



77²³26

GARDEN WARS 2018

· PRECISION AGRICULTURE ·

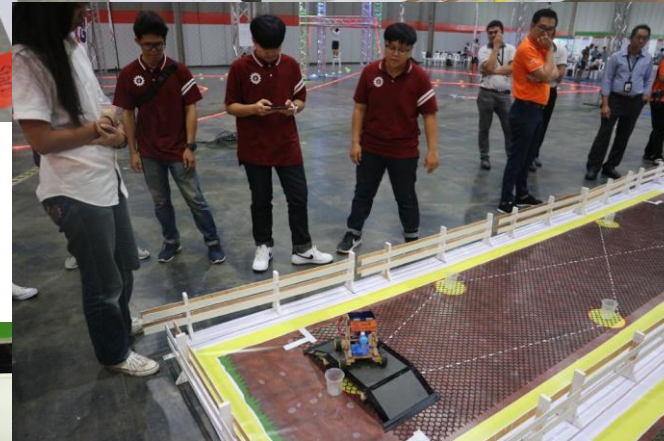
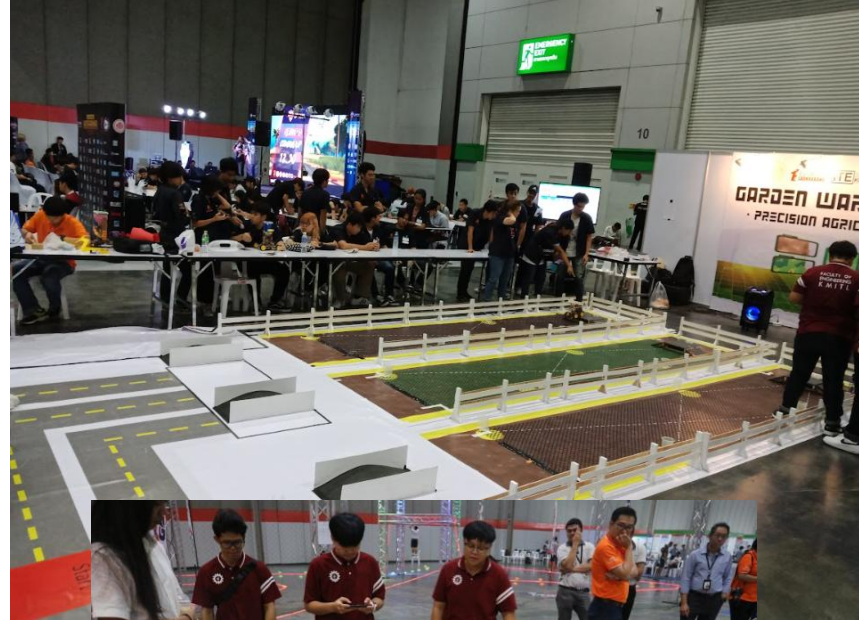






GARDEN WARS 2018

· PRECISION AGRICULTURE ·



Team project: Seed Drilling Robot

Garden Wars 2023

งานประกวดการแข่งหุ่นยนต์หอบอดเมล็ดของนักศึกษาชั้นปีที่ 1 หลักสูตร ASE
เมื่อ วันที่ 9 พฤศจิกายน 2566

ในวิชา Introduction to Agro-industrial Systems Engineering
ซึ่งมาในธีม รถหอบอดปุ๋ยที่บังคับด้วยรีโมทระยะไกล โดยกติกาเข้าแข่งขันดังนี้

1. หุ่นยนต์จะต้องหอบอดปุ๋ยในโซนเริ่มต้นก่อนเริ่มการแข่งขัน
2. หุ่นยนต์จะต้องหอบอดปุ๋ยตามจุดที่กำหนดไว้
3. ทีมผู้เข้าแข่งขันจะต้องมีผู้บังคับหุ่นยนต์รอบละ 1 คนเท่านั้น
4. นาฬิกาจะเริ่มทำการจับเวลาเมื่อเริ่มการแข่งขัน และหยุดเมื่อเสร็จสิ้นภารกิจ
5. เมื่อเริ่มแข่งขันจะไม่สามารถสัมผัสหุ่นยนต์ไม่ว่ากรณีใด ๆ
ขอแสดงความยินดีกับทีมผู้ชนะเลิศด้วยค่ะ



Shop 2 ตึกเรียน CCA



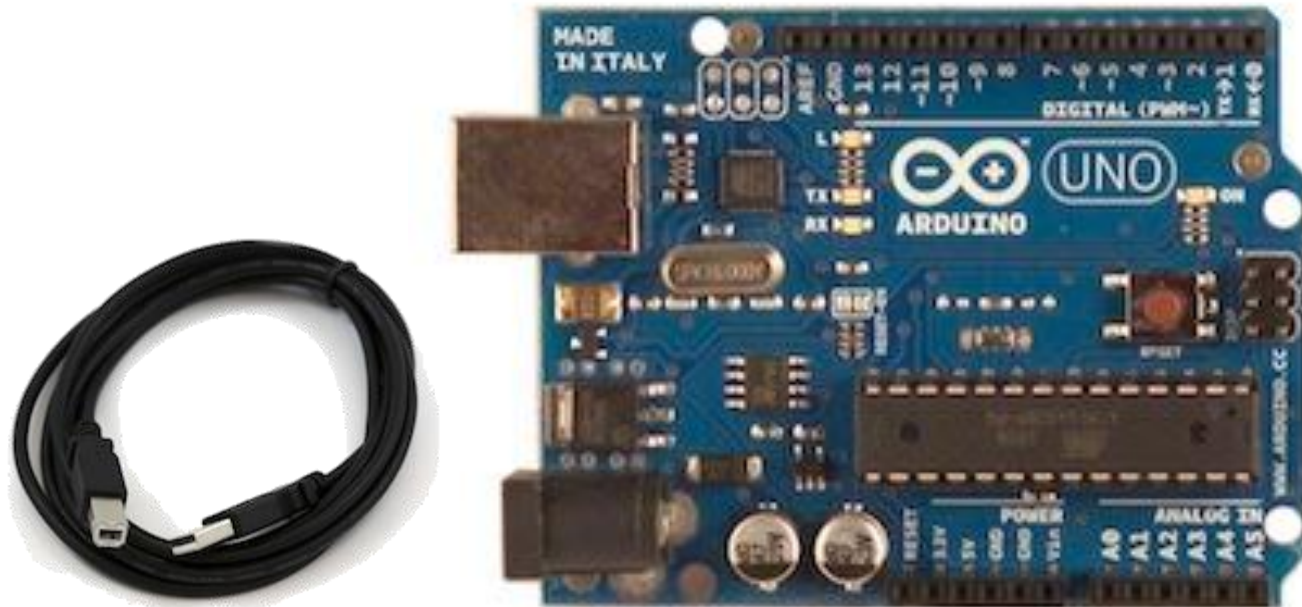


77²³26

GARDEN WARS 2018

· PRECISION AGRICULTURE ·





```
Blink | Arduino 1.0

/*
 * 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
}

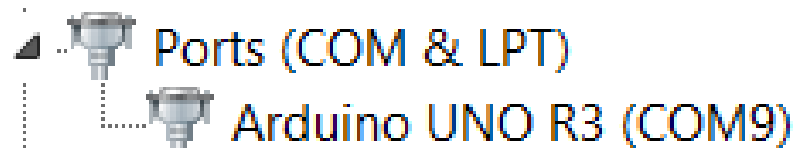
1 Arduino Uno on /dev/tty.usbmodemfd131
```

Basic Arduino

Installation

(Windows Only!)

