Project 1

MCHE 416 – Mechatronics System Design

by

Mohammad Omar Shehab, 201801047

Submitted to the
Department of Mechanical Engineering
Faculty of Engineering

Instructor

Dr. Amine Abou Moughlbay

Spring 2019-2020

Table of Contents

Mini Project 1 – Controlling Traffic 5
1.1 Introduction 5
1.2 Inputs and Outputs 5
1.2.1 Inputs 5
1.2.2 Outputs 5
1.3 Pseudocode 5
1.4 Flowchart 7
1.5 Arduino Circuit 8
1.6 Arduino Code 9
1.7 Simulation Results 12
Mini Project 2 – Temperature Alarm 13
2.1 Introduction 13
2.2 Inputs and Outputs 13
2.2.1 Inputs 13
2.2.2 Outputs 13
2.3 Pseudocode 13
2.4 Flowchart 14
2.5 Arduino Circuit 14
2.6 Arduino Code 15
2.7 Simulation Results 18
Mini Project 3 – Single-Cell Battery Tester 19
3.1 Introduction 19
3.2 Inputs and Outputs 19
3.2.1 Inputs 19

3.2.2 Outputs 19
3.3 Pseudocode 20
3.4 Flowchart 21
3.5 Arduino Circuit 22
3.6 Arduino Code 22
3.7 Simulation Results 24
Appendix 27

List of Figures

Figure 1 Flowchart of Mini Project 1 7

Figure 2 Schematic Representation of The Circuit of Mini Project 1 8

Figure 3 Flowchart of Mini Project 2 14

Figure 4 Schematic Representation of The Circuit of Mini Project 2 15

Figure 5 Beeping Cycle for a Temperature Range of [26°C, 32°C[........ 17

Figure 6 Beeping Cycle for a Temperature Greater Than or Equal to 32°C 18

Figure 7 Flowchart of Mini Project 3 21

Figure 8 Schematic Representation of The Circuit of Mini Project 3 22

Figure 9 Supplying a Voltage of 1.6V 25

Figure 10 Supplying a Voltage of 1.4V 25

Figure 11 Supplying a Voltage of 1V 26

Mini Project 1 – Controlling Traffic

1.1 Introduction

The aim is to control the flow of traffic on a one-way bridge. Initially, the traffic should flow from west to east. Then when a car approaches from the east, the flow will reverse to allow movement from east to west. In addition, a car coming from the west will return the flow to its initial direction. Finally, enough time is given to clear the bridge before the flow of traffic switches.

1.2 Inputs and Outputs

The variables in this system are the cars and the traffic lights.

1.2.1 Inputs

A car approaching each side of the bridge is simulated by the press of a momentary button. Hence, there are 2 push buttons, one on each side of the bridge.

Since momentary buttons are digital inputs. Therefore, the west sided button, named as "West_button" is connected to the Arduino's digital pin 2, and the east sided button, named as "East_button" is connected to digital pin 10.

1.2.2 Outputs

In order to control the direction of flow of traffic, the traffic lights on both sides of the bridge should be controlled. Moreover, since every traffic signal has 3 different lights, green, yellow, and red, there is a total of 6 lights represented by 6 LEDs to be controlled.

Each LED, a digital output device, is connected to a digital pin of the Arduino whereby the west sided LEDs, green, yellow, and red, are connected to pins 5, 4, and 3 respectively. Whereas, the east sided LEDs, green, yellow, and red, are connected to pins 11, 12, and 13 respectively.

1.3 Pseudocode

A pseudocode is a simplified version of a code; it is formed using English statements. In addition, it has special formatting and indentations so that the logical order can be understood effectively.

This project's pseudocode is given below.

```
Start Program
     Turn on the west sided green light and the east sided red
light.
     The traffic is flowing from west to east.
Loop Forever
     IF a car approaches the bridge from the east THEN
          Turn off the west sided green light.
          Turn on the west sided yellow light after 1 second.
          After 4 seconds, turn off the west sided yellow light,
but turn on the west sided red light.
          Three seconds later, turn on east sided yellow light,
but turn off the east sided red light.
          After 3 seconds, turn on the east sided green light,
but turn off the east sided yellow light.
          Hence, the traffic now flows from east to west until
a car arrives from the other direction.

     IF a car approaches the bridge from the west THEN
          Turn off the east sided green light.
          Turn on the east sided yellow light after 1 second.
          After 4 seconds, turn off the east sided yellow light,
but turn on the east sided red light.
          Three seconds later, turn on west sided yellow light,
but turn off the west sided red light.
          After 3 seconds, turn on the west sided green light,
but turn off the west sided yellow light.
          Hence, the traffic now flows in the initial direction
until another car approaches the east side of the bridge.
End Loop
```

1.4 Flowchart

A flowchart is a visual representation of a code; it shows the flow of operations of a certain process.

