**Dr.Mahalingam College of Engineering and Technology,Pollachi-642003.**

**(An autonomous Institution,affiliated to Anna University Chennai)**

**Department of Information Technology**

**Mini Project Report**

**TEMPERATURE DETECTION USING ARDUINO WITH LM 35**

**Submitted by,**

**NIVEDHA.S(17BIT022)**

**GOKULA PRIYA.E(17BIT056)**

**KOWSALYA.K(17BIT008)**

**Submitted to,**

**Mr.P.Sathiya Murthi**

**Assistant Professor,**

**Department of IT.**

**Dr.Mahalingam College of  Engineering and Technology,**

**Pollachi-642003.**

**(An autonomous Institution,affiliated to Anna University Chennai)**

**BONAFIDE CERTIFICATE**

Certifide that this mini project report, “Temperture detection using arduino” is the

bonafide work of

**NIVEDHA.S(17BIT022)**

**GOKULA PRIYA.E(17BIT056)**

**KOWSALYA.K(17BIT008)**

  who carried out the project work.

Submitted to

**Mr.P.Sathiyamurthi,**

IT department.

**ABSTRACT:**

We searched for some arduino projects on internet there  we found about temperature detection using an lm35.

So, we finally decided to make detecting an temperatures using lm35.

And the arduino is of an special device in which program can be inserted to it for which detection of an temperature will be easier.

And hence of which an LCD display is connected to it for which used to display an desired output which is sensed by an lm35.

Lm35 is an chip like sensor which is used for sensing an temperatures and also which is also used in an different application.

Hence by using an of buzzer it will alert the environment that the temperature is above the limit.

By connecting an LCD pins to an arduino to which program is inserted to it and accordingly output will be displayed.

Hence this is of the main idea of temperature detection using an lm35 with arduino.

**COMPONENTS REQUIRED:**

ARDUINO UNO

Temperature sensor(LM35)

Bread board

Male to male(jumper wires)

Buzzer

LCD

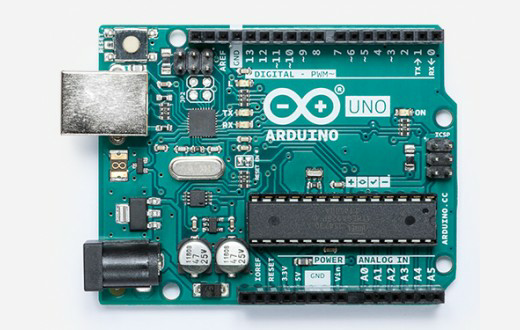
**COST ANALYSIS:**

|  |  |  |
| --- | --- | --- |
|  | Quantity | cost |
| Arduino uno | 1 | 550 |
| Bread board | 1 | 80 |
| Lcd display | 1 | 150 |
| Temperature sensor lm35 | 1 | 50 |
| Buzzer | 1 | 30 |
| Connecting wires | 15 | 70 |
|  |  |  |

TOTAL COST=930

**ARDUINO UNO:**

The UNO is the best board to get started with electronics and coding. If this is your first experience tinkering with the platform, the UNO is the most robust board you can start playing with. The UNO is the most used and documented board of the whole Arduino family.



**Arduino Uno** is a microcontroller board based on the ATmega328P . It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your UNO without worring too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

"Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

The Arduino Uno can be programmed with the ([Arduino Software](https://www.arduino.cc/en/Main/Software) (IDE)). Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board). The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader  that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol .You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using [Arduino ISP](https://www.arduino.cc/en/Main/ArduinoISP) or similar;

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available in the Arduino repository. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated .

The project goal was to create simple, low-cost tools for creating digital projects by non-engineers. The Wiring platform consisted of a [printed circuit board](https://en.wikipedia.org/wiki/Printed_circuit_board) (PCB) with an [ATmega](https://en.wikipedia.org/wiki/ATmega)168 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2003, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they [forked](https://en.wikipedia.org/wiki/Fork_(software_development)) the project and renamed it *Arduino*. Early [arduino](https://en.wikipedia.org/wiki/Arduino) boards used the FTDI USB-to-serial driver chip and an [ATmega](https://en.wikipedia.org/wiki/ATmega)168.

