คำสั่งต่างๆในArduino

การเริ่มตั้งค่าไมโครคอนโทรลเลอร์ทุกครั้งต้องเริ่มด้วย

```
void setup()
{
 //ค่าเริ่มต้น
}
void loop()
                        {
 //คำสั่งต่างๆในการทำงาน
}
```

//.....ตามด้วยข้อความ..... ➔ใช้เขียนเพื่อจดบันทึกหรือคอมเม้นท์โปรแกรมแต่ละบรรทัด

ทุกคำสั่งต้องลงท้ายด้วย ; เสมอ ยกเว้นบางคำสั่ง

<u>คำสั่งพื้นฐานในชิป</u>

pinMode(ขาที่จะใช้ ,OUTPUT) =ตั้งค่าให้ขานั้นเป็นเอาท์พุทใช้ได้ทั้งอะนาลอกและดิจิตอล

pinMode(ขาที่จะใช้ ,INPUT) =ตั้งค่าให้ขานั้นเป็นอินพุทใช้ได้ทั้งอะนาลอกและดิจิตอล

digitalWrite(ขาที่จะใช้,สถานะที่จะแสดง)=ใช้สั่งขานั้นเป็นเอาท์พุทดิจิตอล เช่น LED หน้าจอ

digitalRead(ขาที่จะใช้) =ใช้สั่งขานั้นรับอินพุทดิจิตอล เช่น สวิตซ์กด เซ็นเซอร์ดิจิตอล

analogWrite(ขาที่จะใช้) =ใช้สั่งขานั้นเป็นเอาท์พุทอะนาลอก เช่น ICควบคุมความเร็วมอเตอร์

analogRead(ขาที่จะใช้) =ใช้สั่งขานั้นรับอินพุทอะนาลอก เช่น Rเกอกม้า LDR เซ็นเซอร์วัดระยะทาง

delay(เวลาหน่วยเป็นมิลลิวินาที)=ใช้หน่วงเวลาทำงานก่อนทำงานคำสั่งต่อไป

delayMicroseconds(เวลาหน่วยเป็นไมโครวินาที) =ใช้หน่วงเวลาทำงานก่อนทำงานคำสั่งต่อไป

int ชื่อ =ใช้ตั้งชื่อตัวแปรสำหรับตั้งค่าเริ่มต้นที่เป็นตัวเลขหรือจองพื้นที่เก็บตัวเลขหรือตั้งชื่อให้ขาไมโครคอนโทรลเลอร์ ชิปจะแยกเองได้ว่าเราตั้งชื่อนี้เพื่อเก็บเลขหรือตั้งชื่อขากันแน่

char ชื่อ[จำนวนอักษรที่จะจองต้องเผื่อไว้อีกหนึ่งช่องเสมอ] =ใช้ตั้งชื่อตัวเปรที่ใช้เก็บตัวอักษร(จองพื้นที่ใช้เก็บอักษร)

Serial.begin(9600) =ตั้งค่าเริ่มต้นเพื่อติดต่อสื่อสารกับคอมพิวเตอร์อัตราเร็ว9600บิตต่อวินาที

Serial.print(“คำพูด”) =ใช้พิมพ์คำพูดเพื่อให้แสดงผลบนจอคอมแบบไม่เว้นบรรทัด

Serial.println(“คำพูด”) =ใช้พิมพ์คำพูดเพื่อให้แสดงผลบนจอคอมแบบเว้นบรรทัด

Serial.available() =ใช้ตรวจสอบว่ามีการกดคีย์บอร์ดหรือไม่

Serial.Read() =ใช้อ่านค่าปุ่มคีย์บอร์ด

&& แปลว่า และ

|| แปลว่า หรือ

โปรแกรมไฟกระพริบ

```
void setup()      //ตั้งค่าเริ่มต้นให้ชิป
{
 pinMode(3,OUTPUT);  //ตั้งให้ขา3เป็นเอาท์พุท
}
void loop()       //ตั้งค่าทำงานให้ชิป
{
 digitalWrite(3,HIGH);   //ให้ขา3เป็นลอจิก 1
 delay(200);             //หน่วงเวลาก่อนเริ่มคำสั่งต่อไป200มิลลิวินาที
 digitalWrite(3,LOW);    //ให้ขา3เป็นลอจิก 0
 delay(300);             //หน่วงเวลาก่อนเริ่มคำสั่งต่อไป300มิลลิวินาที

}
```

