ติดตั้งแพคเกจ rosserial_arduino

```
sudo apt-get install ros-${ROS_DISTRO}-rosserial-arduino
```

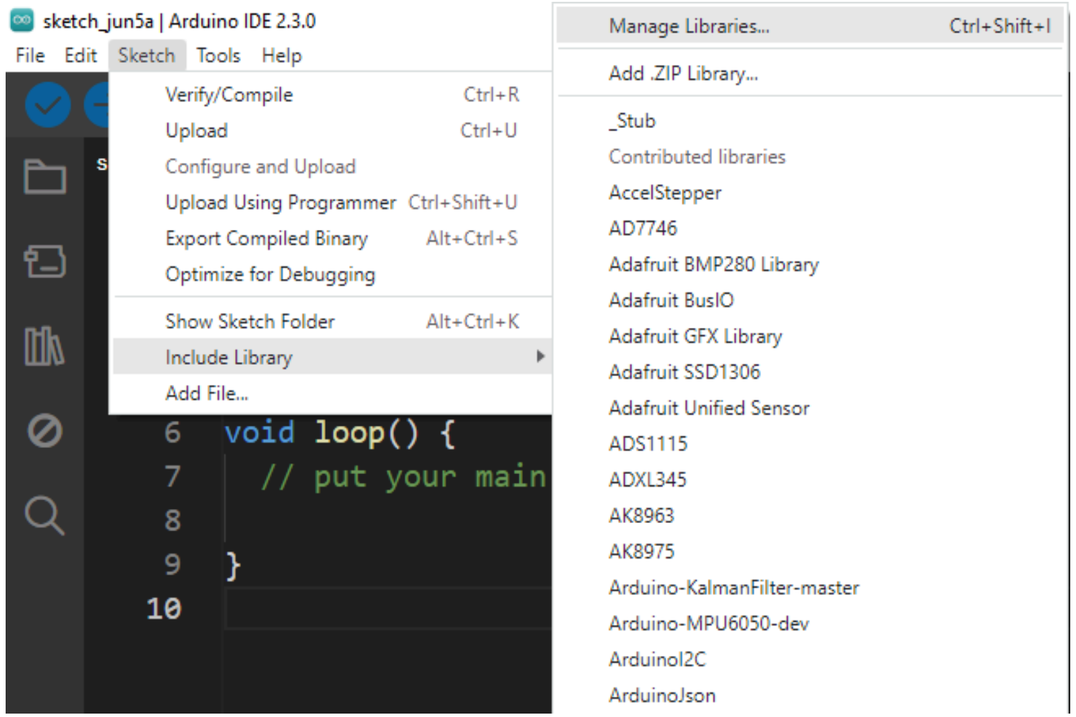
```
sudo apt-get install ros-${ROS_DISTRO}-rosserial
```

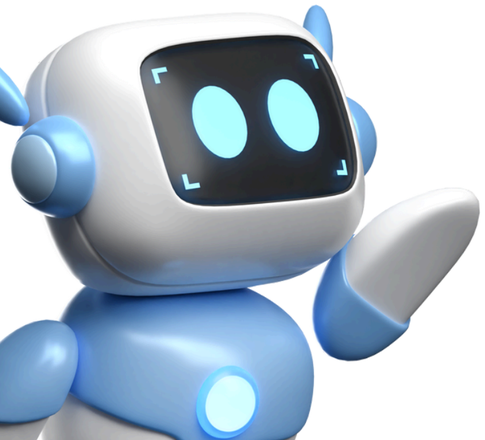
ติดตั้ง libraries rosserial_arduino ใน arduino ide(arduino integrated development environment)

เปิดโปรแกรม arduino ide version 2

เลือก Sketch ที่เมนูบาร์

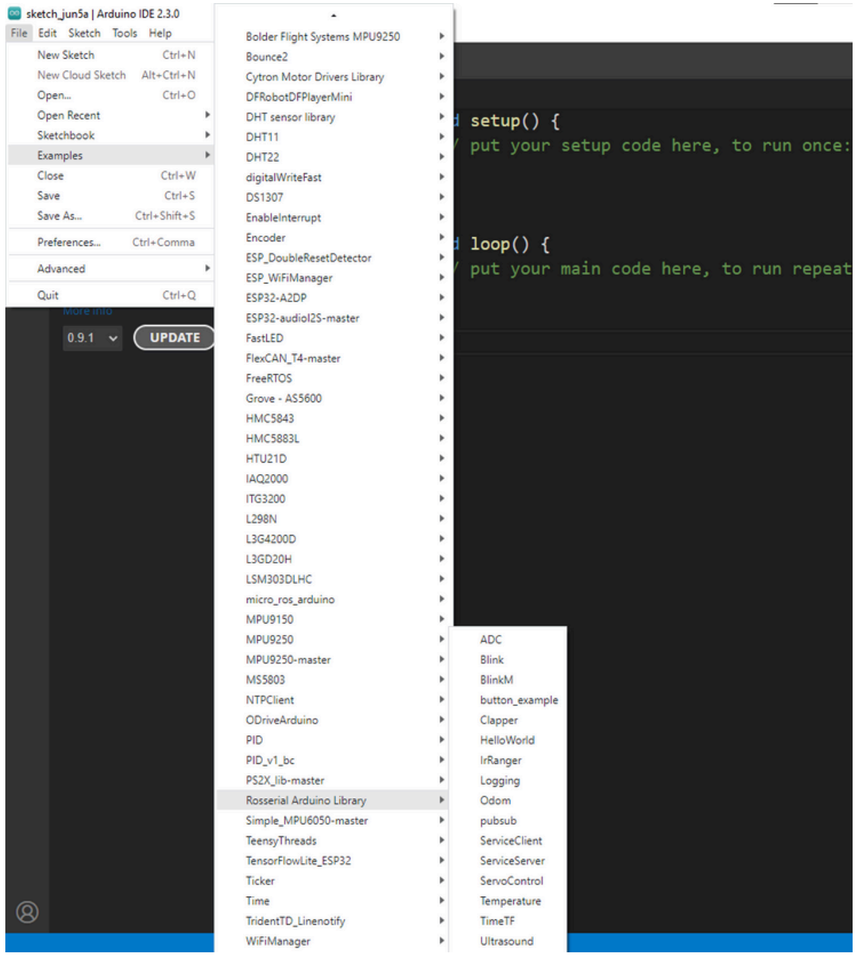
เลือก include Library >> กด Manage Libraries

