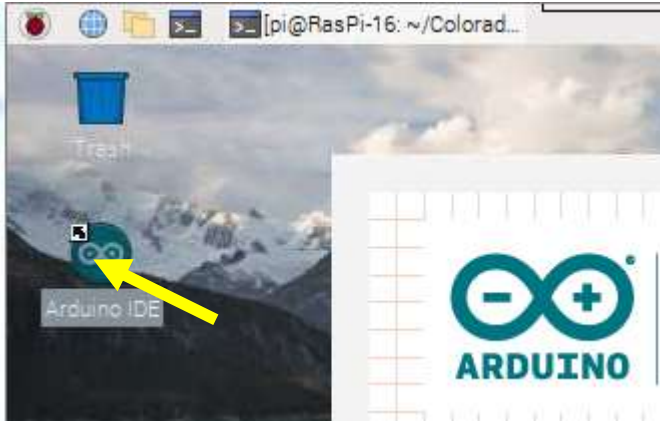
Plug in the round circuit board, "Circuit Playground" to a USB port. Then type: **python game_circle.py** Notice what happens when the marker gets to the center.



Look at Arduino Programming

Close the game_circle program if running.

Double click the Arduino IDE on the desktop.



This is the program loaded into the Circuit Playground

```
Accel_Buttons | Arduino 1.8.17
File Edit Sketch Tools Help
[Icons]
Accel_Buttons
#include <Adafruit_CircuitPlayground.h>

bool leftButtonPressed;
bool rightButtonPressed;
bool slideSwitch;

float X, Y, Z;
String buf;
uint8_t pixel_pointer = 0;
uint32_t pixel_set_word;
int sa,i;

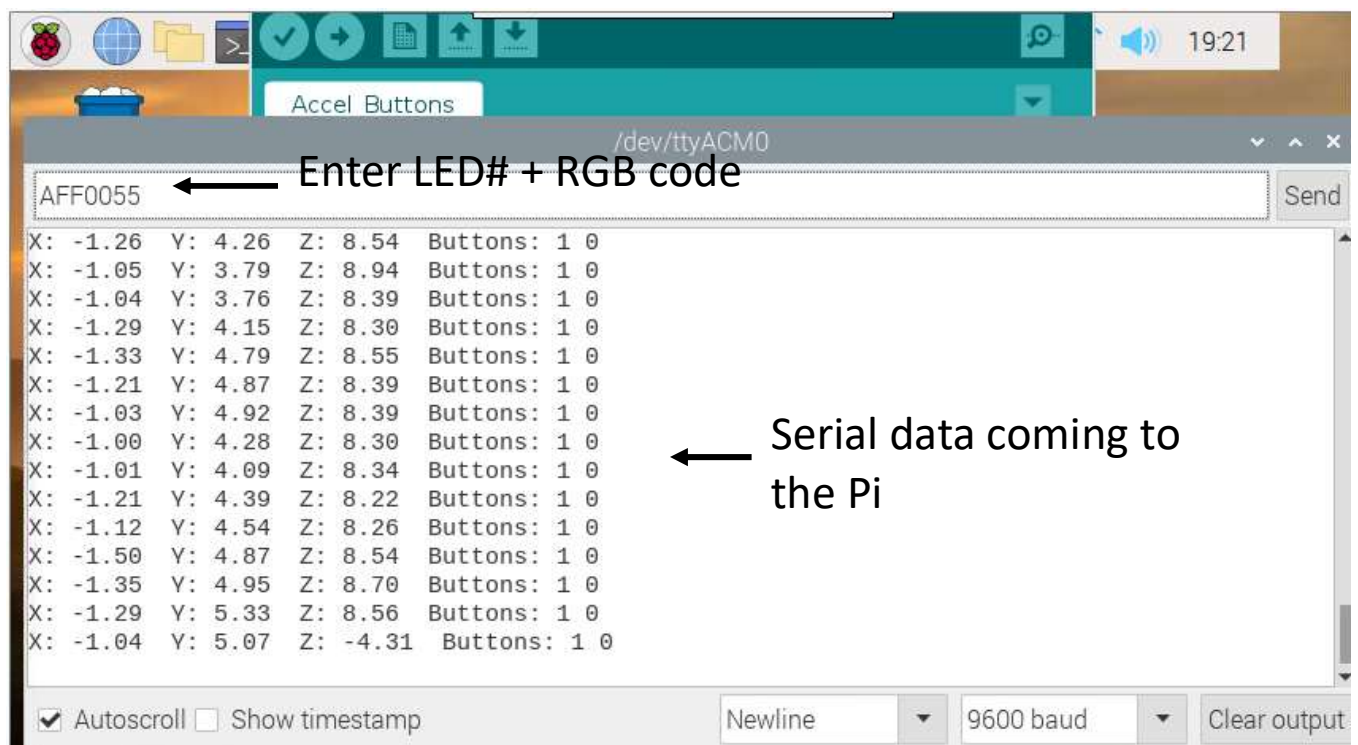
unsigned char atoh(unsigned char data) // Convert ASCII to Hex Digit (4 bit nibble)
{
  if (data > '9')
  {
    data -= 9;
  }
  data = data & 0x0F;
  return (data);
}

void setup() {
  Serial.begin(9600);
}
```

