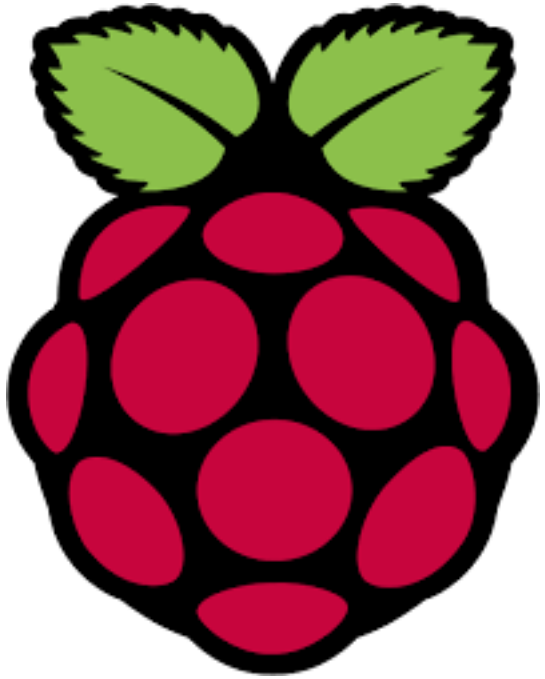
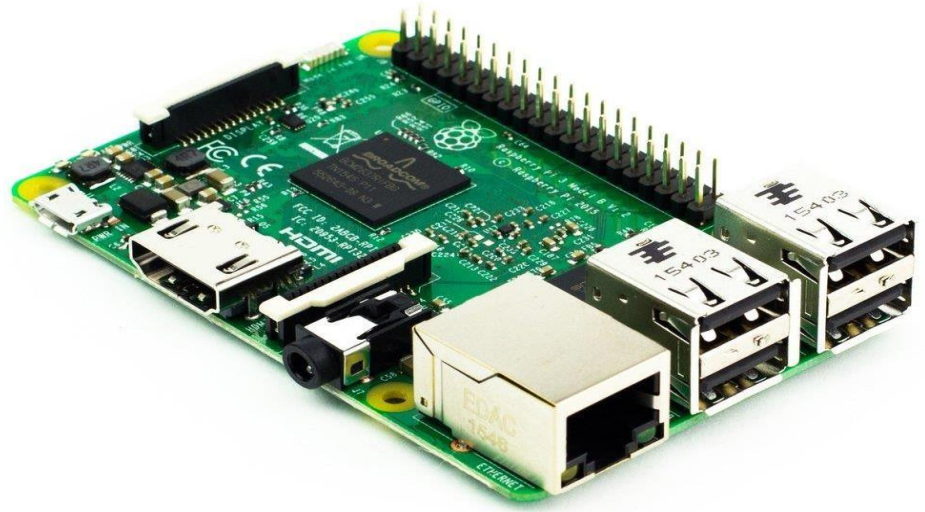
Raspberry Pi Training





Dr. Vinay Chamola, Assistant Professor, EEE Department, BITS-Pilani

Agenda

- Introduction
- Pin Diagram
- Setup
- SSH
- RealVNC
- Wireless remote access
- Running program on Pi
- LED blink
- LED PWM

What is Rpi

- The **Raspberry Pi** is a series of small single-board omputers developed in the United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and in developing countries.
- The original model became far more popular than anticipated, selling outside its target market for uses such as robotics.
- It does not include peripherals (such as keyboards, mice and cases). However, some accessories have been included in several official and unofficial bundles.

History

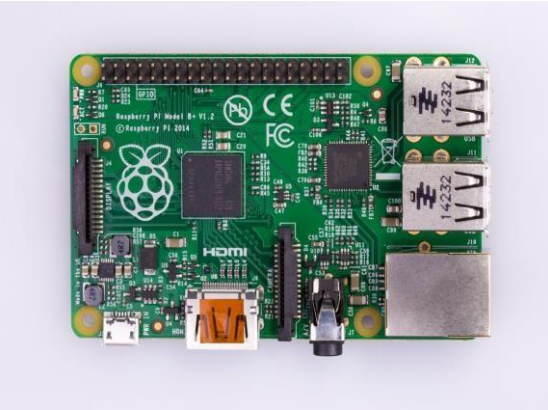
- In 2006, early concepts of the Raspberry Pi were based on the Atmel ATmega644 microcontroller. Its schematics and PCB layout are publicly available. Foundation trustee Eben Upton assembled a group of teachers, academics and computer enthusiasts to devise a computer to inspire children.
- The Model A, Model B and Model B+ names are references to the original models of the British educational BBC Micro computer, developed by Acorn Computers.

Rpi Boards

Rpi 3 model B

Rpi 2 model B

Rpi 1 model B+



Rpi Zero W

Rpi Zero

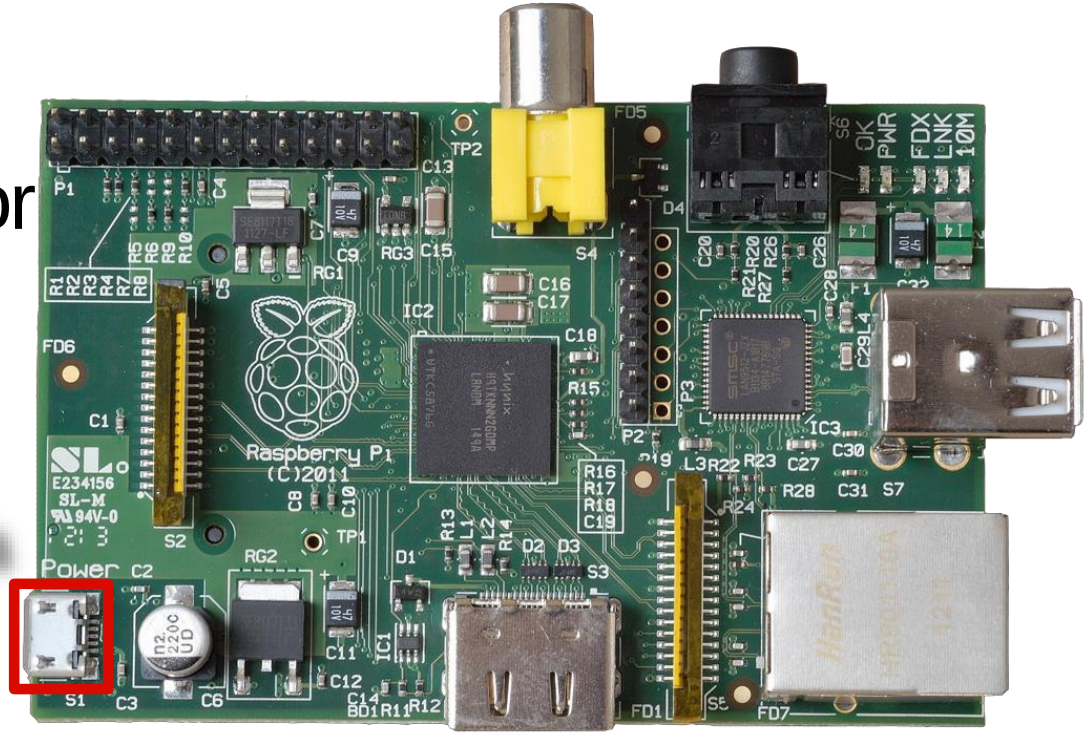
Rpi Industrial Compute module 3

Why RPi

- Very Low Cost
- Great tool for **Learning Programming**, Computers & Concepts of **Embedded Linux**, etc
- Support for **all Age Groups** (School Children, College Undergraduates, Professional Developers, Programmers)
- Supports & runs **Free** and **Open Source** Linux OS
- Consumes **less than 5W** of Power
- Supports **Full HD Video** Output (1080p), Multiple USB Ports , etc
- Fun to learn & explore. You are limited by your imagination

Power

5v micro USB connector

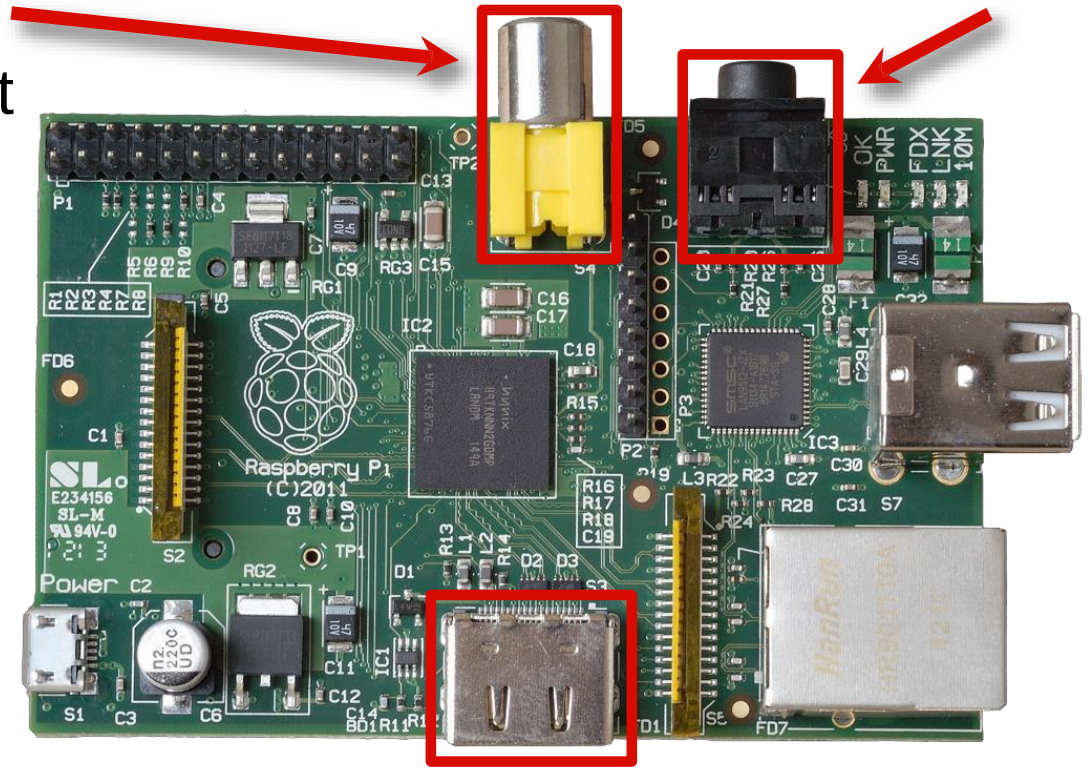


(Similar to the one on a lot of mobile phones!)

A/V (Audio/Video)

RCA Video
(works with most
older TVs)

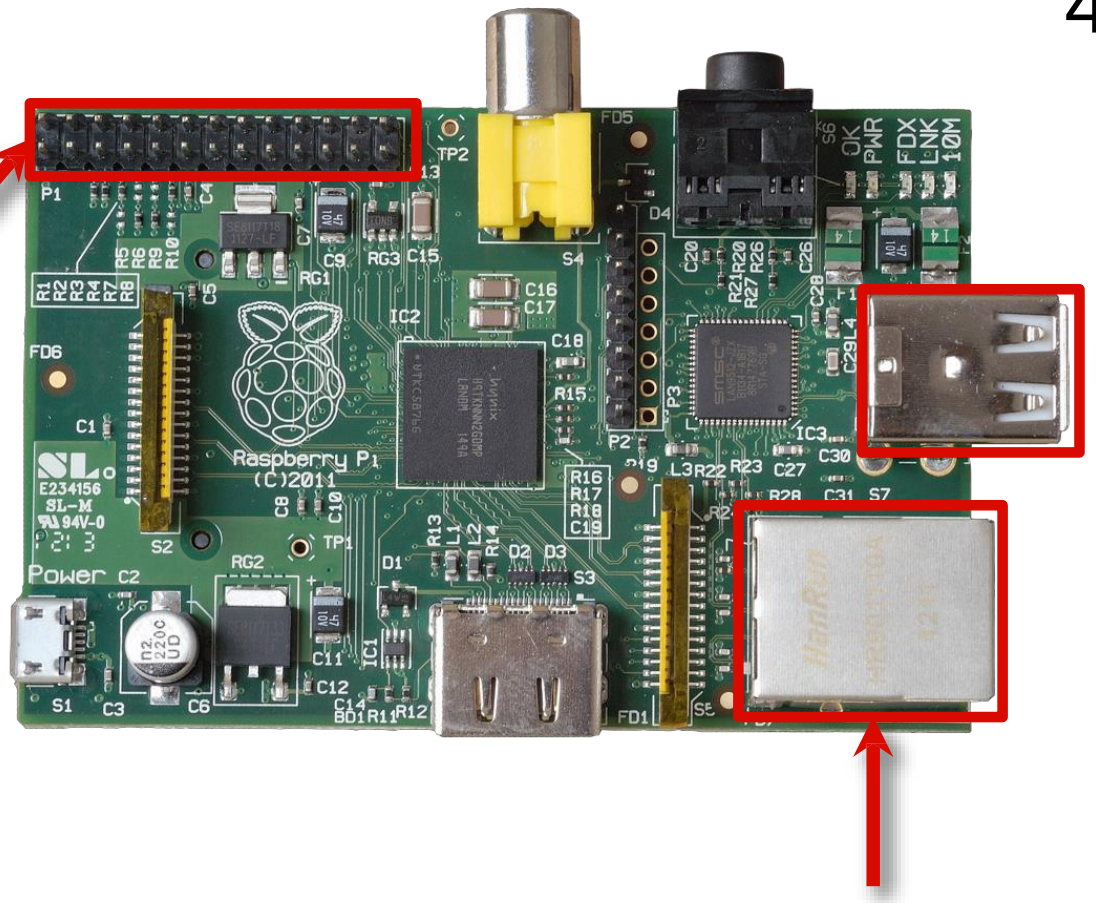
3.5mm Audio
Standard
headphone
socket



HDMI Audio & Video

Connectivity

GPIO (General Purpose Input & Output)

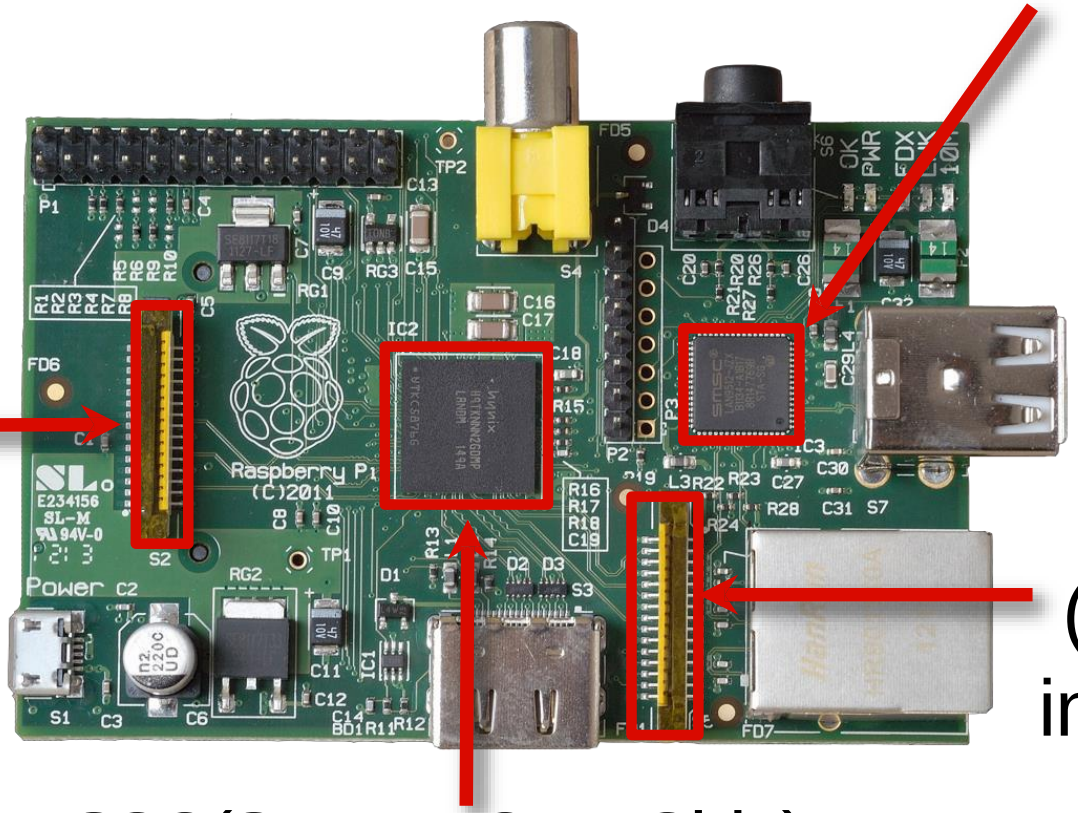


4 x USB 2.0 ports

Internals

LAN Controller

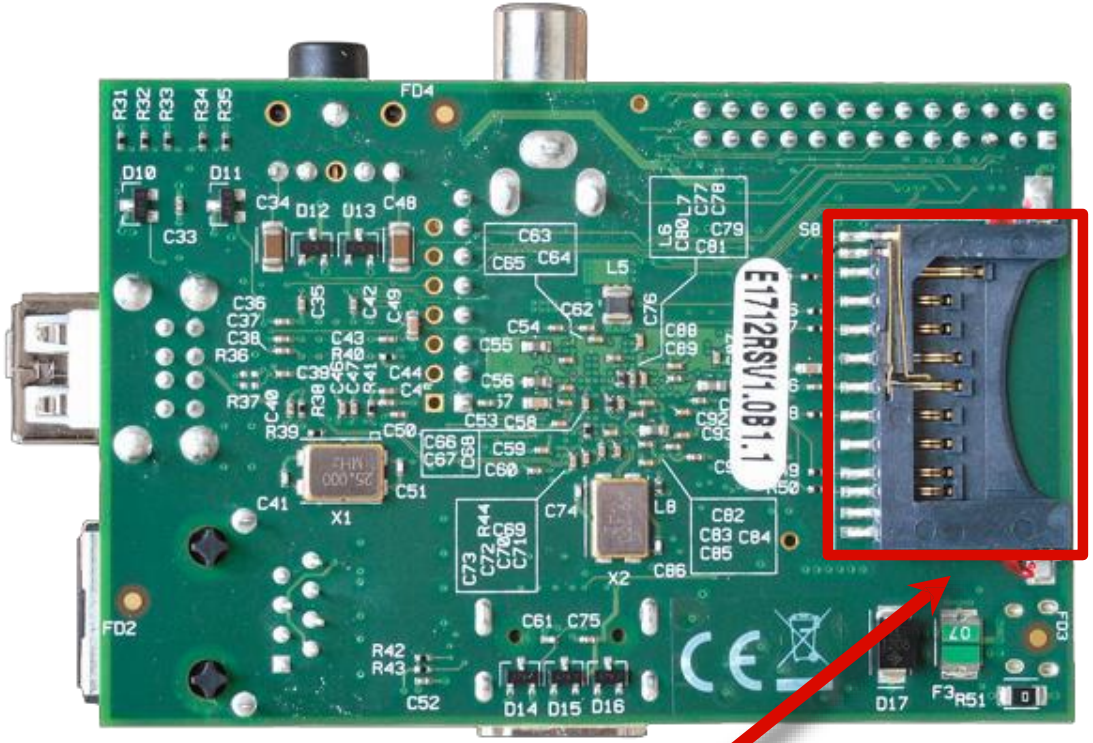
DSI
(display
interface)



CSI
(camera
interface)

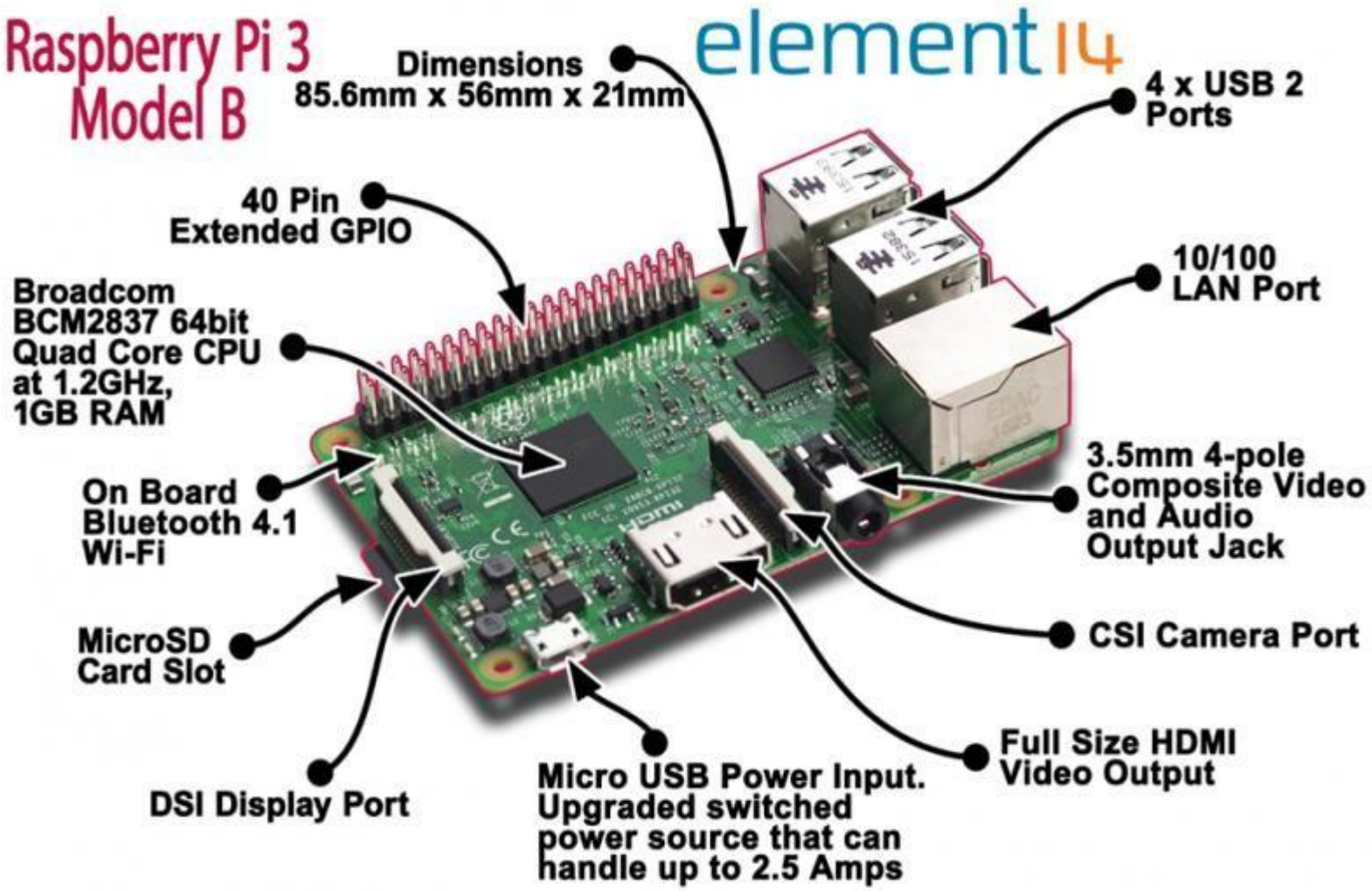
SOC(System On a Chip)
Broadcom BCM2835 700Mhz