หรือสามารถทำได้อีกวิธี

```
int LED=3;        //ให้ขา3เป็นขาไมโครคอนโทรลเลอร์ที่ชื่อLED
void setup()
{
 pinMode(LED,OUTPUT);
}
void loop()
{
 digitalWrite(LED,HIGH);
 delay(200);
 digitalWrite(LED,LOW);
 delay(300);
}
```

คำสั่งที่เกี่ยวข้องกับคณิตศาสตร์

ชื่อตัวแปร++ =เพิ่มค่าตัวแปรนั้นทีละหนึ่ง = +1
ชื่อตัวแปร-- =ลดค่าตัวแปรนั้นทีละหนึ่ง = -1

/ =หาร
* =คูณ

+ =บวก
_ =ลบ

= =เท่ากับ
== =เท่ากันกับ(สามารถใช้เปรียบเทียบคำสั่งกับคำสั่งก็ได้ไม่จำเป็นต้องใช้กับตัวเลข)หรือเทียบความเหมือนกัน

% =หารเอาเศษ เช่น 6%3 =0 , 6%4 =2 เพราะ 6 หาร 4 ได้ 1 เหลือเศษ 2 ส่วน 6 หาร 3 ลงตัว
!= =ไม่เท่ากับ

pi =22/7
pow(x,y) =ยกกำลัง เช่น pow(2,3) = 2^3
sqrt(x) =รากที่สอง

>
<
>= มากกว่าหรือเท่ากับ
<= น้อยกว่าหรือเท่ากับ

ตัวอย่างโปรแกรม

```
int x=2;  //ให้ค่าเริ่มต้นของตัวแปรที่ชื่อ x เท่ากับ 2
int y=3;  //ให้ค่าเริ่มต้นของตัวแปรที่ชื่อ y เท่ากับ 3
int z=0;  //ให้ค่าเริ่มต้นของตัวแปรที่ชื่อ z ว่างเปล่า
void setup()
{
 z=x+y;   //จากการประมวลผล z จะมีค่าเท่ากับ 5  ห้ามตั้งสมการว่า x+y=z
 pinMode(z,OUTPUT);
}
void loop()
{
 digitalWrite(z,HIGH);
 delay(200);
 digitalWrite(z,LOW);
 delay(300);
}
```

คำสั่ง do...while

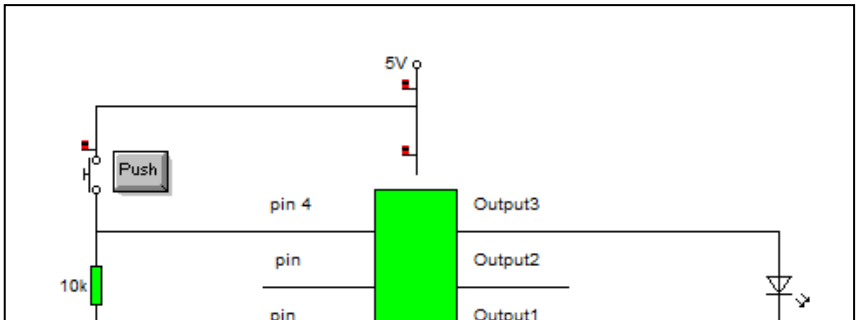
```
do
{
        //คำสั่ง 1
}while(เงื่อนไข);

        //คำสั่ง 2
```

บางครั้งเราอาจใช้แค่ while อย่างเดียวก็ได้

> **แปลว่า ทำคำสั่ง 1 เมื่อเงื่อนไขยังคงจริงหรือถูกต้อง** ใช้ทำงานตามคำสั่ง 1เมื่อเงื่อนไขในwhile ยังคงเป็นจริง แต่ถ้าเงื่อนไขนั้นไม่เป็นจริงชิปก็จะหยุดทำคำสั่ง 1 แล้วไปทำตามคำสั่ง 2

