

Section 2



อัฒน์มหาวิทยาลัยเชียงใหม่



DIPROM
สถาบันวิจัยและพัฒนาคุณภาพการเรียน
Institute for Research and Development of Quality Education

โดย กองพัฒนาธุรกิจก่ออุตสาหกรรม

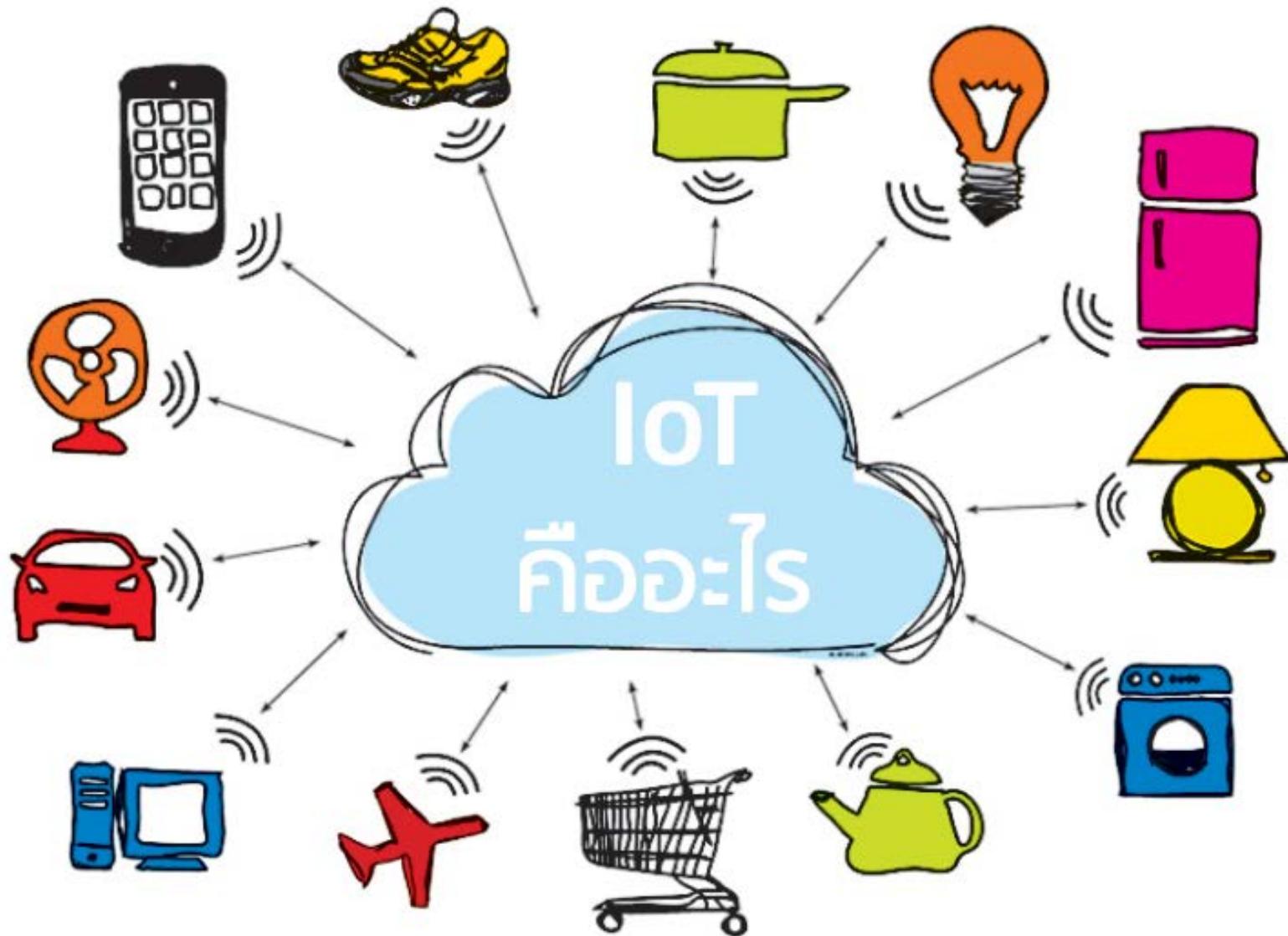
การประยุกต์ระบบ IoT เพิ่มผลิตภ้าภการผลิต (Productivity) ด้วยโปรแกรม Arduino

ผ่านระบบออนไลน์ จำนวน **80** ท่าน **ฟรี**

วันที่ 7 - 9 กันยายน 2564 เวลา 9.00-16.00 น

อ.พสิษฐ์ ธนาโชติอนันต์กุล pasidthdr@gmail.com ผู้เชี่ยวชาญระบบ IoT

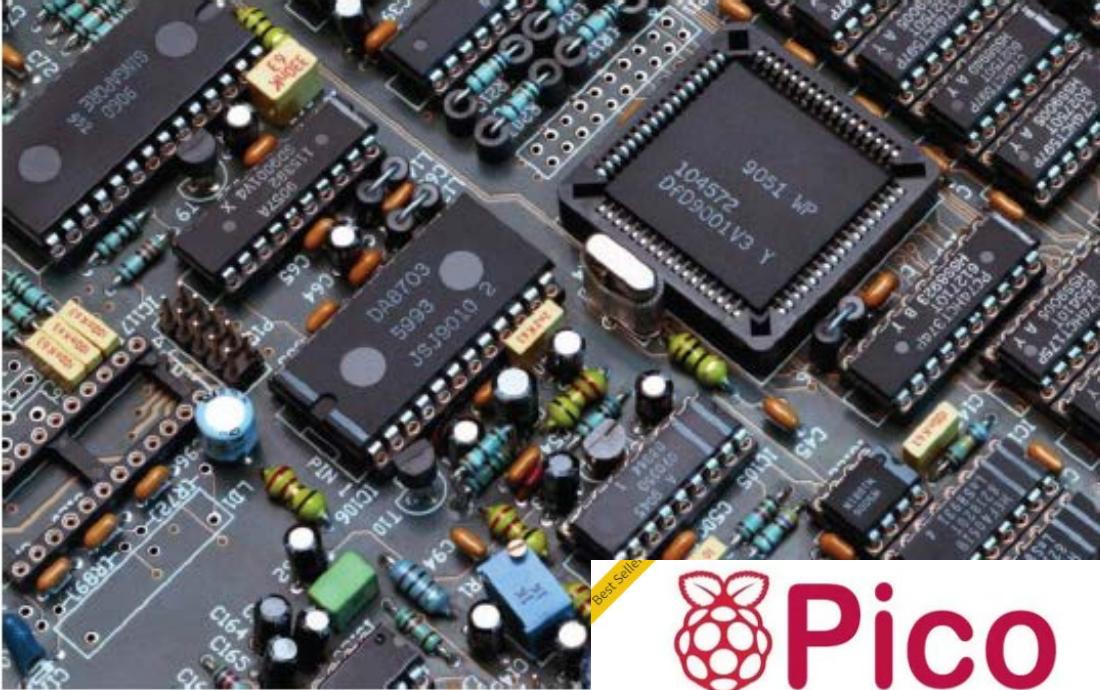
อ.อัชฎง ภูรักษ์ asdongph@gmail.com

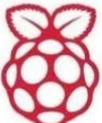


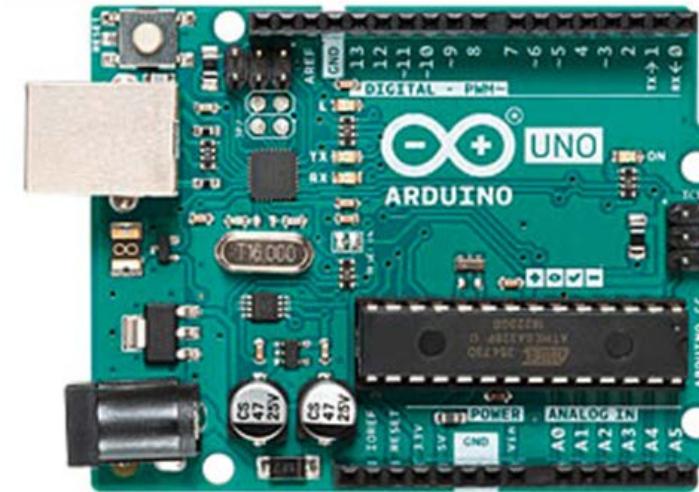
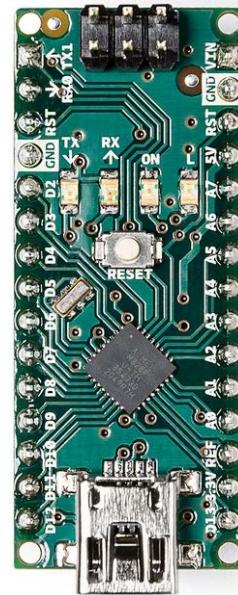
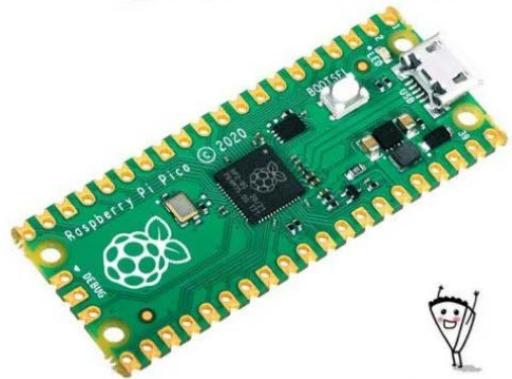
อินเทอร์เน็ตของสรรพสิ่ง (IoT)

อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) หรือ IoT หมายถึง เครื่อข่ายของวัตถุ อุปกรณ์ พาหนะ สิ่งปลูกสร้าง และสิ่งของอื่นๆ ที่ มีวงจร อิเล็กทรอนิกส์ ซอฟต์แวร์ เทช์นเซอร์ และ การเชื่อมต่อกับเครื่อข่ายฝังตัว (Embedded system) และทำให้วัตถุเหล่านั้น สามารถเก็บบันทึกและแลกเปลี่ยนข้อมูลได้ อินเทอร์เน็ตของสรรพสิ่งทำให้วัตถุสามารถรับรู้สภาพแวดล้อมและ ถูกควบคุมได้จากระยะไกลผ่านโครงสร้างพื้นฐานเครือข่ายที่มีอยู่แล้ว

Microcontroller - Embedded systems



 **Pico**

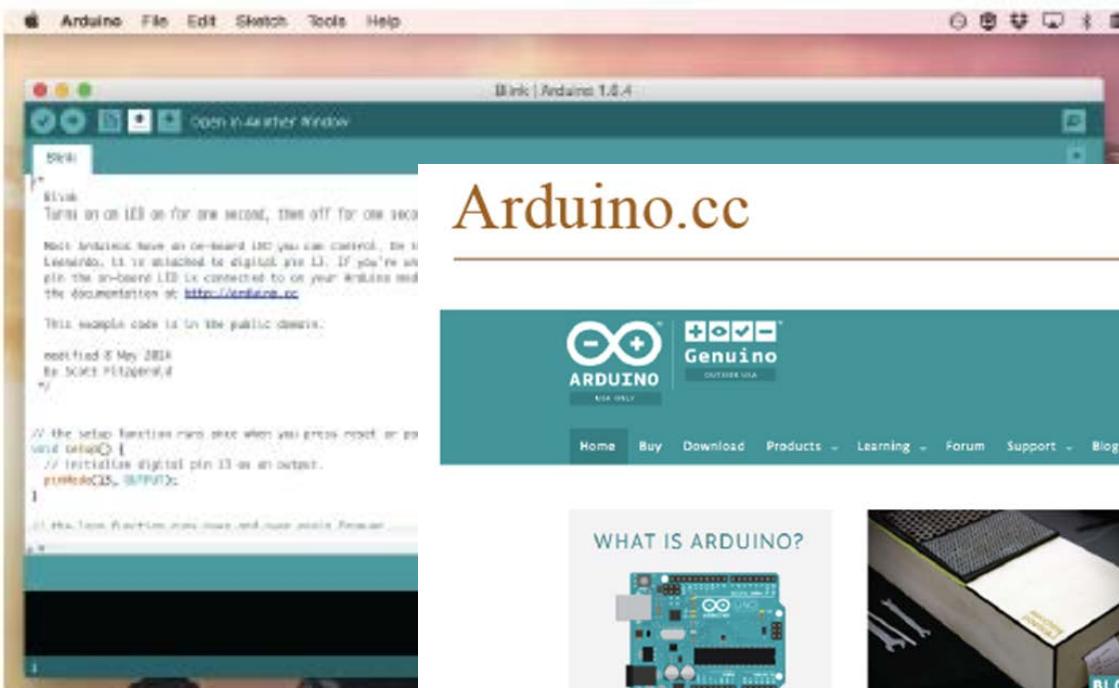


Arduino

Hardware



Software + IDE (Integrated Development Environment)



Arduino.cc

WHAT IS ARDUINO?

BUY AN ARDUINO

LEARN ARDUINO

MACHINA POETICA
CONVERTS SOUNDS INTO
ONOMATOPOEIC WORDS

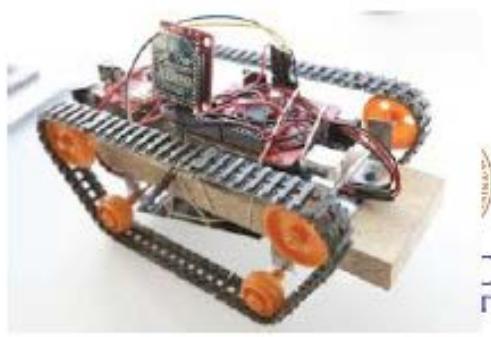
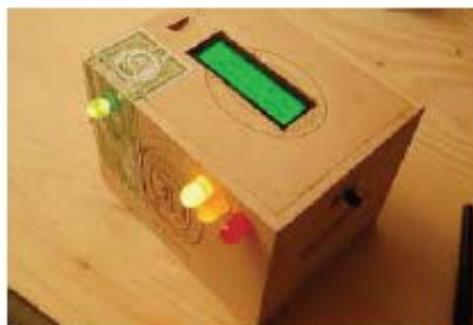
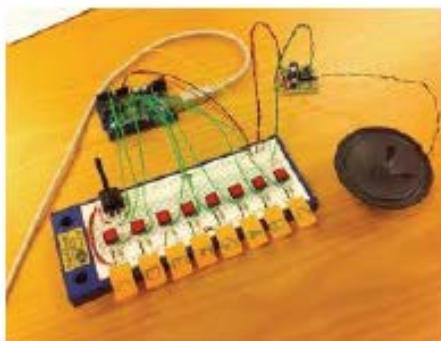
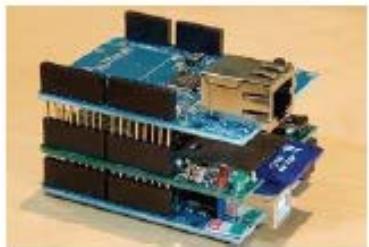
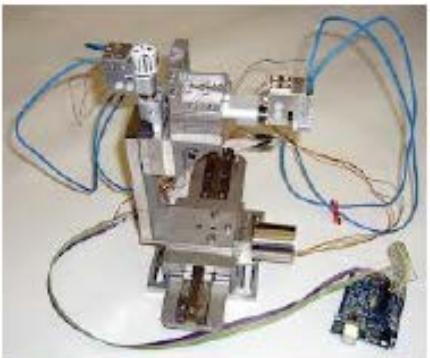
ARDUINO DAY 2016

SUBMISSIONS ARE OPEN!