**POWER:**

The Arduino Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector.

The board can operate on an external supply from 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

* Vin. The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
* 5V.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
* 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
* GND. Ground pins.
* IOREF. This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.

**USES OF ARDUINO:**

* **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
* **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
* **Simple, clear programming environment** - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
* **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
* **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the [breadboard version of the module](https://www.arduino.cc/en/Main/Standalone) in order to understand how it works and save money.

**TEMPERATURE SENSOR LM35:**

A **temperature sensor** is a device which is designed specifically to measure the hotness or coldness of an object.**LM35** is a precision IC temperature sensor with its output proportional to the temperature (in °C).With LM35,the temperature can be measured more accurately than with a thermistor. It also possess low self heating and does not cause more than 0.1 °C temperature rise in still air. The operating temperature range is from **-55°C to 150°C**.The LM35’s low output impedance,linea r output, and precise inherent calibration make interfacing to readout or control circuitry especially easy.It has find its applications on power supplies,battery management,appliances,etc.click [**here**](https://wiki.eprolabs.com/index.php?title=File:SEN-0001.pdf) for datasheet.

## LM35 Temperature Sensor

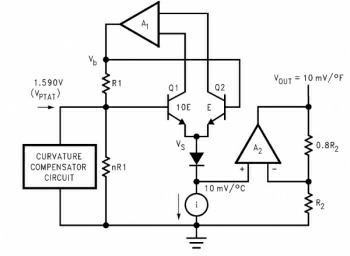


**LM35 Temperature sensor Pinout**

The LM35 is an integrated circuit sensor that can be used to measure temperature with an electrical output proportional to the temperature (in °C).It can measure temperature more accurately than a using a thermistor. The sensor circuitry is sealed and not subject to oxidation.The LM35 generates a higher output voltage than thermocouples and may not require that the output voltage be amplified.The LM35 has an output voltage that is proportional to the Celsius temperature.The scale factor is .01V/°C.

The LM35 does not require any external calibration or trimming and maintains an accuracy of +/-0.4°C at room temperature and +/-0.8°C over a range of 0°C to +100°C.Another important characteristic of the LM35 is that it draws only 60 micro amps from its supply and possesses a low self-heating capability.The LM35 comes in many different packages such as TO-92 plastic transistor-like package,T0-46 metal can transistor-like package,8-lead surface mount SO-8 small outline package.

## Working Principle of LM35



There are two transistors in the center of the drawing. One has ten times the emitter area of the other. This means it has one tenth of the current density, since the same current is going through both transistors. This causes a voltage across the resistor R1 that is proportional to the absolute temperature, and is almost linear across the range.The "almost" part is taken care of by a special circuit that straightens out the slightly curved graph of voltage versus temperature.

The amplifier at the top ensures that the voltage at the base of the left transistor (Q1) is proportional to absolute temperature (PTAT) by comparing the output of the two transistors.

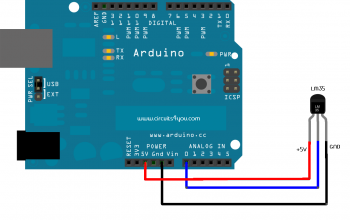
The amplifier at the right converts absolute temperature (measured in Kelvin) into either Fahrenheit or Celsius, depending on the part (LM34 or LM35).The little circle with the "i" in it is a constant current source circuit.

The two resistors are calibrated in the factory to produce a highly accurate temperature sensor.

The integrated circuit has many transistors in it -- two in the middle, some in each amplifier, some in the constant current source, and some in the curvature compensation circuit. All of that is fit into the tiny package with three leads.

