

Lesson Plan: Digital Thermometer Using Arduino Learner Kit

Overview:

This lesson use Arduino Learner Kit to make a simple digital thermometer using LM35 temperature sensor and LCD display. Both LM35 sensor and LCD are built-in in Arduino Learner Kit. Student will just use few jumper wires to connect the sensor and display with Arduino. After making the connection they will write code and upload it the Arduino.

After completing the lesson students will be able to:

- Design a circuit using analog sensors and LCD display
- Utilize simple sketch to read, calculate and visualize simple sensor data to a LCD display.
- Understand the analog to digital conversion process in Arduino
- Calculate the voltage form the data returned from Arduino analog read.

Important Questions:

- What is the difference between analog and digital?
- Why digital conversion is required?
- What is resolution and why it is important?
- What is a sensor?

Materials needed:

- 1 Arduino Learner Kit with USB cable
- 6 male to male jumper wires
- 1 male to female jumper wire
- Access to a PC

Some Basics:

The LM35 Temperature Sensor:

The LM35 is a precession integrated circuit temperature sensor, whose produces a variable output voltage in it output pin based on the temperature around it. It is a small and cheap IC

which can be used to measure temperature anywhere between -55°C to 150°C . It can easily be interfaced with any Microcontroller that has ADC functionality like Arduino.

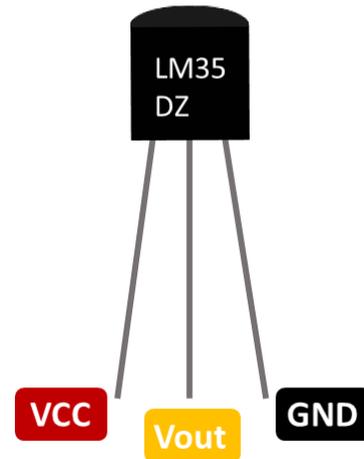
Main advantage of LM35 is that it is linear i.e. $10\text{mv}/^{\circ}\text{C}$ which means for every degree rise in temperature the output of LM35 will rise by 10mv . So if the output of LM35 is 220mv or 0.22V the temperature will be 22°C . So if room temperature is 32°C then the output of LM35 will be 320mv i.e. 0.32V .

Pin-out and Connections:

The LM35 has only three pins, VCC, V_{out} and GND. Here's the typical connections you need to make between the LM35 and the Arduino: But in Arduino Learner Kit VCC and GND connections are already made to keep it simple to use. You just need to connect V_{out} pin to any Arduino analog pin.

Table 1: Connection of LM35

LM35	Arduino
VCC	5V
V_{out}	Any analog pin
GND	GND



The LCD Display:

LCD (Liquid Crystal Display) is very popular and broadly used in electronics projects as they are good for displaying information like sensors data from your project, and also they are very cheap.

The LCD used in the kit has 16 pins and the first one from left to right is the Ground pin. The second pin is the VCC which we connect the 5 volts pin on the Arduino Board. Next is the V_o pin on which we can attach a potentiometer for controlling the contrast of the display.

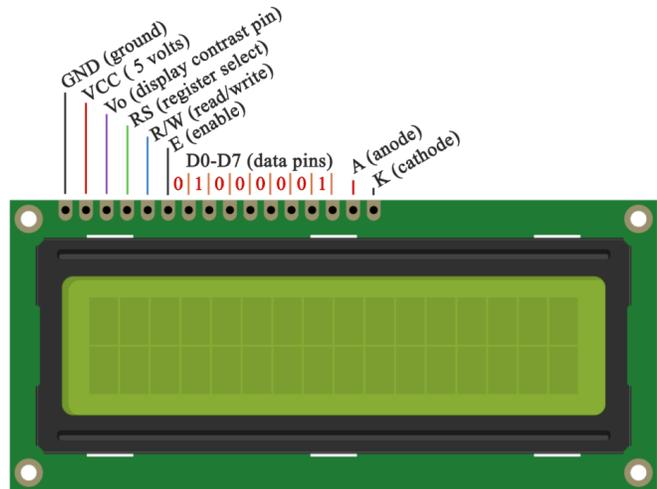
Next, the RS pin or register select pin is used for selecting whether we will send commands or data to the LCD. For example if the RS pin is set on low state or zero volts, then we are sending commands to the LCD like: set the cursor to a specific location, clear the display, turn off the display and so on. And when RS pin is set on High state or 5 volts we are sending data or characters to the LCD.