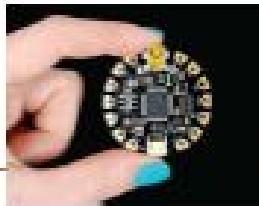
APRIL
2nd 2016

DAY.ARDUINO.CC

Project using an Arduino



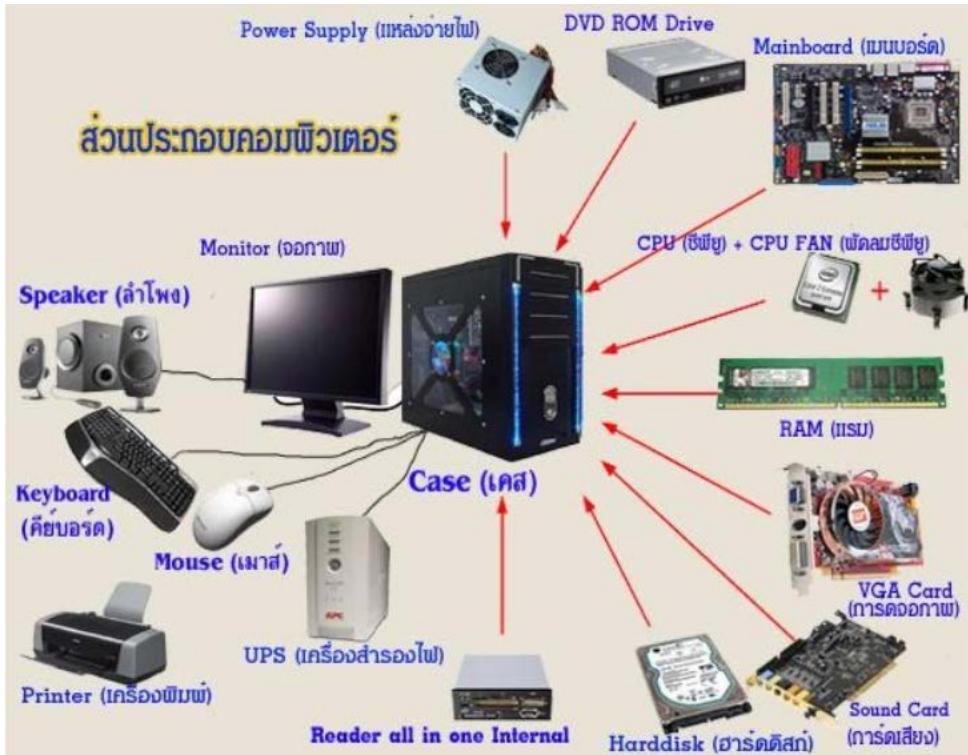
Wearable Electronics



Flora, Lilypad, Gemma

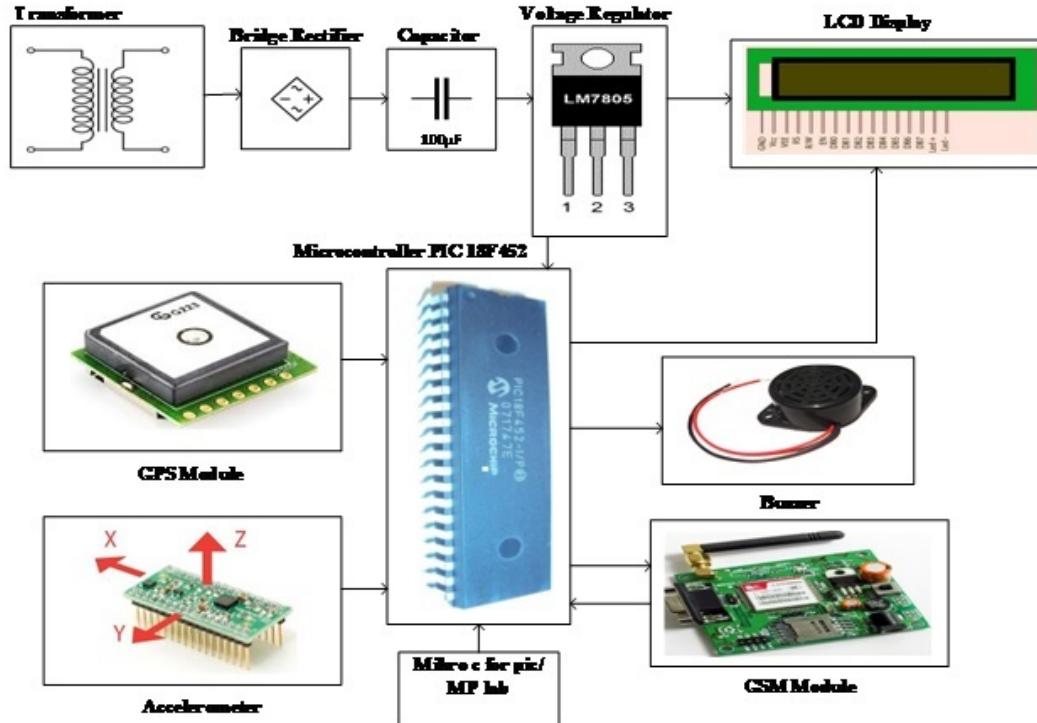


ระบบคอมพิวเตอร์



ไมโครคอมพิวเตอร์ (microcomputer)
คือ คอมพิวเตอร์ที่ ไม่มีไมโครโปรเซสเซอร์
เป็นหน่วยประมวลผลกลาง คุณสมบัติ
ประการหนึ่งของคอมพิวเตอร์ชนิดนี้คือ
จะกินพื้นที่น้อยกว่าเมื่อเทียบกับ
มินิคอมพิวเตอร์ (minicomputer) และ
เครื่องメインเฟรม (mainframe)
ไมโครคอมพิวเตอร์ที่ประกอบไปด้วยคีย์
บอร์ดและจอภาพ โดยปกติเรียกว่าเป็น
คอมพิวเตอร์ส่วนบุคคล (personal
computer)

ไมโครคอนโทรลเลอร์



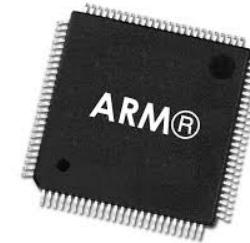
The further information and detail about this, Accidental Alarm System Using Pic Microcontroller is available at website <http://microcontrollerslab.com>

ไมโครคอนโทรลเลอร์
(microcontroller หรือ MCU)
คือ อุปกรณ์ควบคุมขนาดเล็ก ซึ่ง
บรรจุความสามารถที่คล้ายคลึงกับ
ระบบคอมพิวเตอร์ โดยใน
ไมโครคอนโทรลเลอร์ได้รวมเอาชิปปุ่ย,
หน่วยความจำ และพอร์ต ซึ่งเป็น
ส่วนประกอบหลักสำคัญของระบบ
คอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำ
การบรรจุเข้าไว้ในตัวถังเดียวกัน

Micro
processor



Intel-Celeron, Pentium, Core i

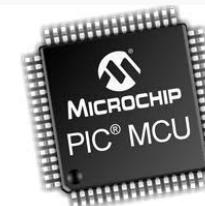
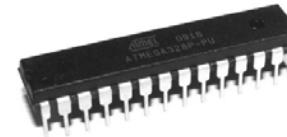


Cortex-
A series

Micro
controller



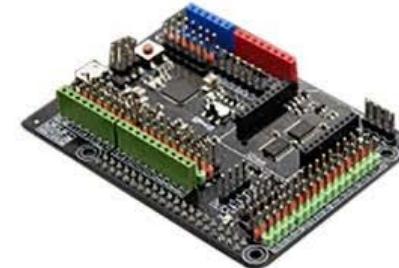
Atmel - ATmega



STM32

Microchip-PIC

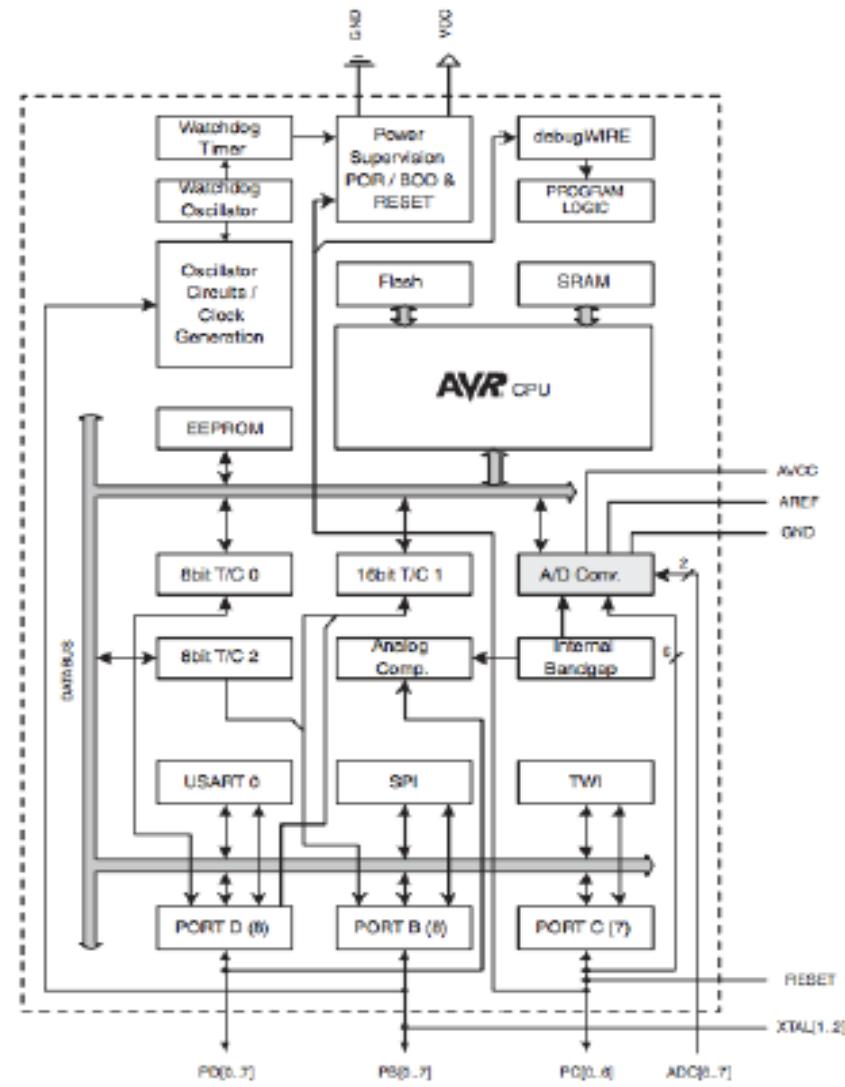
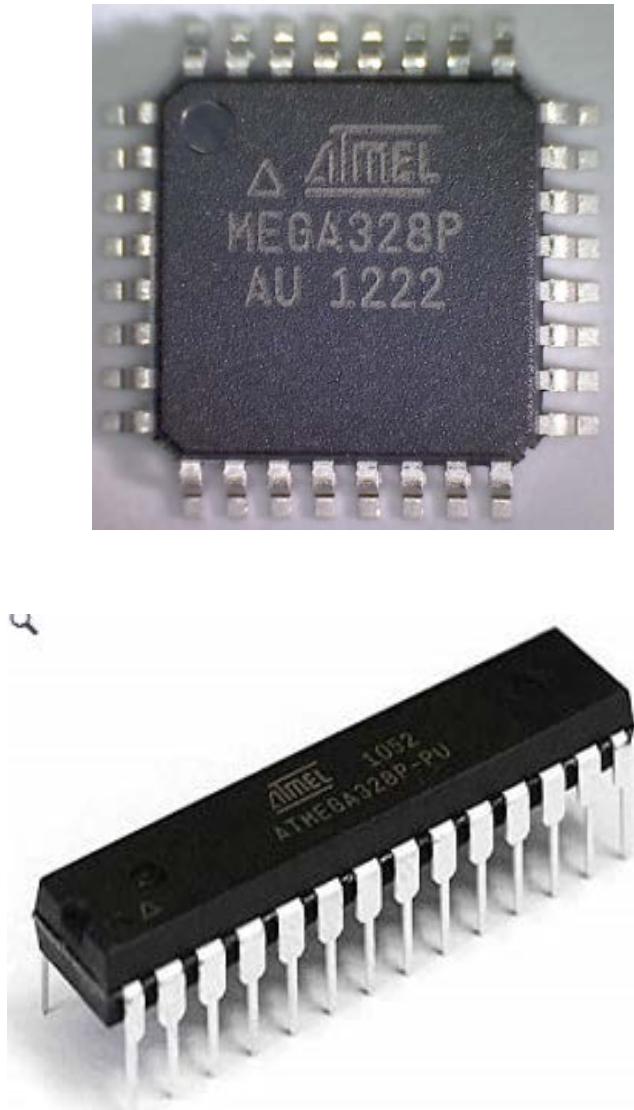
Shield



* CHECK NOTICES
AND EXCLUSIONS

Inside the microcontroller

Block Diagram





Arduino Uno



Arduino Leonardo



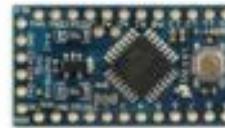
Arduino Robot



Arduino Esplora



Arduino Micro



Arduino Pro Mini



Arduino Due



Arduino Yún



Arduino BT



Arduino Mega 2560



Arduino Pro



Arduino Fio



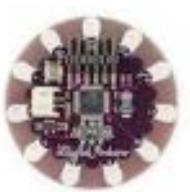
Arduino Mega ADK



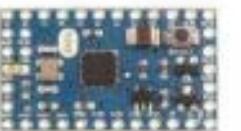
Arduino Ethernet



LilyPad Arduino USB



LilyPad Arduino Simple



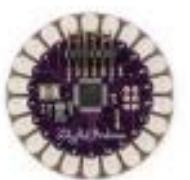
Arduino Mini



Arduino Nano



LilyPad Arduino SimpleSnap



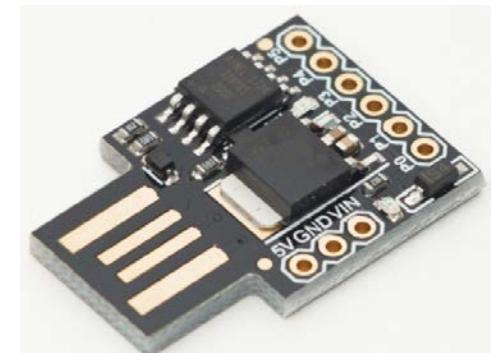
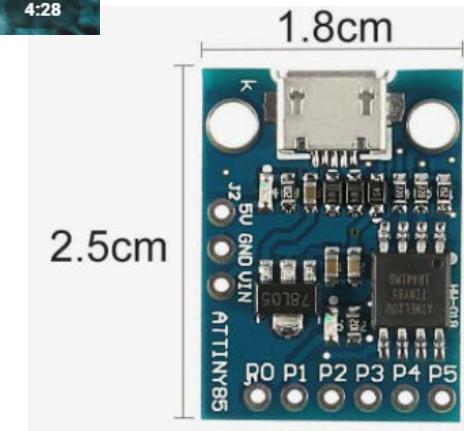
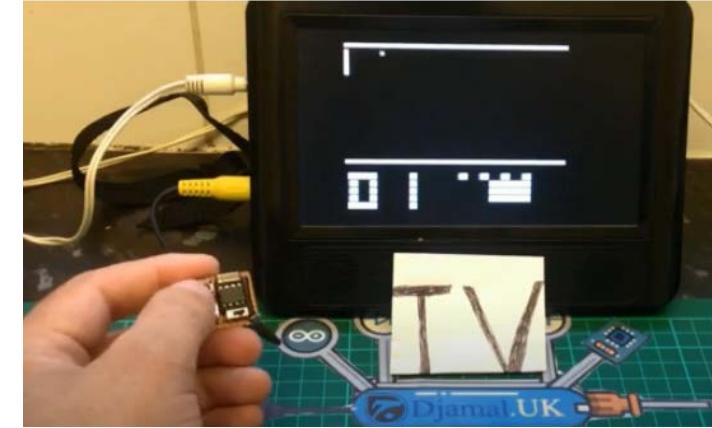
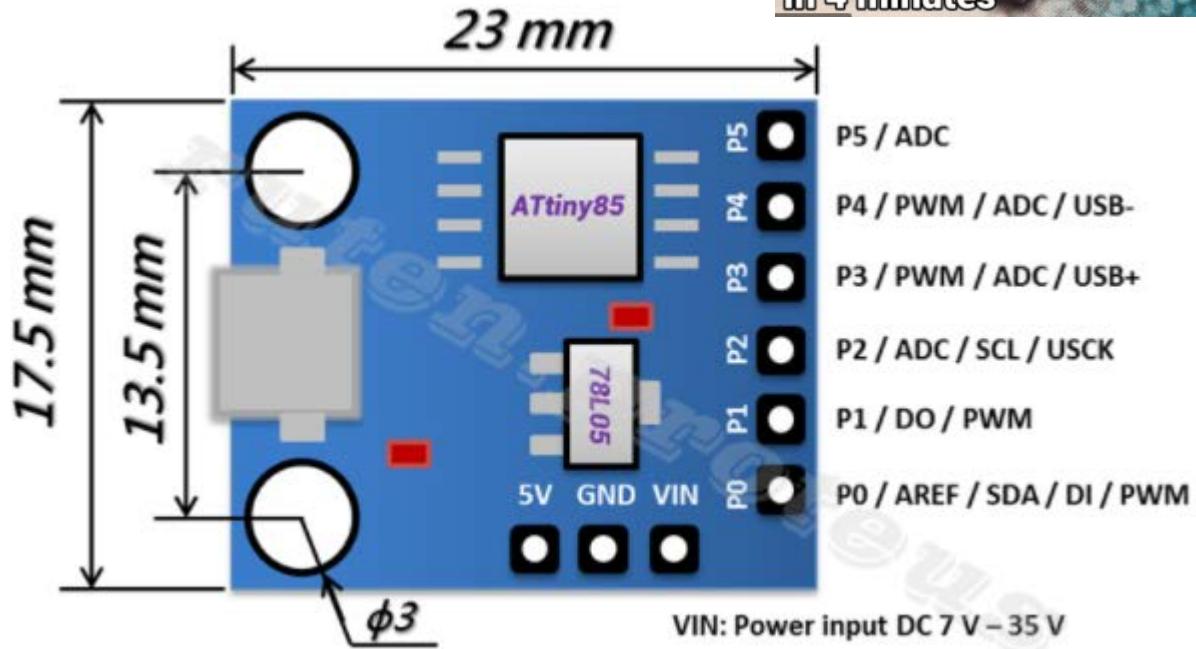
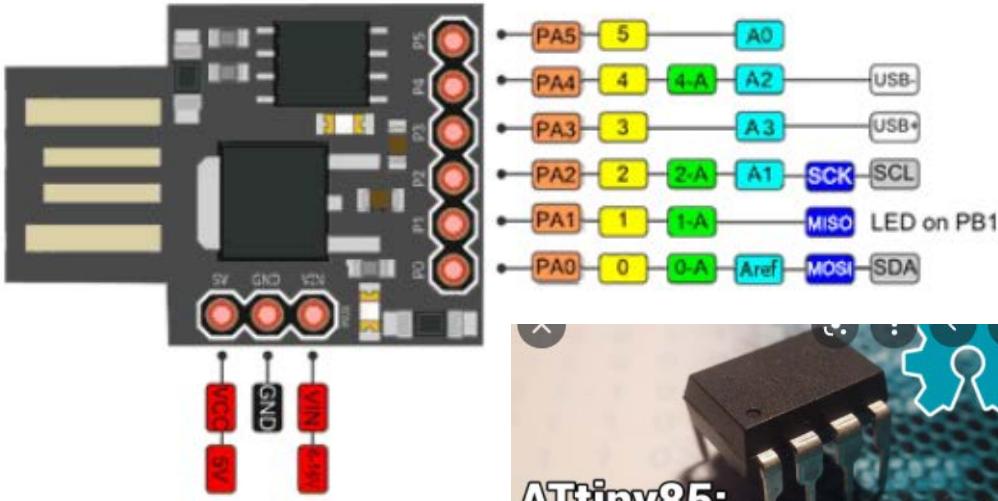
LilyPad Arduino

บอร์ดอาร์ดูโน่รุ่น
ต่างๆ

Board	Micro-controller	Clock Speed	Flash Memory	SRAM	EEPROM
Arduino UNO	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Lenonardo	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino DUE	AT91SAM3X8E	84 MHz	512 KB	96 KB	-
Arduino YUN	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino Mega ADK	ATmega2560	16 MHz	256 KB	8 KB	4 KB
Arduino Ethernet	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Mega 2560	ATmega2560	16 MHz	256 KB	8 KB	4 KB
Arduino BT (Bluetooth)	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Micro	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino Pro Mini	ATmega168	8 MHz	16 KB	1 KB	512 Bytes
Arduino Pro	ATmega168	8 MHz	16 KB	1 KB	512 Bytes
Arduino Pro	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Mini	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Nano	ATmega168	16 MHz	16 KB	1 KB	512 Bytes
Arduino Nano	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Fio	ATmega328P	8 MHz	32 KB	2 KB	1 KB
Arduino Robot	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino Esplora	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
LilyPad Arduino USB	ATmega32u4	8 MHz	32 KB	2.5 KB	1 KB
LilyPad Arduino Simple	ATmega328	8 MHz	32 KB	2 KB	1 KB
LilyPad Arduino SimpleSnap	ATmega328	8 MHz	32 KB	2 KB	1 KB
LilyPad Arduino	ATmega168V	8 MHz	16 KB	1 KB	512 Bytes
LilyPad Arduino	ATmega328V	8 MHz	16 KB	1 KB	512 Bytes

