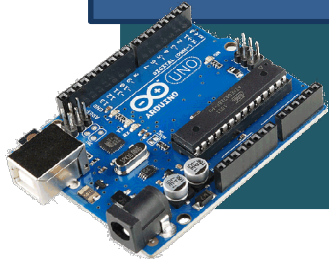


2022년 IoT기반 스마트 솔루션 개발자 양성과정



Firmware [펌웨어]

11-LM35 Sensor

담당 교수 : 유근택
010-5486-5376
rgt3340@naver.com



충북대학교 공동훈련센터

LM35DZ

- Operates from 4 V to 30 V
- 10 mV/°C scale factor
- 0.5°C Ensured Accuracy (at 25°C)
- Non-Linearity Only $\pm 1/4^\circ\text{C}$ Typical
- Rated for Full -55°C to 150°C Range
- Less than 60- μA Current Drain
- Low Self-Heating, 0.08°C in Still Air

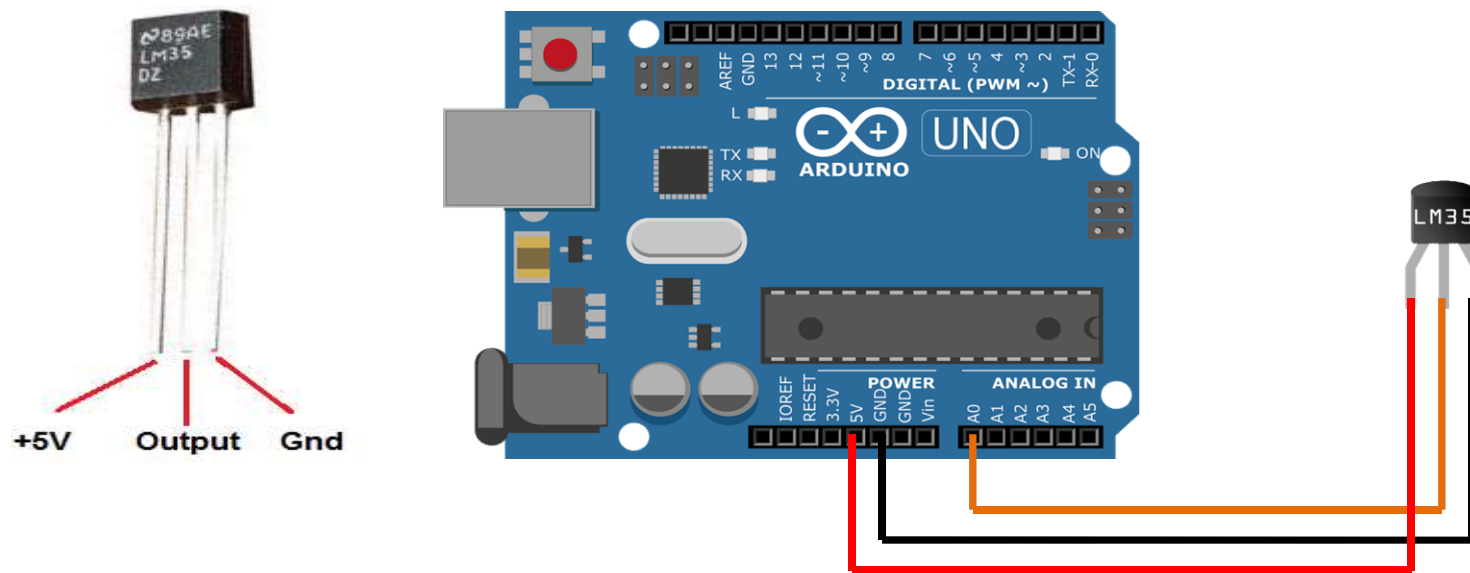


Temperature

- 250 mV output at 25°C
 - $250\text{mV} \times 100 = 25^\circ\text{C}$
- Voltage (V) = ADC * (5.0/1024) //Input 5.0V
- 섭씨온도[°C] = V x 100 //
- 화씨온도[°F] = [°C]×9/5+32, [°C] // °F ≪ °C
- 섭씨온도[°C] = ([°F]−32)×5/9 // °C ≪ °F
- 켈빈온도[K] = °C + 273.15 // K ≪ °C