ตัวอย่าง

```
void setup()
{
 pinMode(4,INPUT);     //ให้ขา4เป็นอินพุทสวิตช์
 pinMode(3,OUTPUT);  //ให้ขา3เป็นเอาท์พุทต่อLED
}
void loop()
{
          do
          {
           digitalWrite(3,HIGH);              //คำสั่ง 1
           }
           while(digitalRead(4)==LOW);   //เมื่อยังไม่มีการกดสวิตช์ที่ขา4ทำให้LEDที่ขา 3 ยังแดงอยู่
           digitalWrite(3,LOW);              //(คำสั่ง 2 ) เมื่อมีการกดสวิตช์ทำให้LEDดับ 5 วินาทีแล้วแดงอีก
           delay(5000);
}
                    // digitalRead(4)==LOW หมายถึง การอ่านค่าแบบดิจิตอลที่ขา4ยังมีค่าเท่ากันกับลอจิก
```



การต่อวงจร

คำสั่ง if

```
if(เงื่อนไข)
{
        //คำสั่ง
}
```

แปลว่า ถ้าเมื่อไหร่ที่นั้นเงื่อนไขถูกต้องหรือเป็นจริงก็ให้เริ่มทำคำสั่งนั้นได้

ตัวอย่าง

```
void setup()
{
 pinMode(4,INPUT);
 pinMode(3,OUTPUT);
}
void loop()
{
 if(digitalRead(4)==HIGH)  //ถ้ามีการกดสวิตซ์ที่ขา4
          {
                  digitalWrite(3,HIGH);  //LED ที่ขา3จะแดง 5วินาทีแล้วจึงดับ
                  delay(5000);
                  digitalWrite(3,LOW);
          }
 delay(100);  //ในการตรวจสอบว่ามีการกดสวิตซ์หรือไม่ ในแต่ละครั้งชิปจะรอดูอยู่0.1วินาที
          //ถ้าไม่มีการกดชิปก็จะตรวจสอบใหม่อีกรอบไปเรื่อยๆจนกว่าจะพบว่ามีการกด
}
```

คำสั่ง if...else

```
if(เงื่อนไข)
{
        //คำสั่ง 1
}else
{
        //คำสั่ง 2
}
```

แปลว่า ถ้าเมื่อไหร่ที่นั้นเงื่อนไขถูกต้องหรือเป็นจริงก็ให้เริ่มทำคำสั่งที่ 1 ได้ แต่ถ้ายังไม่เป็นจริงก็ให้ทำคำสั่งที่ 2 รอไปก่อน

ตัวอย่าง

```
void setup()
{
 pinMode(4,INPUT);
 pinMode(3,OUTPUT);
 pinMode(5,OUTPUT);
}
void loop()
{
 if(digitalRead(4)==HIGH)  //ถ้ามีการกดสวิตซ์ที่ขา4
 {
  digitalWrite(3,HIGH);  //LED ที่ขา 3 จะแดง 5 วินาทีแล้วจึงดับและ LEDที่ขา 5 จะดับ 5 วินาที
  digitalWrite(5,LOW);
  delay(5000);
 }else                  //ถ้ายังไม่มีการกดสวิตซ์LEDที่ขา 5 จะแดงแต่ที่ขา 3 จะดับ
 {
  digitalWrite(5,HIGH);
  digitalWrite(3,LOW);
 }
 delay(100);  //ในการตรวจสอบว่ามีการกดสวิตซ์หรือไม่ ในแต่ละครั้งชิปจะรอดูอยู่0.1วินาที
          //ถ้าไม่มีการกดชิปก็จะตรวจสอบใหม่อีกรอบไปเรื่อยๆจนกว่าจะพบว่ามีการกด
}
```

คำสั่ง if...else if...else
คำสั่งนี้สามารถต่อไปได้เรื่อยๆจนกว่าจะพอใจ

```
if(เงื่อนไข 1)
{
        //คำสั่ง 1
}else if(เงื่อนไข 2)
{
        //คำสั่ง 2
}else
{
        //คำสั่ง 3
}       ไปเรื่อยๆๆ
```

แปลว่า ถ้าเมื่อไหร่ที่นั้นเงื่อนไข 1ถูกต้องหรือเป็นจริงก็ให้เริ่มทำคำสั่งที่ 1 ได้ แต่ถ้ายังไม่เป็นจริงก็ไปตรวจสอบที่เงื่อนไขที่ 2 ถ้าเงื่อนไขที่ 2 เป็นจริงก็ทำคำสั่งที่ 2 แต่ถ้าไม่เป็นจริงทั้งสองคำสั่งก็จะทำคำสั่งที่ 3 รอไปก่อน