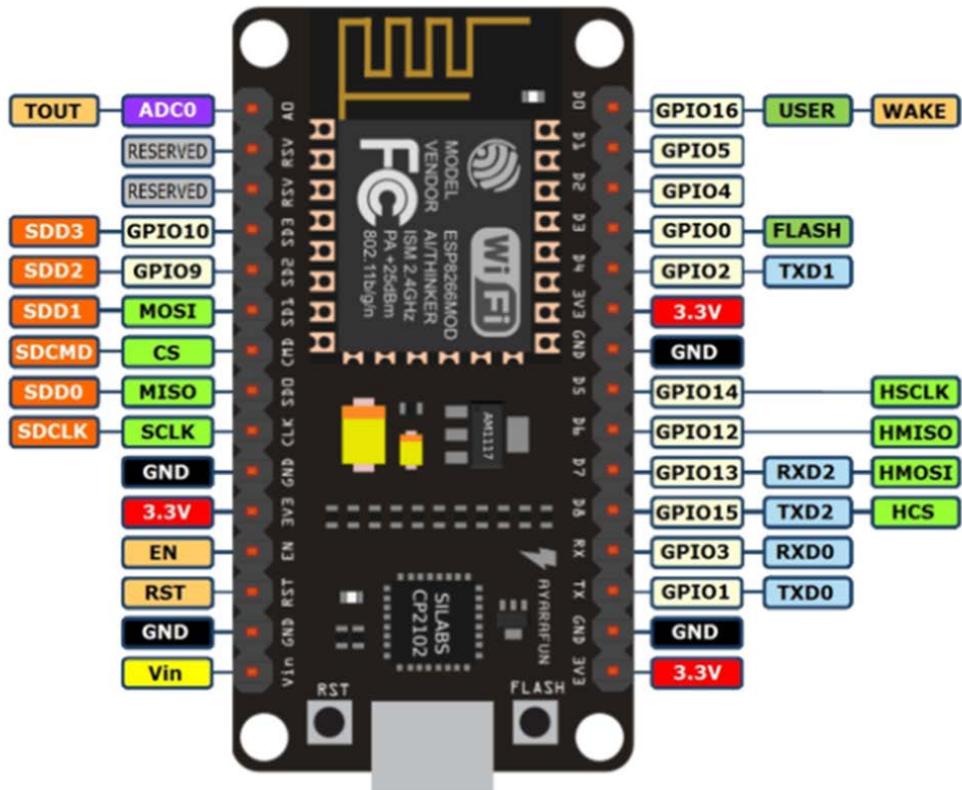
GROUND
POWER
PHYSICAL PIN
DIGITAL PIN
digitalRead() digitalWrite() pinMode()
PWM PIN
analogWrite()
ANALOG PIN
analogRead() analogReference()
SPI
SPI Library
UART/LIN
Serial1 Library
I2C
Wire Library
USB PIN
Programming Serial Library DigiKeyboard Library DigiMouse Library DigiJoystick Library

World smallest Arduino

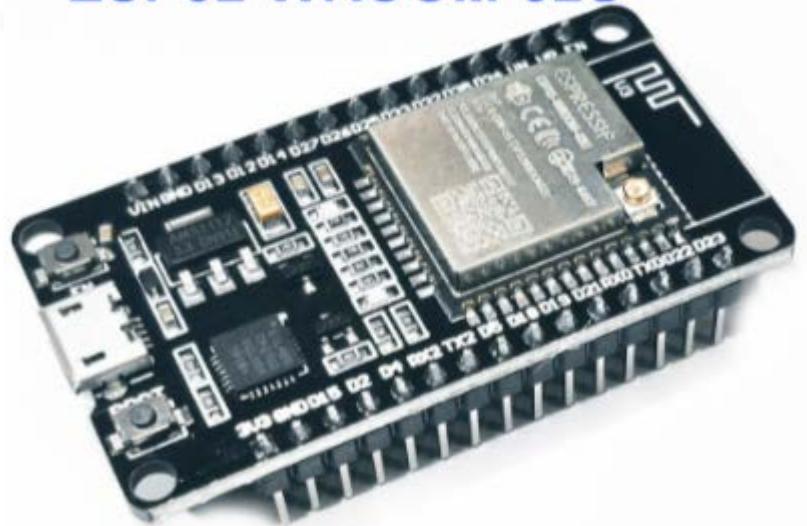


https://www.youtube.com/watch?v=IB-4kZko3E8&ab_channel=maxticmaxtic

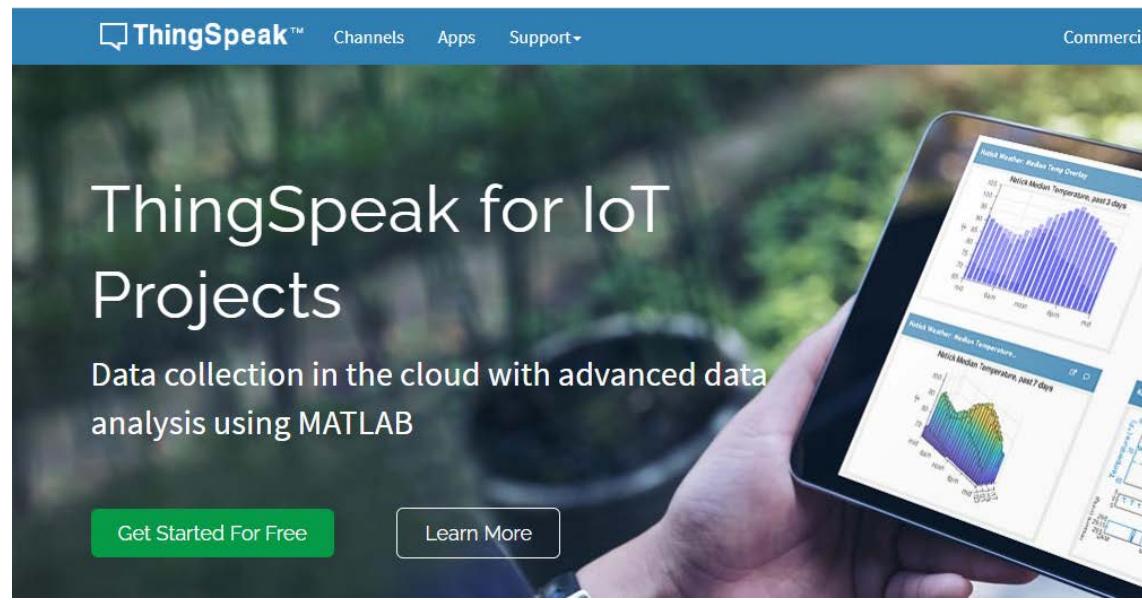
ESP series for IoT projects



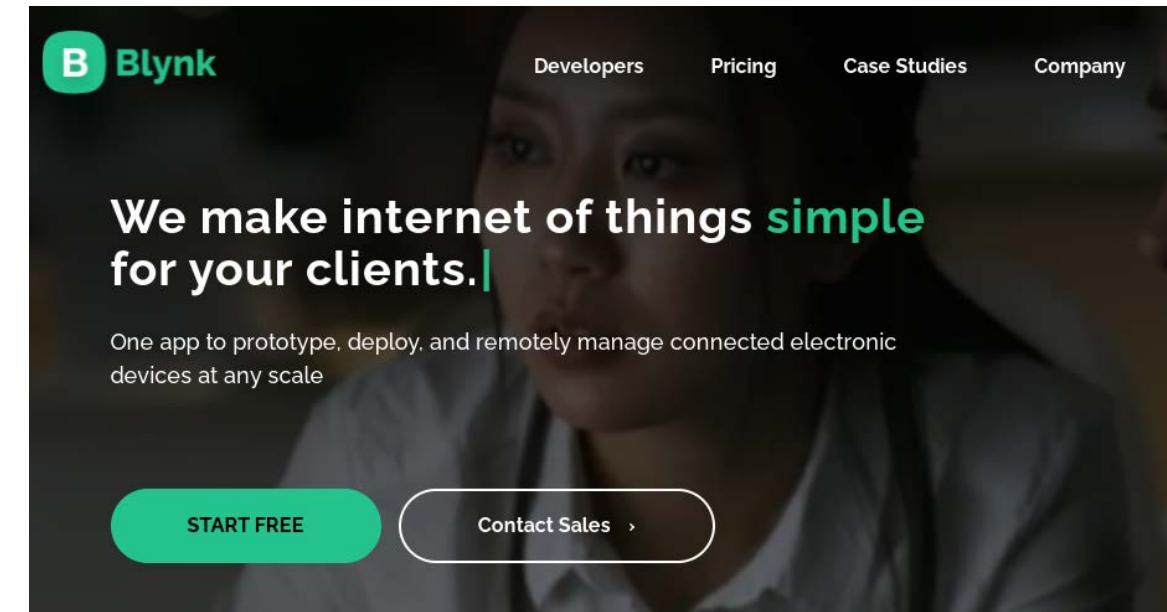
**30PIN
ESP32-WROOM-32U**



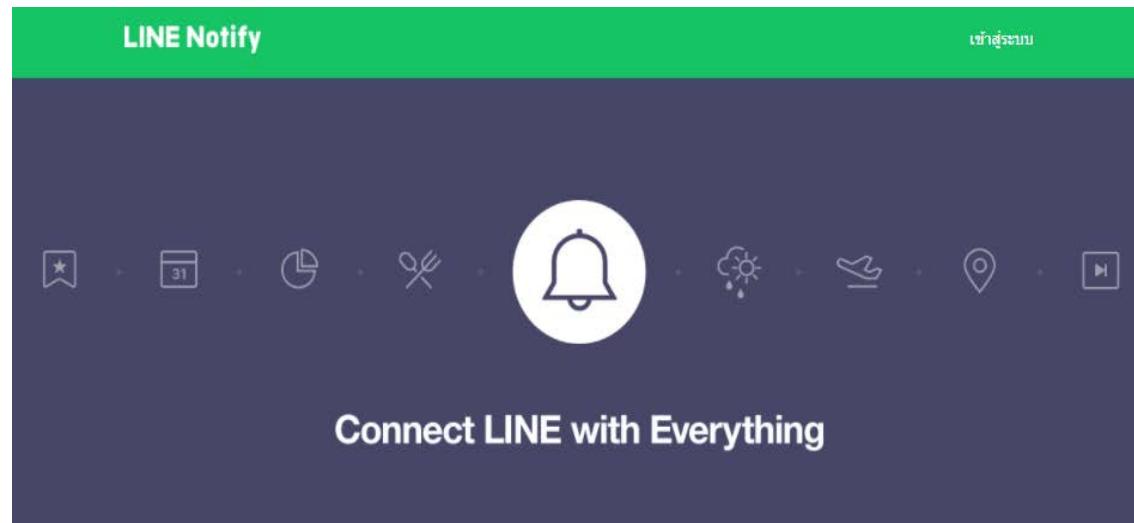
IoT platform & Dashboard



The screenshot shows the ThingSpeak website. At the top, there's a navigation bar with links for "ThingSpeak™", "Channels", "Apps", "Support", and "Commercial". The main heading "ThingSpeak for IoT Projects" is displayed prominently. Below it, a sub-headline reads "Data collection in the cloud with advanced data analysis using MATLAB". A photograph of a person's hand holding a smartphone displays a MATLAB-based dashboard with various charts and graphs. Two buttons at the bottom are "Get Started For Free" and "Learn More".



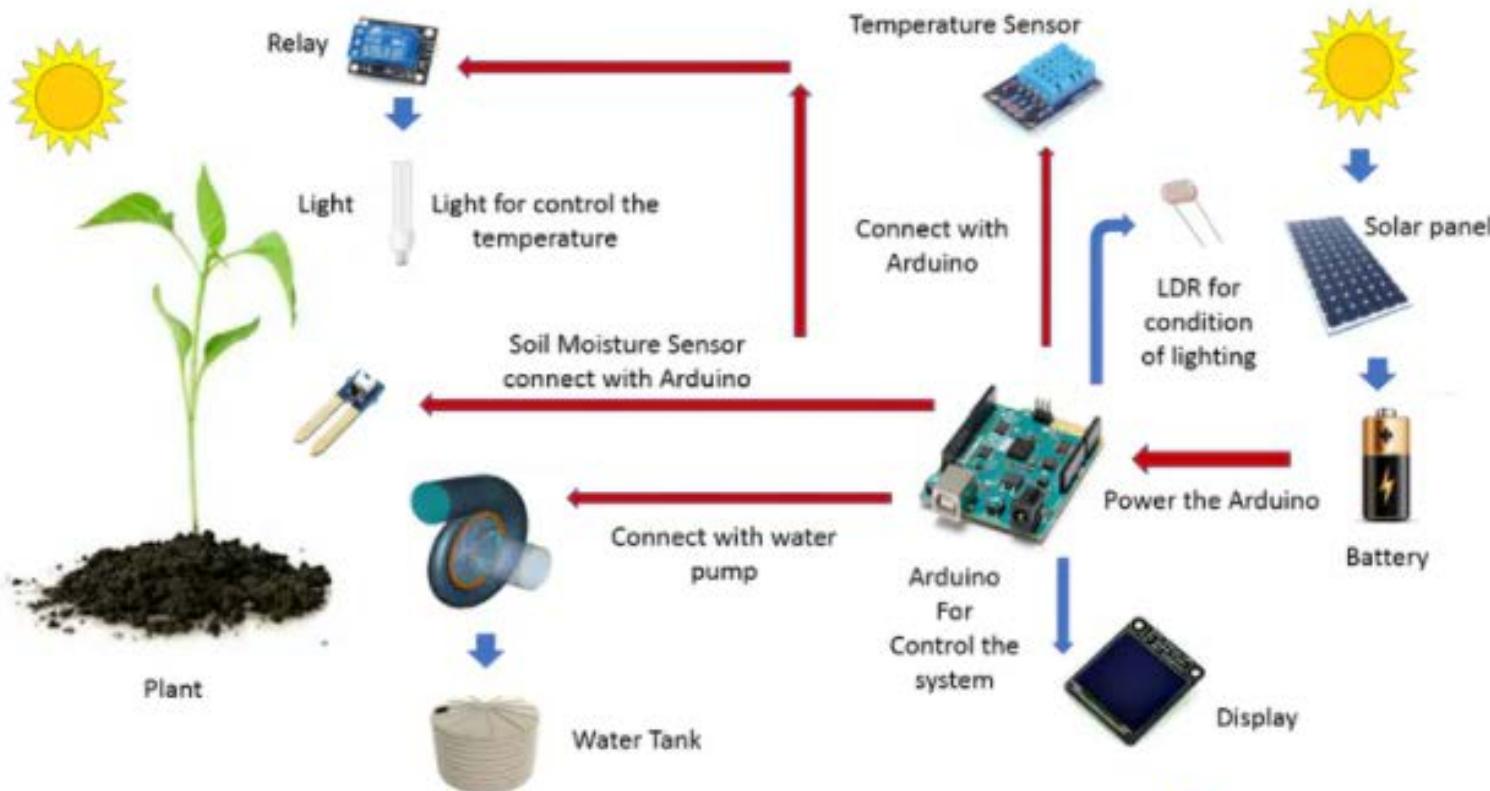
The screenshot shows the Blynk website. The header includes the Blynk logo, "Blynk", and navigation links for "Developers", "Pricing", "Case Studies", and "Company". The main headline is "We make internet of things simple for your clients." Below it, a sub-headline states "One app to prototype, deploy, and remotely manage connected electronic devices at any scale". At the bottom, there are two buttons: "START FREE" and "Contact Sales".

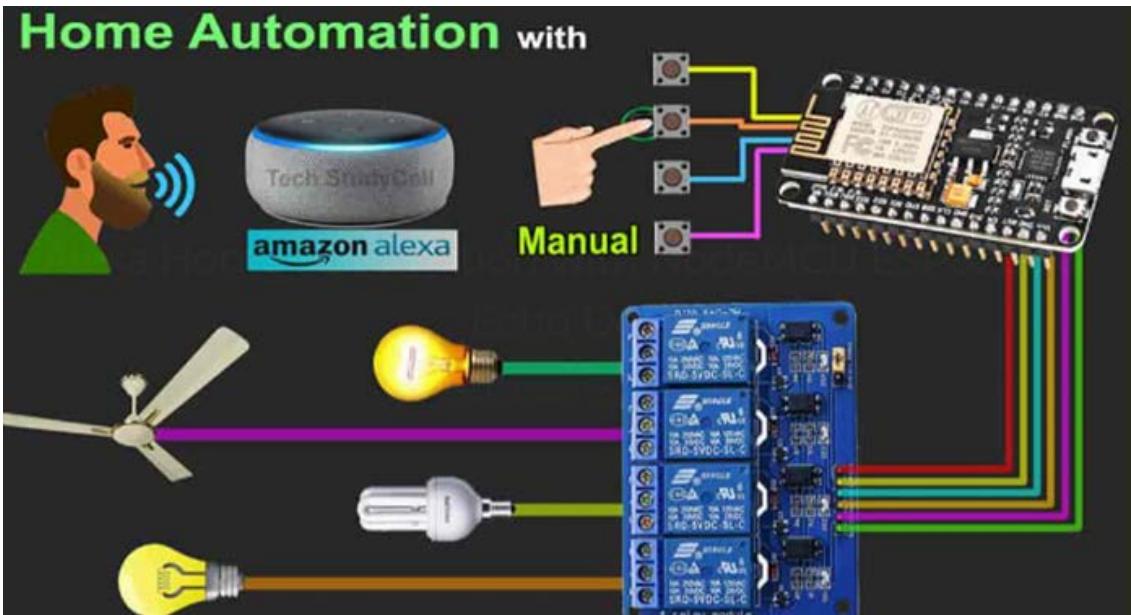


The screenshot shows the LINE Notify website. It features a green header with the text "LINE Notify" and "เข้าสู่ระบบ". Below the header is a dark blue navigation bar with icons for a star, a calendar, a pie chart, a scissor-like icon, a bell (highlighted in white), a sun with rain, a hand, a location pin, and a square. The main content area has the text "Connect LINE with Everything".