Figure 1 represents the flowchart of mini project 1.

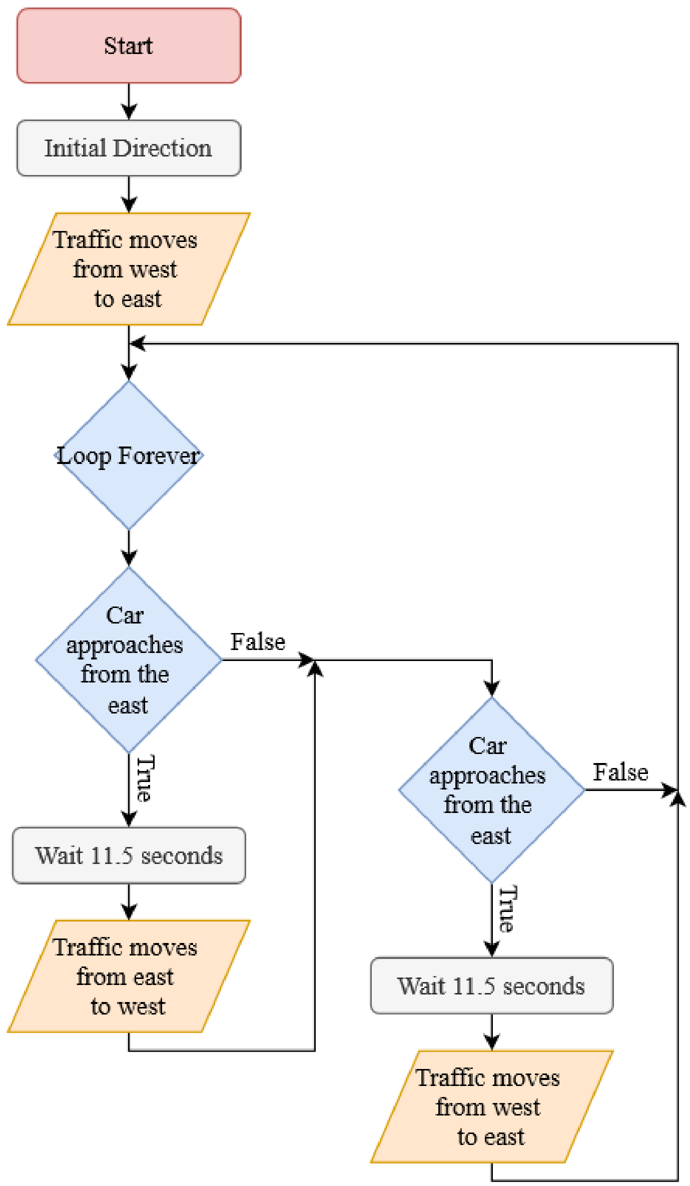


Figure 1 Flowchart of Mini Project 1

The "Wait 11.5 seconds" represents the sequence of turning on/off traffic lights described in the pseudocode of this project.

1.5 Arduino Circuit

The following is a list of the components used in this project:

1- 2 momentary buttons
2- 2 green LEDs
3- 2 red LEDs
4- 2 yellow LEDs
5- 6 220Ω resistors
6- 2 10KΩ resistors
7- 1 Arduino Uno board and its USB cable
8- 1 9V DC battery to power the Arduino without needing a USB connection
9- Connecting wires

Using Tinkercad, a schematic representation, figure 2, of the circuit is drawn.

