VIT®
UNIVERSITY
(Estd. u/s 3 of UGC Act 1956)

Temperature Based Fan Speed Control and Monitoring

PROJECT REPORT

Submitted for the course: Measurement and Instrumentation
(EEE 2004)

Submitted By :-

Danish Safdar	**17BEE0067**
Ayushman Ranu	**17BEE0214**
Ritvik Arya	**17BEE0231**

SLOT: F2

Name of the faculty: **Vijaya Priya P**

CERTIFICATE

This is to certify that the project work entitled "Temperature Based Fan Speed Control and Monitoring" that is being submitted by Danish, Ritvik and Ayushman for Measurement and Instrumentation (EEE 2004) is a record of bonafide work done under my supervision. The contents of this Project work, in full or in parts, have neither been taken from any source nor have been submitted for any other NON CAL course.

Place: Vellore

Date: 22/4/2017

Signature of the faculty	**Signature of the students**
	Danish Safdar
Vijaya Priya P	**Ritvik Arya**
	Ayushman Ranu

ACKNOWLEDGEMENTS

We would like to express gratitude to our professor **Prof. Vijaya Priya P** for her continuous support throughout the project. We would also like to thank our classmates and seniors for their cooperation in our endeavours and aiding us in successful completion of our project.

We would also like to thank **Dr. Arulmozhivarman, Dean SELECT,** and the staff and management of **VIT University.**

Danish Safdar (17BEE0067)

Ritvik Arya (17BEE0231)

Ayushman Ranu (17BEE0214)

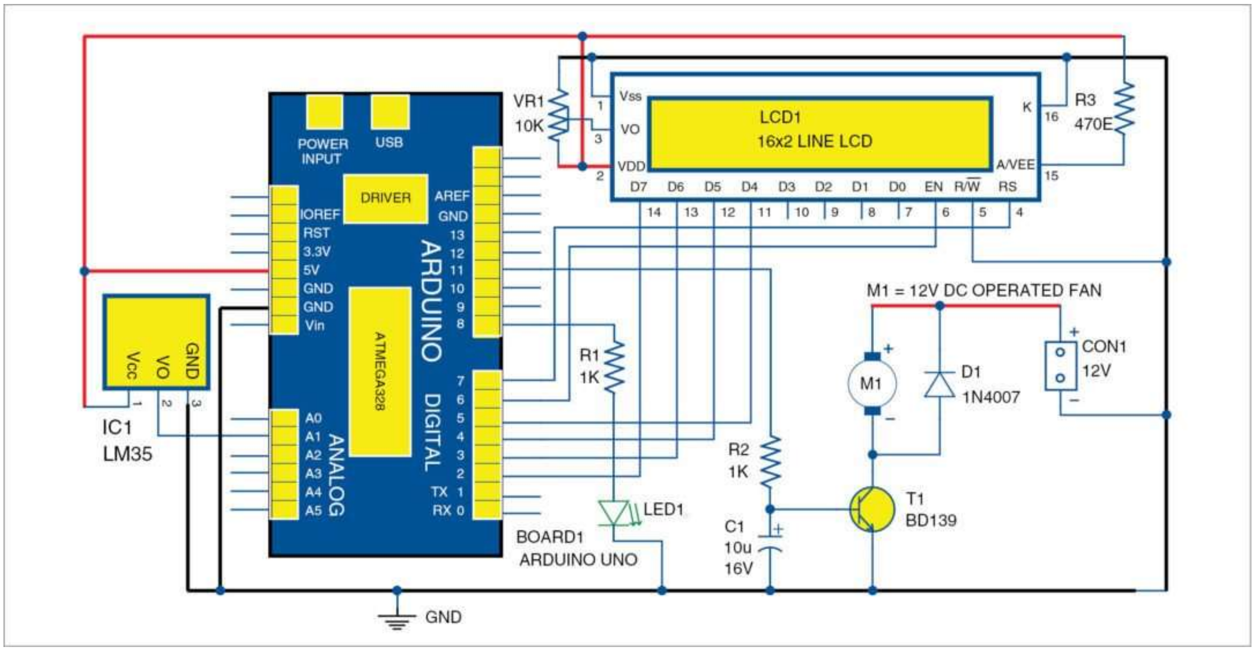
ABSTRACT

The purpose of our project was to develop a This a standalone automatic fan speed controller that controls the speed of an electric fan according to the requirement. Use of embedded technology makes this closed-loop feedback-control system efficient and reliable. The microcontroller (MCU) ATMega8/168/328 allows dynamic and faster control and the LCD makes the system user-friendly. Sensed temperature and fan speed levels are simultaneously displayed on the LCD panel

