

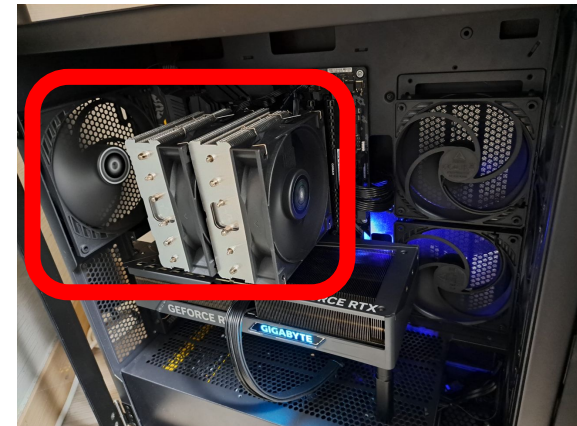
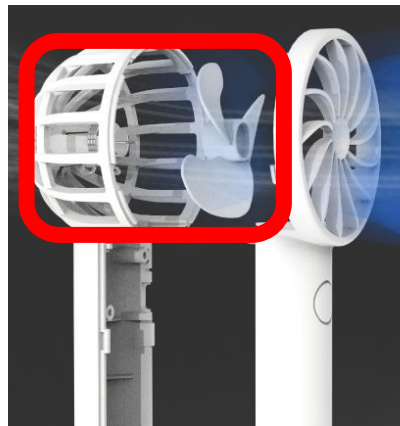
KOSS 겨울학기 봉사

1. 쿨링팬 실습

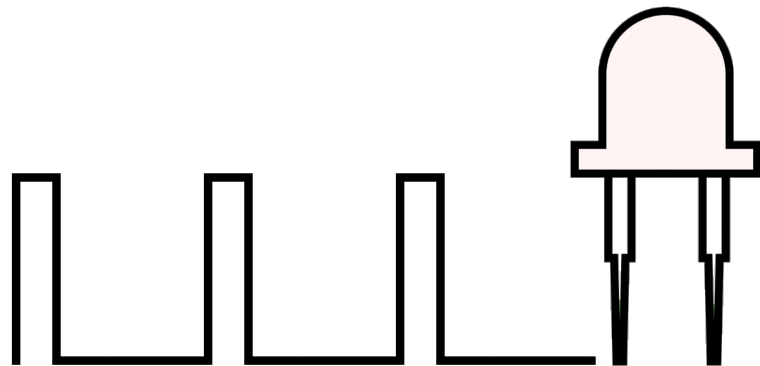


쿨링 팬(Cooling Fan)이란?

- DC모터를 사용하여 동작시키는 Fan으로 (+) 단자와 (-) 단자에 충분한 전원을 인가하여 동작시킬 수 있으며 **PWM 방식**으로 속도를 제어할 수 있습니다.
- 컴퓨터 내부 쿨링 팬이나, 손 선풍기등 전자기기에서 발생하는 열을 식히기 위한 쿨링 장치로 많이 이용됩니다.



1. 쿨링팬 실습



PWM 제어란?

- Pulse Width Modulation의 줄임말로 한국어로 “펄스 폭 변조”라는 의미입니다.
- 전기를 주는 간격을 활용하여 부품에 공급되는 평균 전압을 원하대로 바꾸는 기술
- Ex) 0V와 5V의 출력만 가능한 아두이노에서 2V 출력 내보내기

1. 쿨링팬 실습



쿨링팬 실습

1. "delay()" 함수를 이용하여 쿨링 팬 작동시간 제어하기
2. "analogWrite()" 함수를 이용하여 PWM 제어 해보기

1. 쿨링팬 실습

delay()

[Time]

설명

매개변수에 정한 시간(밀리 초) 동안 프로그램 멈춤. (1초는 1000 밀리 초)

문법

`delay(ms)`

매개변수

ms: 멈출 밀리 초 (unsigned long)

delay() 함수 사용법

- "delay(ms)"의 형식으로 사용합니다.
- 여기서 "ms"는 millisecond의 약자로, 1 / 1000초를 의미합니다.
- 따라서, 1000ms = 1s(초)가 됩니다.

1. 쿨링팬 실습

analogWrite()

[Analog I/O]

설명

아날로그 값(PWM 파)을 핀에 출력합니다. LED를 다양한 밝기로 켜거나 다양한 속도로 모터를 돌리는 데 쓸 수 있습니다. `analogWrite()` 를 호출하면, 해당 핀에 다음 `analogWrite()` (또는 `digitalRead()` 또는 `digitalWrite()`)가 불릴 때까지 특정 듀티 사이클의 구형파를 발생시킵니다. 대부분 핀에서 PWM 신호의 주파수는 약 490Hz입니다. Uno, 혹은 비슷한 보드에서, 5, 6번 핀은 약 980Hz의 주파수를 가집니다.

대부분의 아두이노 보드(ATmega168 또는 ATmega328P가 장착된)에서, 이 함수는 3, 5, 6, 9, 10, 11번 핀에서 동작합니다.

아두이노 메가에서는 2 - 13번 핀 그리고 44 - 46번 핀에서 동작합니다.

ATmega8을 탑재한 구형 아두이노 보드는 9, 10, 11번 핀에서만 `analogWrite()` 를 지원합니다.

아두이노 DUE는 2번 핀에서 13번 핀까지, 그리고 DAC0과 DAC1 핀에서 `analogWrite()` 를 지원합니다.

PWM 핀과 달리, DAC0과 DAC1은 디지털을 아날로그로 바꾸는 컨버터이며, 실제 아날로그 출력처럼 동작합니다.

`analogWrite()` 를 부르기 전에 핀을 출력으로 설정하기 위해 `pinMode()` 를 부를 필요는 없습니다.

`analogWrite` 함수는 아날로그 핀 또는 `analogRead` 함수와 아무 관련이 없습니다.

문법

```
analogWrite(pin, value)
```

매개변수

pin: 출력할 핀. 허용되는 자료형: int value: 듀티 사이클: 0 (언제나 꺼짐)과 255 (언제나 켜짐) 사이. 허용되는 자료형: int

반환

없음

analogWrite() 함수 사용법

- “`analogWrite(pin, value)`”의 형식으로 사용합니다.
- 여기서 “value”는 0(off) ~ 255(5v) 사이에서 설정할 수 있습니다.
- 단, “value”는 자연수여야만 합니다.
(1.2와 같은 소숫점 불가)

1. 쿨링팬 실습

쿨링팬 실습 - 1

- "delay()"함수를 이용하여 쿨링 팬 작동시간 제어하기

```
1 int Fan = 11;
2
3 void setup() {
4   pinMode(Fan, OUTPUT);
5 }
6
7 void loop() {
8   digitalWrite(FAN, HIGH);
9   delay(3000);
10  digitalWrite(FAN, LOW);
11  delay(3000);
12 }
```

쿨링 팬에 전원을 넣음

3초간 프로그램을 멈춤

쿨링 팬에 전원을 넣지 않음

3초간 프로그램을 멈춤

1. 쿨링팬 실습

쿨링팬 실습 - 2

- "analogWrite()" 함수를 이용하여 PWM 제어 해보기

```
1 int Fan = 11;
2
3 void setup() {
4   pinMode(Fan, OUTPUT);
5 }
6
7 void loop() {
8   analogWrite(FAN, 255);
9   delay(3000);
10  analogWrite(FAN, 0);
11  delay(3000);
12 }
```

← 쿨링 팬을 PWM 제어함

← 쿨링 팬에 전원을 넣지 않음

2. LCD(I2C) 디스플레이 실습



1602 (16x2)

LCD(I2C) 디스플레이 모듈이란?

