



๗๗ ๘๘  
๒๖

# GARDEN WARS 2018

• PRECISION AGRICULTURE •







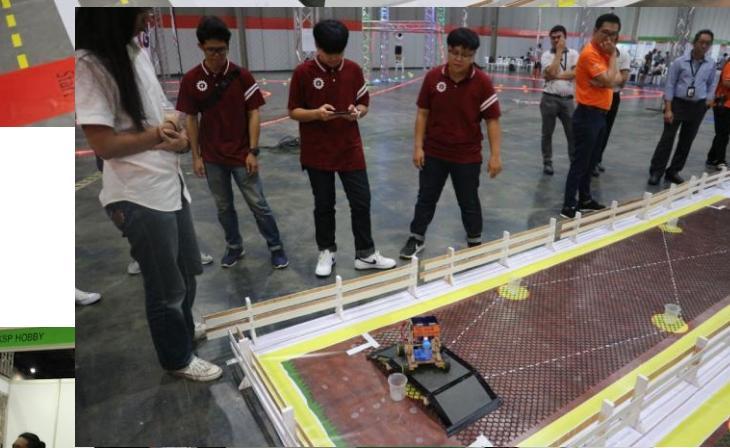
ENGINEERS  
LADKRABANG

ITE  
INFORMATION  
ENGINEERING  
KMITL



# GARDEN WARS 2018

• PRECISION AGRICULTURE •



# Team project: Seed Drilling Robot



**Garden Wars 2023**

กิจกรรมแข่งขันหุ่นยนต์ที่ดูแลเมล็ดของนักศึกษาปีที่ 1 หลักสูตร ASE  
เมื่อ วันที่ 9 นฤคจิกายน 2566

ในวิชา **Introduction to Agro-industrial Systems Engineering**  
ซึ่งนำเสนอ รหัสบัตร์บังคับด้วยบอร์ดไมโครคอนโทรลเลอร์ ควบคุมพืชด้วยหุ่นยนต์

- หุ่นยนต์จะต้องหุ่นยนต์ในโหมดเริ่มต้นต่อหน้าเริ่มการแข่งขัน
- หุ่นยนต์จะต้องหุ่นยนต์ตามเส้นทางที่กำหนดไว้
- กิมผู้บังคับหุ่นยนต์จะต้องมีผู้บังคับหุ่นยนต์ละ 1 คนเท่านั้น
- นาฬิกาจะเริ่มก้ากๆ กับเวลาเมื่อเริ่มการแข่งขัน และหุ่นยนต์เมื่อเสร็จสิ้นภารกิจ
- เมื่อเริ่มแข่งขันจะไม่สามารถล้มหุ่นยนต์ได้ หุ่นยนต์จะต้องหุ่นยนต์ไปท่ามกลางคนดู

ภาพแสดงความสำเร็จในการแข่งขัน

Shop 2 ผู้เขียน CCA





๗๗ ๘๘ ๒๖

# GARDEN WARS 2018

• PRECISION AGRICULTURE •





Blink | Arduino 1.0

Blink

```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/

void setup() {
    // initialize the digital pin as an output.
    // Pin 13 has an LED connected on most Arduino boards:
    pinMode(13, OUTPUT);
}

void loop() {
    digitalWrite(13, HIGH);      // set the LED on
    delay(1000);                // wait for a second
    digitalWrite(13, LOW);       // set the LED off
    delay(1000);                // wait for a second
}
```

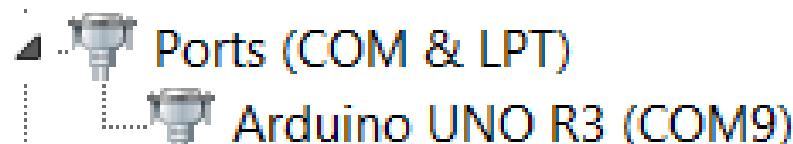
1

Arduino Uno on /dev/tty.usbmodemfd131

# Basic Arduino

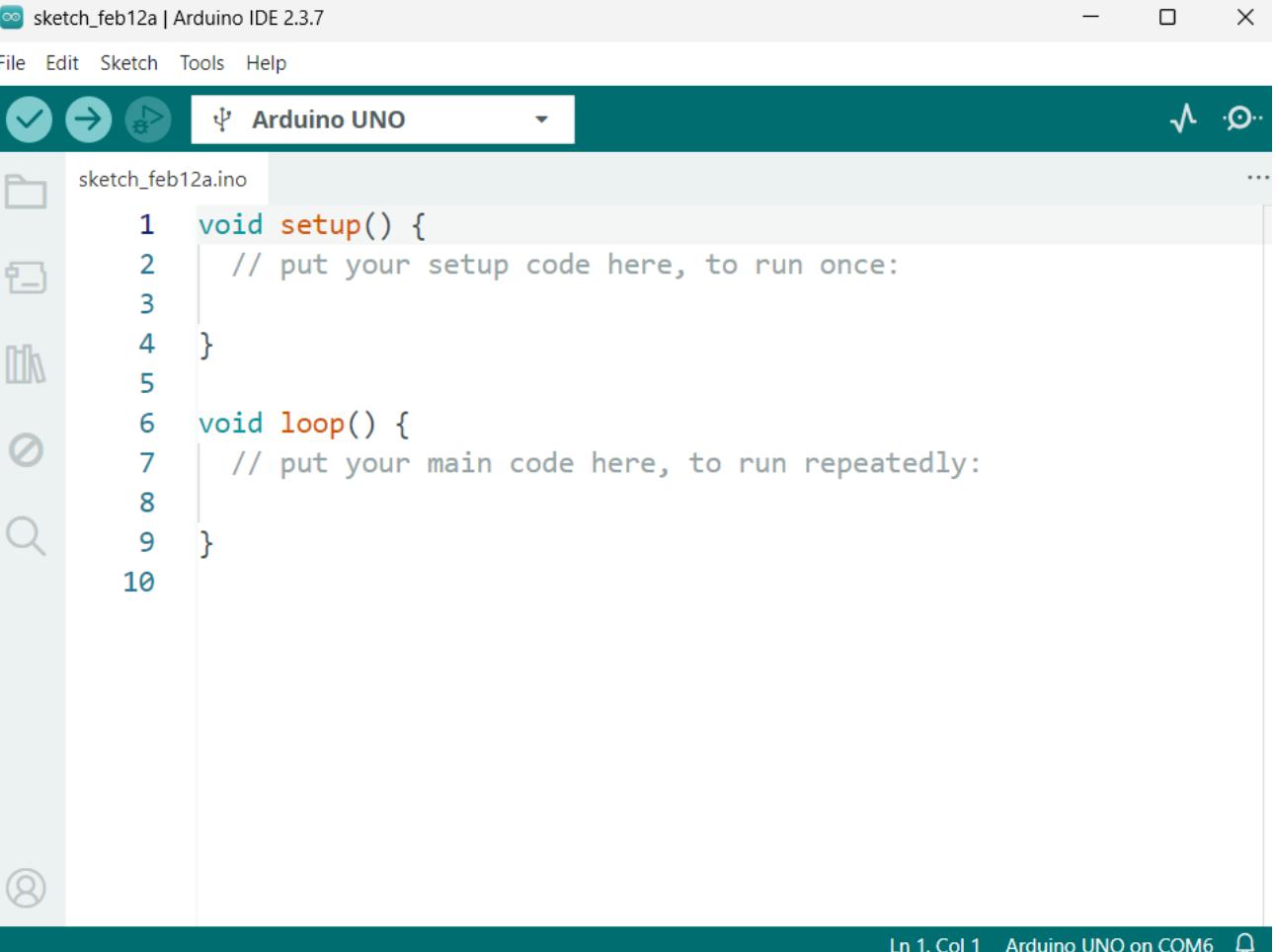
# Installation (Windows Only!)