Next is the E pin which enables the writing to the registers and next 8 data pins from D0 to D7. And the last two pins A and K, or anode and cathode are for the LED back light.

The display can work both 4 bit mode and 8 bit mode. We need to connect D4-D7 to Arduino to work in 4 bit mode. D0-D3 should be unconnected. In the kit GND, VCC, Vo, A, K are already connected to appropriate source. You need to make the following connection only:

Table 2: LCD and Arduino Connection

LCD	Arduino
RS	9
E	8
D4	4
D5	5
D6	6
D7	7



Analog to Digital Conversion:

Microcontrollers are capable of detecting binary (digital) signals: is the button pressed or not? These are digital signals. When a microcontroller is powered from five volts, it understands zero volts (0V) as a binary 0 and a five volts (5V) as a binary 1. The world however is not so simple and likes to use shades of gray. What if the signal is 2.75V? Is that a zero or a one? We often need to measure signals that vary; these are called analog signals. All our surrounding natural signals like temperature, sound, light are analog. If we want to measure any of the parameter we need to use sensor. Usually an electrical sensor produce a voltage output according to the value of sensed parameter. A 5V analog sensor may output 0.01V or 4.99V or anything in between. Luckily, nearly all microcontrollers have a device built into them that allows us to convert these voltages into values that we can use in a program to make a decision.

Not every pin on a microcontroller has the ability to do analog to digital conversions. On the Arduino board, these pins have an 'A' in front of their label (A0 through A5) to indicate these pins can read analog voltages.

ADCs can vary greatly between microcontroller. The ADC on the Arduino is a 10-bit ADC meaning it has the ability to detect 1,024 (2^{10}) discrete analog levels. In the other way we can say the resolution is 10-bit. Conversion accuracy increases dramatically with the increase of resolution. Some microcontrollers have 8-bit ADCs ($2^8 = 256$ discrete levels) and some have 16-bit ADCs ($2^{16} = 65,536$ discrete levels).

Some Math:

The ADC provides a ratiometric value. This means that the ADC assumes 5V is 1023 and anything less than 5V will be a ratio between 5V and 1023 where the resolution is 10-bit. From this ratiometric value actual voltage can be calculated using the following relation:

$$\frac{\text{Resolution of the ADC}}{\text{System Voltage}} = \frac{\text{ADC Reading (ratiometric value)}}{\text{Analog Voltage Measured}}$$

Analog to digital conversions are dependent on the system voltage. Because we predominantly use the 10-bit ADC of the Arduino on a 5V system, we can simplify this equation slightly:

$$\frac{1023}{5} = \frac{\text{ADC Reading}}{\text{Analog Voltage Measured}}$$

Practice Session:

1. If the analog voltage is 2.72V what will the ADC report as a value?

Solution: Suppose the value is X.

Then,

$$\frac{1023}{5} = \frac{X}{2.72}$$

Rearrange things a bit and we get:

$$X = \frac{1023}{5} * 2.72 = 556$$

2. Suppose, you get 420 by reading an analog pin. What is the actual analog voltage? (Try yourself)

Recommended Procedure:

1. Connect V_{out} pin of LM35 to A0 pin of the Arduino using a male to female jumper wire. The V_{out} pin is marked as T_CONN in the kit which is below the right side of the LCD.
2. Connect the LCD pins to the Arduino pins according to the table 2 using male to male jumper wires.
3. From the PC run Arduino IDE and copy the following sketch in a new file.
4. Save it with a name.
5. Connect the kit to PC using a the USB cable and upload the sketch to the kit.

Arduino Sketch:

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

// initialize the library by associating any needed LCD
//interface pin
// with the arduino pin number it is connected to
const int rs = 9, en = 8, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {
  // set up the LCD's number of columns and rows:
  //it a 16 character 2 line lcd
  //following line is required to initilized the lcd
  lcd.begin(16, 2);
}

void loop() {
  //read the temperature sensor
  int sensorValue = analogRead(A0);
  //convert the adc value to analog voltage
  float voltage = (sensorValue * 5.0)/1023.0;
  //convert the voltage into milivolts
  float voltageInMV = voltage * 1000;
  //according to datasheet 10mV for 1 degree Celsius
  float temperature = voltageInMV/10.0;
  //print the result as : Temp: 23.57 C
  lcd.print("Temp: ");
  lcd.print(temperature);
  lcd.print("C");
}
```

Assignment:

Make a project where LCD will display the temperature in Fahrenheit scale. [Note: You just need to change in sketch. Other things will remain same.]