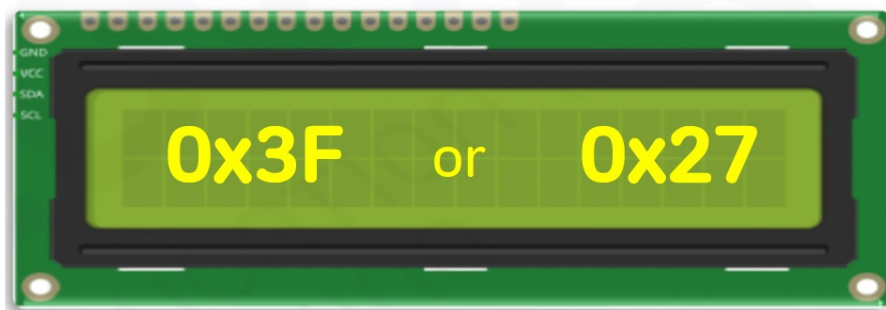
- LCD는 글자나 숫자를 표현할 수 있는 표시 장치입니다.
- 액정 표시 장치는 액체처럼 유체의 성질을 가지면서 고체처럼 광학적 성질을 가지는 물질로 액체와 고체의 중간 상태인 액정의 전기적 성질을 이용하여 문자를 표현합니다.
- 이 실습에서 사용하는 1602(16x2) 모델은 가장 보편적인 LCD로 뒷면에 I2C변환 모듈이 부착되어있어, 단 4개의 핀으로 쉽게 제어할 수 있도록 만들어진 모델입니다.
(I2C변환 모듈이 없는 경우, 약 10개의 핀이 사용해 제어해야함)

2. LCD(I2C) 디스플레이 실습



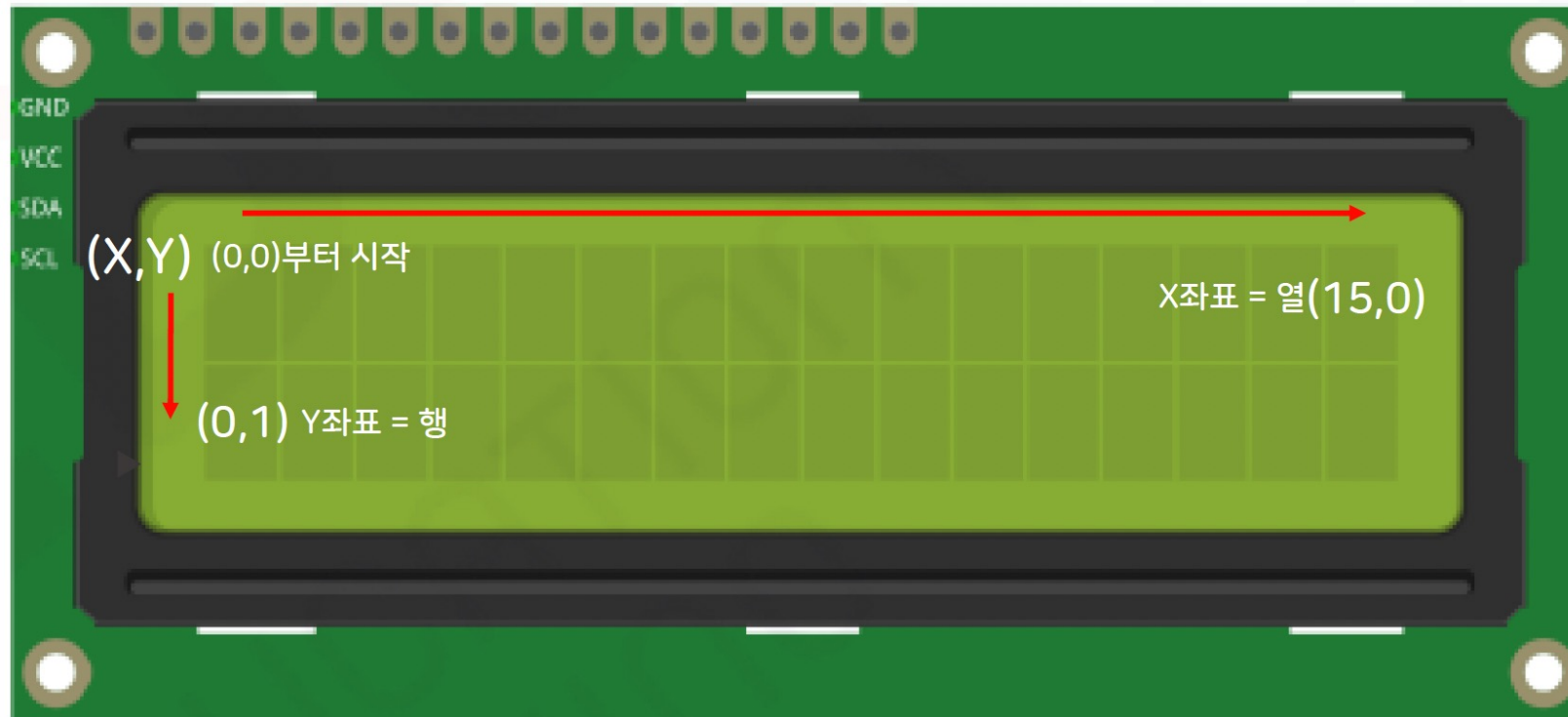
LCD(I2C) 모듈에 글자가 안보이는 경우

1. LCD 밝기 조절하기
 - 화면 창 밝기가 너무 밝거나 어두워서 보이지 않을 수 있습니다.
 - 가변 저항을 돌려 LCD 밝기 조절하기
2. 주소 변경하기
 - 접근 주소를 변경해보세요
 - 0x3F에서 0x27로, 0x27에서 0x3F로 변경해보세요.



