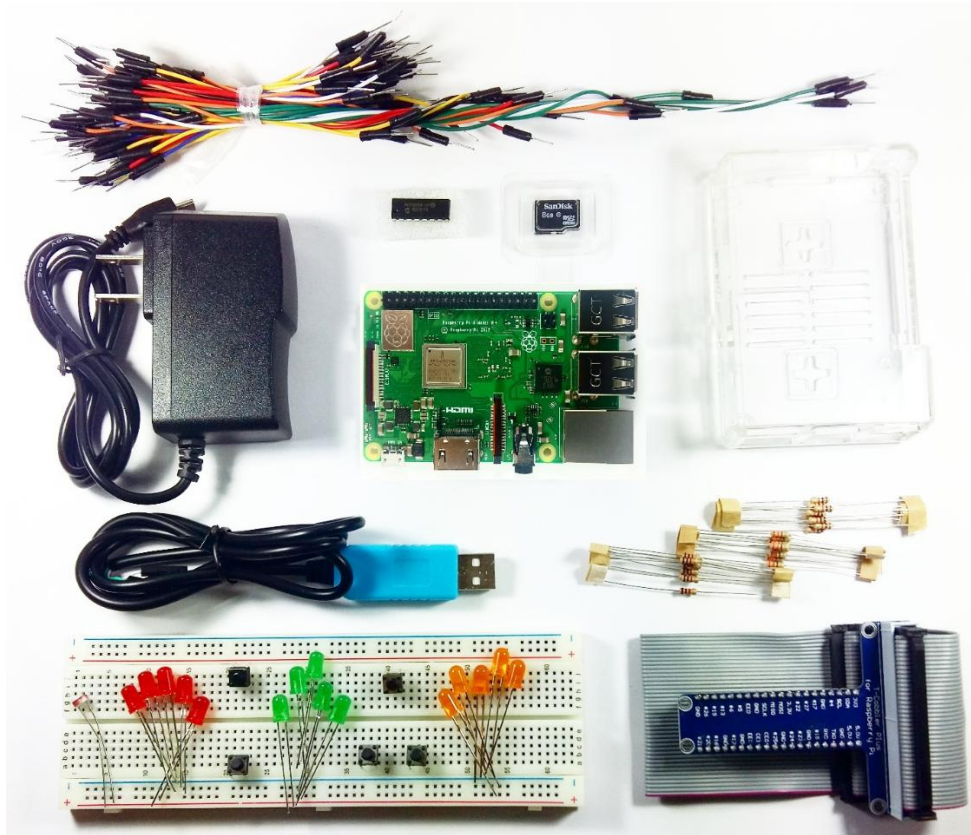


Raspberry Pi Experiment Guide Part 1

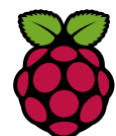
(Getting Started and Setup)



The **Raspberry Pi** is a great platform for learning the basics of a computer system such as the computer hardware, software programs and computing. It is like a personal computer on which you can use to learn software programming and you can also use it for electronics hardware development.

This **Raspberry Pi 3 Starter Kit (with Experiment Manual)** will give you to a good introduction and familiarity of this new environment for a Linux based small-board-computer.

The Raspberry Pi offers you a platform for learning about scripting languages (like Python), Linux and programming. It's also a good electronics board for your Internet connected projects because it connects to the internet easily. It can run Python scripts, or even compile programs written in C. It's like a complete computer plus a microcontroller rolled into one. You can also use it to play computer games.



00 Introduction



What is Raspberry Pi?

The **Raspberry Pi** is a tiny and affordable “credit card-sized” “single-board computer” that you can use to learn programming, hardware electronics and computer. It carries all the necessary components to form a personal computer such as a main Processor, RAM, Power circuitry, Storage Device (thru an SD Card), Ethernet and Wireless Connectivity, General Pin connectors and USB connectivity. It's a computer at a minimum shape.

If you have tried Arduino or any microcontroller programming environment, you will see some similarities and differences and this guide will help you understand easily the concepts and transition when using a Linux based environment. Some of your Arduino projects can also be run in the Raspberry Pi by installing some libraries.

Before we can use the Raspberry Pi, we need to configure and setup its needed software, hardware and internal system configuration. The steps may require more minutes than setting up an Arduino but it will only be on the first time. The next time will be a bit easier and faster for you already. It is also a great way to get a feel and familiarity with the Linux environment...

You can create projects which involves connecting of LCD touch displays and monitors, internet and wireless related, connecting webcams and do image processing and run full complex programs requiring powerful processing. You can add the Raspberry to any of your existing microcontroller project, your imagination is the limit!

For all other information about the Raspberry Pi board and the community, you can official website at <https://www.raspberrypi.org/>

01 Tutorial Content

In this guide we will show the steps on setting up the Raspberry Pi and program it in different programming languages. We have included example codes and circuit setup you can follow to easily get started. Some of the things we will explore are:

- Pi's Hardware and Software Overview
- Pi Kit Assembly
- Methods of using the Pi
- Configure Pi's wireless/Ethernet network connection
- Using the Pi in SSH
- Other Pi computer setup

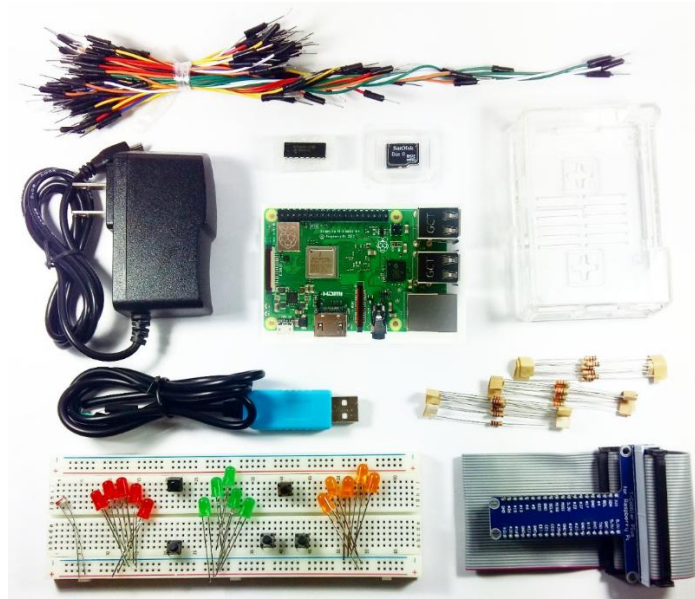
This part 1 will focus on overview and initial setup. In part 2 we will connect some electronics and do code programming. In part 3 we will explore other ways to interface the Pi to other electronics including showcase of project from the community.

We will need a host computer to configure the Pi.