Smart Garden





IoT Circuit Hub
makes IoT simple

[HOME](#) [PROJECTS](#) [TUTORIALS](#) [CONTACT](#) [SEARCH](#) [VIDEO](#)

Blynk 2.0

Infrared Manual Temperature LDR

with NO Wi-Fi

ESP32

IOT Projects Ideas for Engineering Students

Explore IoT projects with Blynk, Amazon Alexa, Google Home, Sinric, Arduino IoT Cloud, IFTTT using ESP8266, NodeMCU, ESP32, Arduino, LoRA module.

All IoT projects explained with Circuit, Source code.

Beginners Intermediate

Smart Home

การทำงานของระบบควบคุมอัตโนมัติ



Sensors



PIR Motion



HC-SR04



Uno

Relays

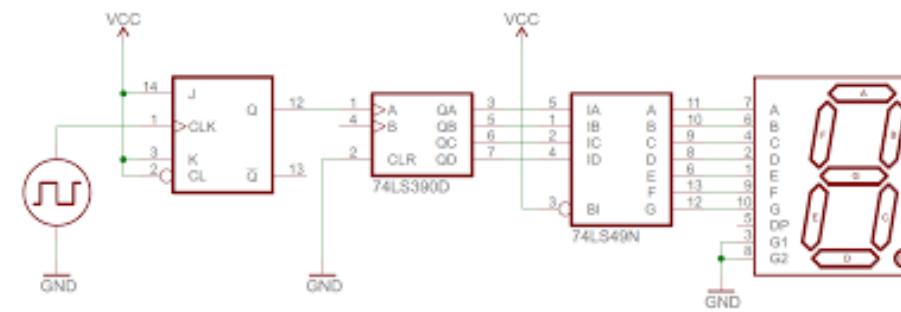
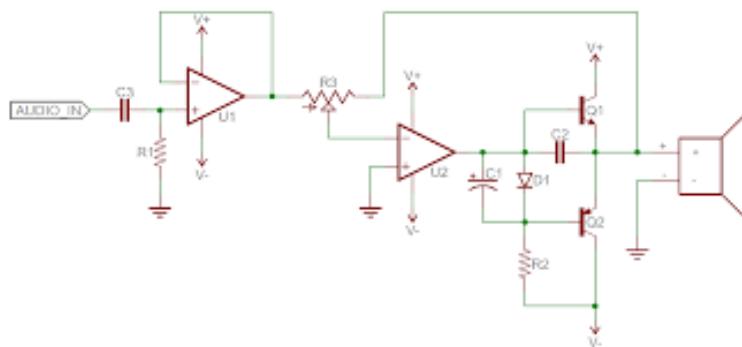
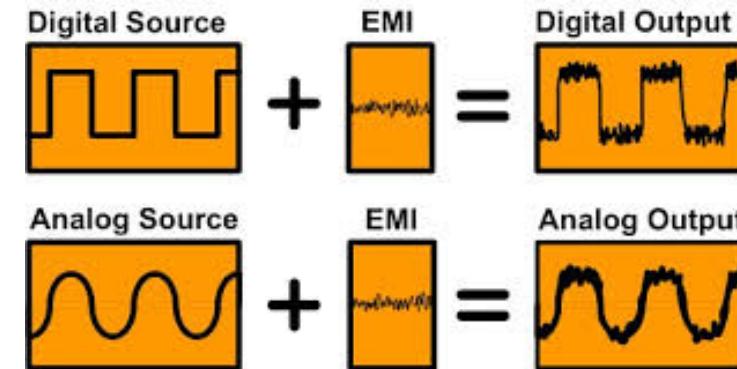
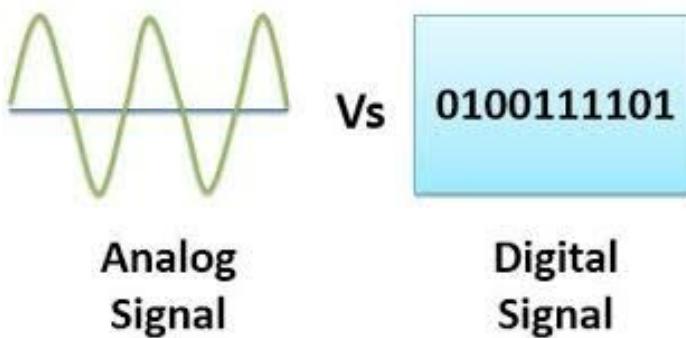


5V Relay

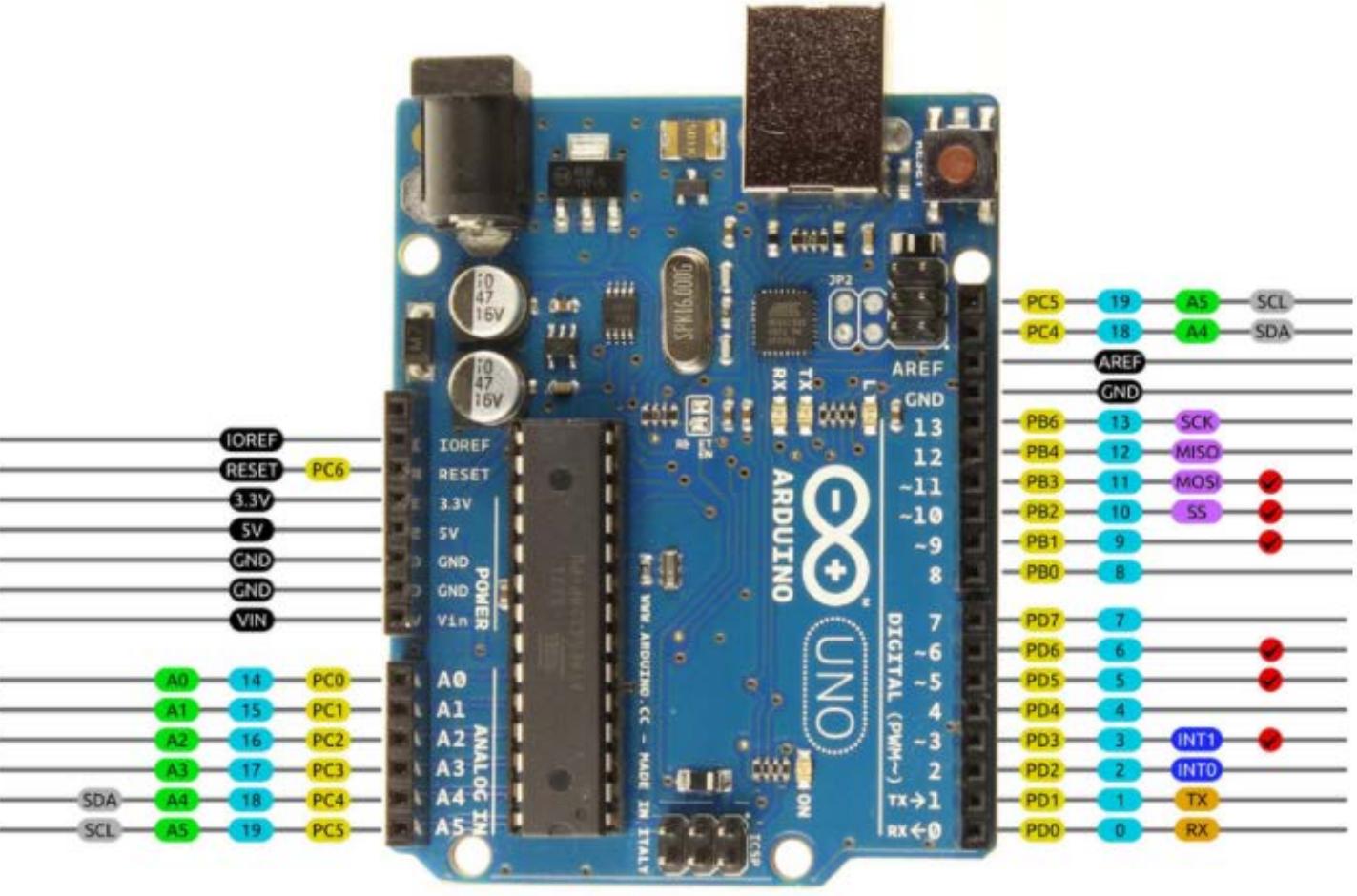
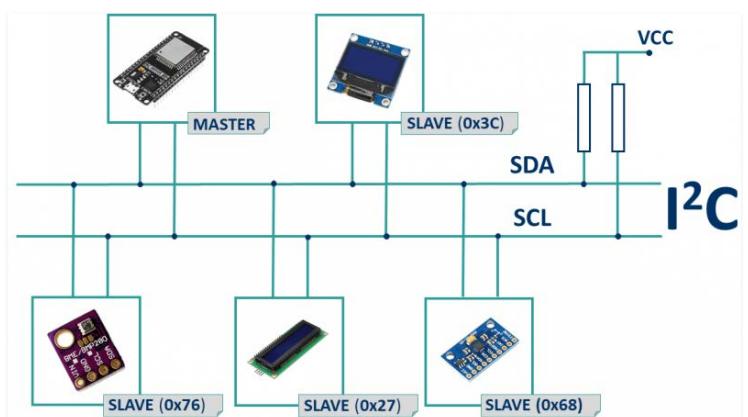
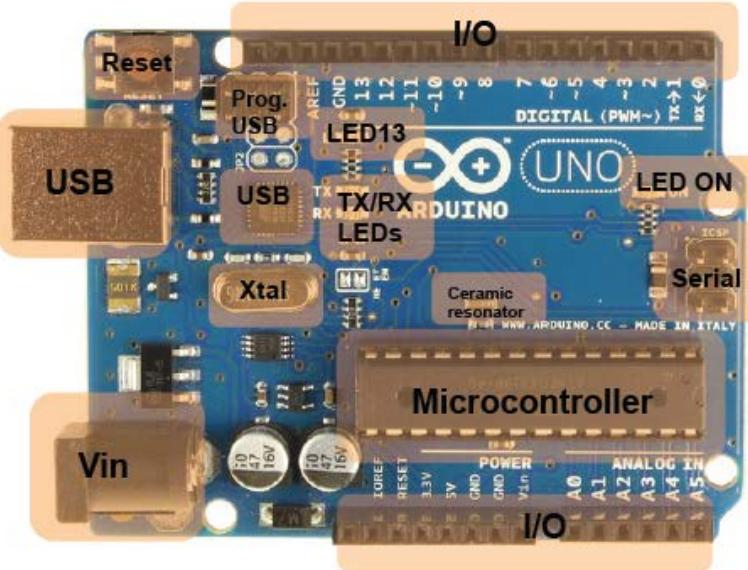


LED - Red

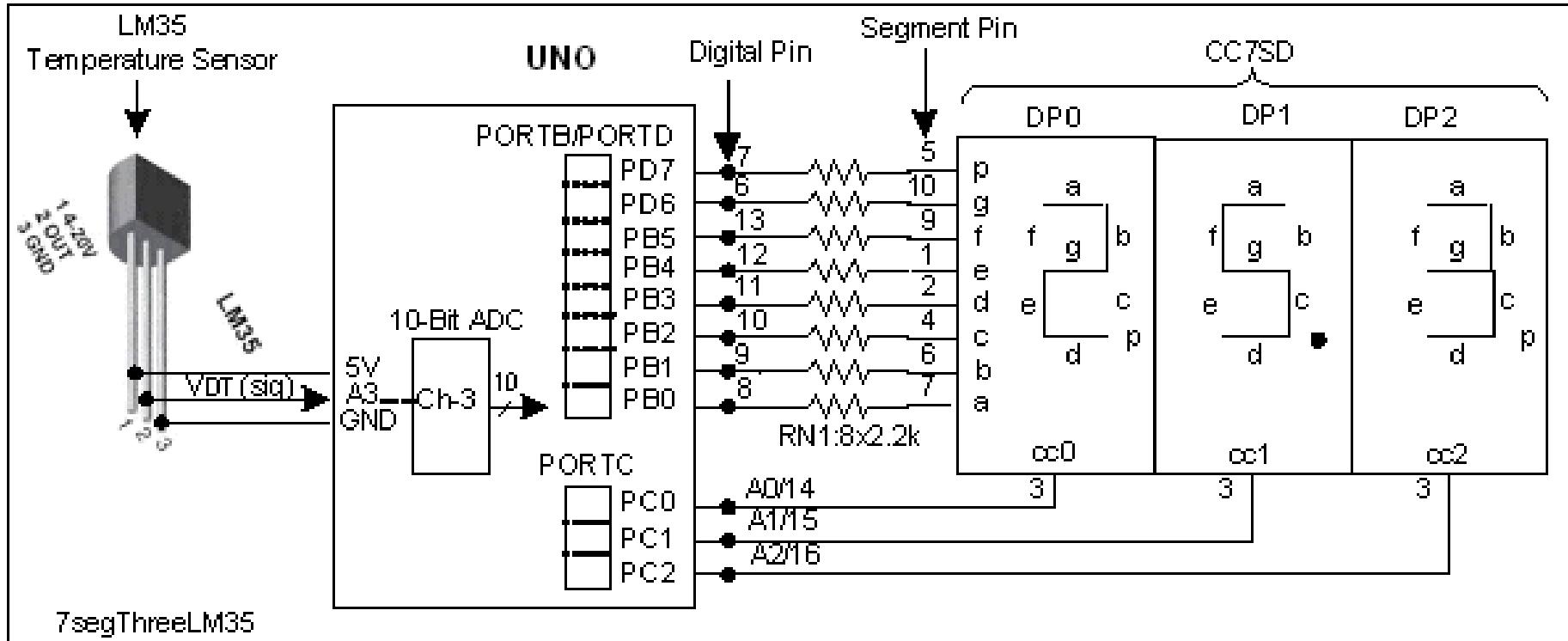
ស័ូល្អានօនាតីក / ឌិជីថល



Arduino UNO



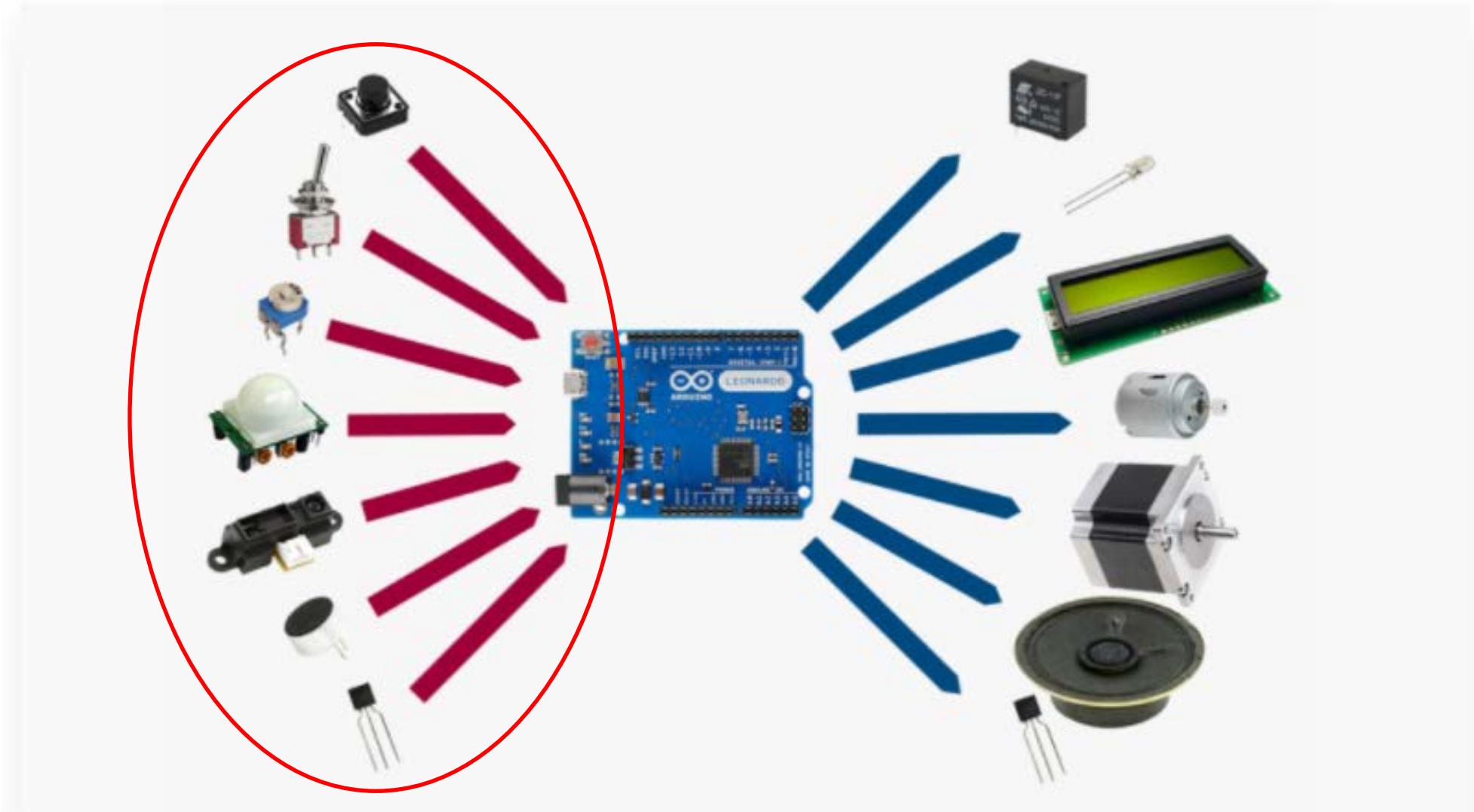
AVR DIGITAL ANALOG POWER SERIAL SPI I²C PWM INTERRUPT

