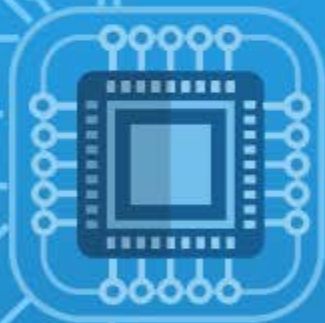


전병현 교수

IoT의 이해

4주차 3강. I2C 통신 실습



학습내용

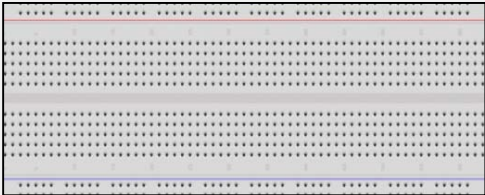
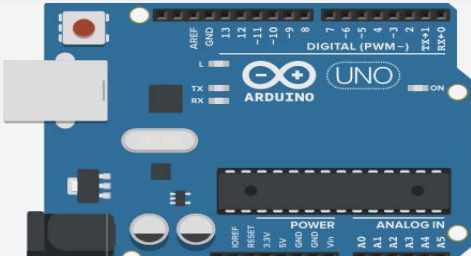

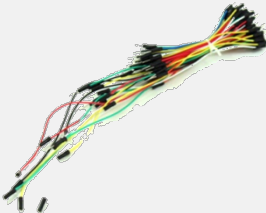
1. 아두이노 I2C 통신하기
2. 마스터/슬레이브 I2C 통신 : LED 제어하기

학습목표

1. 아두이노 I2C 통신을 실행할 수 있다.
2. 마스터/슬레이브 I2C 통신을 사용해 LED를 제어할 수 있다.

1. 아두이노 I2C 통신하기(1)

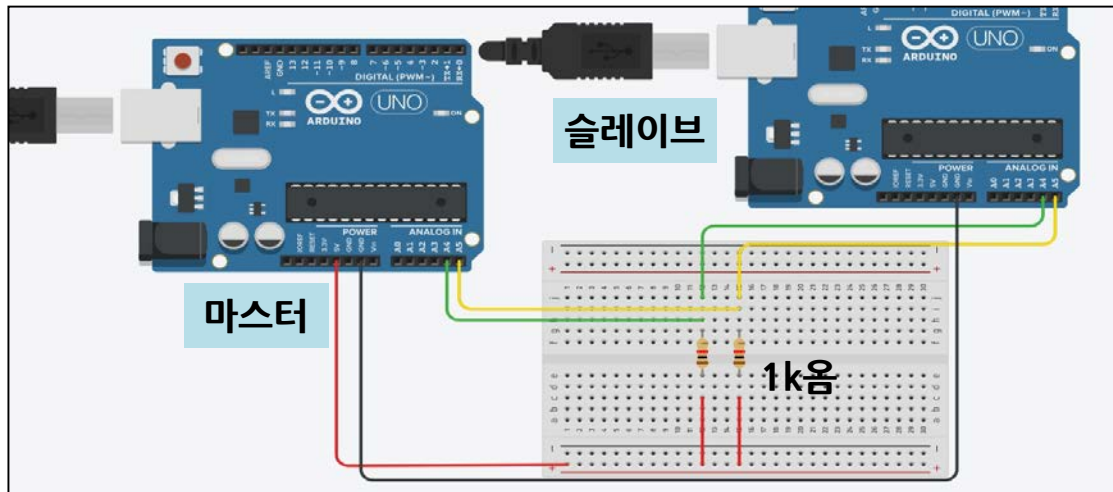
실습 준비물

이미지			
준비물	브레드 보드(1개)	아두이노 보드(2개)	
이미지			
준비물	1k Ω 저항(2개)	점프선 소량	

1. 아두이노 I2C 통신하기(2)

회로도 구성(1)

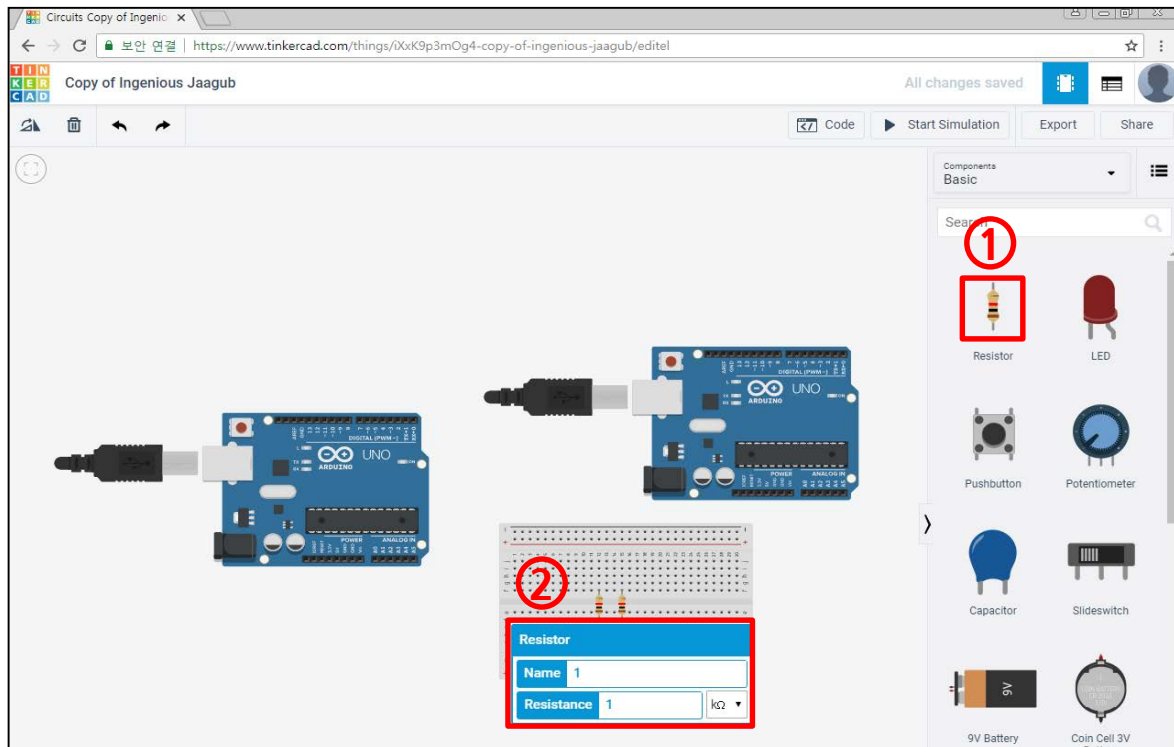
- 5V전원에 1k옴 풀업 저항을 연결
- 마스터 아두이노 A4(SDA) → 슬레이브 아두이노 A4(SDA)
- 마스터 아두이노 A5(SCL) → 슬레이브 아두이노 A5(SCL)



