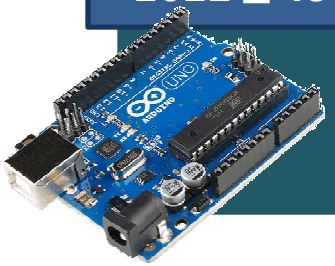
2022년 IoT기반 스마트 솔루션 개발자 양성과정



Firmware [펌웨어]

7-Serial Receive

담당 교수 : 유근택
010-5486-5376
rgt3340@naver.com
http://cafe.naver.com/cbdsp

Serial.begin/available

- Serial.begin(baud rate) //전송 속도를 설정
  - Baud rate : 초당 전송 비트수
  - 9600, 14400, 19200, 28800, 38400, 57600, 115200
  - EX) Serial.begin(115200);

- Serial.available( ) //수신 버퍼의 바이트 수
  - EX) int k=Serial.available( );
  - EX) if ((Serial.available( )>0) { ----------}
  - EX) if ((Serial.available( )) { ----------}

Serial.read/parseInt

- Serial.read( ) //수신 버퍼를 읽어옴
  - EX) int inByte=Serial.read( );

- Serial.parseInt( ) //수신 데이터를 정수로 변환
  - 숫자로 변환 가능한 문자열-> 정수로
  - EX) int Number=Serial.parseInt( );
  - "A12" -> 12
  - "-34" -> 34

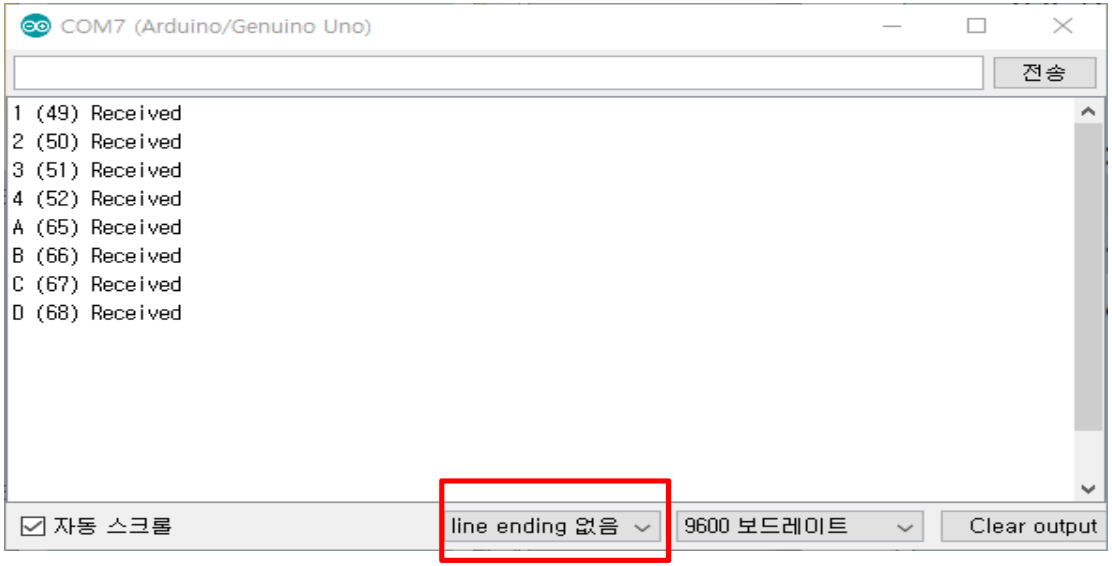
A7-1 : Serial.read( )