Storage

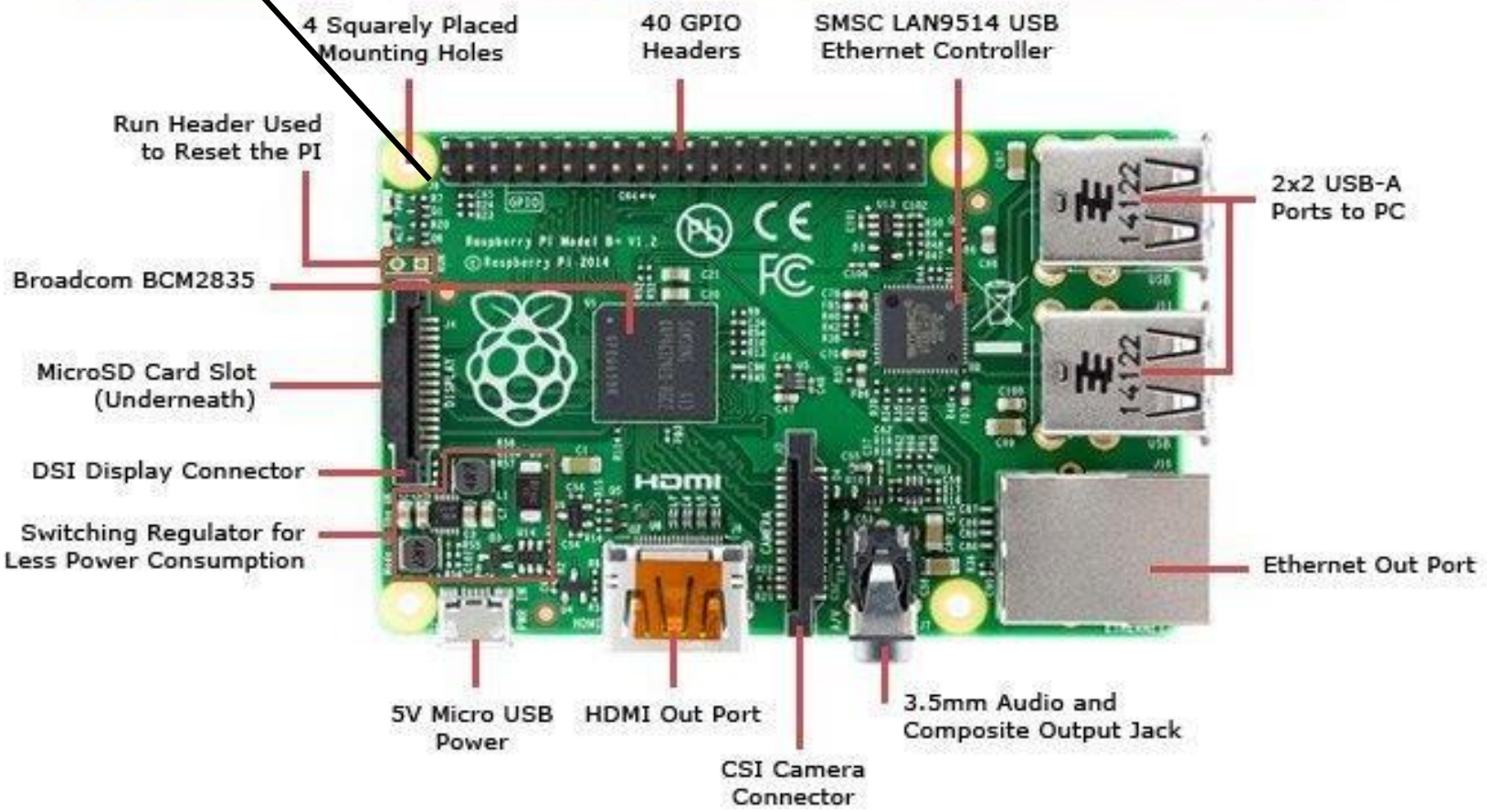


SDCard Slot
(supports SDcards up to 32GB)

Rpi 3 Module B



GPIO Pinout Diagram



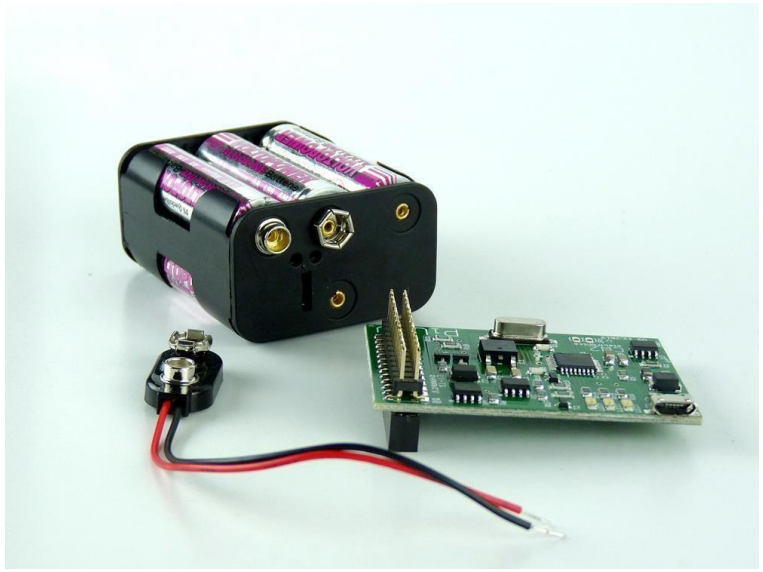
Raspberry Pi 3 Model B (J8 Header)

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
30	SDA0 (I2C ID EEPROM)	27	28	SCL0 (I2C ID EEPROM)	31
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

Add-ons



Handle with care

- The Raspberry Pi was built to be used, but not abused. Every P.C.B. should be handled with care.
- **Handle the Pi only by the edges** of the board itself. **Avoid touching or holding** any of the components on the board.
- **Rough Handling** can cause solder points to fail and may result in **short circuits**, but be careful with the GPIO pins as well. Best practice: Use the Rpi Case
- While connecting Hardware, double check the connections and then power it up. Raspberry Pi GPIO's are **not 5V tolerant**.

Ground

- The Ground pins on the Raspberry Pi are all electrically connected, so it doesn't matter which one you use if you're wiring up a voltage supply.
- Generally the one that's most convenient or closest to the rest of your connections is tidier and easier, or alternatively the one closest to the supply pin that you use.

WiringPi

- WiringPi is an attempt to bring Arduino-wiring-like simplicity to the Raspberry Pi.
- The goal is to have a single common platform and set of functions for accessing the Raspberry Pi GPIO across multiple languages. WiringPi is a Clibrary at heart, but it's available to both Ruby and Python users who can "gem install wiringpi" or "pip install wiringpi2" respectively.
- WiringPi uses its own pin numbering scheme.

Setting up Raspberry pi (NOOBssetup)

- **How to get and installNOOBS?**
- **BUY PREINSTALLED SD CARD**

1. SDcards with NOOBSpreinstalled are available from many distributors and independent retailers, such as Pimoroni, Adafruit and The Pi Hut. For older models of Raspberry Pi, you'll need a full-size SDcard; for the Pi Zero, A+, B+, Pi 2 and Pi 3 you'll need a micro SDcard.

- **DOWNLOAD**

1. recommended to use an SDcard with a minimum capacity of 8GB.
2. Using a computer with an SDcard reader, visit the Downloads page.
3. Click on **NOOBS**, then click on the Download ZIP button under 'NOOBS (offline and network install)', and select a folder to save it to.
4. Extract the files from the zip.

Contd…

- **FORMAT YOUR SD CARD**

1. It is best to format your SD card before copying the NOOBS files onto it. To do this:
2. Visit the SD Association's website and download SD Formatter 4.0 for either Windows or Mac.
3. Follow the instructions to install the software.
4. Insert your SD card into the computer or laptop's SD card reader and make a note of the drive letter allocated to it, e.g. G:/
5. In SD Formatter, select the drive letter for your SD card and format it.

Contd…

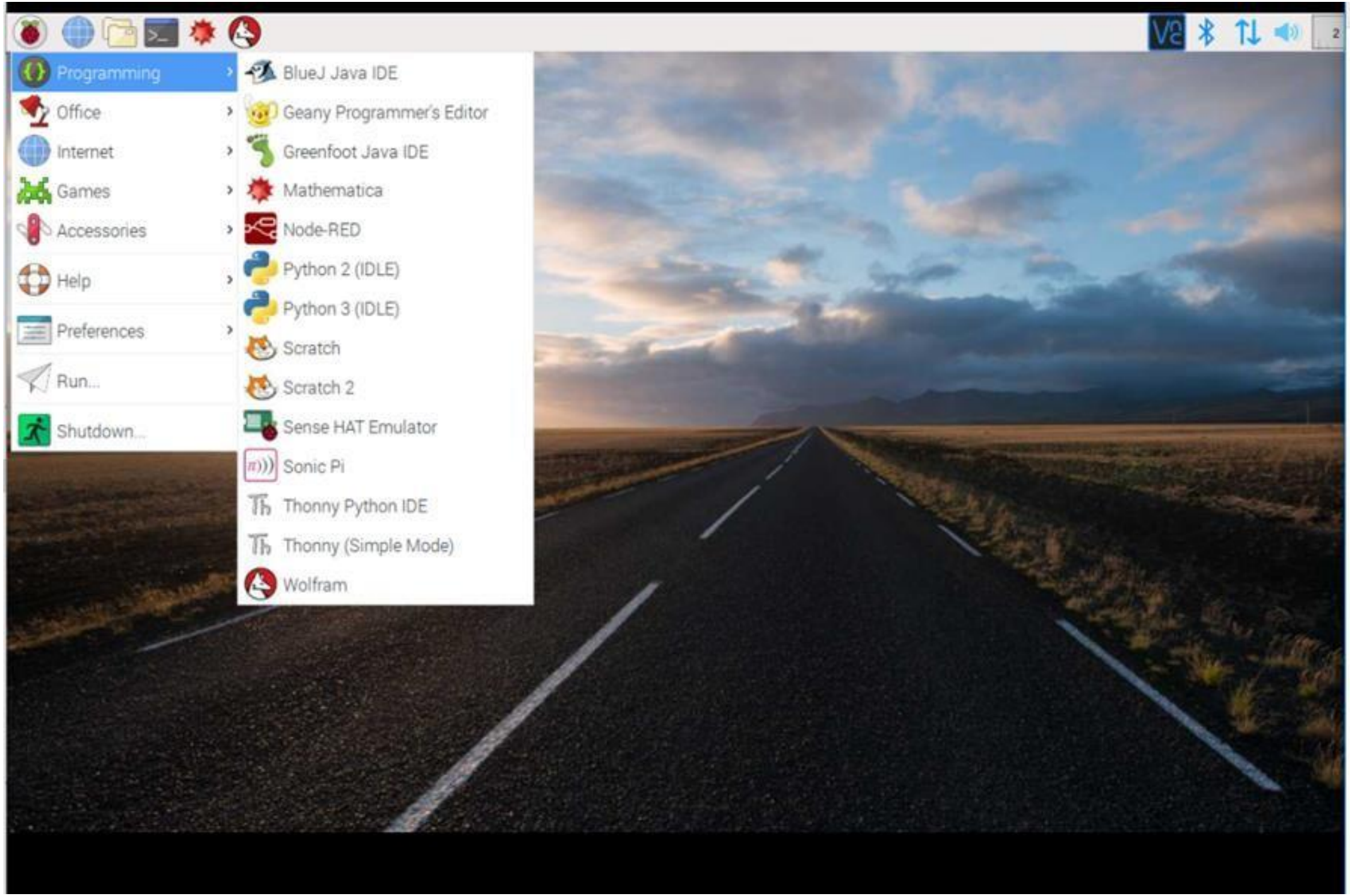
- **DRAG AND DROP NOOBS FILES**

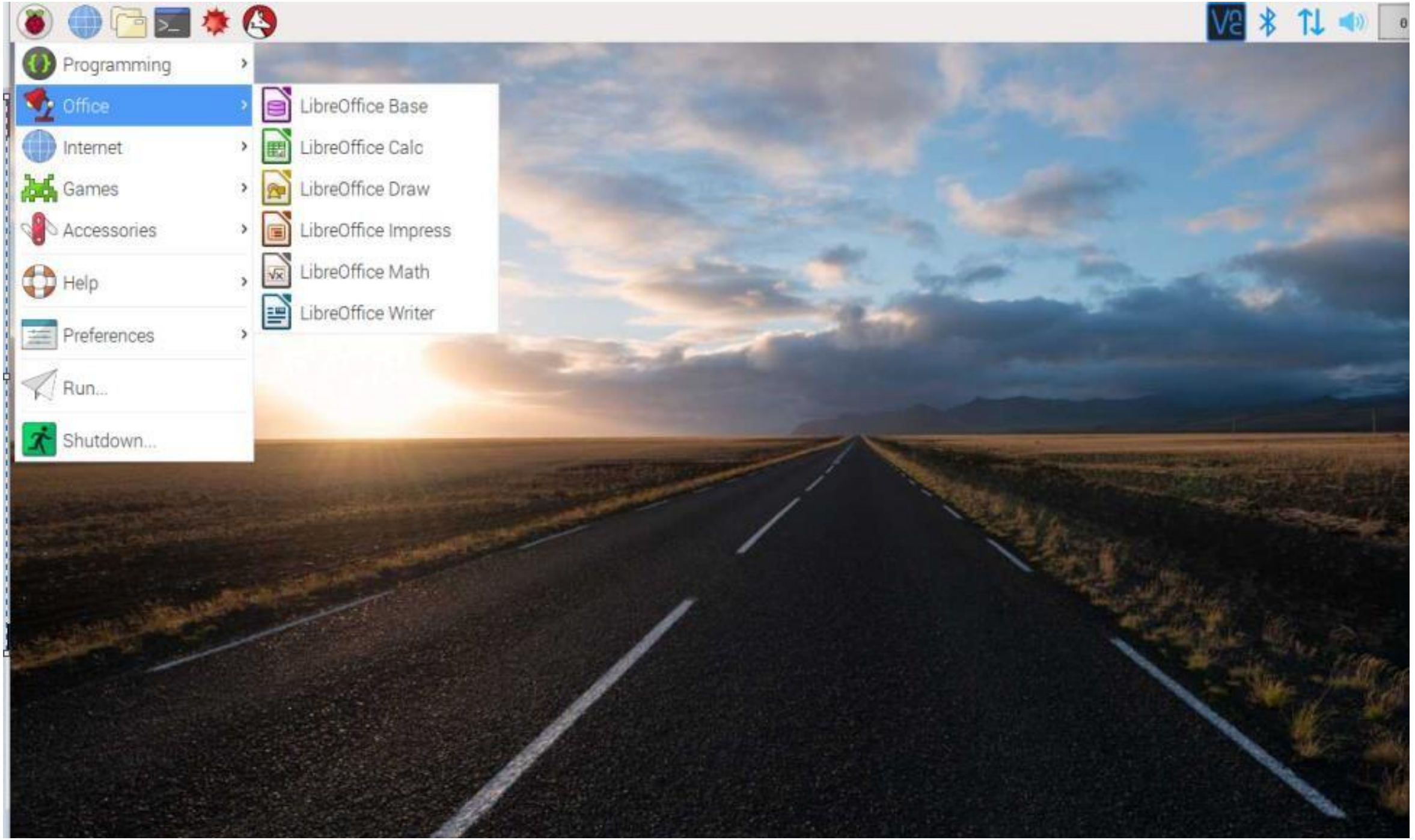
1. Once your SD card has been formatted, drag all the files in the extracted NOOBS folder and drop them onto the SD card drive.
2. The necessary files will then be transferred to your SD card.
3. When this process has finished, safely remove the SD card and insert it into your Raspberry Pi.