• 출처 : AUTODESK - TINKERCAD. <https://www.tinkercad.com>

1. 아두이노 I2C 통신하기(3)

회로도 구성(2)



• 출처 : AUTODESK - TINKERCAD. <https://www.tinkercad.com>

1. 아두이노 I2C 통신하기(4)

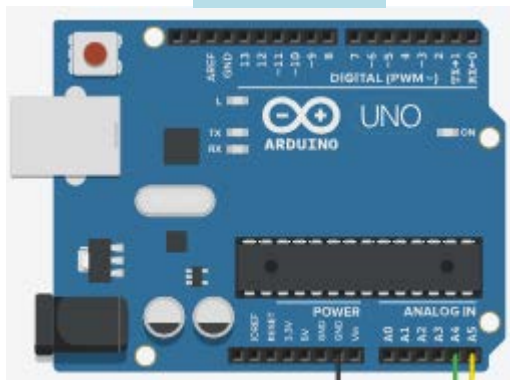
회로도 구성(3)

마스터



5V전원공급

슬레이브



A4 - A4
A5 - A5
GND - GND

1. 아두이노 I2C 통신하기(5)

◎ 실습 소스 : 마스터 코드

```
#include <Wire.h>    // Wire.h 라이브러리 사용
void setup() {
  Wire.begin();      // 인자 값이 없으면 마스터용임을 나타냄
  Serial.begin(9600); // 시리얼 모니터 시작
}
void loop() {
  Wire.requestFrom(7, 6); // 슬레이브 (주소, 바이트)가져오기
  while (Wire.available()) { // 읽을 데이터가 있다면
    char c = Wire.read();    // 문자형 변수 c에 저장
    Serial.print(c);        // 받은 문자를 시리얼 모니터에 출력
  }
  delay(500);
}
```

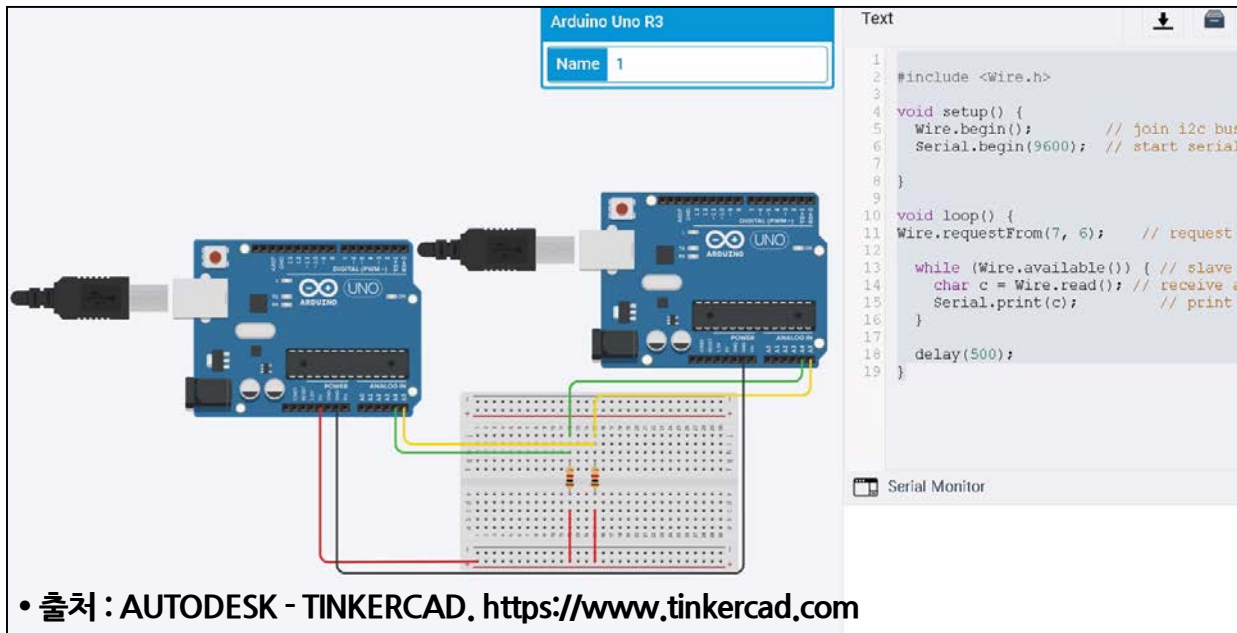
1. 아두이노 I2C 통신하기(6)