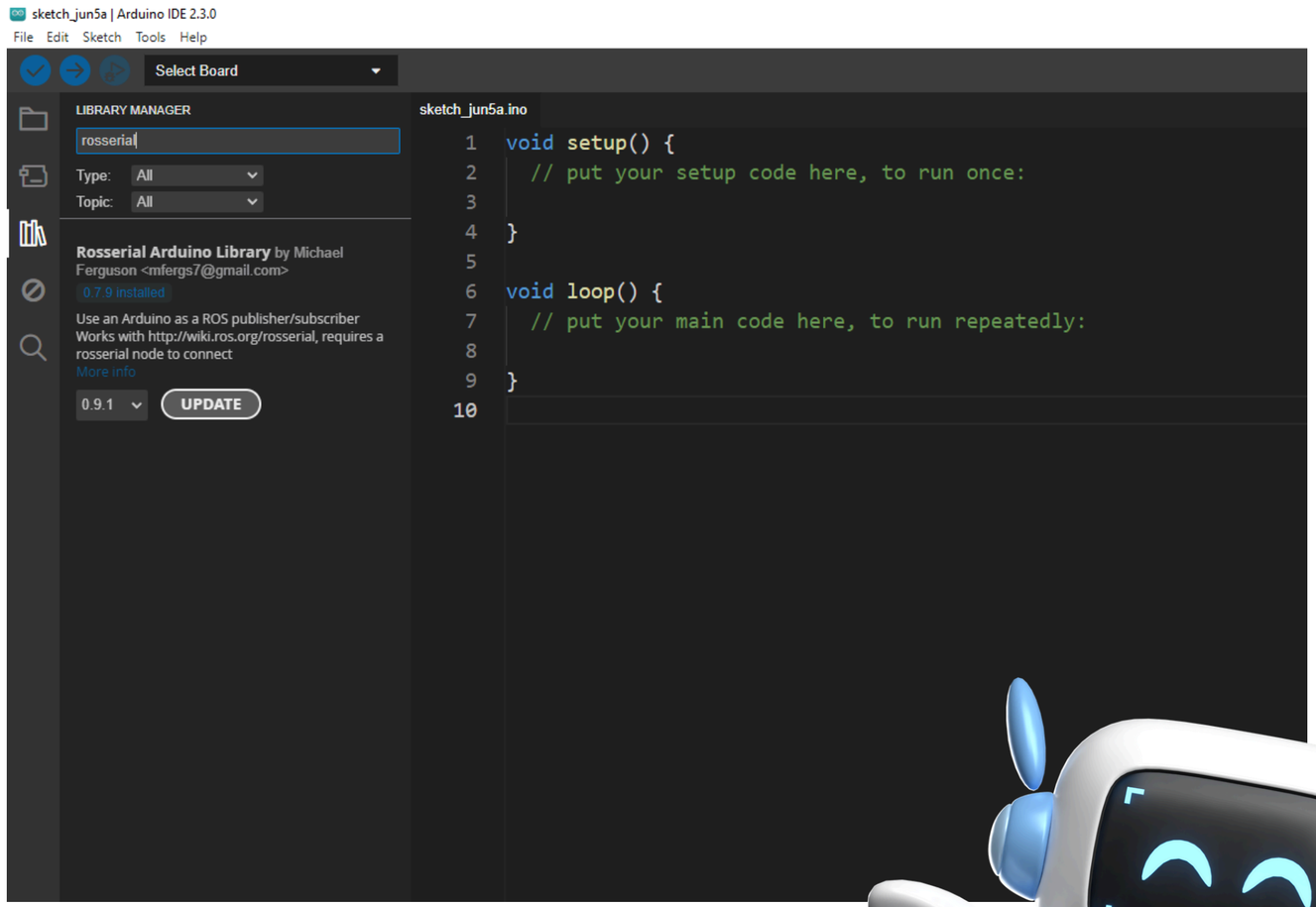


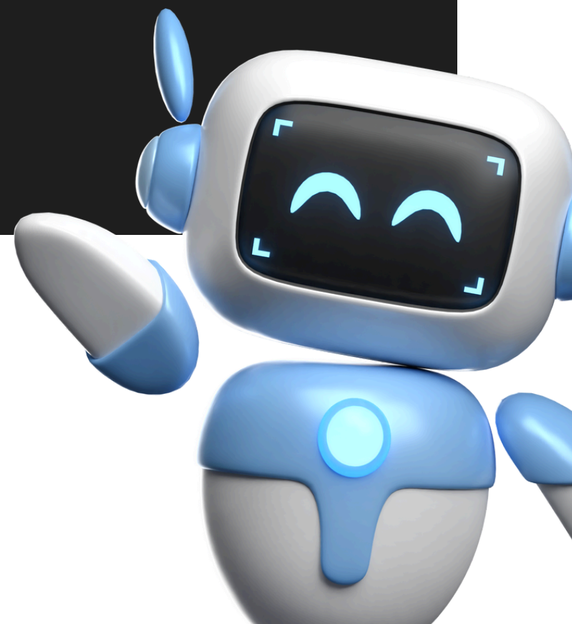


พิมพ์ rosserial ในช่องค้นหา

ให้เลือกติดตั้ง version 0.9.1







PUBLISHER

อธิบายโค้ด Publisher

Publisher | Arduino IDE 2.3.0

File Edit Sketch Tools Help

Select Board

Publisher.ino

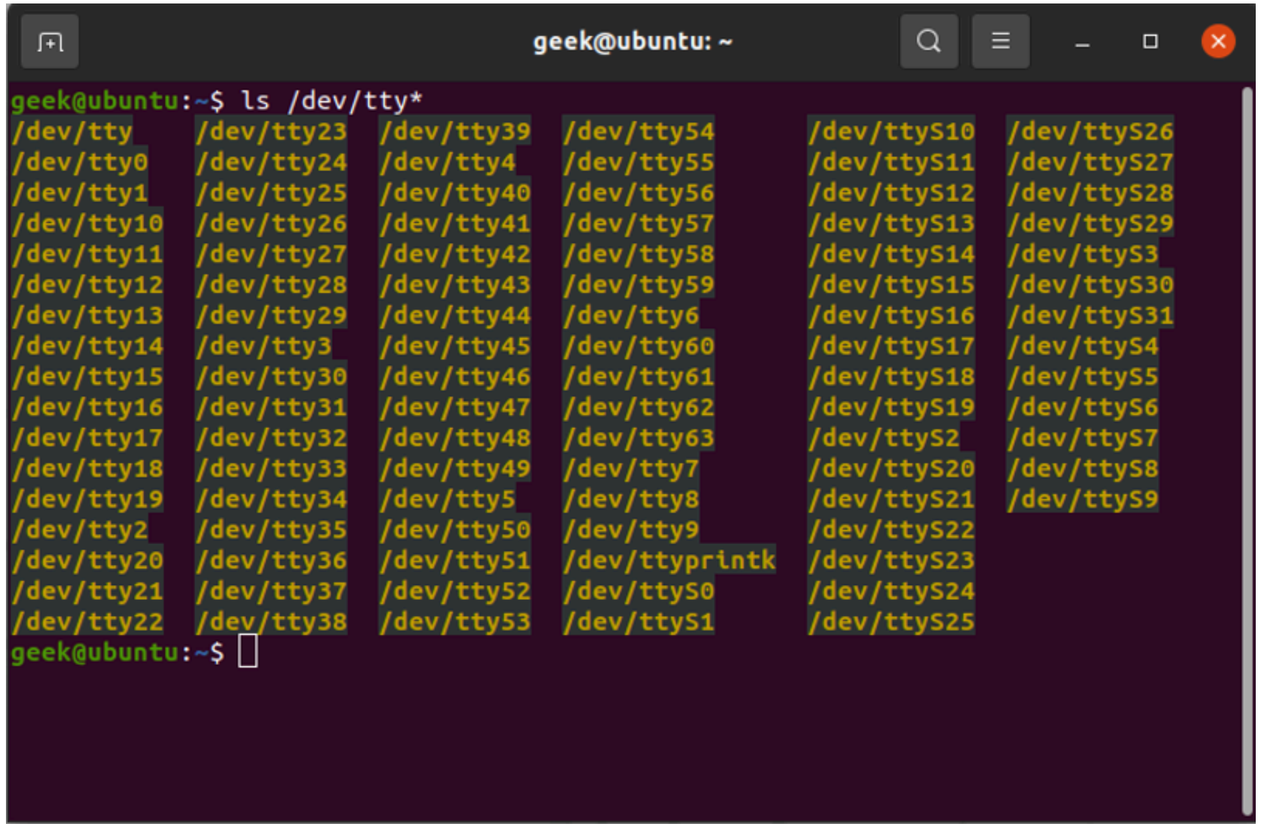
```
/*
 * rosserial Publisher Example
 * Prints "hello world!"
 */

#include <ros.h>   // เรียกใช้ library ของ ROS
#include <std_msgs/String.h>// เรียกใช้ library std_msgs type เป็น String(ข้อความ)

ros::NodeHandle  nh; //การสร้าง NodeHandle ชื่อ nh

std_msgs::String str_msg;  //สร้างตัวแปรชื่อ str_msg มารับตัวจัดการ msgs type String
