

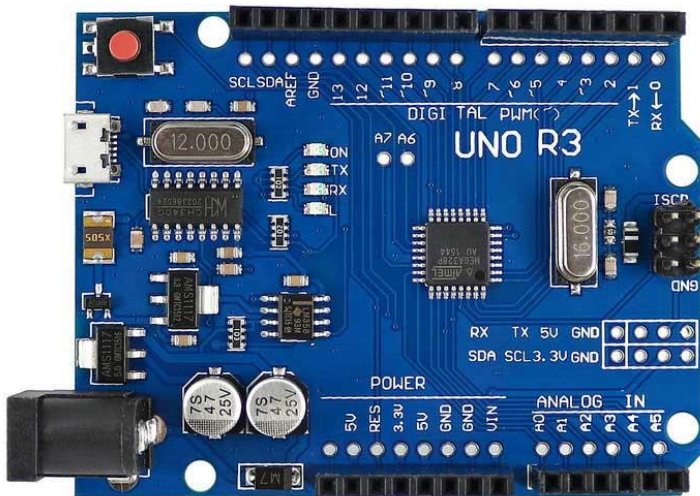
BASIC MCU PROGRAMING

- Variable
- Operator
- Loop
- Read Valve Form Sensors
- Output Control For Automation

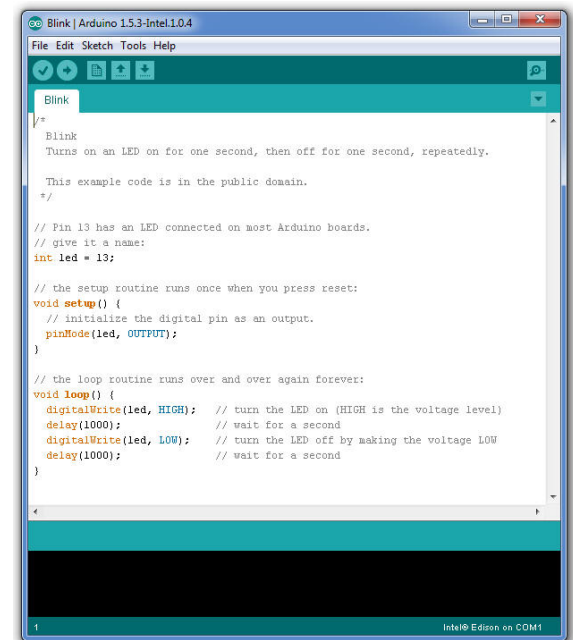




ARDUINO BOARD



ARDUINO IDE



ARDUINO HARDWARE



ARDUINO UNO



ARDUINO LEONARDO



ARDUINO 101



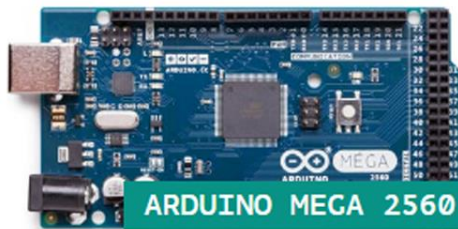
ARDUINO NANO



ARDUINO MINI



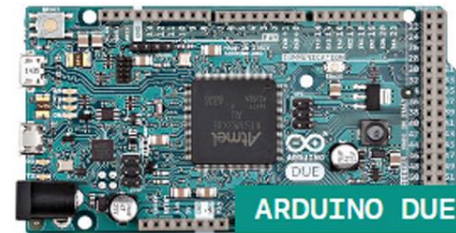
ARDUINO MICRO



ARDUINO MEGA 2560



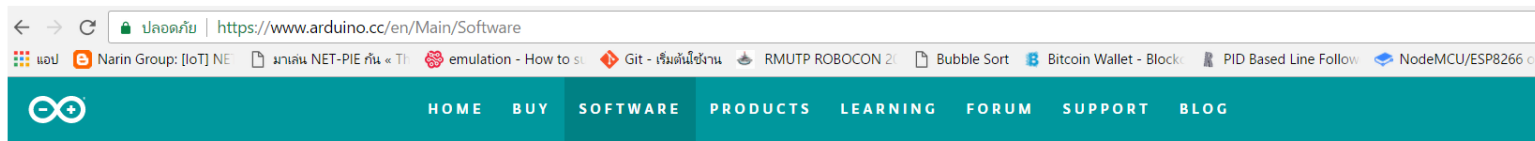
ARDUINO ZERO



ARDUINO DUE

ARDUINO IDE สามารถดาวน์โหลดได้จาก

www.arduino.cc



Download the Arduino IDE



ARDUINO 1.8.2

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

Windows Installer
Windows ZIP file for non admin install

Windows app [Get](#) 