Circuit



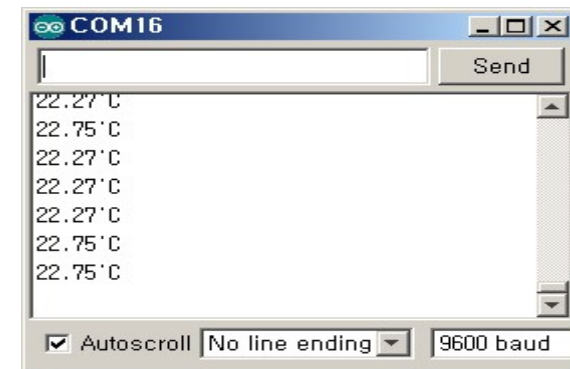
Sketch Program

```
#define LM35 0

void setup( )
{
  analogReference(DEFAULT);  //INTERNAL(2,6V), EXTERNAL
  Serial.begin(9600);
}

void loop( )
{
  int analogInput = analogRead(LM35);
  float temp = (analogInput*5.0*100.0) / 1024.0;

  Serial.print(temp);
  Serial.println(" C");
  delay(1000);
}
```



Sketch for °C & °F

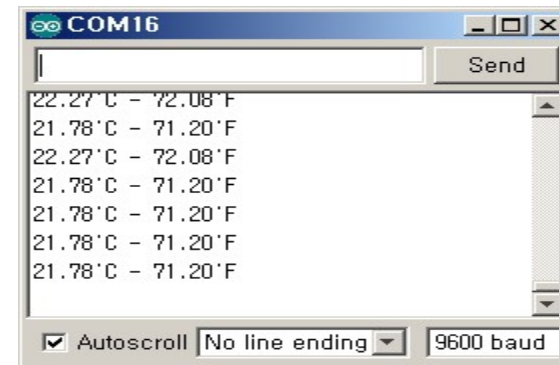
```
#define LM35 0

void setup()
{
  analogReference(DEFAULT); //INTERNAL(2,6V), EXTERNAL
  Serial.begin(9600);
}

void loop()
{
  int analogInput = analogRead(LM35);
  float tempC = (analogInput*5.0*100.0) / 1024.0;
  float tempF = (tempC *9.0 / 5.0) + 32.0;

  Serial.print(tempC);
  Serial.println(" C ");
  Serial.print(tempF);
  Serial.println(" F");

  delay(1000);
}
```



