

การสื่อสารแบบอนุกรม

สายค่าน์โหลดของอาร์ดูโนเป็นพอร์ตอนุกรม สามารถใช้สายนี้สื่อสารกับคอมพิวเตอร์ที่มีพอร์ตอนุกรมได้ การทดลองนี้จะใช้พอร์ตอนุกรมของอาร์ดูโน สื่อสารกับเครื่องพีซี โดยที่พีซีจะใช้โปรแกรม

- Serial Monitor ของโปรแกรม Arduino
- โปรแกรม Hyper terminal หรือ
- ใช้โปรแกรม Processing

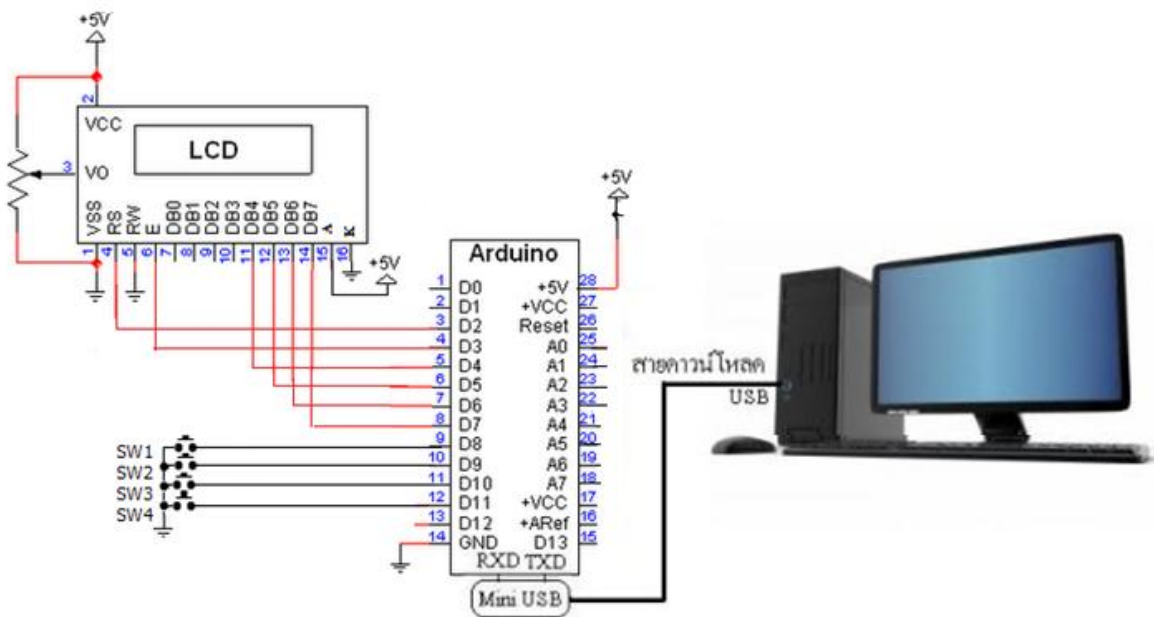
หมายเหตุ พอร์ตอนุกรมใช้ขาสัญญาณเพื่อการติดต่ออยู่ 2 ขาคือ

ขารับข้อมูล RxD ใช้ขา D0

ขาส่งข้อมูล TxD ใช้ขา D1

ดังนั้นเมื่อใช้พอร์ตอนุกรม ขา D0 และ D1 จึงใช้เป็นสัญญาณ Digital I/O ไม่ได้

ต่อวงจรตามรูปที่ 1



ตัวอย่างที่ 1 พิมพ์ข้อมูลทั้งอักขระออกทางพอร์ตอนุกรม RS232

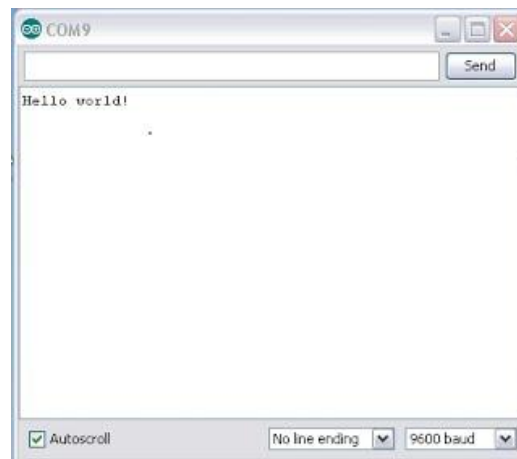
```
void setup()
{
  Serial.begin(9600); // set baudrate

  Serial.println("Hello, I'm Arduino."); // prints title with ending line break
}
void loop()
{
}
```