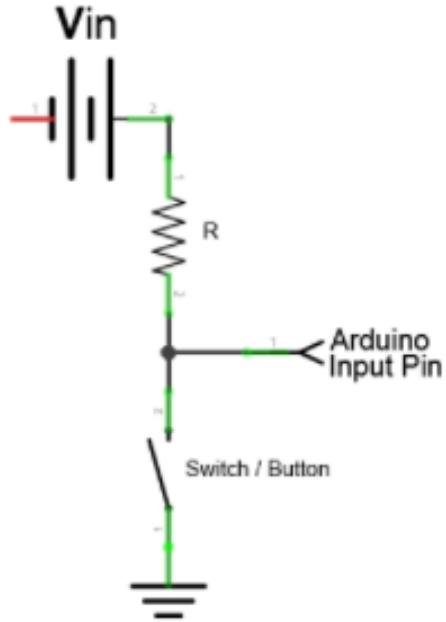


Binary 0101010101 = 2exp.10 (ADC 10 bit) = analog 1,024 steps

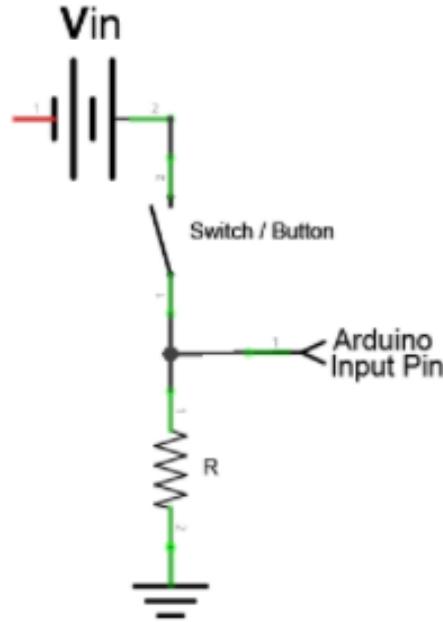
Analog / digital value

Sensor input





Pull-UP

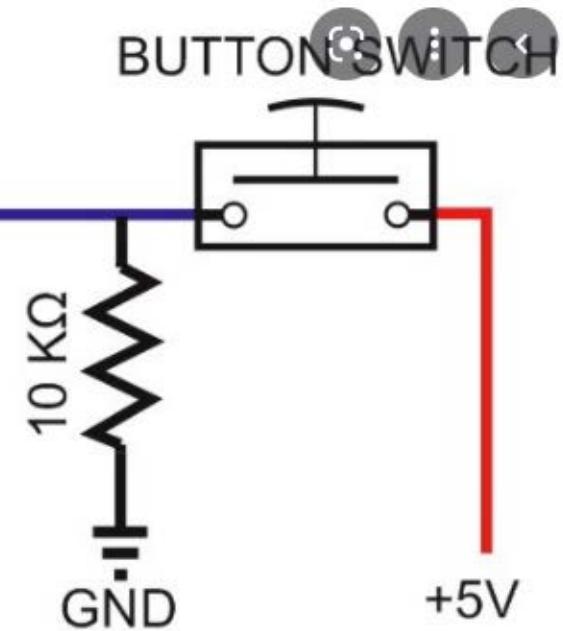


Pull-DOWN

▲ การต่อวงจร Pull-UP / Pull-Down

```
pinMode (SWpin, INPUT_PULLUP);
pinMode (SWpin, INPUT_PULLDOWN);
buttonState = digitalRead (buttonPin);
```

Switch



LDR, PIR, Ultrasonic



LDR เช่นเซอร์วัดความเข้มแสง ใช้ได้กับแสงสว่างที่นำไป output เป็นค่า Analog

analogRead (Apin);

val = digitalRead (PIRpin)



PIR เช่นเซอร์ตรวจจับการเคลื่อนไหวแบบแพสซีพ IR เช่นใช้ปิดเปิดประตูคนเดินเข้าออก



Ultrasonic เช่นเซอร์อัลตร้าโซนิค เสียงความถี่สูง 40Khz ใช้ตรวจจับและคำนวนระยะสิ่งกีดขวางได้

```
digitalWrite(trigPin, LOW); delayMicroseconds(2);
digitalWrite(trigPin, HIGH); delayMicroseconds(10);
digitalWrite(trigPin, LOW); duration = pulseIn(echoPin, HIGH);
```

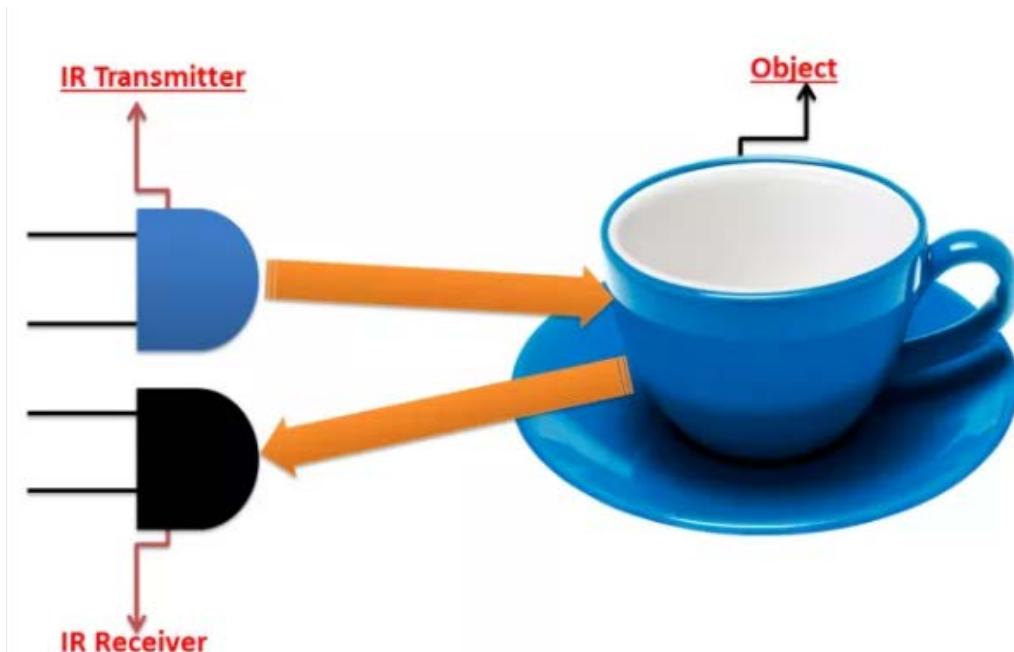


เซนเซอร์แสง Infrared แบบสะท้อน

ใช้ได้กับวัสดุที่มีรูปร่างทึบและลักษณะพื้นผิวที่แสง IR สะท้อนได้แบบอุตสาหกรรมใช้ตัวสีเหลือง (ที่ไม่ไวต่อแสงสว่างภายนอก)
ใช้ในการนับจำนวนได้ดี ตรวจจับวัตถุได้ในระยะ 2-30 cm
คำสั่งที่ใช้ interrupt



IR Sensor Module



Operation of IR Sensor





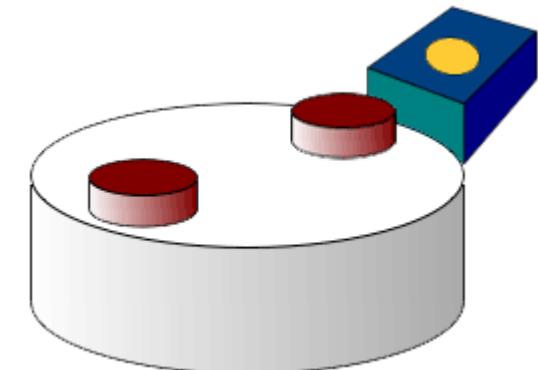
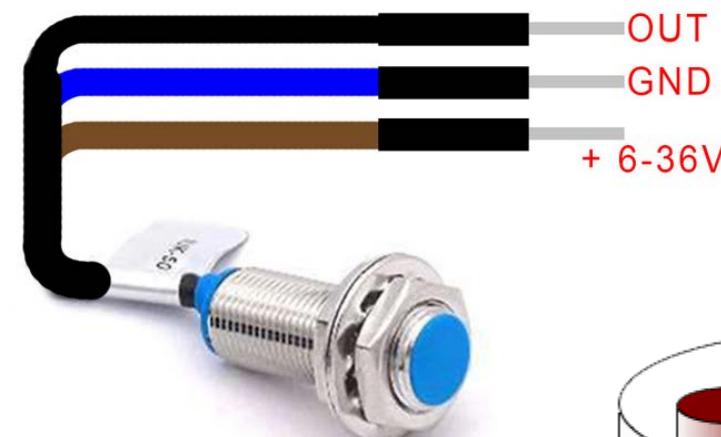
เซนเซอร์แบบตัดลำแสง(เลเซอร์)

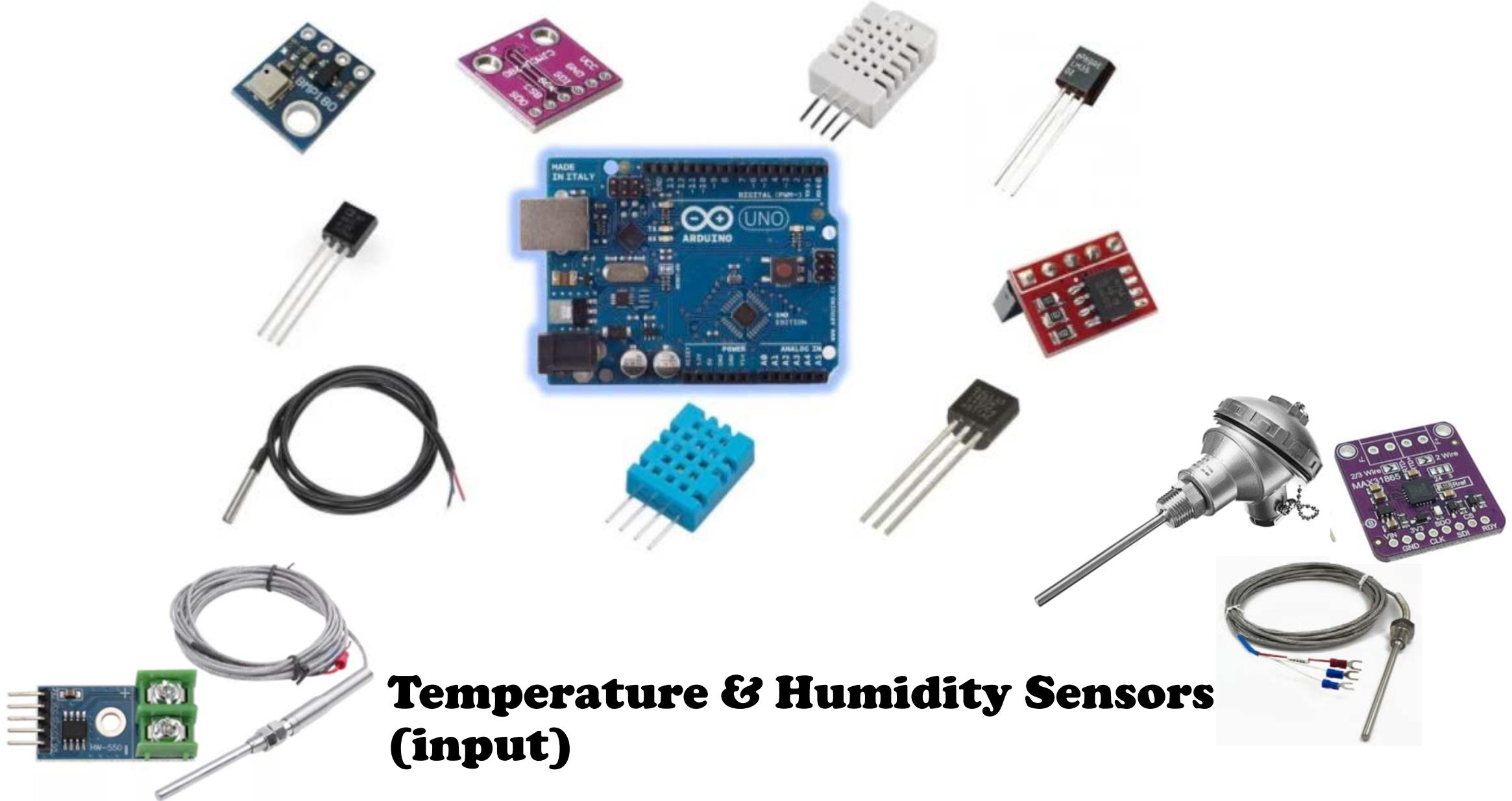
เซนเซอร์ ตัวส่งและรับเลเซอร์ ใช้ตรวจจับวัตถุที่ตัดลำแสง
สำหรับการนับจำนวนได้ในระยะไกลสูงสุด 20 ม. แต่เซนเซอร์ทั้ง2
ความมีการ alignment ที่ดี คำสั่งที่ใช้ interrupt



เซนเซอร์ Proximity แม่เหล็กตรวจจับโลหะ

เซนเซอร์ชนิด Hall effect ใช้ตรวจจับโลหะที่อยู่ใกล้ได้ ควรใช้แหล่งจ่ายไฟภายนอกในช่วงตั้งแต่ 6V ขึ้นไป คำสั่งที่ใช้ interrupt



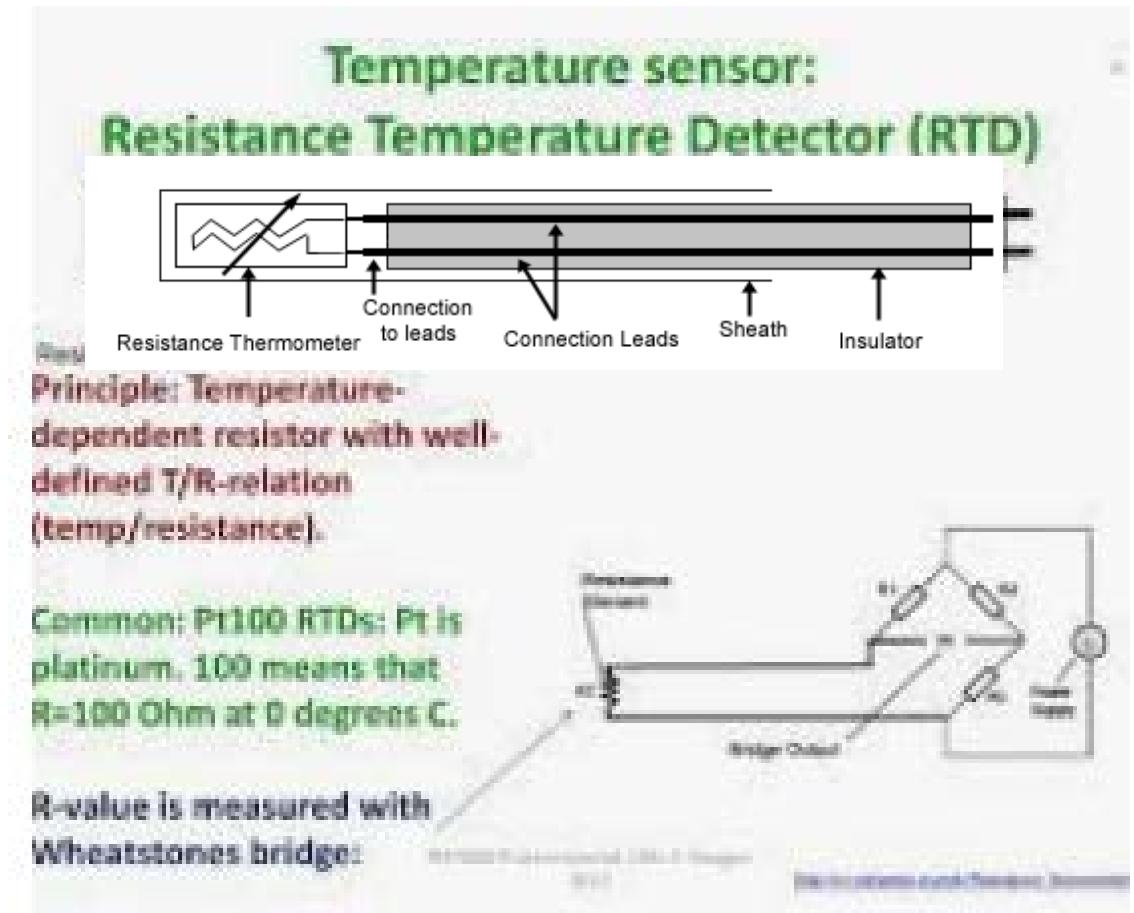
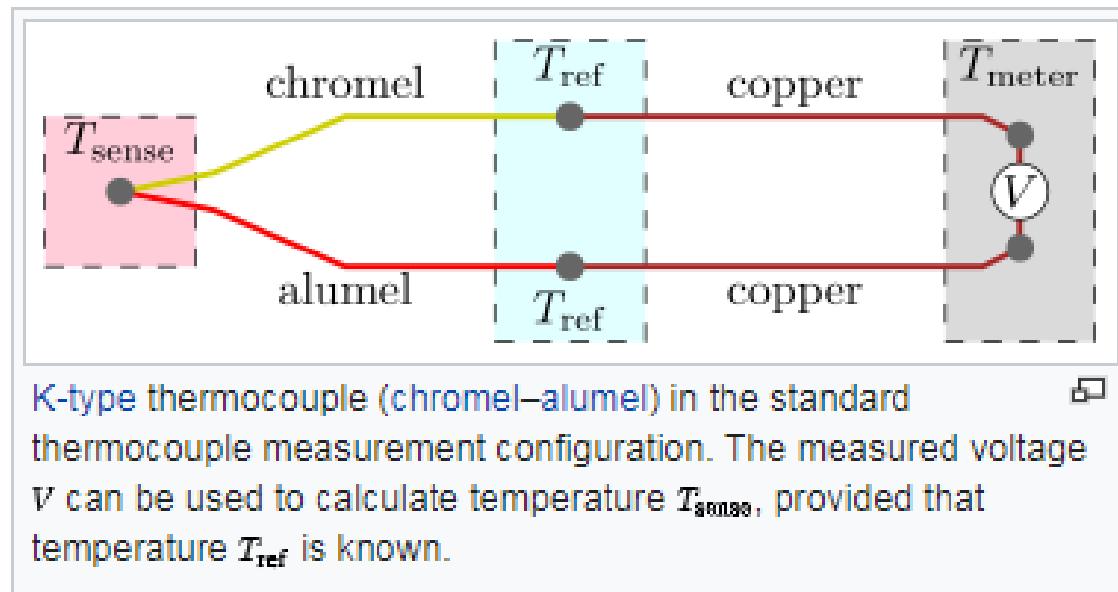


Temperature & Humidity Sensors (input)