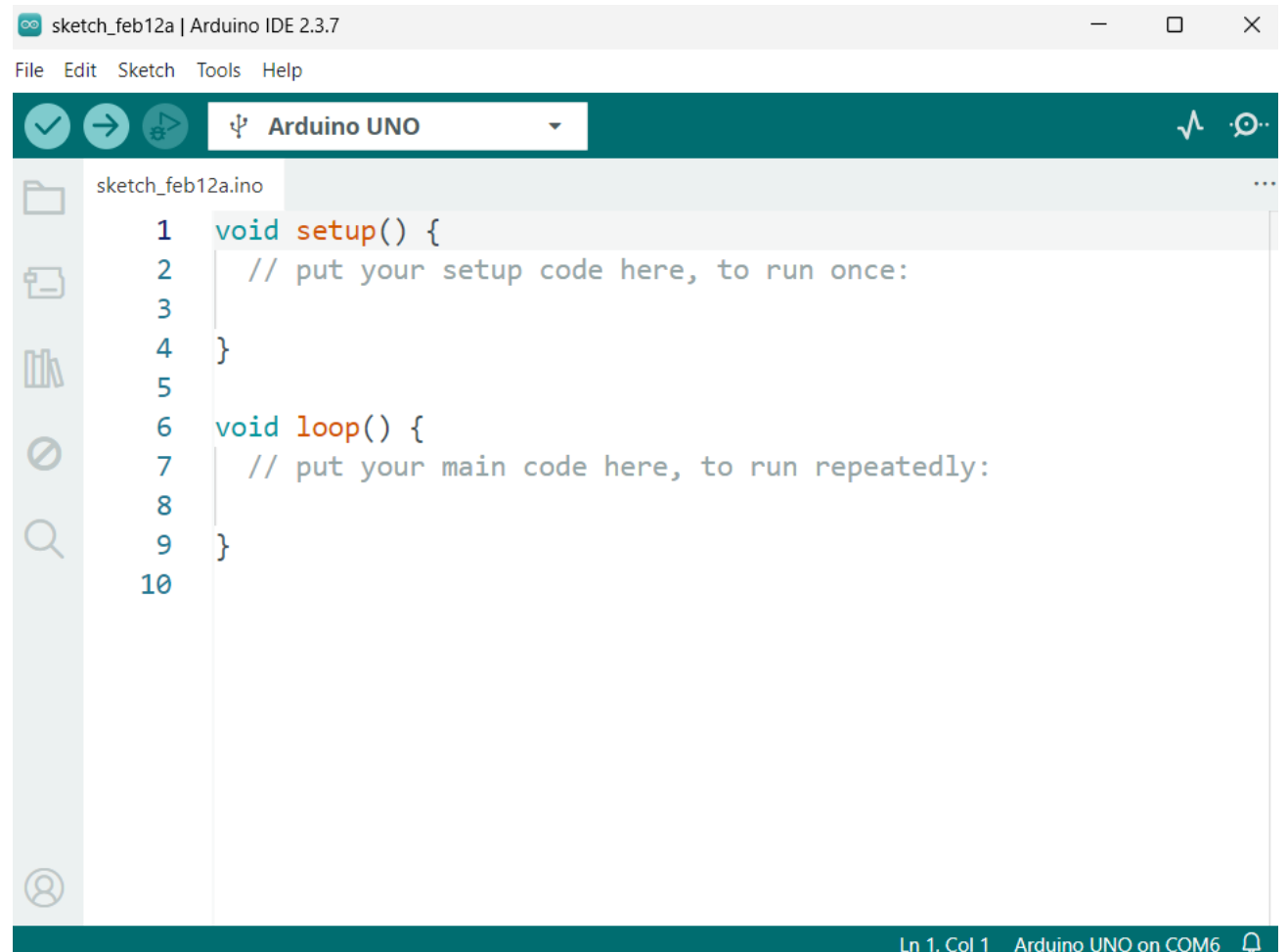
- Visit www.arduino.cc ; click on 'Download'
- Download and install 'Windows Installer'; Follow the instruction (check out all components including USB Driver)
- Connect Arduino board to the PC via a USB cable, wait until the driver installed.
 - Check if the "Arduino UNO R3 (COMxx)" exists under 'Ports(COM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