ตรวจสอบการทำงานโดยใช้คำสั่ง Serial Monitor ของโปรแกรม Arduino ดังนี้



เมื่อปรากฏหน้าต่าง และrun โปรแกรมบนบอร์ด Arduino จะได้ผลลัพธ์ดังนี้
(สามารถ RUN โปรแกรมบน Arduino ได้ใหม่โดยกดสวิทช์รีเซ็ต บนบอร์ด Arduino)



ตัวอย่างที่ 2 รับข้อมูลจากพอร์ตอนุกรม โดยใช้ Serial Monitor เป็นตัวส่งข้อมูล แล้วแสดงผลออกทาง LCD
ข้อมูลที่รับความยาวไม่เกิน 16 ตัวอักษร หรือ กดคีย์ Enter

**** โปรแกรม Arduino ****

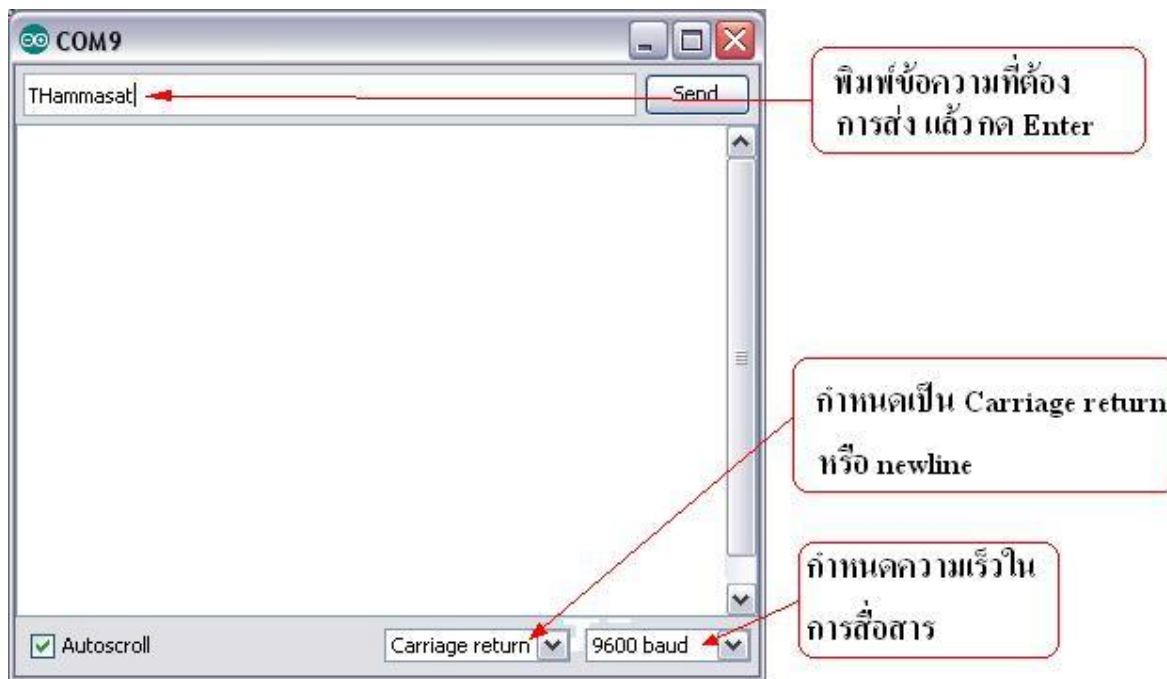
```
/* Read a text received on serial arduino port and display on LCD*/  
#include <LiquidCrystal.h>  
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);  
#define INLENGTH 16  
#define lf 10 //ASCII code Line feed  
#define cr 13 //ASCII code Carriage return  
char inString[16];  
int inCount;  
int incomingByte = 0;  
  
void setup()  
{  
  Serial.begin(9600);  
  lcd.begin(16, 2);  
  pinMode(13, OUTPUT);  
  digitalWrite(13, HIGH); // set the LED on
```

```

lcd.println("Ready to receive");
}

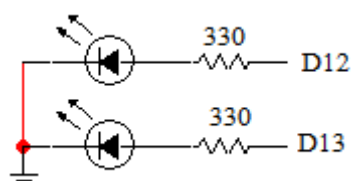
void loop()
{
  inCount = 0;
  do
  {
    while (!Serial.available());    // wait for input
    inString[inCount] = Serial.read(); // get it
    if ((inString[inCount] == cr)|| (inString[inCount] == lf)) break;
    ++inCount;
  } while(inCount < INLENGTH);
  inString[inCount] = 0;             // null terminate the string
  lcd.setCursor(0,1); //col, row
  lcd.print(" ");
  lcd.setCursor(0,1); //col, row
  lcd.print(inString);
}

