

Mục lục

1	Điều khiển và sao chép dữ liệu với Raspberry Pi	2
1.1	Điều khiển bằng cách kết nối trực tiếp với màn hình, bàn phím và chuột	2
1.1.1	Màn hình hỗ trợ cổng HDMI	2
1.1.2	Màn hình hỗ trợ cổng VGA	2
1.2	Điều khiển bằng giao tiếp nối tiếp thông qua cổng RS232	2
1.3	Điều khiển từ xa khi Raspberry Pi có kết nối mạng	3
2	Mạng Internet với Raspberry Pi	4
2.1	Kết nối internet với dây mạng LAN	4
2.2	Kết nối internet với USB Wifi	4
2.2.1	Cài đặt Drive	4
2.2.2	Cài đặt địa chỉ IP tĩnh	5
2.3	Sử dụng chung Wifi với Laptop	6
2.3.1	Hệ điều hành Ubuntu	6
3	Tự động đăng nhập Raspberry Pi sau khi khởi động	9
4	Chạy chương trình Python trên Raspberry Pi	10
4.1	Python trên Raspberry Pi	10
4.2	Tự động chạy một chương trình python sau khi Reboot	10
4.2.1	Sử dụng crontab	10
4.2.2	Sử dụng profile	12
4.2.3	Sử dụng cron kết hợp với tạo file .sh	12

Chủ đề 1

Điều khiển và sao chép dữ liệu với Raspberry Pi

1.1 Điều khiển bằng cách kết nối trực tiếp với màn hình, bàn phím và chuột

- *Điều khiển*: Kết nối chuột và bàn phím qua các cổng USB. Với màn hình thông thường có 2 loại: màn hình hỗ trợ cổng HDMI và màn hình hỗ trợ cổng VGA.
- *Sao chép dữ liệu*: Sử dụng USB.

1.1.1 Màn hình hỗ trợ cổng HDMI

Ta kết nối màn hình qua cable HDMI. Có thể bạn sẽ cần tùy chỉnh một số thông số sau cho phù hợp:

1.1.2 Màn hình hỗ trợ cổng VGA

Để hiển thị được, ta cần có cable chuyển đổi từ VGA sang HDMI.

1.2 Điều khiển bằng giao tiếp nối tiếp thông qua cổng RS232

- Thực hiện kết nối Pi và module RS232 như sau:

Pi	RS232
3.3V	VCC
TX	TX
RX	RX
GND	GND

- Cài đặt gói phần mềm `screen`: `sudo apt-get install screen` trên máy tính Ubuntu.
- Chạy lệnh sau: `sudo screen /dev/ttyUSB0 115200`
- Thực hiện xong lệnh trên, ta nhấn Enter một lần nữa để kết nối với Pi.
- Nhập username và password để đăng nhập.
- Sao chép dữ liệu: dùng USB.

- * Ta có thể dùng Putty (trên hệ điều hành Window) để điều khiển: chọn **Serial**, điền vào khung **Serial line** tên cổng (ví dụ: COM1, COM2,...), trong khung **Speed** điền tốc độ là 115200. Nhập username và password để đăng nhập.

1.3 Điều khiển từ xa khi Raspberry Pi có kết nối mạng

Khi Raspberry Pi có kết nối mạng Internet, ta có thể dùng các phần mềm: **SSH**, **Remote Desktop**, **VNC**,... để điều khiển.

- Kiểm tra địa chỉ IP của Pi bằng phần mềm: **ipscan** (trên Windows) hoặc **nmap** (trên Ubuntu).
- Chọn chương trình phù hợp để điều khiển Raspberry Pi:
 - Với **SSH**: không hỗ trợ giao diện đồ họa.
 - Với **Remote Desktop** (Pi cần cài đặt: **xrdp**, dùng lệnh: `sudo apt-get install xrdp`), **VNC**: có hỗ trợ giao diện đồ họa.
- Tùy theo chương trình bạn chọn: ta cần phải nhập địa chỉ IP, username và password (nếu có yêu cầu điền số **port**: ta điền 22).
- Sao chép dữ liệu:
 - Trên Window: dùng **Winscp**.
 - Trên Ubuntu: dùng **FileZilla**.
 - * Ta cũng cần nhập vào thông tin như trên để truy cập được Pi.
- *Lưu ý*: Phần trình bày trên áp ngay cho mạng nội bộ, khi không phải mạng nội bộ ta cần cấu hình mạng rồi mới áp dụng được hướng dẫn ở phần này.