- Visit [www.arduino.cc](http://www.arduino.cc) ; click on ‘Download’
- Download and install ‘Windows Installer’; Follow the instruction (check out all components including USB Driver)
- Connect Arduino board to the PC via a USB cable, wait until the driver installed.
  - Check if the "Arduino UNO R3 (COMxx)" exists under ‘Ports(COM & LPT)’ in the ‘Device Manager’



# The IDE (Integrated Development Environment)

- Launch the IDE
- Explorer the menu
- Configure the right Arduino board and COM port



```
sketch_feb12a | Arduino IDE 2.3.7
File Edit Sketch Tools Help
Arduino UNO
sketch_feb12a.ino
1 void setup() {
2 // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7 // put your main code here, to run repeatedly:
8
9 }
10
```

Ln 1, Col 1 Arduino UNO on COM6

# Sketch (program) Structures

```
/*
Multiline comment 1
Multiline comment 2
*/
// End-of-line comment
// Define public variable
Int led = 13;
```

Header

```
void setup() {
    // initialize the digital pin as an output.
}
```

the setup routine  
runs once when you press reset

```
void loop() {
    // The main code
}
```

the loop routine  
runs over and over again forever

# Blink.ino

```
void setup() {
    pinMode(13,OUTPUT);
}

void loop() {
    digitalWrite(13,HIGH);
    delay(1000);
    digitalWrite(13,LOW);
    delay(1000);
}
```

# Blink.ino

```
int ledPin = 13;

void setup() {
    pinMode (ledPin,OUTPUT);
}

void loop() {
    digitalWrite (ledPin,HIGH);
    delay(1000);
    digitalWrite (ledPin,LOW);
    delay(1000);
}
```

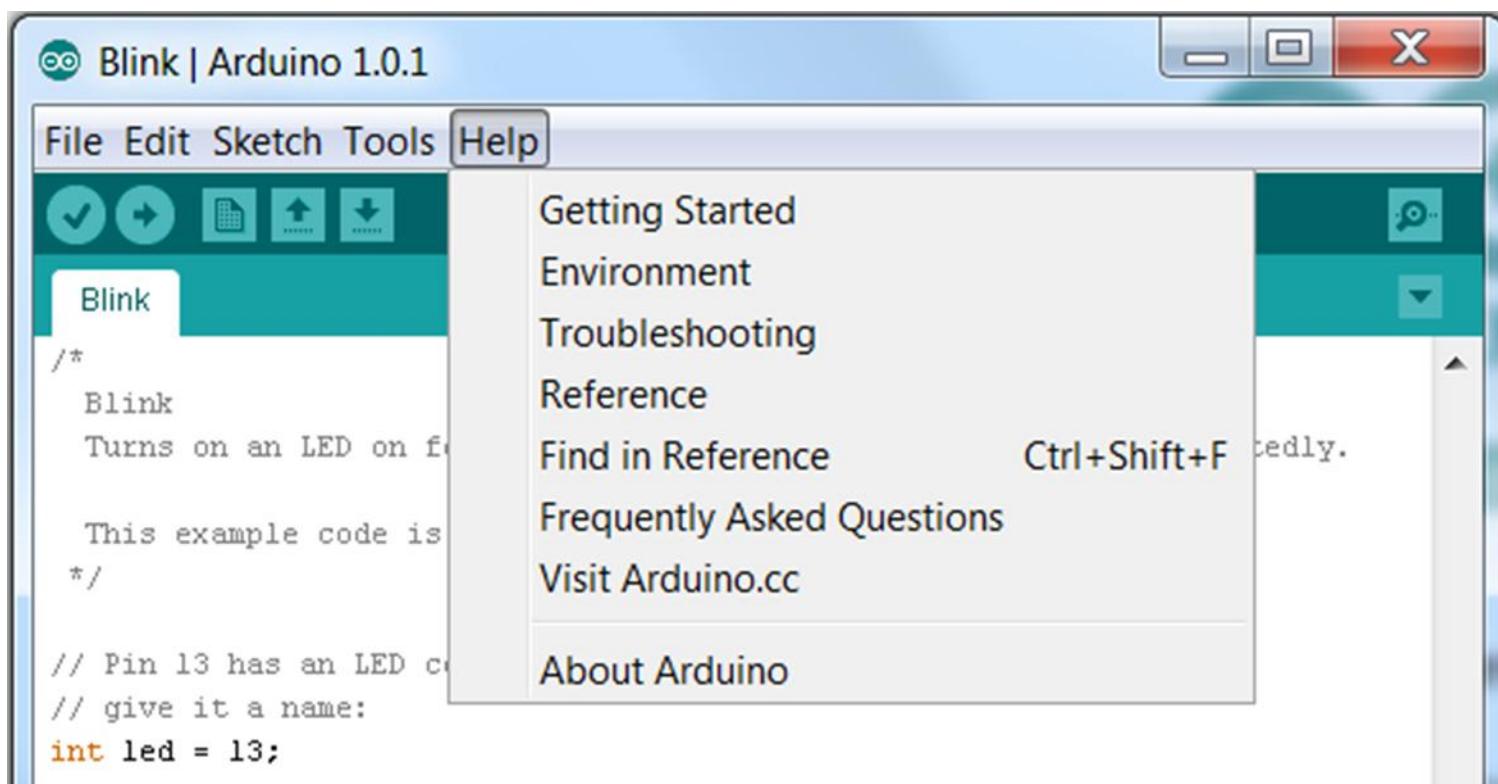
# Pre-installed Examples

- File → Examples → 01.Basics → Blink
- Verify/Compile
- Upload
- Modify
- Save



```
/*  
 *  Blink  
 *  
 *  Turns on an LED on for one second, then off for one second, repeatedly.  
  
 *  This example code is in the public domain.  
 */  
  
// Pin 13 has an LED connected on most Arduino boards.  
// give it a name:  
int led = 13;  
  
// the setup routine runs once when you press reset:  
void setup() {  
    // initialize the digital pin as an output.  
    pinMode(led, OUTPUT);  
}  
  
// the loop routine runs over and over again forever:  
void loop() {  
    digitalWrite(led, HIGH);    // turn the LED on (HIGH is the voltage level)  
    delay(1000);              // wait for a second  
    digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW  
    delay(1000);              // wait for a second  
}
```