02 Parts List

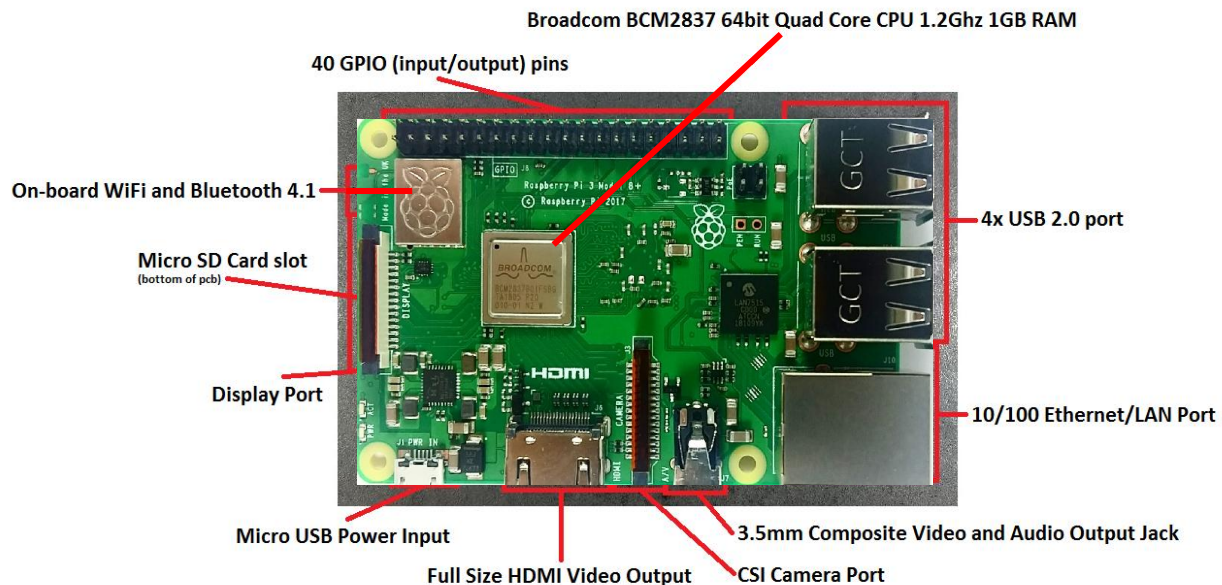
To get started with your Raspberry Pi up and running, we need the following hardware parts.

(click on the link provided to check product information in the shop)



- [Raspberry Pi 3 Model B+ \(B plus\)](#)
- [Breadboard solderless - Full size 830points](#)
- [Raspberry Pi GPIO Adapter - T Cobbler \(with ribbon cable\)](#)
- [Connecting Wires - Male-Male, 65pcs \(assorted-size\)](#)
- [MicroSD Card with Raspberry Pi OS \(8GB, class10\)](#)
- [LED - Red \(5mm\)](#)
- [LED - Green \(5mm\)](#)
- [LED - Yellow \(5mm\)](#)
- [Push Button tactile switch \(SPDT\)](#)
- [Resistors \(330 ohms 10pcs\)](#)
- [Resistors \(10k ohms 10pcs\)](#)
- [Resistors \(1k ohms 10pcs\)](#)
- [LDR \(Light Dependent Resistor\)](#)
- [Power Supply Wall Adapter - 5V/2A \(2000mA\) \(micro-USB\)](#)
- [Raspberry Pi Case \(Clear\)](#)
- [MCP3008 - 8-Channel 10-Bit ADC With SPI Interface](#)
- [USB to RS232 Serial TTL Converter Cable \(PL2303\)](#)

03 Hardware Overview



An overview of a Raspberry Pi board. It contains all the minimum required peripherals for a personal computer such as a **Main processor (BCM2837 Quad Core chip)**, **RAM**, **USB ports**, **Ethernet/LAN and WiFi**, **HDMI display port**, **Power Supply port**, **File Storage (SD card expandable)** and **40 GPIO accessible pins** for connecting your electronic components and sensors. In addition it also has a camera port and a display port.

Specifically, the Pi has the following specs:

- 4GHz 64-bit quad-core ARM Cortex-A53 CPU (BCM2837)
- 1GB RAM (LPDDR2 SDRAM)
- On-board wireless LAN – dual-band 802.11 b/g/n/ac (CYW43455)
- On-board Bluetooth 4.2 HS low-energy (BLE) (CYW43455)
- 4 x USB 2.0 ports
- 300Mbit/s ethernet
- 40 GPIO pins
- Full size HDMI 1.3a port
- Combined 3.5mm analog audio and composite video jack
- Camera interface (CSI)
- Display interface (DSI)
- microSD slot
- VideoCore IV multimedia/3D graphics core @ 400MHz/300MHz

The **Raspberry Pi 3 Model B+** is the latest hardware for the Raspberry Pi family which has a built-in WiFi and Bluetooth Low Energy (BLE) compared to the previous versions which does not have these two features.

04 Software Overview

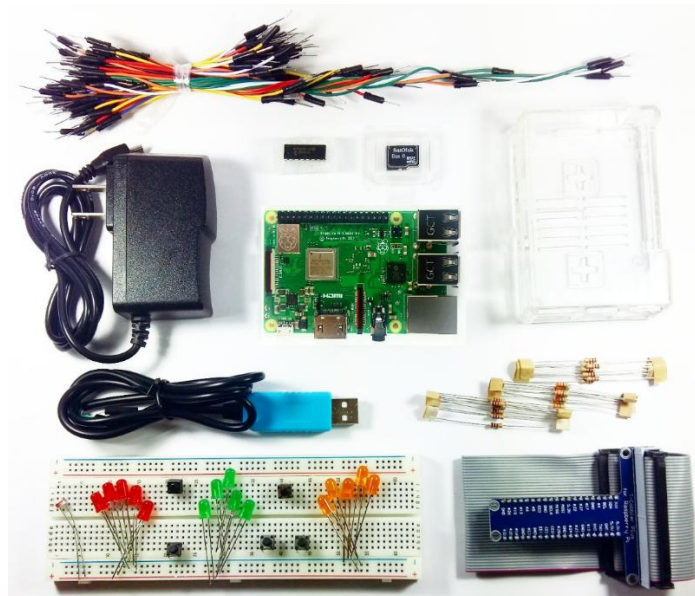


The Raspberry Pi will need an Operating System (OS) just like a normal computer. The Raspberry Pi Kit includes an *Micro SD Card (class10)* that has pre-installed **Raspberry Pi OS (Raspbian)** through the **NOOBS** operating system easy installer. The Raspbian-Jessie is the latest and stable release used in many Raspberry Pi projects. The Raspberry Pi runs on Linux OS and the Raspbian OS is an optimized version for the Raspberry Pi from the Linux OS (Debian).

To know more about Raspbian, you can check it all here: <https://www.raspbian.org/>

05 Raspberry Pi Assembly

Prepare the parts listed from the Parts List section earlier and let us assemble our Raspberry Pi computer and development system!