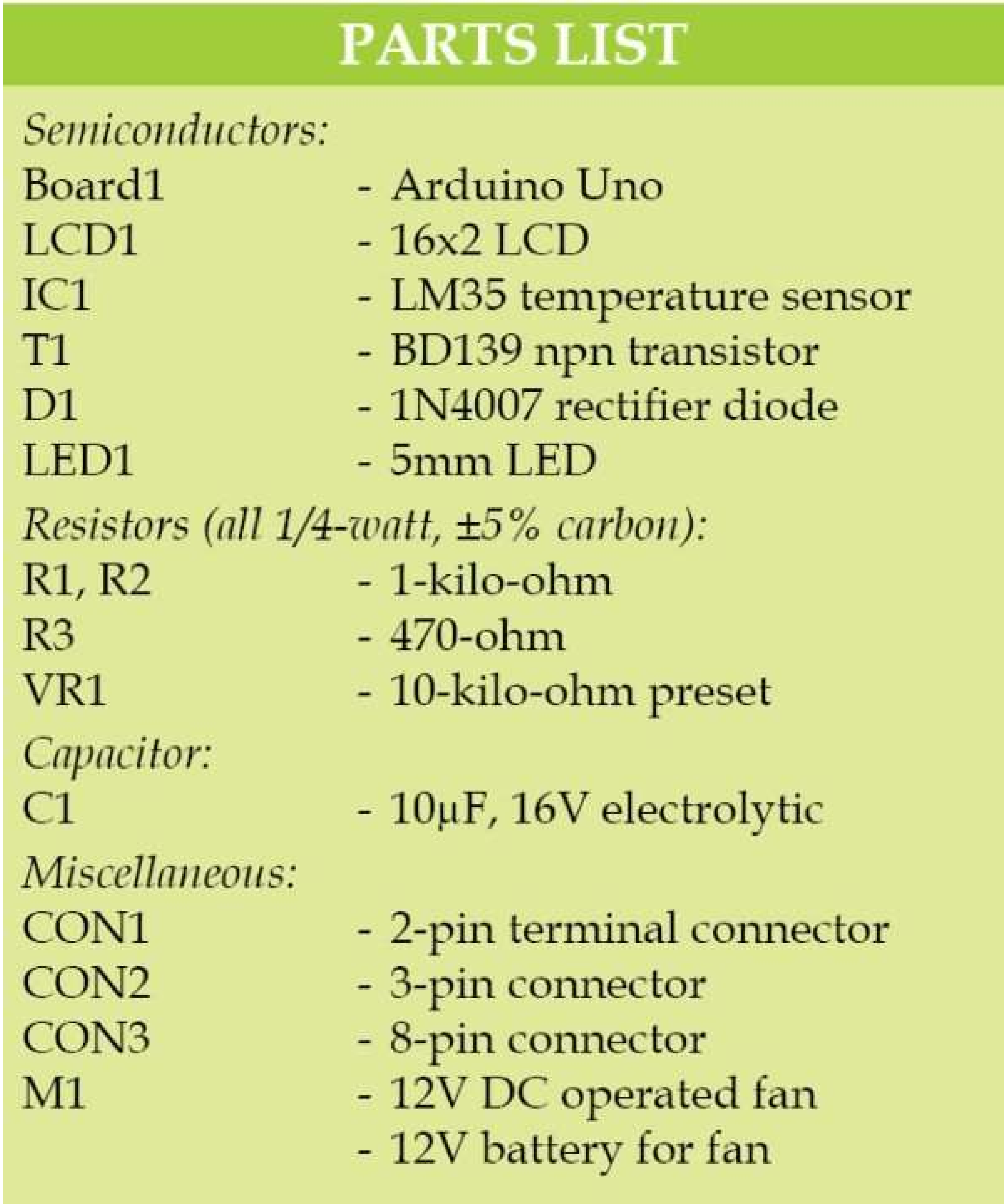
Introduction

A computer fan is any fan inside, or attached to, a computer case used for active cooling, and may refer to fans that draw cooler air into the case from the outside, expel warm air from inside, or move air across a heat sink to cool a particular component. Generally these are found in axial and sometimes centrifugal forms. The former is sometimes called a "electric" fan

Circuit Diagram



Components



PARTS LIST

Semiconductors:	
Board1	- Arduino Uno
LCD1	- 16x2 LCD
IC1	- LM35 temperature sensor
T1	- BD139 npn transistor
D1	- 1N4007 rectifier diode
LED1	- 5mm LED
Resistors (all 1/4-watt, ±5% carbon):	
R1, R2	- 1-kilo-ohm
R3	- 470-ohm
VR1	- 10-kilo-ohm preset
Capacitor:	
C1	- 10µF, 16V electrolytic
Miscellaneous:	
CON1	- 2-pin terminal connector
CON2	- 3-pin connector
CON3	- 8-pin connector
M1	- 12V DC operated fan
	- 12V battery for fan

Component Description

Temperature Sensor LM35

The LM35 series are precision integrated-circuit temperature devices with an output voltage linearly-proportional to the Centigrade temperature

Arduino Uno

Arduino is an open-source electronics platform based on easy-to-use hardware and software.