- Serial로 받은 문자를 리턴 함 -> Loop back test

A7-1 : Program

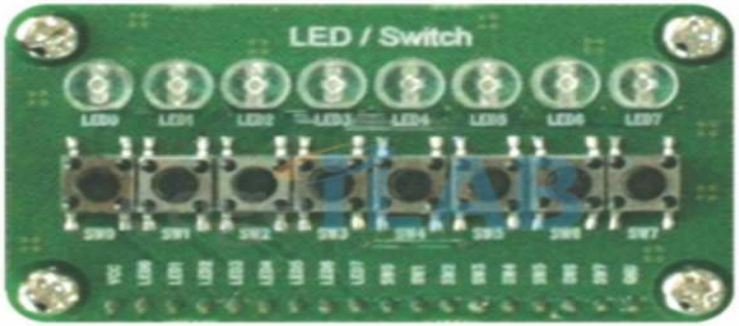
```
void setup( ) {
  Serial.begin(9600);
}

void loop( ) {
  if (Serial.available( )){
    unsigned char inByte = Serial.read( );

    Serial.write(inByte);
    Serial.print(" (");
    Serial.print(inByte);
    Serial.println(") Received");
  }
}
```

A7-1 : Serial Monitor

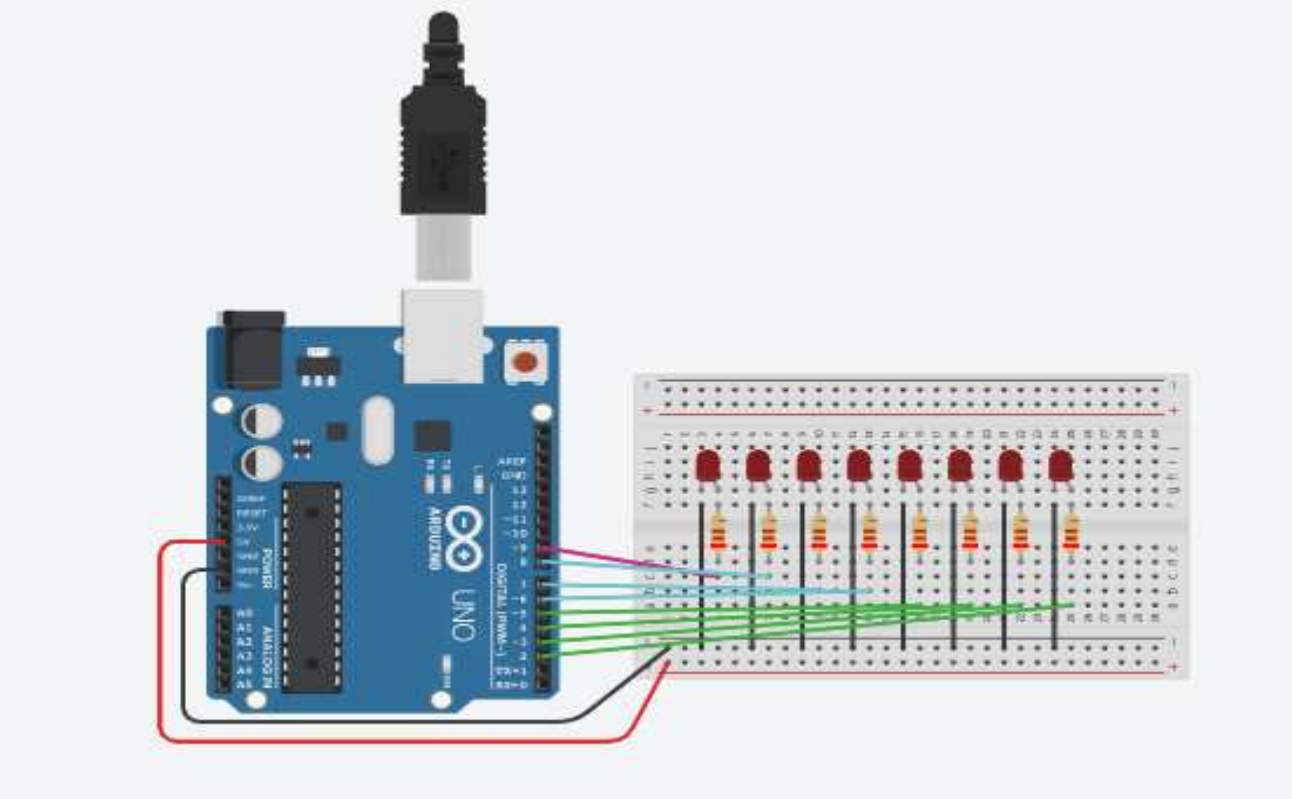


A7-2 : Serial.parseInt( )