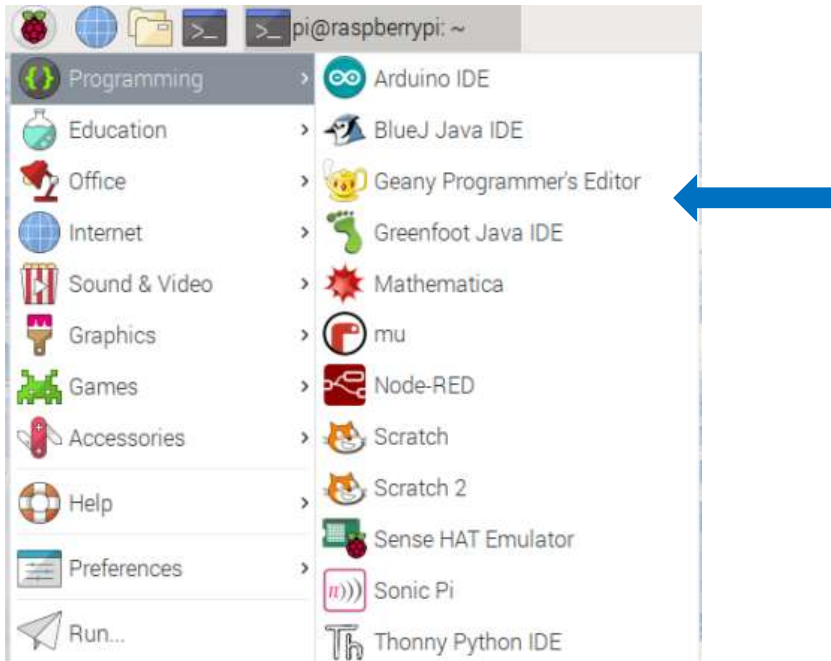
Look at serial monitor



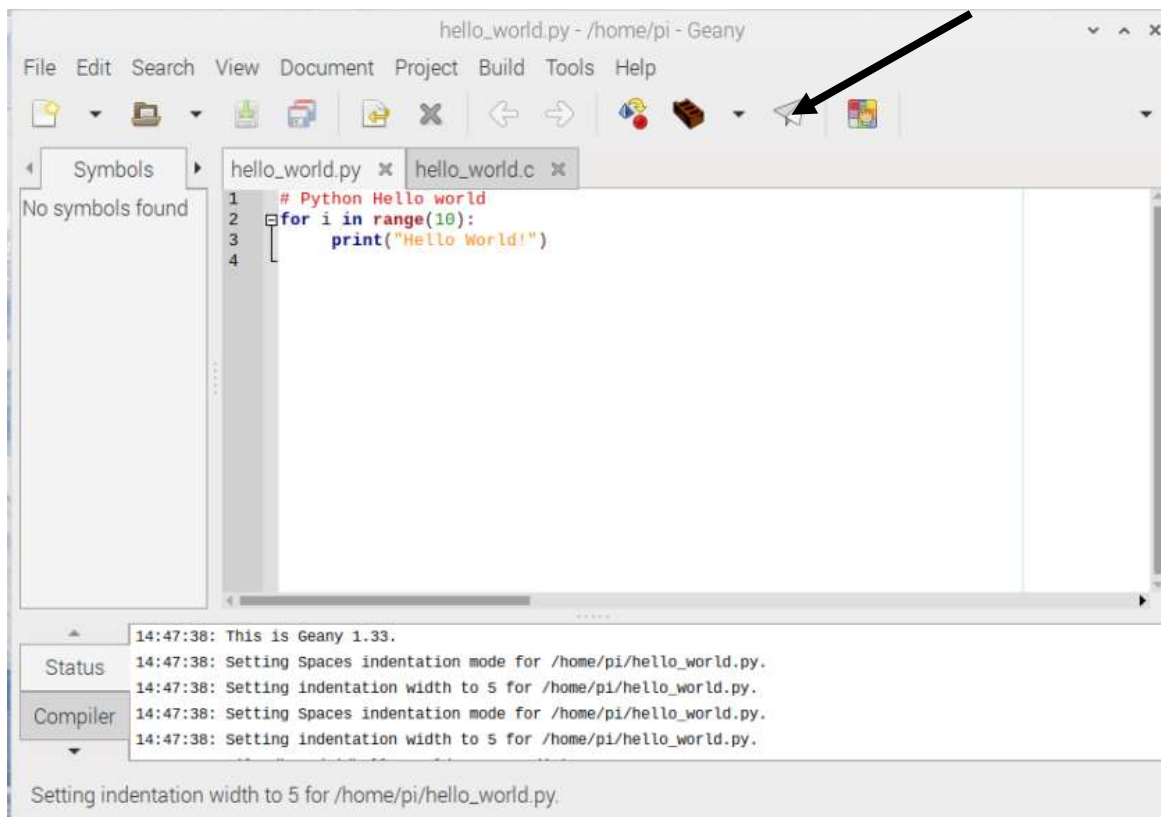