◎ 실습 소스 : 슬레이브 코드

```
#include <Wire.h>           // Wire.h 라이브러리 사용
void setup() {
    Wire.begin(7);          // 슬레이브 주소 값
    Wire.onRequest(requestEvent); // 마스터에서 요청이 오면
}                             // requestEvent()함수 호출
void loop() {
    delay(100);              // 대기 상태
}
void requestEvent() {        // requestEvent()함수
    Wire.write("hello ");    // 마스터에게 보낼 데이터
}
```


1. 아두이노 I2C 통신하기(7)

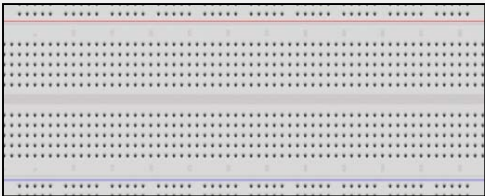
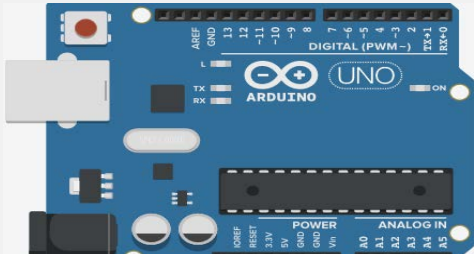



결과 확인



- 슬레이브에서 보내는 “hello” 메시지를 마스터 모니터에 표시

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(1)

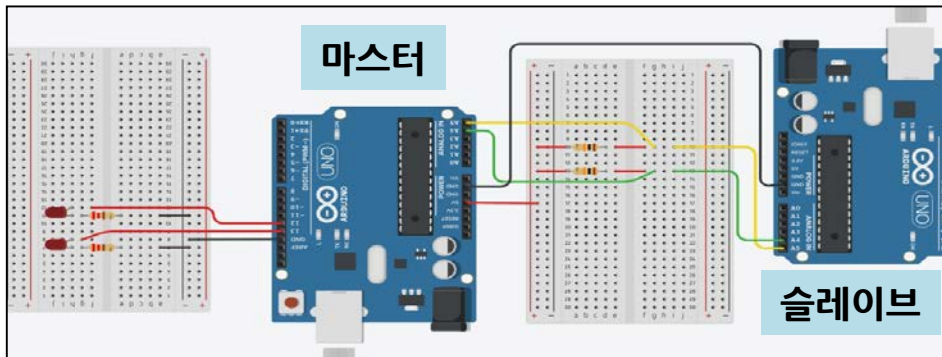
실습 준비물

이미지			
준비물	브레드 보드(2개)		아두이노 보드(2개)
이미지			
준비물	LED(2개)	220 Ω 저항(2개) 1k Ω 저항(2개)	점프선 소량

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(2)

회로도 구성(1)

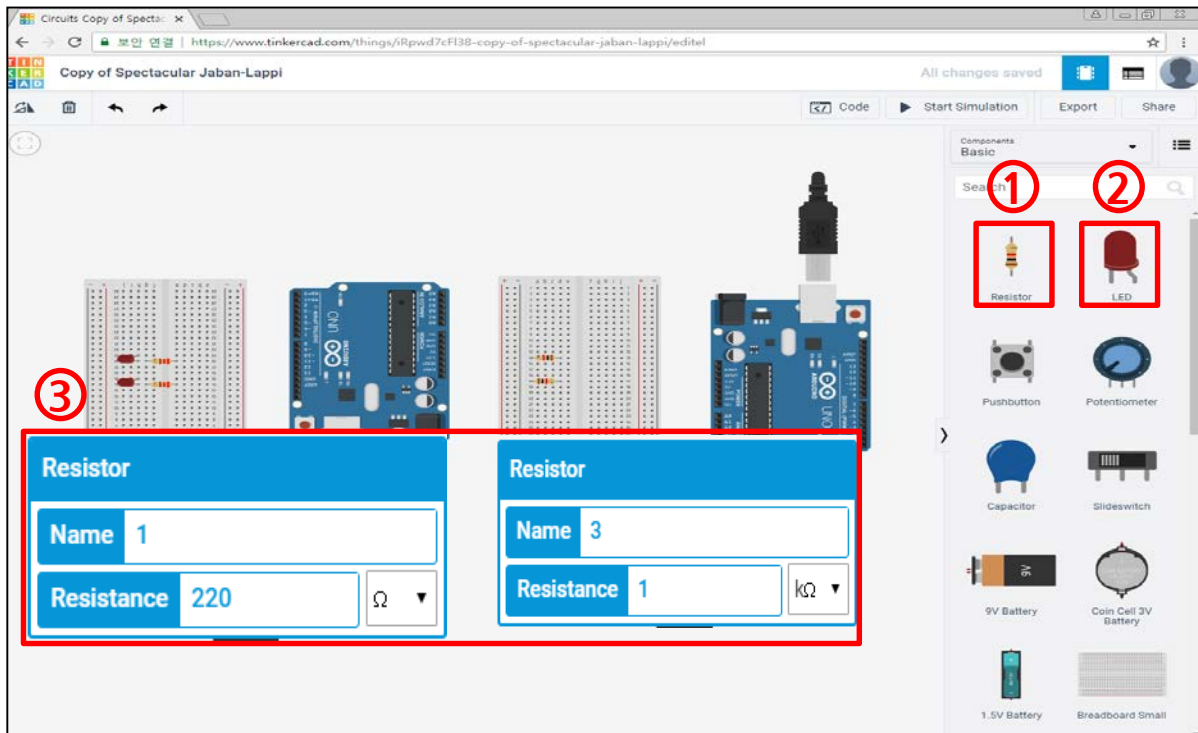
- 5V전원에 1k옴 풀업 저항을 연결
- 마스터 아두이노 A4(SDA) → 슬레이브 아두이노 A4(SDA)
- 마스터 아두이노 A5(SCL) → 슬레이브 아두이노 A5(SCL)
- 마스터 13번 핀과 LED_01 (+), 12번 핀과 LED_02 (+) 연결
- LED의 (-)는 220옴 저항을 연결 후 아두이노 GND에 연결