```



ตัวอย่างที่ 3 รับข้อมูลจากพอร์ตอนุกรม เพื่อควบคุมการติดดับของ LED โดยใช้ Serial Monitor เป็นตัวส่งคำสั่ง

ต่อ LED กับความต้านทาน 330 โอห์ม เข้ากับ D6 และ D7 ของ Arduino ตามรูป



เขียนโปรแกรมดังนี้

ทดลองกดเป็นตัวเลข '0' '1' '2' และ '3' แล้วสังเกตการทำงานของ LED และ LCD

```
/* Read a text received on serial arduino port and display on LCD*/
#include <LiquidCrystal.h>
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
void setup() {
  Serial.begin(9600); // initialize serial:
  lcd.begin(16, 2); //set up the LCD's number of columns and rows:
  lcd.print("hello, world!"); // Print a message to the LCD.
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  digitalWrite(12,LOW);
  digitalWrite(13,LOW);
}

void loop() {
  lcd.setCursor(0, 1); // set the cursor to column 0, line 1
  lcd.print(millis()/1000); // print the number of seconds since reset:
}

void serialEvent() {
  char inChar = (char)Serial.read();
  switch (inChar) {
    case '0':
      digitalWrite(12,LOW);
      break;
    case '1':
      digitalWrite(12,HIGH);
      break;
    case '2':
      digitalWrite(13,LOW);
      break;
    case '3':
      digitalWrite(13,HIGH);
      break;
    default:
      break;
  }
}
```

คำถาม ทำไม Arduino ถึงรับตัวเลขจาก serial monitor ได้ ทั้งๆที่ไม่มีคำสั่งรับข้อมูลจากพอร์ตอนุกรมภายในฟังก์ชัน loop ?

ตัวอย่างที่ 4 รับข้อมูลจากสวิตช์ sw1 ถึง sw4 ส่งสัญญาณออกพอร์ตอนุกรม เพื่อแสดงผลที่ Serial Monitor ตัวอย่างนี้ Serial Monitor ทำหน้าที่เป็นทั้งตัวส่งและตัวรับ เป็นตัวส่งคำสั่งต่อ สวิตช์ กับความต้านทาน 10 K เข้ากับ D8 และ D9 ของ Arduino ตามรูป

เขียนโปรแกรมดังนี้

```
// Read a text received on serial arduino port and display on LCD
// Arduino read sw status and sent to serial monitor
#include <LiquidCrystal.h>
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
unsigned char vin1 = 0;
unsigned char vin2 = 0;
unsigned char vin3 = 0;
unsigned char vin4 = 0;

void setup() {
  Serial.begin(9600); // initialize serial:
  lcd.begin(16, 2); //set up the LCD's number of columns and rows:
  lcd.print("Press switch"); // Print a message to the LCD.
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  pinMode(8, INPUT);
  pinMode(9, INPUT);
  pinMode(10, INPUT);
  pinMode(11, INPUT);
  digitalWrite(8,HIGH);
  digitalWrite(9,HIGH);
  digitalWrite(10,HIGH);
  digitalWrite(11,HIGH);
  digitalWrite(12,LOW);
  digitalWrite(13,LOW);
}

void loop() {
  lcd.setCursor(0, 1); // set the cursor to column 0, line 1
  lcd.print(millis()/1000); // print the number of seconds since reset:
  vin1 = digitalRead(8);
  vin2 = digitalRead(9);
  vin3 = digitalRead(10);
  vin4 = digitalRead(11);

  if(vin1==0) Serial.println("Switch 1 close.");
  else Serial.println("Switch 1 open.");
  if(vin2==0) Serial.println("Switch 2 close.");
  else Serial.println("Switch 2 open.");
  if(vin3==0) Serial.println("Switch 3 close.");
  else Serial.println("Switch 3 open.");
  if(vin4==0) Serial.println("Switch 4 close.");
  else Serial.println("Switch 4 open.");

  delay(400);
}

void serialEvent() {
  char inChar = (char)Serial.read();
  switch (inChar) {
    case '0': digitalWrite(6,LOW); break;
    case '1': digitalWrite(6,HIGH); break;
    case '2': digitalWrite(7,LOW); break;
    case '3': digitalWrite(7,HIGH); break;
    default: break;
  }
}
```

ทดลองกดเป็นตัวเลข และสับสวิตช์ แล้วสังเกตการทำงานของ serial monitor LED และ LCD