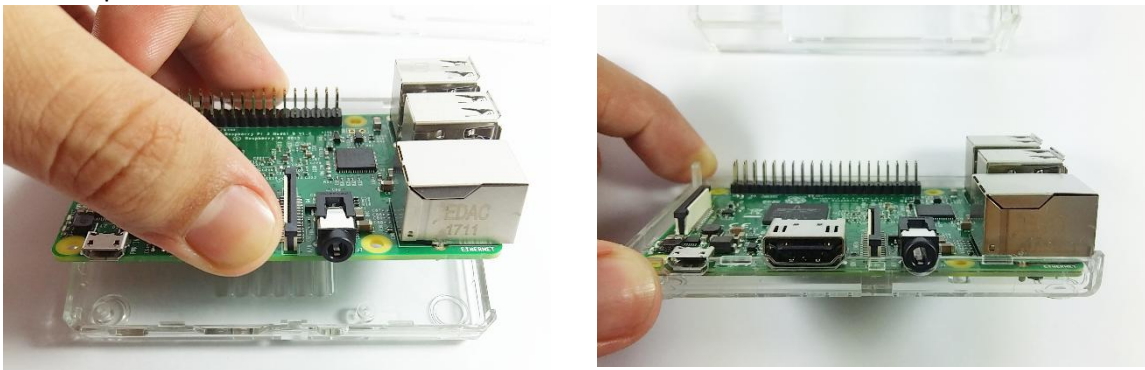


Let's assemble our Pi!

Prepare the Raspberry Pi board and the Raspberry Pi (Clear) Case



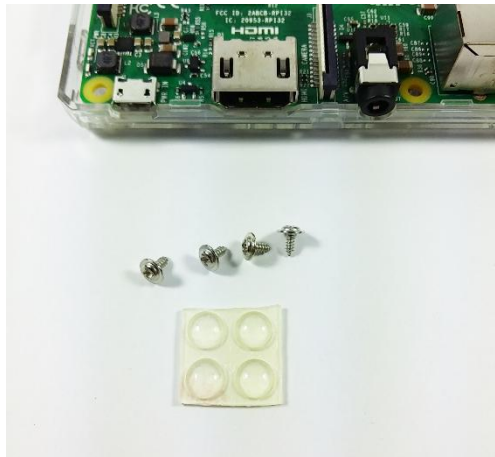
Insert and snap the Pi board to the bottom cover case.



Insert and snap the top case cover on top of the board.



You may also choose to secure your Pi board inside the Pi Case with the screws. And also you may put a rubber foot on the bottom portion of the case.



At the bottom you will find the SD Card slot. Insert the included Micro SD Card in the SD card slot with the right orientation. The card will softly slide into the SD card slot. The Micro SD Card included in this kit already has a pre-installed Raspberry Pi (Raspbian) OS so we can go ahead and use the Pi as you get the kit.



Note: We will not insert any power connection to our Raspberry Pi board yet. We will have a way to supply the Pi in the next steps.

The Raspberry Pi is assembled, head on to the next steps to start using the Pi!

06 Using the Raspberry Pi

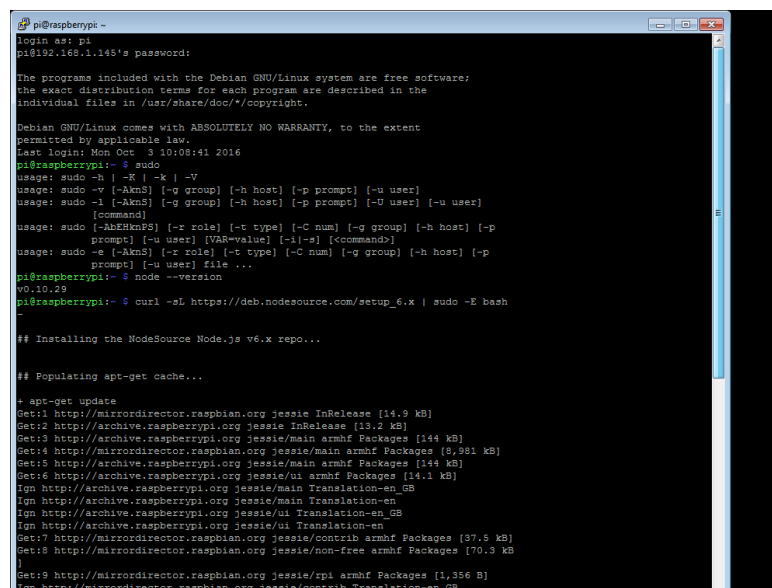
There are a few methods of using the Raspberry Pi. These methods are listed below.

- **Desktop HDMI Out** - Here we use the Raspberry Pi like a normal desktop computer. We need to connect accessories such as a Mouse, keyboard and an HDMI Monitor and we can see the Raspberry Pi's desktop icons, access the internet with an internet browser and do anything to our Raspberry Pi board.
- **Serial Terminal communication** - In this method, we use and access the Raspberry Pi using a USB to Serial TTL Console cable. This method is suitable when you want to use the Raspberry Pi and you don't have a computer monitor available. This method works well but not always stable. *In this guide we will use this method only to setup the Raspberry Pi's Network settings (connect to a router and get the IP Address) and will use the next method (SSH) for most of the projects.*
- **"headless" Using SSH/Telnet (Secure Shell) Communication** - In this method we will use a separate computer/laptop to use the Raspberry Pi. The computer/laptop and the Raspberry Pi should be connected to the same Router Network. As long as we know the IP address of our Raspberry Pi, we can access it with another computer without having to connect any cable from the Pi to the computer.

For most projects, **"headless" Using SSH/Telnet** method is used since this will not require any computer accessories connected to the Pi, we only need to know the IP address of the Raspberry Pi and Run an **SSH Terminal software program** called **"Putty"** to the computer that will control our Raspberry Pi. Using the software **"Putty"** we can do about anything to the Raspberry Pi.

Putty is one type of a **Command Line Interface (CLI)** and a **CLI software program** used widely in Linux systems which here we will get familiar.

Download the **SSH Terminal** software **"Putty.exe"** here <http://www.putty.org/>
Save it to your local disk and we will use them in the next steps.



```
pi@raspberrypi ~$
login as: pi
pi@192.168.1.145's password:

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: Mon Oct 3 10:08:41 2016
pi@raspberrypi:~$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AkmS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AkmS] [-g group] [-h host] [-p prompt] [-U user] [-u user]
       [command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
       prompt] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AkmS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
       prompt] [-u user] file ...
pi@raspberrypi:~$ node --version
v0.10.29
pi@raspberrypi:~$ curl -sL https://deb.nodesource.com/setup_6.x | sudo -E bash
.

## Installing the NodeSource Node.js v6.x repo...

## Populating apt-get cache...

* apt-get update
Get:1 http://mirrordirector.raspbian.org jessie InRelease [14.9 kB]
Get:2 http://archive.raspbian.org jessie InRelease [12.2 kB]
Get:3 http://archive.raspbian.org jessie/main armhf Packages [144 kB]
Get:4 http://mirrordirector.raspbian.org jessie/main armhf Packages [8,981 kB]
Get:5 http://archive.raspbian.org jessie/main armhf Packages [144 kB]
Get:6 http://archive.raspbian.org jessie/ui armhf Packages [14.1 kB]
Ign http://archive.raspbian.org jessie/main Translation-en_GB
Ign http://archive.raspbian.org jessie/main Translation-en
Ign http://archive.raspbian.org jessie/ui Translation-en_GB
Ign http://archive.raspbian.org jessie/ui Translation-en
Get:7 http://mirrordirector.raspbian.org jessie/contrib armhf Packages [37.5 kB]
Get:8 http://mirrordirector.raspbian.org jessie/non-free armhf Packages [70.3 kB]
Get:9 http://mirrordirector.raspbian.org jessie/cpi armhf Packages [1,356 B]
Ign http://mirrordirector.raspbian.org jessie/contrib Translation-en_GB
```