Chủ đề 2

Mạng Internet với Raspberry Pi

2.1 Kết nối internet với dây mạng LAN

Trên Raspberry Pi có hỗ trợ cổng Ethernet, chúng ta có thể kết nối dây mạng trực tiếp vào đây.

2.2 Kết nối internet với USB Wifi

Trong phần này, mình sử dụng USB Wifi TP Link 725N

2.2.1 Cài đặt Drive

Tham khảo tại: <https://www.raspberrypi.org/forums/viewtopic.php?p=462982>
Thực hiện theo các bước sau:

- Xác định phiên bản hệ điều hành Raspbian:

```
1 $ uname -a
2 Linux raspberrypi 4.1.13+ #826 PREEMPT Fri Nov 13 20:13:22 GMT 2015 armv6l
   GNU/Linux
```

Trong ví dụ trên, phiên bản hệ điều hành là 4.1.13+ #826.

- Vào địa chỉ bên dưới để tải drive:

<https://www.raspberrypi.org/forums/viewtopic.php?p=462982>

- Cài đặt drive, thực hiện các lệnh bên dưới:

```
1 wget https://dl.dropboxusercontent.com/u/80256631/8188eu-201xyyzz.tar.gz
2 tar -zxvf 8188eu-201xyyzz.tar.gz
3 sudo install -p -m 644 8188eu.ko /lib/modules/$(uname
   -r)/kernel/drivers/net/wireless
4 sudo insmod /lib/modules/$(uname -r)/kernel/drivers/net/wireless/8188eu.ko
5 sudo depmod -a
```

Đối với Raspberry Pi 2, chúng ta chỉ cần thực hiện 2 lệnh sau:

```
1 tar xzf 8188eu-201xyyzz.tar.gz
2 ./install.sh
```

Với 8188eu-201xyyzz.tar.gz là drive phù hợp với phiên bản hệ điều hành của bạn.

Bạn có thể tải drive từ máy tính rồi chép vào Raspberry Pi để cài đặt (cách này dùng cho Pi chưa được kết nối với Internet).

- * Trong quá trình cập nhật các phiên bản mới của hệ điều hành, khi đó Raspberry Pi không còn nhận USB Wifi nữa, lúc đó ta cần cài đặt Drive mới cho USB Wifi.

2.2.2 Cài đặt địa chỉ IP tĩnh

Tham khảo tại địa chỉ:

<http://weworkweplay.com/play/automatically-connect-a-raspberry-pi-to-a-wifi-network/>

Ta sửa đổi nội dung của 2 tập tin dưới đây:

- Tập tin: `interfaces`, mở tập tin:

```
1 $ sudo nano /etc/network/interfaces
```

và thay đổi nội dung như sau:

```
1 auto lo
2 iface lo inet loopback
3
4 auto eth0
5 allow-hotplug eth0
6 iface eth0 inet manual
7
8 auto wlan0
9 allow-hotplug wlan0
10 iface wlan0 inet manual
11 wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
12
13 auto wlan1
14 allow-hotplug wlan1
15 iface wlan1 inet manual
16 wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
17
18 iface wlan0 inet static
19 address 192.168.0.105 #Địa chỉ IP của bạn
20 netmask 255.255.255.0 #Thay đổi
21 gateway 192.168.0.1 #Thay đổi
22 #Đề xem các thông số, dùng lệnh: ifconfig và netstat -rn
```

- Tập tin: `wpa_supplicant.conf`, mở tập tin:

```
1 $ sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

và thay đổi nội dung như sau:

```
1 ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
2 update_config=1
3
4 network={
5     ssid="Tam3" # Thay đổi: tên Wifi, ví dụ: Tam3
6     psk="21019400" #Thay đổi: mật khẩu wifi: 21019400
7     proto=WPA
8     key_mgmt=WPA-PSK
```

