

C h a p t e r

06



시리얼 통신(Serial)

공두이노 보드를 이용하여 PC와 시리얼 통신하는 방법에 대해서 학습합니다.

■ 시리얼 통신 (직렬 통신; Serial Communication)

공두이노 시스템이 PC 또는 다른 시스템과 데이터를 주고 받으려면 선을 연결하여 통신을 하는데, 주로 시리얼 통신을 이용합니다.



시리얼 통신의 특징은 다음과 같습니다.

- ① 시리얼 통신은 송신(TX), 수신(RX)의 2개 데이터 선을 이용하여 통신을 하기 때문에 비교적 간단하게 통신 연결이 가능합니다.
- ② 바이트(8비트)의 데이터를 비트 단위로 나누어 차례대로 전송하기 때문에 통신 시간이 빠르지 않습니다.
- ③ 공두이노와 PC와의 시리얼 통신은 현재 사용하고 있는 USB 포트를 이용하기 때문에 별도의 선 연결이 필요 없습니다.
- ④ 시리얼 통신을 하기 위해서는 속도(bps), 패리티(Parity), 스탑비트(Stop bit), 데이터 비트(Data bits) 설정이 필요하며, 일반적으로 속도 9,600 bps 패리티는 No parity, 1 스탑비트, 8 데이터 비트를 사용합니다.

실습 07

시리얼 통신 – 문자열 전송

공두이노 보드에서 PC로 문자열을 전송하는 방법을 학습합니다.

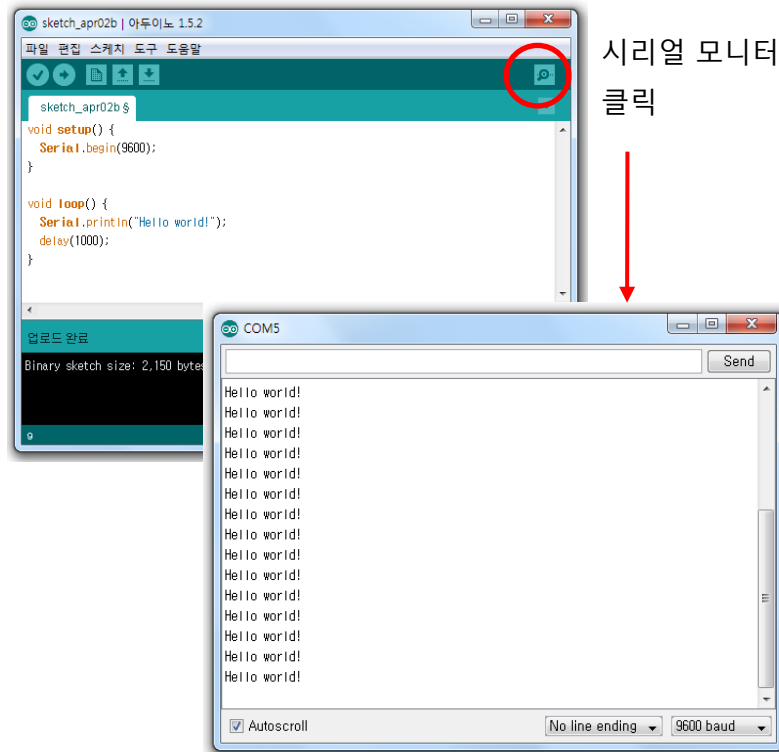
하드웨어 연결

1. 공두이노 보드와 PC가 USB로 연결되어 있으면 다른 연결은 필요 없습니다.

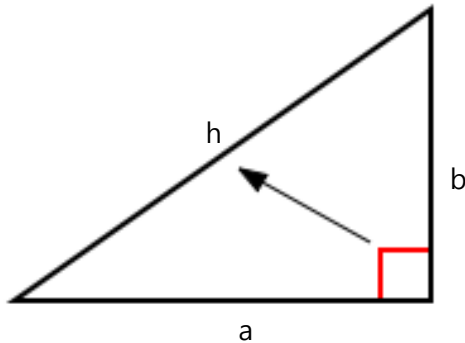
프로그램 작성1

```
void setup() {  
    // 9,600bps 속도 설정  
    Serial.begin(9600);  
}  
  
void loop() {  
    // PC로 문자열 전송  
    Serial.println("Hello world!");  
    delay(1000);  
}
```

프로그램 동작



피타고라스의 정리를 이용하여 빗변(h)의 길이를 계산하여 그 결과를 시리얼 통신으로 전송하는 프로그램을 작성합니다.



```
#include "math.h" // 수학관련 함수 포함

int a = 3;
int b = 4;
int h;

void setup() {
    Serial.begin(9600);
    Serial.println("Calculate");
    Serial.print("a = ");
    Serial.println(a);
    Serial.print("b = ");
    Serial.println(b);

    h = sqrt( a*a + b*b );

    Serial.print("h = ");
    Serial.println(h);
}

void loop() { // 반복 동작은 하지 않음
}
```

하드웨어 연결

1. 공두이노 보드와 PC가 USB로 연결되어 있으면 다른 연결은 필요 없습니다.