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM

[Release Notes](#)
[Source Code](#)
[Checksums \(sha512\)](#)

ARDUINO SOFTWARE HOURLY BUILDS

LAST UPDATE
26 April 2017 18:13:6 GMT

Download a preview of the incoming release with the most updated features and bugfixes.

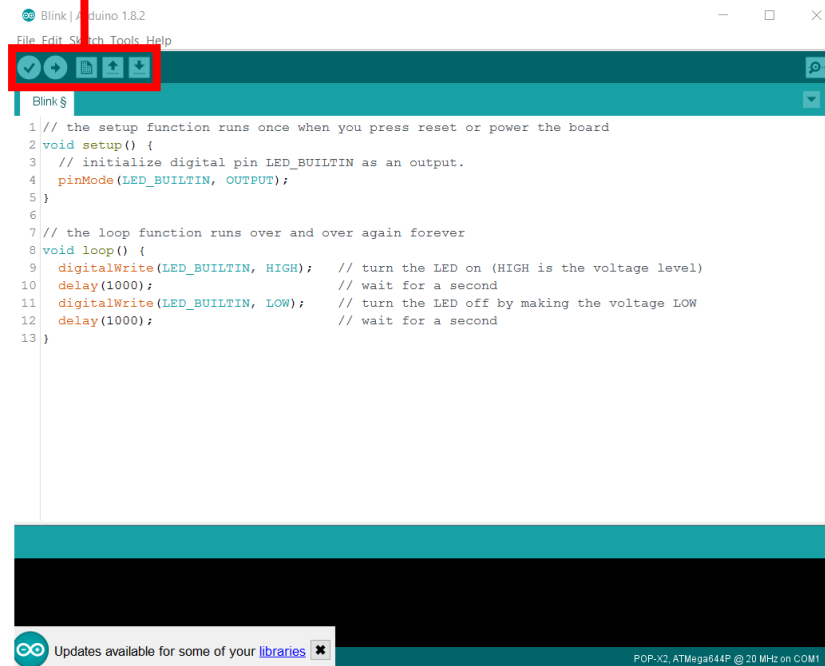
ARDUINO 1.0.6 / 1.5.x / 1.6.x PREVIOUS RELEASES

Download the [previous version of the current release](#), the classic [Arduino 1.0.x](#), or the [Arduino 1.5.x Beta version](#).