2. LCD(I2C) 디스플레이 실습

LCD 좌표 값



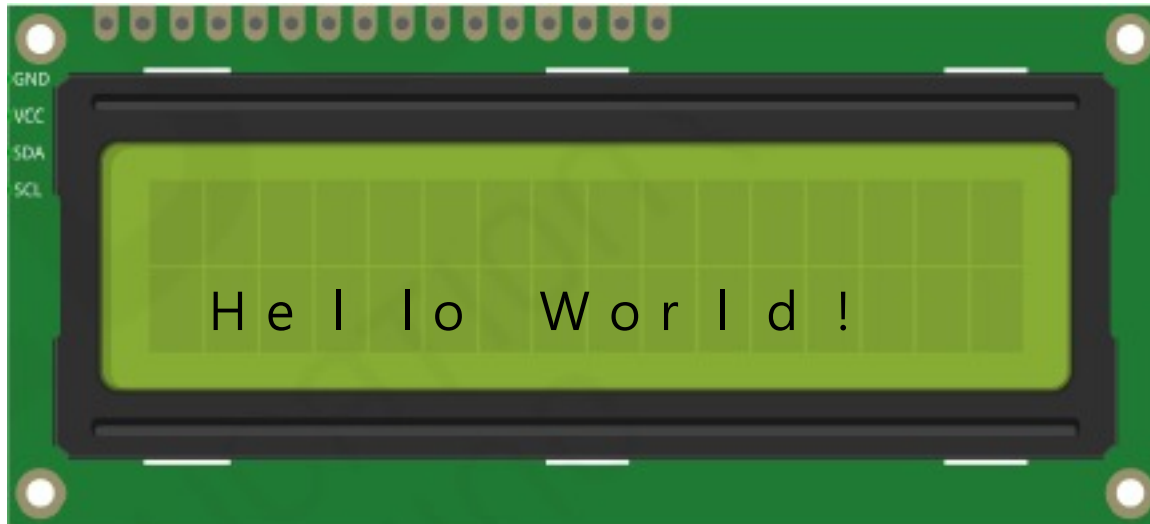
2. LCD(I2C) 디스플레이 실습



Q1

(0, 0)부터 시작해서 "Hello World!"를 출력하는 경우

```
lcd.setCursor(0, 0);  
lcd.print("Hello world!");
```

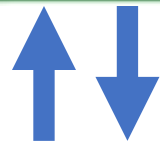


Q2

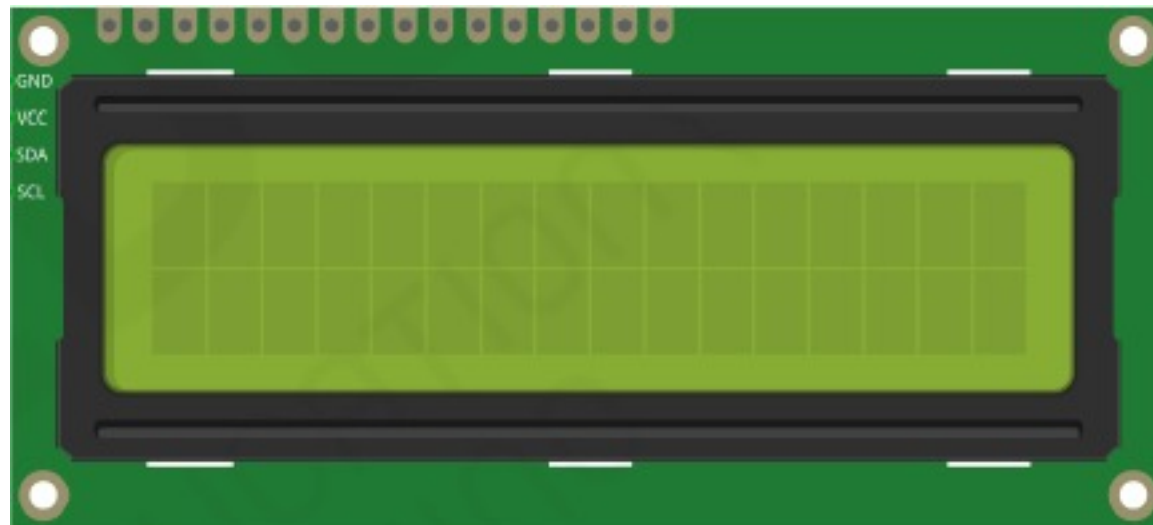
(2, 1)부터 시작해서 "Hello World!"를 출력하는 경우

```
lcd.setCursor(2, 1);  
lcd.print("Hello world!");
```

2. LCD(I2C) 디스플레이 실습



1초 간격으로 반복



만약, LCD화면을 초기화 하고 싶다면?

- "lcd.clear();"를 사용하면 됩니다.

```
void loop()
{
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print("Hello world!");
    delay(1000);
    lcd.clear();
    delay(1000);
}
```

2. LCD(I2C) 디스플레이 실습

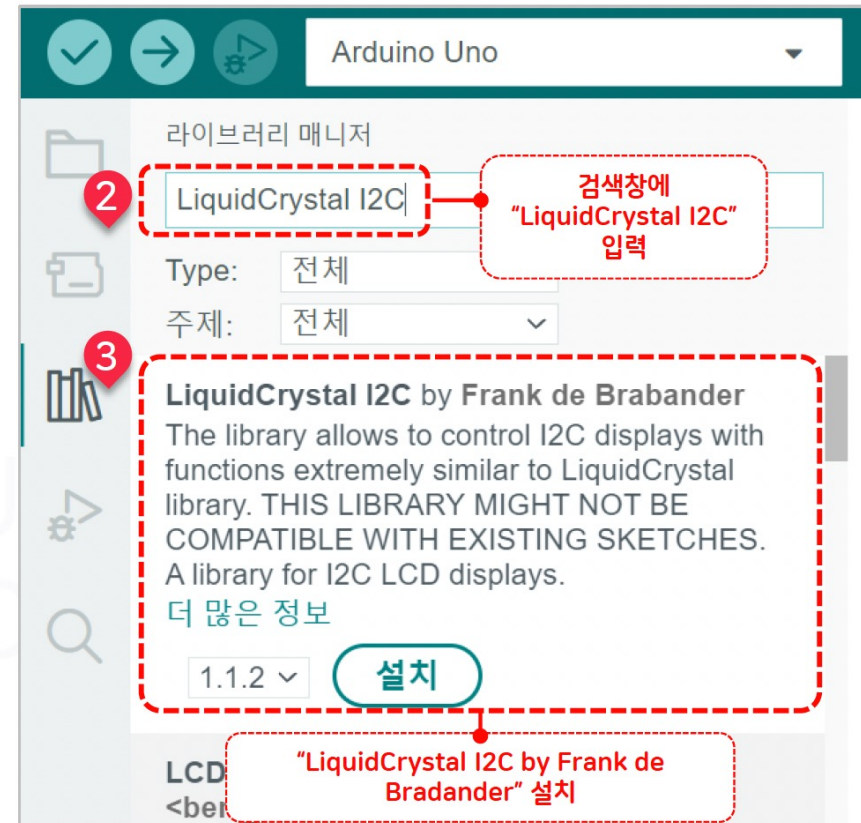
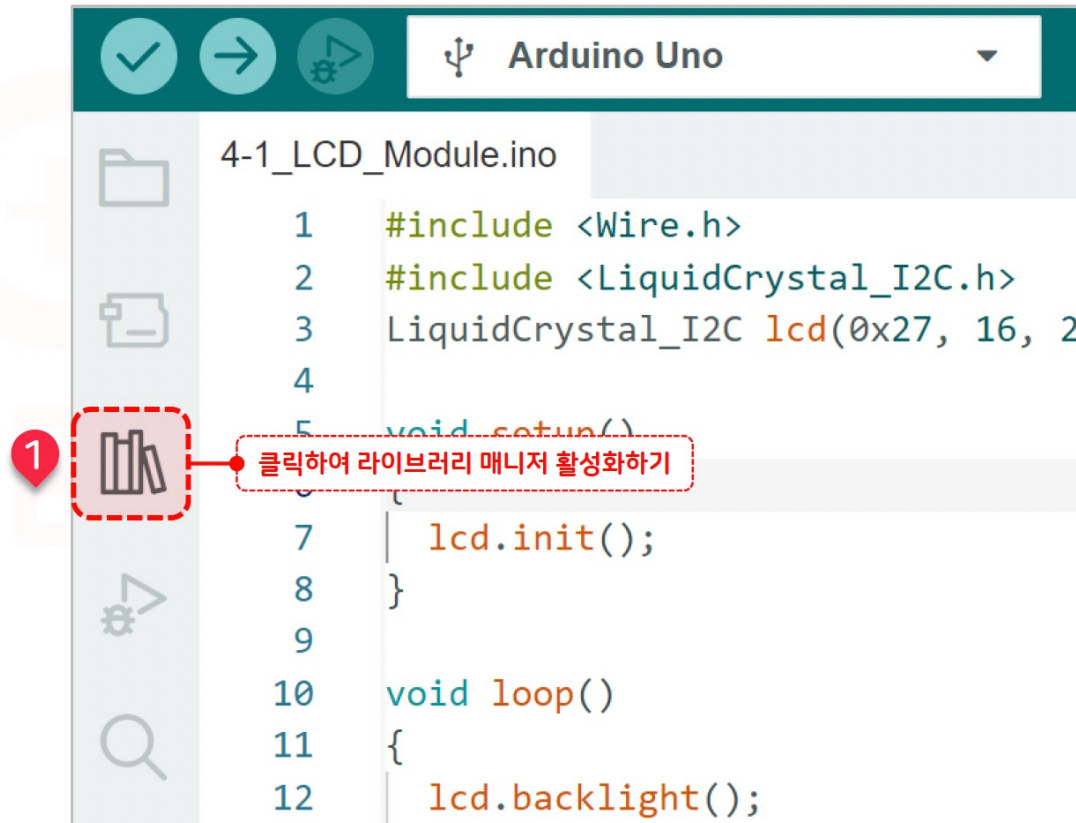