• 출처 : AUTODESK - TINKERCAD. <https://www.tinkercad.com>

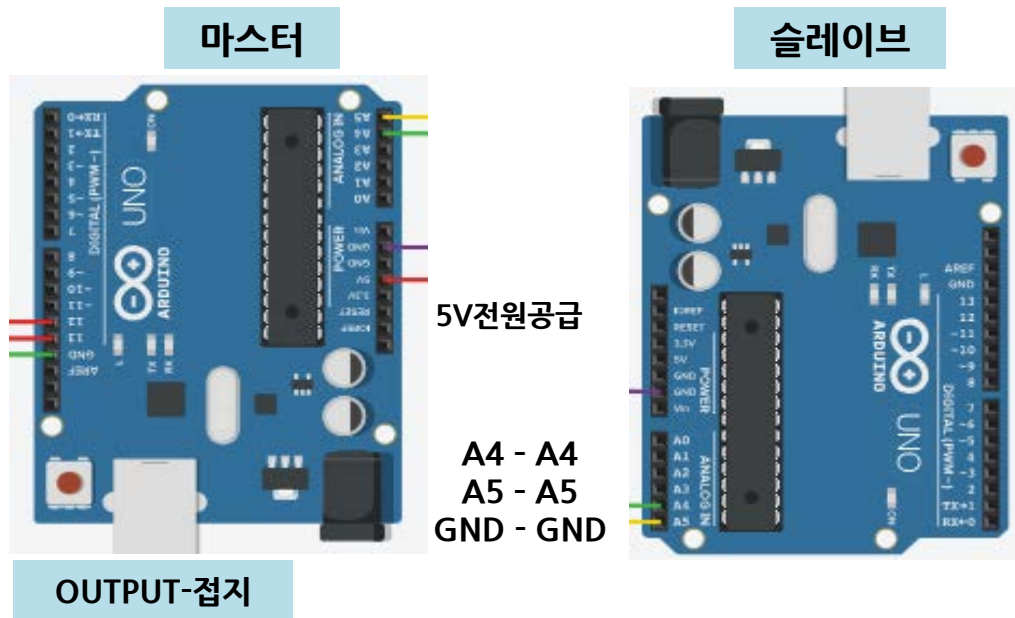
2. 마스터/슬레이브 I2C 통신 : LED 제어하기(3)

회로도 구성(2)



• 출처 : AUTODESK - TINKERCAD. <https://www.tinkercad.com>

회로도 구성(3)



•출처 : AUTODESK - TINKERCAD, <https://www.tinkercad.com>

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(5)

○ 실습 소스 : 마스터 코드(1)

```
#include <Wire.h>           // Wire.h 라이브러리 사용
void setup() {
  pinMode(13,OUTPUT);       // LED 핀 번호
  pinMode(12,OUTPUT);
  Wire.begin();             // 인자 값이 없으면 마스터용임
  Serial.begin(9600)        // 시리얼 모니터 시작
}
void loop() {
  Wire.requestFrom(7, 6);   // 슬레이브(주소, 바이트)가져오기
  while (Wire.available())  // 읽을 데이터가 있다면
    char c = Wire.read();   // 문자형 변수 c에 저장
  Serial.print(c);          // 받은 문자를 시리얼 모니터에 출력
  if(c == 65){              // 받은 문자가 A라면
    digitalWrite(13, HIGH); // 13번 핀에 연결된 LED ON
  }
```

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(6)

실습 소스 : 마스터 코드(2)

```
delay(1000);           // 1초 delay
}
if(c == 66) {          // 받은 문자가 B이면
    digitalWrite(12, HIGH); // 12번 핀에 연결된 LED ON
    delay(1000);
}
else{                  // 위 조건에 맞지 않으면
    digitalWrite(13, LOW); // LED OFF
    digitalWrite(12, LOW);
    delay(500);         // 0.5초 delay
}
}
```