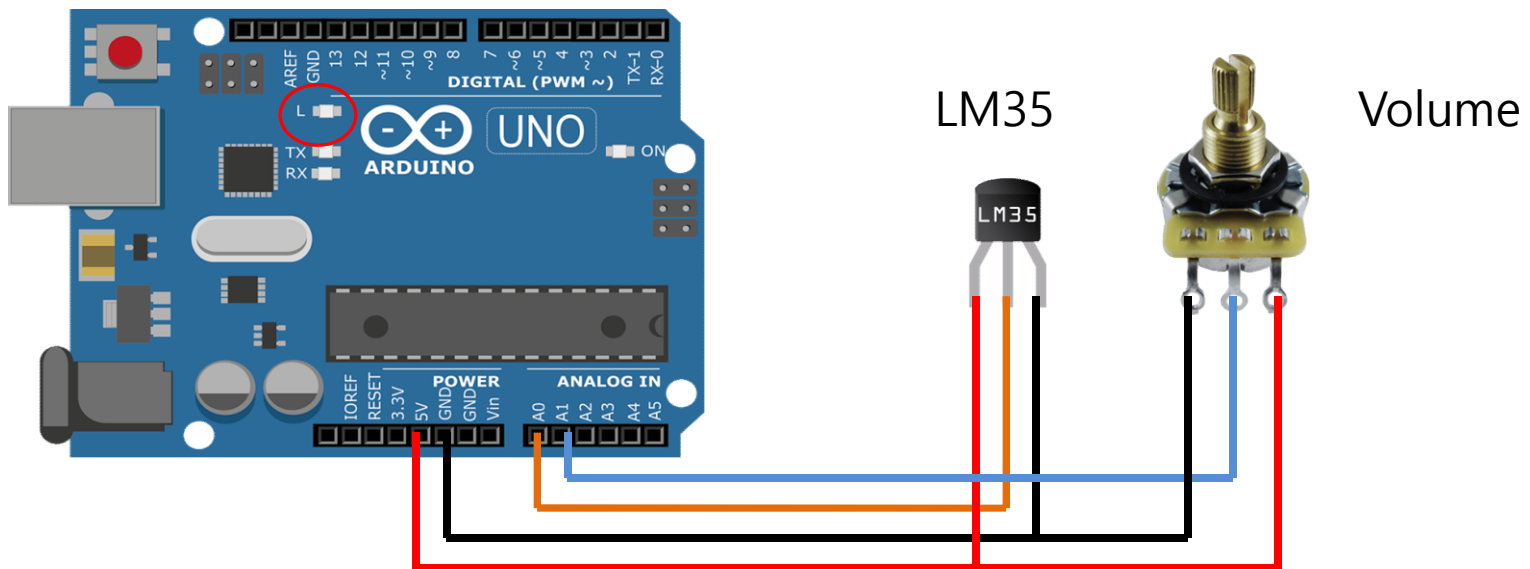
Temperature Controller

- LM35 : A0 (Vcc는 3.3V)
- Volume : A1 (Vcc는 5.0V)
- Output : Digital #13 (L LED와 연동 됨)
- 온도조절 범위 : 10 ~ 30 °C
- 설정온도 : Volume
- 현재온도 : LM35
- 현재온도 값이 설정온도 보다 높으면 LED OFF
- 현재온도 값이 설정온도 보다 작으면 LED ON