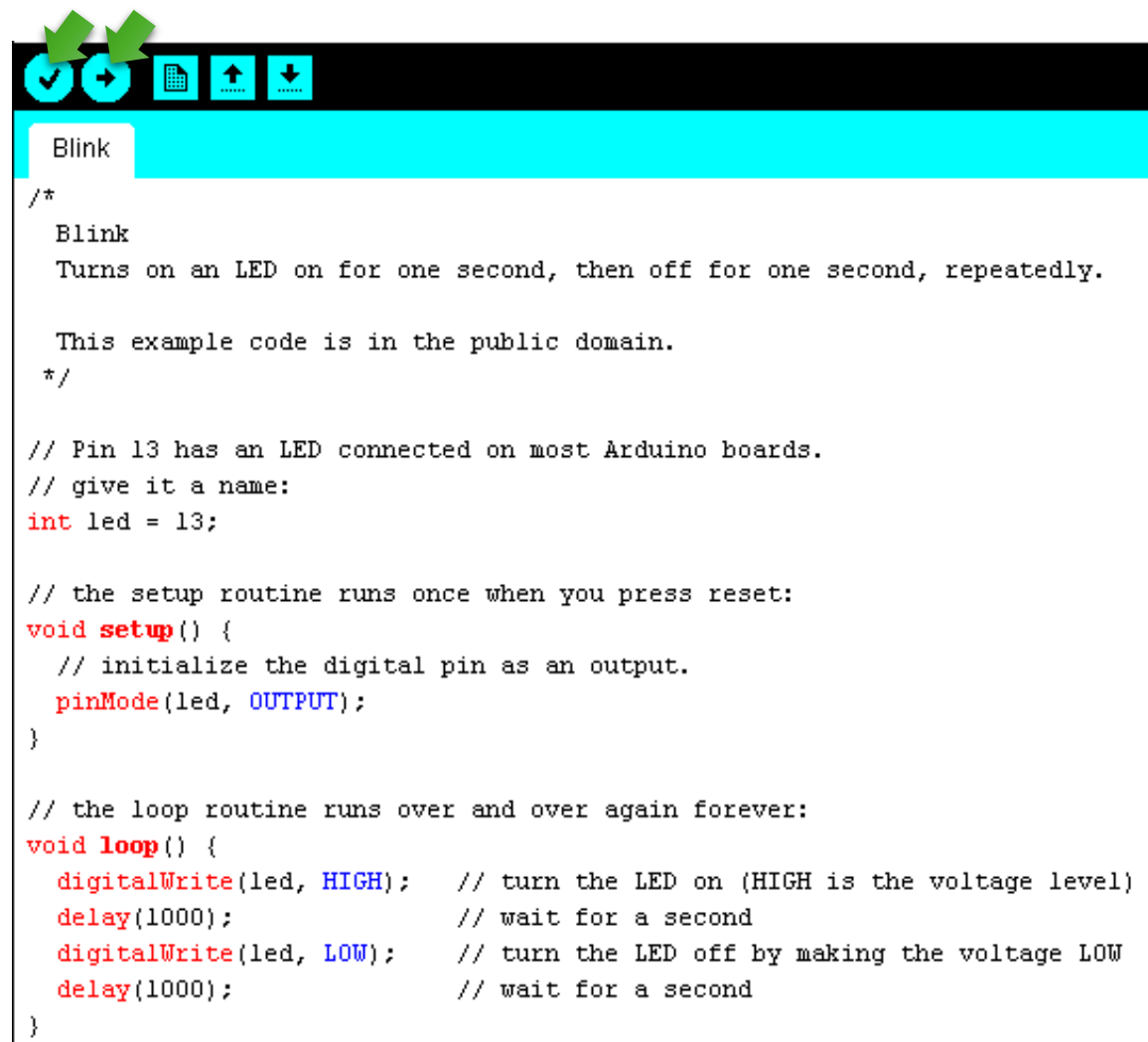
```
int ledPin = 13;

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.

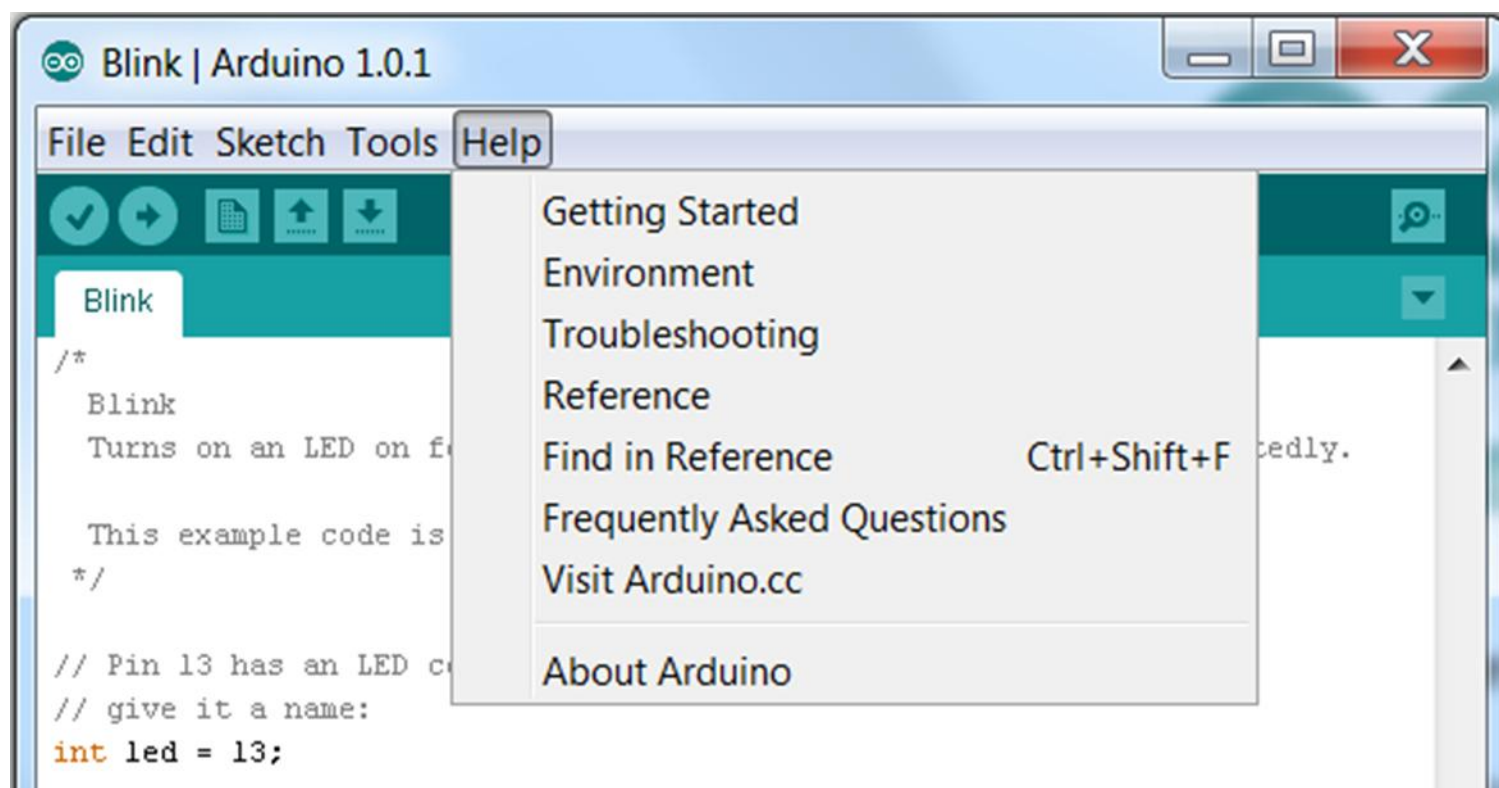
  This example code is in the public domain.
  */

// 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 (สังเกตได้จากสัญลักษณ์ 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 แต่เราก็สามารถใช้ Object “Serial” ในการรับส่งข้อมูลตามที่เราต้องการได้เช่นกัน
- ฟังก์ชันพื้นฐานของ “Serial” ได้แก่
 - Serial.begin (baud) ใช้ในการกำหนดความเร็วในการรับส่งข้อมูล (baud คือ bit/second) ค่ามาตรฐานเริ่มต้นคือ 9600 ซึ่งสามารถเปลี่ยนไปใช้ค่ามาตรฐานอื่น ๆ ได้ในภายหลัง
 - Serial.print() ใช้ในการส่งข้อมูลในรูปแบบของรหัส ASCII ออกทาง serial port ข้อมูลที่เป็น numeric จะถูกแปลงเป็น ASCII code text ก่อนส่งออก ต่อไปนี้เป็นตัวอย่างการใช้งาน

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

`Serial.println()` เหมือน `Serial.print()` แต่จะขึ้นบรรทัดใหม่ให้ด้วย

Hello, World!

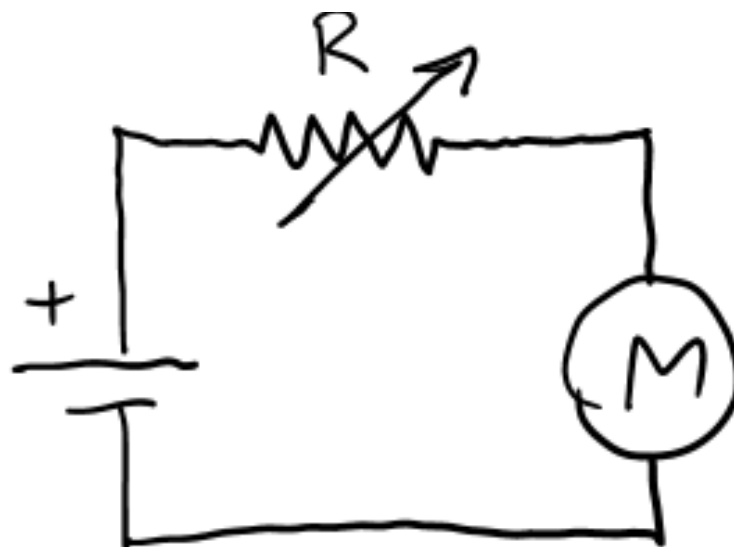
- ให้ทดลองเขียนโปรแกรมต่อไปนี้ แล้ว upload ลงใน Arduino