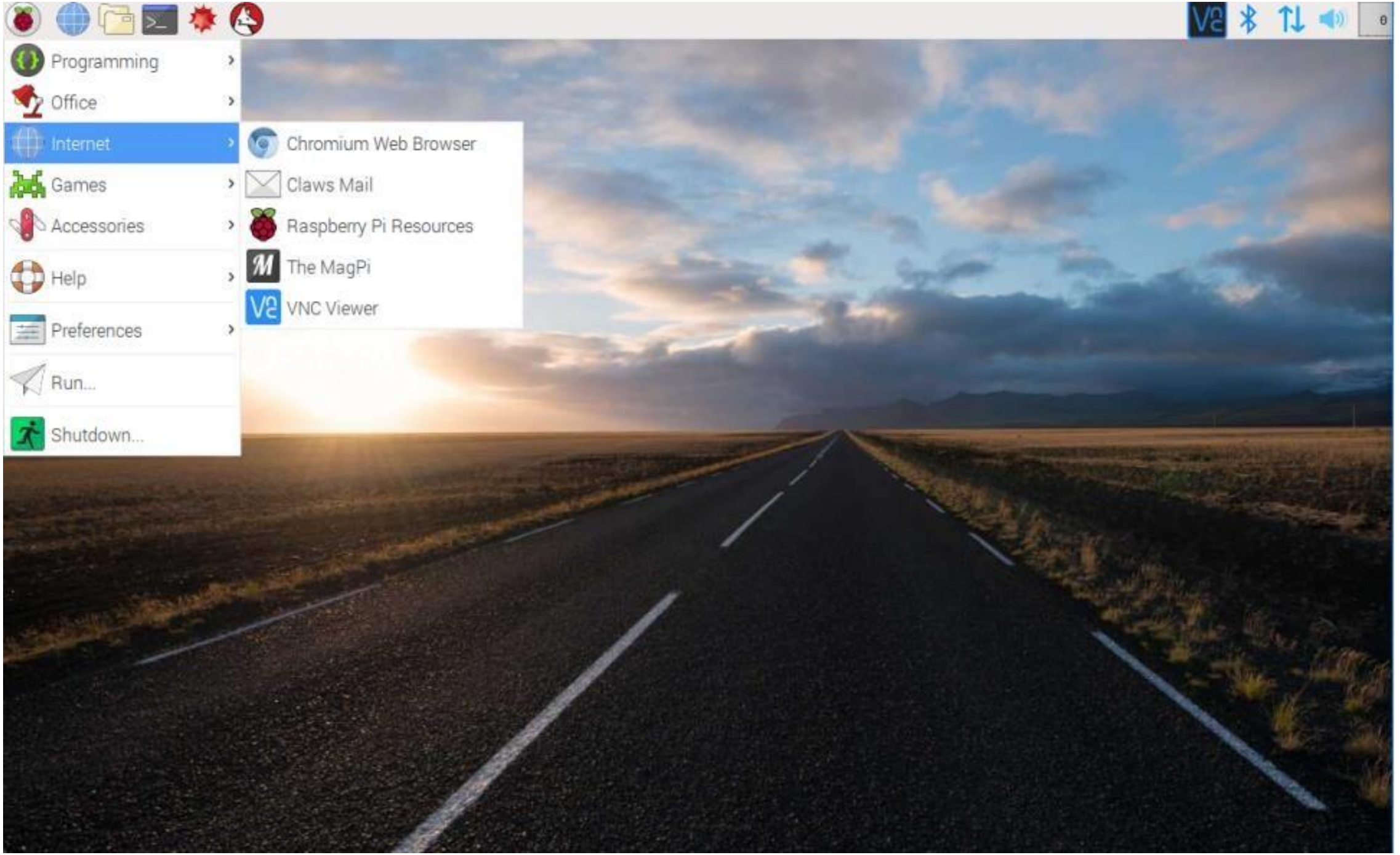
Contd…

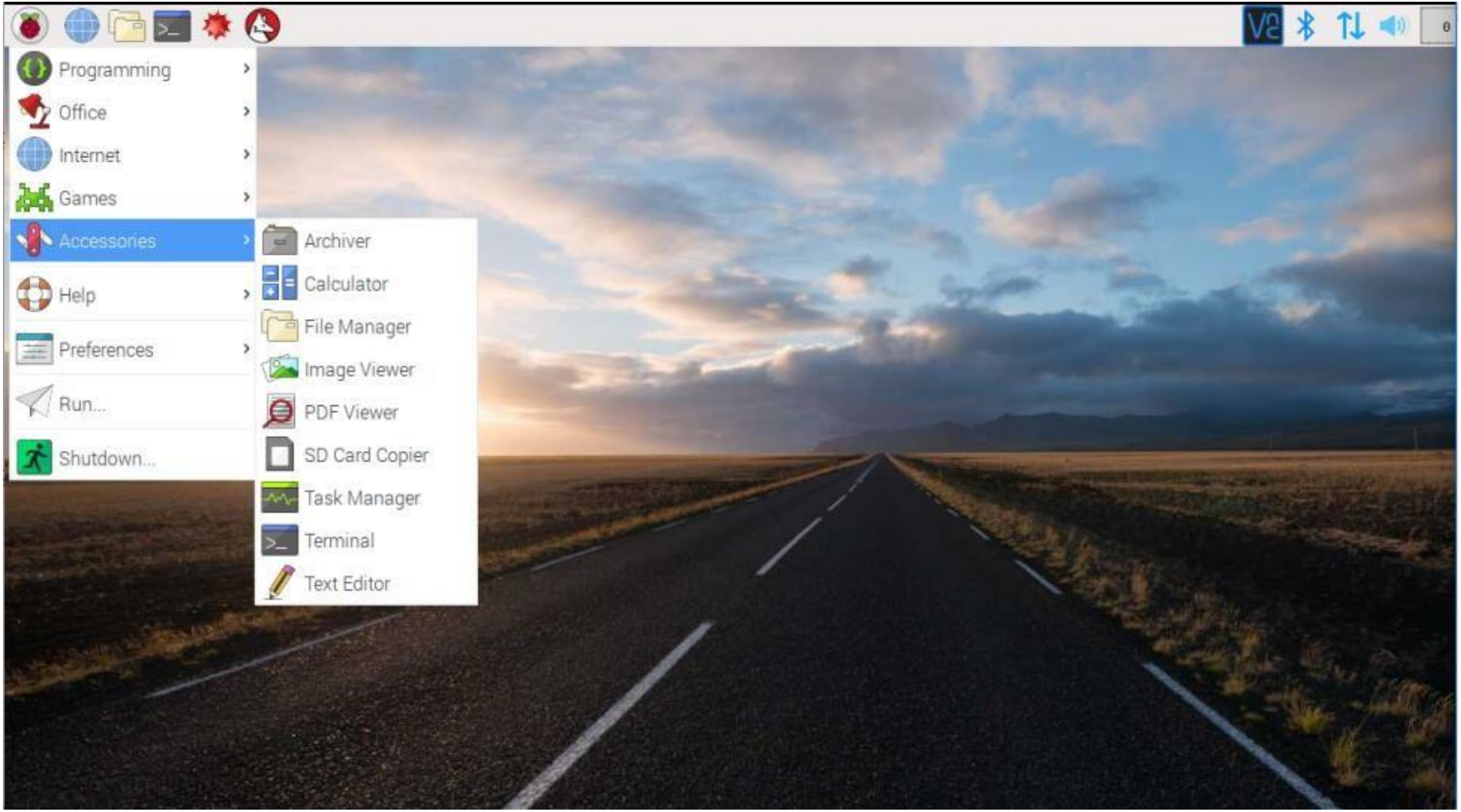
- **FIRST BOOT**

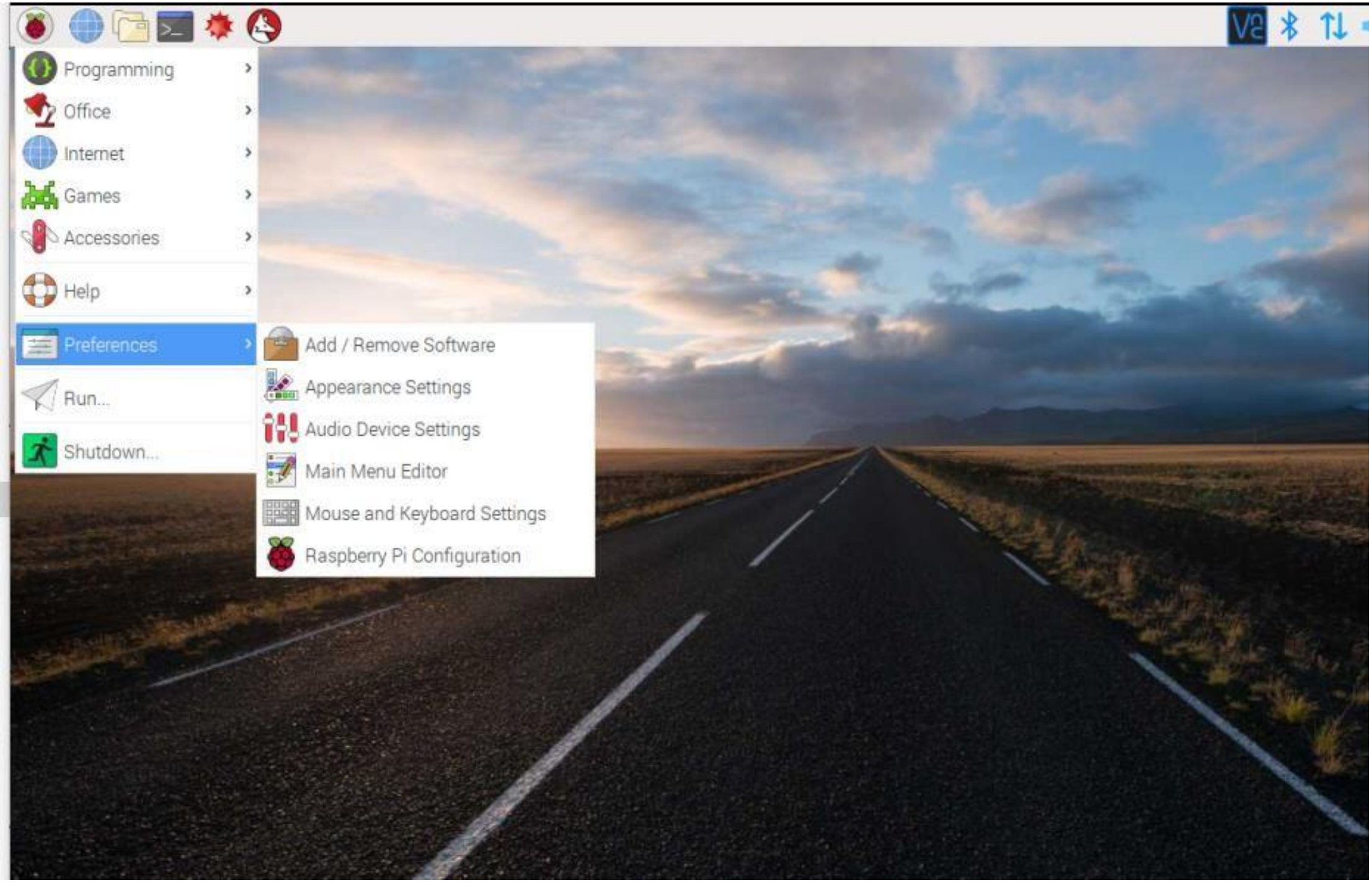
1. Plug in your keyboard, mouse, and monitor cables.
2. Now plug the USB power cable into your Pi.
3. Your Raspberry Pi will boot, and a window will appear with a list of different operating systems that you can install. We recommend that you use Raspbian – tick the box next to Raspbian and click on Install.
4. Raspbian will then run through its installation process. Note that this can take a while.
5. When the install process has completed, the Raspberry Pi configuration menu (raspi-config) will load. Here you are able to set the time and date for your region, enable a Raspberry Pi camera board, or even create users. You can exit this menu by using **Tab** on your keyboard to move to Finish.

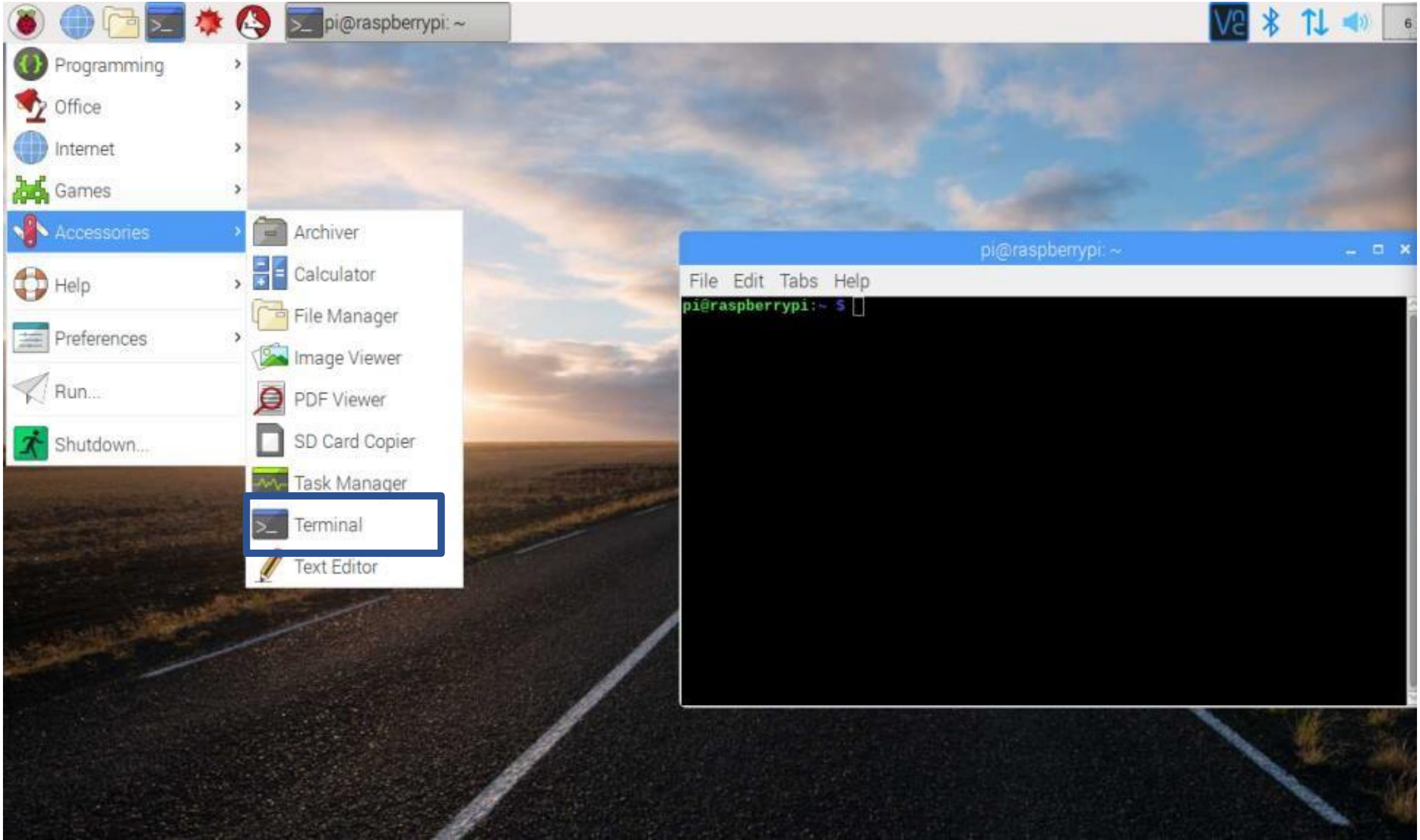

Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
BlueJ Java IDE
Geany Programmer's Editor
Greenfoot Java IDE
Mathematica
Node-RED
Python 2 (IDLE)
Python 3 (IDLE)
Scratch
Scratch 2
Sense HAT Emulator
Sonic Pi
Thonny Python IDE
Thonny (Simple Mode)
Wolfram


Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
LibreOffice Base
LibreOffice Calc
LibreOffice Draw
LibreOffice Impress
LibreOffice Math
LibreOffice Writer


Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
Chromium Web Browser
Claws Mail
Raspberry Pi Resources
The MagPi
VNC Viewer


Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
Archiver
Calculator
File Manager
Image Viewer
PDF Viewer
SD Card Copier
Task Manager
Terminal
Text Editor


Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
Add / Remove Software
Appearance Settings
Audio Device Settings
Main Menu Editor
Mouse and Keyboard Settings
Raspberry Pi Configuration


pi@raspberrypi: ~
Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
Archiver
Calculator
File Manager
Image Viewer
PDF Viewer
SD Card Copier
Task Manager
Terminal
Text Editor
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $

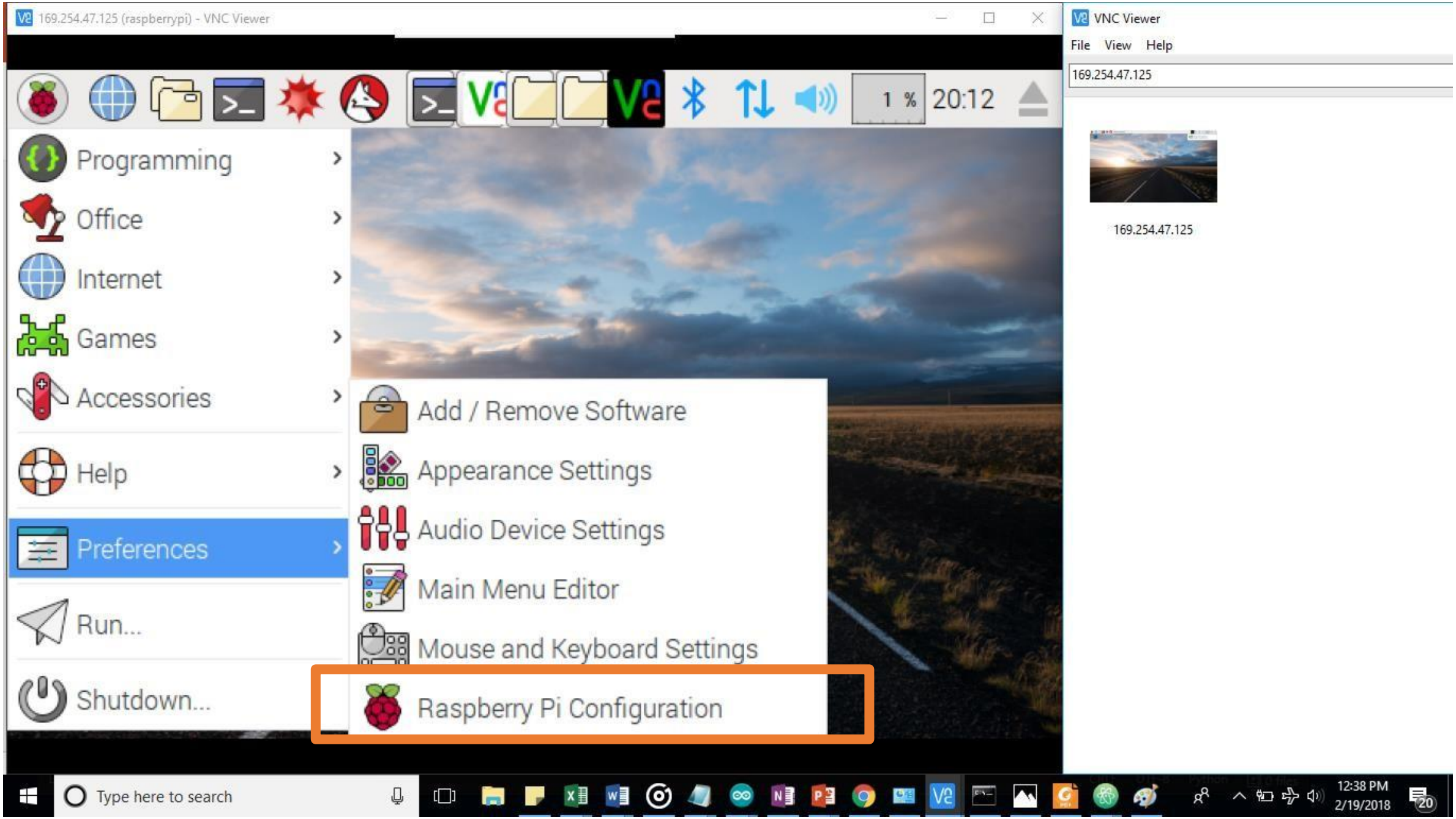
Powering up the Rpi

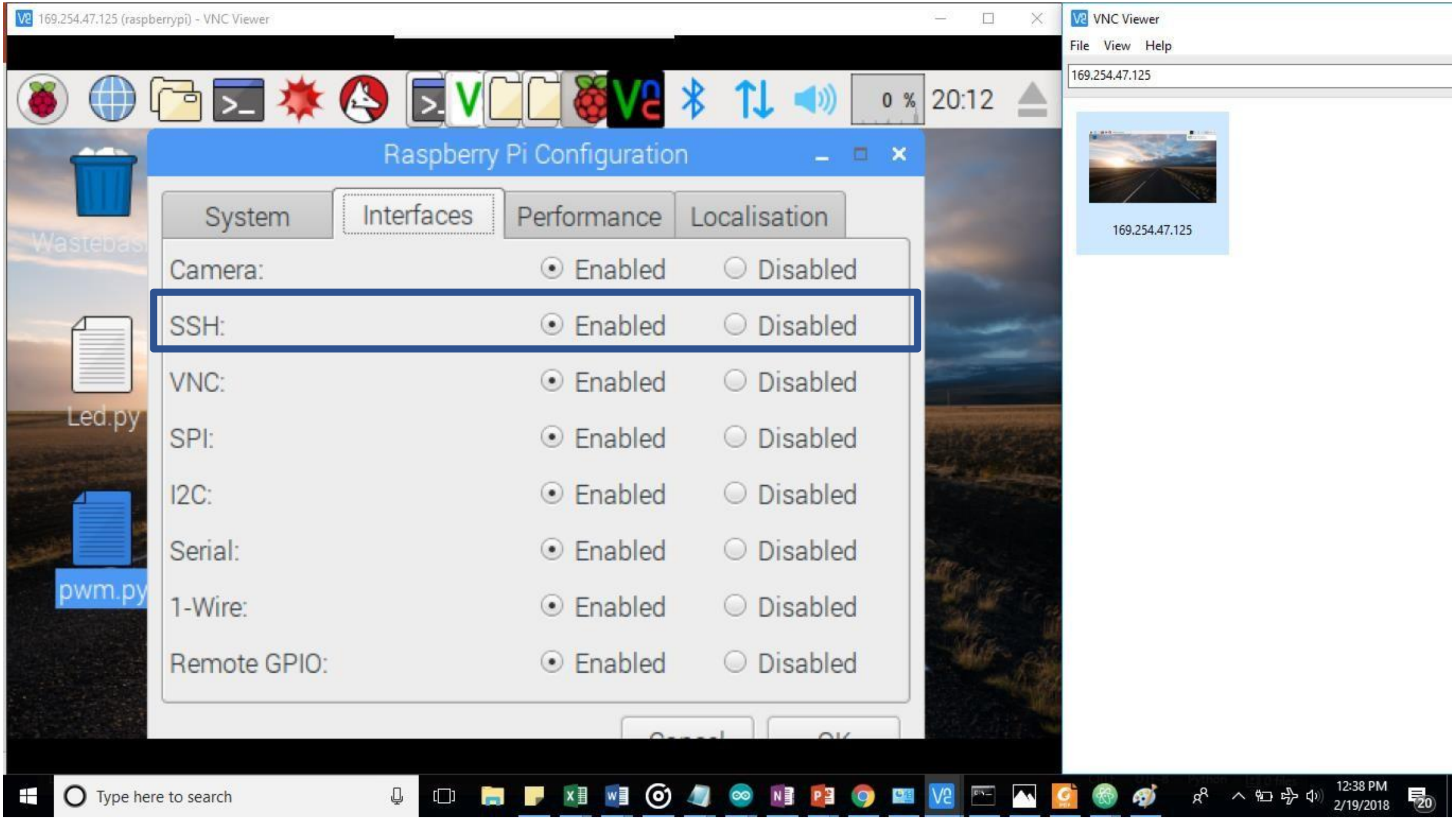
SSH(Secure Shell)

- SSH, or *Secure Shell*, is a protocol used to securely log onto remote systems. It is the most common way to access remote Linux and Unix-like servers.
- ssh remote_host
- The *remote_host* in this example is the IP address or domain name that you are trying to connectto.