# HELP !



# “Serial” interface

- Arduino UNO R3 board มี On-board serial port ทางขา digital pins 0 และ 1 (สั่งเกตุได้จากสัญลักษณ์ Rx และ Tx) เพื่อเชื่อมต่อกับ serial devices อื่นๆ นอกจากนี้ ทั้ง 2 pins ยังเชื่อมต่อกับ Serial/USB Emulator ทำให้ PC สามารถมองเห็นและเชื่อมต่อกับ Arduino board ทาง COM port ที่สร้างขึ้นผ่านสาย USB ได้  
ปกติ COM port นี้ถูกใช้ในการ upload/download โปรแกรมระหว่าง PC กับ board แต่เราสามารถใช้ Object “Serial” ในการรับส่งข้อมูลตามที่เราต้องการได้ เช่น กัน ฟังก์ชันพื้นฐานของ “Serial” ได้แก่
  - Serial.begin (baud) ใช้ในการกำหนดความเร็วในการรับส่งข้อมูล (baud คือ bit/second) ค่ามาตรฐานเริ่มต้นคือ 9600 ซึ่งสามารถเปลี่ยนไปใช้ค่ามาตรฐานอื่น ๆ ได้ในภายหลัง
  - Serial.print() ใช้ในการส่งข้อมูลในรูปของรหัส ASCII ออกทาง serial port ข้อมูลที่เป็น numeric จะถูกแปลงเป็น ASCII code text ก่อนส่งออก ต่อไปนี้เป็นตัวอย่างการใช้งาน

Serial.print(78)	→ "78"
Serial.print(1.23456)	→ "1.23"
Serial.println(1.23456, 0)	→ "1"
Serial.println(1.23456, 2)	→ "1.23"
Serial.println(1.23456, 4)	→ "1.2346"
Serial.print("Hello, World!")	→ "Hello world."
Serial.print(78, BIN)	→ "1001110"
Serial.print(78, DEC)	→ "78"
Serial.print(78, HEX)	→ "4E"

Serial.println() เมื่อ Serial.print() แต่จะขึ้นบรรทัดใหม่ให้ด้วย

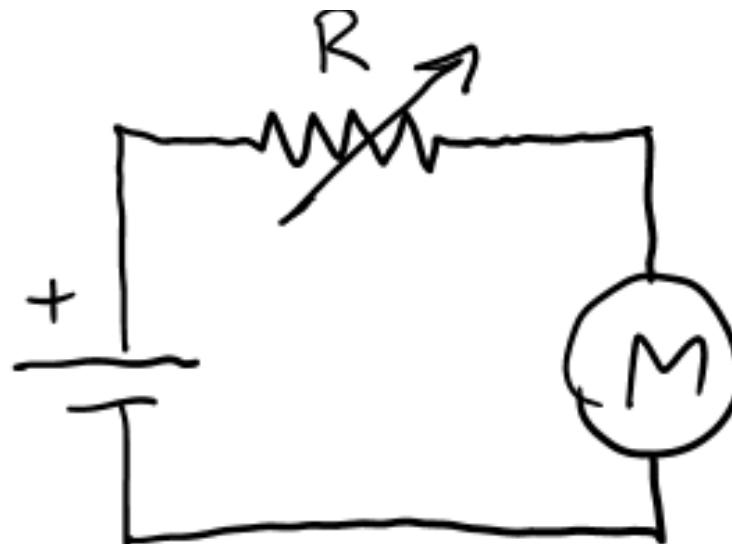
# Hello, World!

- ให้ทดลองเขียนโปรแกรมต่อไปนี้ แล้ว upload ลงใน Arduino

```
void setup() {  
    Serial.begin(9600);  
    Serial.println("Hello, World!");  
}  
  
void loop() {  
    Serial.println("I am Arduino. Nice to meet you.");  
    delay(1000);  
}
```

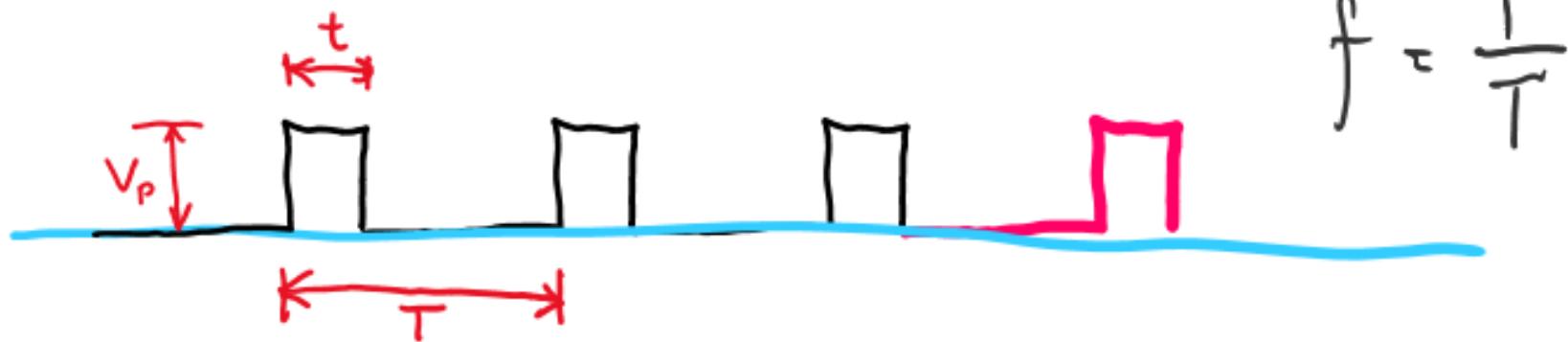
# Motor Control

## Basic Concept



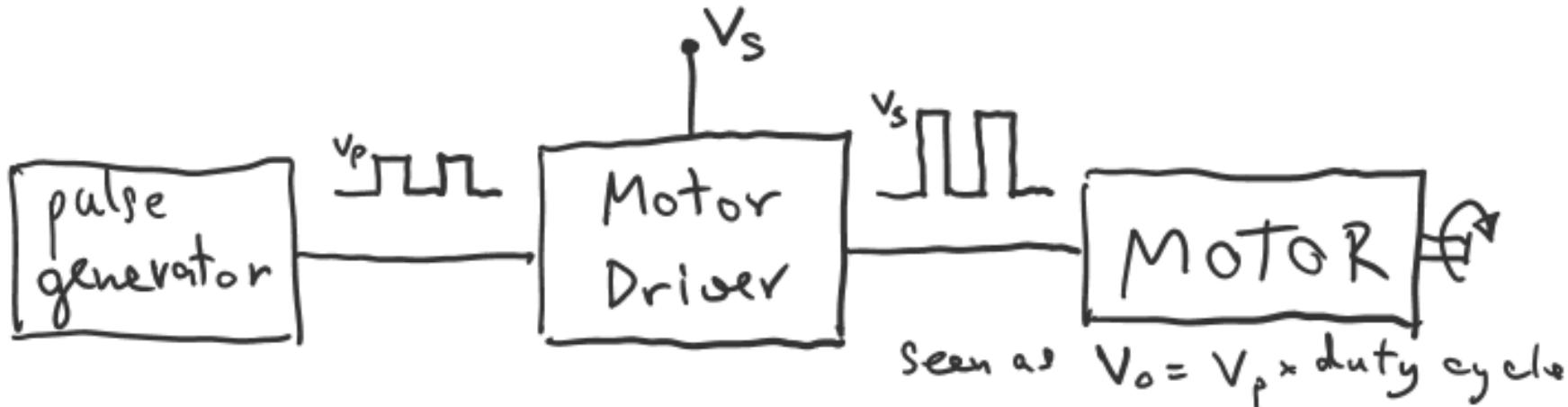
ความเร็วของมอเตอร์แปรผันตามแรงดันไฟฟ้า (Analog Voltage) ที่ป้อน แต่ไม่ง่ายในการใช้วงจรดิจิตอลสร้างแรงดันไฟฟ้าแบบ Analog ดังนั้นในไมโครคอนโทรลเลอร์จึงนิยมใช้สัญญาณแบบ Pulse-Width Modulation ในการขับ DC Motor