프로그램 작성1

exam025

PC에서 문자를 입력하면 그 문자에 해당하는 아스키 코드(ASCII) 를 10진수로 PC에 다시 전송하여 표시합니다.

```
int rx = 0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  if (Serial.available() > 0) { // 받은 데이터가 있으면 처리
    rx = Serial.read();
    Serial.print("I received: ");
    Serial.println(rx, DEC); // 받은 데이터의 10진 ASCII코드 출력
  }
}
```

프로그램 동작

시리얼 모니터 창에 'a' 를 입력하고 <Send> 버튼을 누르면 'I received : 97' 이 표시됩니다.
'a' 문자는 ASCII 코드 97에 해당합니다. (앞의 ASCII 코드 표 참조)



하드웨어 연결

1. 공두이노 보드와 베이스 보드(Gongduino-base-board)의 전원 커넥터를 케이블로 연결합니다.
2. 베이스 보드의 FND 연결 핀 8개를 연결선으로 공두이노 보드와 연결합니다.

공두이노 보드	연결방향	베이스 보드
6번 PIN	→	FND a
7번 PIN	→	FND b
8번 PIN	→	FND c
9번 PIN	→	FND d
10번 PIN	→	FND e
11번 PIN	→	FND f
12번 PIN	→	FND g
13번 PIN	→	FND DP

프로그램 작성2

exam026

PC에서 숫자(0-9)를 입력하면 FND에 숫자를 표시합니다. 그 외 입력이면, FND를 끕니다.

```
int rx = 0;

void out_fnd(int num) {
    unsigned int fnd[] = {0x03, 0x9f, 0x25, 0x0d, 0x99, 0x49, 0x41, 0x1f, 0x01, 0x19, 0xff};
    for (int i = 6; i <= 13; i++) {
        if (bitRead(fnd[num], 13-i)) {digitalWrite(i, HIGH);}
        else {digitalWrite(i, LOW);}
    }
}
```

// fnd[10] = FND를 모두 OFF

프로그램 계속...

```

void setup() {
    Serial.begin(9600);
    for (int i=6; i<=13; i++) {pinMode(i, OUTPUT);};
}


void loop() {
    if (Serial.available() > 0) { // 받은 데이터가 있으면 처리
        rx = Serial.read();
        if ((rx >= '0') && (rx <= '9')) {out_fnd(rx - '0');}
        else {out_fnd(10);}; // 숫자가 아니면 FND 모두 OFF
    }
}

```

- ✓ ‘0’ 부터 ‘9’ 까지의 아스키 코드 값은 48-57 까지 입니다. 이 코드 값을 out_fnd(int num)의 num 숫자인 0부터 9까지로 바꾸려면 수신 받은 rx 값에서 ‘0’ 의 아스키 코드 값인 48을 빼주어야 합니다.

예) out_fnd (rx - ‘0’)



 exam027

- ✓ “프로그램 작성2” 를 수정하여 PC에서 키보드에서 숫자 ‘0’ 부터9’ 까지,
그리고 알파벳 “A” 부터 “F” 까지 누르면 FND에 눌린 키를 표시하는 프로그램을 작성하세요.
만약, 그 외 키가 눌리면 FND를 모두 끕니다.

PC에서 문자열(2개 이상의 문자)을 입력하여 화면에 다시 표시하는 프로그램을 작성합니다.

```
char rd[20]; // 문자열 배열의 선언

void setup() {
  Serial.begin(9600);
  Serial.println("What's your name ?");
}

void loop() {
  if (Serial.readBytesUntil('.', rd, 20)) {
    Serial.print("Hello, ");
    Serial.println(rd);
    for(int i=0; i<20; i++) rd[i]=0; // 배열의 초기화
  }
}
```

byte Serial.readBytesUntil(문자, 배열, 길이)

해당하는 문자가 수신될 때까지 기다리며, 수신된 문자를 배열에 넣습니다. 배열의 크기는 길이인자에 지정합니다.

- 수신된 문자의 수를 리턴합니다.
- 수신될 때까지 기다리는 시간은 기본 1초입니다. 이 시간은 **Serial.setTimeout(밀리초)** 함수로 변경할 수 있습니다.

PC에서 2개의 숫자열(A, B 값)을 입력 받아 두 수를 더하고, 빼는 계산 결과를 출력하는 프로그램을 작성합니다.

```
int a = 0;           // a 값 저장 변수
int b = 0;           // b 값 저장 변수
int c = 0;           // a, b의 계산 결과 저장 변수
```

```
void setup() {
    Serial.begin(9600);
    Serial.print("A, B = ");
}

    ⋮
```

```
void loop() {
    while (Serial.available() > 0) { // 수신 데이터가 있으면
        a = Serial.parseInt();       // 첫번째 입력 숫자 = a 변수
        b = Serial.parseInt();       // 두번째 입력 숫자 = b 변수

        Serial.print(a);              // 화면에 입력한 a, b 표시
        Serial.print(", ");
        Serial.println(b);

        c = a + b;
        Serial.print ("A + B = ");
        Serial.println(c);

        c = a - b;
        Serial.print ("A - B = ");
        Serial.println(c);

        Serial.print("A, B = ");     // 다시 입력 받기 위해 표시
    }
}
```