- 숫자를 Serial data로 입력받아 해당 LED를 켠다
- 숫자의 범위 : 0~7



Wiring



A7-2 : Program

```
unsigned char LED[8]  = {9, 8, 7, 6, 5, 4, 3, 2};
unsigned char Mask[8] = {0x80, 0x40, 0x20, 0x10, 0x08, 0x04, 0x02, 0x01};
unsigned char LED_Buff=0x00;

void setup( ) {
  for (int k = 0; k < 8; k++) pinMode(LED[k], OUTPUT);
  Serial.begin(9600);
}
```

```
void loop( ) {
  if (Serial.available( )){
    unsigned char inByte = Serial.parseInt( );
    if (inByte>7){
      Serial.print(inByte);
      Serial.println(" LED failed");
    } else {
      LED_Buff = 1<<inByte;

      for (int k = 0; k < 8; k++)
          digitalWrite(LED[k], LED_Buff & Mask[k]);
      Serial.print(inByte);
      Serial.println(" LED ok!");
    }
  }
}
```

A7-2 : Serial Monitor

A7-3 : 계산(+) 결과 보이기 1

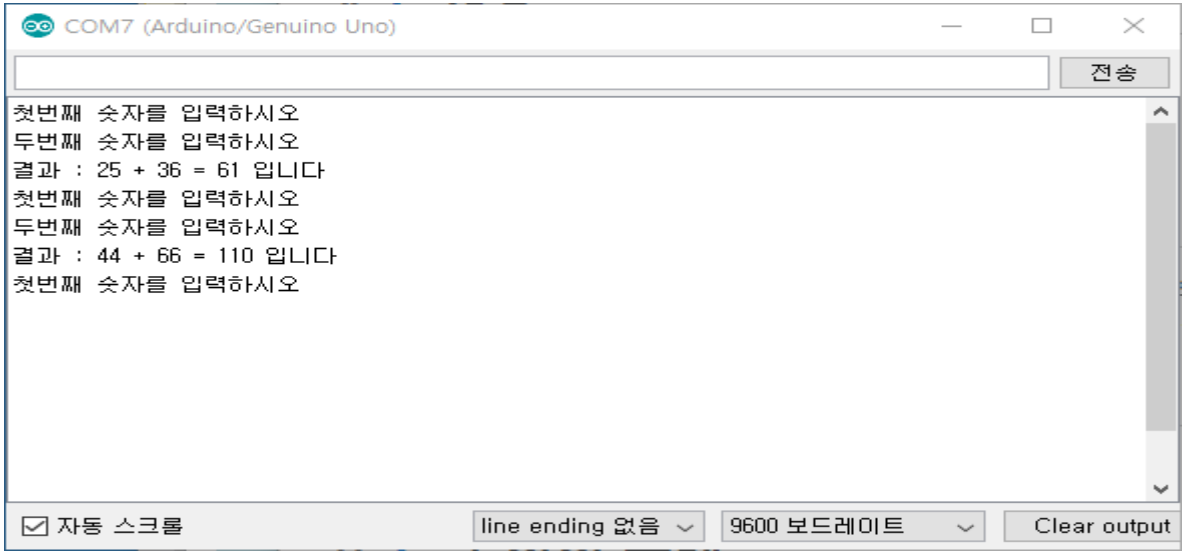
- Scenario
  - 첫번째 숫자를 입력하시오 -> 25
  - 두번째 숫자를 입력하시오 -> 36
  - 결과 : 25 + 36 = 61 입니다