ros::Publisher chatter("chatter", &str_msg); //สร้างตัวแปรชื่อ chatter มารับความสามารถ Publisher    ชื่อตัวแปร("ชื่อ TOPIC", &ตัวแปรที่สร้างให้มาจัดการ msgs);

char hello[13] = "hello ros!";//สร้างตัวแปรชนิด  char ให้เก็บอักขระได้ 13 ตัว hello เก็บอักขระต่อกันเป็นคำว่า  hello ros!

void setup()
{
  nh.initNode(); // สร้าง node
  nh.advertise(chatter); //ประกาศ Publisher
}

void loop()
{
  str_msg.data = hello; //เอาตัวแปร str_msg ที่สร้างจากบรรทัดที่ 11 มารับค่าของ hello คือคำว่า hello ros!
  chatter.publish( &str_msg );//เอาตัวแปร chatter ที่เป็น Publisher มาส่งข้อมูลไปยัง  ros โดยส่งคำว่า hello ros! ที่เก็บในตัวแปร str_msg
  nh.spinOnce();//ให้ ros ทำซ้ำ
  delay(1000);
}
```

Upload code

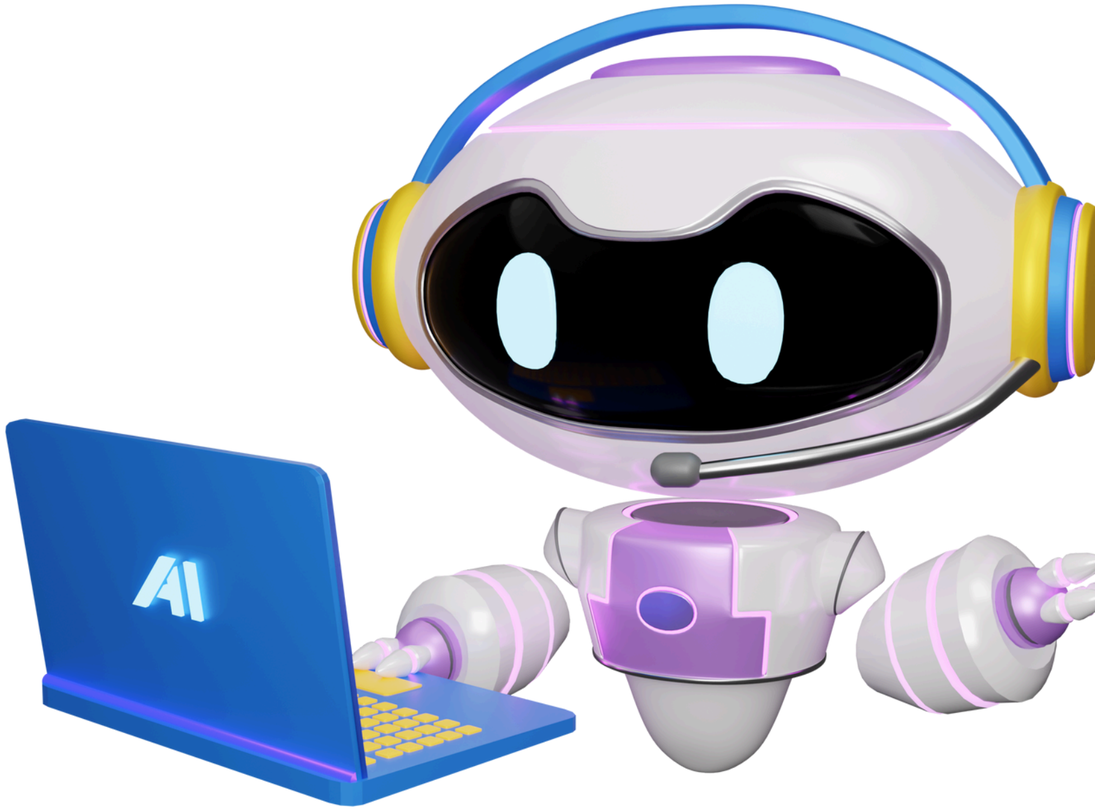
เปิด Vmware และเข้า ubuntu Crrl+Alt+T เปิด terminal