```
9 pairwise=TKIP
10 auth_alg=OPEN
11 }
```

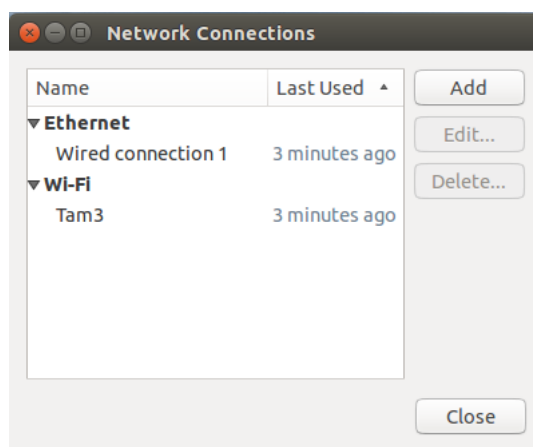
2.3 Sử dụng chung Wifi với Laptop

Ta kết nối cổng Ethernet của Pi và Laptop với nhau. Sử dụng tính năng Share Wifi trên Laptop:

2.3.1 Hệ điều hành Ubuntu

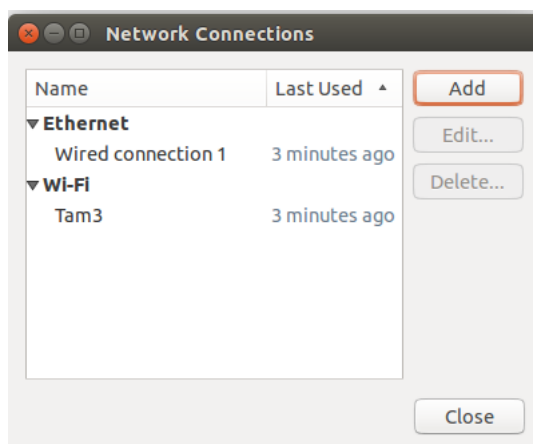
Thực hiện theo các bước sau:

- Trong thanh tìm kiếm Dash: gõ vào Network Connections, chọn Network Connections để mở lên.



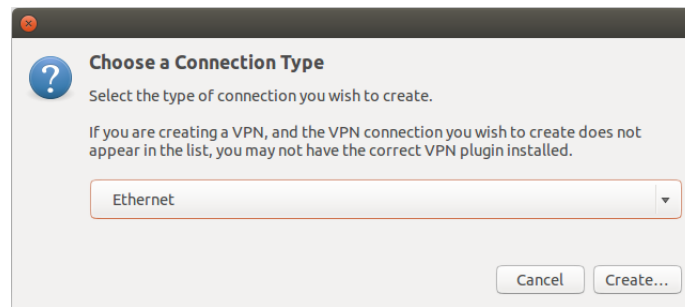
Hình 2.1: Mở Network Connections

- Chọn Add:



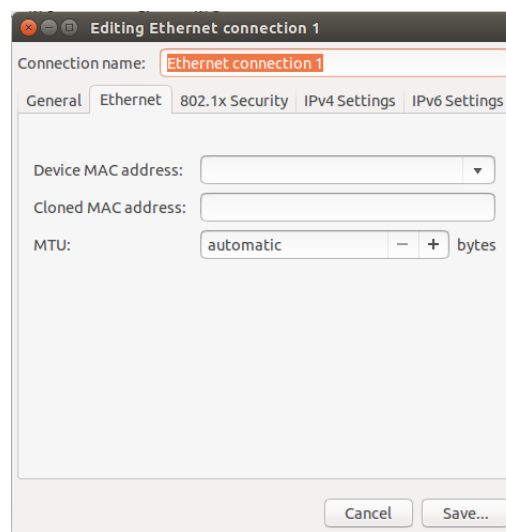
Hình 2.2: Chọn Add

- Chọn Create:



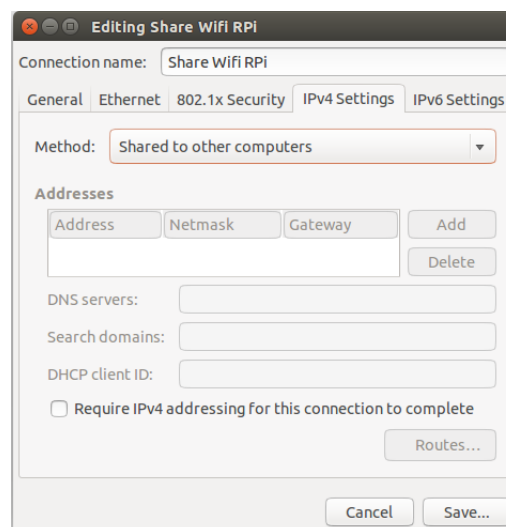
Hình 2.3: Chọn Create. . .

- Điền tên trong ô Connection name:



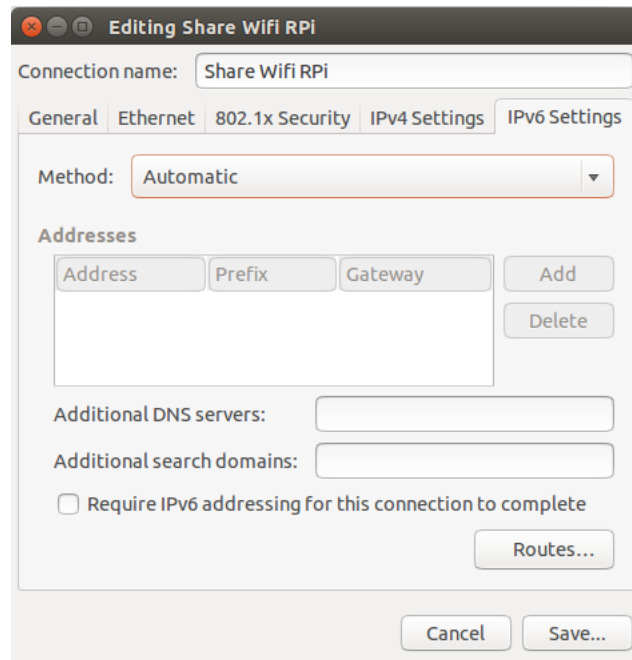
Hình 2.4: Điền tên trong ô Connection name

- Chọn tab IPv6 Settings, chọn Method là Automactic:



Hình 2.5: Chọn tab IPv6 Settings, chọn Method là Automactic

- Chọn tab IPv4 Settings, chọn Method là Share to other computers:



Hình 2.6: Chọn tab IPv6 Settings, chọn Method là Share to other computers

- Chọn Save rồi chọn Close.
- Mở cửa sổ lệnh Terminal gõ lệnh:

```
1 $ sudo cat /var/lib/misc/dnsmasq.leases
2 1461927547 b8:27:eb:6a:bf:9a 10.42.0.31 raspberrypi
   ff:eb:6a:bf:9a:00:01:00:01:1c:dd:60:6b:b8:27:eb:6a:bf:9a
```

- Địa chỉ của Pi lúc này là 10.42.0.31
- Truy cập qua SSH bằng lệnh:

```
1 $ ssh pi@10.42.0.31
```

- Nhập password đăng nhập tài khoản username là pi.

Chủ đề 3

Tự động đăng nhập Raspberry Pi sau khi khởi động

Tham khảo tại:

<http://www.raspberrypi-spy.co.uk/2015/02/how-to-autorun-a-python-script-on-raspberry-pi-boot/>

Mặc định khi Raspberry Pi khởi động, cần phải đăng nhập username và password mới sử dụng được. Với một số ứng dụng thực tế, cần tự động đăng nhập mới có thể hoạt động được.

Thực hiện theo các bước sau:

- Mở file `inittab`, dùng lệnh:

```
1 $ sudo nano /etc/inittab
```

- Tìm đến dòng dưới và thêm dấu `#` vào trước nó:

```
1 1:2345:respawn:/sbin/getty 115200 tty1
```

(tìm đến dòng có `1:2345:respawn:/sbin/getty` là được, còn các tham số phía sau tùy thuộc vào phiên bản hệ điều hành).

- Thay thế dòng trên bằng dòng sau (với `pi` là tên username):

```
1 1:2345:respawn:/bin/login -f pi tty1 </dev/tty1 >/dev/tty1 2>&1
```

- Nhấn `Ctrl - X - Y` để lưu lại nội dung thay đổi và thoát.
- Khởi động lại Pi: `sudo reboot`.

Chủ đề 4

Chạy chương trình Python trên Raspberry Pi

4.1 Python trên Raspberry Pi

Ta phân biệt theo 2 trường hợp sau:

- Chạy các lệnh không liên quan đến phần cứng là các chân GPIO, ta sẽ gõ các lệnh sau:

- Chạy ở chế độ dòng lệnh:

```
1 $ python
```

- Khi đã có sẵn một file `.py` (ví dụ: `file.py`):