A7-3 : Program

```
unsigned char Number_1, Number_2;
unsigned long Value;
unsigned char Count=0;

void setup( ) {
  Serial.begin(9600);
  Serial.println("첫번째 숫자를 입력하시오");
}
```

```
void loop( ) {
  if (Serial.available( )){
    unsigned char inNumber = Serial.parseInt( );
    if (Count==0){
      Number_1=inNumber;
      Count+=1;
      Serial.println("두번째 숫자를 입력하시오");
    } else {
      Number_2=inNumber;
      Value=Number_1+Number_2;

      Serial.print("결과 : ");
      Serial.print(Number_1);  Serial.print(" + ");  Serial.print(Number_2);
      Serial.print(" = ");   Serial.print(Value);   Serial.println(" 입니다");

      Serial.println("첫번째 숫자를 입력하시오");
      Count=0;
    }
  }
}
```

A7-3 : Serial Monitor



Serial.readStringUntil( )

- Syntax
  - Serial.readStringUntil(terminator)
- Parameter
  - terminator : 종료 문자 '₩n', '₩r'
- Example
  - String rxString = Serial.readStringUntil('₩n');

A7-4 : Hello

- Scenario
  - 이름을 입력하시오
  - RYU
  - RYU님 안녕하십니까?

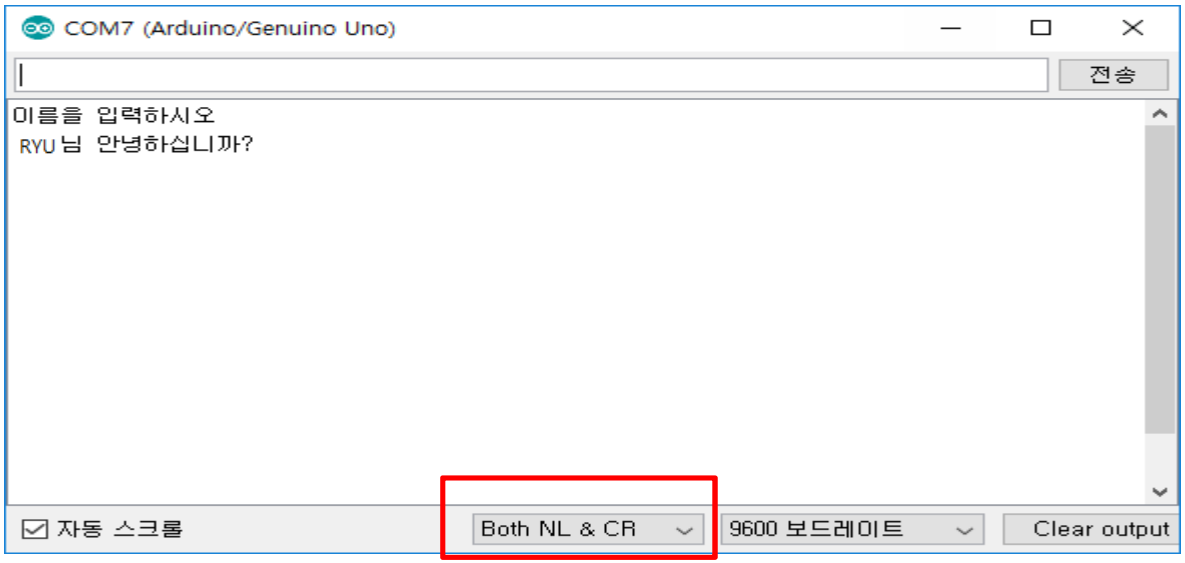
  - 한글은 Unicode형태로 되어 있어서 문자열 처리가 어렵다

A7-4 : Program

```
String rxString = "";

void setup( ) {
  Serial.begin(9600);
  Serial.println("이름을 입력하시오");
}
```

```
void loop( ) {
  if(Serial.available()){
    rxString = Serial.readStringUntil('₩n');
  }

  if (rxString != ""){
    Serial.write(rxString);
    Serial.write(" 님 안녕하십니까?");
    Serial.write('₩n');
    rxString = "";
   }
}
```