Circuit

Output LED



충북대학교 공동훈련센터

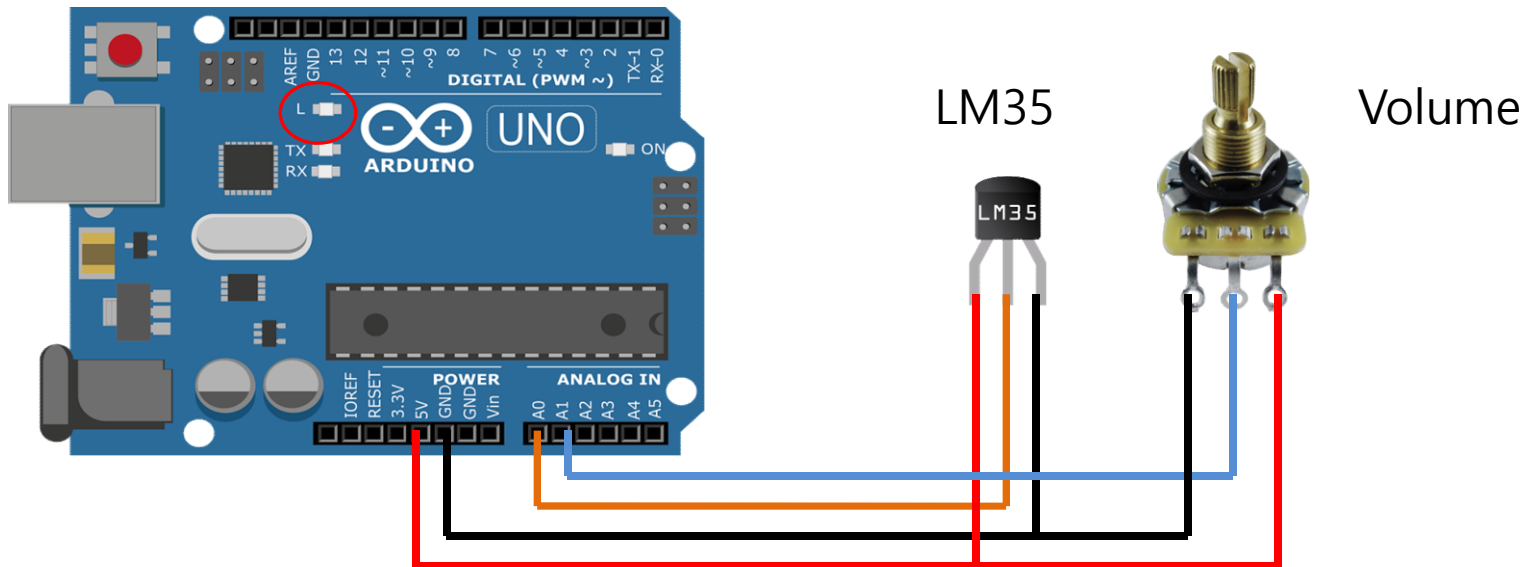
boolean

- A boolean holds one of two values, **true** or **false**.
(Each boolean variable occupies one byte of memory.)
- Example
`boolean running = false;`