IMP…(Enable SSH)

- As of the November 2016 release, Raspbian has the SSH server disabled by default. It can be enabled manually from the Raspberry pi desktop (must have a keyboard, mouse, and a display attached to the RPI)
- Launch Raspberry Pi Configuration from the Preferences menu
- Navigate to the Interfaces tab
- Select Enabled next to SSH
- Click OK

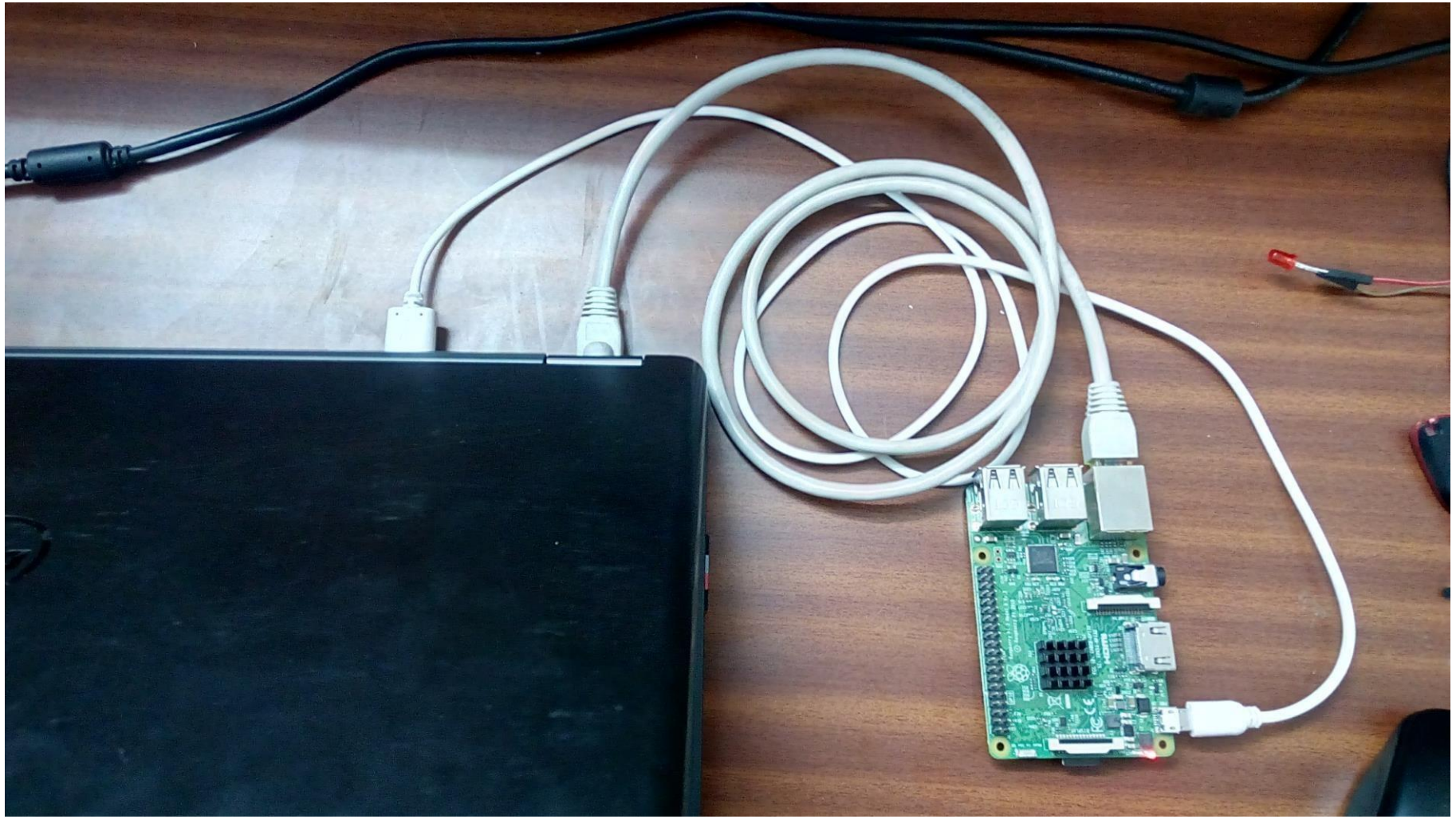

169.254.47.125 (raspberrypi) - VNC Viewer
VNC Viewer
File View Help
169.254.47.125
169.254.47.125
1 %
20:12
Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
Add / Remove Software
Appearance Settings
Audio Device Settings
Main Menu Editor
Mouse and Keyboard Settings
Raspberry Pi Configuration
Type here to search
12:38 PM
2/19/2018

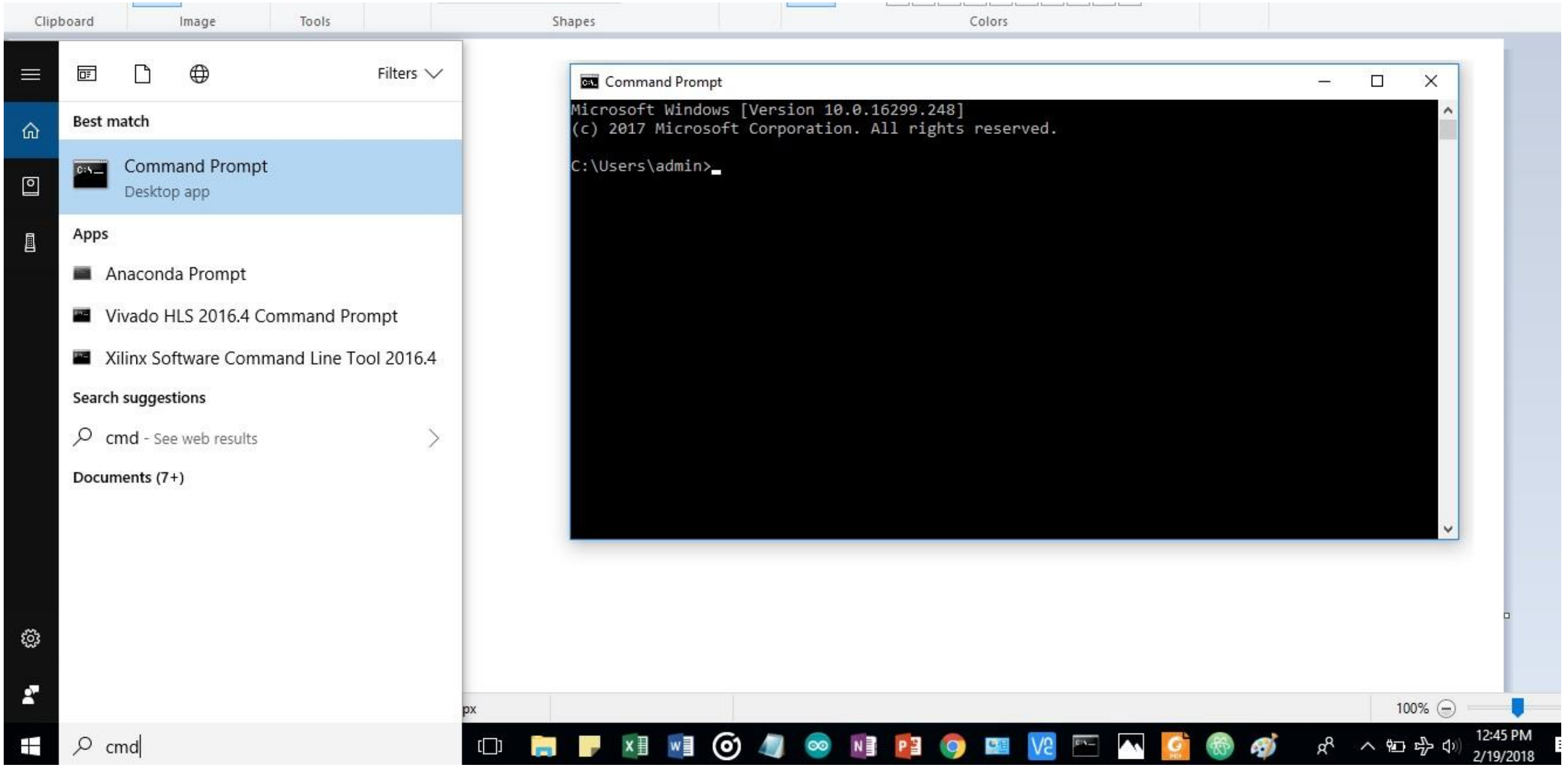

169.254.47.125 (raspberrypi) - VNC Viewer
0 %
20:12
Raspberry Pi Configuration
System
Interfaces
Performance
Localisation
Camera: Enabled Disabled
SSH: Enabled Disabled
VNC: Enabled Disabled
SPI: Enabled Disabled
I2C: Enabled Disabled
Serial: Enabled Disabled
1-Wire: Enabled Disabled
Remote GPIO: Enabled Disabled
Led.py
pwm.py
VNC Viewer
File View Help
169.254.47.125
169.254.47.125
Type here to search
12:38 PM
2/19/2018

Enable SSHon a headless Raspberry Pi (add file to SDcard on another machine)

- For headless setup, SSHcan be enabled by placing a file named ssh, without any extension, onto the boot partition of the SDcard from another computer. When the Pi boots, it looks for the ssh file. If it is found, SSHis enabled and the file is deleted. The content of the file does not matter; it could contain text, or nothing at all.
- If you have loaded Raspbian onto a blank SDcard, you will have two partitions. The first one, which is the smaller one, is the boot partition. Place the file into this one.

Connect the Rpi by Ethernet to your PC




Clipboard
Image
Tools
Shapes
Colors
Filters
Best match
Command Prompt
Desktop app
Apps
Anaconda Prompt
Vivado HLS 2016.4 Command Prompt
Xilinx Software Command Line Tool 2016.4
Search suggestions
cmd - See web results
Documents (7+)
cmd
Command Prompt
Microsoft Windows [Version 10.0.16299.248]
(c) 2017 Microsoft Corporation. All rights reserved.
C:\Users\admin>
100%
12:45 PM
2/19/2018

Finding IP address

- ***For own System***
- ***Cmd prompt:***

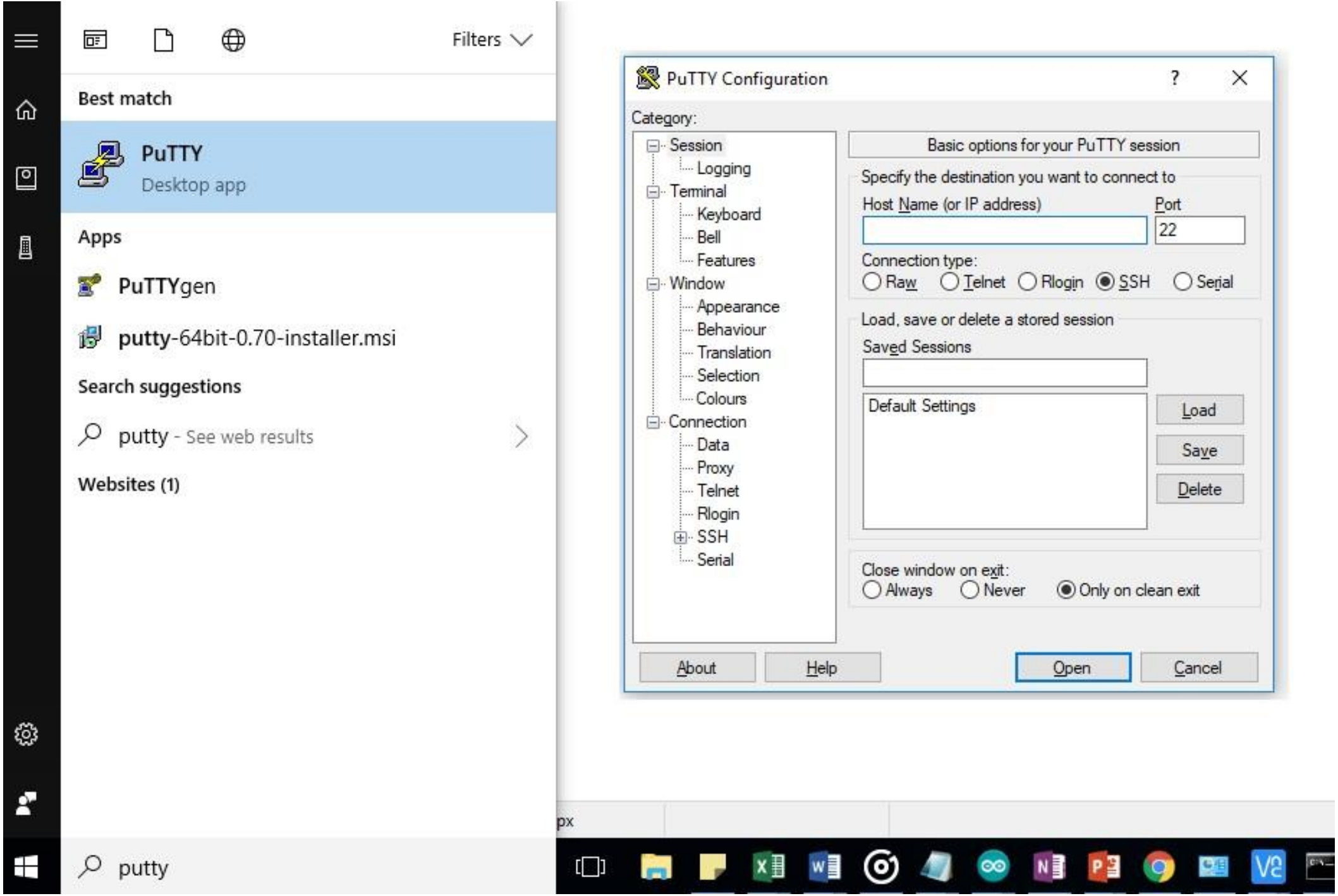
➢ *ipconfig*

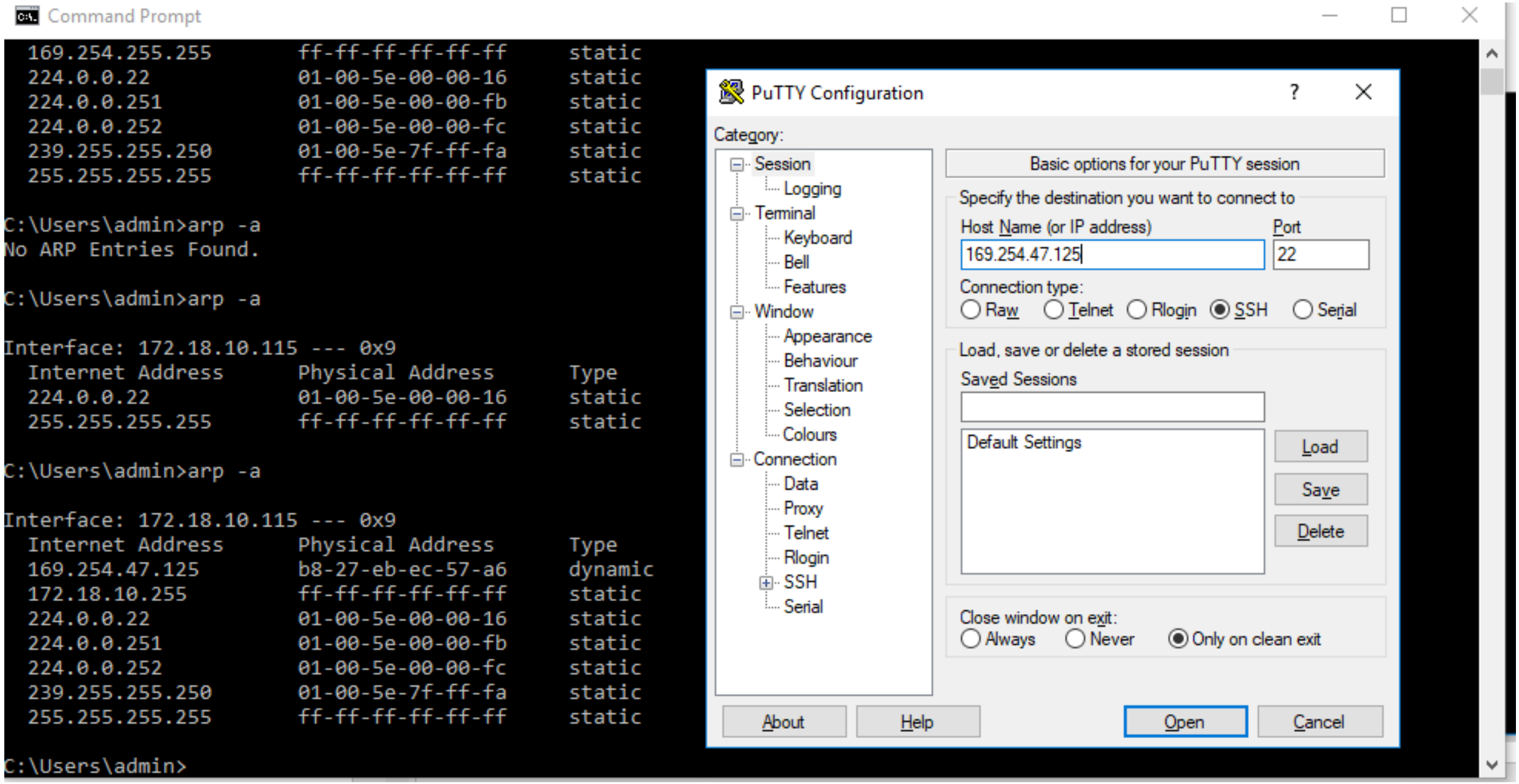
- ***FOR R-Pi***
- **Terminal**

➢ ifconfig

SSHusing Windows (putty)

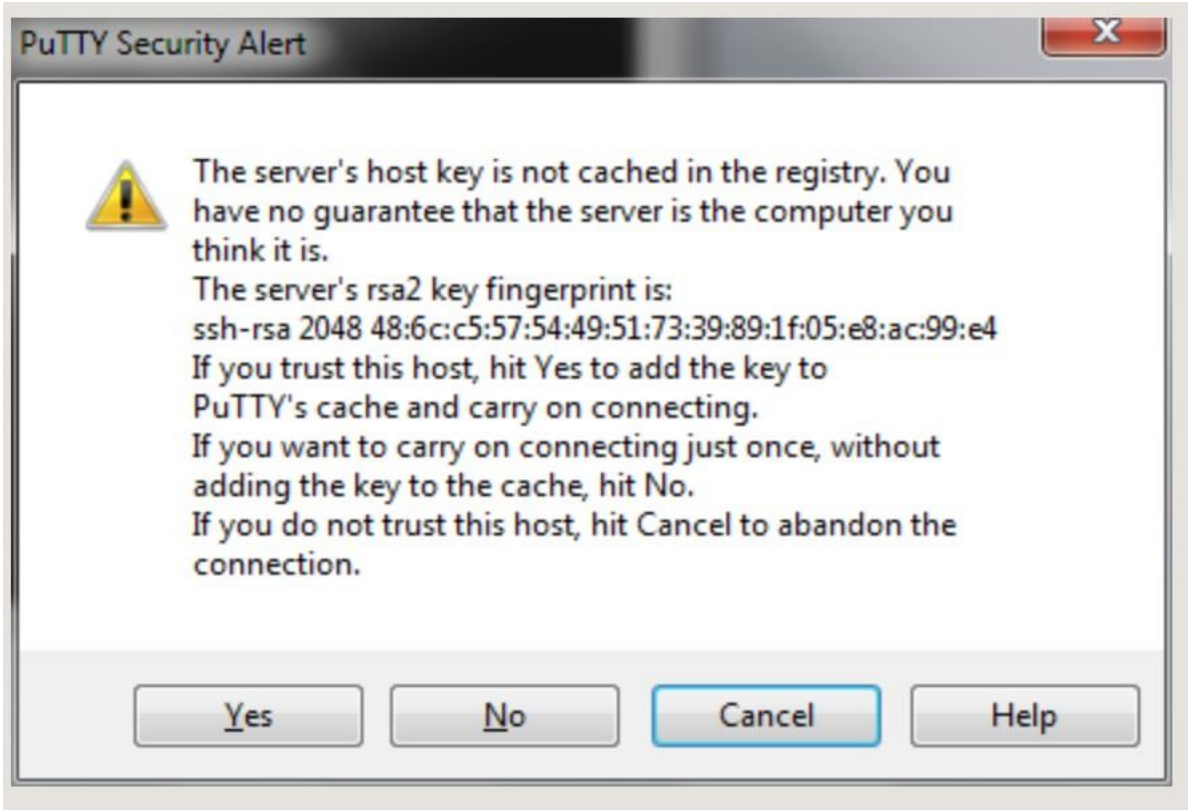
- In Windows you will need to download an SSHclient. The most commonly used client is called PuTTY. Look for putty.exe under the heading For Windows on Intel x86.
- Type the IP address of the Pi into the Host Name field and click the Openbutton. If nothing happens when you click the Open button, and you eventually see a message saying Network error: Connection timed out, it is likely that you have entered the wrong IP address for the Pi.