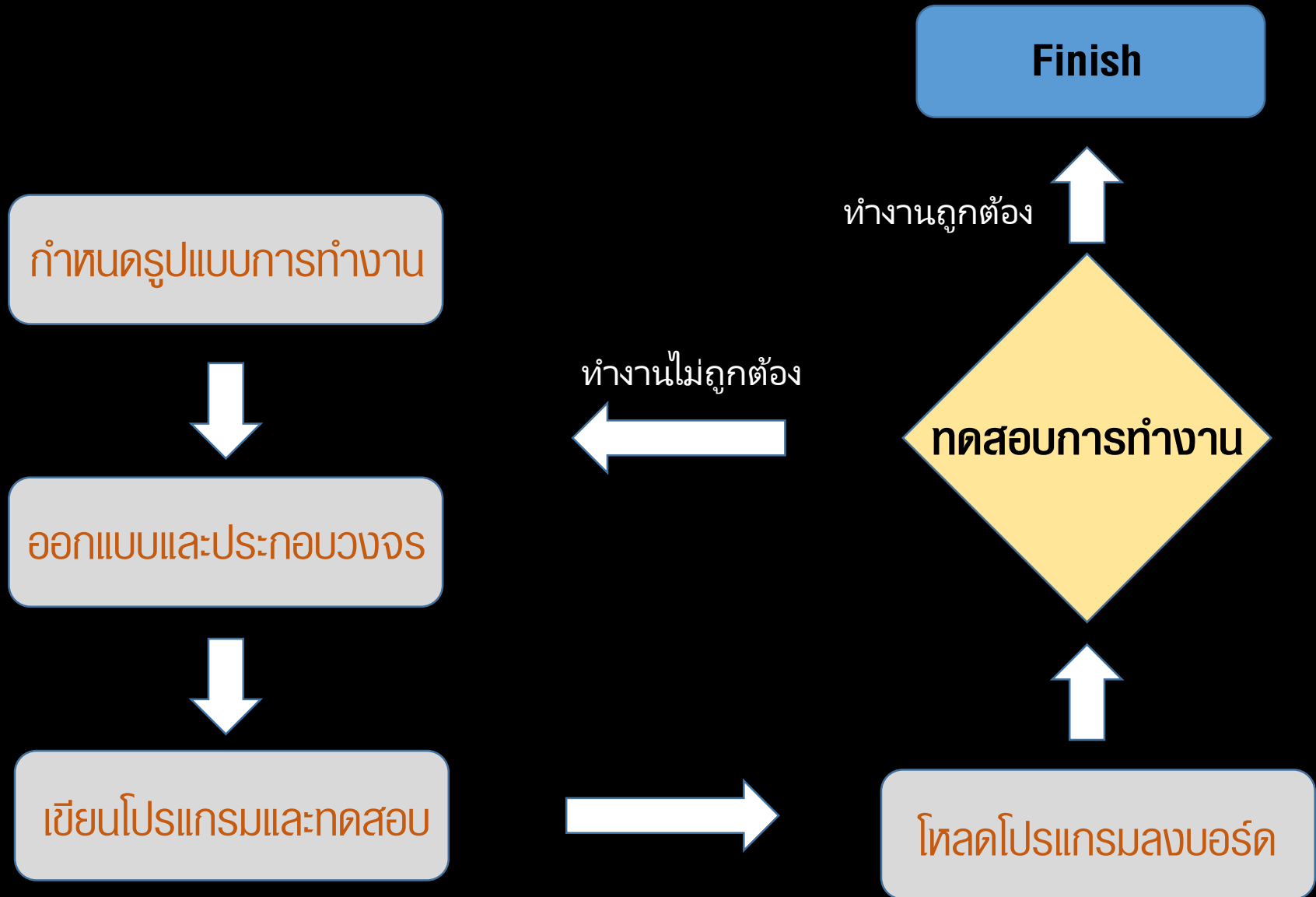
Verify Upload Open Save



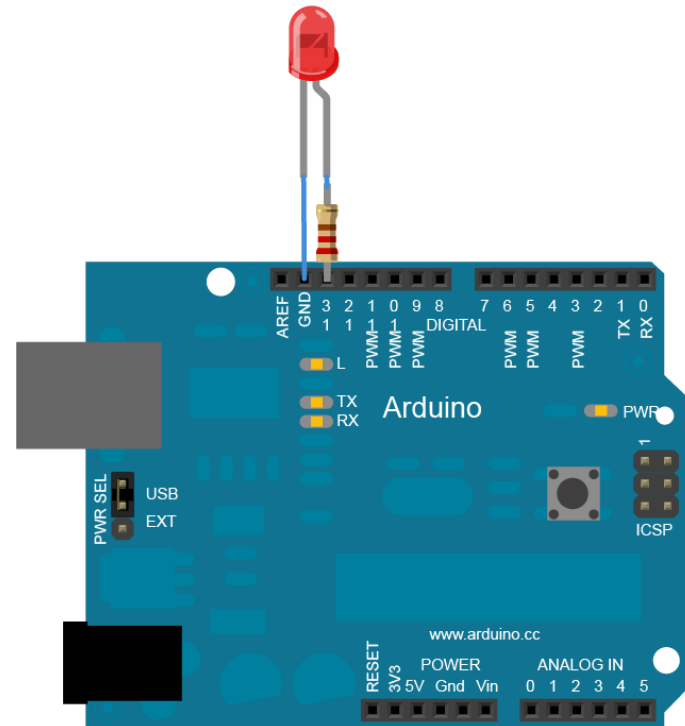
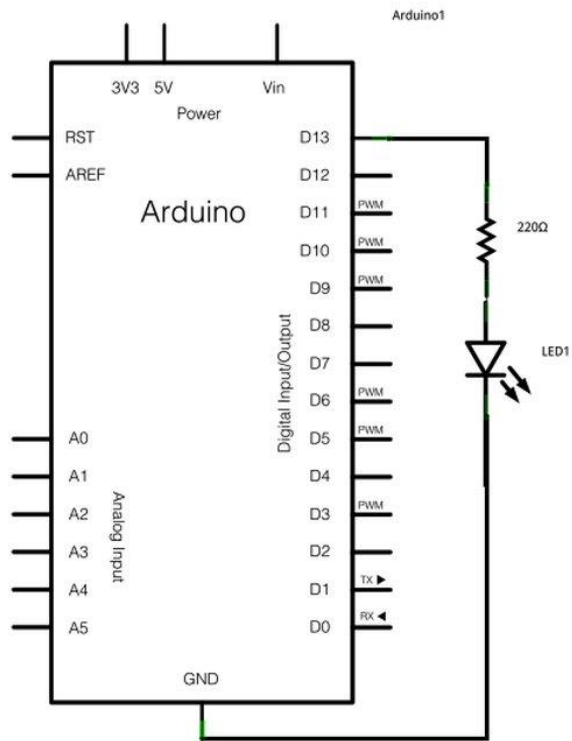
New



ขั้นตอนการสร้างและพัฒนาระบบด้วย Arduino



การออกแบบวงจร และการประกอบวงจร



ตัวอย่างการเขียนโปรแกรม

Blink | Arduino 1.8.2

File Edit Sketch Tools Help



Blink §

```
1 // the setup function runs once when you press reset or power the board
2 void setup() {
3   // initialize digital pin LED_BUILTIN as an output.
4   pinMode(LED_BUILTIN, OUTPUT);
5 }
6
7 // the loop function runs over and over again forever
8 void loop() {
9   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
10  delay(1000); // wait for a second
11  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
12  delay(1000); // wait for a second
13 }
```