Circuit

Output LED

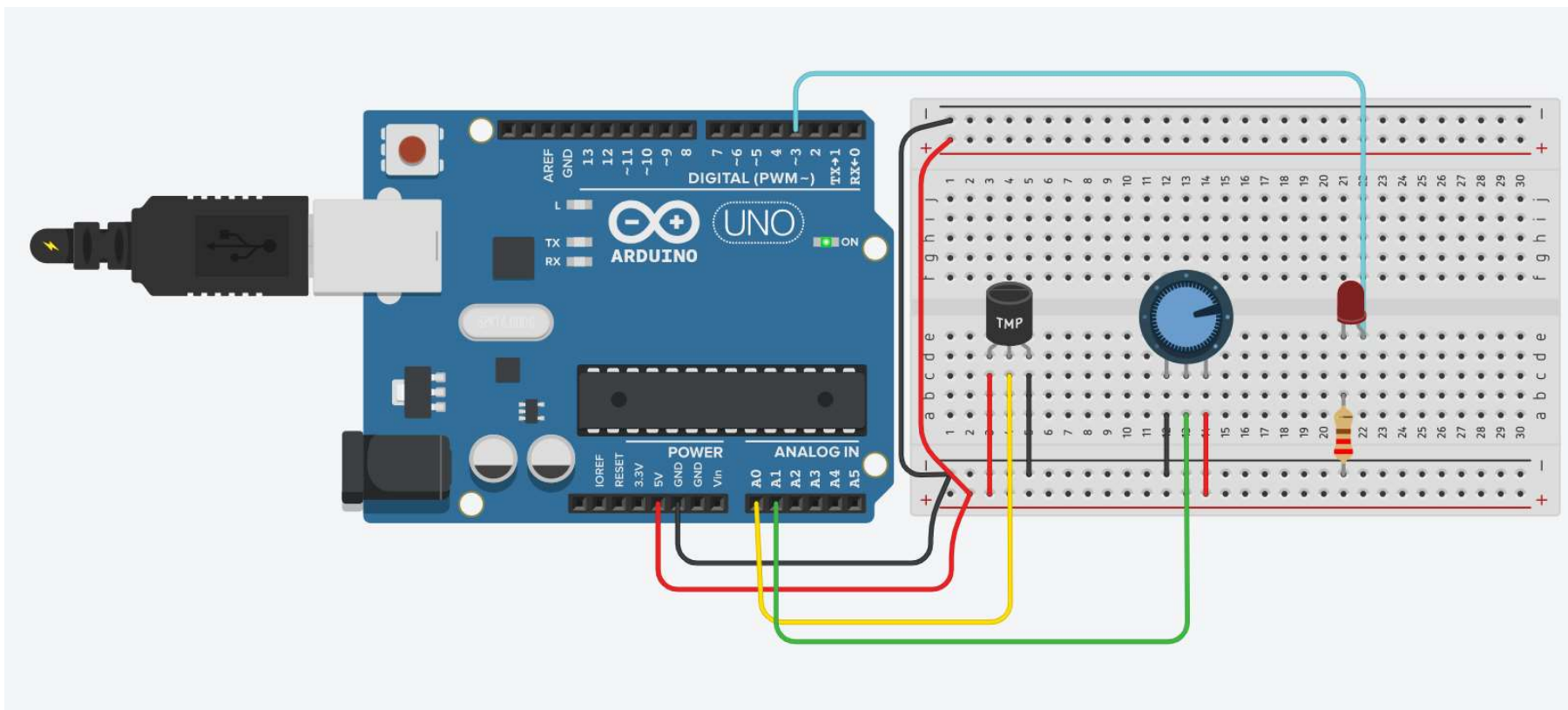


Hysteresis On/Off제어를 구현하자



충북대학교 공동훈련센터

simulation



충북대학교 공동훈련센터

Sketch Program (On/Off제어)

```
#define LM35 A0
#define Volume A1
#define LED 13

float CurrentTemp;
float SetTemp;
boolean Heat;

void setup( ) {
  Serial.begin(9600);
  pinMode(LED,OUTPUT);
}

void SerialOut( ){
  Serial.print(CurrentTemp);
  Serial.print(" - ");
  Serial.print(SetTemp);
  Serial.print(" - ");

  if (Heat) { Serial.println("LED ON"); }
  else { Serial.println("LED OFF"); }
}
```

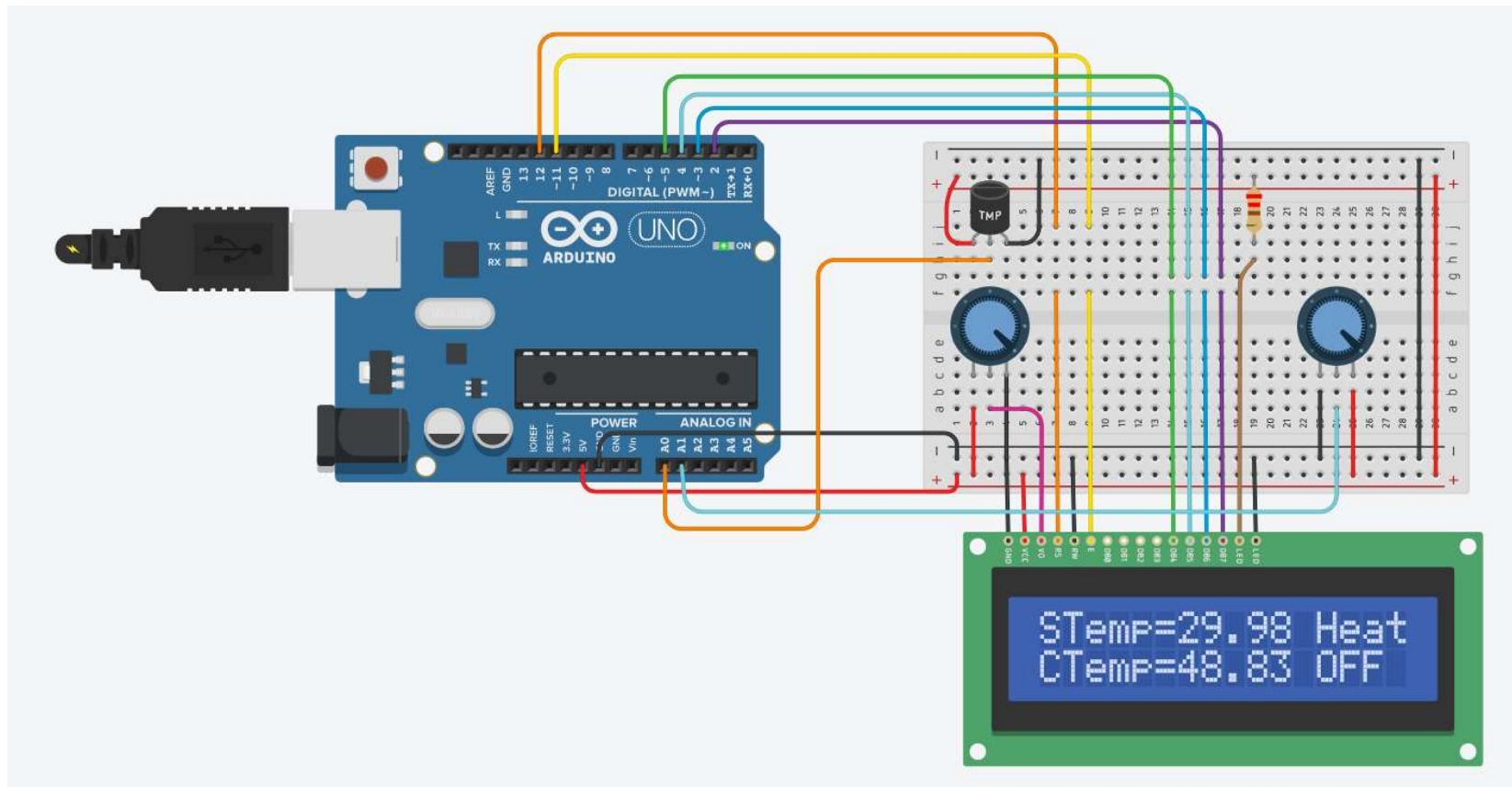
```
void loop( ) {
  CurrentTemp = ( (analogRead(LM35) * 5.0* 100.0 ) / 1024.0 ) ;
  SetTemp = ( ( 20 * analogRead(Volume) ) / 1024.0 ) + 10.0 ;

  if (CurrentTemp > SetTemp ){
    digitalWrite( LED, LOW );
    Heat=false;
  } else {
    digitalWrite( LED, HIGH );
    Heat=true;
  }

  SerialOut( );
  delay(100);
}
```