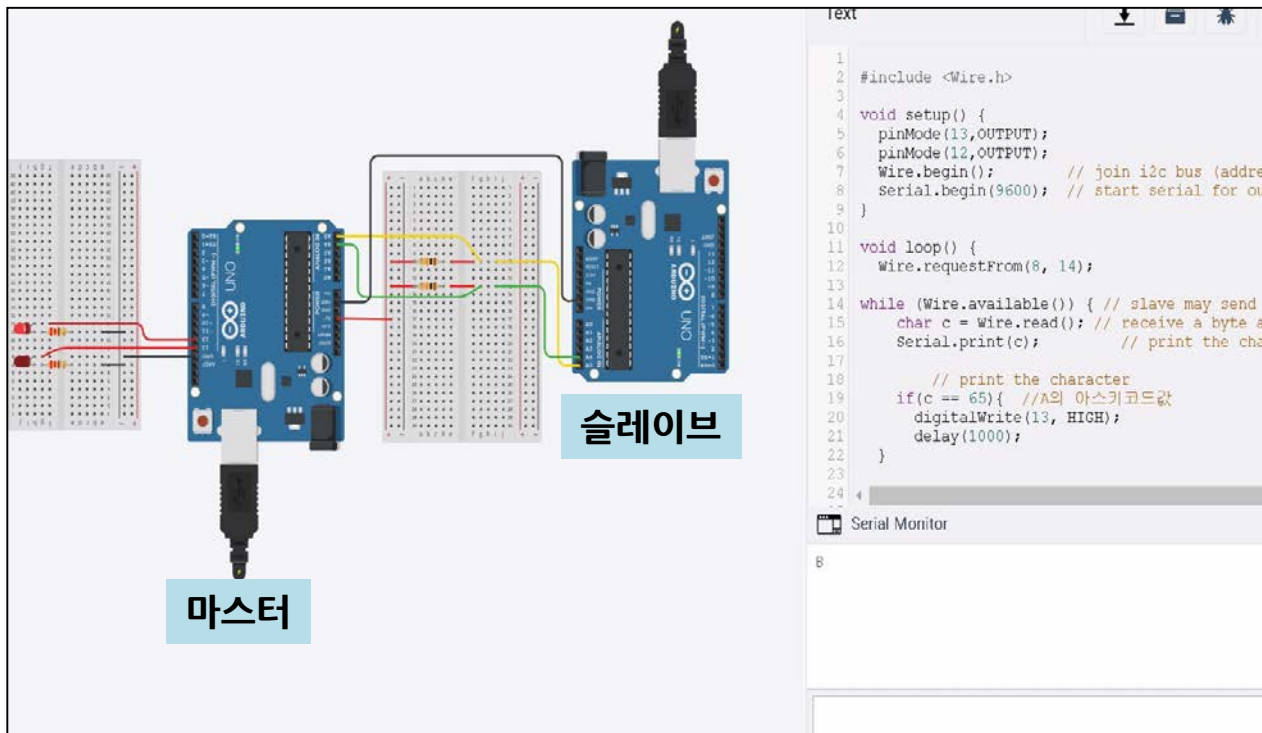
2. 마스터/슬레이브 I2C 통신 : LED 제어하기(7)

④ 실습 소스 : 슬레이브 코드

```
#include <Wire.h>                // Wire.h 라이브러리 사용
void setup() {
  Wire.begin(8);                 // 슬레이브 주소값(8)
  Wire.onRequest(requestEvent); // 마스터에서 요청이 오면
                                // requestEvent()함수 호출
}
void loop() {                   // 마스터에서 요청이 올 때까지 기다림
  delay(100);
}
                                // 슬레이브 (주소, 바이트)가져오기
void requestEvent() {          // requestEvent()함수
  Wire.write("01B23A456B789A "); // 마스터에게 보낼 데이터
}
```


2. 마스터/슬레이브 I2C 통신 : LED 제어하기(8)

결과 확인



• 출처 : AUTODESK - TINKERCAD. <https://www.tinkercad.com>

프로그램 실습





※ 자료실에서 프로그램 소스 코드를 다운로드 받으세요.

1. 아두이노와 I2C 통신하기(1)

실습 준비물

1. 아두이노 I2C 통신하기

실습 준비물

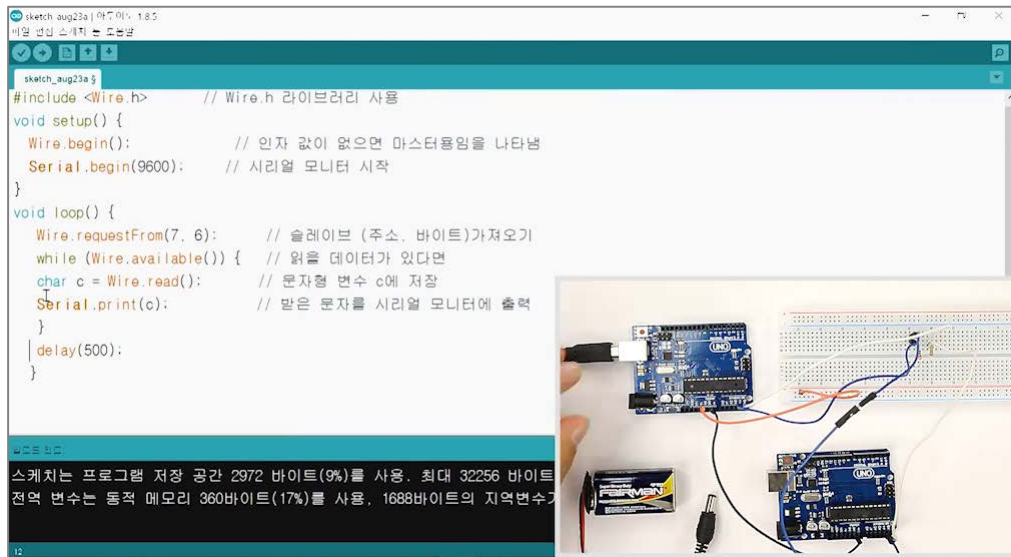
이미지		
		
준비물	브레드 보드(1개)	아두이노 보드(2개)
이미지		
준비물	1kΩ 저항(2개)	점프선 소량

실습

1. 실습 준비물 : 아두이노보드(2개), 브레드보드(1개), 1kΩ 저항(2개)
2. 두 개의 아두이노 보드간의 I2C 통신을 실행한다.