Click to open monitor



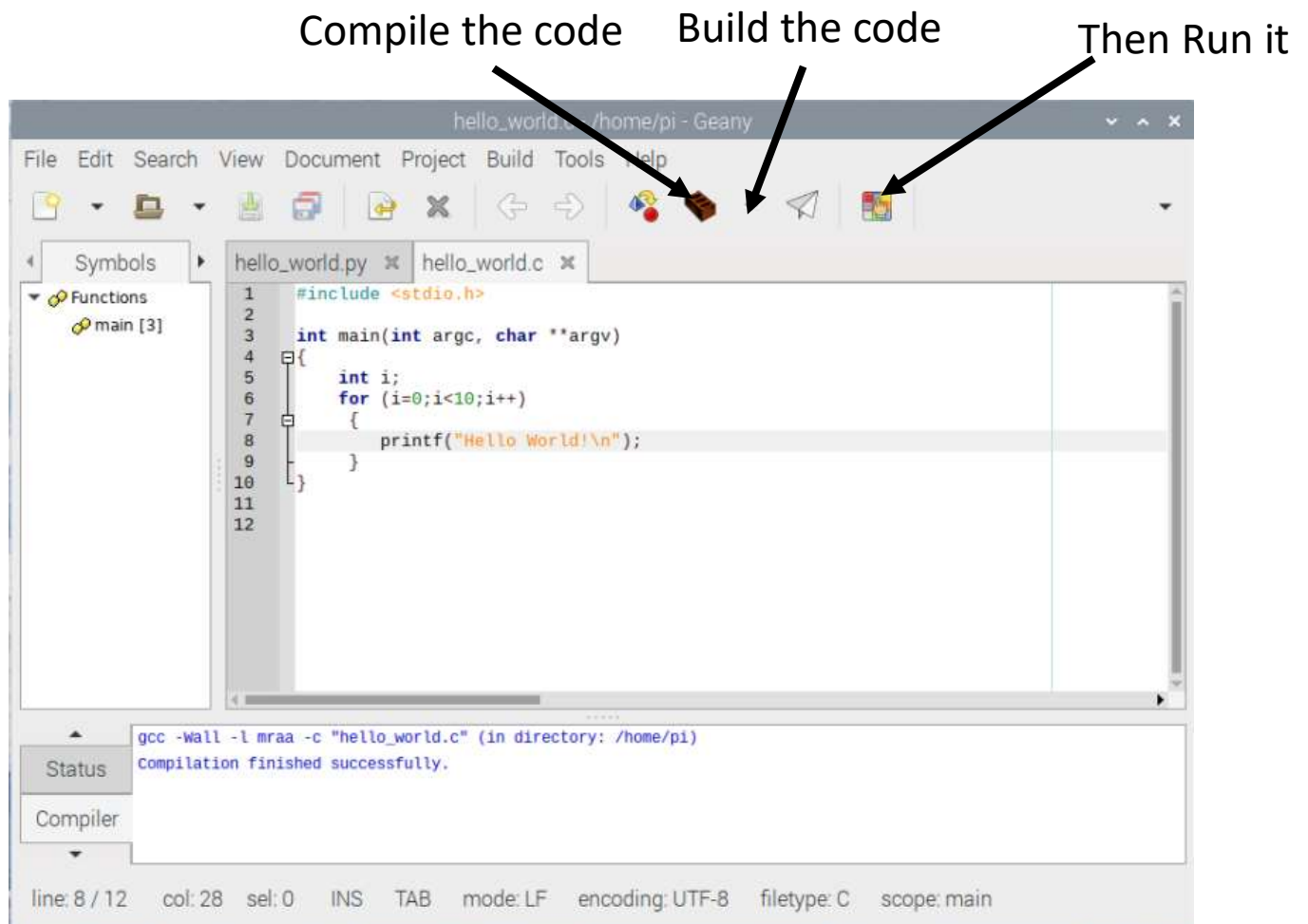
Geany – A Raspberry Pi IDE



Change the code, save and run it.