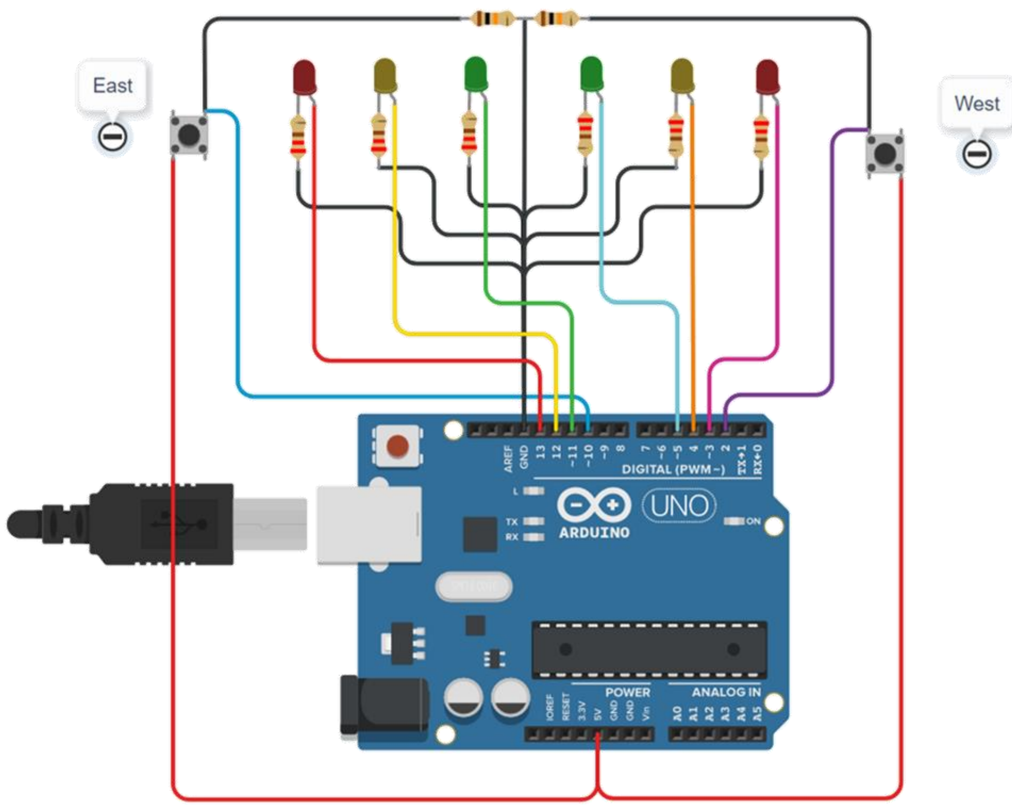


Figure 2 Schematic Representation of The Circuit of Mini Project 1

1.6 Arduino Code

The code is constructed using simple C – language, and it is the following.

First, the input and output pins are defined where each input, the momentary buttons, are labelled according to its location, east or west, plus the word 'button'. Moreover, each output, the LEDs, are represented by 2 letters. The first letter indicates the color and the second the location.

```
//West Side
int West_button = 2;
int GW = 5;
int YW = 4;
int RW = 3;
int bw; //Indicates the state of west_button (HIGH or LOW)

//East Side
int East_button = 10;
int GE = 11;
int YE = 12;
int RE = 13;
int be; //Indicates the state of east_button (HIGH or LOW)

int p = 0;
```

Next, the inputs and outputs are set up, and the initial direction of traffic flow is initiated.

```
void setup() {
//Inputs
pinMode(West_button, INPUT);
pinMode(East_button, INPUT);
```

```
//OUTPUTS
pinMode(GW, OUTPUT);
pinMode(YW, OUTPUT);
pinMode(RW, OUTPUT);
pinMode(GE, OUTPUT);
pinMode(YE, OUTPUT);
pinMode(RE, OUTPUT);

//Initially or Default
digitalWrite(GW, HIGH);
digitalWrite(RE, HIGH);
}
```

Lastly, the traffic flow changes direction according the following sequence.

```
void loop() {
 bw = digitalRead(West_button);
 be = digitalRead(East_button);

 //A car approaches from the east.
 if (be == HIGH && p == 0 && bw == LOW){p = 1;}
 if (p == 1){
  delay(500);
  digitalWrite(GW, LOW);
  delay(1000);
  digitalWrite(YW, HIGH);
  delay(4000);
  digitalWrite(YW, LOW);
  digitalWrite(RW, HIGH);
```

```
   delay(3000);
   digitalWrite(YE, HIGH);
   digitalWrite(RE, LOW);
   delay(3000);
   digitalWrite(GE, HIGH);
   digitalWrite(YE, LOW);
   p = 2;
  }

  //A car approaches from the west.
  if (bw == HIGH && p == 2 && be == LOW){p = 3;}
  if (p == 3){
   delay(500);
   digitalWrite(GE, LOW);
   delay(1000);
   digitalWrite(YE, HIGH);
   delay(4000);
   digitalWrite(YE, LOW);
   digitalWrite(RE, HIGH);
   delay(3000);
   digitalWrite(YW, HIGH);
   digitalWrite(RW, LOW);
   delay(3000);
   digitalWrite(GW, HIGH);
   digitalWrite(YW, LOW);
   p = 0;
  }
 }
```

1.7 Simulation Results

Simulation using Tinkercad produces satisfying results; the circuit behaved as expected. In addition, a video of the simulation named as “Traffic Control Simulation Video” has been uploaded alongside this document since screenshots of the simulation cannot demonstrate the functioning of this circuit.