1. 아두이노와 I2C 통신하기(2)

마스터 실습 소스




실습

1. 두 개의 아두이노 보드를 브레드 보드(마스터 보드 - 슬레이브 보드)와 연결한다.
2. `wire.h` 라이브러리를 탑재한다.
3. `wire.begin()` : 인자 값이 없으면 마스터 프로그램
4. `wire.requestFrom(7, 6)` : 슬레이브(주소, 바이트)를 가져오는 함수

1. 아두이노와 I2C 통신하기(3)

슬레이브 실습 소스



```
sketch_sep13a | 아두이노 1.8.5
비밀 보안 스케치 | 도움말

sketch_sep13a
#include <Wire.h> // Wire.h 라이브러리 사용

void setup() {
  Wire.begin(7); // 슬레이브 주소 값
  Wire.onRequest(requestEvent); // 마스터에서 요청이 오면 requestEvent() 함수 호출
}

void loop() {
  delay(100); // 대기 상태
}

void requestEvent() { // requestEvent () 함수
  Wire.write("hello "); // 마스터에게 보낼 데이터
}

40도 온도
스케치는 프로그램 저장 공간 2148 바이트(6%)를 사용. 최대 32256 바이트.
전역 변수는 동적 메모리 197바이트(9%)를 사용, 1851바이트의 지역변수가 남음. 최대는 2048 바이트.
```

실습

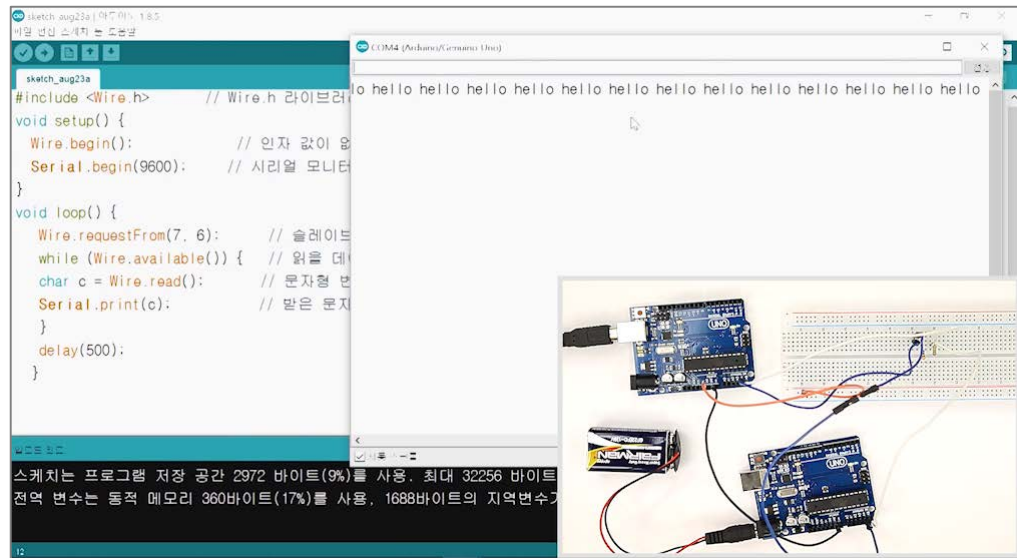
5. `wire.begin(7)` : 슬레이브 주소 값

6. `Wire.onRequest(requestEvent)` : `requestEvent` 이벤트 함수를 호출

7. `Void requestEvent()` : `requestEvent()` 함수

1. 아두이노와 I2C 통신하기(4)

● 실행하기



실습





1. 슬레이브 소스를 컴파일한 후 파일을 업로드 한다.
2. 마스터 소스를 컴파일한 후 파일을 업로드 한다.
3. 파일을 업로드 하기 전 반드시 아두이노 포트가 설정 되었는지 확인한다.

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(1)

실습 준비물

2. 마스터/슬레이브 I2C 통신 : LED 제어하기

실습 준비물

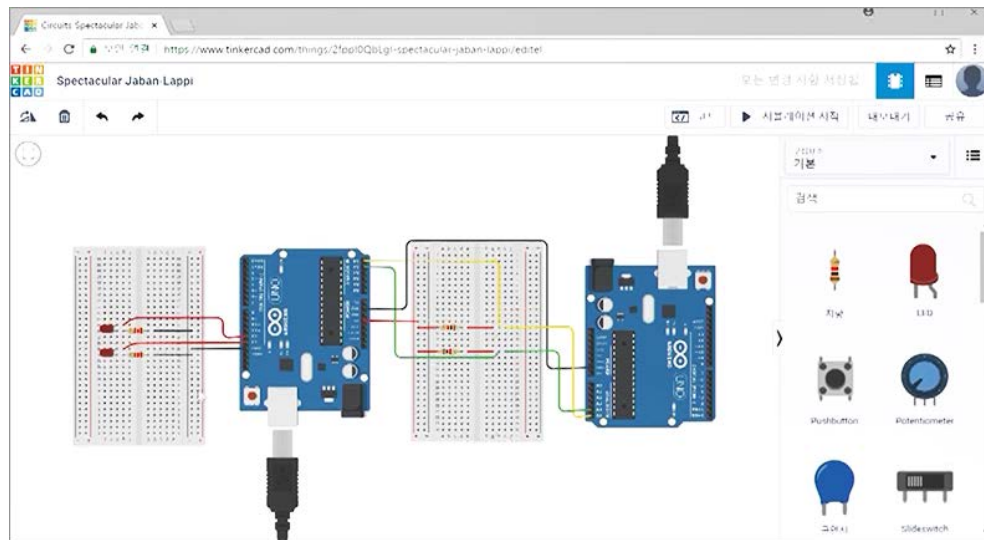
이미지			
준비물	브레드 보드(2개) 아두이노 보드(2개)		
이미지			
준비물	LED(2개)	220Ω 저항(2개) 1kΩ 저항(2개)	점프선 소량