```
void setup() {  
    Serial.begin(9600);  
    Serial.println("Hello, World!");  
}
```

```
void loop() {  
    Serial.println("I am Arduino. Nice to meet you.");  
    delay(1000);  
}
```

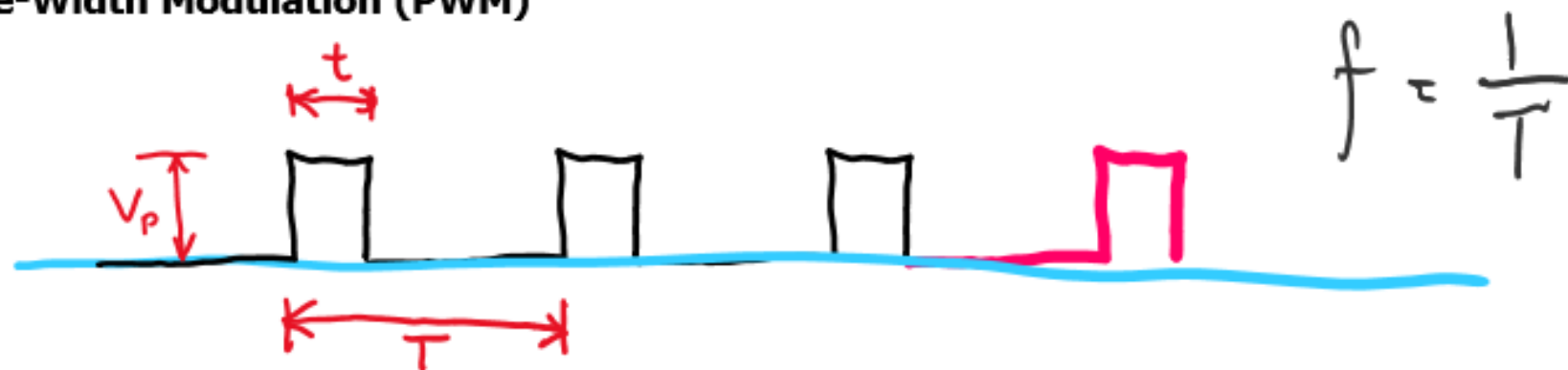

Motor Control

Basic Concept



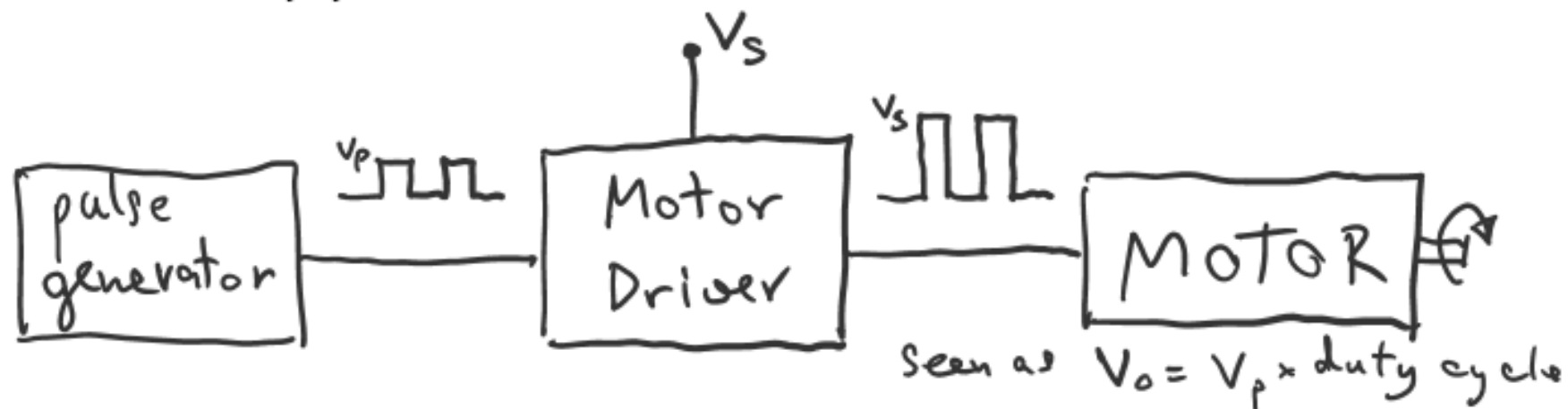
ความเร็วรอบมอเตอร์แปรผันตามแรงดันไฟฟ้า (Analog Voltage) ที่ป้อน แต่ไม่ง่ายในการใช้วงจรดิจิทัลสร้างแรงดันไฟฟ้าแบบ Analog ดังนั้นในไมโครคอนโทรลเลอร์จึงนิยมใช้สัญญาณแบบ Pulse-Width Modulation ในการขับ DC Motor

Pulse-Width Modulation (PWM)



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

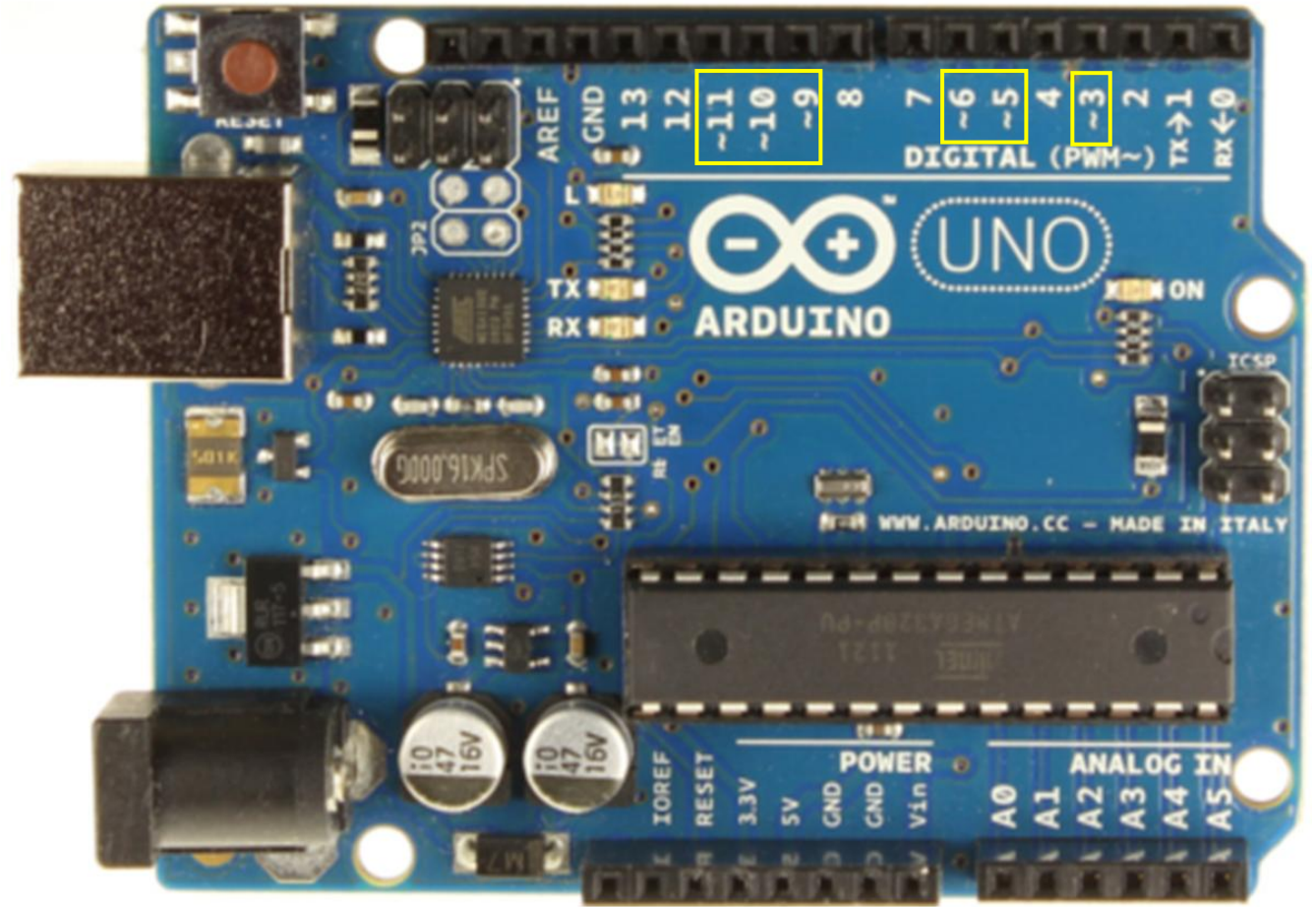
เนื่องจากความถี่ ($f = 1/T$) ของสัญญาณค่อนข้างสูง ทำให้มอเตอร์มองเห็นระดับของแรงดันเป็นแรงดันเฉลี่ยต่อคาบเวลา $V_o = V_p \times \text{Duty cycle}$ เพื่อการควบคุมมอเตอร์

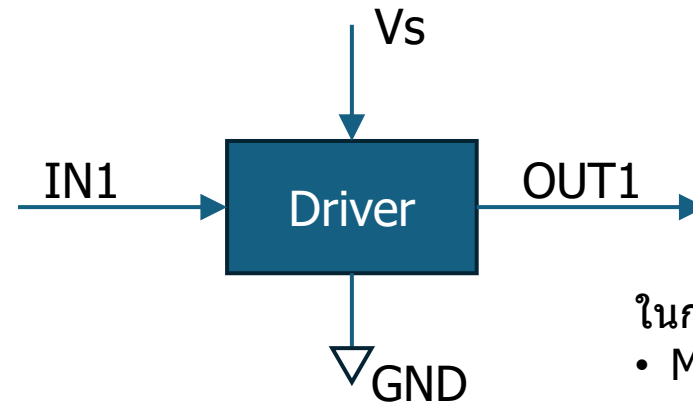