หา port ของบอร์ด arduino ด้วยคำสั่ง

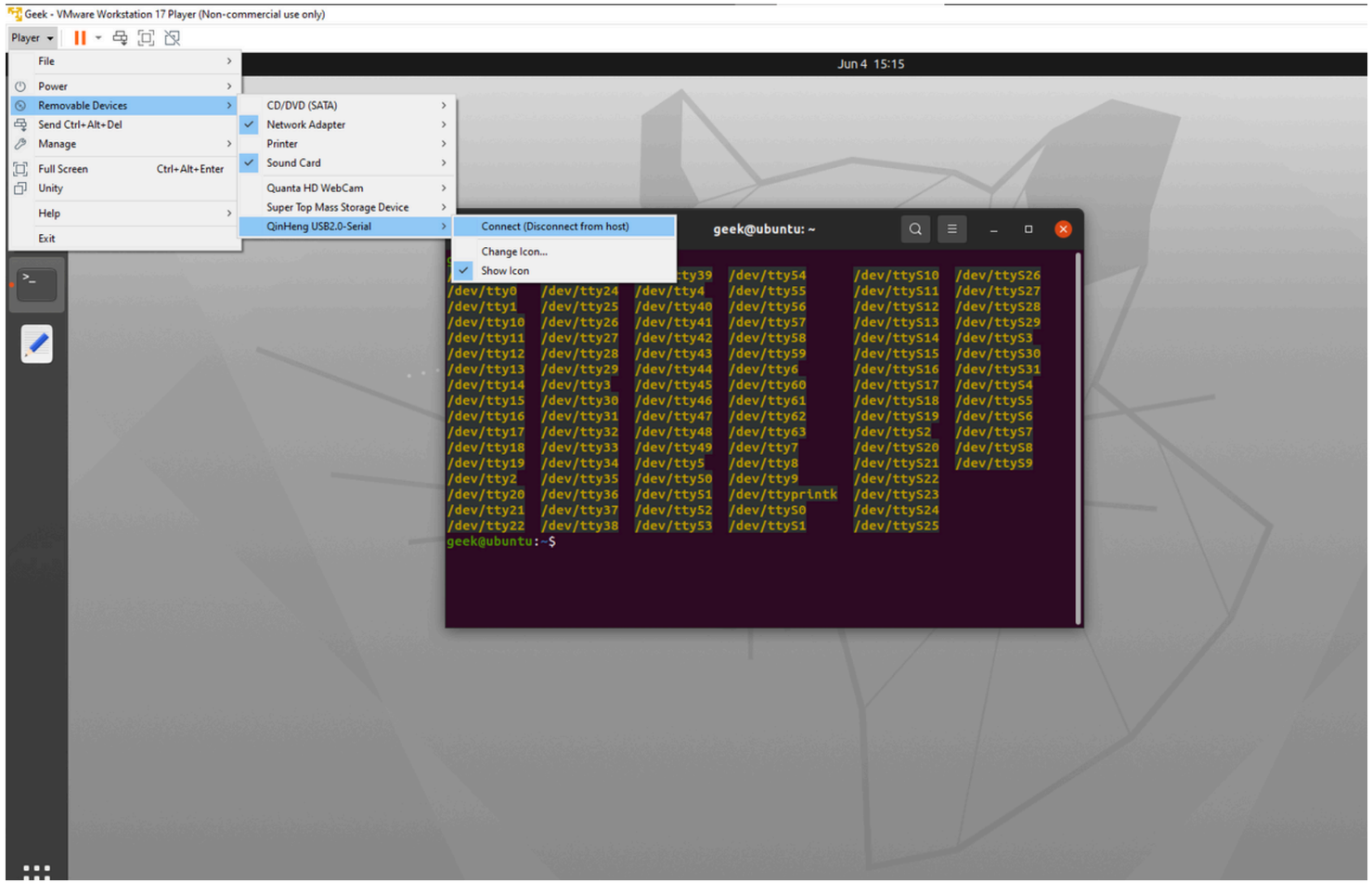
```
ls /dev/tty*
```



จะเห็นว่าไม่เจอ port ที่เพิ่มเข้ามา ไม่เป็น ACM ก็จะเป็น USB



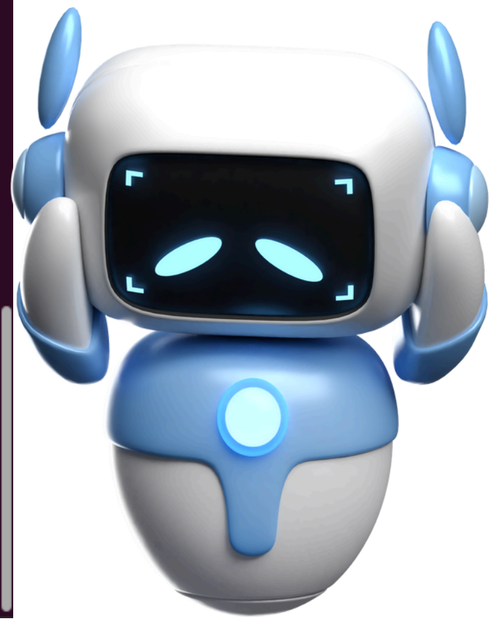
ทำตามดังรูปที่แสดง กด connect



ls /dev/tty*

เมื่อเช็คอีกครั้งจะพบ port USB0 ที่เพิ่มเข้ามา

```
geek@ubuntu: ~
/dev/tty19  /dev/tty34  /dev/tty5   /dev/tty8        /dev/ttyS21  /dev/ttyS9
/dev/tty2   /dev/tty35  /dev/tty50  /dev/tty9        /dev/ttyS22
/dev/tty20  /dev/tty36  /dev/tty51  /dev/ttyprintk   /dev/ttyS23
/dev/tty21  /dev/tty37  /dev/tty52  /dev/ttyS0       /dev/ttyS24
/dev/tty22  /dev/tty38  /dev/tty53  /dev/ttyS1       /dev/ttyS25
geek@ubuntu:~$ ls /dev/tty*
/dev/tty    /dev/tty23  /dev/tty39  /dev/tty54       /dev/ttyS10  /dev/ttyS26
/dev/tty0   /dev/tty24  /dev/tty4   /dev/tty55       /dev/ttyS11  /dev/ttyS27
/dev/tty1   /dev/tty25  /dev/tty40  /dev/tty56       /dev/ttyS12  /dev/ttyS28
/dev/tty10  /dev/tty26  /dev/tty41  /dev/tty57       /dev/ttyS13  /dev/ttyS29
/dev/tty11  /dev/tty27  /dev/tty42  /dev/tty58       /dev/ttyS14  /dev/ttyS3
/dev/tty12  /dev/tty28  /dev/tty43  /dev/tty59       /dev/ttyS15  /dev/ttyS30
/dev/tty13  /dev/tty29  /dev/tty44  /dev/tty6        /dev/ttyS16  /dev/ttyS31
/dev/tty14  /dev/tty3   /dev/tty45  /dev/tty60       /dev/ttyS17  /dev/ttyS4
/dev/tty15  /dev/tty30  /dev/tty46  /dev/tty61       /dev/ttyS18  /dev/ttyS5
/dev/tty16  /dev/tty31  /dev/tty47  /dev/tty62       /dev/ttyS19  /dev/ttyS6