A7-4 : Serial Monitor



A7-5 : 계산(+) 결과 보이기 2

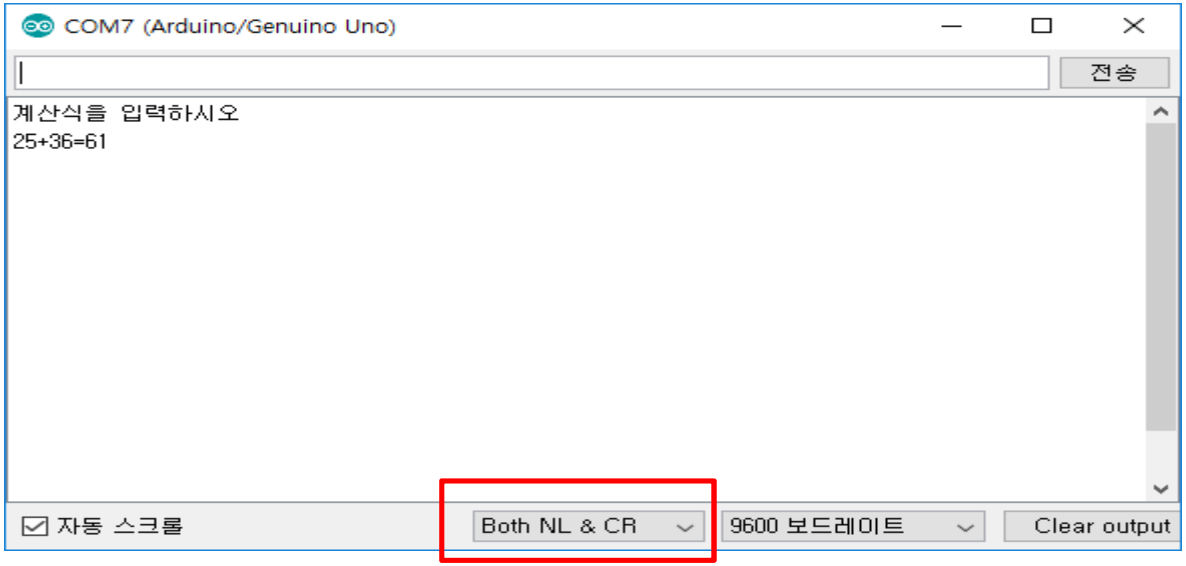
- Scenario
  - 계산식을 입력하시오
  - 25+36
  - 25+36=61

A7-5 : Program

```
String rxString = "";

void setup( ) {
  Serial.begin(9600);
  Serial.println("계산식을 입력하시오");
}
```

```
void loop( ) {
  if(Serial.available( )){
    rxString = Serial.readStringUntil('₩n');

    int strlength = rxString.length( );
    int op = rxString.indexOf("+");
    long value1 = rxString.substring(0, op).toInt( );
    long value2 = rxString.substring(op+1,strlength).toInt( );

    Serial.print(value1);
    Serial.print("+");
    Serial.print(value2);
    Serial.print("=");
    Serial.println(value1+value2);
  }
}
```