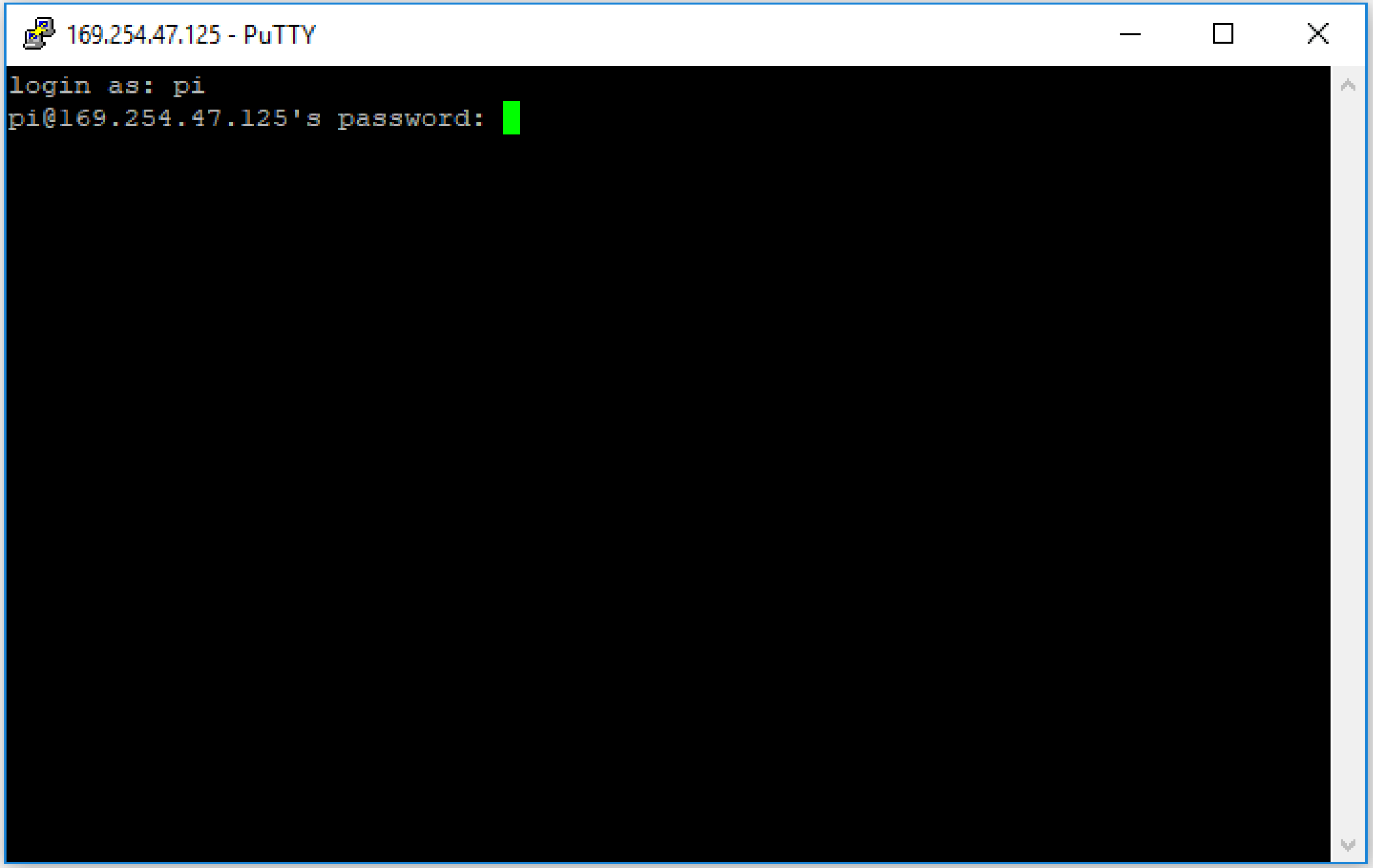

Filters
Best match
PuTTY
Desktop app
Apps
PuTTYgen
putty-64bit-0.70-installer.msi
Search suggestions
putty - See web results
Websites (1)
putty
PuTTY Configuration
Category:
Session
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
Telnet
Rlogin
SSH
Serial
Basic options for your PuTTY session
Specify the destination you want to connect to
Host Name (or IP address)
Port
22
Connection type:
Raw
Telnet
Rlogin
SSH
Serial
Load, save or delete a stored session
Saved Sessions
Default Settings
Load
Save
Delete
Close window on exit:
Always
Never
Only on clean exit
About
Help
Open
Cancel


Command Prompt
169.254.255.255 ff-ff-ff-ff-ff-ff static
224.0.0.22 01-00-5e-00-00-16 static
224.0.0.251 01-00-5e-00-00-fb static
224.0.0.252 01-00-5e-00-00-fc static
239.255.255.250 01-00-5e-7f-ff-fa static
255.255.255.255 ff-ff-ff-ff-ff-ff static
C:\Users\admin>arp -a
No ARP Entries Found.
C:\Users\admin>arp -a
Interface: 172.18.10.115 --- 0x9
Internet Address Physical Address Type
224.0.0.22 01-00-5e-00-00-16 static
255.255.255.255 ff-ff-ff-ff-ff-ff static
C:\Users\admin>arp -a
Interface: 172.18.10.115 --- 0x9
Internet Address Physical Address Type
169.254.47.125 b8-27-eb-ec-57-a6 dynamic
172.18.10.255 ff-ff-ff-ff-ff-ff static
224.0.0.22 01-00-5e-00-00-16 static
224.0.0.251 01-00-5e-00-00-fb static
224.0.0.252 01-00-5e-00-00-fc static
239.255.255.250 01-00-5e-7f-ff-fa static
255.255.255.255 ff-ff-ff-ff-ff-ff static
C:\Users\admin>
PuTTY Configuration
Category:
Session
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
Telnet
Rlogin
SSH
Serial
Basic options for your PuTTY session
Specify the destination you want to connect to
Host Name (or IP address)
Port
169.254.47.125
22
Connection type:
Raw
Telnet
Rlogin
SSH
Serial
Load, save or delete a stored session
Saved Sessions
Default Settings
Load
Save
Delete
Close window on exit:
Always
Never
Only on clean exit
About
Help
Open
Cancel

CONNECT

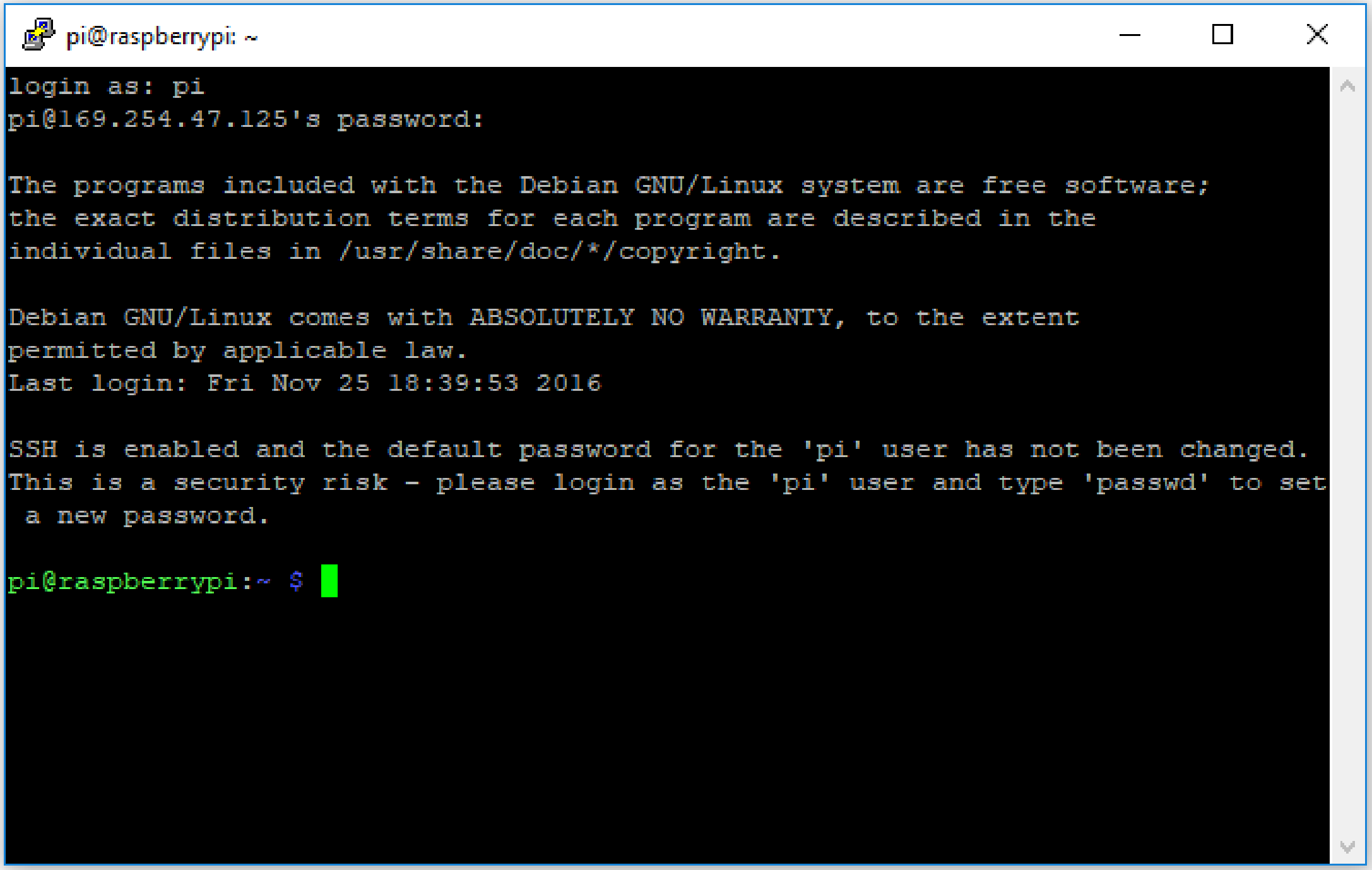
- When the connection works you will see the security warning shown below. You can safely ignore it, and click the 'Yes' button. You will only see this warning the first time PuTTY connects to a Raspberry Pi that it has not seen before.




169.254.47.125 - PuTTY
login as: pi
pi@169.254.47.125's password:

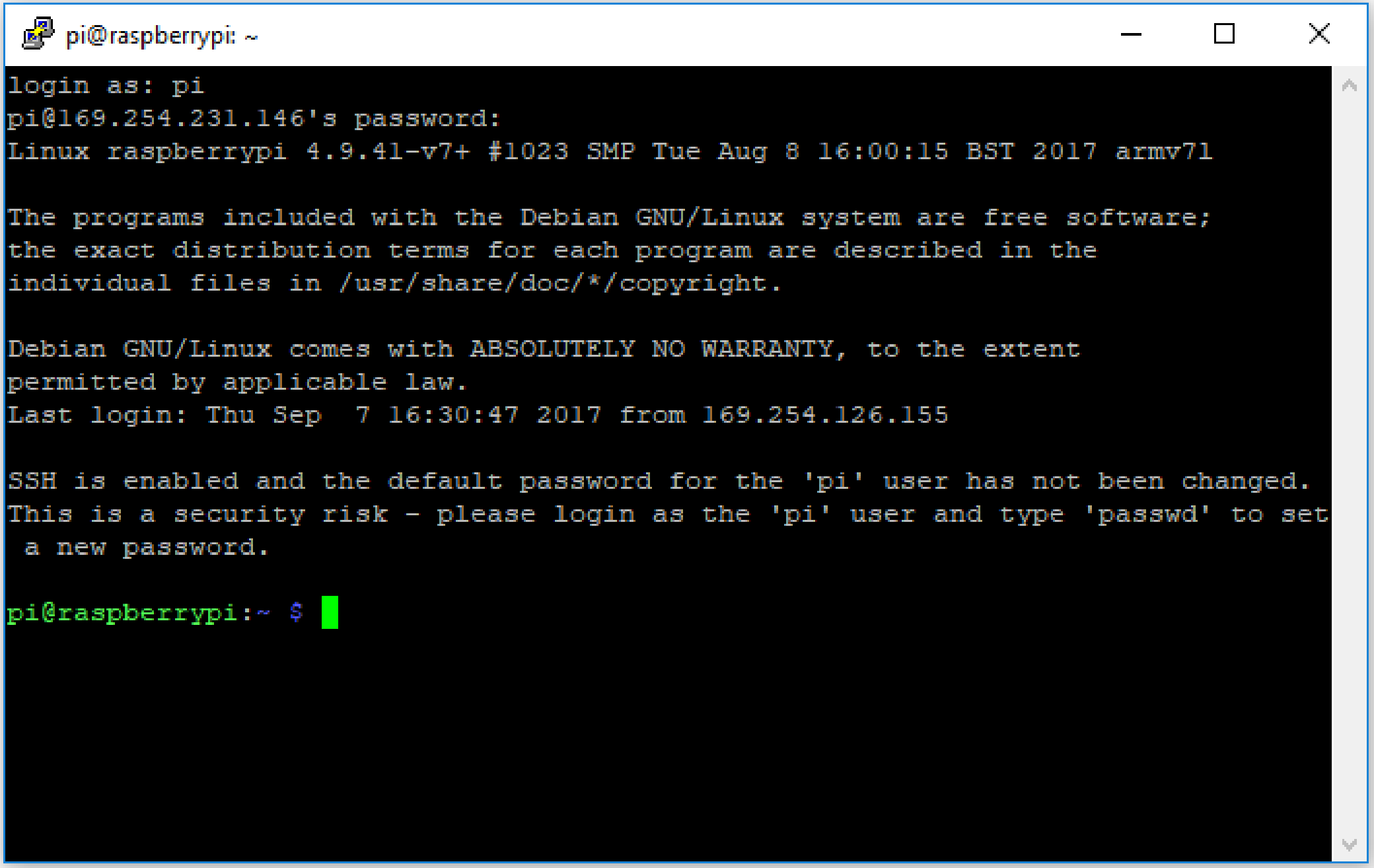
Contd…

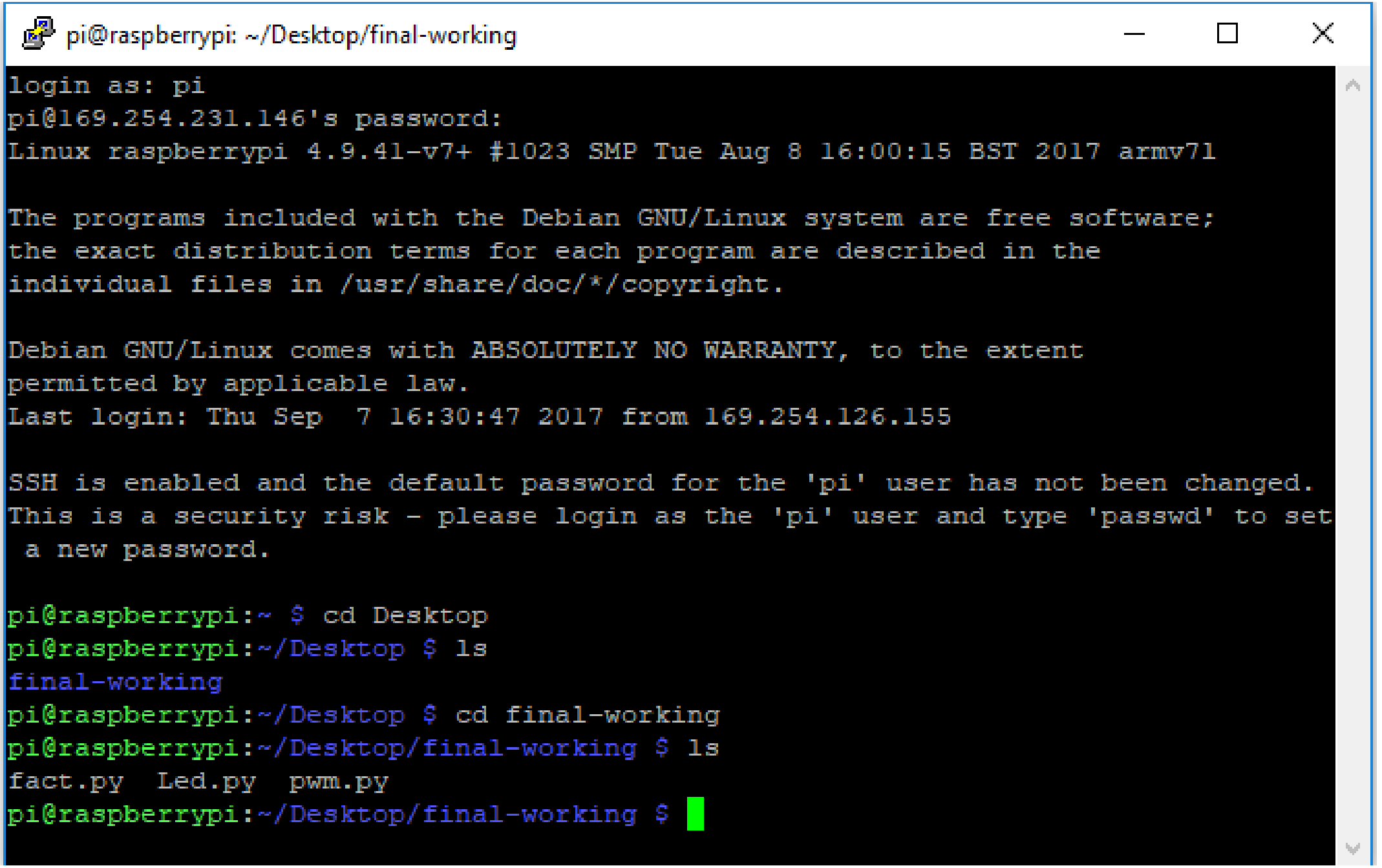
- You will now see the usual login prompt. Log in with the same username and password you would use on the Pi itself. The default login for Raspbian is
- **Login**: pi
- **Password**: raspberry.
- You should now have the Raspberry Pi prompt which will be identical to the one found on the Raspberry Pi itself.
- pi@raspberrypi ~ $
- Detailed instruction to set up putty.
- https://www.raspberrypi.org/documentation/remote-access/ssh/windows.md

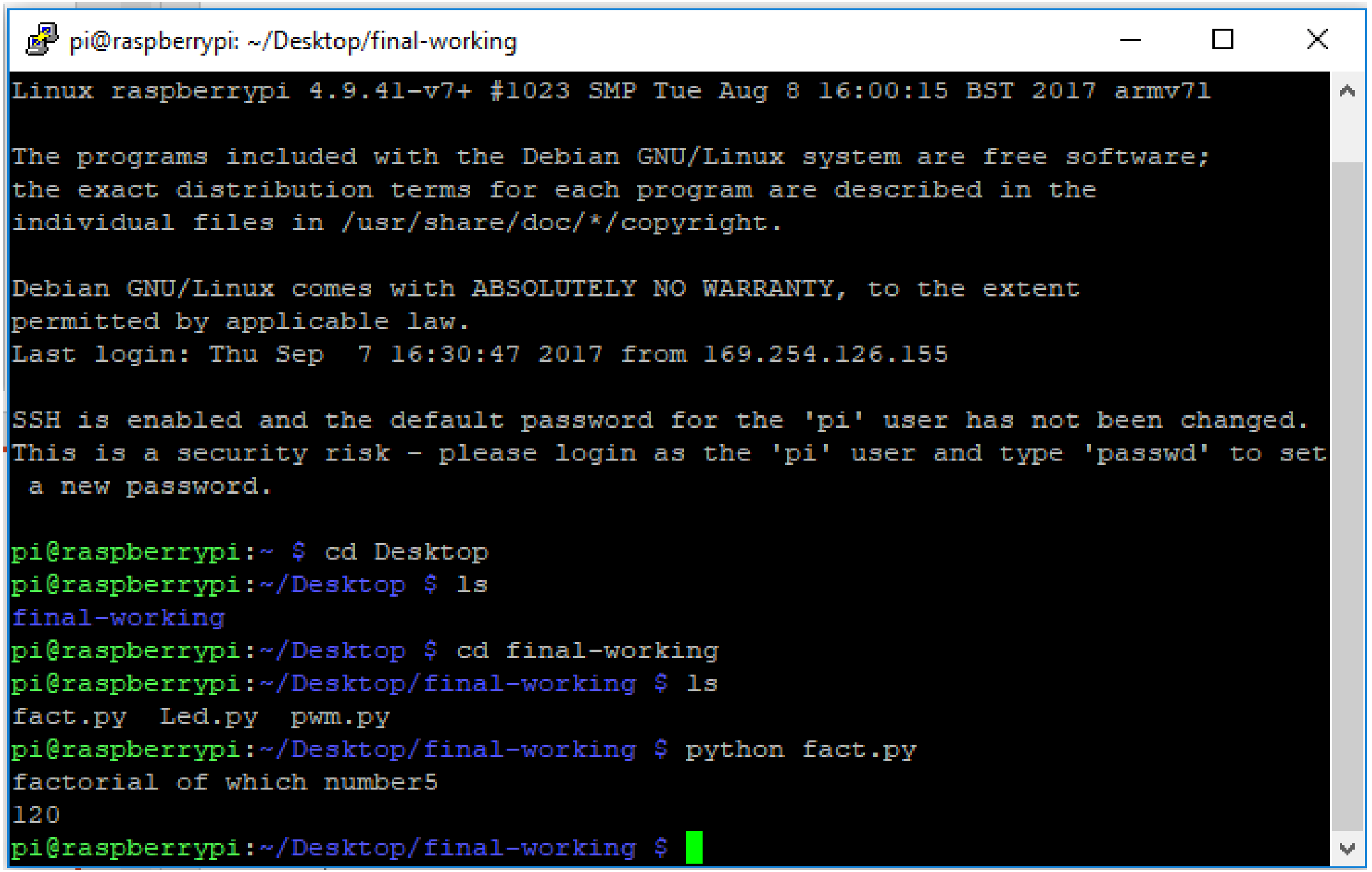

pi@raspberrypi: ~
login as: pi
pi@169.254.47.125's password:
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Nov 25 18:39:53 2016
SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.
pi@raspberrypi:~ $