실습

1. 실습 준비물 : 아두이노 보드(2개), 브레드 보드(2개), LED(2개), 1kΩ 저항(2개), 220Ω 저항(2개)
2. 두 개의 아두이노 보드간의 I2C 통신을 실행한다.

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(2)

회로도 구성



실습

1. 마스터 아두이노 A4(SDA) → 슬레이브 아두이노 A4(SDA)
2. 마스터 아두이노 A5(SCL) → 슬레이브 아두이노 A5(SCL)
3. 마스터 13번 핀과 LED_01 (+), 12번 핀과 LED_02 (+) 연결한다.
4. LED의 (-)는 220Ω 저항을 연결 후 아두이노GND에 연결한다.

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(3)

실습 소스(1)



실습

1. 마스터 보드와 슬레이브 보드 간의 회로도를 구성한다.
2. 슬레이브 소스를 업로드 한 후 9v 건전지로 전원을 공급한다.
3. 마스터 소스를 업로드 한다.

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(4)

실습 소스(2)



```
sketch_sep14a | 아두이노 1.8.5
파일 편집 스케치 도구들

sketch_sep14a
#include <Wire.h> // Wire.h 라이브러리 사용

void setup() {
  Wire.begin(8); // 슬레이브 주소값(8)
  Wire.onRequest(requestEvent); // 마스터에서 요청이 오면
                                // requestEvent() 함수 호출
}

void loop() { // 마스터에서 요청이 올 때까지 기다림
  delay(100);
} // 슬레이브 (주소, 바이트) 가져오기
void requestEvent() { // requestEvent() 함수
  Wire.write("01823A456B789A "); // 마스터에게 보낼 데이터
}

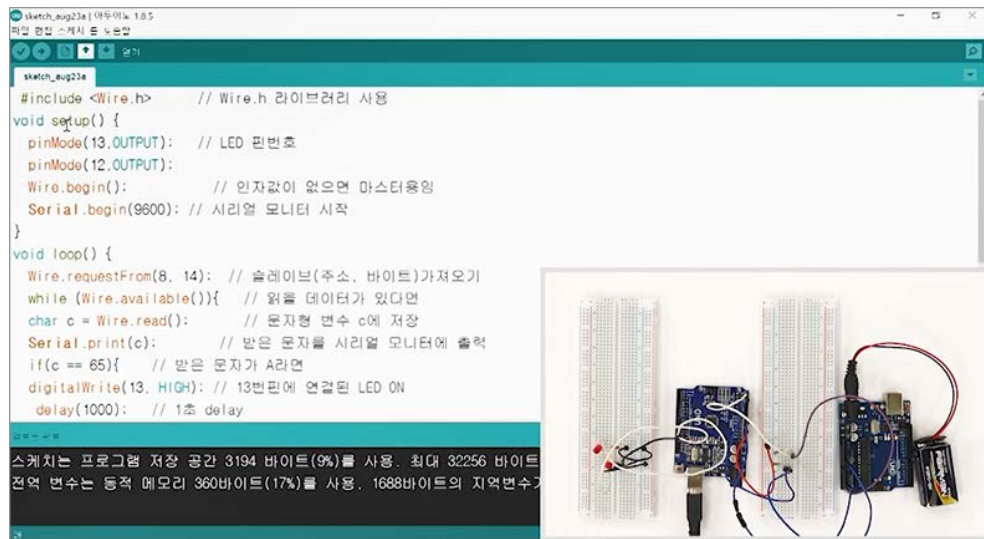
스케치는 프로그램 저장 공간 2156 바이트(6%)를 사용. 최대 32256 바이트.
전역 변수는 동적 메모리 205바이트(10%)를 사용, 1843바이트의 지역변수가 남음. 최대는 2048 바이트.
```

실습

4. 슬레이브 소스
5. <wire.h> 라이브러리 탑재한다.
6. wire.begin(8) : 슬레이브의 주소 값
7. Wire.onRequest(requestEvent) : 마스터에서 요청이 오면
8. Wire.write("01823A456B789A ") : 마스터에게 보낼 데이터

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(5)

실습 소스(3)