ตัวอย่าง

```
void setup()
{
  pinMode(4,INPUT);
  pinMode(6,INPUT);
  pinMode(3,OUTPUT);
  pinMode(5,OUTPUT);
}
void loop()
{
  if(digitalRead(4)==HIGH)  //ถ้ามีการกดสวิตซ์ที่ขา4
  {
    digitalWrite(3,HIGH);  //LED ที่ขา 3 จะแดง 5 วินาทีแล้วจึงดับและ LEDที่ขา 5 จะดับ 5 วินาที
    digitalWrite(5,LOW);
    delay(5000);
  }else if(digitalRead(6)==HIGH) //ถ้ามีการกดสวิตซ์ที่ขา 6 LEDที่ขา 5 จะแดงแต่ที่ขา 3 จะดับ
  {
    digitalWrite(5,HIGH);
    digitalWrite(3,LOW);
    delay(5000);
  }else               //ถ้าไม่มีการกดสวิตซ์ LED จะกระพริบทั้งคู่
  {
    digitalWrite(3,LOW);
    digitalWrite(5,LOW);
    delay(500);
    digitalWrite(3,HIGH);
    digitalWrite(5,HIGH);
    delay(500);
  }
  delay(100);  //ในการตรวจสอบว่ามีการกดสวิตซ์หรือไม่ ในแต่ละครั้งชิปจะรอดูอยู่0.1วินาที
        //ถ้าไม่มีการกดชิปก็จะตรวจสอบใหม่อีกรอบไปเรื่อยๆจนกว่าจะพบว่ามีการกด
}
```

```
for(เงื่อนไขเริ่มต้น; เงื่อนไขสุดท้าย; การทำงานของเงื่อนไข)
{
        //คำสั่ง
}
```

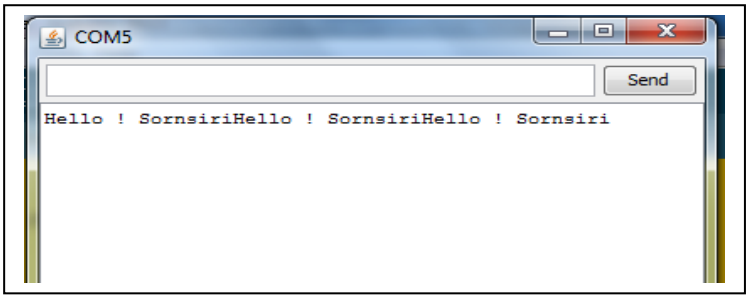
แปลว่า ให้ทำคำสั่งที่ได้ไปจนกว่าเงื่อนไขจะเป็นจริงครบทั้ง3ข้อ โดยคำสั่งเหล่านี้จะต้องมีความสัมพันธ์กันโดยตรงด้วย

ตัวอย่าง

```
int a=0;  //ให้ตัวแปรaเก็บตัวเลข  โดยเริ่มต้นที่ 0
void setup()
{
 pinMode(3,OUTPUT);
}
void loop()
{
for(a=0;a<100;a++)  //ให้aเริ่มต้นมีค่าเท่ากับ 0 , ให้โปรแกรมยังคงทำงานเมื่อ a น้อยกว่า 100 ในการวนแต่ละรอบให้a
                    //เพิ่มขึ้นทีละ 1
 {
  digitalWrite(3,HIGH);  //ถ้า a ยังน้อยกว่า 100 LEDที่ขา3 จะแดง
  delay(10);             //ในการเพิ่ม a แต่ละรอบจะหน่วงเวลา 10 millisec
 }
  digitalWrite(3,LOW);   //ถ้า a มากว่า 100ไปแล้วให้LEDดับ 2 วินาที  ก่อนจะเริ่มนับ a ใหม่
  delay(2000);
}
```