Commonly Used Linux Commands

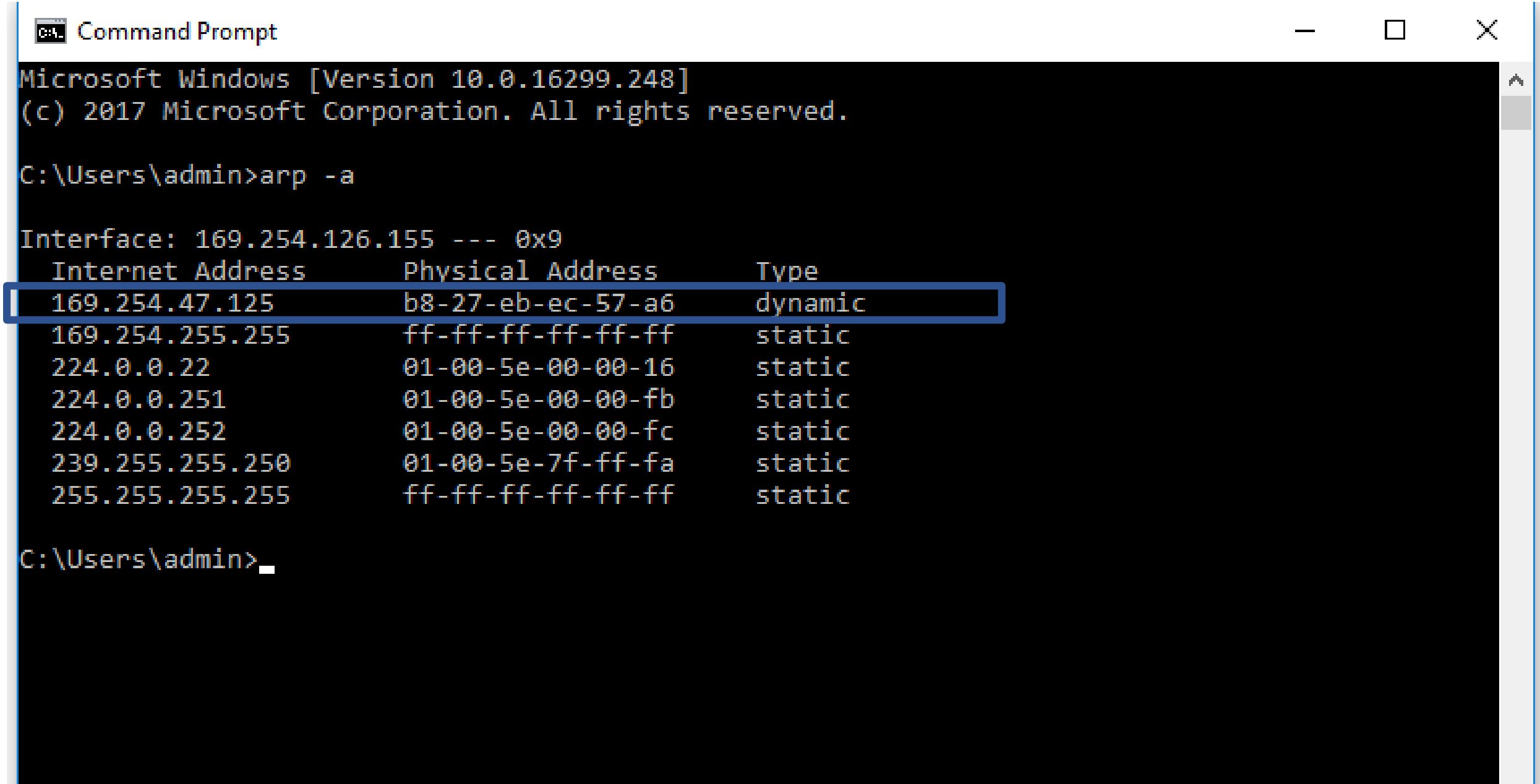
Command	Description
man	display details about an instruction
ls	list contents of the current directory
ls -l	detailed listing of directory contents, shows permissions, owner, etc.
ls -a	list all files (including hidden files)
ls -la	detailed listing of all files (note that options can be combined)
cd	change directory
cd ../	backup one level from the current directory
pwd	print current working directory
touch	create an empty file
mkdir	make a new directory
rm	remove files and directories
rm -rf	recursively remove all files and directories under the specified directory
cat	list file contents
less	list file contents – one screen at a time
tail	list the end of the file (default – displays last 10 lines)
tail -n	list the last n lines of a file
cp	copy file
mv	rename a file
echo	echo values to the screen example: echo $PATH – prints the value of the PATH variable
grep	command line text search utiltity example: grep blue colors.txt – list all lines with the word blue from the colorlist.txt file
ps	list currently running processes


pi@raspberrypi: ~
login as: pi
pi@169.254.231.146's password:
Linux raspberrypi 4.9.41-v7+ #1023 SMP Tue Aug 8 16:00:15 BST 2017 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: Thu Sep 7 16:30:47 2017 from 169.254.126.155
SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.
pi@raspberrypi:~ $


pi@raspberrypi: ~/Desktop/final-working
login as: pi
pi@169.254.231.146's password:
Linux raspberrypi 4.9.41-v7+ #1023 SMP Tue Aug 8 16:00:15 BST 2017 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: Thu Sep 7 16:30:47 2017 from 169.254.126.155
SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.
pi@raspberrypi:~ $ cd Desktop
pi@raspberrypi:~/Desktop $ ls
final-working
pi@raspberrypi:~/Desktop $ cd final-working
pi@raspberrypi:~/Desktop/final-working $ ls
fact.py Led.py pwm.py
pi@raspberrypi:~/Desktop/final-working $


pi@raspberrypi: ~/Desktop/final-working
Linux raspberrypi 4.9.41-v7+ #1023 SMP Tue Aug 8 16:00:15 BST 2017 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: Thu Sep 7 16:30:47 2017 from 169.254.126.155
SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.
pi@raspberrypi:~ $ cd Desktop
pi@raspberrypi:~/Desktop $ ls
final-working
pi@raspberrypi:~/Desktop $ cd final-working
pi@raspberrypi:~/Desktop/final-working $ ls
fact.py Led.py pwm.py
pi@raspberrypi:~/Desktop/final-working $ python fact.py
factorial of which number5
120
pi@raspberrypi:~/Desktop/final-working $

Type arp –a on your laptop cmd prompt



SSHusing MAC

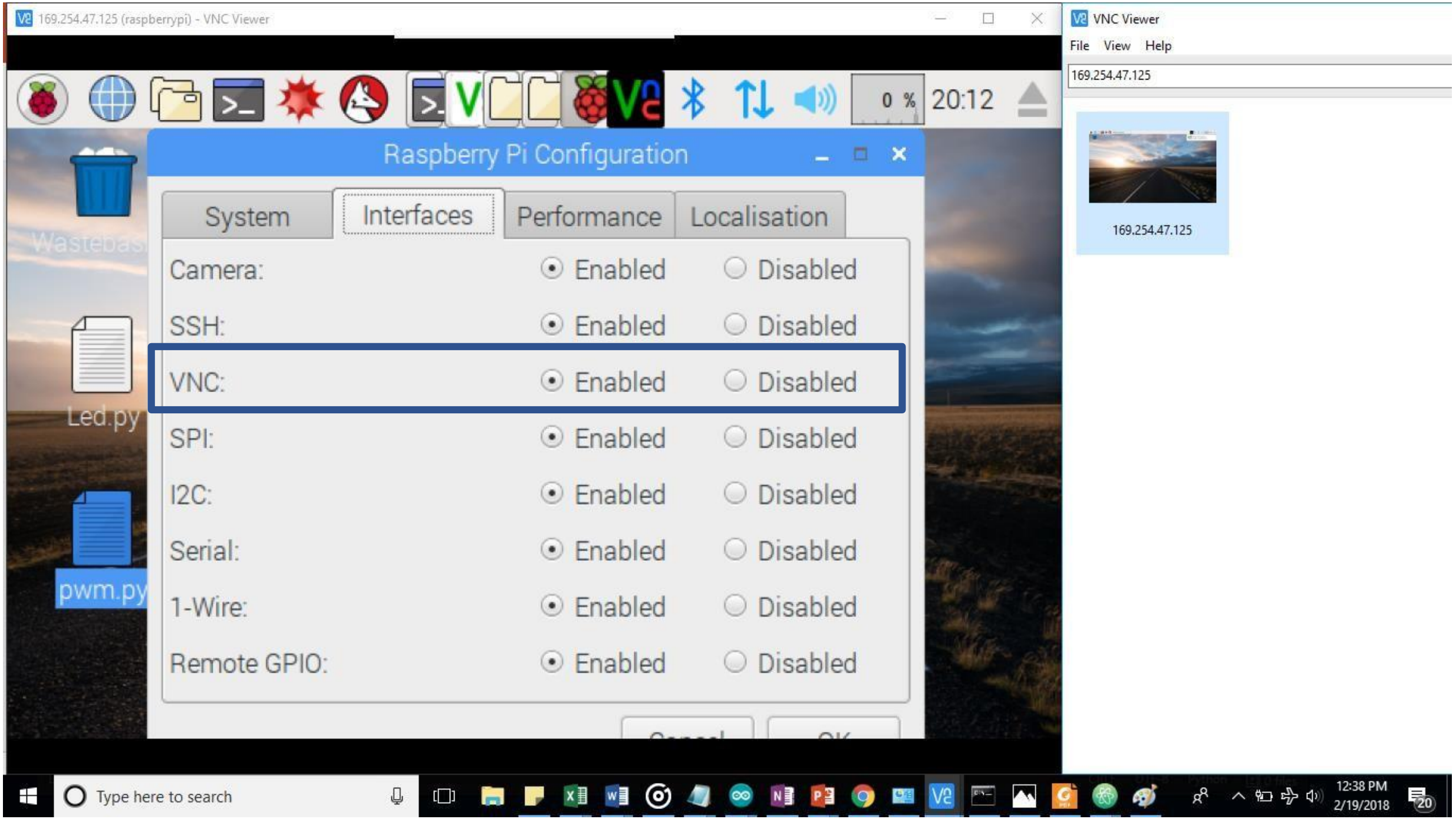
- You can use SSH to connect to your Raspberry Pi from a Linux computer, a Mac, or another Raspberry Pi, without installing additional software
- **“ssh pi@<IP>”**
- Where IP refers to the IP address of raspberry pi found earlier.

VNC remote access to Rpi

- VNC (Virtual Network Computing) is a great technology included with Raspbian. With this tech, you can remotely control your Raspberry Pi from another computer, such as a PC or Mac, or even another Raspberry Pi board.
- Sometimes it's not convenient to work directly on a Raspberry Pi. This can be because it's not easy to access, or because your keyboard and monitor are being used on your main computer.
- Using remote networking, you can open the Raspbian desktop interface from your Raspberry Pi inside a window on your computer.

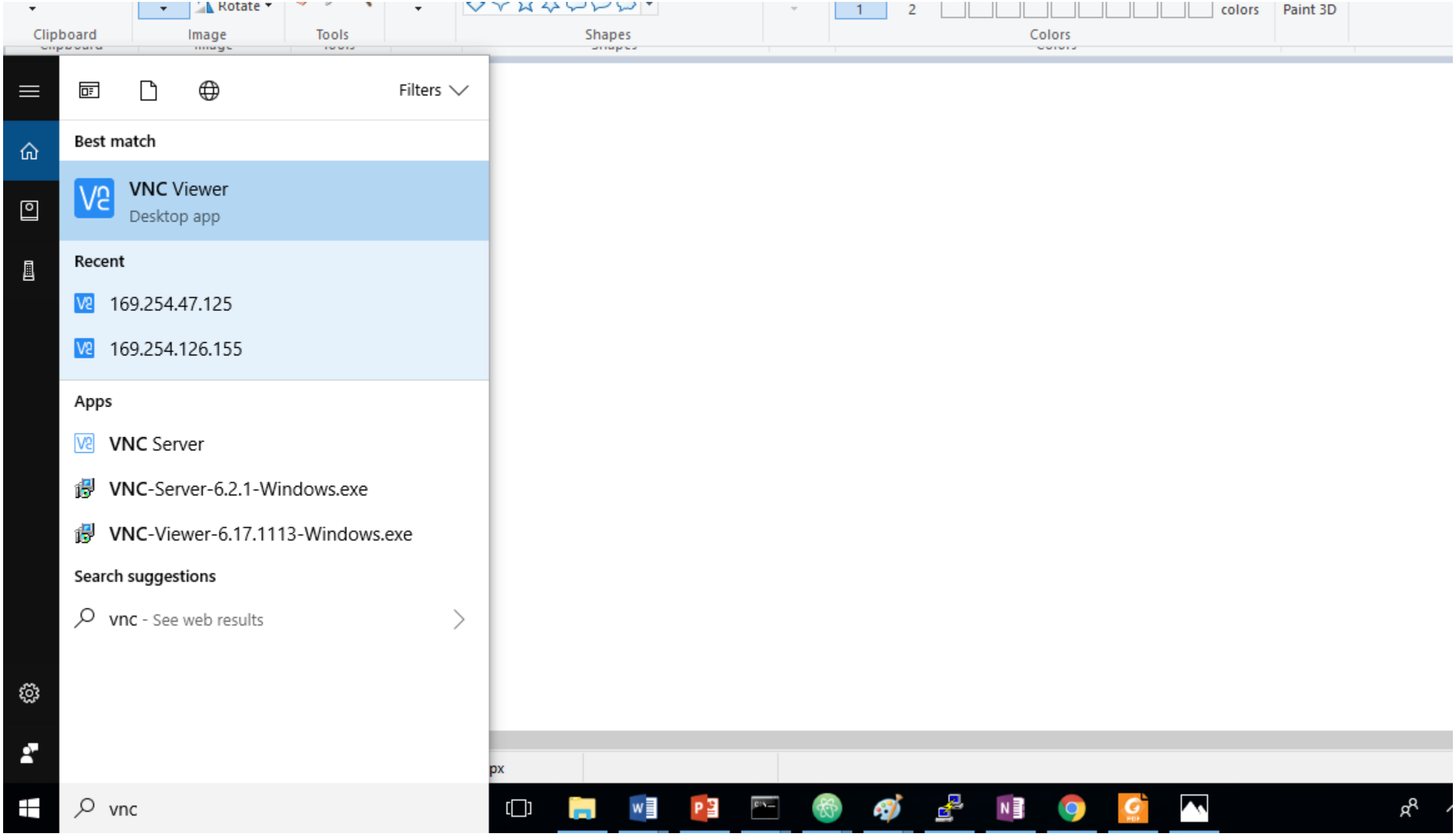
Contd…

- By default, VNC Server from RealVNC gives you direct control over your Raspberry Pi, just as though you were sitting in front of it.
- However, you will need to enable it yourself. From then on, Server will be loaded every time you switch on your Raspberry Pi.(Just like we enabled SSH)
- raspi-config can be used in the terminal:
- Enter ”sudo raspi-config” in a terminal window (This is once we have already SSHed into the Rpi)
- Select Interfacing Options
- Navigate to and select VNC
- Choose Yes
- Select Ok
- Choose Finish


169.254.47.125 (raspberrypi) - VNC Viewer
0 %
20:12
Raspberry Pi Configuration
System
Interfaces
Performance
Localisation
Camera: Enabled Disabled
SSH: Enabled Disabled
VNC: Enabled Disabled
SPI: Enabled Disabled
I2C: Enabled Disabled
Serial: Enabled Disabled
1-Wire: Enabled Disabled
Remote GPIO: Enabled Disabled
Led.py
pwm.py
VNC Viewer
File View Help
169.254.47.125
169.254.47.125
Type here to search
12:38 PM
2/19/2018

Contd…

- Download and install Viewer on your computer from **RealVNC**. Open the app and enter the IP Address of the Rpi into the VNC Server field. Ensure Encryption is set to “Let VNC Server choose” and click Connect. OK. The first time you connect it will display a “VNC Server not recognized” alert. Click Continue.
- Enter Pi into the Username field, and “raspberry” (or your Raspbian password) in the Password field. The Raspbian PIXEL interface will appear inside a window on your computer. You can click on the Menu, open programs and run Terminal commands on your Raspberry Pi. If you have your Raspberry Connected to a monitor, you’ll see it move as you remote control it.

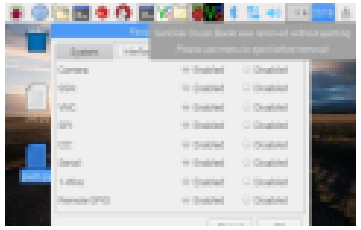

Clipboard
Image
Rotate
Tools
Shapes
1
2
colors
Paint 3D
Colors
Filters
Best match
VNC Viewer
Desktop app
Recent
169.254.47.125
169.254.126.155
Apps
VNC Server
VNC-Server-6.2.1-Windows.exe
VNC-Viewer-6.17.1113-Windows.exe
Search suggestions
vnc - See web results
vnc