TM35-LCD : simulation



충북대학교 공동훈련센터

TM35-LCD Code

```
1. // include the library code:
2. #include <LiquidCrystal.h>

3. #define TM35 A0
4. #define VOLUME A1

5. float CurrentTemp = 0.0;
6. float SetTemp = 0.0;

7. // initialize the library with the numbers of the interface pins
8. LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

9. void setup() {
10.   Serial.begin(9600);
11.   // set up the LCD's number of columns and rows:
12.   lcd.begin(16, 2);
13.   // Print a message to the LCD.
14.   lcd.print("Temp. Control SYS");
15.   delay(1000);
16. }

17. void loop() {
18.   SetTemp = ((20 * analogRead(VOLUME))/1024.0) + 10.0;
19.   CurrentTemp = ((analogRead(TM35)* 5.0 *100.0)/1024.0);
20.   String(SetTstr) = String(SetTemp);
21.   String(CurTstr) = String(CurrentTemp);

22.   lcd.setCursor(0, 0);
23.   lcd.print("STemp=" + SetTstr);
24.   Serial.print(SetTemp);

25.   Serial.print(",");
26.   lcd.setCursor(0, 1);
27.   lcd.print("CTemp=" + CurTstr);
28.   Serial.println(CurrentTemp);

29.   lcd.setCursor(12, 0);
30.   lcd.print("Heat");

31.   if(CurrentTemp > SetTemp){
32.     lcd.setCursor(12, 1);
33.     lcd.print("OFF ");
34.   }else{lcd.setCursor(12, 1);
35.     lcd.print("ON ");
36.   }

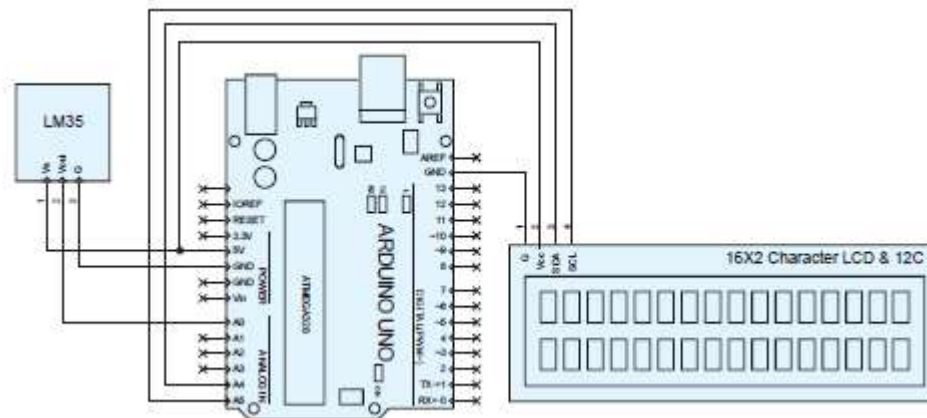
36.   delay(700);
37.   lcd.clear();
38. }
```