Lastly, a prototype of this project has been constructed. The testing video of this prototype is named “Traffic Control Prototype Test Video”.

Mini Project 2 – Temperature Alarm

2.1 Introduction

An alarm will sound in case the temperature increases above a threshold of 26°C so that for a temperature range of [26°C, 32°C[, the beeping frequency will be 1Hz. Then after the temperature trespasses 32°C, the piezo transducer will beep faster at 10 times per second.

2.2 Inputs and Outputs

This system has one input, the temperature, and one output, the piezo transducer's beeping.

2.2.1 Inputs

The temperature is measured by an LM35DZ temperature sensor, an analog sensor, that is connected to the analog pin A0 of the Arduino. This sensor is linear and can measure a range of [0, 100°C] with a gain of 10mV/°C. The sensor's datasheet can be found in the appendix.

2.2.2 Outputs

The 9V piezo transducer, considered as a digital output, is connected to the digital pin 13.

2.3 Pseudocode

The pseudocode for this project is as follows.

```
Start Program
Loop Forever
     Measure temperature using sensor.
     IF the temperature is less than 26°C THEN
          Turn off the piezo transducer.
     IF the temperature is between 26°C and 32°C THEN
          Turn on the alarm for 500ms.
          Turn off the alarm for 500ms.
     IF the temperature is greater than 32°C THEN
          Turn on the alarm for 50ms.
          Turn off the alarm for 50ms.
End Loop
```

2.4 Flowchart

Figure 3 visualizes the flow of processes of this project.

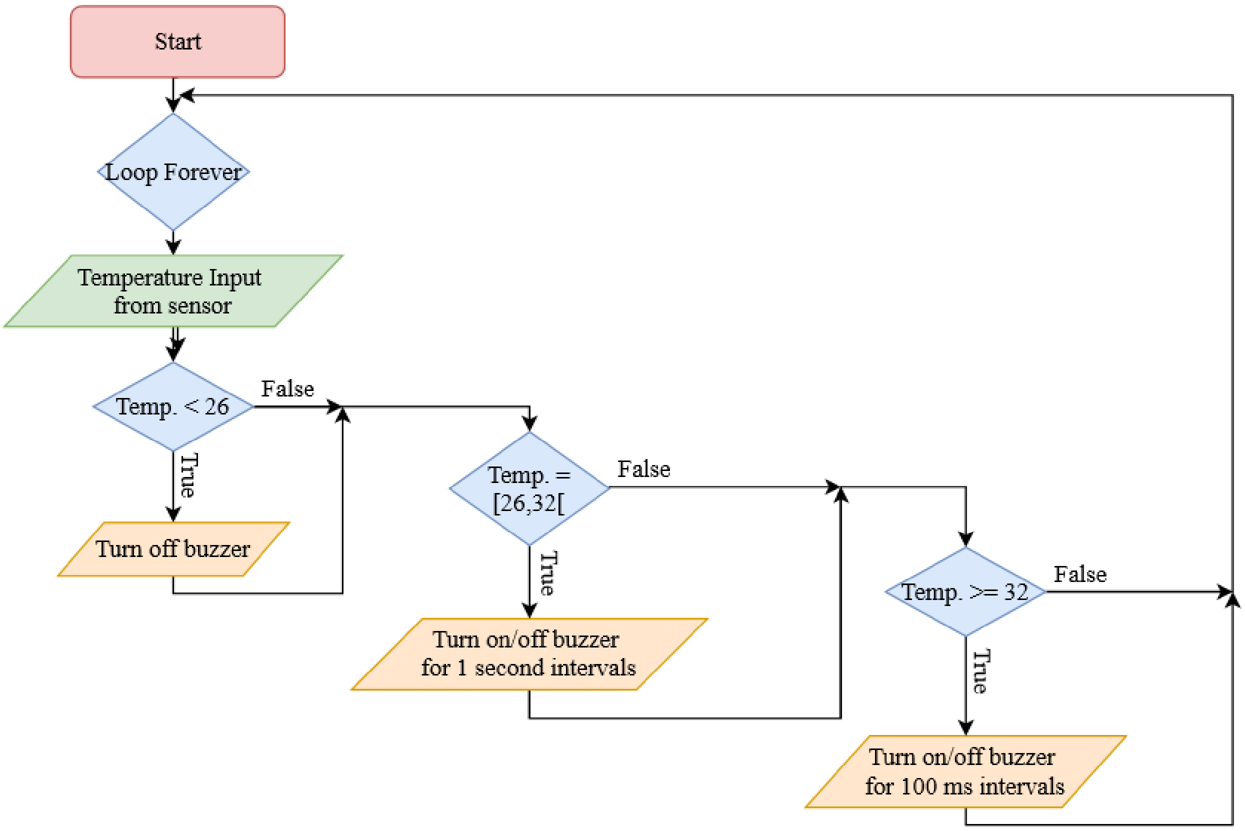


Figure 3 Flowchart of Mini Project 2

2.5 Arduino Circuit

The following components are used for this project:

1- An Arduino Uno with its USB cable
2- 1 LM35DZ temperature sensor
3- 1 9V piezo transducer
4- Connecting wires
5- A 1KΩ resistor
6- A 1.5KΩ resistor

Figure 4 is a schematic representation of the circuit; it is drawn using Fritzing.

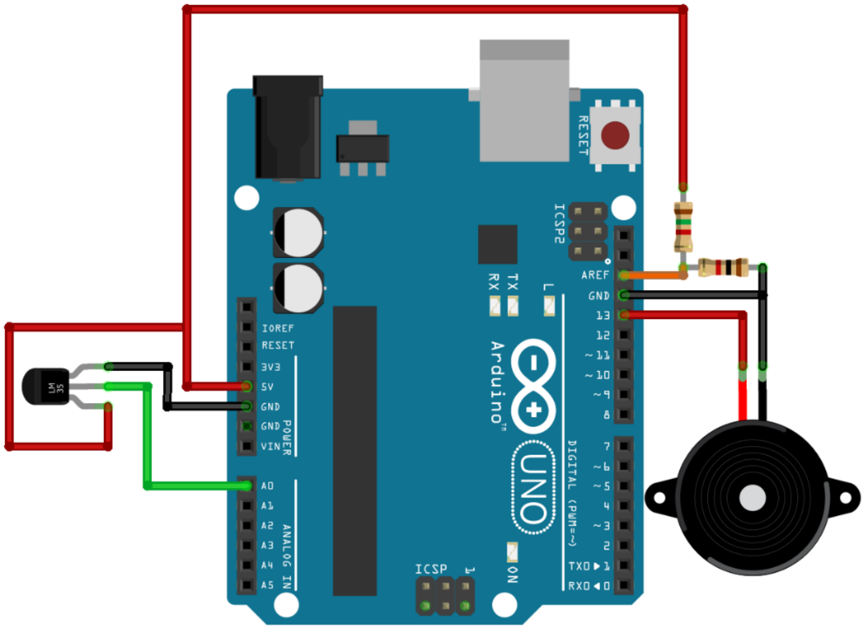


Figure 4 Schematic Representation of The Circuit of Mini Project 2

2.6 Arduino Code

Before presenting the code, the relationship between the quantization level and temperature must be derived.

The relationship between the voltage output of the sensor and the temperature is expressed by equation 1.

$$V = 0.01 \times T \tag{1}$$

Hence, it is expected that the voltage range of the sensor is between 0V and 1V. However, during testing, the sensor has outputted a maximum voltage of 1.5V, and this important when deciding upon the reference voltage to be used by the Arduino.

The 10-bit analog pins of the Arduino can receive a voltage between 0V and 5V; hence using a voltage reference of 2V is plausible. This reference voltage is acquired via a voltage divider and the 5V pin of the Arduino where:

$$2 = 5 \times \frac{R_1}{R_1 + R_2}; hence, R_2 = 1.5 \times R_1$$

Assuming R_1 to be 1KΩ, then R_2 is 1.5KΩ.

Hence, the Voltage-Quantization Level relationship is now:

$$Q_{level} = \frac{V - V_{min}}{V_{max} - V_{min}} \times Q_{level,max} = \frac{V-0}{2-0} \times 1023 = 511.5 \times V \quad (2)$$

Substituting for V from equation 1 in 2 results in equation 3:

$$Q_{level} = 5.115 \times T \quad (3)$$

Finally, the code is the following.

```
int buzzer = 13;

void setup() {
 pinMode(A0, INPUT);
 pinMode(buzzer, OUTPUT);
 Serial.begin(9600);
 analogReference(EXTERNAL);
}

void loop() {
 float temperature = analogRead(A0)/5.115;
 Serial.print("The temperature is: ");
 Serial.println(temperature);

 if (temperature < 26) {digitalWrite(buzzer, LOW);}

 if (temperature >= 26 && temperature < 32){
   digitalWrite(buzzer, HIGH);
   delay(500);
   digitalWrite(13, LOW);
   delay(500);
 }
```

```
  if (temperature >= 32){
    digitalWrite(buzzer, HIGH);
    delay(50);
    digitalWrite(buzzer, LOW);
    delay(50);
  }
}
```

For the temperature range [26°C, 32°C[, the time delay is chosen to be 500ms so that the total time period of the on/off becomes 1000ms or 1 second (as indicated in figure 5). Hence, the beeping frequency becomes 1Hz.