Geany and C programming



Handy Tools I2C

- The i2c bus is a bidirectional bus with 2 lines SDA and SCL.
- This bus can have up to 117 devices.
- i2cdetect - Show the devices on an i2c bus
- i2cget – read a register value from an i2c device
- i2cset – write a value to an i2c device.
- i2cdump – dump all the values available from an i2c device.


```

pi@CSUPi9: ~
File Edit Tabs Help
pi@CSUPi9:~$ i2cdetect -y 1
    0 1 2 3 4 5 6 7 8 9 a b c d e f
00:  -- -- -- -- -- -- -- -- --
10:  -- -- -- -- -- -- -- -- --
20:  -- -- -- -- -- -- -- -- --
30:  -- -- -- -- -- -- -- -- --
40:  -- -- -- -- -- -- -- -- --
50:  -- -- -- -- -- -- -- -- --
60: 60 -- -- -- -- -- -- -- -- --
70:  -- -- -- -- -- -- -- -- --
pi@CSUPi9:~$ i2cget -y 1 0x60 0x26
0x00
pi@CSUPi9:~$ i2cset -y 1 0x60 0x26 0xa9
pi@CSUPi9:~$ i2cget -y 1 0x60 0x26
0xa9
pi@CSUPi9:~$ i2cdump -y 1 0x60
No size specified (using byte-data access)
    0 1 2 3 4 5 6 7 8 9 a b c d e f      0123456789abcdef
00: 00 06 3f 00 18 b0 00 ff ff b0 00 00 c4 00 00 00      .???.??...?..?...
10: 00 01 00 00 c5 e7 00 00 00 00 00 00 06 3e 90 18      .?..??.....?>??
20: b0 06 3f 90 18 b0 a9 00 00 00 00 00 00 00 00 00      ????????.....
30: aa aa aa aa aa aa aa aa aa aa aa aa aa aa aa

```

Handy Tools GPIO

In a terminal in the Colorado-Hack-a-Thon folder

Type **./set_inputs**

gpio readall

./set_inputs.sh gpio readall											
-----Pi 4B-----											
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	
		3.3v			1	2		5v			
2	8	SDA.1	ALT0	1	3	4		5v			
3	9	SCL.1	ALT0	1	5	6		0v			
4	7	GPIO. 7	IN	1	7	8	1	ALT5	TxD	15	14
		0v			9	10	1	IN	RxD	16	15
17	0	GPIO. 0	IN	0	11	12	0	IN	GPIO. 1	1	18
27	2	GPIO. 2	IN	0	13	14		0v			
22	3	GPIO. 3	IN	0	15	16	0	IN	GPIO. 4	4	23
		3.3v			17	18	1	IN	GPIO. 5	5	24
10	12	MOSI	IN	0	19	20		0v			
9	13	MISO	IN	0	21	22	1	IN	GPIO. 6	6	25
11	14	SCLK	OUT	0	23	24	1	OUT	CE0	10	8
		0v			25	26	1	IN	CE1	11	7
0	30	SDA.0	ALT0	1	27	28	1	ALT0	SCL.0	31	1
5	21	GPIO.21	IN	0	29	30		0v			
6	22	GPIO.22	IN	0	31	32	0	IN	GPIO.26	26	12
13	23	GPIO.23	IN	1	33	34		0v			
19	24	GPIO.24	IN	0	35	36	0	IN	GPIO.27	27	16
26	25	GPIO.25	IN	1	37	38	0	IN	GPIO.28	28	20
		0v			39	40	0	IN	GPIO.29	29	21
-----Pi 4B-----											
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	

Connecting the Raspberry Pi with PC or MAC

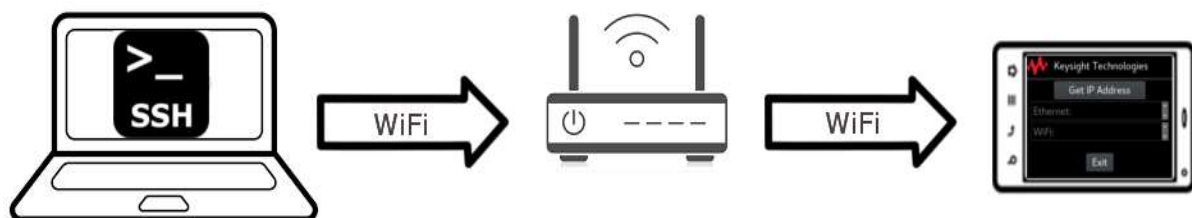
Overview

This section gives an brief overview of the hardware in the Keysight Hacking Platform (KHP), how to connect the the KHP over SSH, and where to find the example programs.