## Temperature Sensor to Arduino Uno

The LM35 IC has 3 pins-2 for the power supply and one for the analog output.It is a low voltage IC which uses approximately +5VDC of power.The output pin provides an analog voltage output that is linearly proportional to the Celsius (centigrade) temperature. Pin 2 gives an output of 1 millivolt per 0.1°C (10mV per degree).So to get the degree value in Celsius, all that must be done is to take the voltage output and divide it by 10-this give out the value degrees in Celsius.



The circuit connections are made as follows:

* Pin 1 of the LM35 goes into +5V of the arduino
* Pin 2 of the LM35 goes into analog pin A0 of the arduino
* Pin 3 of the LM35 goes into ground (GND) of the arduino

**LCD DISPLAY:**



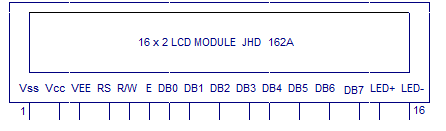
|  |  |  |
| --- | --- | --- |
| **Pin** | **Function** | **Label** |
| 1 | Ground (0v) | GND |
| 2 | Supply Voltage (5v) | Vcc |
| 3 | Contrast Adjustment | Vee |
| 4 | Register Select (0 for Command Register, 1 for Data Register) | Register Select |
| 5 | Read/write select (0 for write, 1 for read) | Read/Write |
| 6 | Data Enable (When pulse is given, send data) | Enable |
| 7 | Data Pin 1 | DB0 |
| 8 | Data Pin 2 | DB1 |
| 9 | Data Pin 3 | DB2 |
| 10 | Data Pin 4 | DB3 |
| 11 | Data Pin 5 | DB4 |
| 12 | Data Pin 6 | DB5 |
| 13 | Data Pin 7 | DB6 |
| 14 | Data Pin 8 | DB7 |
| 15 | Backlight Supply Voltage | LED+ |
| 16 | Backlight Ground | LED- |

LCD modules form a very important part in many arduino based embedded system designs. So the knowledge on interfacing LCD module to arduino is very essential in designing embedded systems. This section of the article is about interfacing an Arduino to 16×2 LCD. JHD162A is the LCD module used here. JHD162A is a 16×2 LCD module based on the HD44780 driver from Hitachi.

The JHD162A has 16 pins and can be operated in 4-bit mode (using only 4 data lines) or 8-bit mode (using all 8 data lines). Here we are using the LCD module in 4-bit mode. First, I will show you how to display a plain text messages on the LCD module using arduino and then  I have designed a useful project using LCD and arduino – a digital thermometer. Before going in to the details of the project, let’s have a look at the JHD162A LCD module.

#### 16×2 LCD Module Pin Out Diagram

The JHD162A lcd module has 16 pins and can be operated in 4-bit mode or 8-bit mode. Here we are using the LCD module in 4-bit mode. Before going in to the details of the project, let’s have a look at the JHD162A LCD module.The schematic of a JHD162A LCD pin diagram is given below.



Pin1(Vss):Ground pin of the LCD module.

Pin2(Vcc): Power to LCD module (+5V supply is given to this pin)

Pin3(VEE):Contrast adjustment pin. This is done by connecting the ends of a 10K potentimeter to +5V and ground and then connecting the slider pin to the VEE pin. The voltage at the VEE pin defines the contrast. The normal setting is between 0.4 and 0.9V.

Pin4(RS):Register select pin.The JHD162A has two registers namely command register and data register. Logic HIGH at RS pin selects data register and logic LOW at RS pin selects command register. If we make the RS pin HIGH and feed an input to the data lines (DB0 to DB7), this input will be treated as data to display on LCD screen. If we make the RS pin LOW and feed an input to the data lines, then this will be treated as a command ( a command to be written to LCD controller – like positioning cursor or clear screen or scroll).

Pin5(R/W): Read/Write modes. This pin is used for selecting between read and write modes. Logic HIGH at this pin activates read mode and logic LOW at this pin activates write mode.