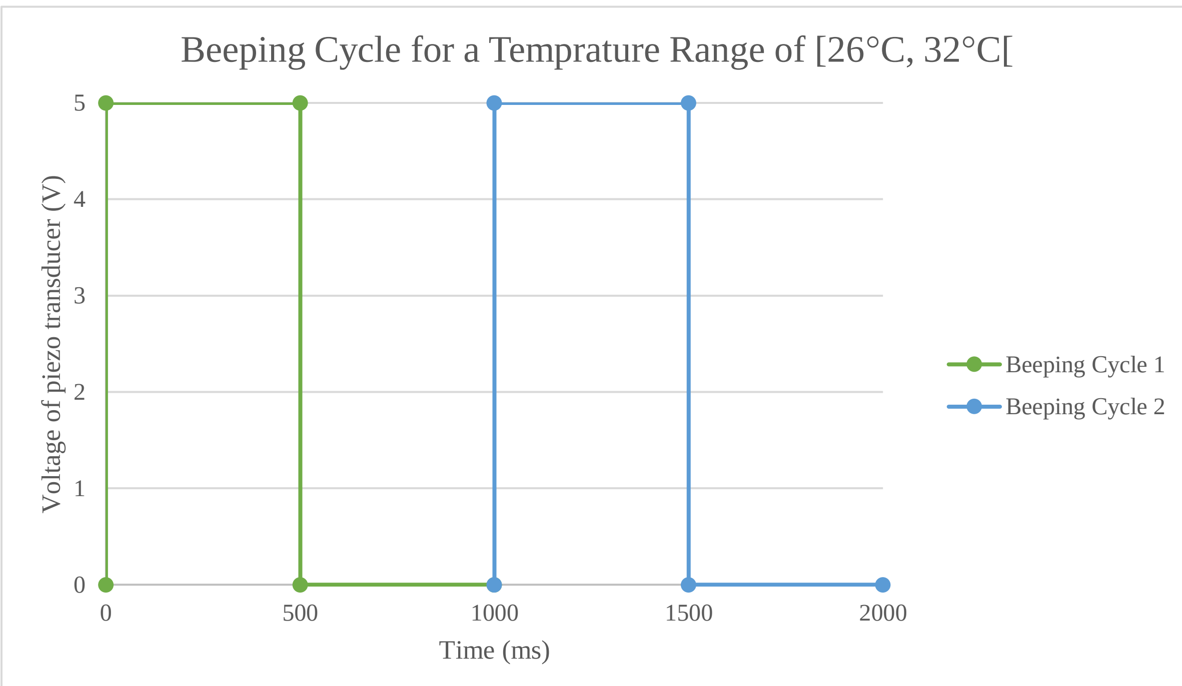


Figure 5 Beeping Cycle for a Temperature Range of [26°C, 32°C[

Similarly, when the temperature is greater than or equal to 32°C, the time delay is 50ms; therefore, the total cycle period becomes 100ms as indicated in figure 6.