LCD(I2C) 디스플레이 모듈 실습

1. "Hello World"를 (0,0)에 출력해보기
2. (0,0)에 "Hello"를, (0,1)에 "World"를 출력해보기
3. "2번"에서 진행했던 내용을 3초 간격으로 문자가 출력되었다가, 지워졌다가를 반복해보기

2. LCD(I2C) 디스플레이 실습

LCD(I2C) 디스플레이 모듈 라이브러리 추가



2. LCD(I2C) 디스플레이 실습

```
1  #include <Wire.h>
2  #include <LiquidCrystal_I2C.h>
3  LiquidCrystal_I2C lcd(0x27, 16, 2);
4
5  void setup() {
6      lcd.init();
7  }
8
9  void loop() {
10     lcd.backlight();
11     lcd.setCursor(0, 0);
12     lcd.print("Hello World");
13 }
```

LCD(I2C) 디스플레이 모듈 실습 - 1

- "Hello World"를 (0,0)에 출력해보기

백라이트 ON

커서를 (0,0)으로 설정

"Hello World"를 출력

2. LCD(I2C) 디스플레이 실습

```
1  #include <Wire.h>
2  #include <LiquidCrystal_I2C.h>
3  LiquidCrystal_I2C lcd(0x27, 16, 2);
4
5  void setup() {
6      lcd.init();
7  }
8
9  void loop() {
10     lcd.backlight();
11     lcd.setCursor(0, 0);
12     lcd.print("Hello");
13     lcd.setCursor(0, 1);
14     lcd.print("World");
15 }
```

LCD(I2C) 디스플레이 모듈 실습 - 2

- (0,0)에 "Hello"를, (0,1)에 "World"를 출력해보기

2. LCD(I2C) 디스플레이 실습

```
1  #include <Wire.h>
2  #include <LiquidCrystal_I2C.h>
3  LiquidCrystal_I2C lcd(0x27, 16, 2);
4
5  void setup() {
6      lcd.init();
7  }
8
9  void loop() {
10     lcd.backlight();
11     lcd.setCursor(0, 0);
12     lcd.print("Hello");
13     lcd.setCursor(0, 1);
14     lcd.print("World");
15     delay(3000);
16     lcd.clear();
17     dleay(3000);
18 }
```

LCD(I2C) 디스플레이 모듈 실습 - 3

- "2번"에서 진행했던 내용을 3초 간격으로 문자가 출력되었다가, 지워졌다가를 반복해보기

← lcd의 모든 문자를 지움

3. 블루투스 모듈(HC-06) 실습



블루투스 모듈 HC-06

블루투스 모듈(HC-06)이란?

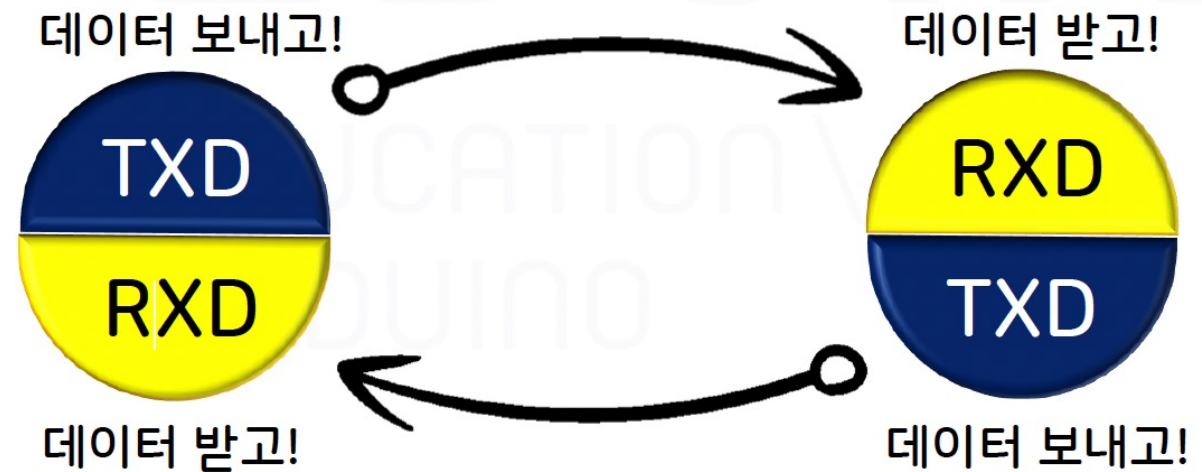
- 1994년에 최초 개발된 근거리 무선 통신 기술입니다
- HC 0606은 시리얼 통신을 이용하여 데이터 값을 주고 받을 수 있는 모듈입니다
- 전자기기와 아두이노에서 다양하게 활용이 됩니다.
- HC-06같은 경우 아이폰(iOS) 에서 사용이 불가능합니다.