## Pulse-Width Modulation (PWM)



$$\text{Duty cycle} = \frac{t}{T} \times 100\%$$

เนื่องจากความถี่ ( $f = 1/T$ ) ของสัญญาณค่อนข้างสูง ทำให้มอเตอร์มีเสียงดังของแรงดันเป็นแรงดันเฉลี่ยต่อ  
เวลา  $V_o = V_p \times \text{Duty cycle}$  เพื่อการควบคุมมอเตอร์

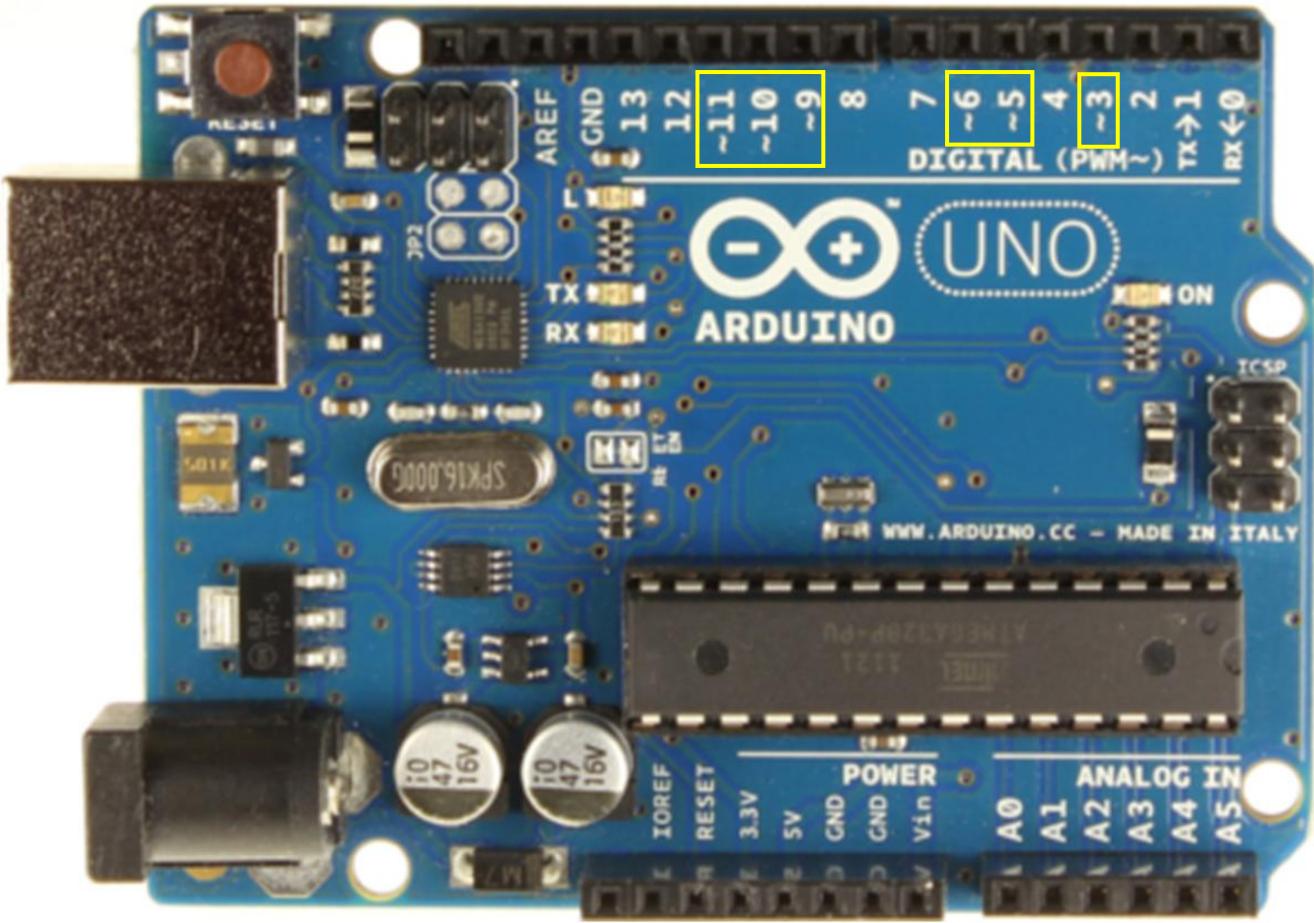


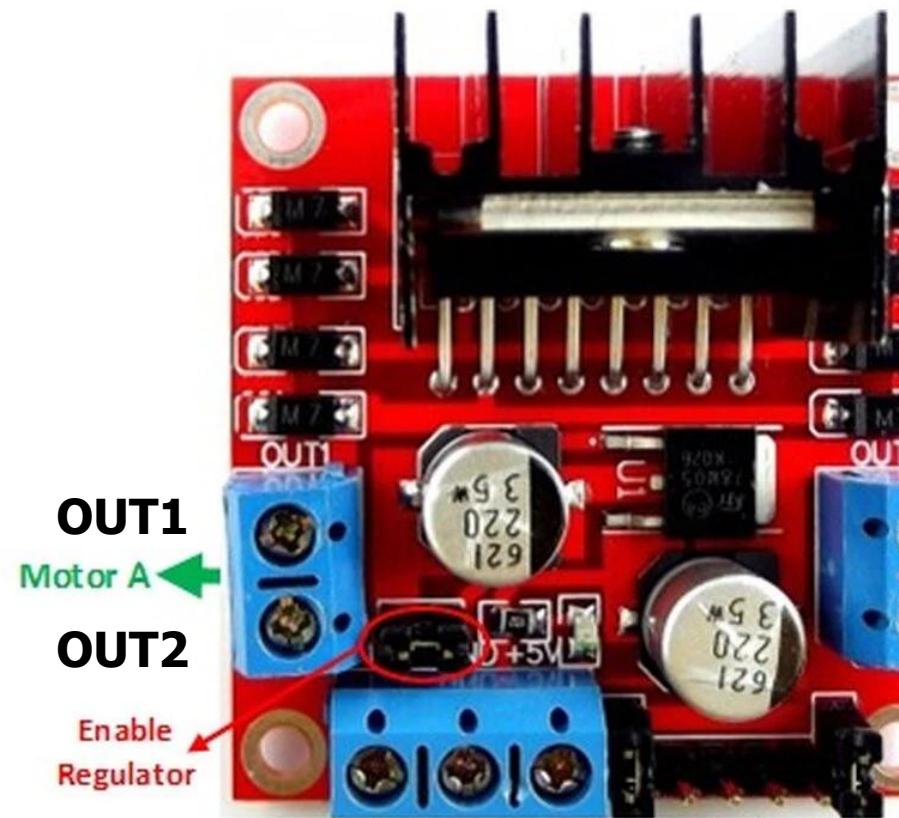
## Arduino PWM

Actual      0%      to      100%

Arduino      0      to      255      (8 bit)