/dev/tty17  /dev/tty32  /dev/tty48  /dev/tty63       /dev/ttyS2   /dev/ttyS7
/dev/tty18  /dev/tty33  /dev/tty49  /dev/tty7        /dev/ttyS20  /dev/ttyS8
/dev/tty19  /dev/tty34  /dev/tty5   /dev/tty8        /dev/ttyS21  /dev/ttyS9
/dev/tty2   /dev/tty35  /dev/tty50  /dev/tty9        /dev/ttyS22  /dev/ttyUSB0
/dev/tty20  /dev/tty36  /dev/tty51  /dev/ttyprintk   /dev/ttyS23
/dev/tty21  /dev/tty37  /dev/tty52  /dev/ttyS0       /dev/ttyS24
/dev/tty22  /dev/tty38  /dev/tty53  /dev/ttyS1       /dev/ttyS25
geek@ubuntu:~$
```



เรียก ros master

```
roscore
```

เรียกใช้แพคเกจ rosserial_arduino ด้วยคำสั่ง rosrun rosserial_arduino serial_node.py ตามด้วยของตัวเอง port เช่น

```
rosrun rosserial_arduino serial_node.py /dev/ttyUSB0
```

หรือ

```
rosrun rosserial_arduino serial_node.py /dev/ttyACM0
```

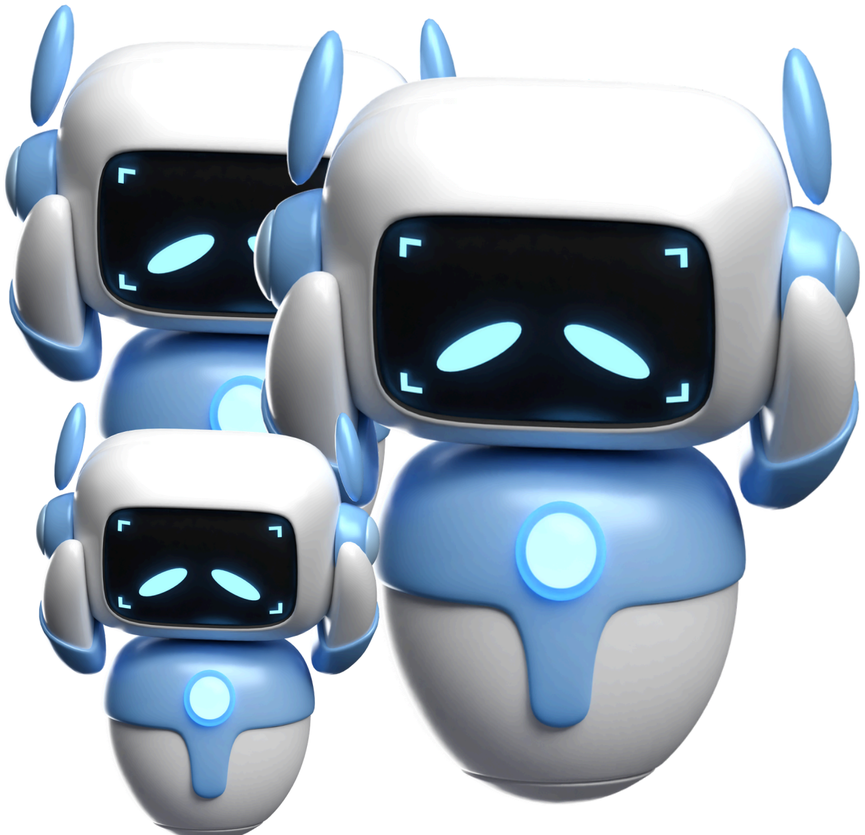
ผลลัพธ์

```
geek@ubuntu: ~
geek@ubuntu: ~        roscore http://ubuntu:11311/
geek@ubuntu:~$ rosrun rosserial_arduino serial_node.py /dev/ttyUSB0
[INFO] [1717539975.331142]: ROS Serial Python Node
[INFO] [1717539975.334219]: Connecting to /dev/ttyUSB0 at 57600 baud
[INFO] [1717539977.442161]: Requesting topics...
[INFO] [1717539978.146278]: Note: publish buffer size is 280 bytes
[INFO] [1717539978.147828]: Setup publisher on chatter [std_msgs/String]
```

เรียกดู topic ด้วยคำสั่ง rostopic list เห็น topic ที่สร้างชื่อ chatter

rostopic list

```
geek@ubuntu
geek@ubuntu: ~        roscore http://ubunt..
geek@ubuntu:~$ rostopic list
/chatter
/diagnostics
/rosout
/rosout_agg
```