3. 블루투스 모듈(HC-06) 실습

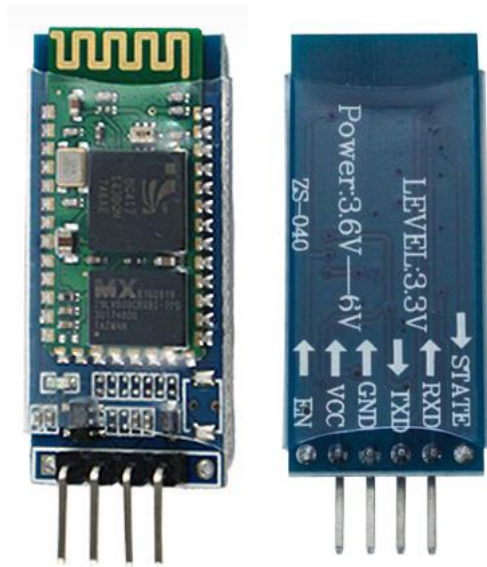


블루투스 모듈(HC-06)과 통신하는 방법

- RXD (Receive data): 데이터를 수신하는 역할을 합니다.
- TXD (Transmit 데이터를 전송하는 역할을 합니다.



3. 블루투스 모듈(HC-06) 실습



블루투스 모듈(HC-06) 실습

1. 블루투스 모듈과 통신해보기
2. 블루투스 모듈의 이름 변경하기

3. 블루투스 모듈(HC-06) 실습

```
1  #include <SoftwareSerial.h>
2
3  #define BT_RXD 5
4  #define BT_TXD 4
5  SoftwareSerial bluetooth(BT_RXD, BT_TXD);
6
7  void setup() {
8      Serial.begin(9600);
9      bluetooth.begin(9600);
10 }
11
12 void loop() {
13     if (bluetooth.available()) {
14         Serial.write(bluetooth.read());
15     }
16     if (Serial.available()) {
17         bluetooth.write(Serial.read());
18     }
19 }
```