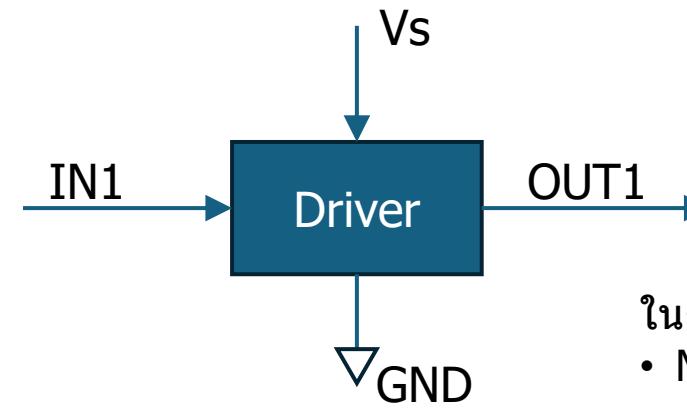
```
analogWrite(pin, value);  
// value is in 8-bit duty cycle (0-255)
```





ไฟเลี้ยงเพื่อขับมอเตอร์ต่อจากแหล่งจ่ายไฟภายนอกหรือขา Vin ของ UNO เพื่อให้มั่นใจว่า LOW = 0V  
(ถ้า UNO ต่อแหล่งจ่ายไฟภายนอก)

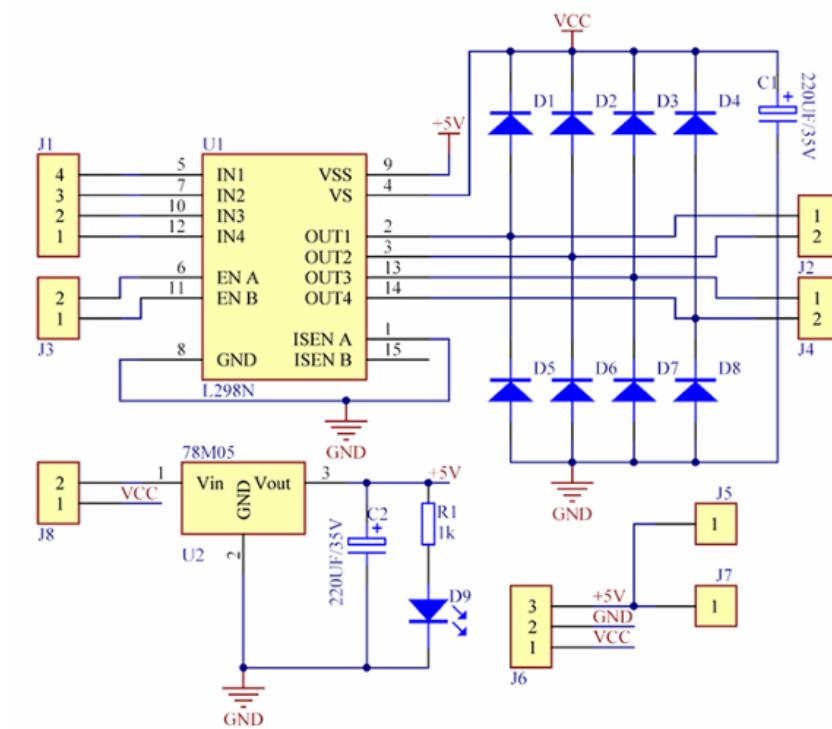
**Regulated 5V**  
(สามารถใช้ขั้นบอร์ด UNO ผ่าน Vin Pin)