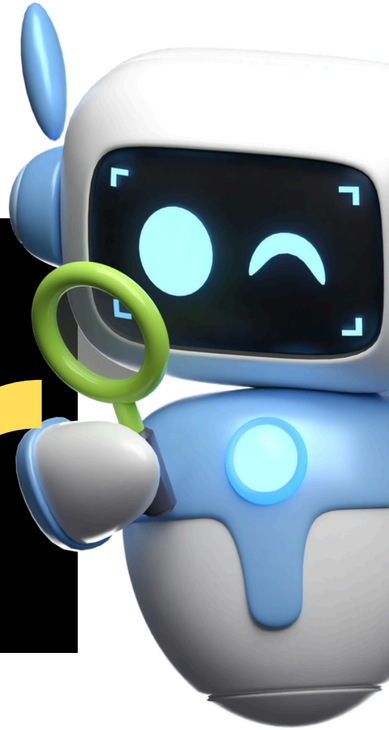
เรียกดูข้อความที่อยู่ใน topic /chatter

```
rostopic echo  /chatter
```

```
geek@ubuntu: ~
geek@ubuntu: ~    roscore http://ubunt...    geek@ubuntu: ~
geek@ubuntu:~$ rostopic list
/chatter
/diagnostics
/rosout
/rosout_agg
geek@ubuntu:~$ ros^C
geek@ubuntu:~$ rostopic echo /chatter
data: "hello ros!"
---
data: "hello ros!"
---
data: "hello ros!"
---
data: "hello ros!"
---
data: "hello ros!"
---
data: "hello ros!"
---
data: "hello ros!"
---
data: "hello ros!"
---
data: "hello ros!"
```

จะเห็นว่าใน topic มีการส่งค่าคำว่า hello ros

Subscriber



อธิบายโค้ด Subscriber

Subscriber | Arduino IDE 2.3.0

File Edit Sketch Tools Help

Select Board

Subscriber.ino

```
#include <ros.h> // นำเข้า header file สำหรับใช้งาน ROS บน Arduino
#include <std_msgs/String.h> // นำเข้า header file สำหรับใช้งานข้อความประเภท String จาก ROS

ros::NodeHandle nh; // สร้างตัวแปร nh สำหรับจัดการ node ใน ROS

void chatterCallback(const std_msgs::String& msg) {
  // ฟังก์ชันนี้จะถูกเรียกเมื่อมีข้อความใหม่ถูกส่งมาจาก topic ชื่อ chatter
  String listen = msg.data; // เก็บข้อความที่ได้รับในตัวแปร listen
  Serial.println(listen); // พิมพ์ข้อความที่ได้รับออกทาง serial monitor

  // หากมีข้อความถูกส่งเข้ามาให้หลอดไฟที่ขา 13 กระพริบ
  digitalWrite(13, HIGH); // เปิดหลอดไฟที่ขา 13
  delay(500); // หน่วงเวลา 500 milliseconds
  digitalWrite(13, LOW); // ปิดหลอดไฟที่ขา 13
  delay(500); // หน่วงเวลา 500 milliseconds
}

ros::Subscriber<std_msgs::String> sub("chatter", chatterCallback);
// สร้างตัวแปร sub สำหรับ subscribe กับ topic ชื่อ chatter และใช้ฟังก์ชัน chatterCallback เมื่อมีข้อความใหม่

void setup() {
  pinMode(13, OUTPUT); // กำหนดให้ขา 13 เป็นขาออก
  Serial.begin(115200); // เริ่มการติดต่อสื่อสารผ่าน serial ที่ความเร็ว 115200 baud rate
  nh.initNode(); // เริ่มต้น node ใน ROS
  nh.subscribe(sub); // subscribe กับ topic ชื่อ chatter โดยใช้ตัวแปร sub
}

void loop() {
  nh.spinOnce(); // ให้ ROS จัดการการสื่อสารและเรียก callback function ที่จำเป็น
  delay(10); // หยุดการทำงานชั่วคราวเป็นเวลา 10 milliseconds เพื่อป้องกันการใช้ CPU สูงเกินไป
}
```