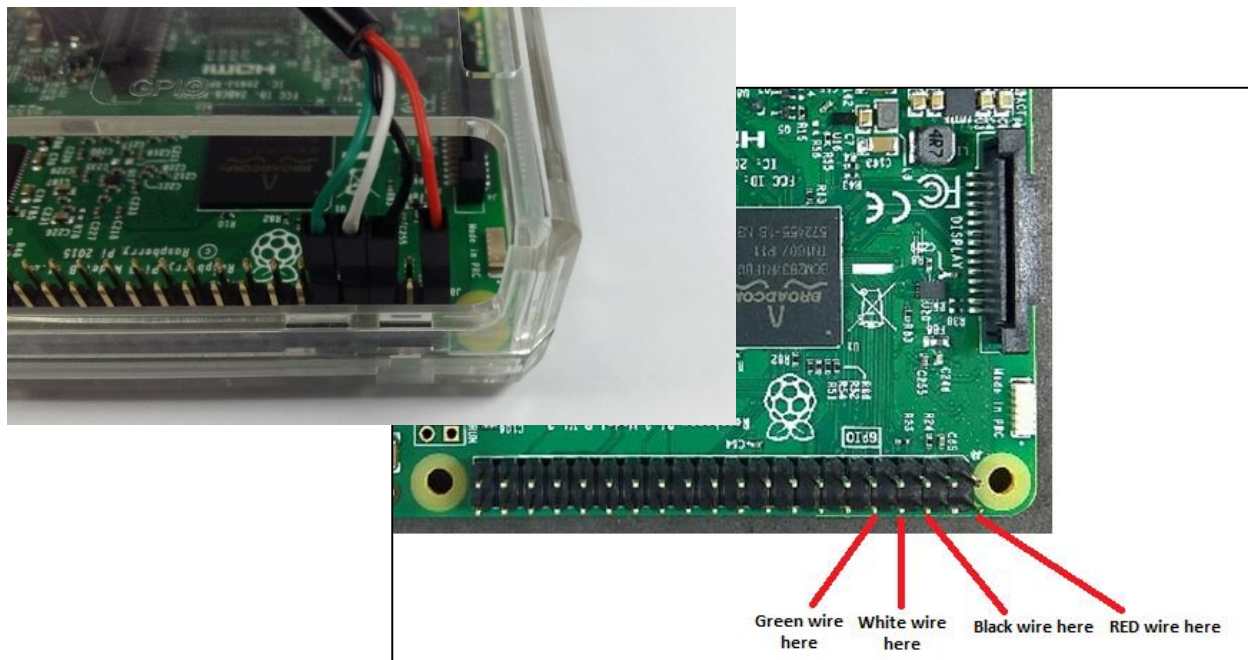
SHELL Terminal Command Line Interface Software "Putty.exe"

07 Setup Raspberry Pi's network using Serial Terminal

Before we can use wirelessly the Raspberry Pi from another computer with “headless” **SSH method**, we first need to connect our Raspberry Pi to our local network Router using the **Serial Terminal** method. The steps may be quite new for the first time, but will be easier as we go along. From your kit parts get the USB to Serial TTL Console cable and follow the diagram below in connecting to the Pi.

If you haven't used this Serial Console cable before, be sure to install first the software driver to your host computer. Follow the link below to download and install the software driver. Don't worry, the installation only takes a few seconds.
<https://drive.google.com/drive/folders/0B5RP5NsysFygOHd5ZTk3NmZ6LVE?usp=sharing>

Now that the software driver for the serial console cable is installed and we have the **COMXX** number we can now communicate with the Raspberry Pi's internal system. Connect the 4 pins of the Console Cable to the Raspberry Pi pins (shown in picture below). Note that you should NOT connect the included Power Supply Wall Adapter to the Pi, only the USB to Serial Console Cable.



Connection: Double Check!

Console Cable RED Wire to Raspberry Pi GPIO pin #2 (5 VDC)
Console Cable BLACK Wire to Raspberry Pi GPIO pin#6 (GND)
Console Cable WHITE Wire to Raspberry Pi GPIO pin#8 (TxD)
Console Cable GREEN Wire to Raspberry Pi GPIO pin#10 (RxD)

Other Option

Note with Caution: You can also use the **Power Supply Wall Adapter** included in the kit to power the Raspberry Pi instead of the **RED Wire (5Vdc)** from the console cable, but if you do this you should remove/disconnect the RED Wire from the Raspberry Pi first before you connect the Power Supply Wall Adapter. You choose one of the two options but **NOT both!** or you will risk of destroying your Pi. Try this method if you find the first method to be slow or buggy but the first method works well most of the time.

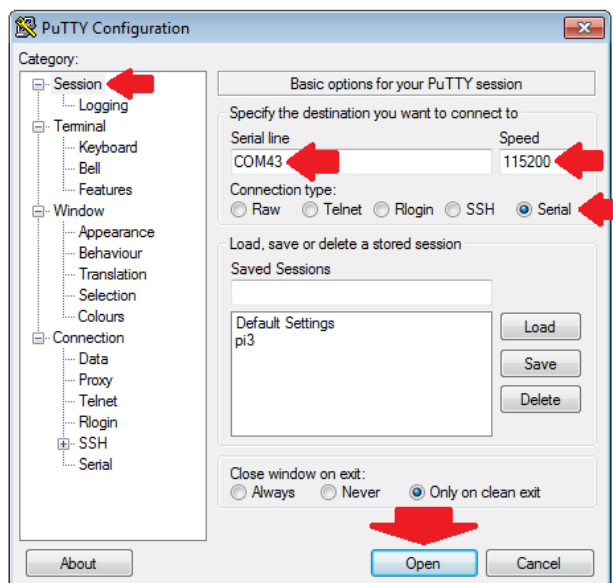
After connecting the power and communication pins (red-black-white-green) to the Raspberry Pi, connect the USB end of the the console cable to your computer. From the installation of the Console Cable earlier you should have the **COMXX** port number already. After connecting you will also notice that the Raspberry Pi's Green and Red LED will light up.



Access Raspberry Pi's system with Serial Terminal and Putty

From your host computer/laptop where your Pi is connected run the file/program **Putty.exe** which you downloaded earlier.

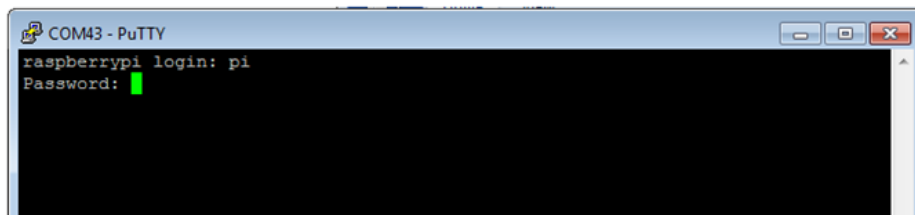
Click on **Session** then **Serial**. In the **Serial line box** type in the **COMXX** number that you got earlier from installing the Serial Console Cable. In this case my **COMXX** is **COM43**. In the **Speed box** type in **115200**. After entering all the details, click **Open**.