VNC Viewer

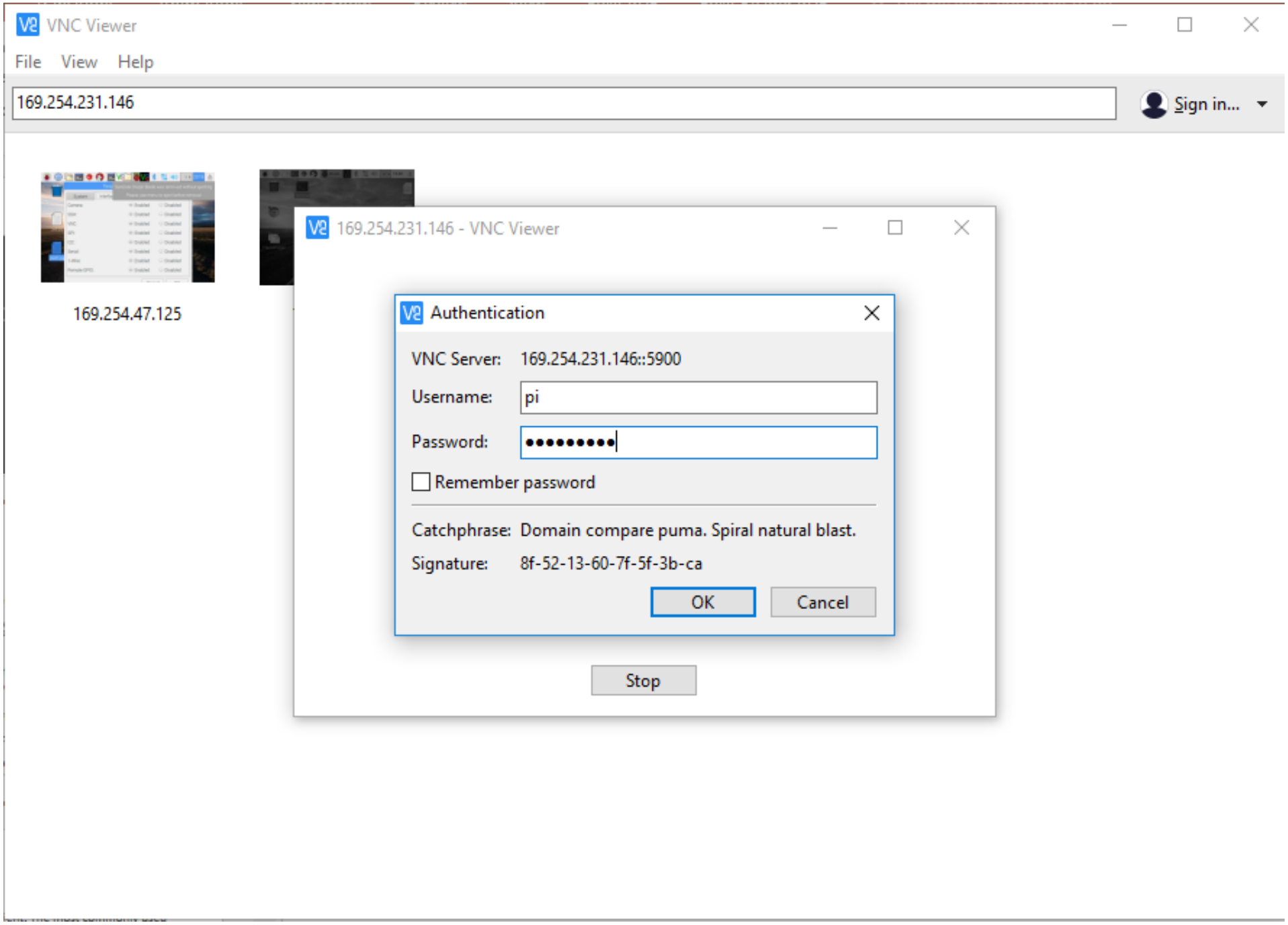
File View Help

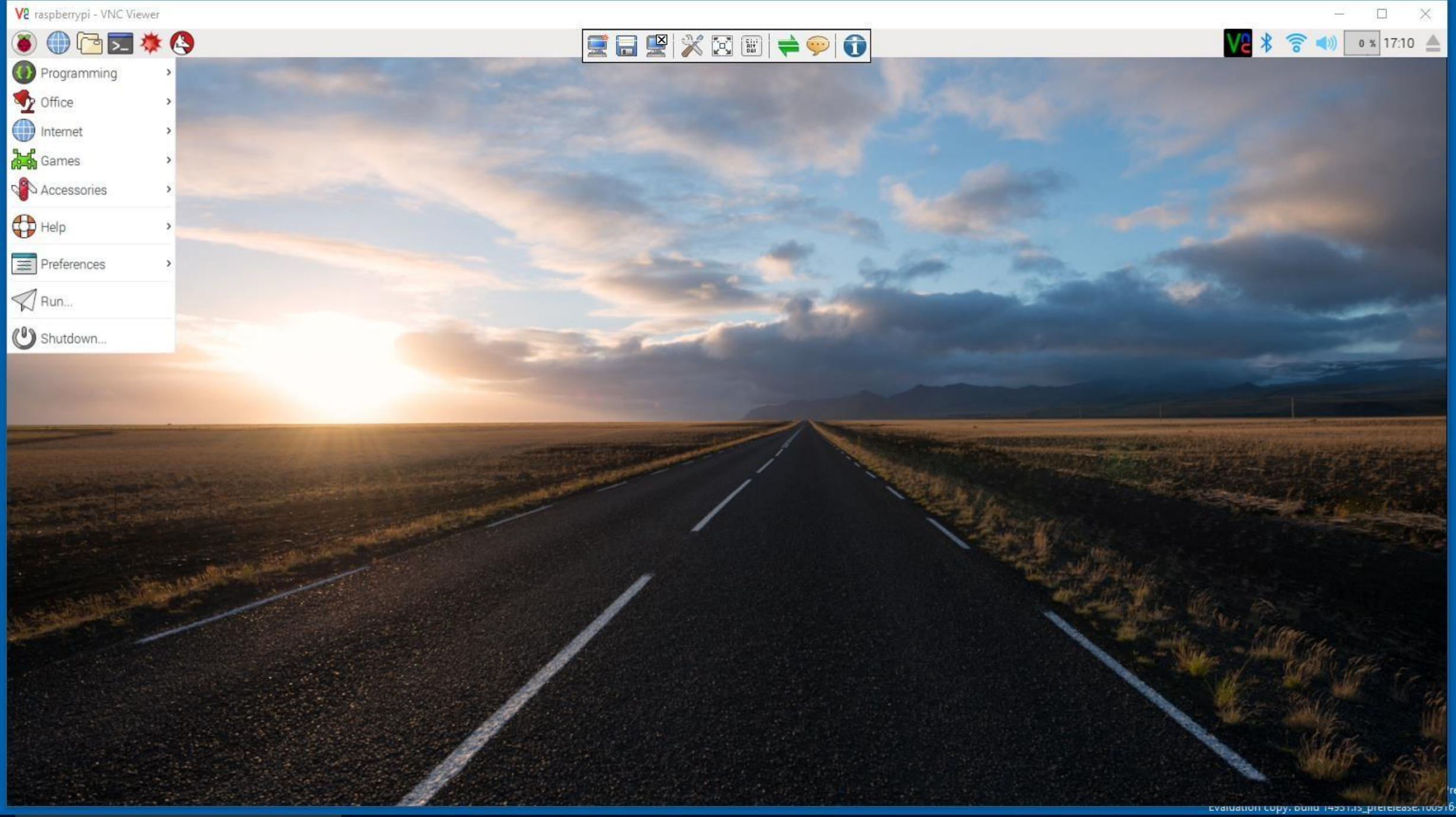
169.254.47.125

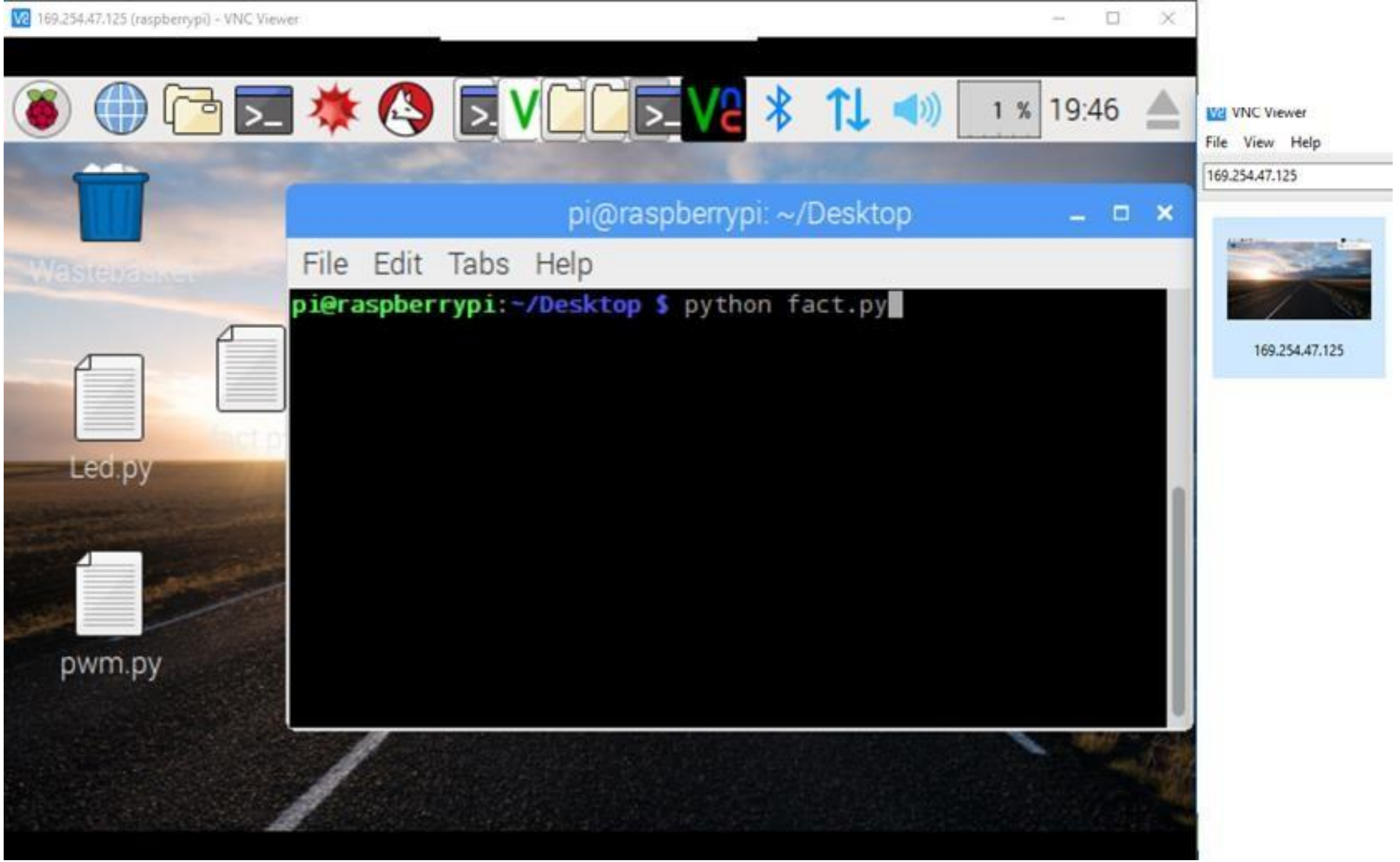
Sign in...

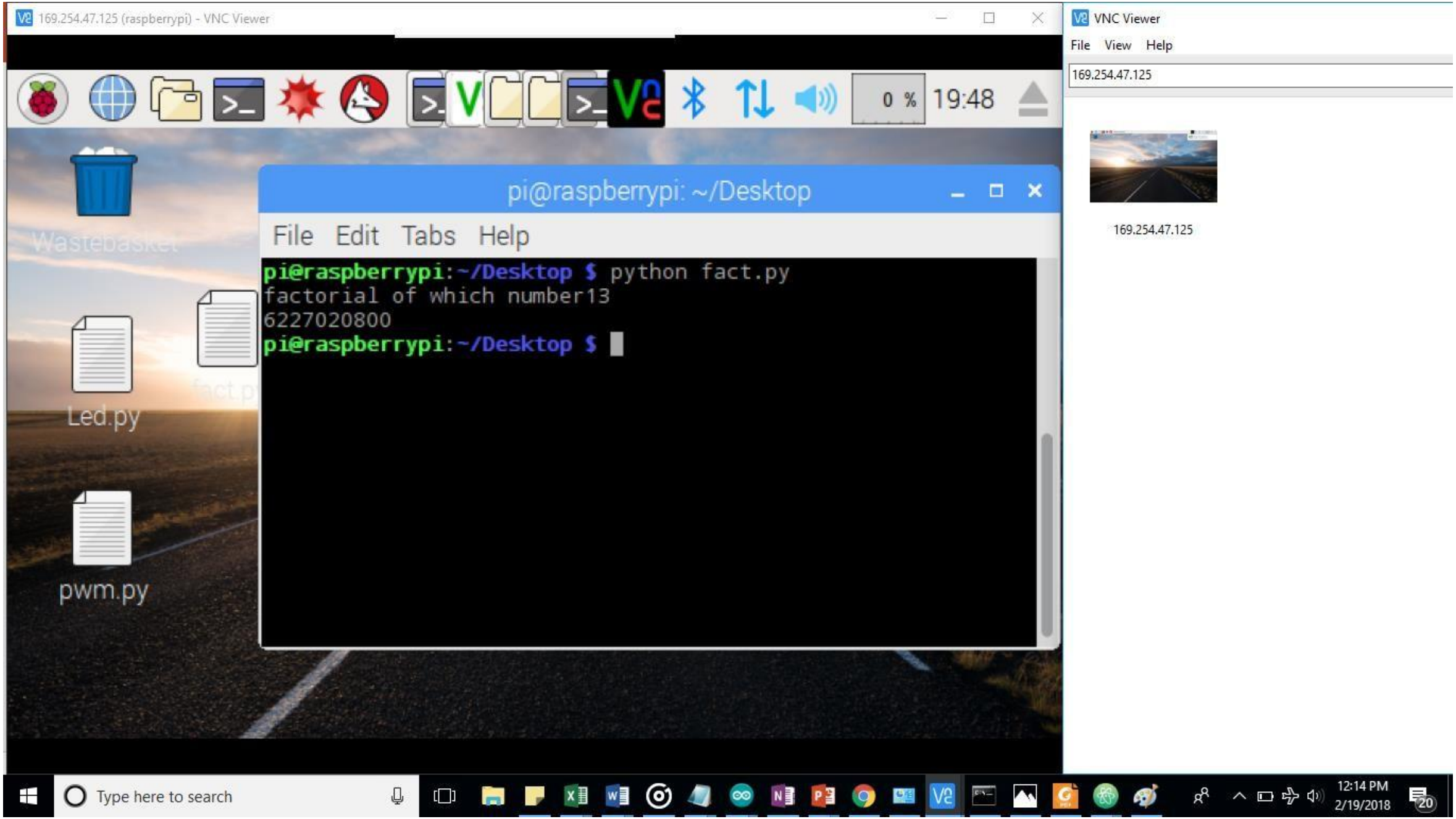


169.254.47.125


VNC Viewer
File View Help
169.254.231.146
Sign in...
169.254.47.125
169.254.231.146 - VNC Viewer
Authentication
VNC Server: 169.254.231.146::5900
Username: pi
Password:
Remember password
Catchphrase: Domain compare puma. Spiral natural blast.
Signature: 8f-52-13-60-7f-5f-3b-ca
OK
Cancel
Stop


raspberrypi - VNC Viewer
Programming
Office
Internet
Games
Accessories
Help
Preferences
Run...
Shutdown...
0 %
17:10
Ask me anything
16:10
31/10/2016


169.254.47.125 (raspberrypi) - VNC Viewer
1 %
19:46
pi@raspberrypi: ~/Desktop
File Edit Tabs Help
pi@raspberrypi:~/Desktop $ python fact.py
Led.py
pwm.py
VNC Viewer
File View Help
169.254.47.125
169.254.47.125


169.254.47.125 (raspberrypi) - VNC Viewer
0 %
19:48
Wastebasket
Led.py
pwm.py
pi@raspberrypi: ~/Desktop
File Edit Tabs Help
pi@raspberrypi:~/Desktop $ python fact.py
factorial of which number13
6227020800
pi@raspberrypi:~/Desktop $
VNC Viewer
File View Help
169.254.47.125
169.254.47.125
Type here to search
12:14 PM
2/19/2018

LED Blink
("Hello World" of Physical computing)

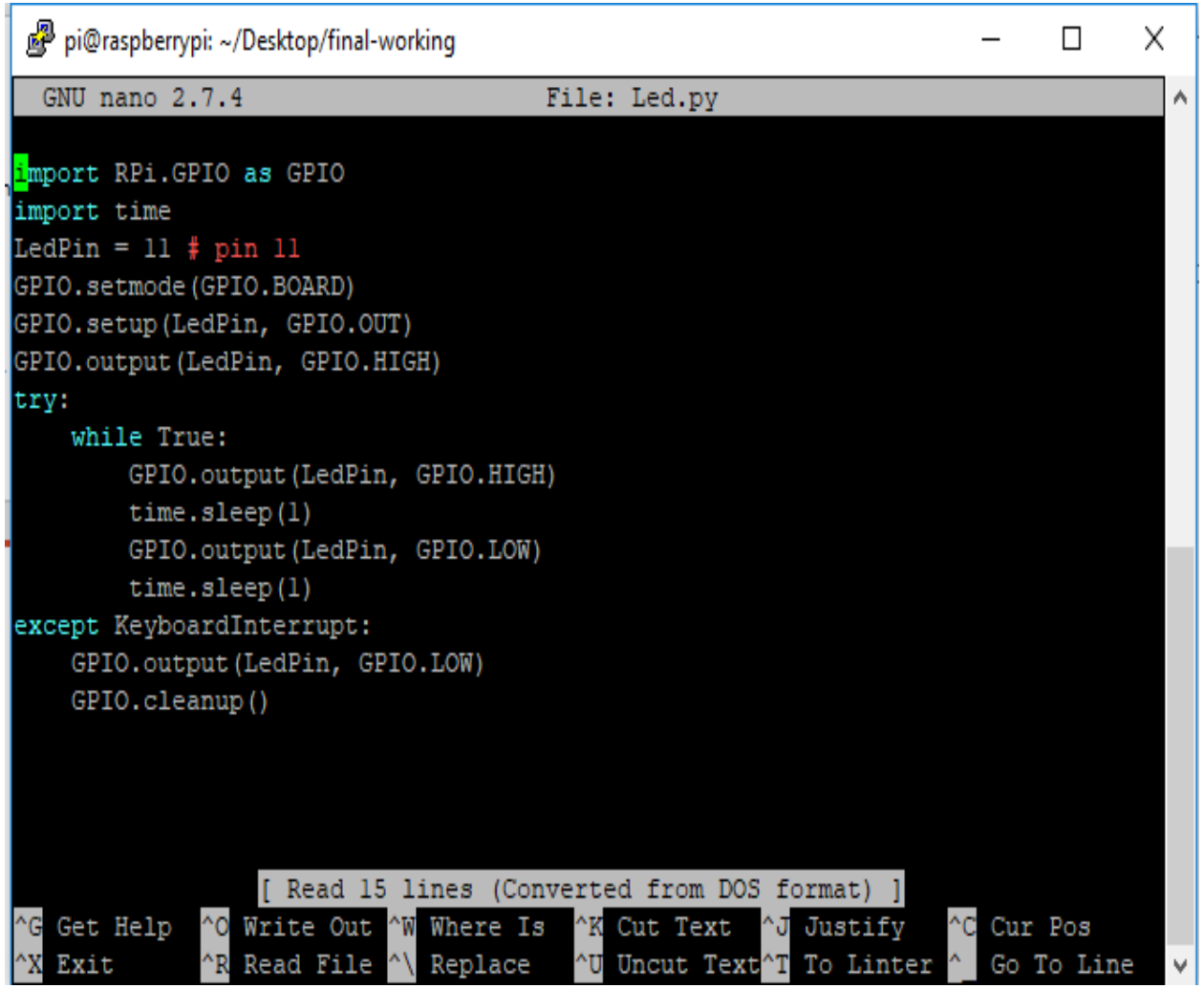
- The simplest thing you can do with an Raspberry Pi to see an actual physical output is to blink an LED! It's more like an electronics version of the classical hello world program that we write when learning a new software language.
- Raspberry Pi has 8 General Purpose Input/Output pins(4,17,18,21,22,23,24,25) which can be configured as input/output and turned on/off via software.
- RPi.GPIO Python library is used to control the LED from Raspberry Pi.

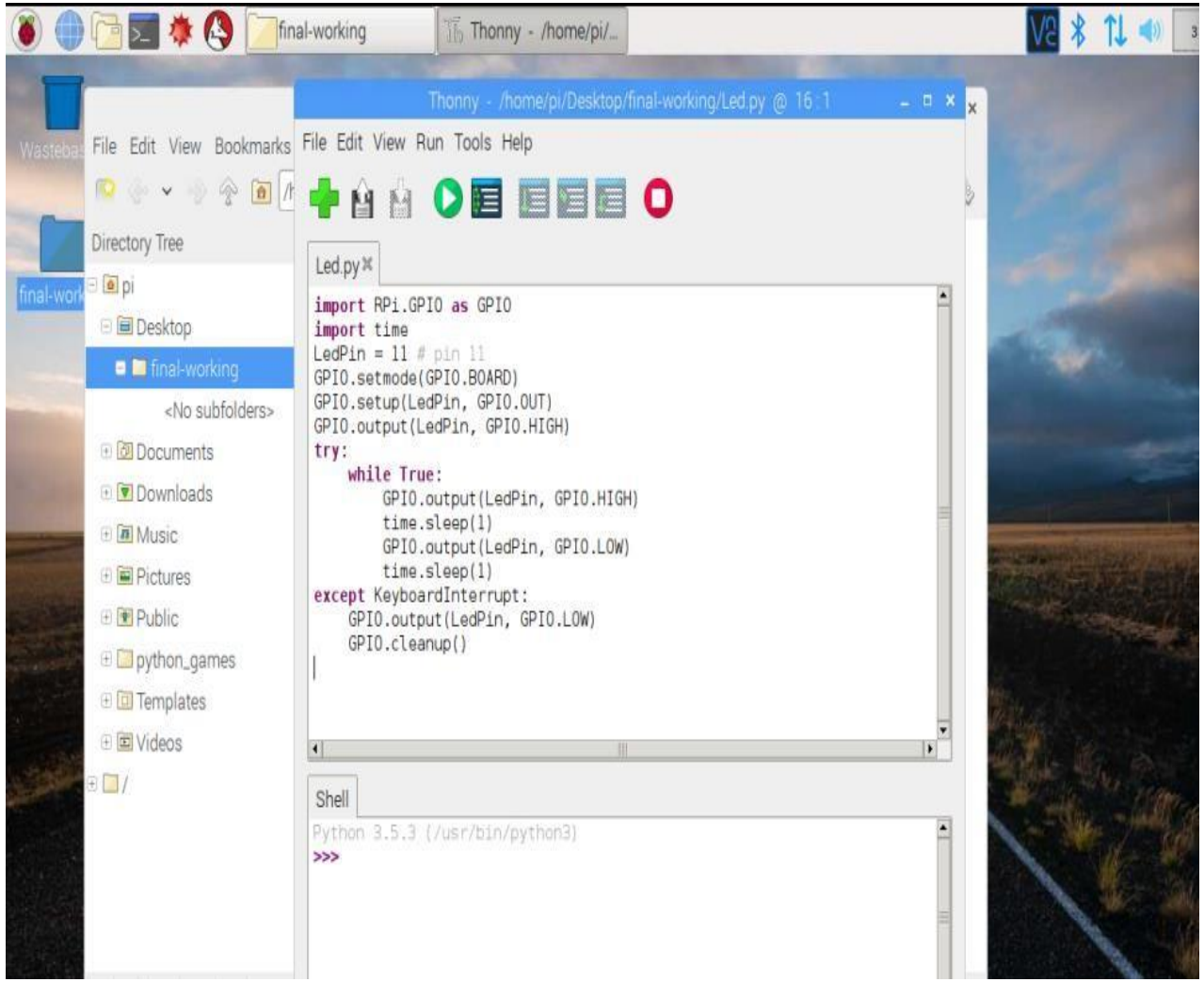
Only SSH

>> sudo nano Led.py

With VNC

See in the Rpi desktop





Python (RPi.GPIO) API

- In order to use RPi.GPIO throughout the rest of your Python script, you need to put this statement at the **top of your file**:

```
import RPi.GPIO as GPIO
```

- This statement "includes" the RPi.GPIO module, and goes a step further by providing a local name – GPIO – which we'll call to reference the module from here on.