The KHP consists of:

- Raspberry Pi 3 with 7" capacitive touch screen in SmartiPi-2 Case
- Raspbian Linux image with some example programs pre-loaded.
- Some additional hardware to hack with (Arduino, Circuit Playground and other devices)

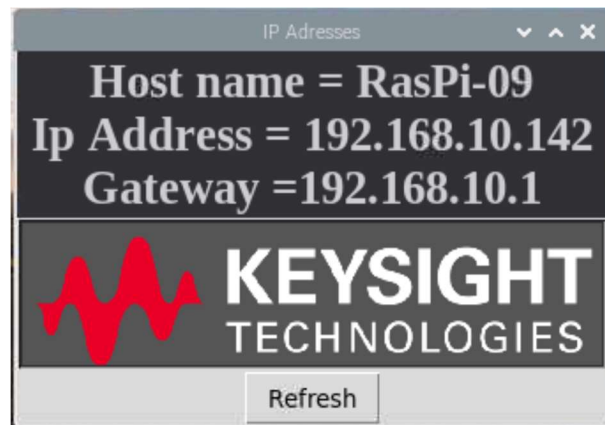
The general workflow for using the KHP consists of connecting to it over Secure Shell (SSH) and transferring files using Session Control Protocol (SCP).



Connecting to the Raspberry Pi Over SSH

The Raspberry Pi will take approximately a minute to boot up after plugging it in. After fully booting up

the Raspberry Pi will automatically start the Keysight IP Finder application. Click the Refresh button to show the configured IP addresses for the Ethernet and WiFi interfaces. Depending on how quickly the interface receives an IP address you may have to click the button a few times.



Write down the IP address of the interface you would like to use. We are now going to use this IP address to connect to the Raspberry Pi over SSH. Follow the instructions for the operating system running on your computer:

SSH Login Credentials

User Name:	Password:
pi	Channel1

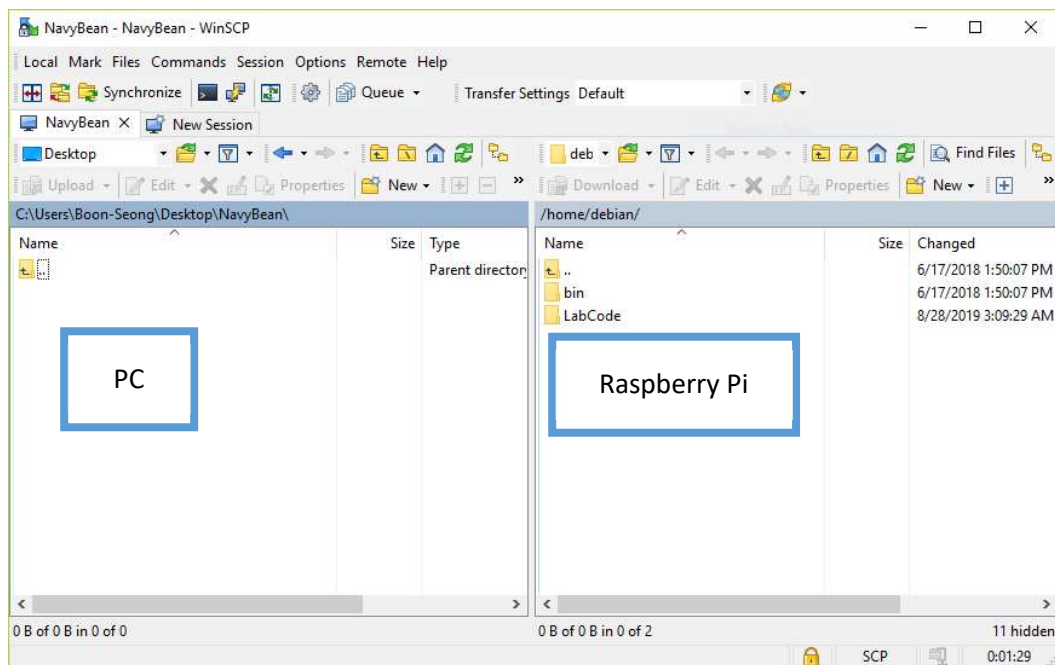
Set Up WinSCP

1. For Windows users, download and install a copy of WinSCP from <https://winscp.net/eng/download.php>. You should see a WinSCP icon on your desktop.
2. Double-click to launch WinWCP and click “New Site”. Then, configure the new site with the following settings.