An Alert window will pop-up the first time, just click Yes to continue.

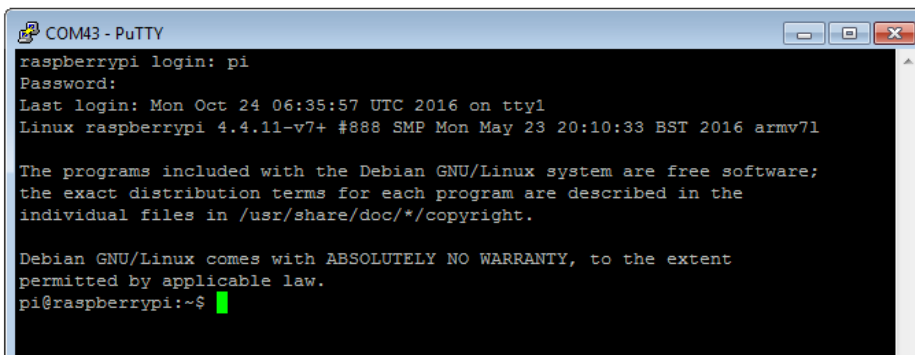
A black window will open, you are now in the **Command Line Interface (CLI)**. Here we could issue commands to the Raspberry Pi to do a job.

Some random text may appear the first time or wait for the **raspberrypi login:** to appear. Either way type the word **pi** and hit Enter. Then you will be asked for the password. By default type the word **raspberry** for the password and hit Enter again to log in with the Pi. Notice that when you type the password it will not appear on the screen.



```
COM43 - PuTTY
raspberrypi login: pi
Password: █
```

You just logged into the Raspberry Pi system!



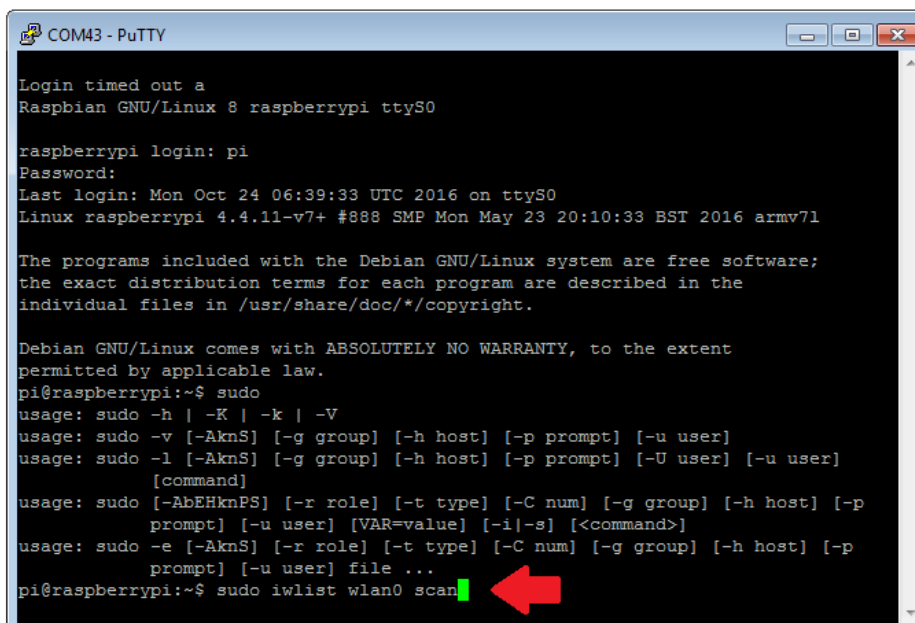
```
COM43 - PuTTY
raspberrypi login: pi
Password:
Last login: Mon Oct 24 06:35:57 UTC 2016 on tty1
Linux raspberrypi 4.4.11-v7+ #888 SMP Mon May 23 20:10:33 BST 2016 armv7l

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.
pi@raspberrypi:~$ █
```

Scan and Connect to your WiFi network

Type in **sudo iwlist wlan0 scan** and hit Enter.



```
COM43 - PuTTY
Login timed out a
Raspbian GNU/Linux 8 raspberrypi ttyS0

raspberrypi login: pi
Password:
Last login: Mon Oct 24 06:39:33 UTC 2016 on ttyS0
Linux raspberrypi 4.4.11-v7+ #888 SMP Mon May 23 20:10:33 BST 2016 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.
pi@raspberrypi:~$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user]
[command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
prompt] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p
prompt] [-u user] file ...
pi@raspberrypi:~$ sudo iwlist wlan0 scan █
```

```
COM43 - PuTTY
```

```
IE: Unknown: DD1A0C904C340B000000000000000000000000000000  
000000000  
  
IE: Unknown: DD0600E04C020160  
Cell 07 - Address: 84:55:A5:E5:68:8A  
Channel:11  
Frequency:2.462 GHz (Channel 11)  
Quality=63/70 Signal level=-47 dBm  
Encryption key:on  
ESSID:"bitstocmobile"  
Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s  
Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 18 Mb/s; 24 Mb/s  
36 Mb/s; 48 Mb/s; 54 Mb/s  
Mode:Master  
Extra:tsf=0000000000000000  
Extra: Last beacon: 90ms ago  
IE: Unknown: 000D62697473746F636D6F62696C65  
IE: Unknown: 010482848B96  
IE: Unknown: 03010B  
IE: Unknown: 2A0100  
IE: Unknown: 320A0C1218243048606C  
IE: IEEE 802.11i/WPA2 Version 1  
Group Cipher : CCMP  
Pairwise Ciphers (1) : CCMP  
Authentication Suites (1) : PSK  
IE: Unknown: 2D1A2C0103FF000000000000000000000000000000  
000000000
```

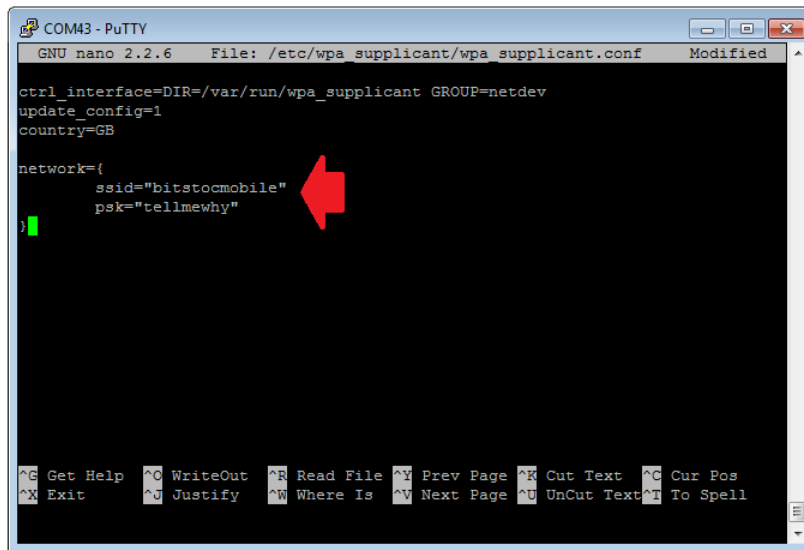
[illegible]

Page 12

You cannot use the mouse to go over in the file. To do this, use the arrow keys in your keyboard to navigate in the black text area. Go to the bottom and add the following...

```
network={
    ssid="your_wifinetwork_name"
    psk="your_wifi_password"
}
```

In our case our network name is **bitstocmobile** and the password (psk) is **tellmewhy**. Replace the **ssid** and **psk** with your WiFi Network name and WiFi network password.