A7-5 : Serial Monitor

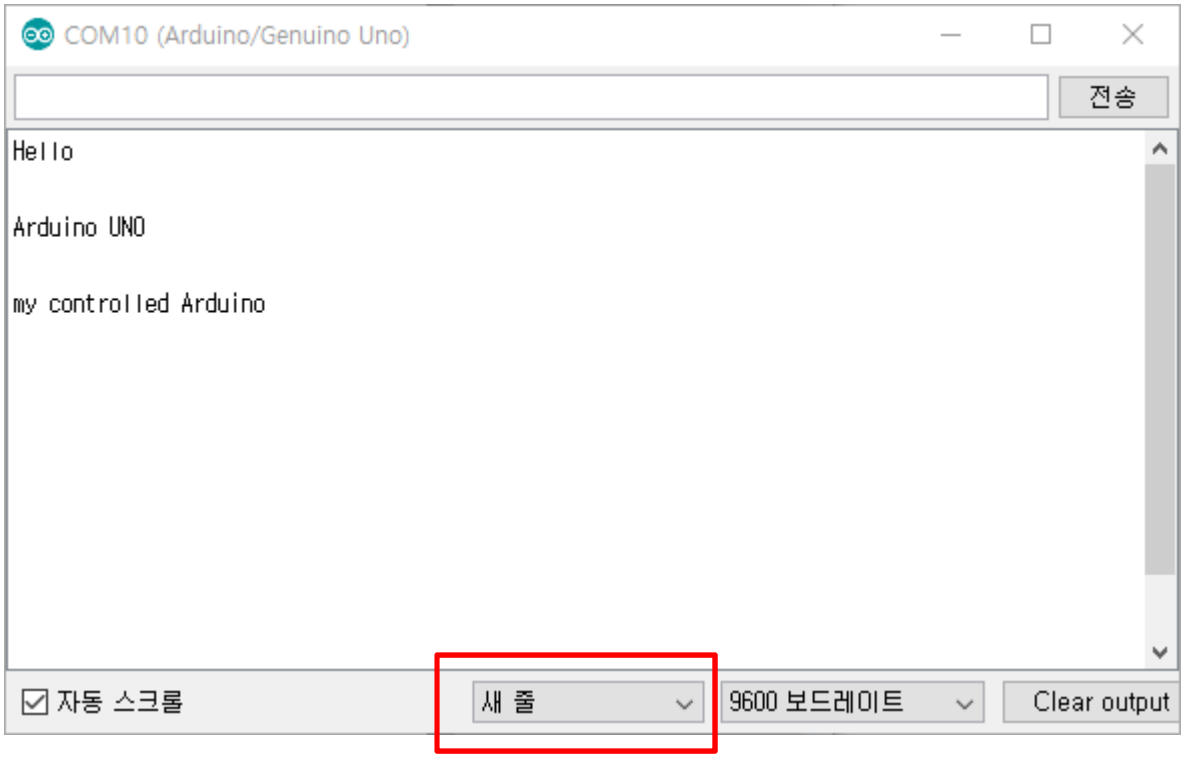


A7-6 : serialEvent( )

```
String inputString = "";
bool stringComplete = false;

void setup( ) {
  Serial.begin(9600);
  inputString.reserve(200);
}

void loop( ) {
  if (stringComplete) {
    Serial.println(inputString);
    inputString = "";
    stringComplete = false;
  }
}
```

```
void serialEvent( ) {
   while ( Serial.available( ) ) {
       char inChar = (char)Serial.read( );
       inputString += inChar;
       if (inChar == '₩n') {
            stringComplete = true;
       }
   }
}
```