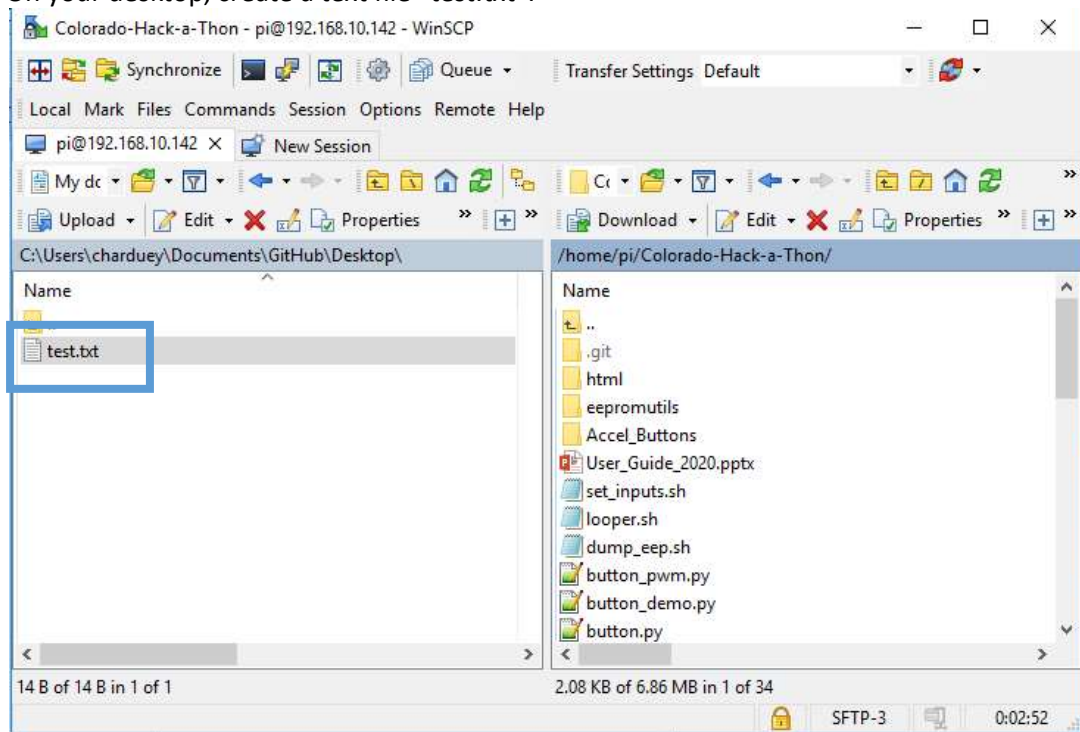
File Protocol	SCP
Host name	192.168.5.xxx
Port Number	22
Username	pi
Password	Channel1

Copy Files with WinSCP

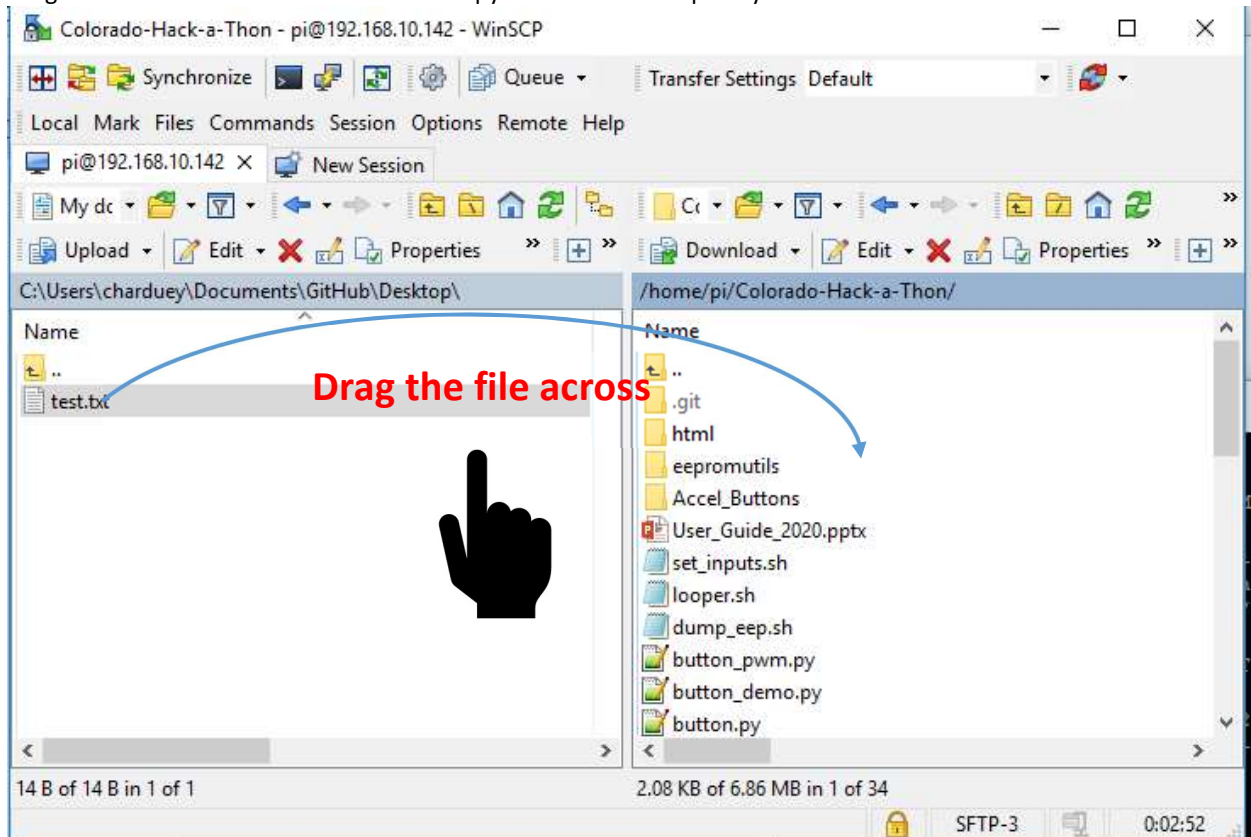
A GUI should open where files can be dragged across from the PC to the Raspberry Pi and Visa-Versa.