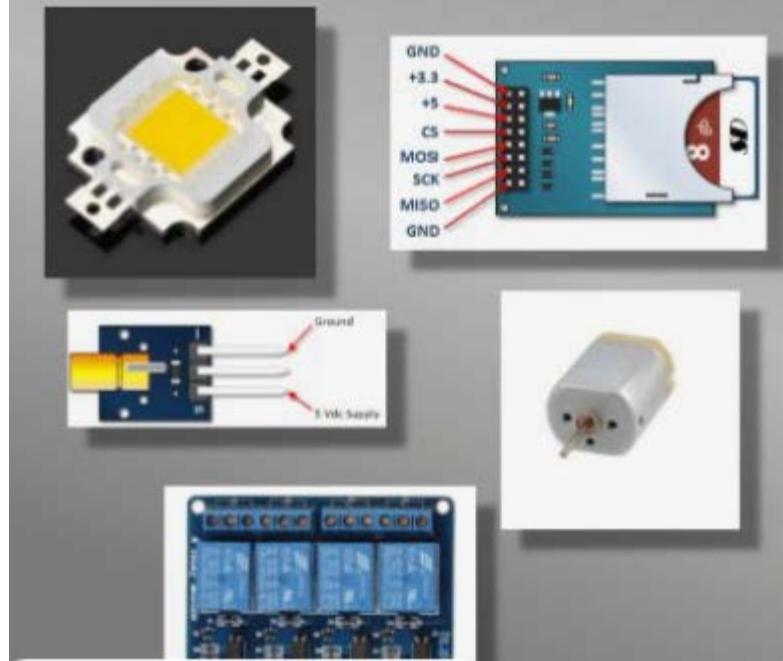
Sensor	Voltage supplied /V	Temp. range / C	Humidity Range / %	Tolerance (+/- %)
DHT11	3 - 5.5	0-50	20-90	2, 5
DHT22	3 – 6	-40 – 80	0-100	0.5, 2
LM35	4 – 30	-55 – 150	n/a	0.5
BMP180	3.3 – 5	0-65	Barometric Pressure	0.5
TMP36	2.7 - 5.5	-40 – 125	n/a	1
LM75	3 - 5.5	-25 – 125	n/a	2
BME280	3.3 – 5	-40 – 85	0-100 , BP	1, 3
DS18b20	3 - 5.5	-55 – 125	n/a	0.5
TC K-Type	3 - 5.5	0 - 800	n/a	0.25
RTD PT100		-50 - 300	n/a	0.15-2



Thermocouple and RTD structure



Arduino Output Devices

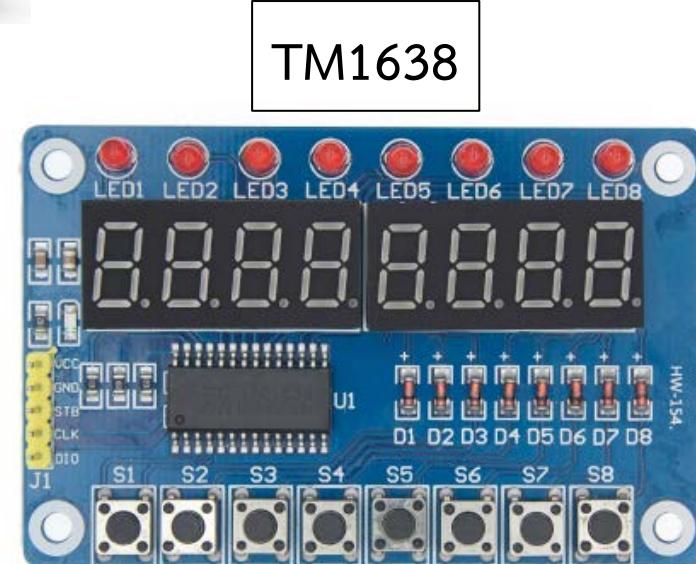
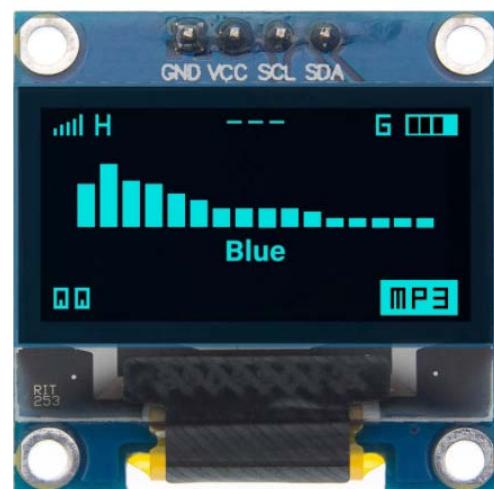
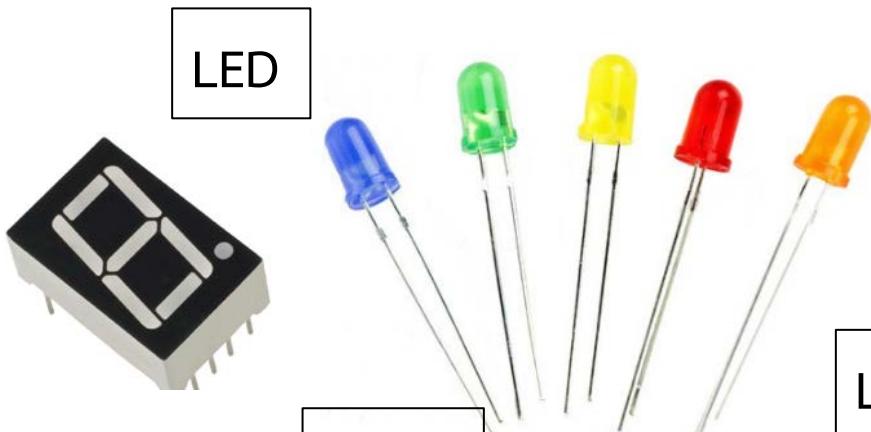


Output device

LCD , LED, 7-segment Display, OLED

- LCD (Liquid Crystal Display) หน้าจอแสดงตัวอักษร มีหลายสี สี่ยอดนิยมคือสีฟ้าหรือสีเขียว และเป็นแบบ 16 ตัวอักษร 2 บรรทัด ทำงานร่วมกับบอร์ด I2C ทำให้ต่อPinลดลงได้ คือ Vcc, Grd, SCA, SCL และใช้ Library: Liquidcrystal_I2C ในการเขียนโปรแกรม
- LED (Light Emitting Diode) เป็นสารกึ่งตัวนำPและN มีPin: Anode (ขายาว) และ Cathode (ขาสั้น) ในการใช้งานต้องต่อR ที่ขา Anode ก่อนรับสัญญาณ
- 7-Segment Display ใช้IC เบอร์ TM1637 หรือ TM1638 เป็นหน้าจอแสดงผลเลข 8 จำนวน 4 หรือ 8 หลัก มีขา DIO และ CLK ใช้งานง่ายเพราะPinน้อย การใช้งานต้องเรียกใช้ Library ในโปรแกรมอาร์ดูโน่
- OLED (Organic Light Emitting Diode) LEDแบบสารอินทรีย์ ใช้ขา SDA, SCL, Vcc, Grd กินกระแสไฟฟ้าต่ำกว่า LCD

อุปกรณ์แสดงผล output



อุปกรณ์แสดงผลด้วยเสียง

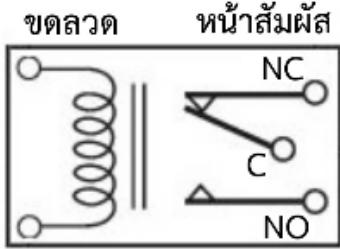
digitalWrite (BZpin, HIGH);
digitalWrite (BZpin, LOW);



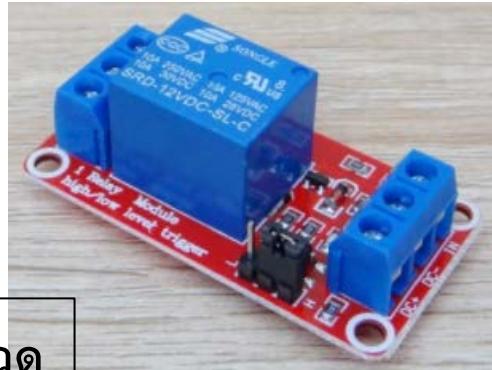
tone (PZpin, frequency);
tone (PZpin, frequency, duration);



- **Active Buzzer** ป้อนไฟ + และ - เสียงจะดังทันที ความถี่ตามผู้ผลิต
- **Passive Buzzer** สัญญาณเสียงมาจากการคำสั่ง tone ในโปรแกรมที่ระบุความถี่ ตั้งแต่เสียงทุมจนถึงเสียงแหลม บางทีเรียกว่า Piezo ทำเป็นลำโพงเสียงให้ MCU ร้องเพลงได้

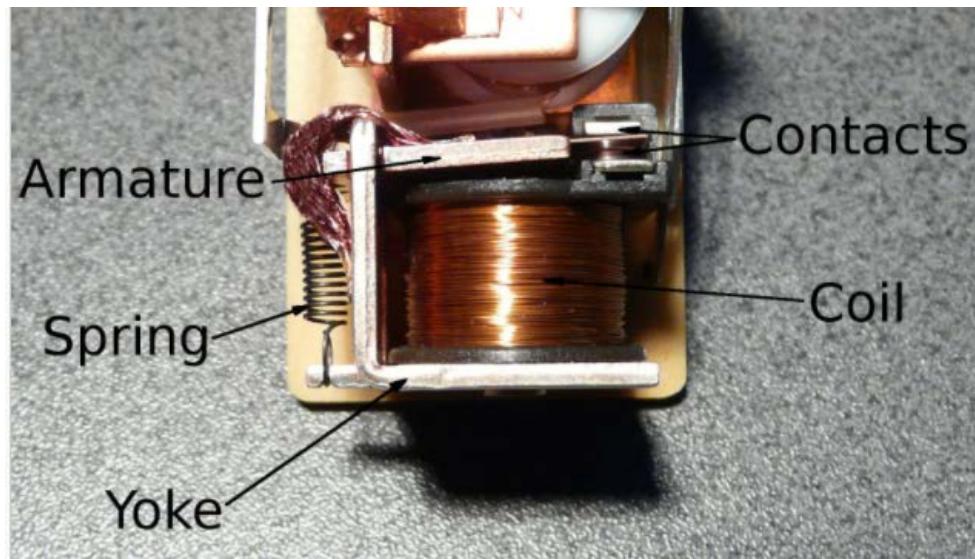


รูป ส่วนประกอบของรีเลย์



แบบขาด漉ด
หรือ Coil

RELAY

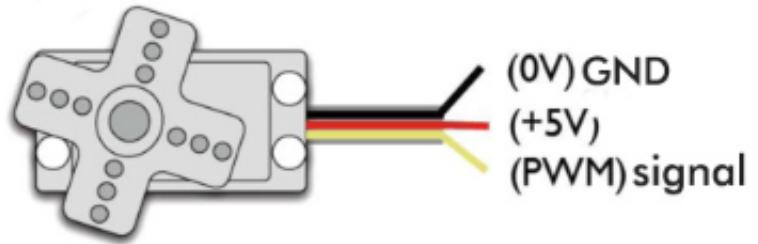


แบบSolid State มี
ทั้ง AC, DC



```
digitalWrite (Relaypin, HIGH);  
digitalWrite (Relaypin, LOW);
```

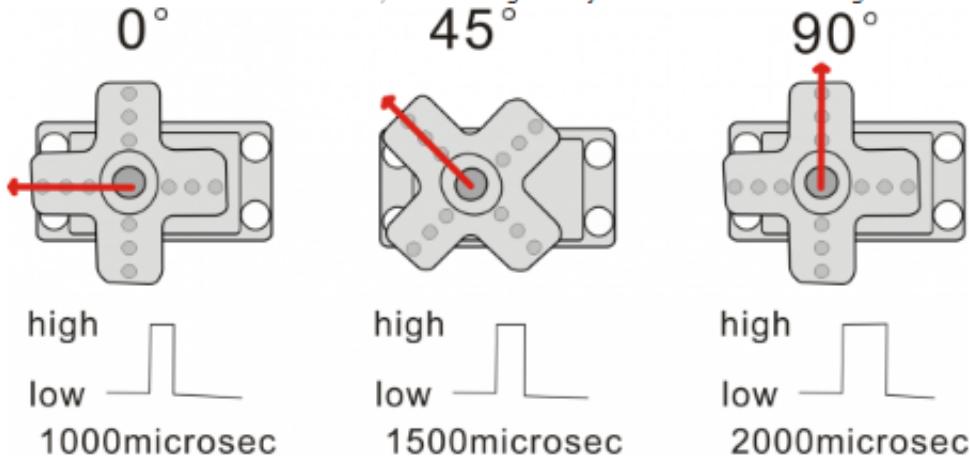
มอเตอร์ และ เซอร์โวมอเตอร์



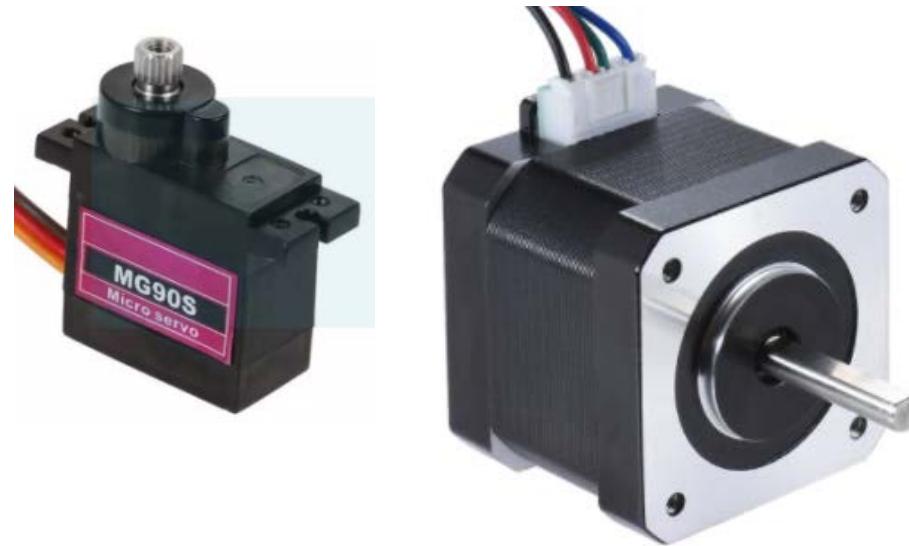
The rotation angle of the servo motor is controlled by regulating the duty cycle of the PWM(Pulse-Width Modulation) signal.

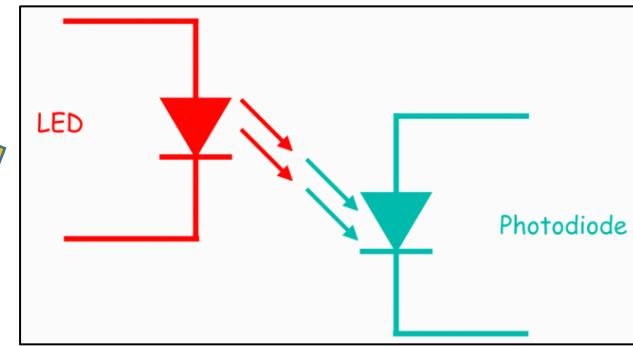
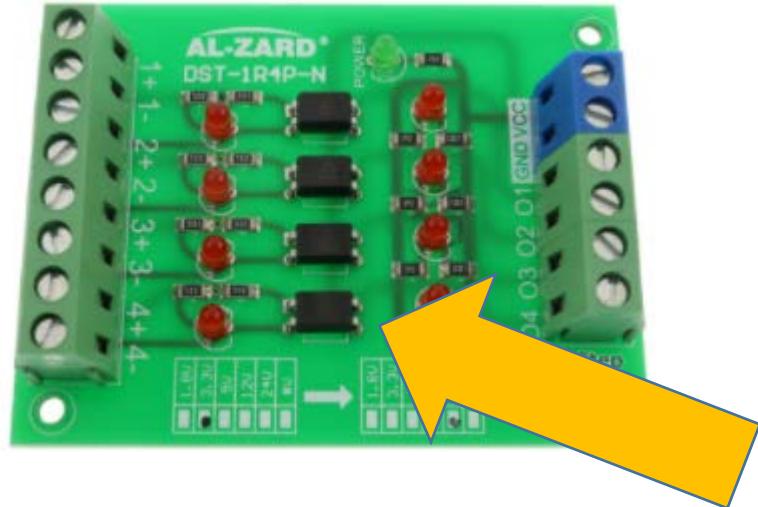
The standard cycle of the PWM signal is fixed at 20ms (50 Hz), and the pulse width is distributed between 1ms-2ms. The width corresponds the rotation angle from 0° to 90°.

But note that for different brand motor, the same signal may have different rotation angle.