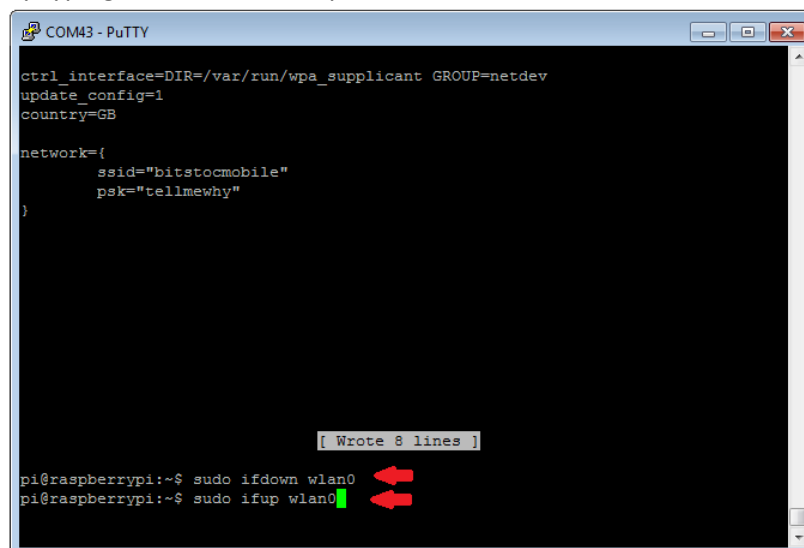


```
COM43 - PuTTY
GNU nano 2.2.6 File: /etc/wpa_supplicant/wpa_supplicant.conf Modified
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=GB

network={
    ssid="bitstocmobile"
    psk="tellmewhy"
}
```

Now save the file by **pressing Ctrl+X** then **Y**, then finally **press Enter**.

At this state, the wpa-supplciant file should notice change in the credentials within a few seconds and will try connecting in to the WiFi network. If it does not, you can manually restart the interface by typing **sudo ifdown wlan0** and hit Enter then type **sudo ifup wlan0** and hit enter again, or reboot your Raspberry Pi by typing **sudo reboot** and press Enter.

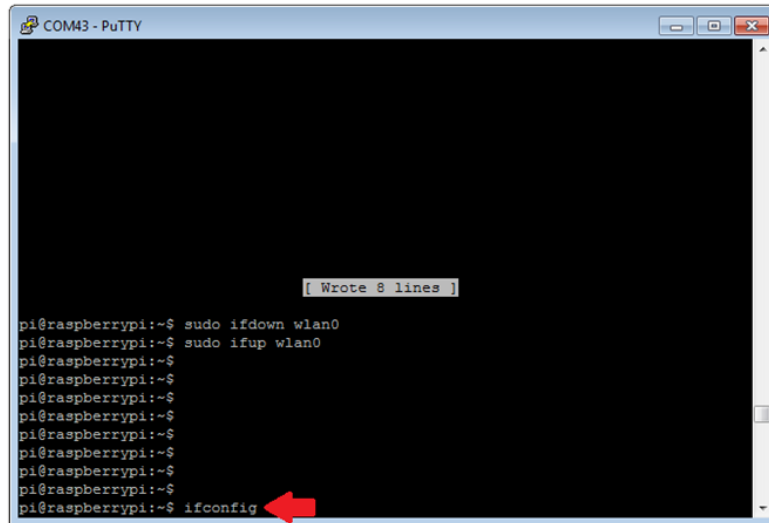


```
COM43 - PuTTY
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=GB

network={
    ssid="bitstocmobile"
    psk="tellmewhy"
}

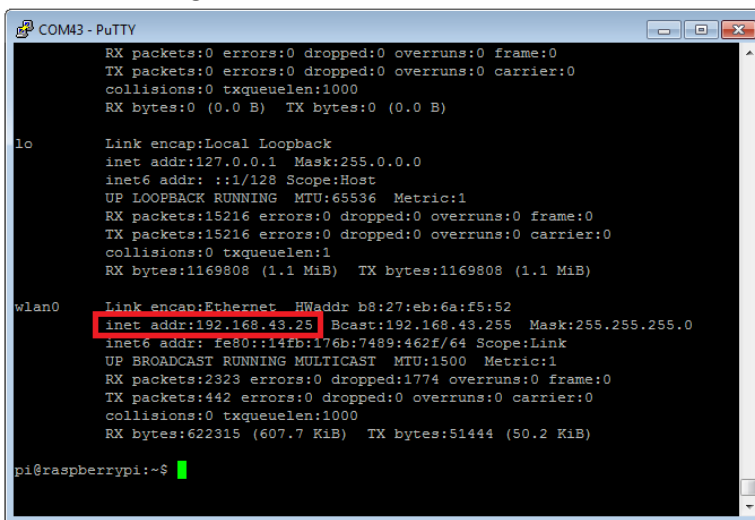
[Wrote 8 lines]
pi@raspberrypi:~$ sudo ifdown wlan0
pi@raspberrypi:~$ sudo ifup wlan0
```


Verify that your Raspberry Pi has successfully connected to your network by typing **ifconfig**.



```
pi@raspberrypi:~$ sudo ifdown wlan0
pi@raspberrypi:~$ sudo ifup wlan0
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$
pi@raspberrypi:~$ ifconfig
```

If the **inet addr:** portion has an IP address beside it, your Raspberry Pi is connected to your Router/WiFi network. If not, check that your WiFi name and password are correct by opening again the **wpa-supplciant** using nano text editor by following the previous steps. In this case our WiFi Router has assigned an IP Address to our Pi which is **192.168.43.25**. Take note of your IP Address as we will need it in connecting to our Pi later using the **SSH** method.



```
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

lo    Link encap:Local Loopback
      inet addr:127.0.0.1 Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
      UP LOOPBACK RUNNING MTU:65536 Metric:1
      RX packets:15216 errors:0 dropped:0 overruns:0 frame:0
      TX packets:15216 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1
      RX bytes:1169808 (1.1 MiB) TX bytes:1169808 (1.1 MiB)

wlan0 Link encap:Ethernet HWaddr b8:27:eb:6a:f5:52
      inet addr:192.168.43.25 Bcast:192.168.43.255 Mask:255.255.255.0
      inet6 addr: fe80::14eb:176b:7489:462f/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:2323 errors:0 dropped:1774 overruns:0 frame:0
      TX packets:442 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:622315 (607.7 KiB) TX bytes:51444 (50.2 KiB)

pi@raspberrypi:~$
```

Type **sudo poweroff** or type **exit** and hit Enter to exit from the Pi and close the Terminal window. Then remove the Console cable connected to the Pi and computer.