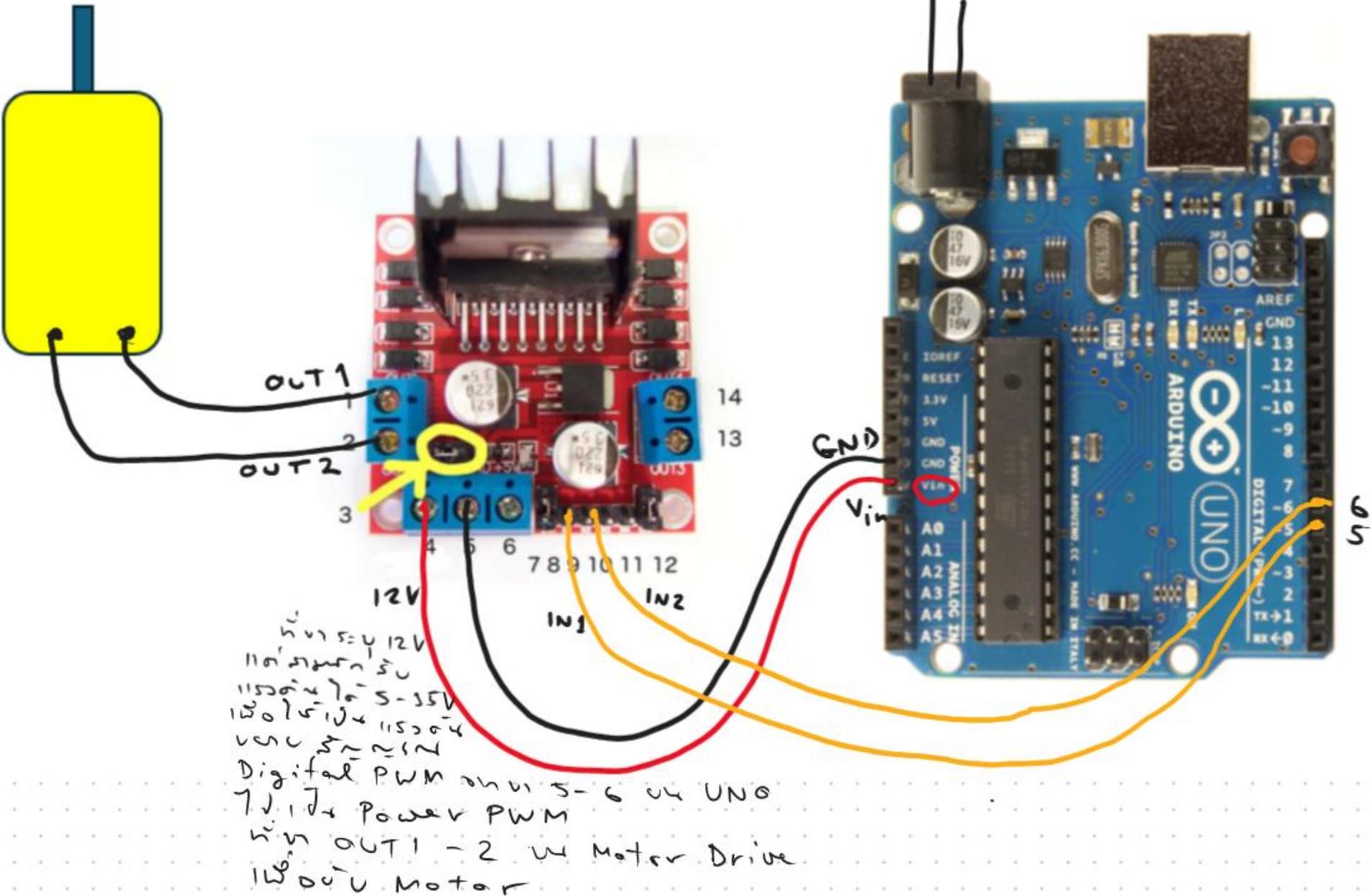
IN	OUT
Low (0)	0
High (5V)	Vs

ในการขับมอเตอร์ มักนิยมจับคู่

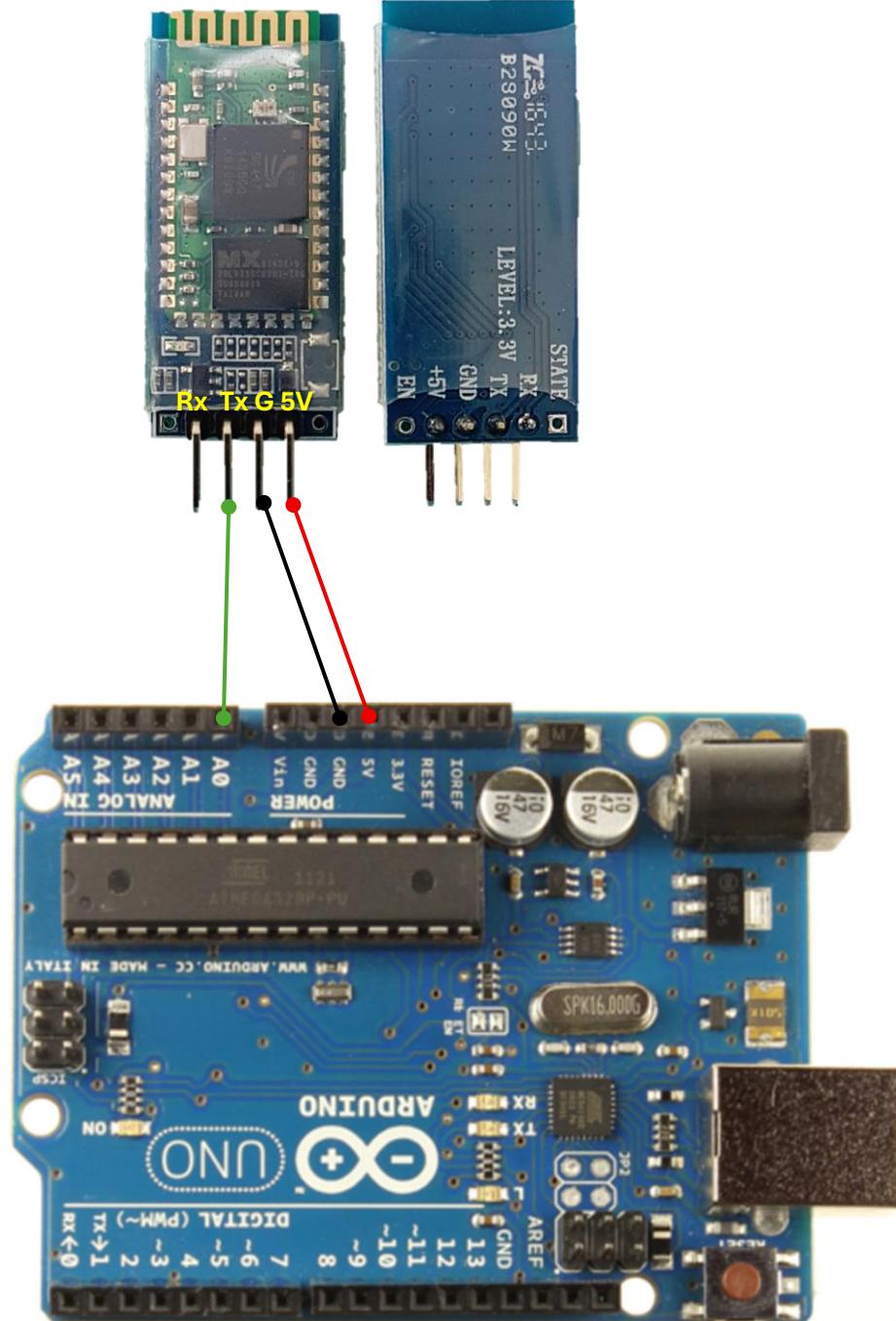
- Motor A
  - IN1/IN2 กับ OUT1/OUT2
- Motor B
  - IN3/IN4 กับ OUT3/OUT4



[https://github.com/vasu-kmitl/arduino\\_robot](https://github.com/vasu-kmitl/arduino_robot)



- โปรแกรมควบคุม motor อ่ายง่าย
  - มอเตอร์เดียว หมุนไปกลับ
  - [https://github.com/vasu-kmitl/arduino\\_robot/blob/main/oneMotor\\_basic/oneMotor\\_basic.ino](https://github.com/vasu-kmitl/arduino_robot/blob/main/oneMotor_basic/oneMotor_basic.ino)
- โปรแกรมควบคุม motor ด้วย PWM
  - ค่อยๆ เพิ่มความเร็ว ทึ้งไปและกลับ
  - ใช้ analogWrite แทน digitalWrite
  - [https://github.com/vasu-kmitl/arduino\\_robot/blob/main/oneMotor\\_PWM/oneMotor\\_PWM.ino](https://github.com/vasu-kmitl/arduino_robot/blob/main/oneMotor_PWM/oneMotor_PWM.ino)
- โปรแกรมควบคุม motor / 2 ตัวอ่ายง่าย
  - [https://github.com/vasu-kmitl/arduino\\_robot/tree/main/twoMotor\\_basic](https://github.com/vasu-kmitl/arduino_robot/tree/main/twoMotor_basic)
- เพิ่มการจัดการการเคลื่อนที่ด้วย function
  - [https://github.com/vasu-kmitl/arduino\\_robot/blob/main/twoMotor\\_function1/twoMotor\\_function1.ino](https://github.com/vasu-kmitl/arduino_robot/blob/main/twoMotor_function1/twoMotor_function1.ino)
- เพิ่ม parameter ใน function เพื่อกำหนดทิศทางการเคลื่อนที่
  - [https://github.com/vasu-kmitl/arduino\\_robot/blob/main/twoMotor\\_function2/twoMotor\\_function2.ino](https://github.com/vasu-kmitl/arduino_robot/blob/main/twoMotor_function2/twoMotor_function2.ino)