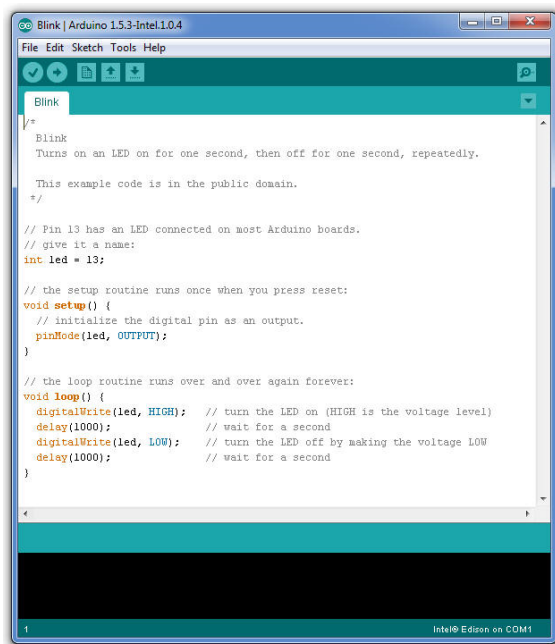


Updates available for some of your [libraries](#) ✖

POP-X2, ATmega644P @ 20 MHz on COM1

วิธีการเขียนโปรแกรมให้กับ Arduino Board

เขียนโปรแกรมภาษา C++



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

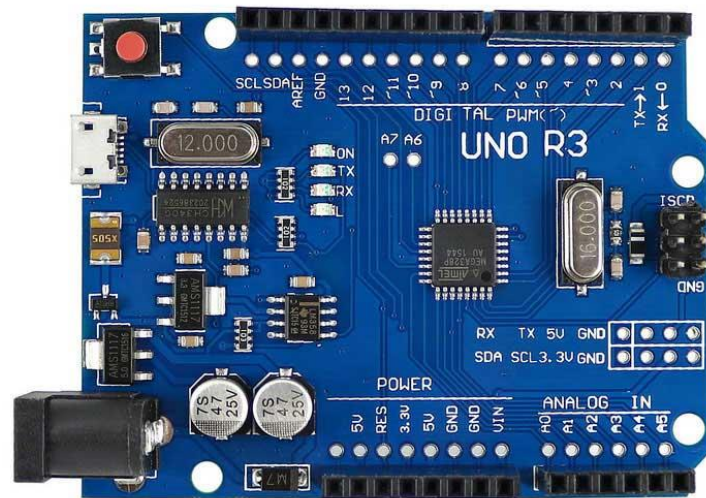
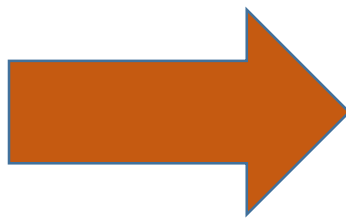
  This example code is in the public domain.
  */

// 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
}
```

คอมไพล์ด้วย Arduino IDE
แล้วโหลดลงบอร์ดผ่าน USB



```
#include "Arduino.h"
```

Preprocessor directive

```
int ledPin ;
```

Declarations

```
void setup()
```

```
{
```

```
  ledPin = 13;
```

```
  pinMode(13,OUTPUT);
```

```
}
```

Definitions

Function

```
Void loop()
```

```
{
```

```
  digitalWrite(13,HIGH);
```

```
  delay(500);
```

```
  digitalWrite(13,LOW);
```

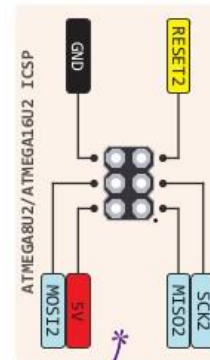
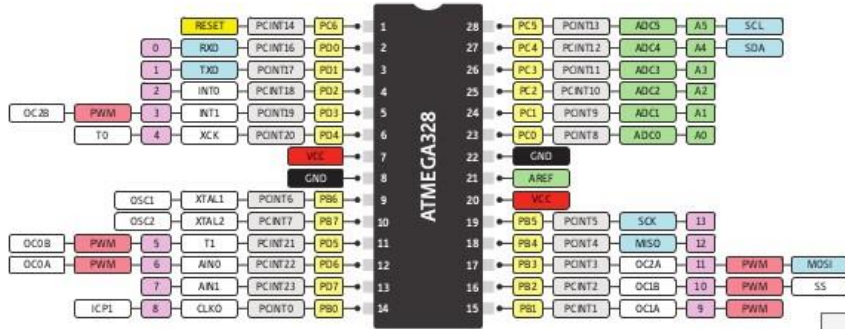
```
  delay(500);
```