After removing the Console Cable, power up your Pi by connecting the included Power Supply Adapter to the USB power port of the Raspberry Pi and to your power outlet.

Our Raspberry Pi will connect to the network Router and have its own IP address. We can now program the Pi wirelessly (without having the wired console cable connection) using the **SSH “headless” method**.

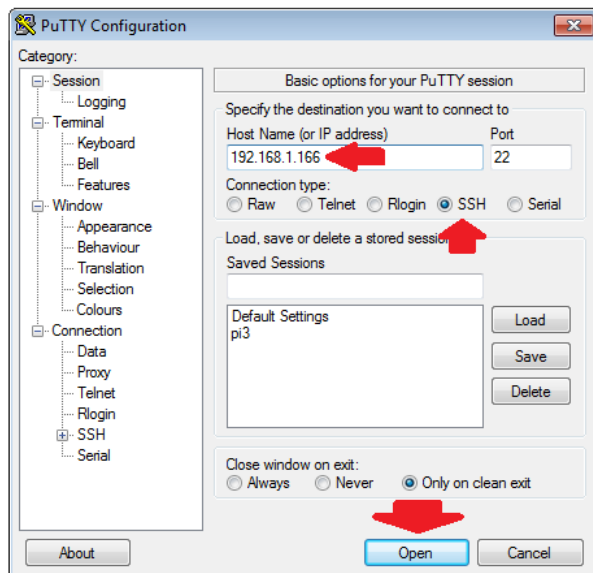
If you wish to connect to a new or other WiFi network Router the next time, simply do the same step as to opening the **wpa_supplicant.conf** using the **nano** text editor, put your wifi credentials and save it!

08 Using SSH ("headless")

SSH (Secure Shell) is the widely used method in using the Pi. SSH is a Linux feature that gives secure terminal communication to the Raspberry Pi from your controller host computer. This gives you wireless access to the Raspberry Pi and issue commands using the Command Line Interface. Here you can issue any commands to the Pi to do any kind of job.

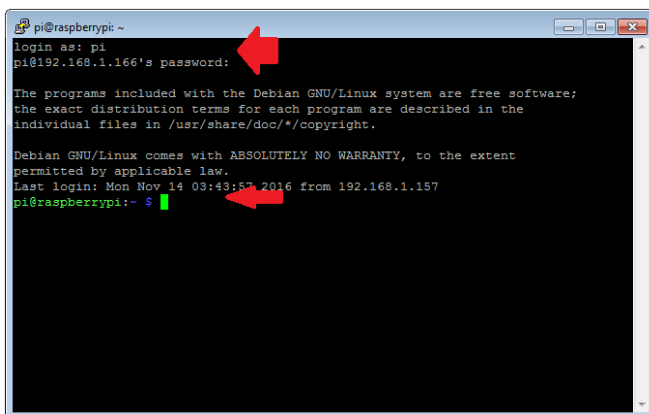
To start, open again **Putty**. In the **Host Name IP Address** type in the IP address you got from the previous step. In this case we connected to a different Router using the steps earlier and we have an IP Address of **192.168.1.166** assigned to the Raspberry Pi. If you did not connect to a different router use the same IP address you got from the previous step.

Click **SSH** from the radio button selection and then click **Open**.



A Putty Alert dialogue box may pop-out in your first use, choose **Yes** button and you will proceed.

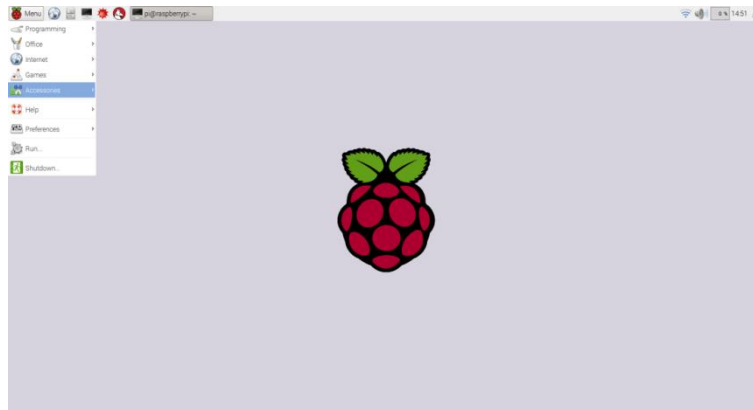
Then the command line interface will appear. Just the like the Serial Terminal, enter **pi** as login and enter **raspberrypi** as password. A welcome message will appear and you will be in the Pi's home directory



Now we can issue commands to the Pi using SSH wirelessly. In the next part we will be exploring this interface by downloading and installing programs into the Pi.

09 Remote Control with VNC (view Pi's Graphical Interface)

VNC (Virtual Network Connection) software allows you to view the Raspberry Pi's desktop graphical interface from another computer (such as a desktop computer or a laptop) without any external accessories connected to your Pi. This setup allows you to see your mouse hover on to the screen as if you are in front of an actual computer with your Pi.



Raspberry Pi's graphical interface, the Raspbian Jessie version.

The set-up:

Raspberry Pi (Running a VNC Server software)

Host computer (Running a VNC Client software)

First, install VNC software into your Pi. There are a number of VNC Server available and the commonly used one is **tightvnc** (www.tightvnc.com). From your previous window you are already logged into the Pi.

Prepare the Server (Raspberry Pi)

From your opened Putty terminal, type the following commands below with Enter key every after line.

```
pi@raspberrypi: ~ cd
pi@raspberrypi: ~ sudo apt-get install tightvncserver
pi@raspberrypi: ~ vncserver :1
```

Note that there is a "space" between letter "r" and ":" sign. Now a VNC server is running at channel 1.

```
pi@raspberrypi: ~
pi@raspberrypi:~ $ cd
pi@raspberrypi:~ $ sudo apt-get install tightvncserver
Reading package lists... Done
Building dependency tree
Reading state information... Done
tightvncserver is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 136 not upgraded.
pi@raspberrypi:~ $ vncserver :1

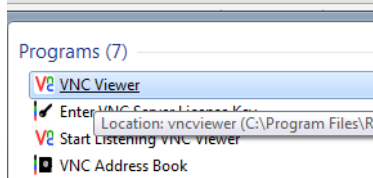
New 'X' desktop is raspberrypi:1

Starting applications specified in /home/pi/.vnc/xstartup
Log file is /home/pi/.vnc/raspberrypi:1.log
pi@raspberrypi:~ $
```

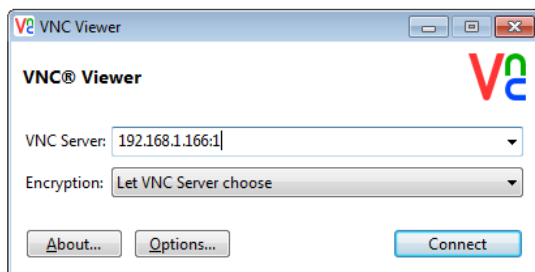
Prepare the Client (Host Computer)

Download a VNC Client software to your computer from here (<https://www.realvnc.com/>). Go to the website's Downloads tab, select your operating system and download the file. Install it to your computer and follow the installation. The file downloaded may install both the Server and Client. It will prompt to install a license for the Server part but neglect it since we will only use the Client **"VNC Viewer"**.