블루투스 모듈의 TX, RX 핀을 설정함

블루투스 모듈을 사용하기 위해 bluetooth라는 변수를 만듦

블루투스 모듈(HC-06) 실습 - 1

- 블루투스 모듈과 통신해보기

블루투스 모듈과 시리얼 모니터를 이용한 통신을 준비함

블루투스 모듈에서 보낸 정보가 있다면

시리얼 모니터에 보여줌

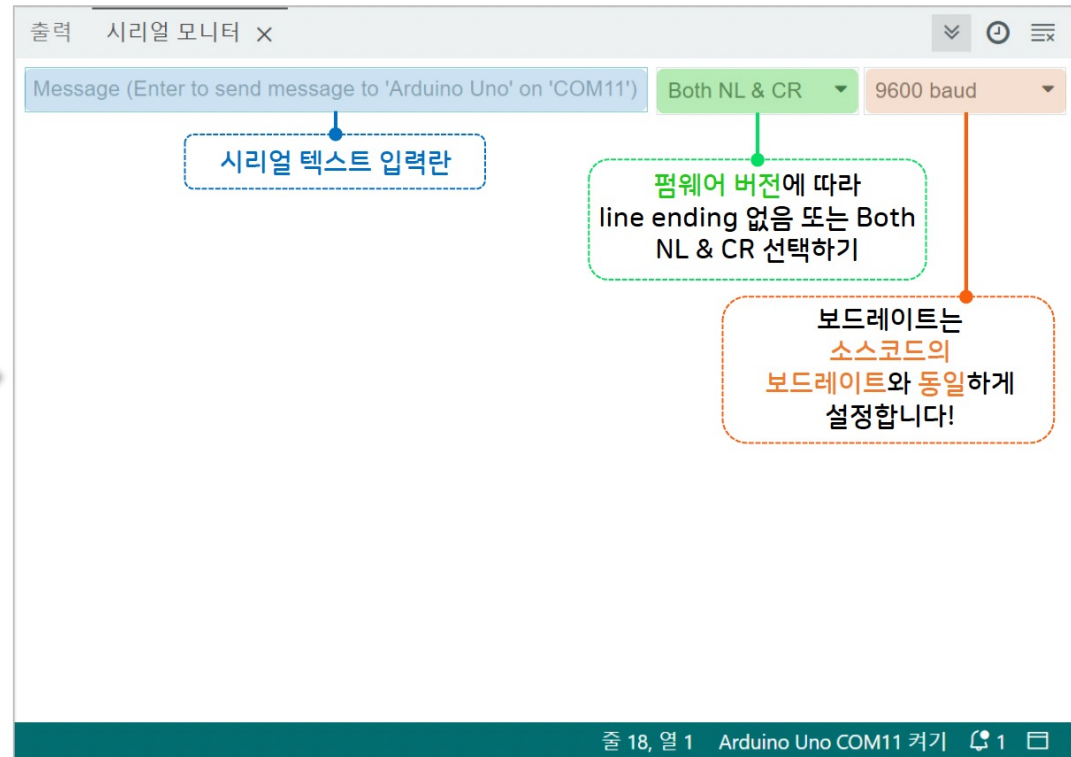
시리얼 모니터에서 입력이 있으면

블루투스 모듈로 전송함

3. 블루투스 모듈(HC-06) 실습

블루투스 모듈(HC-06) 실습 - 1

- 블루투스 모듈과 통신해보기



3. 블루투스 모듈(HC-06) 실습

블루투스 모듈(HC-06) 실습 - 2

- 블루투스 모듈의 이름 변경하기

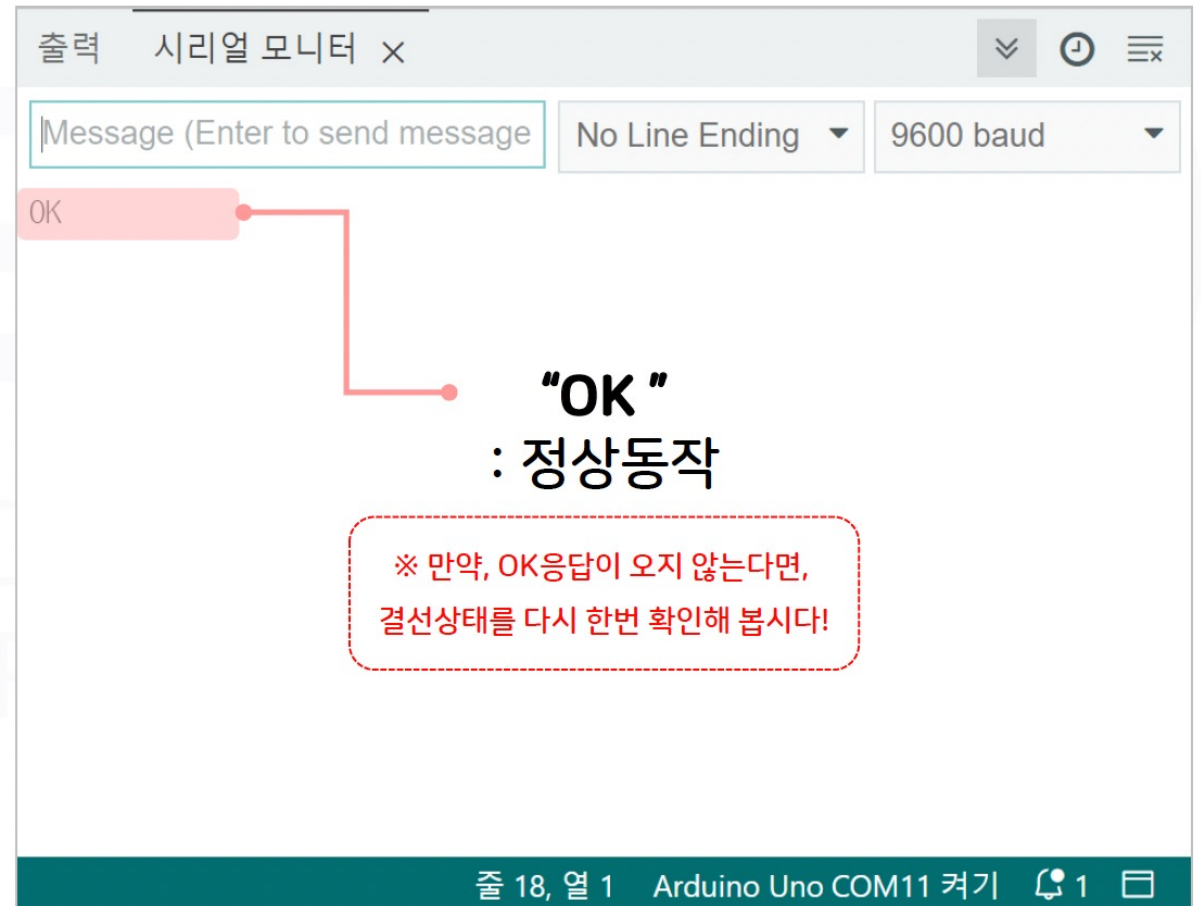
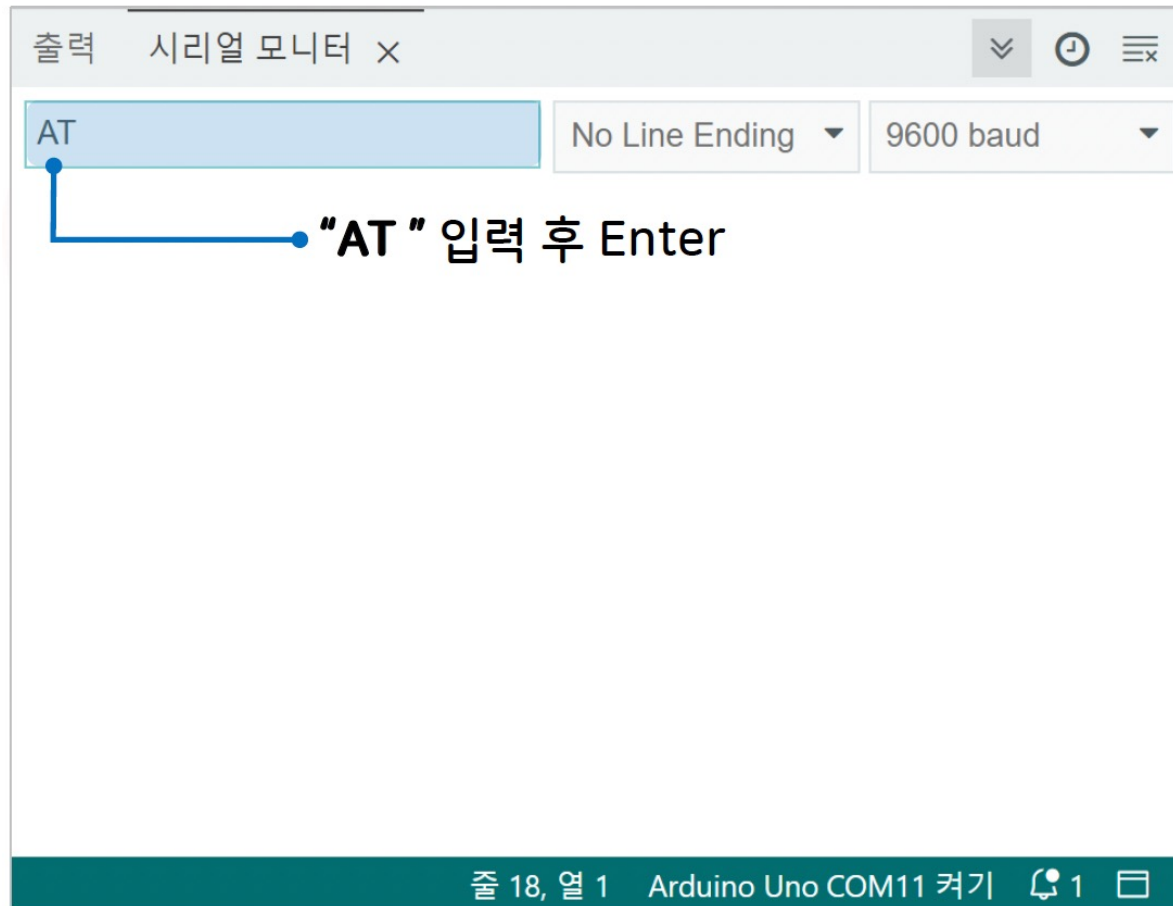
1.8 버전	AT 명령어	응답	3.0 버전	AT 명령어	응답
통신상태 확인	AT	OK	통신상태 확인	AT	'OK' 출력
이름 변경	AT+NAME[이름]	OKsetname	이름 변경	AT+NAME=[이름]	'OK' 출력
비밀번호 변경	AT+PIN[비밀번호]	OKsetPIN	이름 확인	AT+NAME?	'+NAME : [이름]' 과 'OK' 출력
버전 확인	AT+VERSION	OKlinvorV1.8 출력	비밀번호 변경	AT+PSWD="비밀번호"	'OK' 출력
			비밀번호 확인	AT+PSWD?	+PIN : "비밀번호" 와 'OK' 출력
			버전 확인	AT+VERSION	'VERSION:3.0-20170609'와 'OK'가 출력

Q) 블루투스 모듈의 이름을 꼭 바꿔야하나요?
A) 네, 다수의 인원이 동시에 블루투스 모듈을 사용할 경우,
나의 블루투스 모듈을 찾기가 매우 어렵습니다