3. On your desktop, create a text file "test.txt".



4. Drag the text.txt file across in WinSCP to copy it over to the Raspberry Pi.



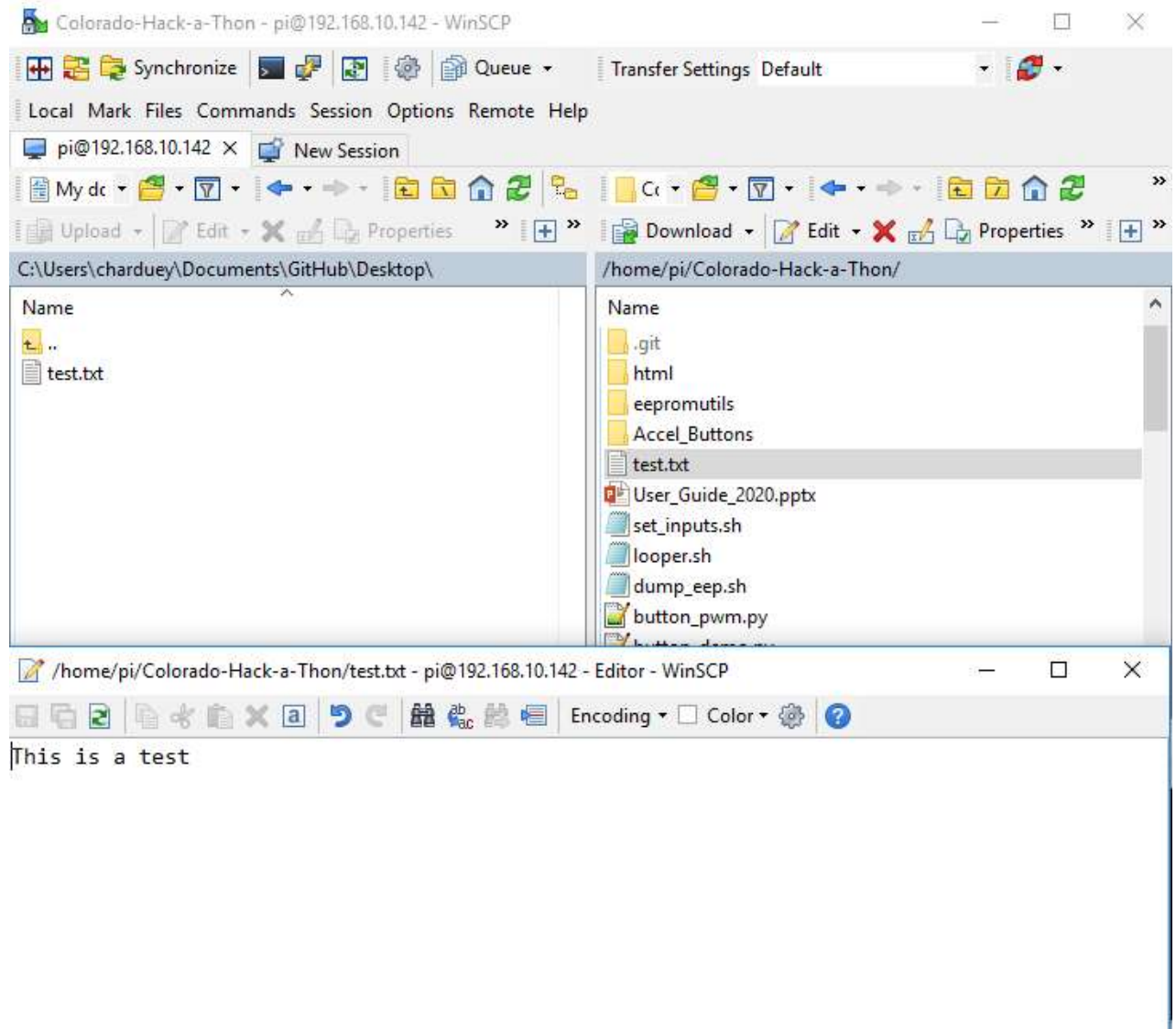
NOTE

For Linux based systems, copy the file using **scp M1-L1.zip pi@192.168.7.2 command.**

Edit Files with WinSCP

5. With the copy of the test.txt file in BeagleBone, right-click the file and click **Edit..**

It should prompt a built-in text editor where you will use it to edit shell scripts with a GUI text editor from PC.