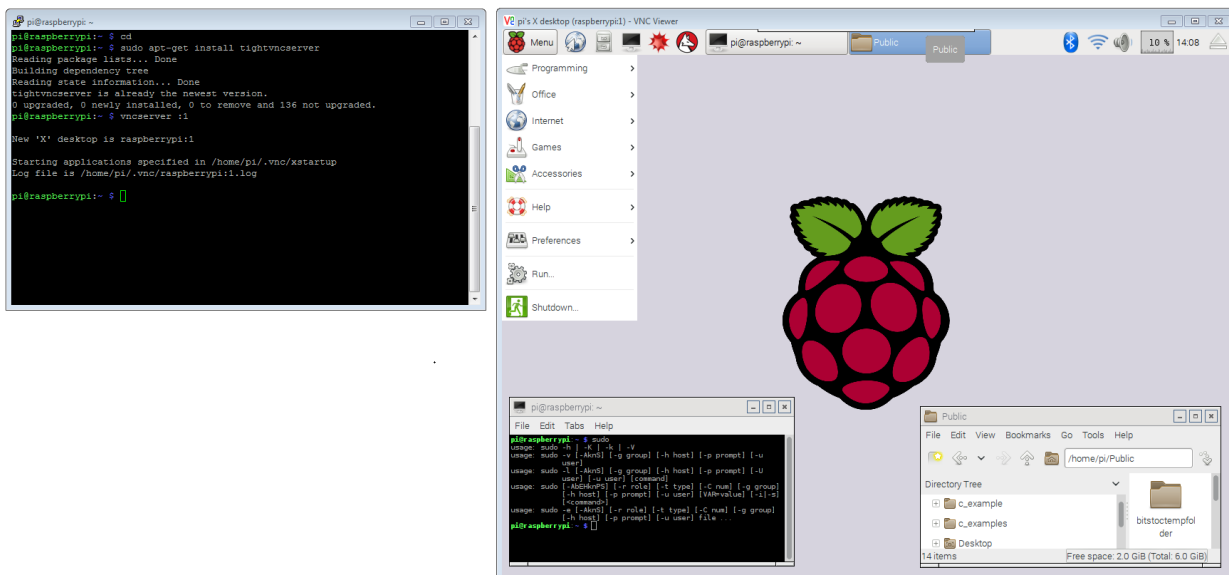
Locate and Run the VNC Viewer from your computer.



Now Enter your Pi's IP Address followed by the characters **" :1 "** since we are running a vnc server in our Pi in channel 1. In this case we Enter as shown below.

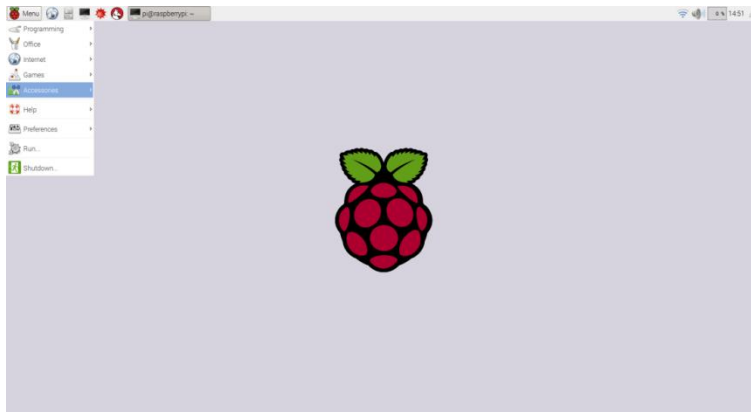


Click *Connect*. A window will pop out for connection confirmation. Just click the *Continue* button. You will then see a window showing the home desktop screen of your Raspberry Pi (with the Raspberry Pi logo).



Go ahead and explore your Raspberry Pi computer like clicking on the web browser or file manager or the Terminal software.

10 Raspberry Pi Complete Desktop Computer Setup



You can setup a complete Raspberry Pi Complete Computer Desktop by connecting accessories to the boards such as a display monitor, mouse, keyboard and including the Power Supply Adapter to power the Pi and an SD card with the Raspbian OS. If your monitor is a VGA type display connector you can get an **HDMI to VGA Adapter Cable** available in the shop.



Our Raspberry Pi Computer desktop setup

After connecting the power you will see the Pi boot up, then the Pi's desktop with a Raspberry logo in the middle. Go and explore your Pi computer like accessing the "internets" and the "googles" by opening the web browser or view the files in the File Manager by clicking through the icons at the top left corner. What you do in "headless" setup can also be done in this set-up by simply opening the **Terminal** program.

Updating and Upgrading

You need to do some update and upgrade on the Pi's system once in a while in order to be able to run some new updated software and other packages. To do this simply issue the commands below in your Terminal.

```
pi@raspberrypi: ~ sudo apt-get update  
pi@raspberrypi: ~ sudo apt-get upgrade
```

Rebooting and Powering Off

To reboot or turn off the Pi, simply issue the commands referred below.

```
pi@raspberrypi: ~ sudo reboot  
pi@raspberrypi: ~ sudo poweroff
```

What's Next...

Now we have finished configuring and setting up our Raspberry Pi development environment and even showing an example of a complete setup of a Raspberry Pi computer desktop. Next up we will go over to topic GPIO and we will learn how we can connect some electronic hardware and create projects involving sensors, LED's, buttons, actuators and all other electronics module.

The Raspberry Pi is a powerful platform in embedded electronics projects with the same excitement and project possibilities of an Arduino. Not only that, there are many projects that you can run on a raspberry pi as an extension of your Arduino project environment...

Parts List

- Raspberry Pi 3 Model B+ (B plus)
- Breadboard solderless - Full size 830points
- Raspberry Pi GPIO Adapter - T Cobbler (with ribbon cable)
- Connecting Wires - Male-Male, 65pcs (assorted-size)
- MicroSD Card with Raspberry Pi OS (8GB, class10)
- LED - Red (5mm)
- LED - Green (5mm)
- LED - Yellow (5mm)
- Push Button tactile switch (SPDT)
- Resistors (330 ohms 10pcs)
- Resistors (10k ohms 10pcs)
- Resistors (1k ohms 10pcs)
- LDR (Light Dependent Resistor)
- Power Supply Wall Adapter - 5V/2A (2000mA) (micro-USB)
- Raspberry Pi Case (Clear)
- MCP3008 - 8-Channel 10-Bit ADC With SPI Interface
- USB to RS232 Serial TTL Converter Cable (PL2303)

Quick Links

- <https://www.raspberrypi.org/>
- <https://www.raspbian.org/>
- <https://drive.google.com/drive/folders/0B5RP5NsysFygOHd5ZTk3NmZ6LVE?usp=sharing>
(USB to Serial Console Cable installation guide)