3. 블루투스 모듈(HC-06) 실습

블루투스 모듈(HC-06) 실습 - 2

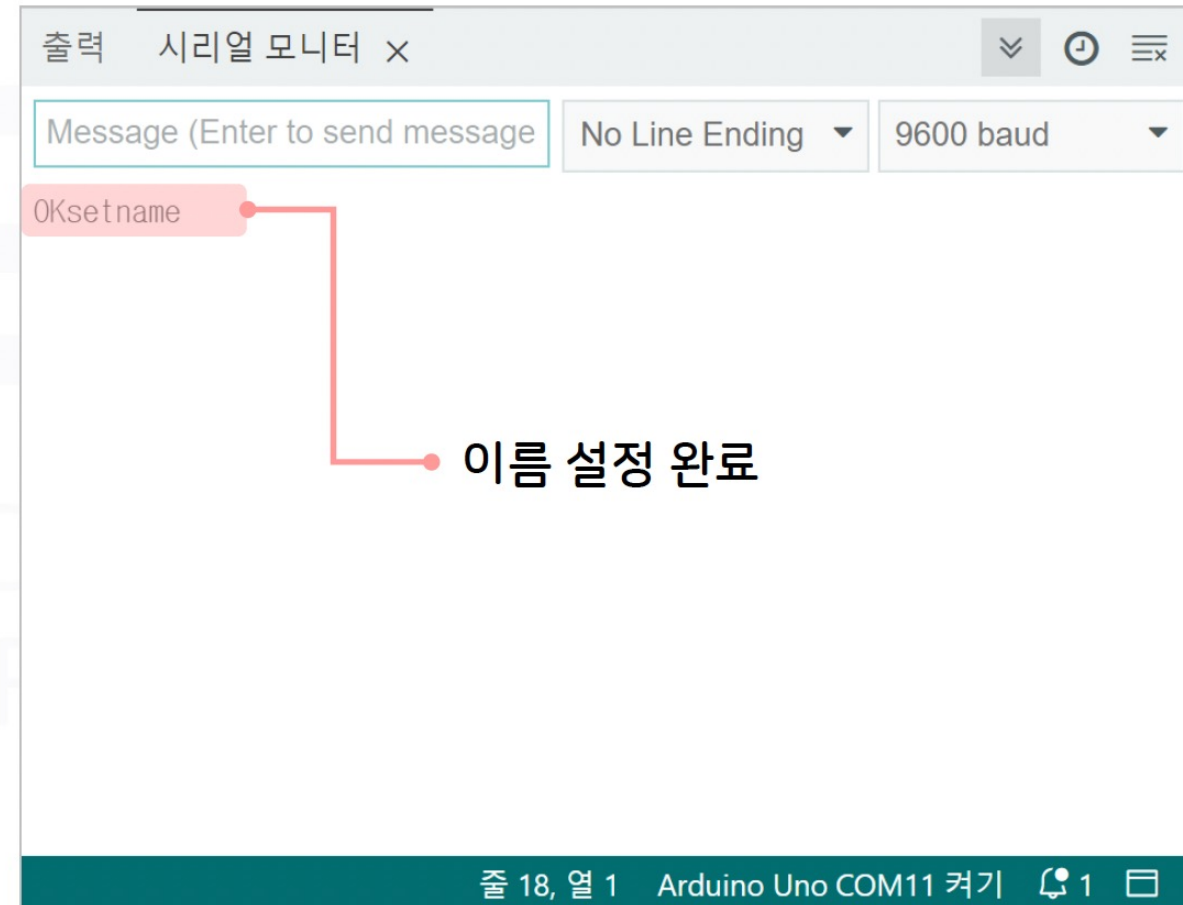
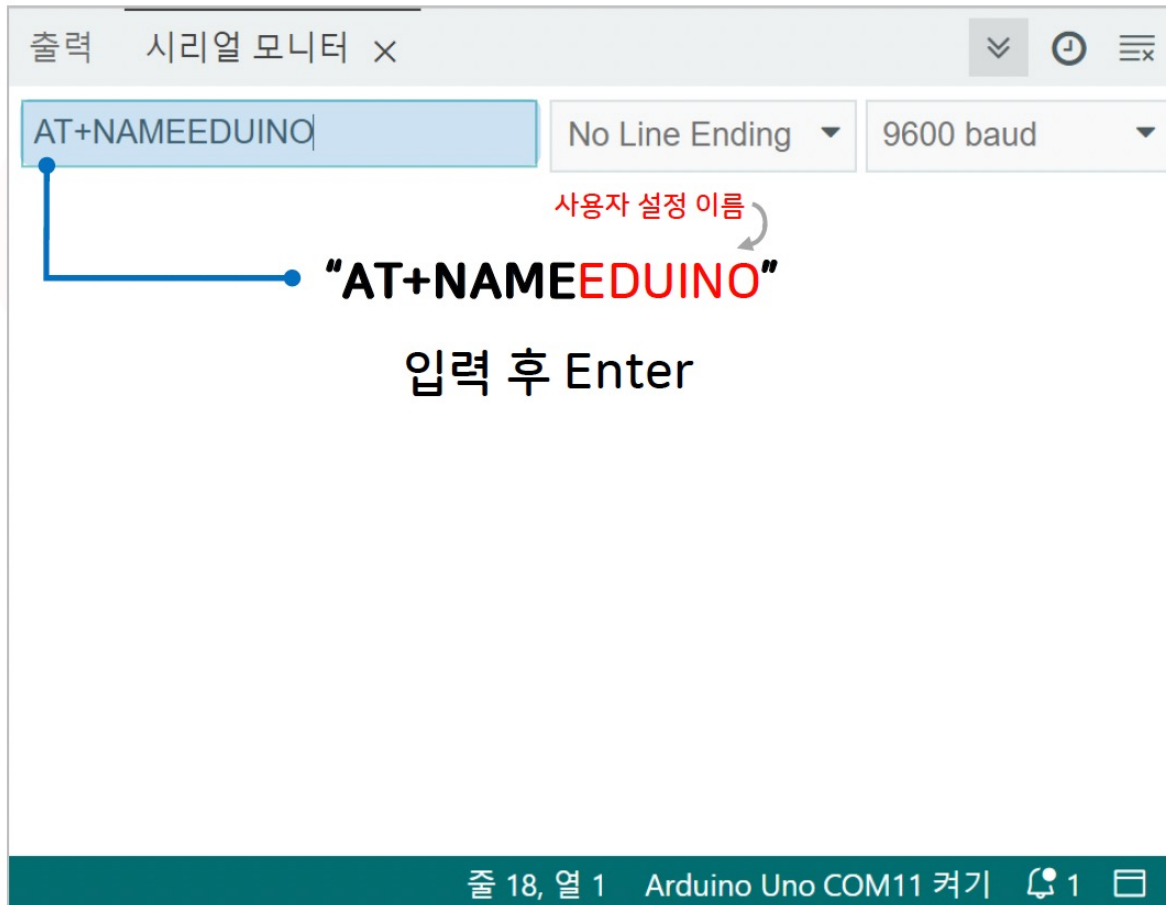
- 블루투스 모듈의 이름 변경하기