Liquid Crystal Display

The term LCD stands for liquid crystal display. It is a dot matrix liquid crystal display that shows alphanumeric letters and also symbols. In the proposed system, a 16X2 LCD digital display has been used to show the temperature of the room.

12V DC fan

It is a 12V fan that is used to blow air into the heated part of the system.

BD139

BD139 is a n-p-n silicon transistor

Arduino CODE

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(7,6,5,4,3,2);
int tempPin = A1;   // the output pin of LM35
int fan = 11;       // the pin where fan is
int led = 8;        // led pin
int tempMin = 30;   // the temperature to start the fan 0%
int tempMax = 60;   // the maximum temperature when fan is at 100%
int fanSpeed;
int fanLCD;
void setup() {
  pinMode(fan, OUTPUT);
  pinMode(led, OUTPUT);
  pinMode(tempPin, INPUT);
  lcd.begin(16,2);
  Serial.begin(9600);
}
void loop()
{
   temp = readTemp();     // get the temperature
   Serial.print( temp );
   if(temp < tempMin)     // if temp is lower than minimum temp
   {
       fanSpeed = 0;      // fan is not spinning
       analogWrite(fan, fanSpeed);
       fanLCD=0;
       digitalWrite(fan, LOW);
   }
  if((temp >= tempMin) && (temp <= tempMax)) // if temperature is higher than minimum temp
  {
     fanSpeed = temp;//map(temp, tempMin, tempMax, 0, 100); // the actual speed of
fan//map(temp, tempMin, tempMax, 32, 255);
     fanSpeed=1.5*fanSpeed;
     fanLCD = map(temp, tempMin, tempMax, 0, 100);  // speed of fan to display on LCD100
     analogWrite(fan, fanSpeed);  // spin the fan at the fanSpeed speed
  }
  if(temp > tempMax) // if temp is higher than tempMax
   {
   digitalWrite(led, HIGH);  // turn on led
   }
  else              // else turn of led
   {
   digitalWrite(led, LOW);
   }
  lcd.print("TEMP: ");
  lcd.print(temp);      // display the temperature
  lcd.print("C ");
  lcd.setCursor(0,1);   // move cursor to next line
  lcd.print("FANS: ");
  lcd.print(fanLCD);    // display the fan speed
  lcd.print("%");
  delay(200);
  lcd.clear();
}
int readTemp() {  // get the temperature and convert it to celsius
 temp = analogRead(tempPin);
 return temp * 0.48828125; }
```

Working

The proposed system is a practical temperature controller used to control the temperature of any device based on its requirement for different industrial applications. It also shows the temperature on an LCD display and the temperature range from 30°C to 60°C. It is built around Arduino Uno board (Board1), 16×2 LCD (LCD1), temperature sensor LM35(IC1) and a few other components.

Arduino is at the heart of the circuit as it controls all functions. LM35 is a precision integrated circuit whose output voltage is linearly proportional to Celsius (Centigrade) temperature. It is rated to operate over a -55°C to 150°C temperature range. It has +10.0mV/Celsius linear-scale factor.

Temperature sensor LM35 senses the temperature and converts it into an electrical (analogue) signal, which is applied to the MCU through an analogue-to-digital converter (ADC). The analogue signal is converted into digital format by the ADC. Sensed values of the temperature and speed of the fan are displayed on the LCD. Temperature and monitoring using Arduino The MCU on Arduino drives the motor driver to control fan speed.

Fan speed control technique. A low-frequency pulse-width modulation (PWM) signal, usually in the range of about 30Hz, whose duty cycle is varied to adjust the fan's speed is used. An inexpensive, single, small pass transistor can be used here. It is efficient because the pass transistor is used as a switch**.**

Applications

CPU Fan

Graphic Card Fan

Chipset Fan

Hard Drive cooling

Result

From this project we were able to develop a Temperature sensing and monitoring device that would not only cool the device but also save electricity while doing so.