การติดต่อกับหน้าจอคอมและคีย์บอร์ด

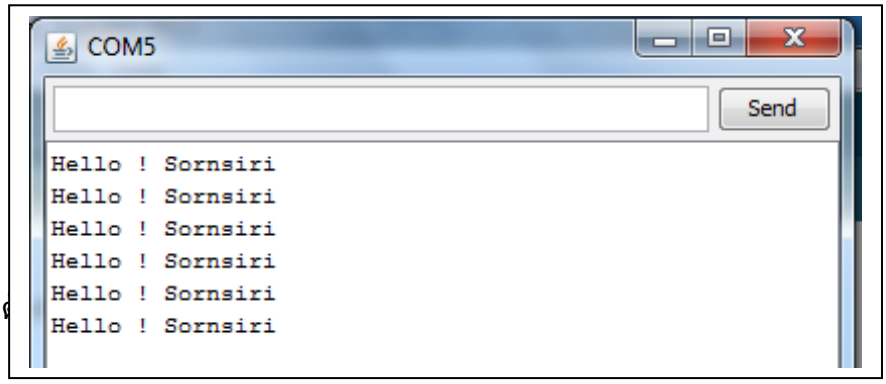
```
void setup()
{
 Serial.begin(9600);       //เริ่มต้นการสื่อสารที่ความเร็ว 9600 bit/sec
}
void loop()
{
```



หรือ

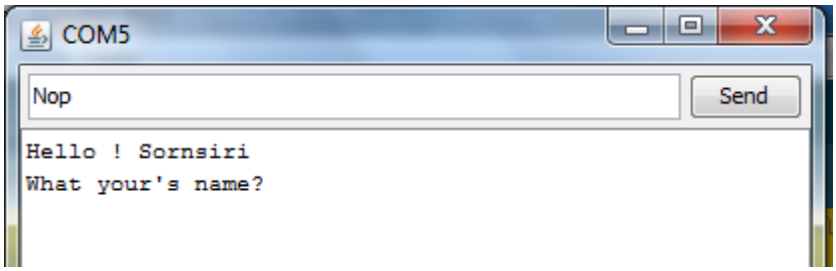
```
void setup()
{
  Serial.begin(9600);       //เริ่มต้นการสื่อสารที่ความเร็ว 9600 bit/sec
}
void loop()
{
  Serial.println("Hello ! Sornsiri");          //ให้แสดงคำว่า Hello ! Sornsiri ไปเรื่อยๆ ครั้งละ1วินาที โดยเว้นบรรทัด
  delay(1000);
}
```

การรับ

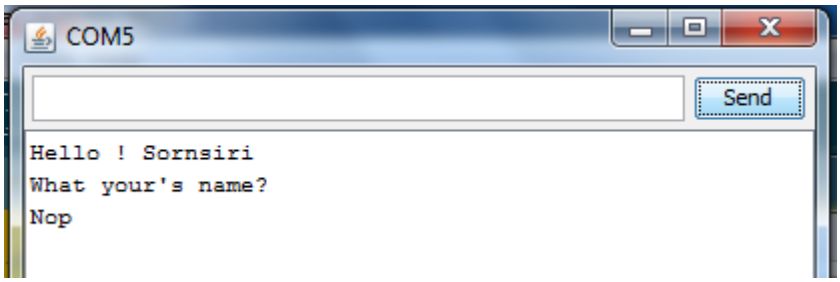


```
char name[10];  //จองพื้นที่ไว้สำหรับเก็บชื่อ ให้ชื่อไดร์ฟว่า name จองไว้ 10 Byte แต่จะใช้ได้แค่ 9 Byte
int i=0;  //จองที่เก็บตัวเลขไว้นับตัวอักษร
void setup()
{
 Serial.begin(9600);
 Serial.println("Hello ! Sornsiri");
 Serial.println("What your's name?");
}
void loop()
{
 if(Serial.available()>0)          //ถ้าตรวจพบว่ามีการกดคีย์บอร์ดให้โปรแกรมกระทำดังต่อไปนี้
 {
  i++;                              //เริ่มนับทีละช่องเมื่อมีการกรอกตัวอักษรแต่ละตัว
  name[i]=Serial.read();           //เอาตัวอักษรที่ได้รับไปเก็บไว้ในช่องดังกล่าว
  Serial.print(name[i]);           //แสดงผลตัวอักษรที่กรอกลงมาทีละช่อง
 }
 delay(100);                       //ในการรับอักษรแต่ละตัวให้รอ 0.1 วินาที
}                                  //  1 Byte = 1 ตัวอักษรหรือตัวเลข
```