## HC-06 JDY Bluetooth Module

<https://www.arduino.com/product/4993>

<https://www.martyncurrey.com/hc-06-zg-b23090w-bluetooth-2-0-edr-modules/>

- เพื่อป้องกันความสับสนระหว่างกลุ่ม เนื่องจากแต่ละโมดูล มีชื่อเหมือนกัน ให้แต่ละกลุ่มแยกไปเชื่อมต่อโมดูลเข้ากับ โทรศัพท์มือถือ โดยไม่ให้สัญญาณปะปนกัน
- โดยการเชื่อมต่อดังภาพ (ใช้คอมพิวเตอร์เพียงเพื่อการ จ่ายไฟให้กับ Arduino และ Bluetooth Module)

## เตรียม code นี้เพื่ออ่านค่าที่ส่งมาจากแอพ Bluetooth RC Controller

[https://github.com/vasu-kmitl/arduino\\_robot/tree/main/bluetooth\\_reader](https://github.com/vasu-kmitl/arduino_robot/tree/main/bluetooth_reader)

```
//*****
// ROBOT SERIES: HC-06 Bluetooth Module Reader
//*****

#include <SoftwareSerial.h>
SoftwareSerial BTSerial(A0,A1); // A0(Assigned Rx) <-> HC06-Tx
                                // A1(Assigned Tx) <-> HC06-Rx

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

void loop() {
    if (BTSerial.available()){
        char c = BTSerial.read();
        Serial.write(c);
    }
}
```

15:21 3.21 LTE 5G 69% 15:21 3.21 LTE 5G 69% 15:22 3.33 LTE 5G 69% 15:22 3.33 LTE 5G 69%

< Bluetooth < Bluetooth < Bluetooth < Device details

เชื่อมต่อ Bluetooth Module เข้ากับ โทรศัพท์ (เฉพาะ Android เนื่องจากโมดูลเป็น Classic BT)

\* Amazfit Neo

\* C0:00:00:00:A9:E4

\* PR BT A6AD

SD1000v2.0.8-574B65

\* Surveyor Rover-9E56

Available devices

□ phol27

ค้นหา และ Pair BT Module

\* JDY-31-SPP

Phone's Bluetooth address: 3C:A2:C3:D7:BE:3F

When Bluetooth is turned on, your device can communicate with other nearby Bluetooth devices

Pair with JDY-31-SPP?  
ป้อน Pairing Password  
1234  
Usually 0000 or 1234  
 PIN contains letters or symbols  
You may also need to type this PIN on the other device.  
 Allow access to your contacts and call history  
Cancel OK

Surveyor Rover-9E56

1 2 3 -  
4 5 6 -  
7 8 9   
, 0 .

< Bluetooth < Bluetooth < Bluetooth < Device details

LENVO-VU

\* Amazfit Neo

\* C0:00:00:00:A9:E4

\* PR BT A6AD

SD1000v2.0.8-574B65

\* Surveyor Rover-9E56

\* JDY-31-SPP

JDY-31-SPP  
Not connected

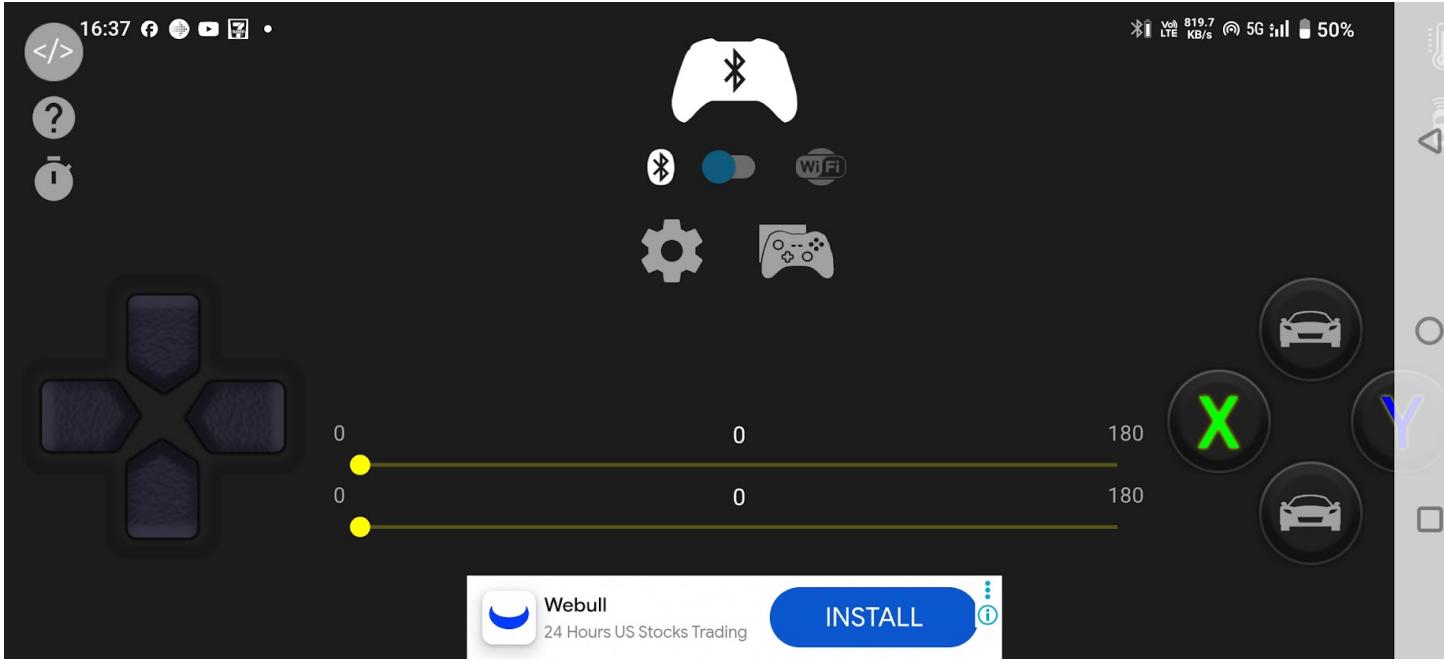
Forget Connect

Allow access to contacts and call history  
Info will be used for call announcements and more