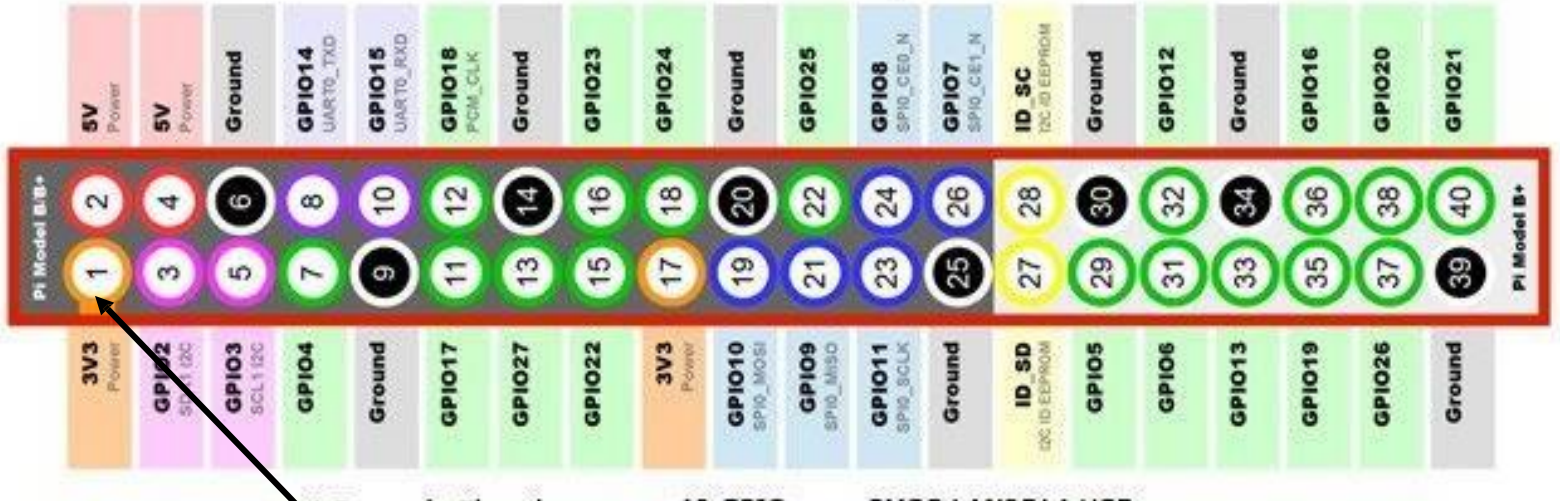
Pin Numbering Declaration

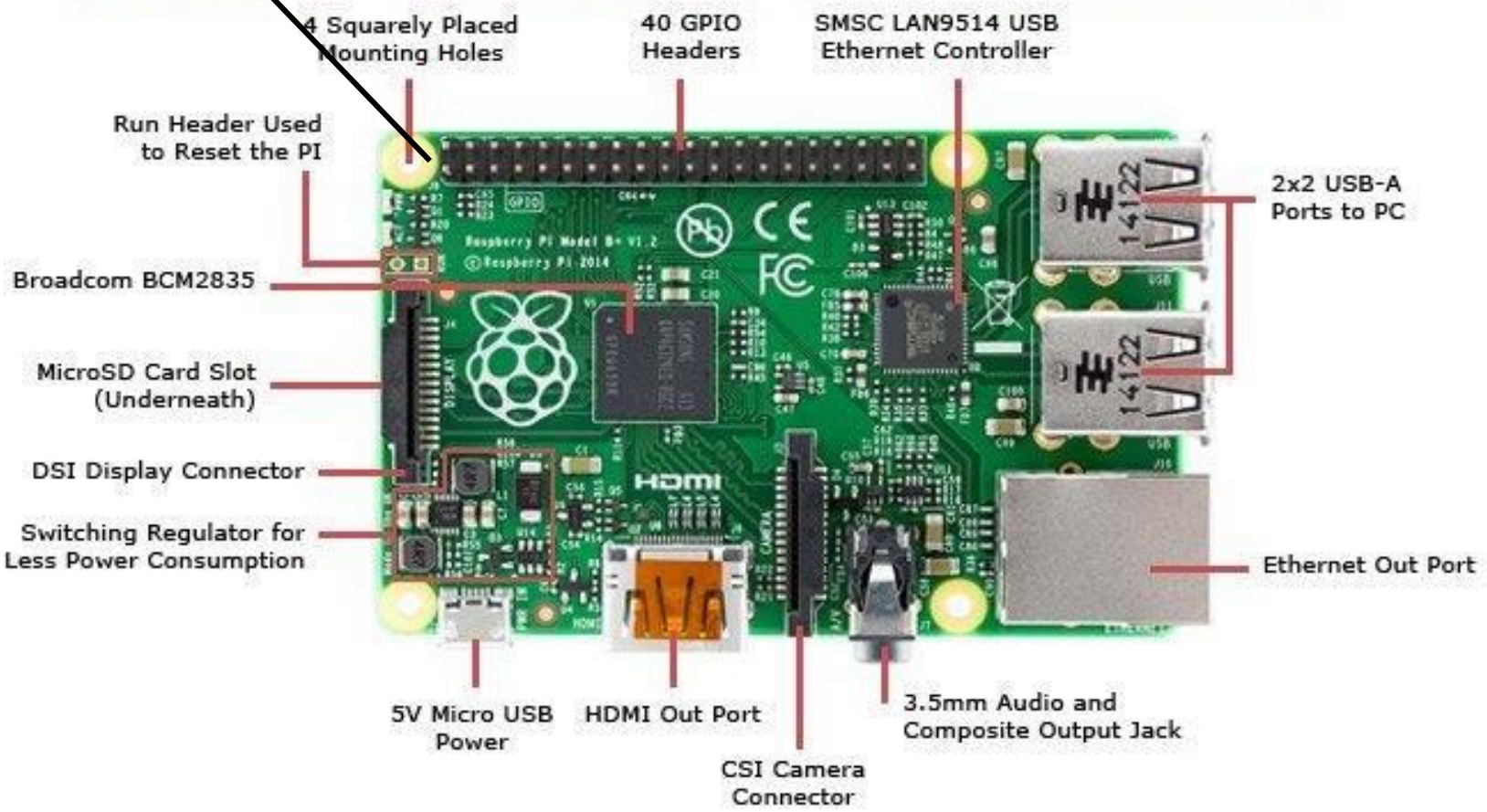
- After you've included the RPi.GPIO module, the next step is to determine the **pin-numbering schemes** you want to use:

- GPIO.BOARD – Board numbering scheme. The pin numbers follow the pin numbers on header P1
- To specify in your code which number-system is being used, use the GPIO.setmode() function.

```
GPIO.setmode(GPIO.BOARD)
```

GPIO Pinout Diagram





Raspberry Pi 3 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
30	SDA0 (I2C ID EEPROM)	27	28	SCL0 (I2C ID EEPROM)	31
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

Setting a Pin Mode

- If you've used Arduino, you're probably familiar with the fact that you have to declare a "pin mode" before you can use it as either an input or output.
- To set a pin mode, use the setup([pin], [GPIO.IN, GPIO.OUT] function. So, if you want to set pin 18 as an output, for example, write:

```
GPIO.setup(18, GPIO.OUT)
```

Outputs

1. Digital Output

- To write a pin high or low, use the GPIO.output([pin], [GPIO.LOW, GPIO.HIGH]) function. For example, if you want to set pin 18 high, write:

```
GPIO.output(18, GPIO.HIGH)
```

- Writing a pin to GPIO.HIGH will drive it to 3.3V, and GPIO.LOW will set it to 0V. Alternative to GPIO.HIGH and GPIO.LOW, you can use either 1, True, 0 or False to set a pin value.

Contd…

2. PWM (“Analog”) Output

- PWM on the Raspberry Pi is about as limited as can be – one, single pin is capable of it: 12
- To initialize PWM, use GPIO.PWM([pin], [frequency]) function. To make the rest of your script-writing easier you can assign that instance to a variable. Then use pwm.start([duty cycle]) function to set an initial value. For example

```
pwm = GPIO.PWM(12, 0.6)
pwm.start(50)
```

- Just don’t forget to set the pin as an output before you use it for PWM.
- To turn PWM on that pin off, use the pwm.stop() command

Inputs

- If a pin is configured as an input, you can use the GPIO.input([pin]) function to read its value. The input() function will return either a True or False indicating whether the pin is HIGH or LOW. You can use an if statement to test this, for example…

```
if GPIO.input(17):
print("Pin 11 is HIGH")
else:
print("Pin 11 is LOW")
```

- It will read pin 17 and print whether it's being read as HIGH or LOW.

Delays

- If you need to slow your Python script down, you can add delays. To incorporate delays into your script, you'll need to include another module: time. This line, at the top of your script, will do it for you:

```
include time
```

- Then, throughout the rest of your script, you can use time.sleep([seconds]) to give your script a rest. You can use decimals to precisely set your delay. For example, to delay 250 milliseconds, write:

```
time.sleep(0.25)
```

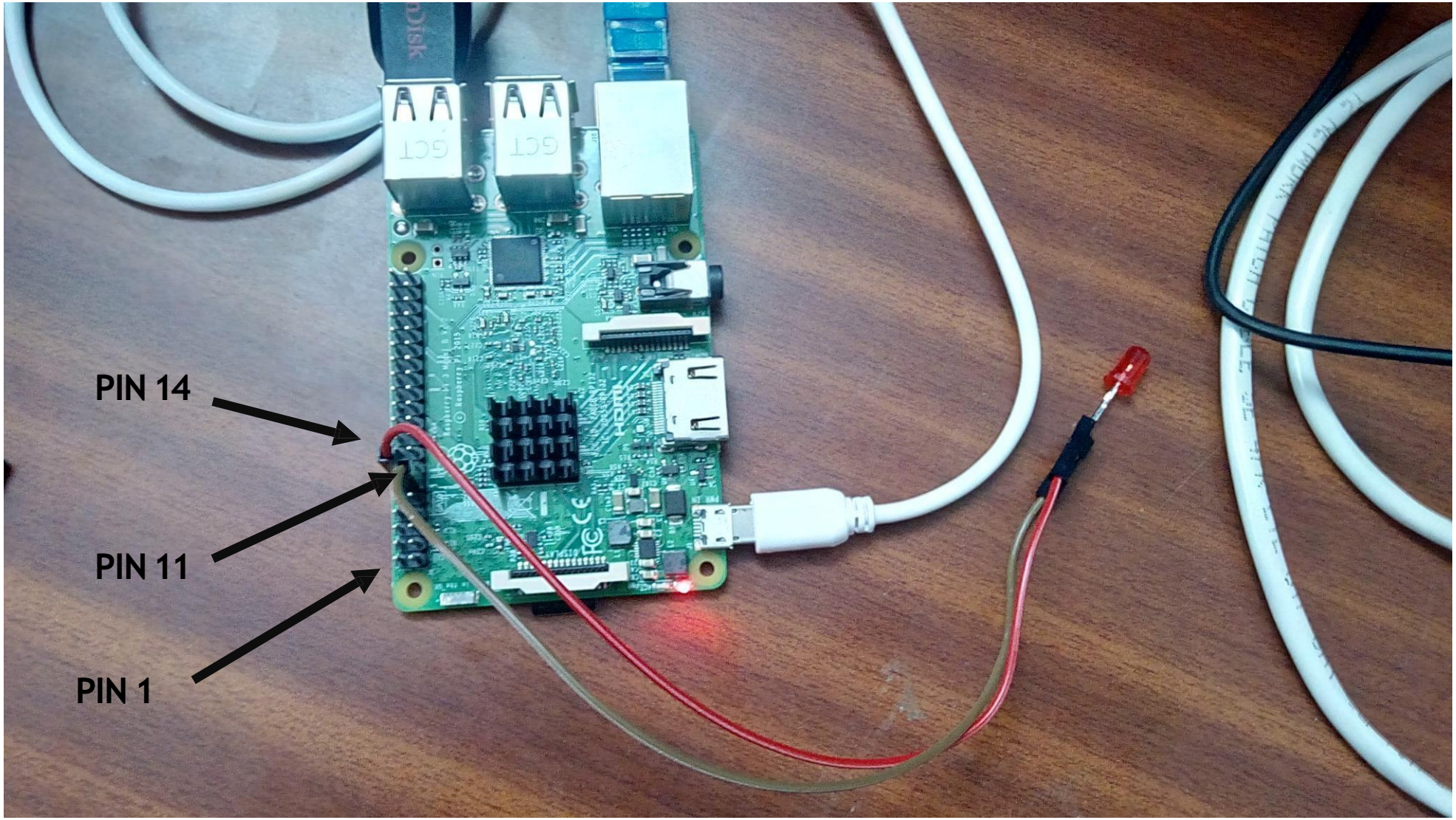
Garbage Collecting

- Once your script has run its course, be kind to the next process that might use your GPIOs by cleaning up after yourself. Use the GPIO.cleanup() command at the end of your script to release any resources your script may be using.
- Your Pi will survive if you forget to add this command, but it is good practice to include wherever you can.

Finally “LED Blink”

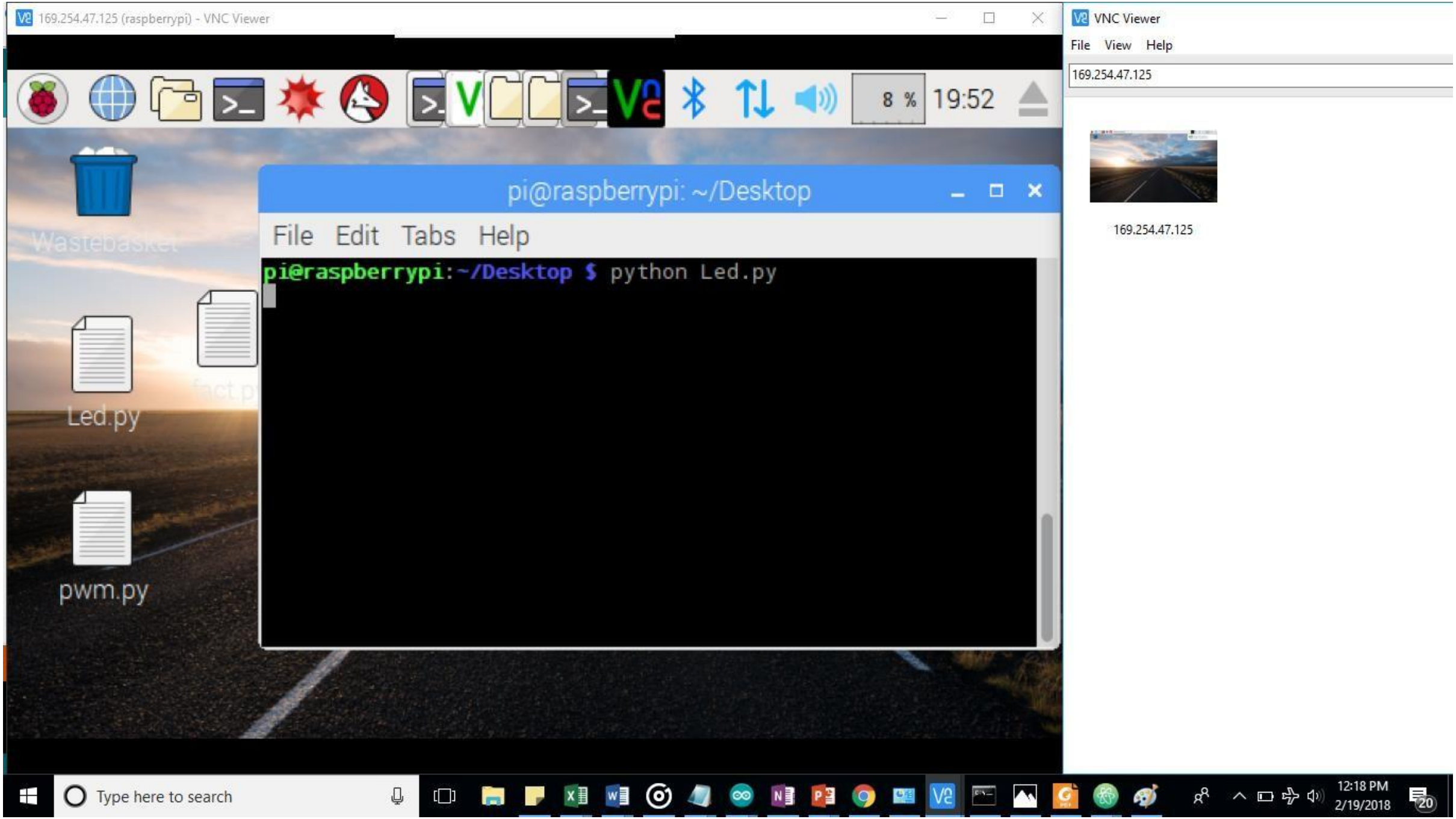
- Write a python code in the text editor provided in the raspberry pi desktop accessories to blink an LED after every 1 sec.
- Save the file in “.py” format at the desktop or wherever you may wish.
- Then run the command ”python filename.py” on the terminal after changing directory to Desktop using ”cd Desktop” command.

LED blink

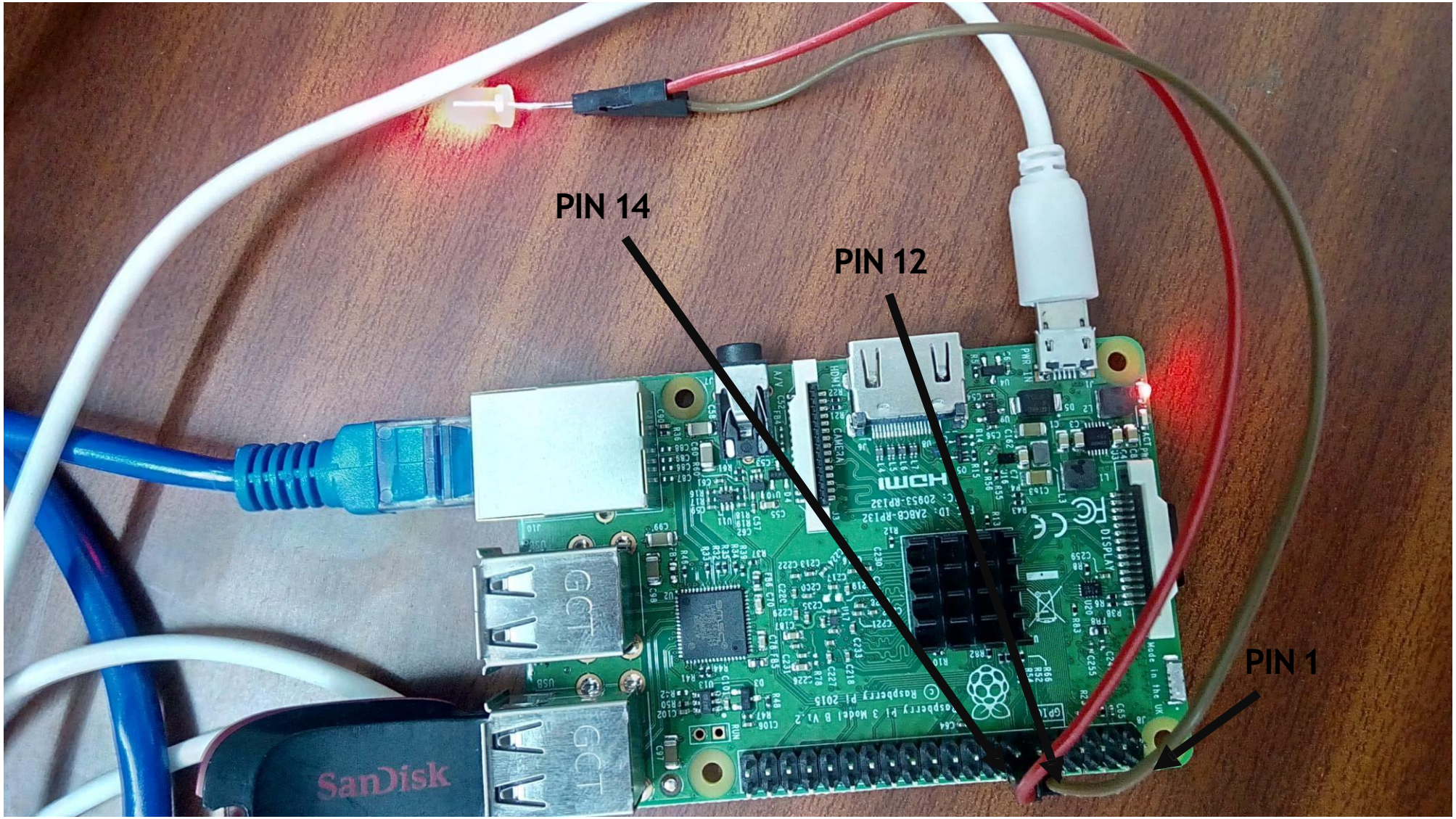


Code

```
import RPi.GPIO as GPIO
import time
LedPin = 11    # pin11
GPIO.setmode(GPIO.BOARD)       # Numbers GPIOs by physical location
GPIO.setup(LedPin, GPIO.OUT)   # Set LedPin's mode is output
GPIO.output(LedPin, GPIO.HIGH) # Set LedPin high(+3.3V) to turn on led
try:
 while True:
    GPIO.output(LedPin, GPIO.HIGH)  # led on
    time.sleep(1)
    GPIO.output(LedPin, GPIO.LOW) # led off
    time.sleep(1)
except KeyboardInterrupt:
  GPIO.output(LedPin, GPIO.LOW)   # led off
  GPIO.cleanup()                  # Release resource
```


169.254.47.125 (raspberrypi) - VNC Viewer
8 %
19:52
pi@raspberrypi: ~/Desktop
File Edit Tabs Help
pi@raspberrypi:~/Desktop $ python Led.py
Led.py
pwm.py
VNC Viewer
File View Help
169.254.47.125
169.254.47.125
Type here to search
12:18 PM
2/19/2018

PWM



PWM

```python
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(12, GPIO.OUT)
pwm12 = GPIO.PWM(12,50)
pwm12.start(0)
try:
    while True:
        for dutyCycle in range(0,100,5):
            pwm12.ChangeDutyCycle(dutyCycle)
            time.sleep(0.1)
        for dutyCycle in range(100,0,-5):
            pwm12.ChangeDutyCycle(dutyCycle)
            time.sleep(0.1)
except KeyboardInterrupt:
    pwm12.stop()
GPIO.cleanup()
```