Putty Communications

Establishing Console Communications between Pi and PC via WiFi

1. If not already done so, download and install PuTTY from <http://www.putty.org/>
[Choose 32-bit or 64-bit, whichever is compatible with your operating system.](#)



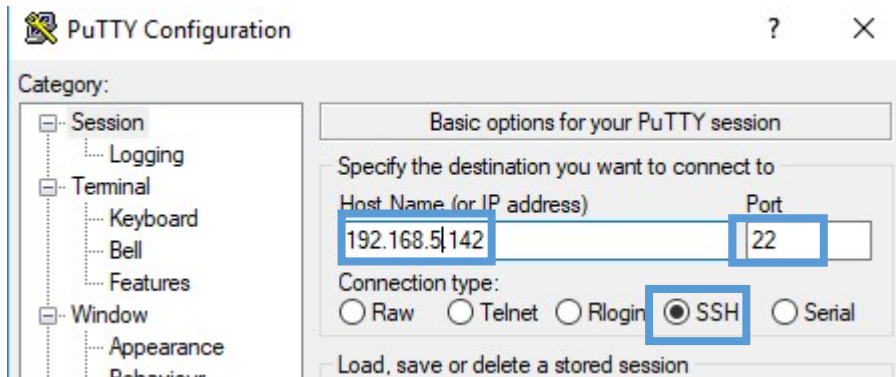
Download PuTTY

PuTTY is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform. PuTTY is open source software that is available with source code and is developed and supported by a group of volunteers.

You can download PuTTY [here](#).

1. A PuTTY Configuration window will pop up to determine the connection type. Select **SSH** for Connection type and enter **192.168.5.xxx** for the **IP address**.

If this is the first time that the computer is connecting to this Beagle Bone, you will receive this message and question to which you should click **Yes**:



PuTTY Security Alert



The server's host key is not cached in the registry. You have no guarantee that the server is the computer you think it is.

The server's ssh-ed25519 key fingerprint is:
ssh-ed25519 256

55:8b:50:2e:50:c2:eb:b3:0a:ab:d1:09:5b:84:01:72

If you trust this host, hit Yes to add the key to PuTTY's cache and carry on connecting.

If you want to carry on connecting just once, without adding the key to the cache, hit No.

If you do not trust this host, hit Cancel to abandon the connection.

Yes

No

Cancel

Help

2. Click **Open** to open the terminal window. Press **Enter** on the PC keyboard to check and verify connectivity.

```
pi@RasPi-09: ~/Colorado-Hack-a-Thon
login as: pi
pi@192.168.10.142's password:
Linux RasPi-09 4.19.97-v7+ #1294 SMP Thu Jan 30 13:15:58 GMT 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Feb 19 16:17:13 2020
pi@RasPi-09:~ $ cd Colorado-Hack-a-Thon/
pi@RasPi-09:~/Colorado-Hack-a-Thon $
```

Default username : pi

Default password : Channel1

Mac OS X

With Mac OS X there are a few different ways that you can go about using SCP with the Raspberry Pi. You can either use the SCP utility from the command line for transferring or you can download and install MacFusion to allow you to browse the Raspberry Pi's system through Finder.

The general usage for the SCP command is: `scp source file destination file`

MAC Transferring a Single File to the Raspberry Pi

- 1.) Open a terminal.
- 2.) Run this command: `scp {file to transfer} pi@{IP Address of Pi}:{ Location to transfer file to}`

MAC Transferring a Single File from the Raspberry Pi

- 1.) Open a terminal.
- 2.) Run this command: `scp pi@{IP Address of Pi}:{ Location of file to transfer}: {location to save file}`

Linux

With Linux there are a few different ways that you can go about using SCP with the Raspberry Pi. You can either use the SCP utility from the command line for transferring or several browsers in Linux natively support mounting a SSH system with SCP. For most common browsers the option to mount the remote system is usually under the File -> Connect to Server... menu option.

The general usage for the SCP command is: `scp {source file} {destination file}`

Transferring a Single File to the Raspberry Pi

- 1.) Open a terminal.
- 2.) Run this command: `scp {file to transfer} pi@{IP Address of Pi}:{Location to transfer file to}`

Transferring a Single File from the Raspberry Pi

- 1.) Open a terminal.
- 2.) Run this command: `scp pi@{IP Address of Pi}:{Location of file to transfer} {location to save file}`