```
}
```

Statements

Function

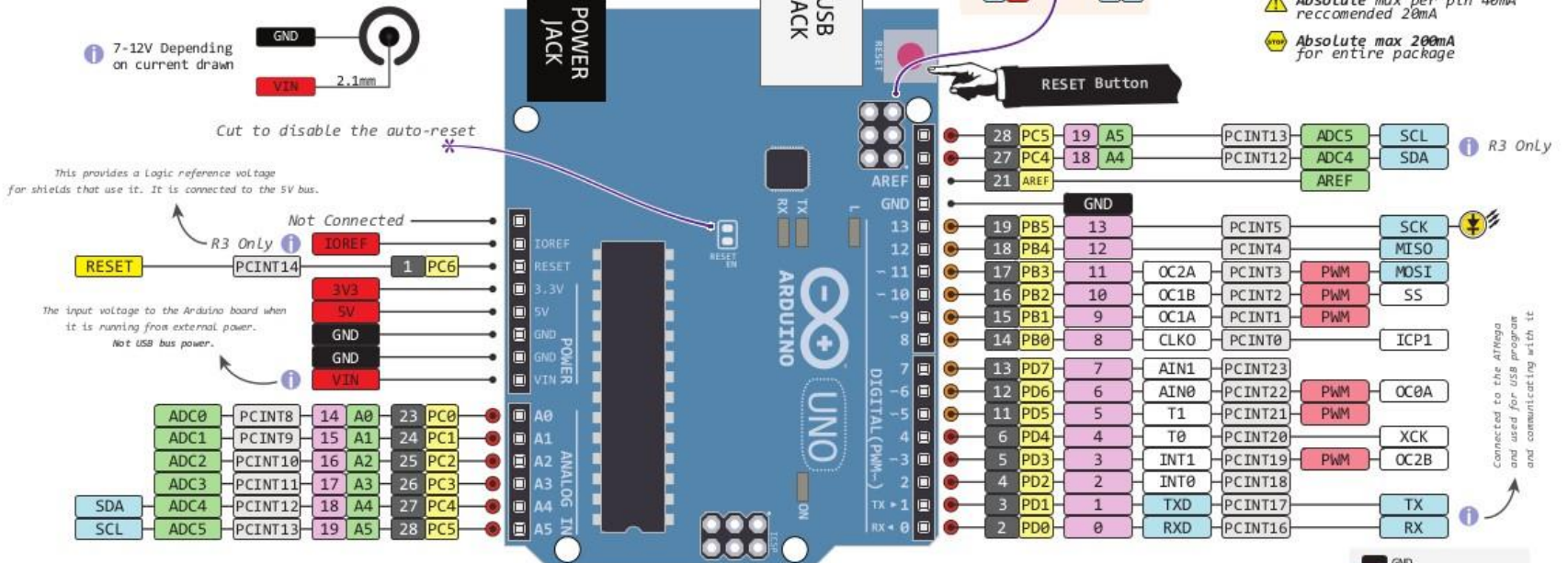
ທ່າໂຮງງານມູນຄ່າ Arduino



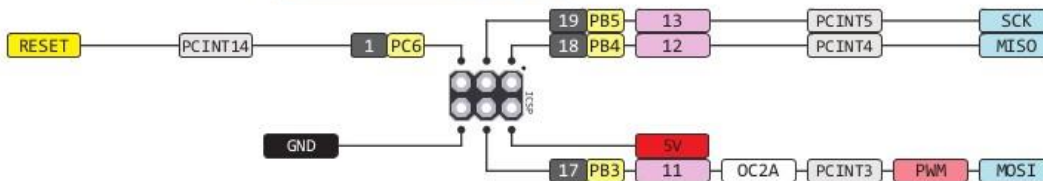
THE
DEFINITIVE
ARDUINO
UNO
PINOUT DIAGRAM

 Absolute max per pin 40mA
recommended 20mA

 Absolute max 200mA
for entire package



ver 2 rev 2 - 05.03.2013



-  GND
-  Power
-  Control
-  Physical Pin
-  Port Pin
-  Pin Function
-  Digital Pin
-  Analog Related Pin
-  PWM Pin
-  Serial Pin
-  IDE
-  Source Total 150mA

คำสั่งกำหนดการทำงาน

รูปแบบคำสั่งกำหนดการทำงาน **pinMode**

```
pinMode(Arduino_Pin, Mode);
```

pinMode(13, OUTPUT);



กำหนดให้ขา 13

ทำงานเป็น OUTPUT

Arduino_Pin คือขาของบอร์ด Arduino ที่เราต้องการกำหนดลักษณะการทำงาน

Mode คือลักษณะการทำงานที่ต้องการกำหนดให้เป็น **INPUT** หรือ **OUTPUT**

คำสั่งควบคุม

รูปแบบคำสั่งกำหนดการทำงาน **digitalWrite**