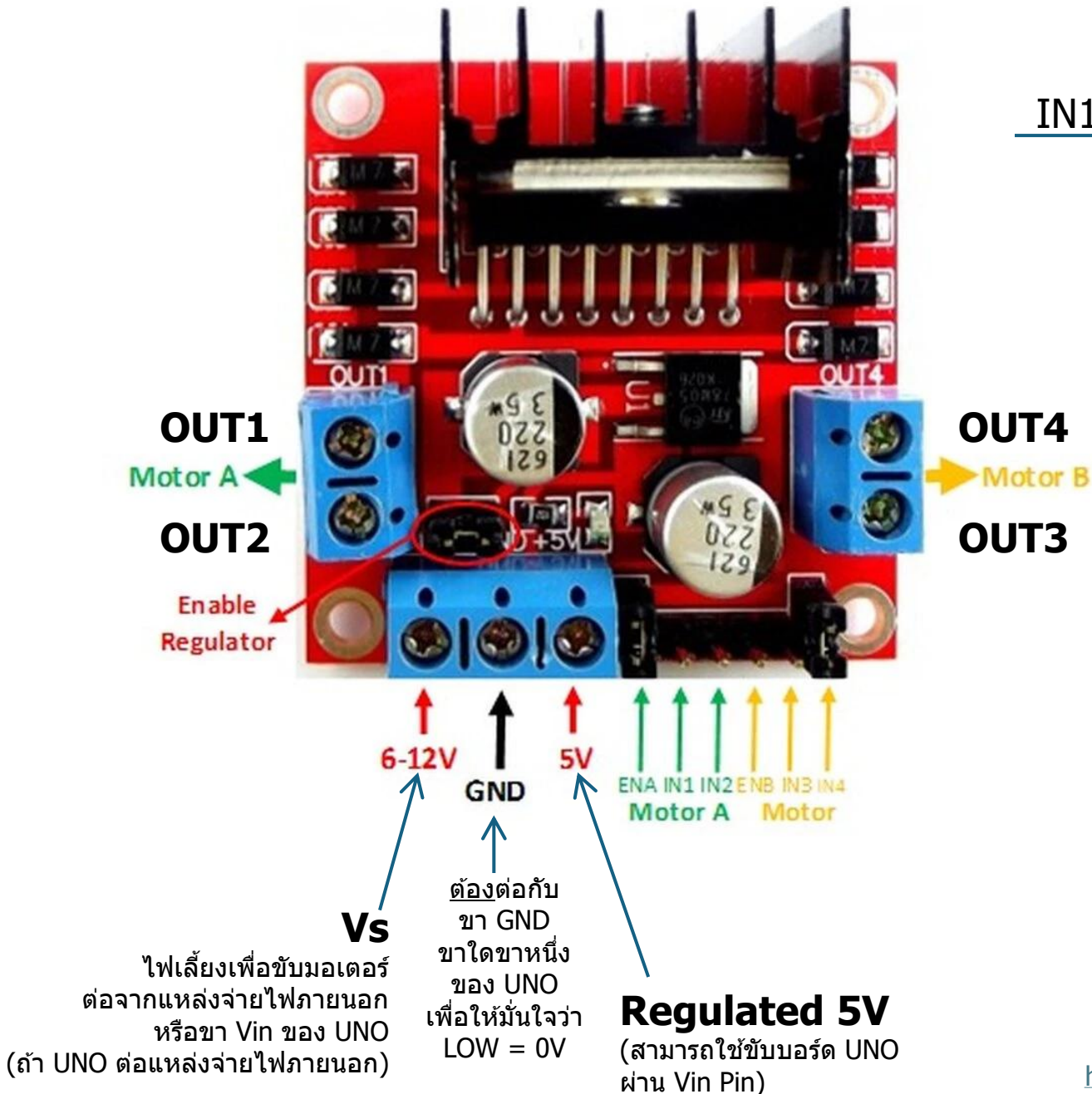


Arduino PWM

<u>Actual</u>	0%	to	100%
<u>Arduino</u>	0	to	255 (8 bit)

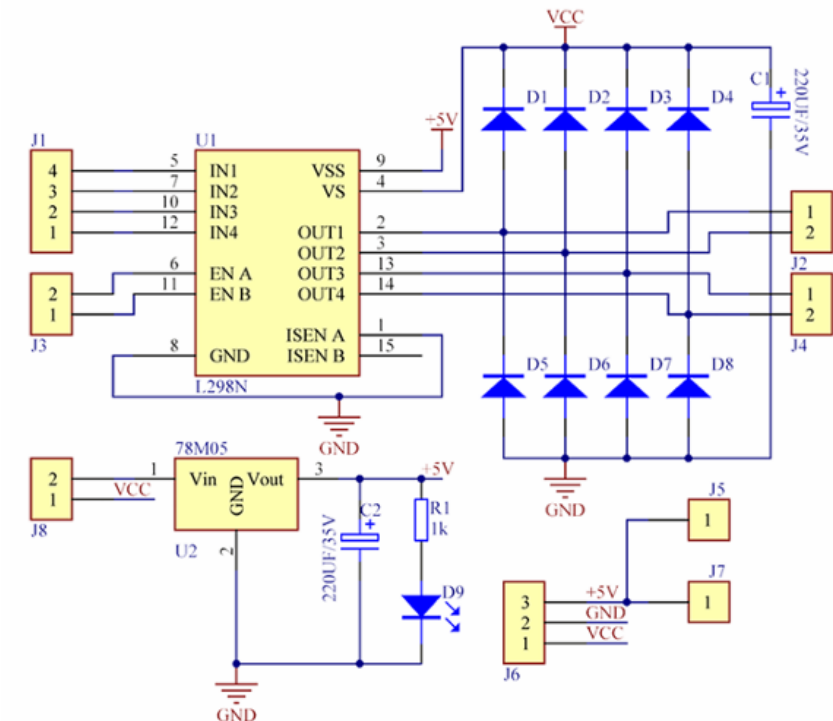
```
analogWrite(pin, value);  
// value is in 8-bit duty cycle (0-255)
```





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

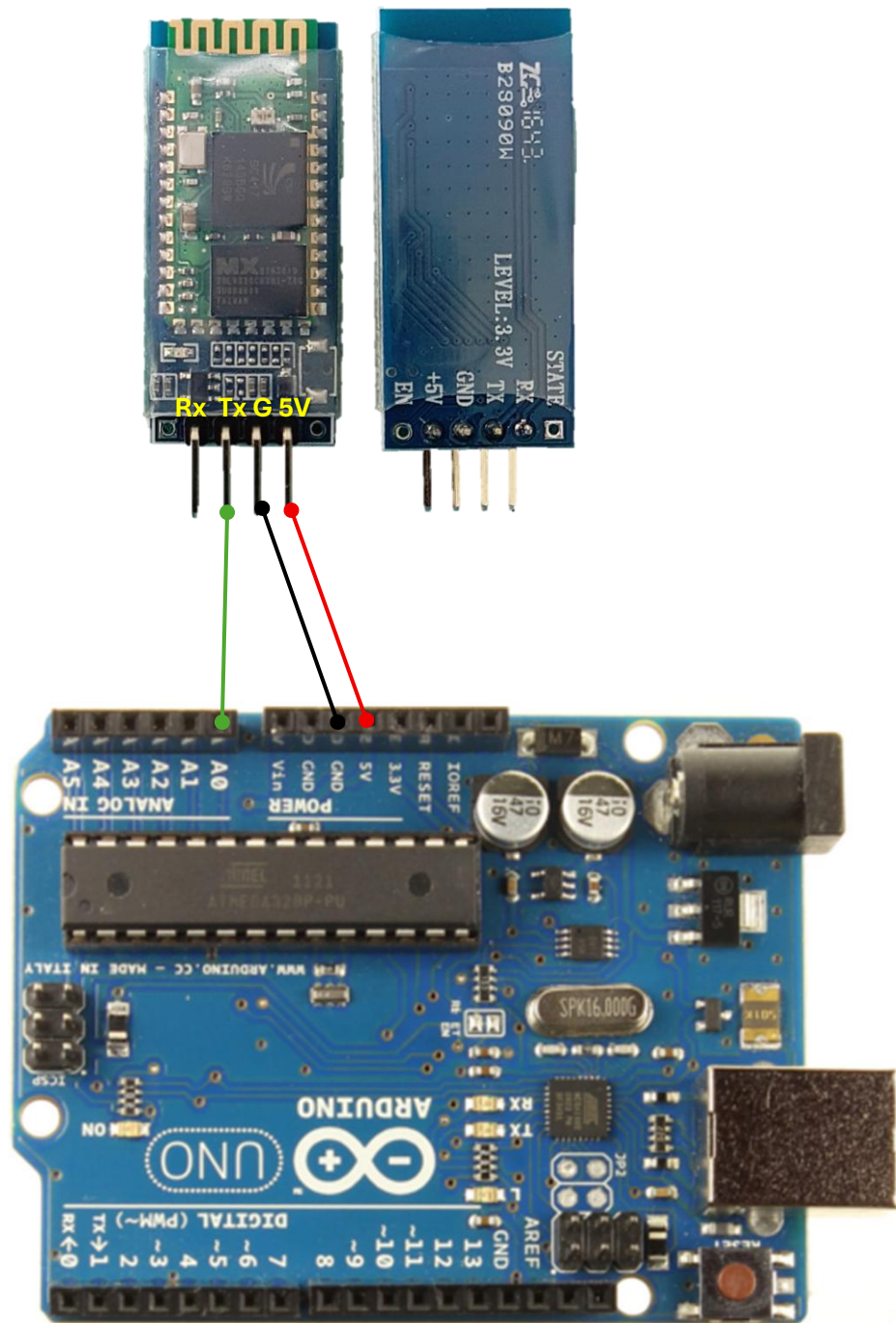
- ในการขับมอเตอร์ มักนิยมจับคู่
- Motor A
 - IN1/IN2 กับ OUT1/OUT2
 - Motor B
 - IN3/IN4 กับ OUT3/OUT4



https://github.com/vasu-kmitl/arduino_robot

110V 5-12V
 110V 5-35V
 110V 5-12V
 110V 5-12V
 Digital PWM 5-6 VV UNO
 110V Power PWM
 110V OUT1 - 2 VV Motor Drive
 110V Motor

- โปรแกรมควบคุม motor อย่างง่าย
 - มอเตอร์เดียว หมุนไปกลับ
 - https://github.com/vasu-kmitl/arduino_robot/blob/main/oneMotor_basic/oneMotor_basic.ino
- โปรแกรมควบคุม motor ด้วย PWM
 - ค่อยๆ เพิ่มความเร็ว ทั้งไปและกลับ
 - ใช้ analogWrite แทน digitalWrite
 - https://github.com/vasu-kmitl/arduino_robot/blob/main/oneMotor_PWM/oneMotor_PWM.ino
- โปรแกรมควบคุม motor / 2 ตัวอย่างง่าย
 - 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
- เพิ่ม parameter ใน function เพื่อกำหนดทิศทางการเคลื่อนที่
 - https://github.com/vasu-kmitl/arduino_robot/blob/main/twoMotor_function2/twoMotor_function2.ino



HC-06 JDY Bluetooth Module

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

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

- เพื่อป้องกันความสับสนระหว่างกลุ่ม เนื่องจากแต่ละโมดูลมีชื่อเหมือนกัน ให้แต่ละกลุ่มแยกไปเชื่อมต่อโมดูลเข้ากับโทรศัพท์มือถือ โดยไม่ให้สัญญาณปะปนกัน
- โดยการเชื่อมต่อดังภาพ (ใช้คอมพิวเตอร์เพียงเพื่อการจ่ายไฟให้กับ Arduino และ Bluetooth Module)

เตรียม code นี้เพื่ออ่านค่าที่ส่งมาจากแอฟ Bluetooth RC Controller

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);  
    }  
}
```