Pin6(E): This pin is meant for enabling the LCD module. A HIGH to LOW signal at this pin will enable the module.

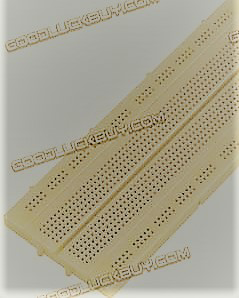
Pin7(DB0) to Pin14(DB7):  These are data pins. The commands and data are fed to the LCD module though these pins.

Pin15(LED+): Anode of the back light LED. When operated on 5V, a 560 ohm resistor should be connected in series to this pin. In arduino based projects the back light LED can be powered from the 3.3V source on the arduino board.

Pin16(LED-): Cathode of the back light LED.

**BREAD BOARD:**

A breadboard is a construction base for prototyping of electronics. Originally the word referred to a literal bread board, a polished piece of wood used for slicing bread[citation needed].[1] In the 1970s the solderless breadboard (a.k.a. plugboard, a terminal array board) became available and nowadays the term "breadboard" is commonly used to refer to these.



Because the solderless breadboard does not require soldering, it is reusable. This makes it easy to use for creating temporary prototypes and experimenting with circuit design. For this reason, solderless breadboards are also popular with students and in technological education. Older breadboard types did not have this property. A stripboard (Veroboard) and similar prototyping printed circuit boards, which are used to build semi-permanent soldered prototypes or one-offs, cannot easily be reused. A variety of electronic systems may be prototyped by using breadboards, from small analog and digital circuits to complete central processing units (CPUs).modern solderless breadboard socket consists of a perforated block of plastic with numerous tin plated phosphor bronze or nickel silver alloy spring clips under the perforations. The clips are often called tie points or contact points. The number of tie points is often given in the specification of the breadboard.

The spacing between the clips (lead pitch) is typically 0.1 inches (2.54 mm). Integrated circuits (ICs) in dual in-line packages (DIPs) can be inserted to straddle the centerline of the block. Interconnecting wires and the leads of discrete components (such as capacitors, resistors, and inductors) can be inserted into the remaining free holes to complete the circuit. Where ICs are not used, discrete components and connecting wires may use any of the holes. Typically the spring clips are rated for 1 ampere at 5 volts and 0.333 amperes at 15 volts (5 watts). The edge of the board has male and female dovetail notches so boards can be clipped together to form a large breadboard.

**JUMP WIRES:**

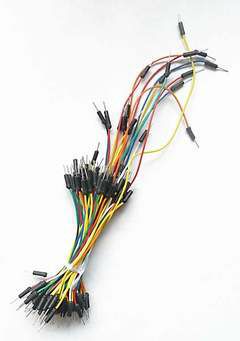
Stranded 22AWG jump wires with solid tip

A **jump wire** (also known as jumper wire, or jumper) is an [electrical wire](https://en.wikipedia.org/wiki/Electrical_wire), or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a [breadboard](https://en.wikipedia.org/wiki/Breadboard) or other prototype or test circuit, internally or with other equipment or components, without soldering.[[1]](https://en.wikipedia.org/wiki/Jump_wire#cite_note-1)

Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the [header connector](https://en.wikipedia.org/wiki/Pin_header#Header_connector) of a circuit board, or a piece of test equipment

There are different types of jumper wires. Some have the same type of [electrical connector](https://en.wikipedia.org/wiki/Electrical_connector) at both ends, while others have different connectors. Some common connectors are:

* Solid tips – are used to connect on/with a breadboard or female header connector. The arrangement of the elements and ease of insertion on a breadboard allows increasing the mounting density of both components and jump wires without fear of short-circuits. The jump wires vary in size and colour to distinguish the different working signals.
* [Crocodile clips](https://en.wikipedia.org/wiki/Crocodile_clip) – are used, among other applications, to temporarily bridge sensors, buttons and other elements of prototypes with components or equipment that have arbitrary connectors, wires, [screw terminals](https://en.wikipedia.org/wiki/Screw_terminal), etc.
* [Banana connectors](https://en.wikipedia.org/wiki/Banana_connector) – are commonly used on test equipment for DC and low-frequency AC signals.
* [Registered jack](https://en.wikipedia.org/wiki/Registered_jack) (RJnn) – are commonly used in telephone (RJ11) and computer networking (RJ45).
* [RCA connectors](https://en.wikipedia.org/wiki/RCA_connector) – are often used for audio, low-resolution composite video signals, or other low-frequency applications requiring a [shielded cable](https://en.wikipedia.org/wiki/Shielded_cable).
* [RF connectors](https://en.wikipedia.org/wiki/RF_connector) – are used to carry [radio frequency](https://en.wikipedia.org/wiki/Radio_frequency) signals between circuits, test equipment, and antennas.



**BUZZER:**



A **buzzer** or **beeper** is an [audio](https://en.wikipedia.org/wiki/Sound)signallingdevice,which may be [mechanical](https://en.wikipedia.org/wiki/Machine), [electromechanical](https://en.wikipedia.org/wiki/Electromechanics), or [piezoelectric](https://en.wikipedia.org/wiki/Piezoelectricity) (*piezo* for short). Typical uses of buzzers and beepers include [alarm devices](https://en.wikipedia.org/wiki/Alarm_devices), [timers](https://en.wikipedia.org/wiki/Timer), and confirmation of user input such as a mouse click or keystroke.

t most commonly consists of a number of switches or sensors connected to a control unit that determines if and which button was pushed or a preset time has lapsed, and usually illuminates a light on the appropriate button or control panel, and sounds a warning in the form of a continuous or intermittent buzzing or beeping sound.

Initially this device was based on an electromechanical system which was identical to an electric bell without the metal gong (which makes the ringing noise).

Often these units were anchored to a wall or ceiling and used the ceiling or wall as a sounding board. Another implementation with some AC-connected devices was to implement a circuit to make the AC current into a noise loud enough to drive a loudspeaker and hook this circuit up to a cheap 8-ohm speaker.

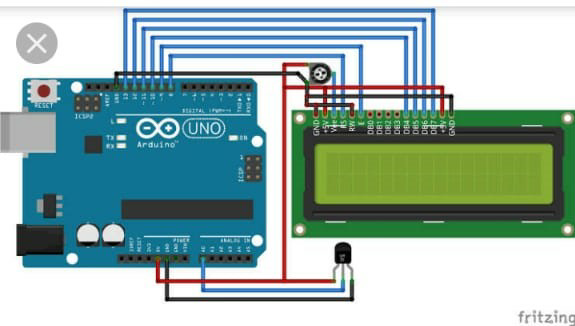
Nowadays, it is more popular to use a ceramic-based piezoelectric sounder like a Sonalert which makes a high-pitched tone. Usually these were hooked up to "driver" circuits which varied the pitch of the sound or pulsed the sound on and off.

In game shows it is also known as a "lockout system," because when one person signals ("buzzes in"), all others are locked out from signalling. Several game shows have large buzzer buttons which are identified as "plungers".

The word "buzzer" comes from the rasping noise that buzzers made when they were electromechanical devices, operated from stepped-down AC line voltage at 50 or 60 cycles. Other sounds commonly used to indicate that a button has been pressed are a ring or a beep.

Some systems, such as the one used on Jeopardy!, make no noise at all, instead using light

**CIRCUIT DIAGRAM:**



**WORKING:**

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius(Centigrade) temperature.

The LM35 does not require any external calibration or trimming to provide typical accuracies of ±1⁄4°C at room temperature and ±3⁄4°C over a full −55 to +150°C temperature range.

Temperature Sensor data will be displayed on the LCD so we need to know the connection of it to the Arduino. How to interface 16×2 LCD with Arduino.

As accordingly an connection is given according to an pin diagram.