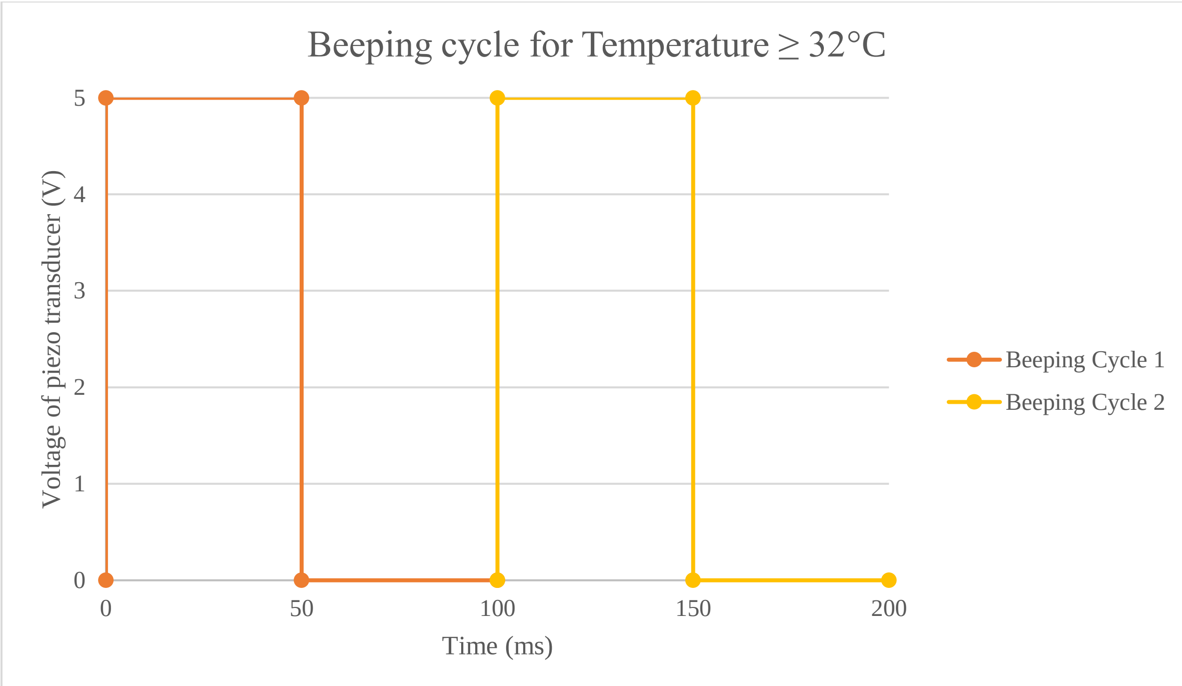


Figure 6 Beeping Cycle for a Temperature Greater Than or Equal to 32°C

2.7 Simulation Results

Simulation using Tinkercad produces satisfying results; the circuit behaved as expected. In addition, a video of the simulation named as "Temperature Alarm Simulation Video" has been uploaded alongside this document since screenshots of the simulation cannot demonstrate its functioning. Lastly, a prototype of this project has been constructed. The testing video of this prototype is named "Temperature Alarm Prototype Test Video".

Furthermore, it is worthy to note that Tinkercad only has the TMP36 temperature sensor. This sensor measures temperature between -40°C and 125°C with a gain of 10mV/°C (a datasheet of this sensor can be found in the appendix). In addition, at 25°C, the voltage output is 0.75V. Using this data, the voltage-temperature transfer function, equation 4, is obtained.

$$V = 0.01 \times T + 0.5 \quad (4)$$

Hence, at a temperature of 125°C, the voltage output is 1.75V. This means that an external reference voltage of 2V is optimal. Therefore, equation 2 is applicable for TMP36. As a result, the temperature-quantization level relationship is the following.

$$Q_{level} = 5.115 \times T + \; 511.5 \times 0.5 = 5.115 \times T + 255.75 \quad (5)$$

Mini Project 3 – Single-Cell Battery Tester

3.1 Introduction

Mini project 3 can be described as a “makeshift voltmeter”1 where the voltage output of single-cell batteries is going to be tested. If the battery is in a near perfect condition, it should have a voltage of 1.6V; as a result, a green led turns on. If the battery is in a good condition, i.e. having a voltage of [1.4V,1.6[, a yellow led turns on. Moreover, if it is in a bad condition, having a voltage of less than 1.4V, then a red led turns on. Lastly, each LED turns on briefly and once per test, and the test can be repeated by pressing the recheck button.

It is worthy to note that the user should not input a voltage more than 5V or connect the positive and negative poles in reverse due to the risk of damaging the Arduino.

3.2 Inputs and Outputs

The variables in this system are the voltage of the battery, the recheck button, and the LEDs.

3.2.1 Inputs

The inputs are the single-cell battery and the recheck button. The single-cell battery, an analog input, is simulated on Tinkercad by using a variable power supply, and it is connected to analog pin A0 of the Arduino. Furthermore, the recheck button, a digital input, is connected to digital pin 3 of the Arduino.

3.2.2 Outputs

The 3 LEDs green, yellow, and red are connected to digital pins 13, 9, and 5 respectively. Since the LEDs have 2 states, on or off, they are digital outputs.

3.3 Pseudocode

The pseudocode for this project is given below.

```
Start Program

Loop Forever

     Measure the battery's voltage.

     WHILE voltage belongs to [1.6V, 5V] THEN

          Turn on the green LED for 2 seconds and only once per test.

     ENDWHILE


     WHILE the voltage belongs to [1.4V, 1.6V[ THEN

          Turn on the yellow LED for 2 seconds and only once per test.

     ENDWHILE


     WHILE the voltage belongs to ]0V, 1.4V[ THEN

          Turn on the red LED for 2 seconds and only once per test.

     ENDWHILE


     IF recheck button has been pressed or the battery is removed THEN

          Repeat the test.

End Loop
```

3.4 Flowchart

Figure 7 represents the flowchart of this project.