```
digitalWrite(Arduino_Pin, Output);
```

digitalWrite(13, HIGH);



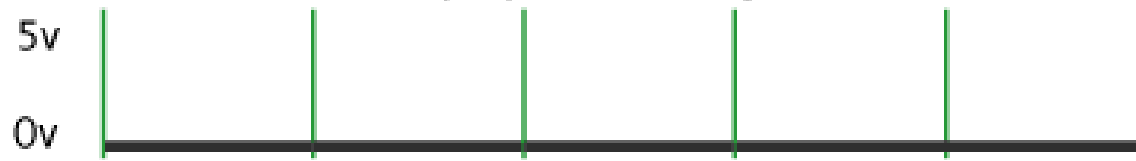
สั่งงานให้ขา 13
สถานะเป็น HIGH

Arduino_Pin คือขาของบอร์ด Arduino ที่เราต้องการควบคุม

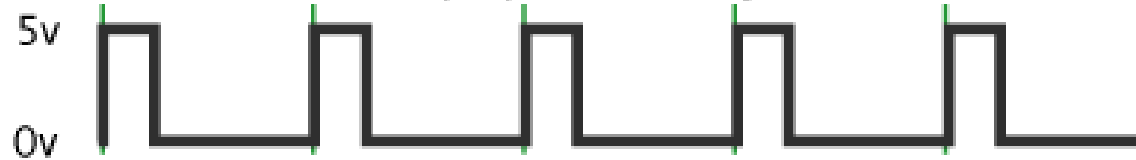
Output คือสถานะการทำงานที่ต้องการกำหนดให้มามีค่า **HIGH** หรือ **LOW**

Pulse Width Modulation

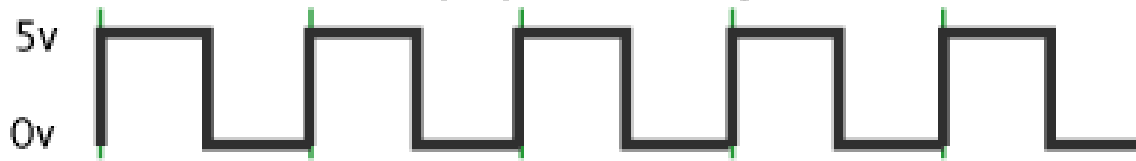
0% Duty Cycle - `analogWrite(0)`



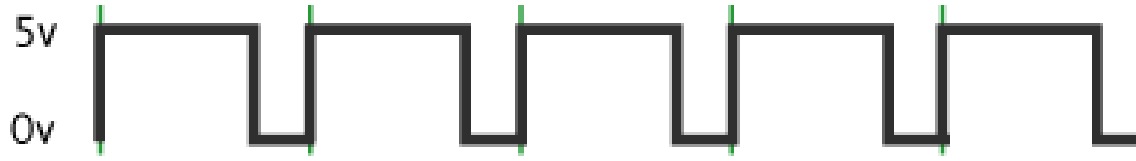
25% Duty Cycle - `analogWrite(64)`



50% Duty Cycle - `analogWrite(127)`



75% Duty Cycle - `analogWrite(191)`



100% Duty Cycle - `analogWrite(255)`



คำสั่งนับเวลา

รูปแบบคำสั่งกำหนดการทำงาน **delay**

```
delay(wait_time);
```

delay(1000);



นับเวลา 1 วินาที

Wait_time คือจำนวนเวลาที่ต้องการนับมีหน่วยเป็น millisec 1000=1 วินาที
เมื่อนับเวลาครบตามที่กำหนดโปรแกรมก็จะทำงานคำสั่งถัดไป

คำสั่งควบคุม

รูปแบบคำสั่งกำหนดการทำงาน **analogWrite**

```
analogWrite(Arduino_Pin, Output);
```

analogWrite(6, 255);



สั่งงานให้ขา 7
สร้างสัญญาณ pwm

Arduino_Pin คือขาของบอร์ด Arduino ที่เราต้องการสร้างสัญญาณ

Output คือค่าความถี่ที่สร้างตั้งแต่ 0 ถึง 255

การสื่อสารกับ Computer

รูปแบบคำสั่งเริ่มต้นการสื่อสาร

```
Serial.begin(baudRate);
```

Serial.begin(9600);



อ่านค่าลอจิกจากขา 13

baudrate คือความเร็วในการสื่อสารข้อมูลระหว่างบอร์ดกับ Computer

การอ่านค่าจากเซ็นเซอร์

รูปแบบคำสั่งกำหนดการทำงาน **digitalRead**