A7-6 : Serial Monitor

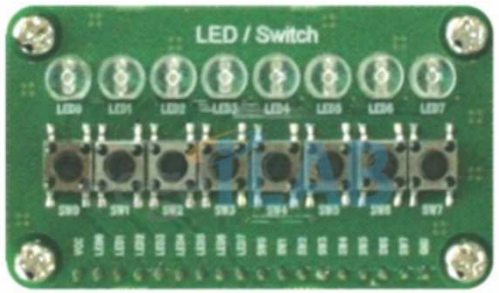


A7-7 : Remote Serial

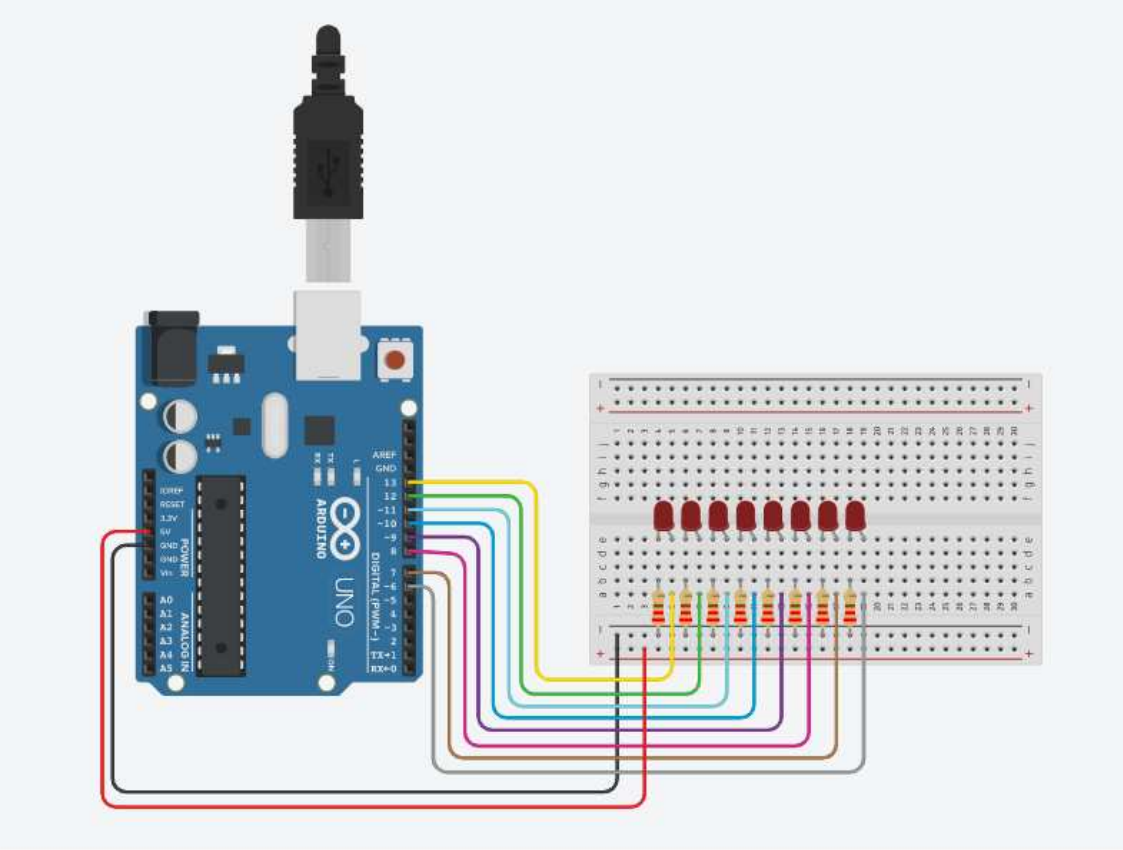
- 다음의 Protocol을 이용하여 LED를 원격 조정하자

Head	LED No	Status	CR
@	0~7	o,x	0x0d

- Ex) "@1o←" => 0x40 0x31 0x6f 0x0d => 1번 LED on
- Ex) "@1x←" => 0x40 0x31 0x78 0x0d => 1번 LED off



Wiring



A7-7 : Serial Remote

```
#define _Header '@'
#define _CR 0x0d

unsigned char LED[8]  = {13, 12, 11, 10, 9, 8, 7, 6};

unsigned char ReceiveString[10];
unsigned char ReceivePoint = 0;
bool stringComplete = false;

void setup( ) {
  Serial.begin(9600);
  for (int k = 0; k < 8; k++) pinMode(LED[k], OUTPUT);
}
```

```
void loop( ) {
  if (stringComplete) {
    Rx_Processing( );
  }
}
```

A7-7 : Serial Event/Processing

```
void serialEvent( ) {
  while (Serial.available()) {
    char inChar = (char)Serial.read();
    ReceiveString[ReceivePoint]=inChar;
    ReceivePoint++;
    if (inChar == _CR) {
       stringComplete = true;
       ReceivePoint=0;
    }
  }
}
```

```
void Rx_Processing( ){
  int LED_No=ReceiveString[1] & 0x0f;
  int LED_Status;
  if (ReceiveString[2]=='o') LED_Status = 1;
  else LED_Status = 0;

  digitalWrite(LED[LED_No], LED_Status);
  Serial.print("LED ");
  Serial.print(LED_No);
  Serial.print(" = ");
  if (LED_Status) Serial.println(" on");
  else Serial.println(" off");


  stringComplete=false;
}
```

Serial Monitor