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

JDY-31-SPP

ค้นหา และ Pair BT Module

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**

1 2 3 -

4 5 6 _

7 8 9 ✕

, 0 . ✓

- LENOVO-VU
- Amazfit Neo
- C0:00:00:00:A9:E4
- PR BT A6AD
- SD1000v2.0.8-574B65
- Surveyor Rover-9E56
- JDY-31-SPP**

Available devices [Search for devices](#)

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

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

ตรวจสอบโมดูลที่ paired แล้ว

JDY-31-SPP

Not connected

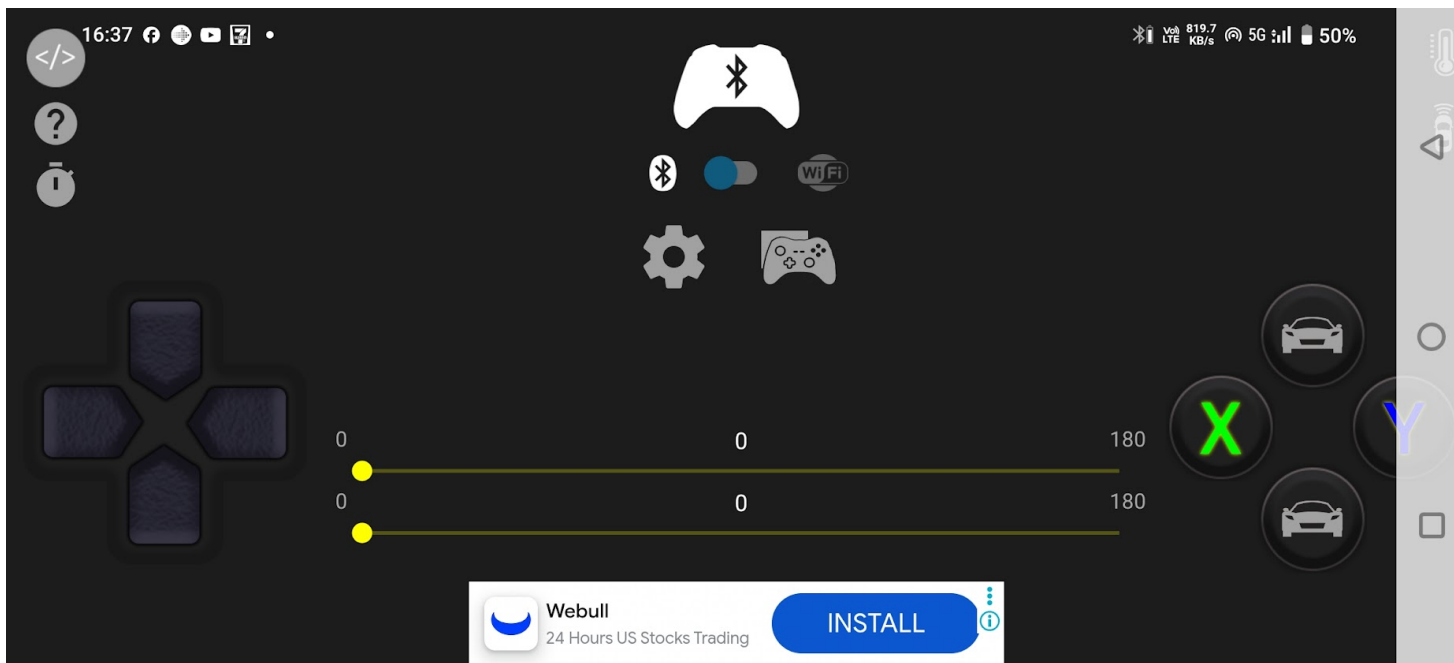
Forget Connect

Allow access to contacts and call history

Info will be used for call announcements and more

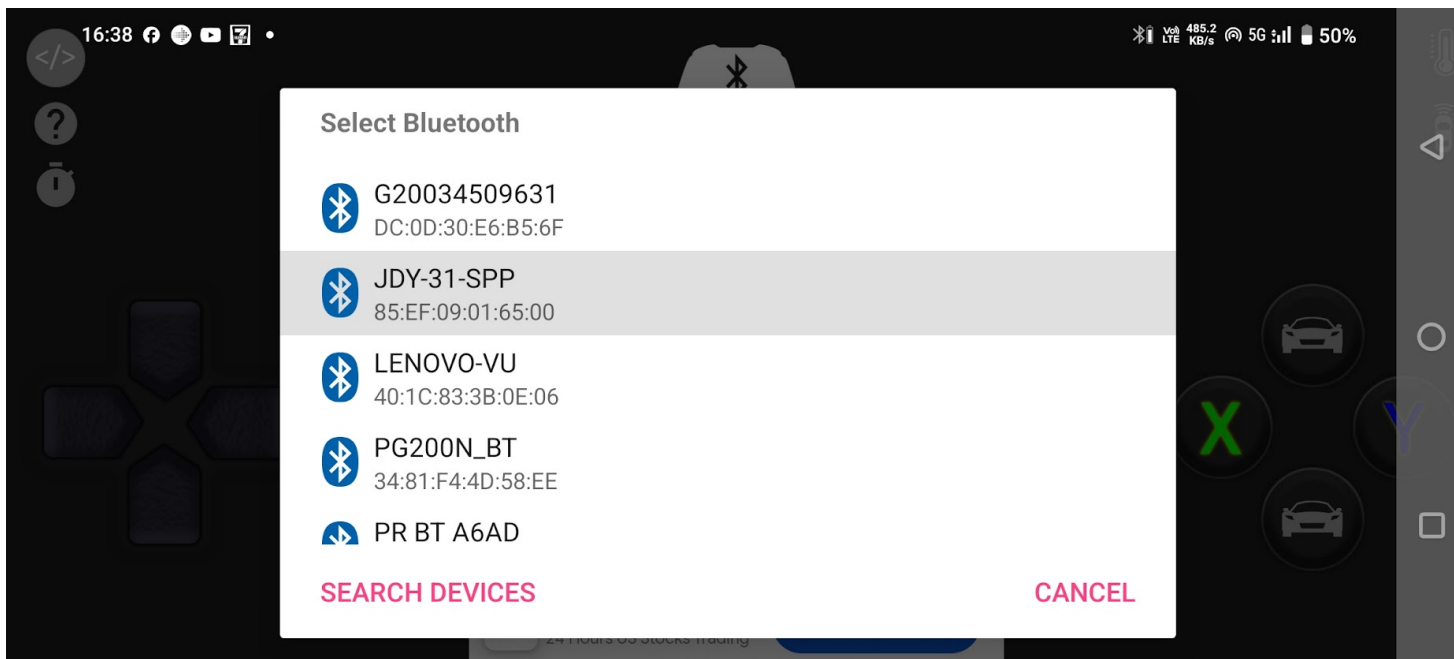
Device's Bluetooth address: **85:EF:09:01:65:00**

จดบันทึก Address ของโมดูล เพื่อป้องกันการสับสนกับของเพื่อน

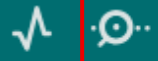


- หลังจาก pair โมดูลกับโทรศัพท์ ให้เปิดแอป Arduino Car

https://play.google.com/store/apps/details?id=com.electro_tex.bluetoothcar