Ex-1 : TM35 – I2C LCD

- 실습목표
1. LM35 센서로부터 현재 온도를 아날로그 입력핀으로 측정한다.
 2. 측정된 값을 LCD에 표시해 보자.

- Hardware
1. LM35의 Vs와 G 핀을 Arduino의 5V와 GND에 연결한다.
 2. LM35의 Vout을 아날로그입력핀 A0에 연결한다.
 3. I2C LCD 모듈의 Vcc, GND를 Arduino의 5V, GND에 연결한다.
 4. I2C LCD 모듈의 SDA는 A4에 SCL은 A5에 연결한다.



EX-2: Commands

- analogRead(아날로그 핀번호)
 - ✓ 아날로그 핀에서 아날로그 값을 읽는다. 0~5V사이의 전압을 0~1023 사이의 값으로 표현한다.
- map(변수명, 범위1 최소값, 범위1 최대값, 범위2 최소값, 범위2 최대값)
 - ✓ 변수명의 변수의 범위1의 범위와 범위2의 범위에 매칭시킨다. 즉 변수가 0~100의 범위를 갖고 이를 50~200의 범위로 매칭하려면 'map(변수명, 0, 100, 50, 200)'의 명령어로 매칭시킬 수 있다.
- LiquidCrystal_I2C(I2C 주소, 가로 글자수, 세로 글자수)
 - ✓ LCD 모듈이 연결된 I2C 주소와 LCD의 가로, 세로 글자수를 설정한다.
- lcd.init();
 - ✓ LCD 모듈을 설정한다.
- lcd.clear()
 - ✓ lcd란 이름의 LCD 모듈의 화면의 모든 표시를 지우고 커서를 왼쪽 위로 옮긴다.
- lcd.home()
 - ✓ lcd란 이름의 LCD 모듈의 커서를 왼쪽 위로 옮긴다.
- lcd.setCursor(행, 열)
 - ✓ lcd란 이름의 LCD 모듈의 커서를 원하는 위치로 이동시킨다.
- lcd.print(데이터)
 - ✓ lcd란 이름의 LCD 모듈에 데이터를 출력한다.
- lcd.noBacklight();
 - ✓ lcd란 이름의 LCD 모듈의 백라이트를 소등한다.
- lcd.backlight();
 - ✓ lcd란 이름의 LCD 모듈의 백라이트를 점등한다.




Ex-2 : TM35 - I2C LCD

Sketch 구성

1. LM35 입력을 받기 위한 아날로그 핀을 설정한다.
2. ADC로 읽은 값과 실제 온도와 관계는 연산을 통하여 계산한다.
3. I2C LCD 모듈에 ADC값과 현재 온도를 출력한다.
4. 온도를 출력 할 때 '°' 기호는 LCD 문자 코드표에서 찾아 코드를 이용하여 출력한다.

실습 결과

ADC 값과 온도가 표시된다.



ADC: 250
Temp. is 25 °C