```
digitalRead(Arduino_Pin);
```

digitalRead(10);



อ่านค่าลอจิกจากขา 10

Arduino_Pin คือขาของบอร์ด Arduino ที่เราต้องการอ่านค่า
โดยการอ่านค่าจากคำสั่งนี้จะได้ค่าออกมาเป็น 0 หรือ 1 เท่านั้น

การอ่านค่าจากเซ็นเซอร์

รูปแบบคำสั่งกำหนดการทำงาน **analogRead**

```
analogRead(Arduino_Pin);
```

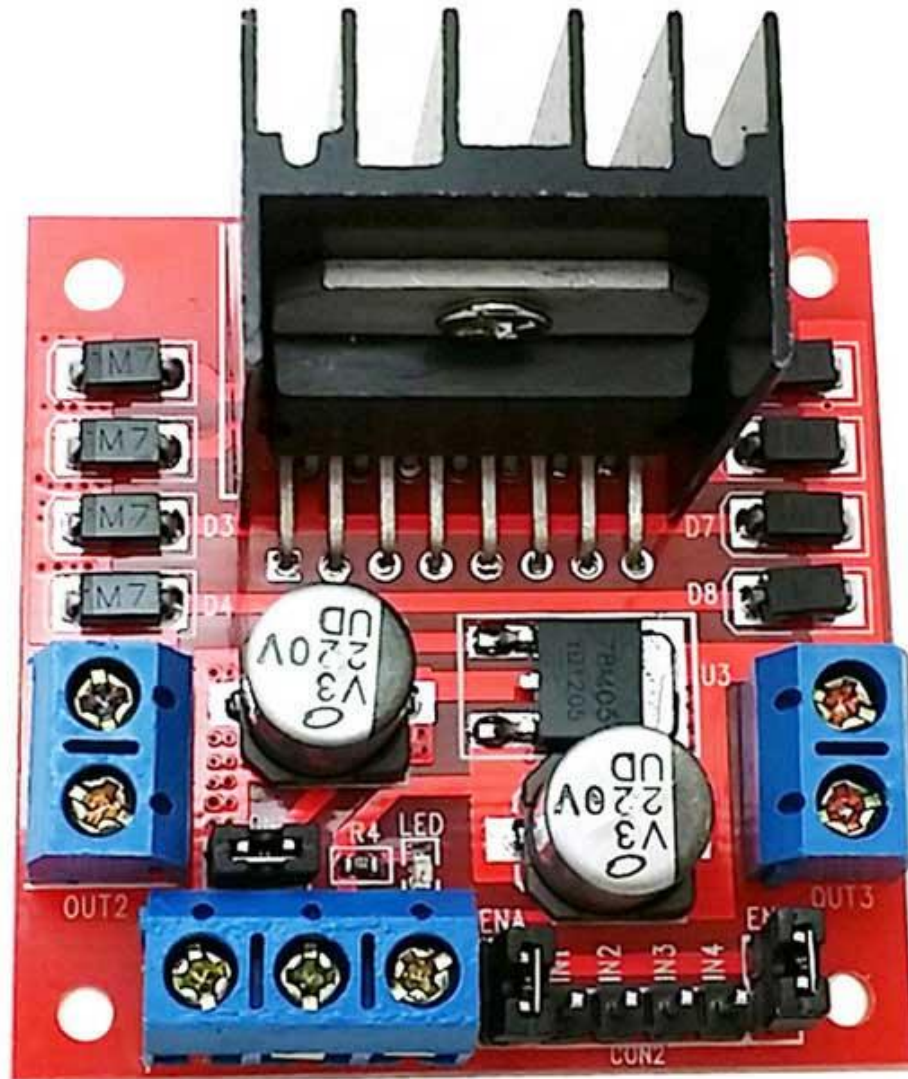
digitalRead(A0);



อ่านค่าสัญญาณจากขา A0

Arduino_Pin คือขาของบอร์ด Arduino ที่เราต้องการอ่านค่า
โดยการอ่านค่าจากคำสั่งนี้จะได้ออกมาเป็น 0 - 1023

L-298N Motor Driver



วงจรรวม

INPUT			OUTPUT		Direction
PWM	CW	CCW	OUT1	OUT2	
L	x	x	—	—	—
H	L	L	GND	GND	—
H	L	H	GND	VS	CCW
H	H	L	VS	GND	CW
H	H	H	VS	VS	—