Servo.write (degree between 0-180);

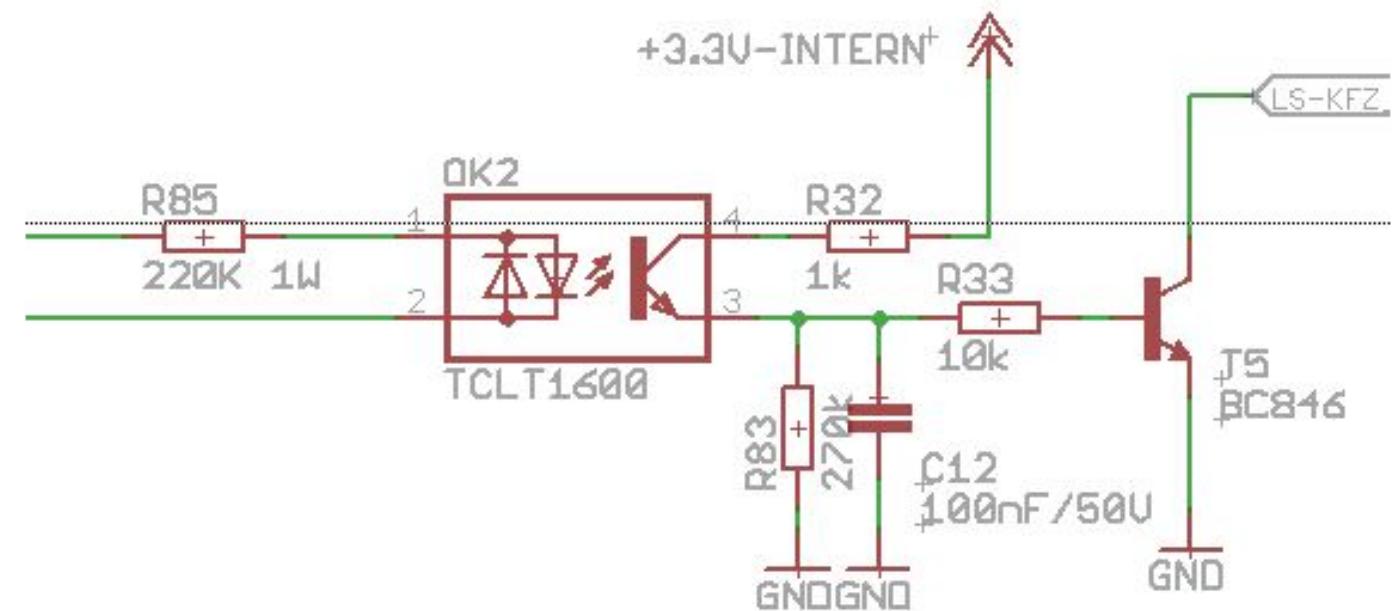
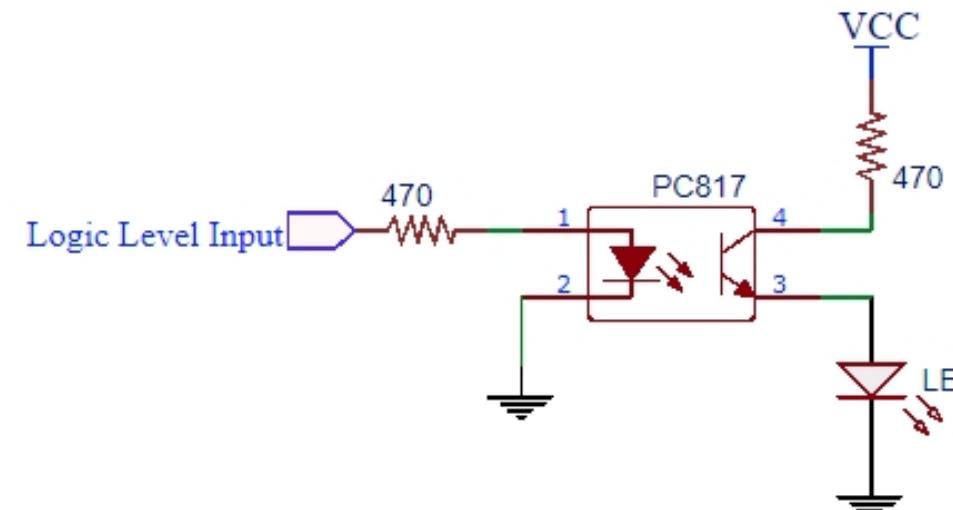
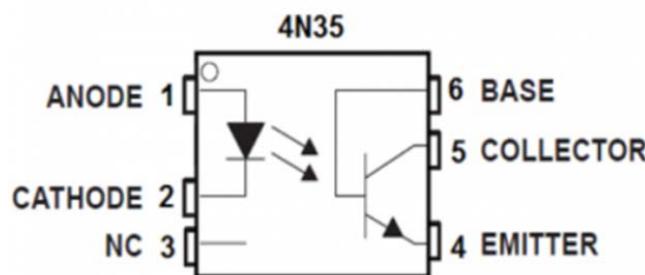
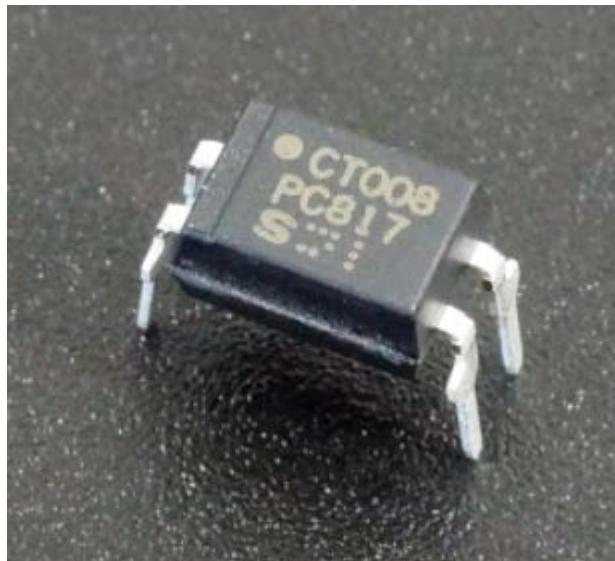




Interfacing device (Input & output)

Optocoupler (DC)

PC817

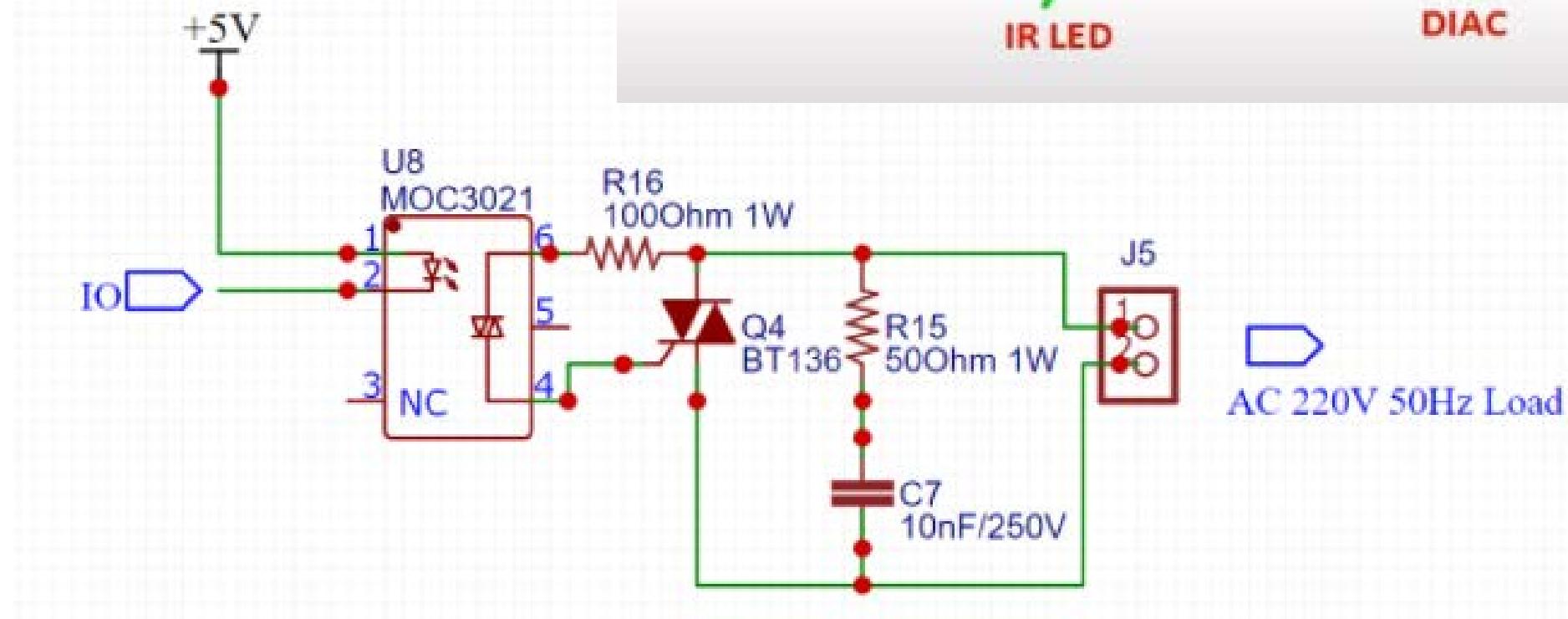
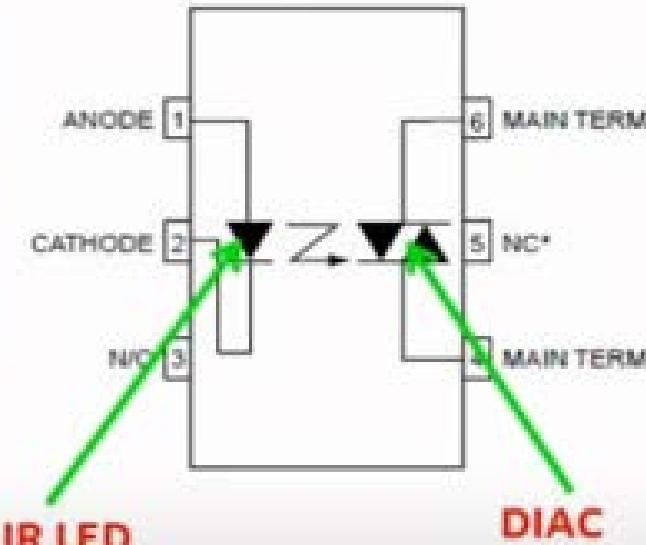


Optocoupler (AC)

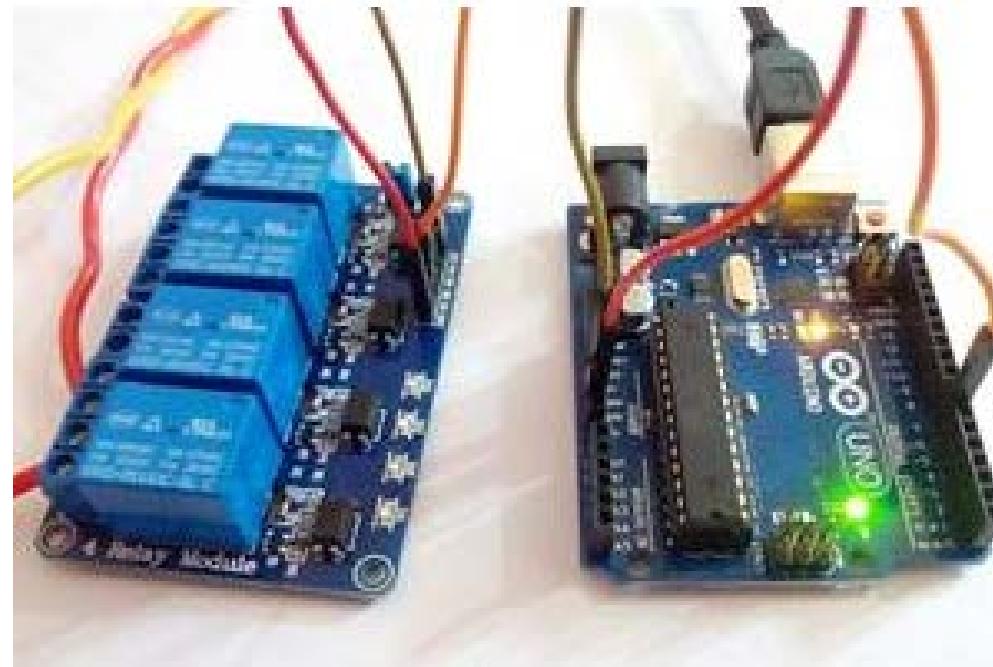
MOC3021



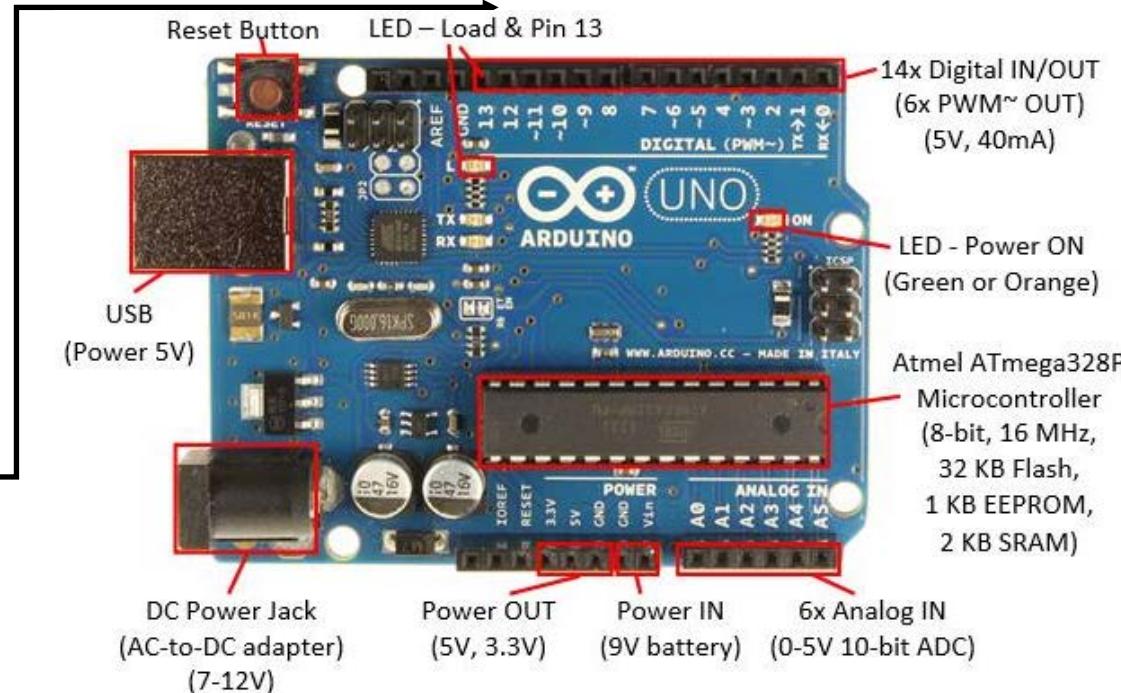
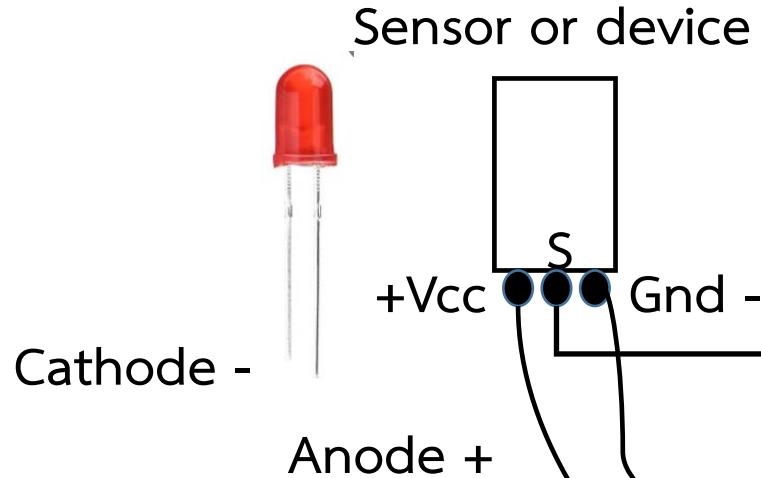
MOC3021



พื้นฐานการเขียนคำสั่ง arduino และ การต่ออุปกรณ์



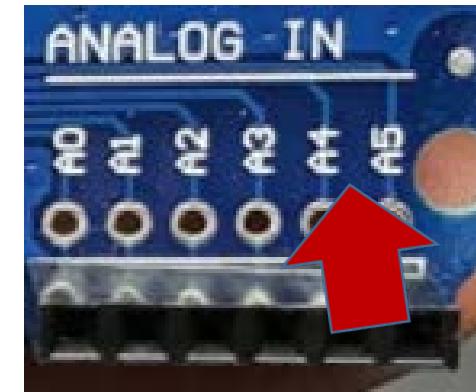
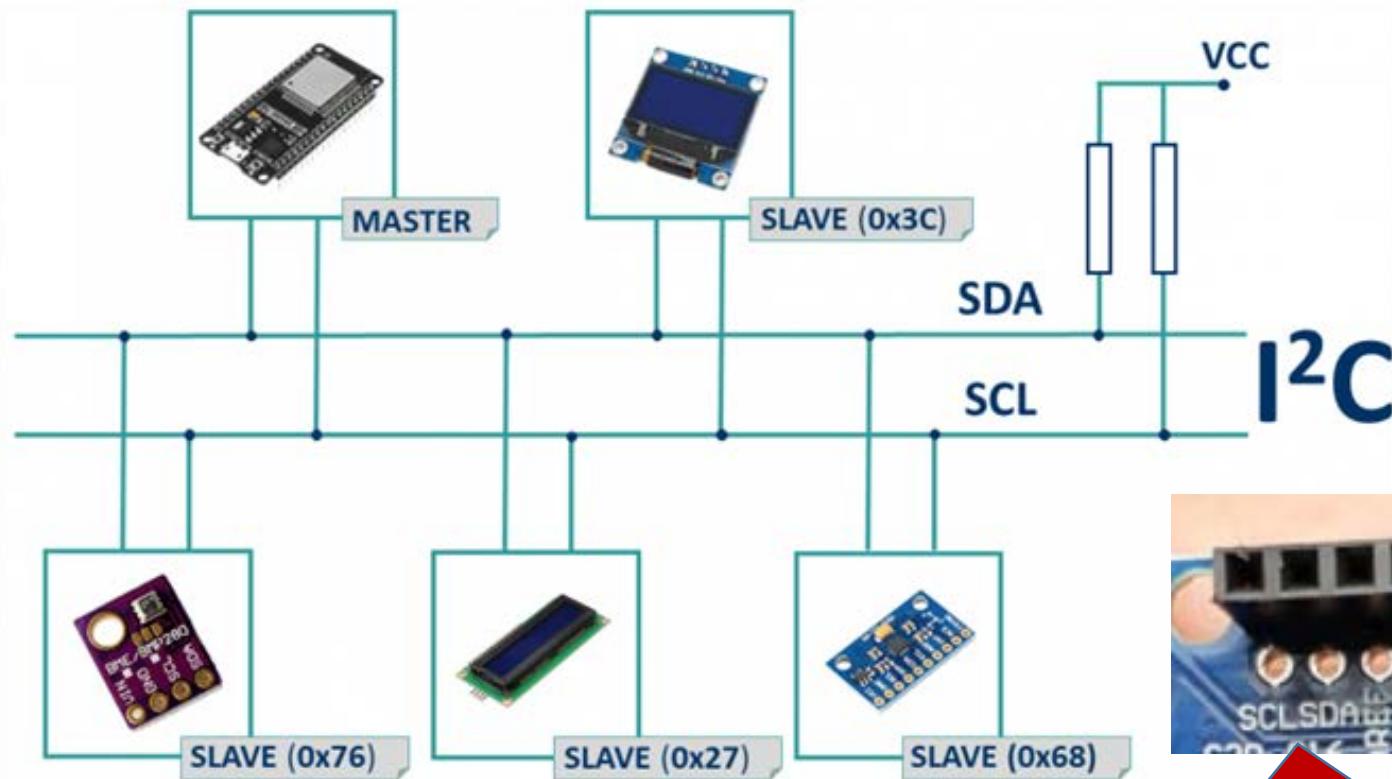
ต่อขา S หรือ IN เข้า Port D หรือ A
สำหรับ LED ต่อกับขา R ที่ Anode (ด้านขาขวา)