And next the code is insertong into the arduino uno software by using an arduino ide aoftware. Hence it helps to do any activity by inserting an program to it.

When power supply is given the arduino board ia getted power and lcd os also gets on. And 5v power or 3.3v connection is given so that it will sence an temperature and calculation is done for converting into an Celsius and farenheat also.

And the final value will be displayed into an lcd device also.hence if the temperture exits an limit of 40 • c then the alert sound is produced by an buzzer which is has an 2 connections like positive and negative.

The positive connection is given to an digital pin of an arduino and the negative pin is connected to an ground pin.

Hence according the working of an temperture sensor is worked when it is conmected to an lcd display and also an connection with an buzzer.

**PROGRAM:**

INTERFACING AND PROGRAMMING OF TEMPERATURE SENSOR WITH ARDUINO

#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins

LiquidCrystal lcd(2 ,3 ,4 ,5 ,6 ,7);

//declare variables

float tempC;

float tempF;

int tempPin = A1; int buzzpin = 8;

void setup(){

pinMode(buzzpin,OUTPUT);

Serial.println(9600);

lcd.begin(16, 2); // set up the LCD's number of columns and rows:

lcd.print("Temp1=");

lcd.setCursor(0, 1);

lcd.print("Temp2=");

}

void loop(){

tempC = analogRead(tempPin); //read the value from the sensor

Serial.println(tempC);

tempC = (5.0 \* tempC \* 100.0)/1024.0; //convert the analog data to temperature

tempC = tempC-2;

tempF = ((tempC\*9)/5) + 32; //convert celcius to farenheit

lcd.setCursor(6, 0);

lcd.print(tempC); // print result to lcd display

lcd.print("'C");

lcd.setCursor(6, 1);

Serial. Println(tempF);

lcd.print(tempF); // print temp in farenheit

lcd.print("'F");

// to buzz buzzer

if (tempC>40)

digitalWrite(8,1);

else

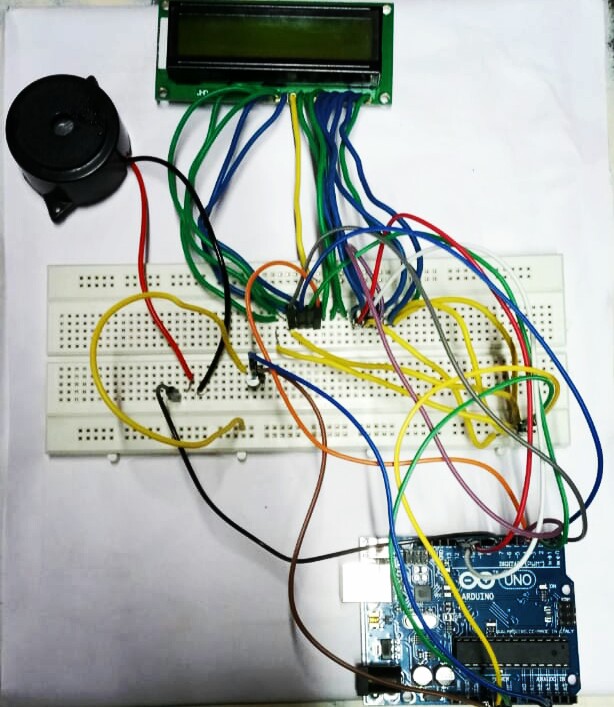
digitalWrite(8,0);

// sleep...

delay(1000);

}

**ORIGINAL PICTURE:**

****

**APPLICATIONS:**

Temperature sensor, with the help of some advanced programming, can display a current temperature in the environment and by using of an LCD display it. The **cube** has 64 **LEDs**. Which is used to check with the certain limit by an LM35 sensor.

Although our plan is for a visually detecting an temperatures, our implementation can easily be adapted for more practical **applications** such as detection of temperature,and other application. Goals and Function: Our project goal is to use lm35 **sensor to display the output in an LCD devices.**