실습

9. 마스터 소스

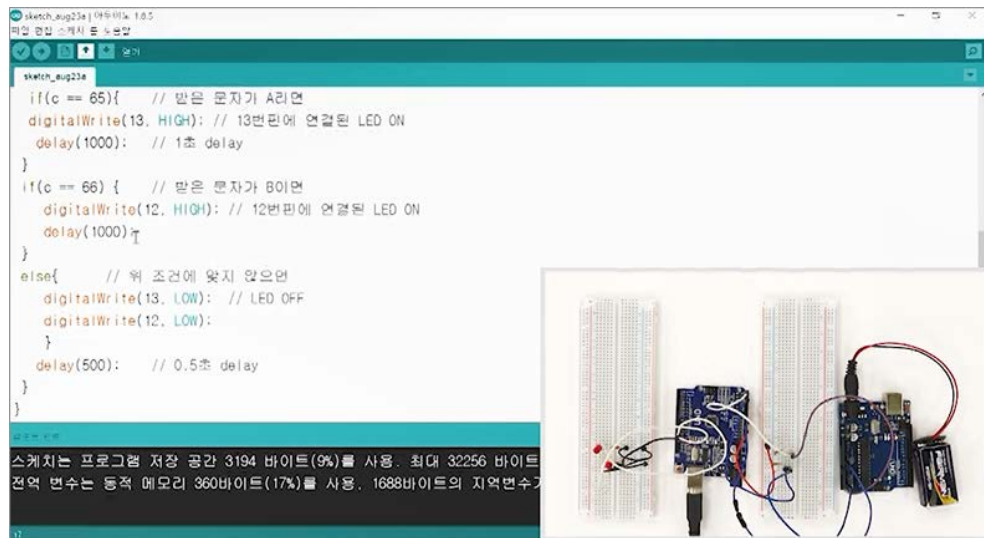
10. `Wire.requestFrom(8, 14)` : 슬레이브(주소, 바이트)를 가져옴

11. `while (Wire.available())` : 읽을 데이터가 있다면 데이터를 읽어옴

12. `if(c == 65){digitalWrite(13, HIGH)}` : 받은 문자가 A라면 13번 핀에 연결된 LED ON

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(6)

실습 소스(4)



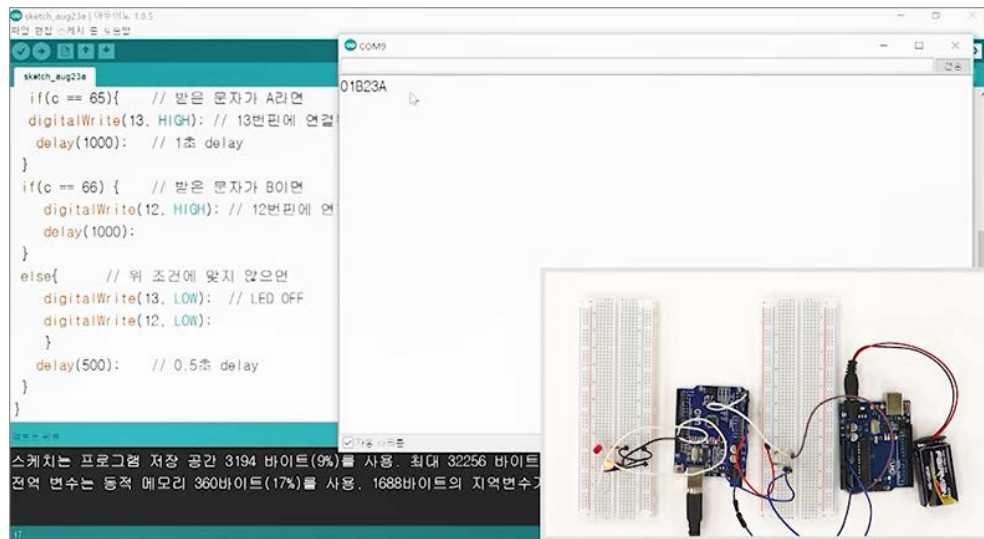
실습

13. if(c == 66) {digitalWrite(12, HIGH) : 받은 문자가 B이면 12번 핀에 연결된 LED ON

14. else : 그렇지 않으면 아래 문장을 수행

2. 마스터/슬레이브 I2C 통신 : LED 제어하기(7)

실습 소스(5)



실습

15.A 값이 들어오면 13번 핀의 LED가 켜진다.

16.B 값이 들어오면 12번 핀의 LED가 켜진다.

실습점검(1)

실습 재미있게 해보셨나요?

'I2C 통신'에 대해 실습한 내용과
자신의 실습 결과물을 비교해 보세요.

1. 아두이노와 I2C 통신을 실행할 수 있나요?

예 ☐ 아니오 ☐

2. 마스터/슬레이브 I2C 통신을 사용해 LED를 제어할 수 있나요?

예 ☐ 아니오 ☐

◎ 교수자 실습 Tip

1. 아두이노 I2C 통신하기

- 회로도 구성을 확인

5V 전원에 1k Ω 풀업 저항 연결 상태 확인

마스터 아두이노 A4(SDA) → 슬레이브 아두이노 A4(SDA)

마스터 아두이노 A5(SCL) → 슬레이브 아두이노 A5(SCL)

- 마스터와 슬레이브 소스를 확인
- 아두이노 통신 포트를 확인

◎ 교수자 실습 Tip

2. 마스터/슬레이브 I2C 통신 : LED 제어하기

- 회로도 구성을 확인

5V 전원에 1k Ω 풀업 저항 연결 상태 확인

마스터 아두이노 A4(SDA) → 슬레이브 아두이노 A4(SDA)

마스터 아두이노 A5(SCL) → 슬레이브 아두이노 A5(SCL)

마스터 13번 핀과 LED_01 (+), 12번 핀과 LED_02 (+) 연결

LED의 (-)는 220 Ω 저항을 연결 후 아두이노 GND에 연결

- 마스터와 슬레이브 소스를 확인
- 아두이노 통신 포트를 확인

다음시간에는...

5주차. 아날로그 입력과 출력

1강. 아날로그 입출력

에 대해 학습해 보겠습니다.

참고문헌

- 킥커카드. <http://www.tinkercad.com>
- (주)메카솔루션. <http://mechasolution.com>