Device's Bluetooth address: 85:EF:09:01:65:00

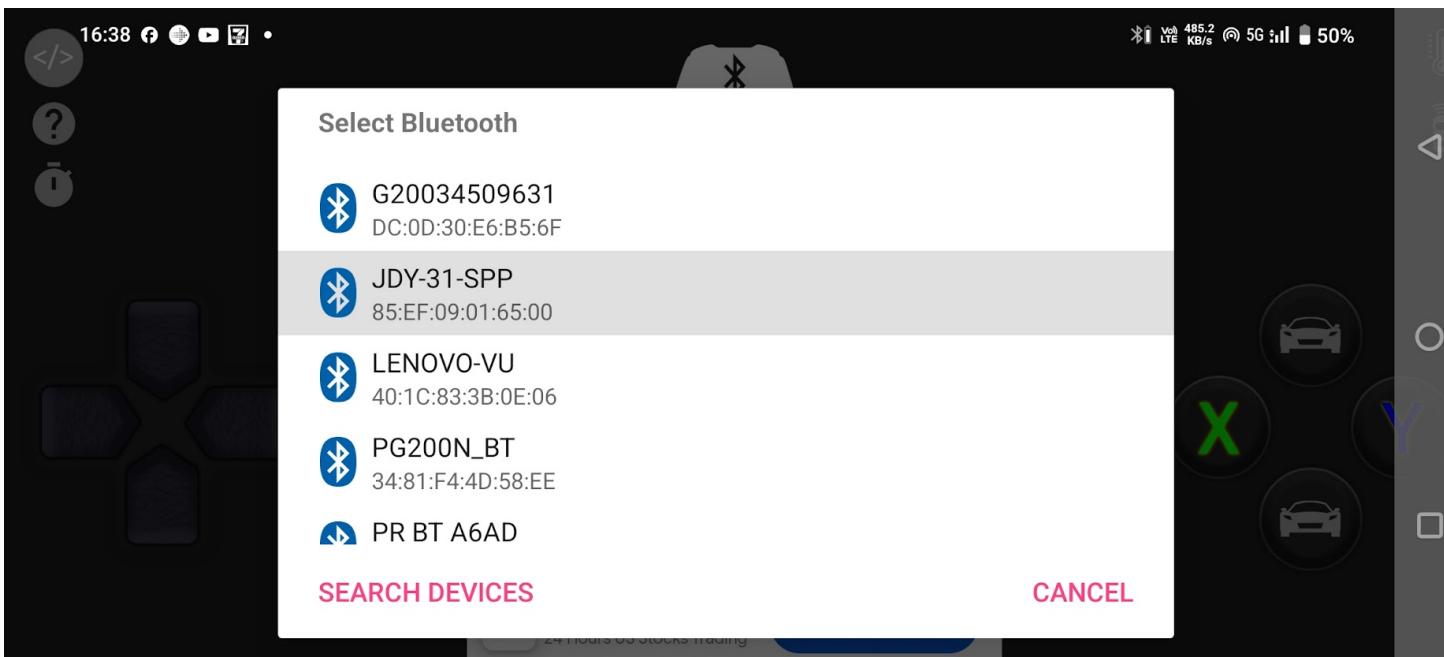
ตรวจสอบโมดูลที่ paired แล้ว

jdับบันทึก Address ของโมดูลเพื่อป้องกันการสับสนกับของเพื่อน



- หลังจาก pair ไมดูแลกับโทรศัพท์ให้เปิดแอพ Arduino Car

[https://play.google.com/store/apps/details?id=com.electro\\_tx.bluetoothcar](https://play.google.com/store/apps/details?id=com.electro_tx.bluetoothcar)



- ทำการ Connect กับโมดูล
- เลือกเชื่อมต่อกับโมดูลให้ถูกต้อง (สังเกต Address ด้วย อย่าไปเลือก โมดูลของกลุ่มอื่น)



Arduino UNO



```
//*****
// ROBOT SERIES: HC-06 Bluetooth Module Reader
//*****  

#include <SoftwareSerial.h>  

SoftwareSerial BTSerial(A0,A1); // A0(Assigned Rx) <-> HC06-Tx  

                           // A1(Assigned Tx) <-> HC06-Rx  

void setup() {  

    Serial.begin(9600);  

    BTSerial.begin(9600);  

}  

void loop() {  

    if (BTSerial.available()){  

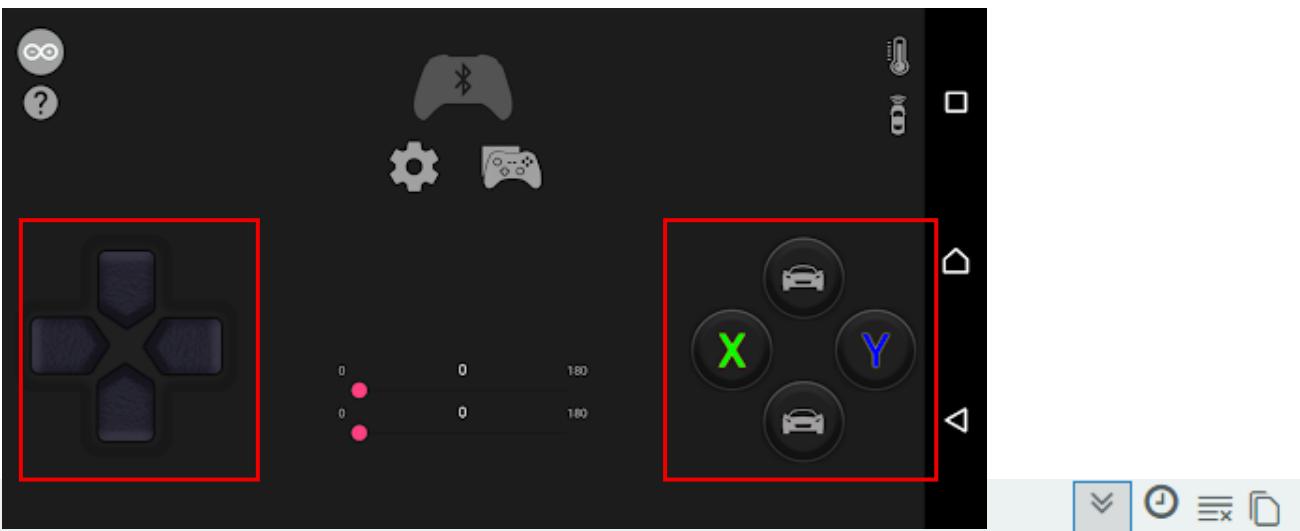
        char c = BTSerial.read();  

        Serial.write(c);  

    }  

}
```

กดเพื่อเปิดใช้งาน  
Serial Monitor



Serial Monitor × Output

Message (Enter to send message to 'Arduino UNO' on 'COM17')

No Line Ending

9600 baud

S  
F  
S  
F

หลังรันโปรแกรมให้กดปุ่มบนแอพ Arduino Car และดูผลในหน้าต่าง  
Serial Monitor (อาจต้องกดเปิดใช้งาน Serial Monitor ก่อน)

- จานนี้ให้ไปศึกษาการเขียน function เพื่อรับข้อมูลจาก bluetooth remote control เพื่อนำไปใช้ในการควบคุมมอเตอร์เพื่อขับเคลื่อนรถไปในทิศทางที่ต้องการ
- การใช้ PWM มาร่วมทำให้การหมุนตัวมีความนุ่มนวลและราบลื่นขึ้น