ก่อนกด ENTER



หลังกด ENTER



การนำเอาท์พุทจากคีย์บอร์ดไปใช้

```
char a;   //สร้างไดร์ฟเพื่อเก็บตัวอักษรขนาด 1 Byte ชื่อ a ในกรณีที่ใช้ตัวอักษรเดียวไม่ต้องเผื่อพื้นที่ให้คอม
void setup()
{
 pinMode(3,OUTPUT);
 pinMode(10,OUTPUT);
 Serial.begin(9600);
 Serial.println("Hello ! Sornsiri");
 Serial.println("A= Open LED 3");
 Serial.println("B= Open LED 10");
}
void loop()
{
 if(Serial.available()>0)     //ถ้ามีการกดตัวอักษรให้โปรแกรมทำดังนี้
 {
  a=Serial.read();          //ให้นำอักษรที่กดนั้นไปเก็บไว้ที่ไดร์ฟ a ก่อนนำไปอ่านตีความ
  if(a=='A')               //ถ้ากดอักษร A ให้ LED 3แดง
  {
   digitalWrite(3,HIGH);
   digitalWrite(10,LOW);
  }else if(a=='B')         //ถ้ากดอักษร B ให้ LED 10แดง
  {
   digitalWrite(10,HIGH);
   digitalWrite(3,LOW);
  }else                    //ถ้ากดอักษรอื่นให้ดับทั้งคู่
  {
   digitalWrite(3,LOW);
   digitalWrite(10,LOW);
  }
 }
 delay(100);
}
```

ใช้เพื่อเชื่อมคำสั่งหรือเงื่อนไขเข้าด้วยกัน โดยโปรแกรมจะทำงานก็ต่อเมื่อคำสั่งหรือเงื่อนไขนั้นถูกต้องทั้งคู่เท่านั้น
หรือเรียกได้ว่าคำสั่งนี้ใช้แทน And gate ได้นั่นเอง

คำสั่ง&&คำสั่ง
เงื่อนไข&&เงื่อนไข

ตัวอย่าง

```
void setup()
{
  pinMode(3,OUTPUT);
  pinMode(4,INPUT);
  pinMode(5,INPUT);
}
void loop()
{
  if((digitalRead(4)>0)&&(digitalRead(5)>0))  //ถ้ามีการกดสวิตช์ที่ขา4,5 พร้อมกันLEDจะแดง 1 วินาทีจึงดับ
                                              // ( >0 สามารถใช้แทน ==HIGH ได้ )
  {
    digitalWrite(3,HIGH);
    delay(1000);
    digitalWrite(3,LOW);
  }
  delay(100);
}
```

การใช้ ||
เป็นการใช้แทน Or gate คำสั่งหรือเงื่อนไขนั้นต้องเป็นจริงอย่างน้อยหนึ่งอย่าง *สำคัญเอาไปใช้งาน*

ตัวอย่าง

```
void setup()
{
  pinMode(3,OUTPUT);
  pinMode(4,INPUT);
  pinMode(5,INPUT);
}
void loop()
{
  if((digitalRead(4)==HIGH)||(digitalRead(5)==HIGH))  //ถ้ามีการกดสวิตซ์ที่ขา4,5 ขาใดขาหนึ่ง LEDจะแดง 1 วินาทีจึงดับ
                                                      // ( >0 สามารถใช้แทน ==HIGH ได้ )
  {
    digitalWrite(3,HIGH);
    delay(1000);
    digitalWrite(3,LOW);
  }
  delay(100);
}
```

การรับอินพุทอะนาลอก

รับได้จาก เซ็นเซอร์อะนาลอก LDR VR แรงดันไฟฟ้าที่ไม่เกิน5 v. และอีกมากมาย

ในกรณีที่ Load มีค่าไม่เกิน 1000 Ohm

ตัวอย่าง

```
int a=0;
void setup()
{
  pinMode(14,INPUT);    //ขาที่ 14 คือขา A0นั่นเอง
  Serial.begin(9600);
}
void loop()
{
  a=analogRead(14);       //ให้ a เก็บค่าที่อ่านได้จากขา 14
  Serial.println(a);          //ให้แสดงผลค่าที่อ่านได้บนจอคอม
  delay(1000);
}
```