3. 블루투스 모듈(HC-06) 실습

블루투스 모듈(HC-06) 실습 - 2

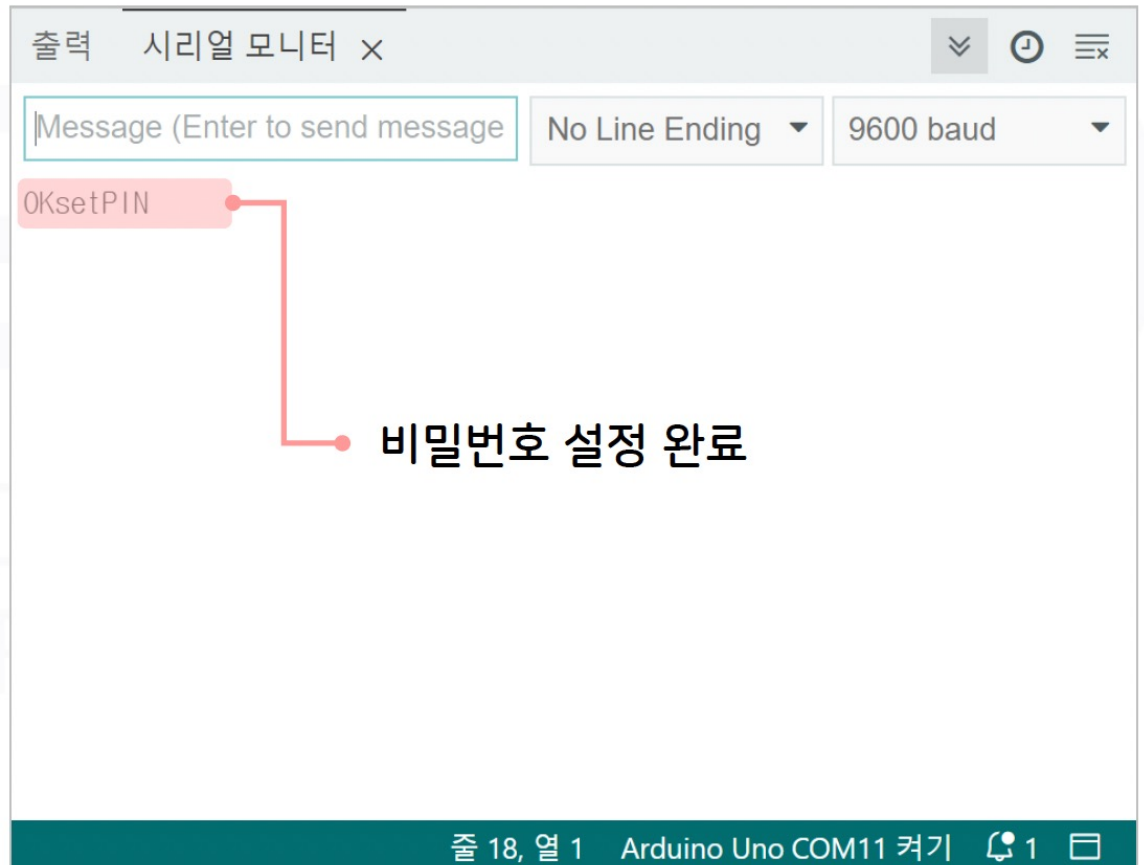
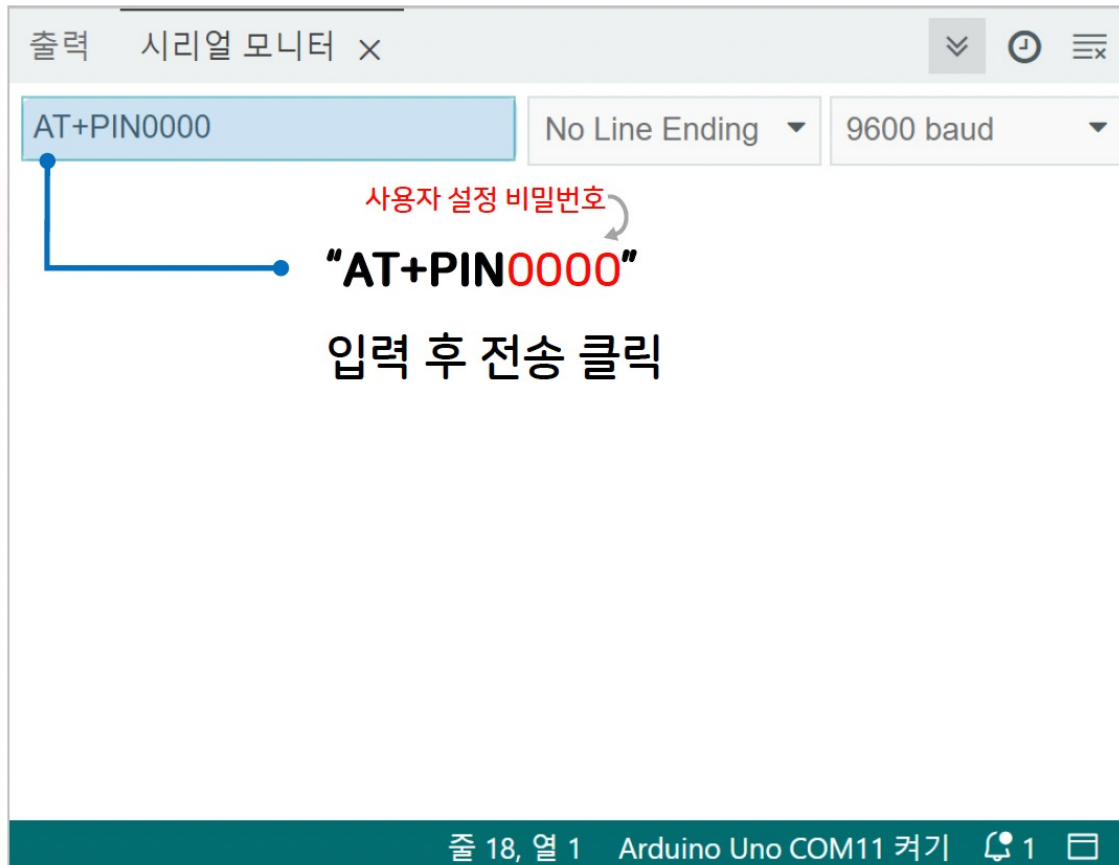
- 블루투스 모듈의 이름 변경하기



3. 블루투스 모듈(HC-06) 실습

블루투스 모듈(HC-06) 실습 - 2

+ 블루투스 모듈의 비밀번호 변경하기



- ✓ 비밀번호를 바꾸는 경우, 다른 사람이 나의 블루투스 모듈과 연결하는 것을 막을 수 있습니다. 다만, 잊어버리는 경우, 다시 이 과정을 거쳐 수정해줘야 한다는 것을 유의해주세요.