```
1 $ python file.py
```

- Các lệnh liên quan đến phần cứng can thiệp vào các chân GPIO hoặc cần quyền root, ta phải chạy với quyền root: dùng `sudo python` hoặc `sudo python file.py` (cách dùng 2 lệnh này giống như trên).

4.2 Tự động chạy một chương trình python sau khi Reboot

Ở phần 4.1, ta phải thực hiện đánh lệnh thì file python mới được gọi, với nhiều ứng dụng tự động, cần tự động chạy chương trình python sau khi reboot. Ta có một số cách sau:

Ví dụ, ta cần chạy file python có tên là `myfile.py`.

4.2.1 Sử dụng cron

Tham khảo tại: <https://www.youtube.com/watch?v=8iU9TnYF0V0>
Thực hiện theo các bước sau:

- Không cần tự động đăng nhập.
- Copy file `myfile.py` đến thư mục `\home\pi` (dùng lệnh `cp`).
- Mở file `crontab` gõ lệnh:

```
1 $ sudo crontab -e
```

- Thêm dòng sau vào cuối file:

```
1 @reboot sudo python myfile.py &
```

Ký hiệu & có nghĩa là file myfile.py sẽ chạy nền.

```

pi@raspberrypi: ~
GNU nano 2.2.6 File: /tmp/crontab.oeI7Q0/crontab Modified

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
@reboot sudo python myfile.py &

```

- Nhấn Ctrl - X - Y để lưu lại nội dung thay đổi và thoát.
- Khởi động lại Pi: `sudo reboot`.

Nói thêm về cron¹ cron là một lịch trình, được khai báo với cú pháp sau:

```
1 1 2 3 4 5 /path/to/command arg1 arg2
```

Ý nghĩa của các tham số:

- 1 = Minutes (0 - 59)
- 2 = Hours (0 - 23)
- 3 = Days (0 - 31)
- 4 = Month (0 - 12)
- 5 = Day of the week (0 - 7) (Sunday is the 0 day)|

Ta có thể thay thế 1 trong 5 tham số trên bằng các tham số dưới đây:

- @reboot = Run once, at startup.
- @yearly = Run once a year

¹<https://embeddedday.com/projects/raspberry-pi/a-step-further/running-python-script-at-boot/>

- @monthly = Run once a month
- @weekly = Run once a week
- @daily = Run once a day
- @midnight = Pretty much the same as @daily
- @hourly = Run once an hour

4.2.2 Sử dụng profile

Tham khảo tại:

<http://www.raspberrypi-spy.co.uk/2015/02/how-to-autorun-a-python-script-on-raspberry-pi-boot/>

Thực hiện theo các bước sau:

- Làm cho Pi có thể tự động đăng nhập được (xem *chủ đề 3 trang 9*).
- Mở file profile, dùng lệnh:

```
1 $ sudo nano /etc/profile
```

- Kéo xuống dòng cuối dòng, thêm nội dung sau vào file:

```
1 sudo python /home/pi/myfile.py &
```

Ký hiệu & có nghĩa là file `myfile.py` sẽ chạy nền.

- Nhấn **Ctrl - X - Y** để lưu lại nội dung thay đổi và thoát.
- Khởi động lại Pi: `sudo reboot`.

4.2.3 Sử dụng cron kết hợp với tạo file .sh

Tham khảo tại:

<http://www.instructables.com/id/Raspberry-Pi-Launch-Python-script-on-startup/?ALLSTEPS>

Thực hiện theo các bước sau:

- Tạo một `.sh` (ví dụ: `launcher.sh`):

```
1 $ nano launcher.sh
```

- Nội dung file như sau (thay đổi nội dung của ví dụ cho phù hợp):

```
1 #!/bin/sh
2 # launcher.sh
3 # navigate to home directory, then to this directory, then execute python
  script, then back home
4
5 cd /
6 cd home/pi/bbt #thư mục chứa file .py
7 sudo python bbt.py #Lệnh chạy file python
8 cd /
```

Nhấn **Ctrl - X - Y** để lưu và thoát.

- Làm cho file `.sh` trở thành file thực thi (executable):

```
1 $ chmod 755 launcher.sh
```

- Kiểm tra file `.sh` ta vừa tạo có thực thi được không:

```
1 $ sh launcher.sh
```

- Tạo thư mục `logs` trong thư mục `\home\pi`:

```
1 $ cd ~  
2 $ mkdir logs
```

- Mở `cron`:

```
1 $ sudo crontab -e
```

- Thêm file `.sh` vào `cron`:

```
1 @reboot sh /home/pi/bbt/launcher.sh >/home/pi/logs/cronlog 2>&1
```

Nhấn `Ctrl - X - Y` để lưu và thoát.

- Khởi động lại Pi: `sudo reboot`