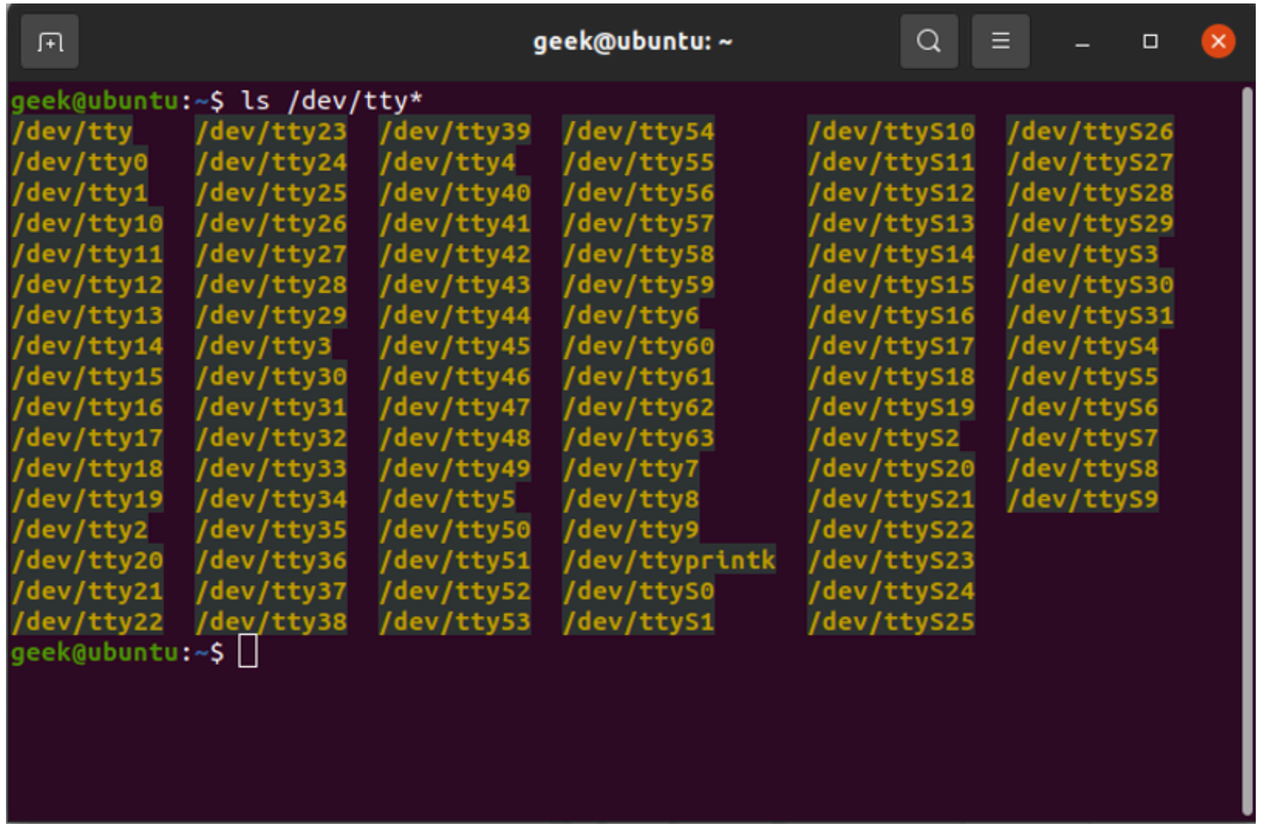
Ln 12, Col 45 ✕ No board selected

Upload code

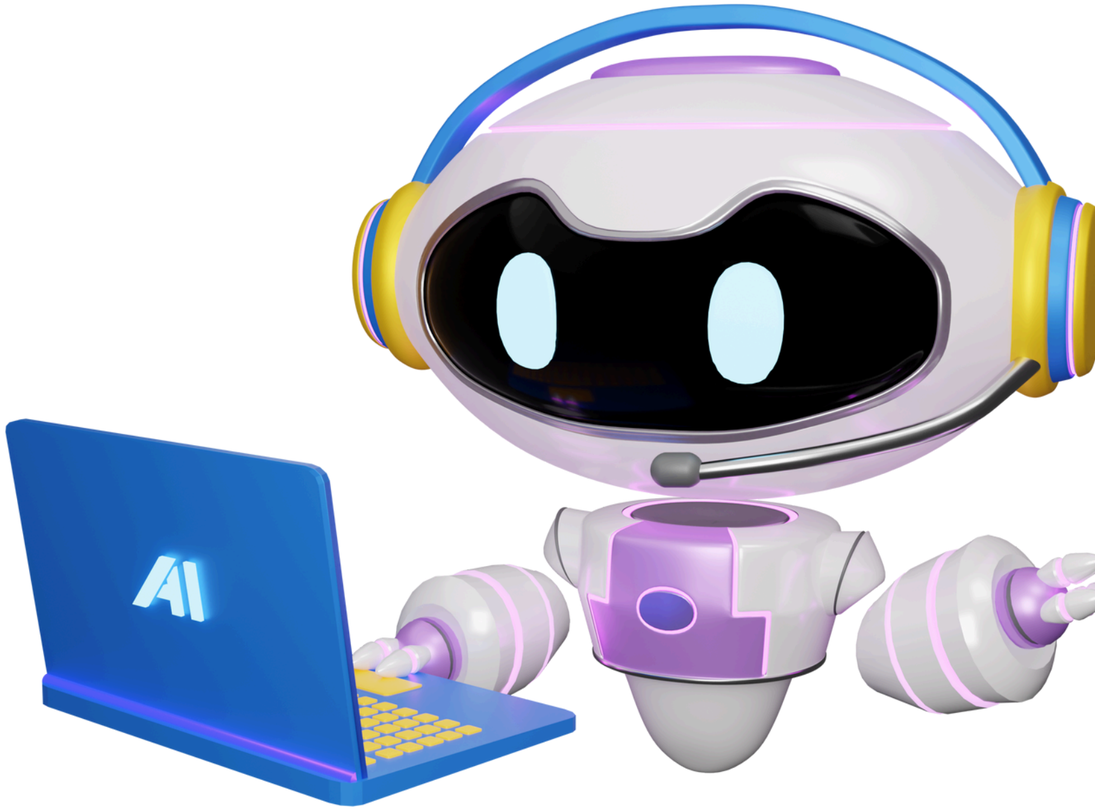
เปิด Vmware และเข้า ubuntu Crrl+Alt+T เปิด terminal