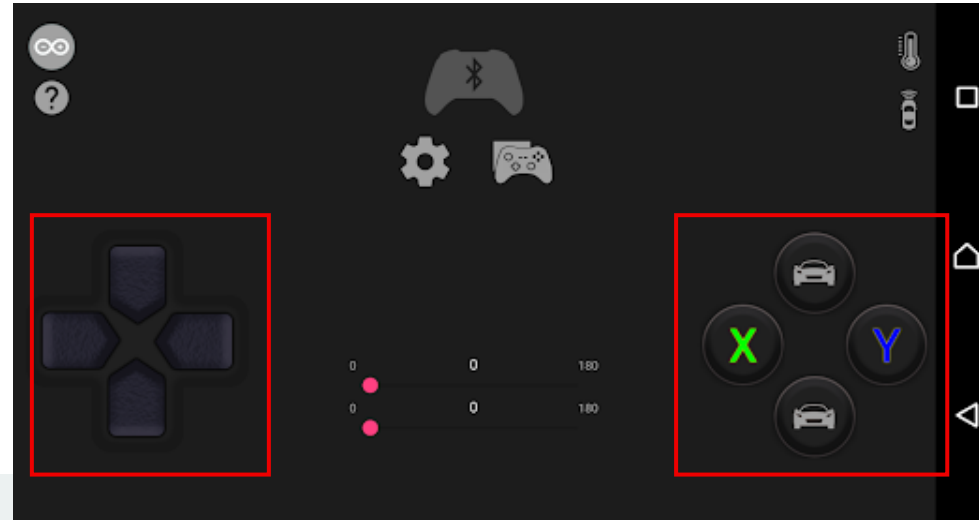


- ทำการ Connect กับโมดูล
- เลือกเชื่อมต่อกับโมดูลให้ถูกต้อง (สังเกต Address ด้วย อย่าไปเลือกโมดูลของกลุ่มอื่น)



กดเพื่อเปิดใช้งาน
Serial Monitor

```
//*****  
// ROBOT SERIES: HC-06 Bluetooth Module Reader  
//*****  
#include <SoftwareSerial.h>  
SoftwareSerial BTRSerial(A0,A1); // A0(Assigned Rx) <--> HC06-Tx  
// A1(Assigned Tx) <--> HC06-Rx  
  
void setup() {  
  Serial.begin(9600);  
  BTRSerial.begin(9600);  
}  
  
void loop() {  
  if (BTRSerial.available()){  
    char c = BTRSerial.read();  
    Serial.write(c);  
  }  
}
```



Serial Monitor X Output

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

No Line Ending

9600 baud

S
F
S
F

หลังรันโปรแกรมให้กดปุ่มบนแอป Arduino Car แล้วดูผลในหน้าต่าง
Serial Monitor (อาจต้องกดเปิดใช้งาน Serial Monitor ก่อน)

- จากนั้นให้ไปศึกษาการเขียน function เพื่อรับข้อมูลจาก bluetooth remote control เพื่อนำไปใช้ในการควบคุมมอเตอร์เพื่อขับเคลื่อนรถไปในทิศทางที่ต้องการ
- การใช้ PWM มาร่วมทำให้การหมุนตัวมีความนุ่มนวลและราบรื่นขึ้น