ต่อ Port: Ground เสมอ (แม้ใช้แหล่งจ่ายไฟภายนอก
ไม่ต่อ ไม่ทำงาน) Led ต่อกับขา cathode (ด้านขาล้ำ)

ต่อ Port: 3.3V or 5V ยกเว้นใช้ไฟสูงกว่า 5V ต่อ Vcc
กับแหล่งจ่ายภายนอก และห้ามต่อไฟ Vcc เข้าพอร์ต 5V
ยกเว้น Ground เฉพาะพอร์ต Vin ต่อไฟเข้าได้ 7-12V





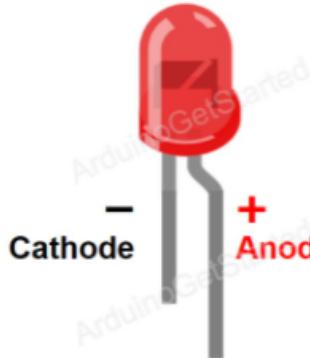
อุปกรณ์ที่มีขาพอร์ตสื่อสาร SDA, SCL ต่อตรงที่พอร์ตระบุไว้ หรือ
ขา A4 (SDA) และ A5 (SCL) ที่บอร์ด Uno, Nano

LED (Light Emitting Diode): digital or PWM output

Pinout

LED includes two pins:

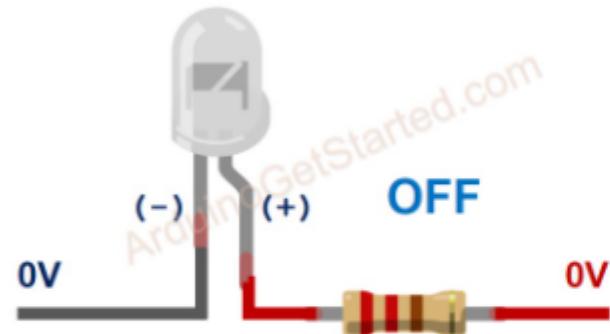
- ◆ Cathode(-) pin needs to be connected to **GND** (0v)
- ◆ Anode(+) pin is used to control LED's state



```
digitalWrite(Led1, HIGH);  
digitalWrite(Led1, LOW);
```

After connecting the cathode(-) to **GND**:

- ◆ If connecting **GND** to the anode(+), LED is **OFF**.
- ◆ If connecting **VCC** to the anode(+), LED is **ON**.



```
val=analogRead(Potpins);  
analogWrite(Ledpin, val/4);
```



PWM output

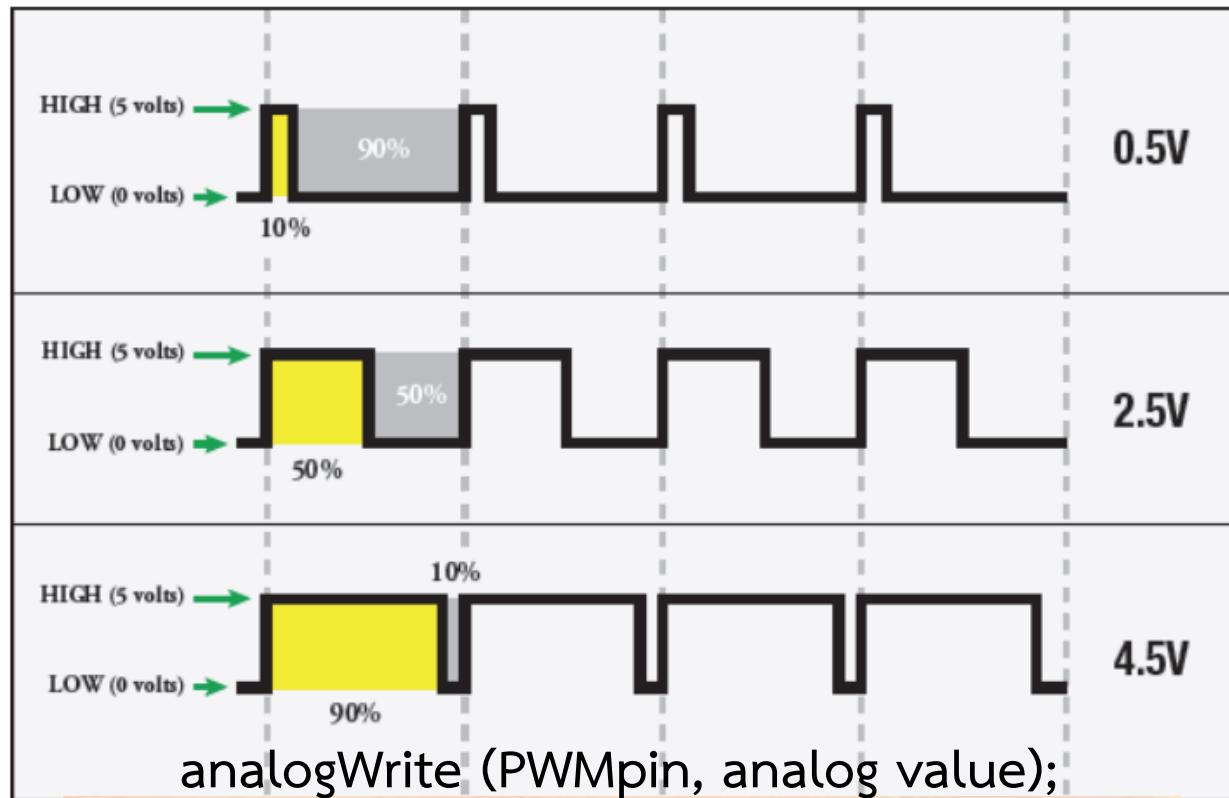
Besides, if generating a PWM signal to the anode(+), the brightness of LED is changed according to PWM value (described in detail in [this tutorial](#))

PWM (pulse width modulation) output

Pinout

Piezo Buzzer usually has two pins:

- ◆ Negative (-) pin needs to be connected to **GND (0v)**
- ◆ Positive (+) pin receive the control signal from Arduino



Download the Arduino IDE



ARDUINO 1.8.9

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, for Windows XP and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10
[Get](#)

Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

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

The screenshot shows the Arduino IDE interface with two sketches open in separate windows:

- Sketch: sketch_feb02a | Arduino 1.6.7**
 - Code:

```
1 void setup() {  
2     // put your setup code here, to run once:  
3 }  
4  
5 void loop() {  
6     // put your main code here, to run repeatedly  
7 }  
8  
9 }
```
- Sketch: sketch_sep03a**
 - Code:

```
1 void setup() {  
2     // put your setup code here, to run once:  
3 }  
4  
5 void loop() {  
6     // put your main code here, to run repeatedly  
7 }  
8  
9 }
```

The Arduino board selection at the bottom of the screen is set to "SP8266 Module, 80 MHz, 40MHz, DIO, 115200, 512K (64K SPIFFS), ck, Disabled, None on COM7".

void

[Data Types]

Description

```
34
35 void setup() {
36     // initialize the LED pin as an output:
37     pinMode(ledPin, OUTPUT);
38     // initialize the pushbutton pin as an input:
39     pinMode(buttonPin, INPUT);
40 }
41
42 void loop() {
```

The `void` keyword is used only in function declarations. It indicates that the function is expected to return no information to the function from which it was called.

```
41
42 void loop() {
43     // read the state of the pushbutton value:
44     buttonState = digitalRead(buttonPin);
45
46     // check if the pushbutton is pressed.
47     // if it is, the buttonState is HIGH:
48     if (buttonState == HIGH) {
49         // turn LED on:
```

โครงสร้างโปรแกรม

1. ส่วนประกาศตัวแปร และ library
2. ส่วนตั้งค่าตัวแปร (setup)
3. ส่วนโปรแกรม ทำงานอย่าง ต่อเนื่อง ไม่รู้จบ จนกว่าจะปิดเครื่อง (loop)
4. ส่วนของโปรแกรมย่อย/ถูกเรียกใช้จากส่วนต่าง ๆ (functions)

```
/*
Blink

Turns an LED on for one second, then off for one second,
repeatedly.

*/
int LED = 13; // define LED 13

// the setup function runs once when you press reset or power the
board

void setup() {
    // initialize digital pin LED_BUILTIN as an output.
    pinMode(LED, OUTPUT);
}

// the loop function 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
}
```

#include

[Further Syntax]

Description

`#include` is used to include outside libraries in your sketch. This gives the programmer access to a large group of standard C libraries (groups of pre-made functions), and also libraries written especially for Arduino.

The main reference page for AVR C libraries (AVR is a reference to the Atmel chips on which the Arduino is based) is [here](#).

Note that `#include`, similar to `#define`, has no semicolon terminator, and the compiler will yield cryptic error messages if you add one.

InterruptTest

```
1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3
4 #define interruptPin D2
5 volatile boolean interruptFlag =0;
6 LiquidCrystal_I2C lcd(0x3f,16,2);
7 unsigned int QTY=0;
8
9 void setup() {
10 Serial.begin(115200);
```

Library

#define

[Further Syntax]

```
// - Adafruit Unified Sensor Lib: https://github.com/adafruit/Adafruit\_Sensor
#include "DHT.h"

#define DHTPIN 2      // Digital pin connected to the DHT sensor
// Feather HUZZAH ESP8266 note: use pins 3, 4, 5, 12, 13 or 14 --
// Pin 15 can work but DHT must be disconnected during program upload.
```

Description

`#define` is a useful C++ component that allows the programmer to give a name to a constant value before the program is compiled. Defined constants in arduino don't take up any program memory space on the chip. The compiler will replace references to these constants with the defined value at compile time.

int

[Data Types]

Description

Integers are your primary data-type for number storage.

On the Arduino Uno (and other ATmega based boards) an int stores a 16-bit (2-byte) value. This yields a range of -32,768 to 32,767 (minimum value of -2^{15} and a maximum value of $(2^{15}) - 1$). On the Arduino

```
6
7 int tempin=0;
8 float tempc;
9 int reading;
10
11 void setup()
12 {
13 lcd.begin();
```

float

[Data Types]

Description

Datatype for floating-point numbers, a number that has a decimal point. Floating-point numbers are often used to approximate analog and continuous values because they have greater resolution than integers. Floating-point numbers can be as large as 3.4028235E+38 and as low as -3.4028235E+38. They are stored as 32 bits (4 bytes) of information.

ชนิดข้อมูล	การเก็บข้อมูล	ขนาด
boolean	จริง (True) หรือ เท็จ (False)	1 บิต
char	ตัวเลข หรือตัวอักษร	1 ไบต์ ใส่ค่าได้ตั้งแต่ -128 ถึง 127
unsigned char	ตัวเลข หรือตัวอักษร	1 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง 255
byte	ไบต์	1 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง 255
int	ตัวเลขจำนวนเต็ม	2 ไบต์ ใส่ค่าได้ตั้งแต่ -32,768 ถึง 32,767
unsigned int	ตัวเลขจำนวนเต็ม	2 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง $65,535 (2^{16} - 1)$
long	ตัวเลขจำนวนเต็มที่มีความยาว	4 ไบต์ ใส่ค่าได้ตั้งแต่ -2,147,483,648 ถึง 2,147,483,647
unsigned long	ตัวเลขจำนวนเต็มที่มีความยาว	4 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง $4,294,967,295 (2^{32} - 1)$
float	ตัวเลขทศนิยมใช้ในการคำนวณ	4 ไบต์ ใส่ค่าได้ตั้งแต่ $3.4028235E+38$ ถึง $-3.4028235E+38$ มีทศนิยมได้ 6 ถึง 7 ตำแหน่ง
double (เฉพาะบอร์ด Arduino Due)	ตัวเลขทศนิยมที่มีความยาวและต้องการความแม่นยำ	8 ไบต์ ใช้ในการคำนวณที่ต้องการประสิทธิภาพสูง
String	ข้อความ	ไม่ระบุ

if

[Control Structure]

```
27 Serial.print ("Temp celcius = ") ;
28 Serial.println(celcius) ;
29 lcd.setCursor(0,1) ;
30 lcd.print ("Temp = ") ;
31 lcd.setCursor(9,1) ;
32 lcd.print (celcius) ;
33 if(celcius >=20) {
34     digitalWrite (ledpin,HIGH) ;
35 }
36 else {
37     digitalWrite (ledpin,LOW) ;
38 }
39 delay (1000) ;
```

Description

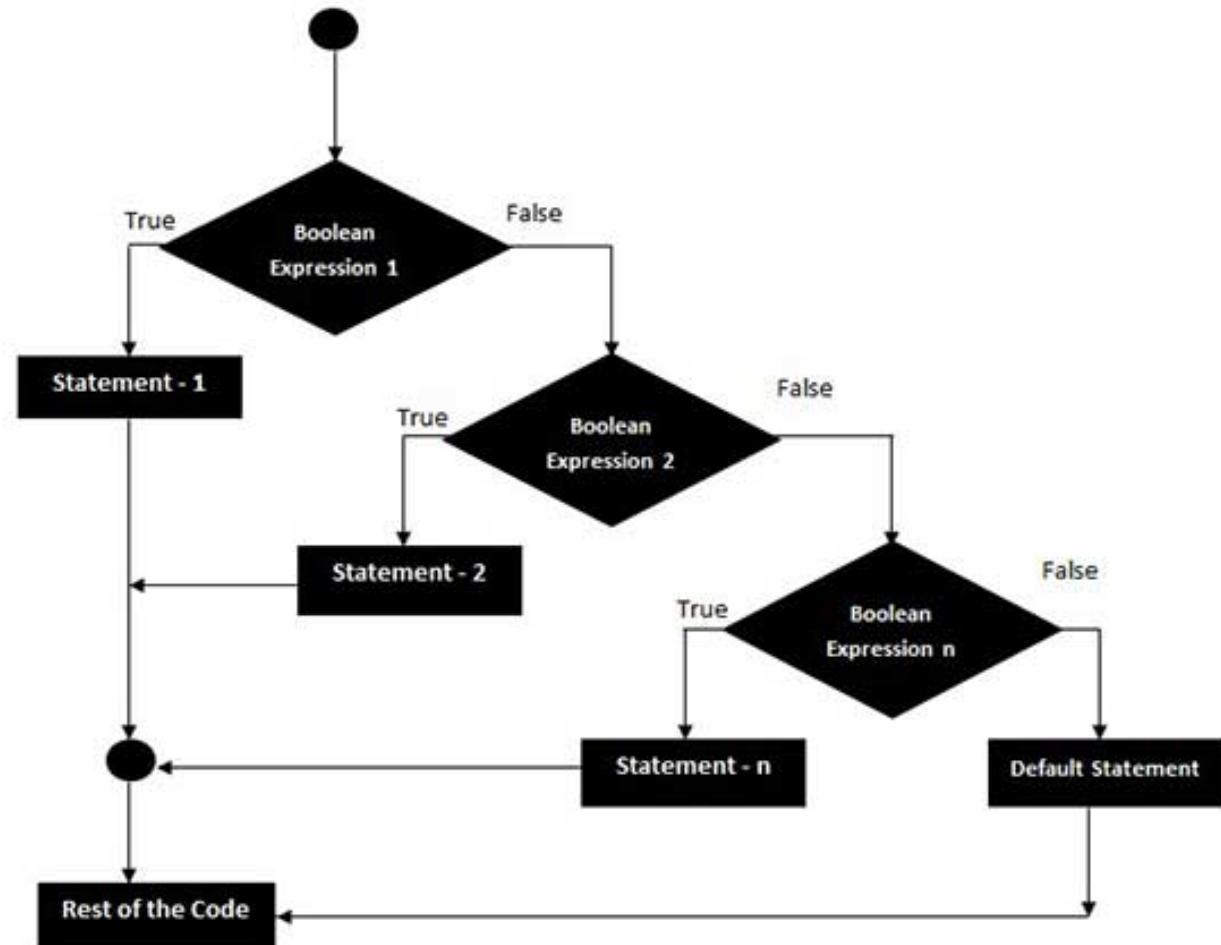
The **if** statement checks for a condition and executes the proceeding statement or set of statements if the condition is 'true'.

```

1 void setup() {
2   Serial.begin(9600);
3 }
4
5 void loop() {
6   int temp = 30;
7   if (temp >= 35) {
8     Serial.println("temp is hot");
9   }
10 }

```

Flow Diagram



```

1 void setup() {
2   Serial.begin(9600);
3 }
4
5 void loop() {
6   int temp = 30;
7   if (temp > 35) {
8     Serial.println("temp is hot");
9   }else if(temp > 25){
10    Serial.println("temp is Good");
11  }else{
12    Serial.println("temp is Cool");
13  }
14 }

```

Code: read, write

```
digitalRead(2);
//pin or parameter, reading result is logic 0 (LOW) or 1(HIGH)
analog Read(A0);
//analog pin selected from A0 to A5, result is value between 0-1023.
The ADC on the Arduino is a 10-bit ADC meaning it has the ability
to detect 1,024 ( $2^{10}$ ) discrete analog levels.
```

```
digitalWrite(3,HIGH); // result is logic 1 (5V) at Pin3
digitalWrite(3,LOW); //result is logic 0 (0V) at Pin3
analogWrite(6,200); //result is analog value (PWM wave) to Pin6 value 200,
(from 0 to 255) the PWM function is available on pins 3, 5, 6, 9, 10, and 11
(Uno)
```

Operand

\geq (มากกว่า หรือ เท่ากับ)

```
if (Temp $\geq$ 28) {  
    digitalWrite(led,HIGH);  
}
```

= (กำหนดค่าตัวแปร)

```
107 float tt = 0;  
108 float hh = 0;
```

\leq (น้อยกว่า หรือ เท่ากับ)

```
else if (Temp $\leq$ 25) {  
    digitalWrite(led,LOW);  
}
```

```
19 void blink() {  
20     state = !state;  
21 } = ! x (ค่าตรงข้ามกัน)
```

$\&\&$ (และ)

```
KeyTmem = keypad.waitForKey();  
if ((KeyTmem=='#')  $\&\&$  (KeySset!='7')  $\&\&$  (KeySset!='8'))
```

$\|$ (หรือ)

```
if ((humidity == 55)  $\|$  (humidity == 58)  $\|$  (humidity == 65))
```

$= =$ (มีค่าเท่ากับ)

```
delay(1000);  
if ((QTY-Qten)==10) {  
    digitalWrite(Buz,HIGH);
```

$!=$ (ไม่เท่ากับ)

```
... -----  
while (WiFi.status() != WL_CONNECTED)
```

The End – บันทึก