หา port ของบอร์ด arduino ด้วยคำสั่ง

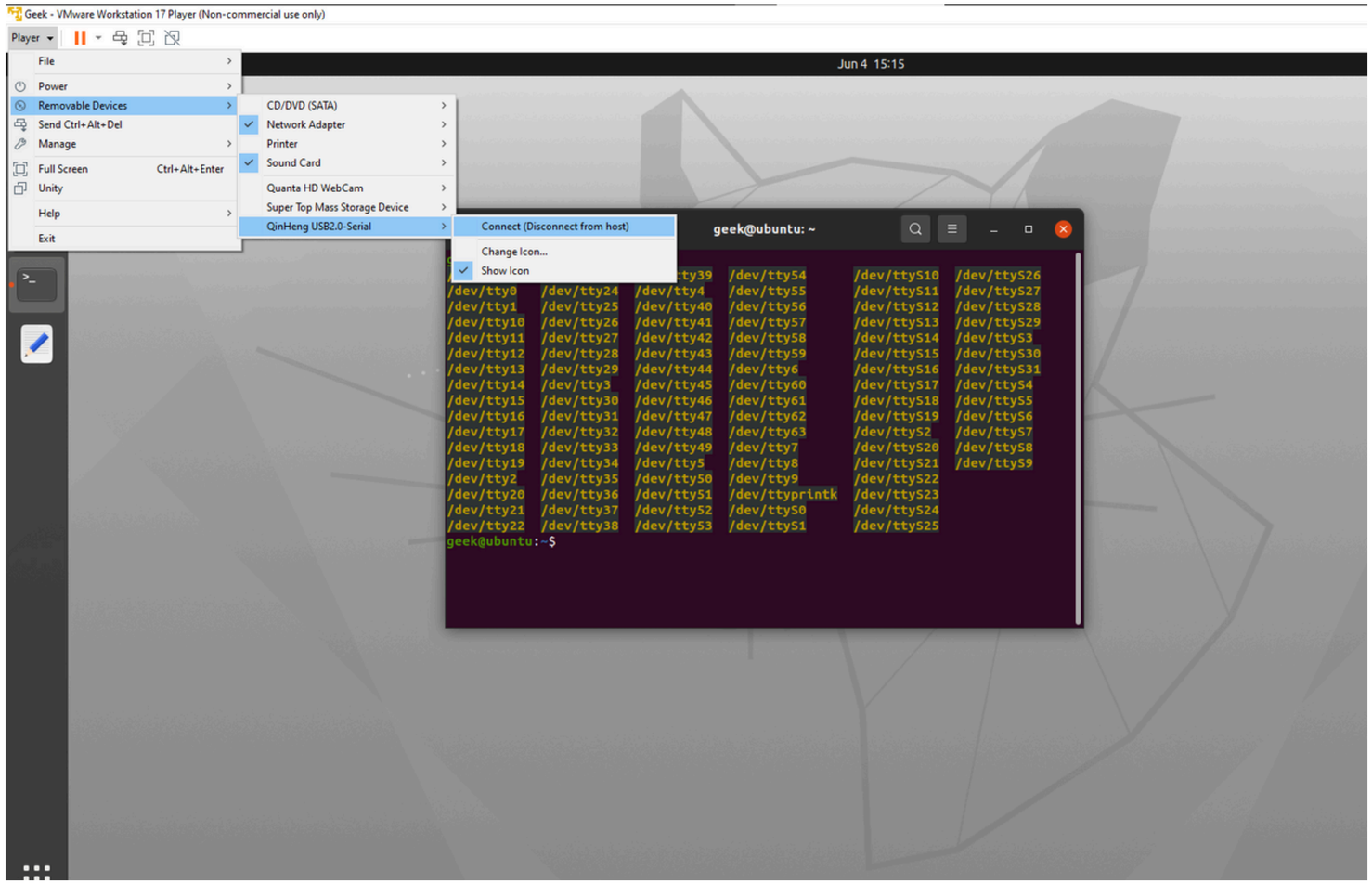
```
ls /dev/tty*
```



จะเห็นว่าไม่เจอ port ที่เพิ่มเข้ามา ไม่เป็น ACM ก็จะเป็น USB



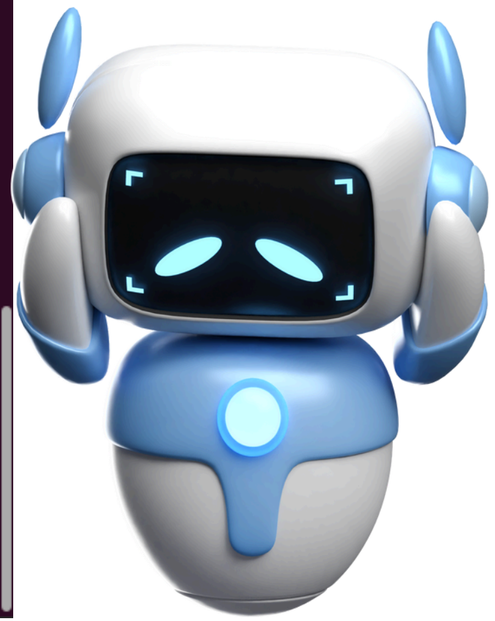
ทำตามดังรูปที่แสดง กด connect



ls /dev/tty*

เมื่อเช็คอีกครั้งจะพบ port USB0 ที่เพิ่มเข้ามา

```
geek@ubuntu: ~
/dev/tty19  /dev/tty34  /dev/tty5   /dev/tty8        /dev/ttyS21  /dev/ttyS9
/dev/tty2   /dev/tty35  /dev/tty50  /dev/tty9        /dev/ttyS22
/dev/tty20  /dev/tty36  /dev/tty51  /dev/ttyprintk   /dev/ttyS23
/dev/tty21  /dev/tty37  /dev/tty52  /dev/ttyS0       /dev/ttyS24
/dev/tty22  /dev/tty38  /dev/tty53  /dev/ttyS1       /dev/ttyS25
geek@ubuntu:~$ ls /dev/tty*
/dev/tty    /dev/tty23  /dev/tty39  /dev/tty54       /dev/ttyS10  /dev/ttyS26
/dev/tty0   /dev/tty24  /dev/tty4   /dev/tty55       /dev/ttyS11  /dev/ttyS27
/dev/tty1   /dev/tty25  /dev/tty40  /dev/tty56       /dev/ttyS12  /dev/ttyS28
/dev/tty10  /dev/tty26  /dev/tty41  /dev/tty57       /dev/ttyS13  /dev/ttyS29
/dev/tty11  /dev/tty27  /dev/tty42  /dev/tty58       /dev/ttyS14  /dev/ttyS3
/dev/tty12  /dev/tty28  /dev/tty43  /dev/tty59       /dev/ttyS15  /dev/ttyS30
/dev/tty13  /dev/tty29  /dev/tty44  /dev/tty6        /dev/ttyS16  /dev/ttyS31
/dev/tty14  /dev/tty3   /dev/tty45  /dev/tty60       /dev/ttyS17  /dev/ttyS4
/dev/tty15  /dev/tty30  /dev/tty46  /dev/tty61       /dev/ttyS18  /dev/ttyS5
/dev/tty16  /dev/tty31  /dev/tty47  /dev/tty62       /dev/ttyS19  /dev/ttyS6
/dev/tty17  /dev/tty32  /dev/tty48  /dev/tty63       /dev/ttyS2   /dev/ttyS7
/dev/tty18  /dev/tty33  /dev/tty49  /dev/tty7        /dev/ttyS20  /dev/ttyS8
/dev/tty19  /dev/tty34  /dev/tty5   /dev/tty8        /dev/ttyS21  /dev/ttyS9
/dev/tty2   /dev/tty35  /dev/tty50  /dev/tty9        /dev/ttyS22  /dev/ttyUSB0
/dev/tty20  /dev/tty36  /dev/tty51  /dev/ttyprintk   /dev/ttyS23
/dev/tty21  /dev/tty37  /dev/tty52  /dev/ttyS0       /dev/ttyS24
/dev/tty22  /dev/tty38  /dev/tty53  /dev/ttyS1       /dev/ttyS25
geek@ubuntu:~$
```



เรียก ros master

```
roscore
```

เรียกใช้แพคเกจ rosserial_arduino ด้วยคำสั่ง rosrun rosserial_arduino serial_node.py ตามด้วยของตัวเอง port เช่น

```
rosrun rosserial_arduino serial_node.py /dev/ttyUSB0
```

หรือ

```
rosrun rosserial_arduino serial_node.py /dev/ttyACM0
```