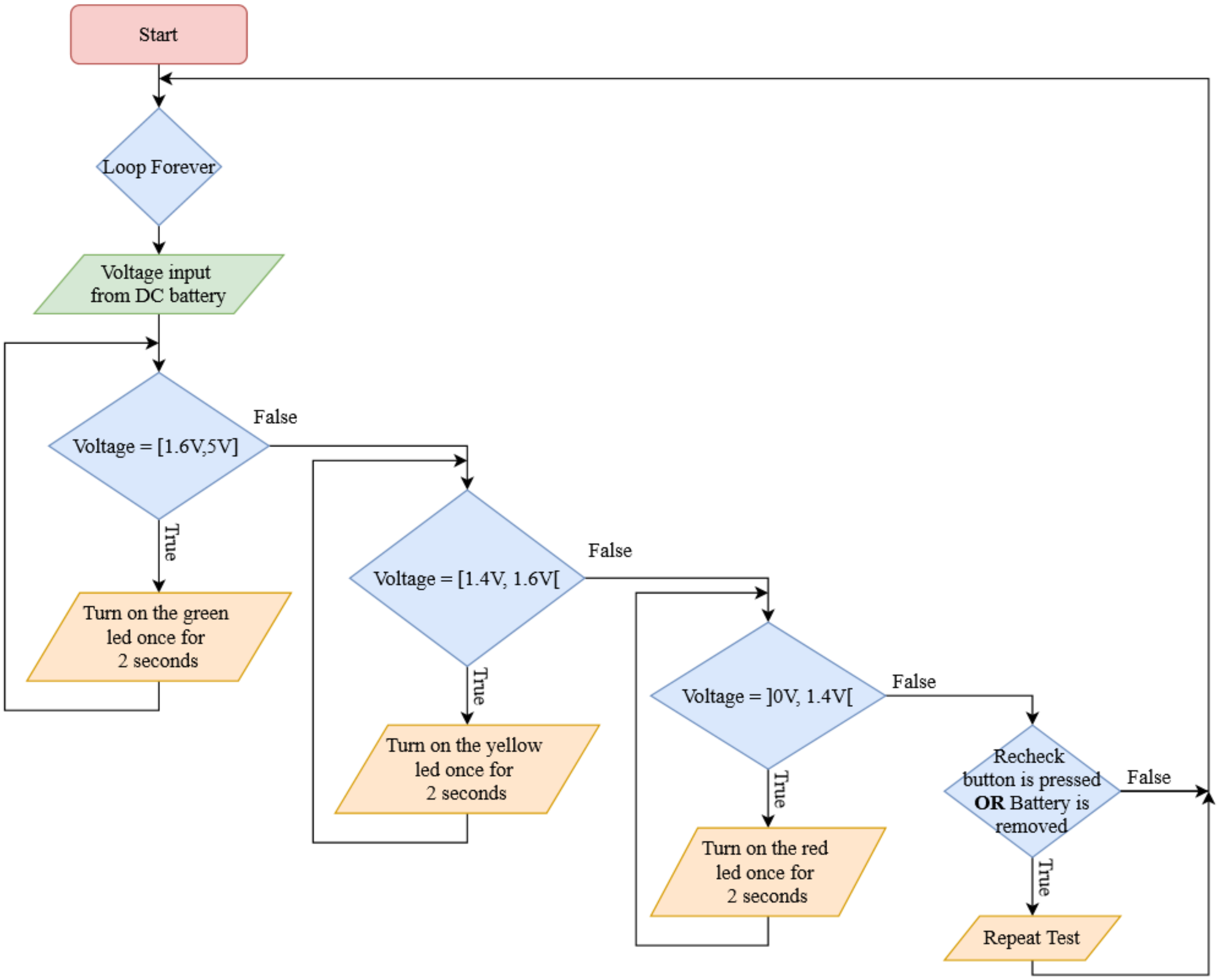


Figure 7 Flowchart of Mini Project 3

3.5 Arduino Circuit

The components used in this project are:

1- An Arduino Uno and its USB cable
2- Several single-cell batteries for testing
3- 3 LEDs (green, yellow, and red)
4- 3 220Ω resistors
5- 1 1KΩ resistor
6- 1 10KΩ resistor
7- A momentary push button
8- Connecting wires

Using Tinkercad, a scheme of the circuit is drawn.

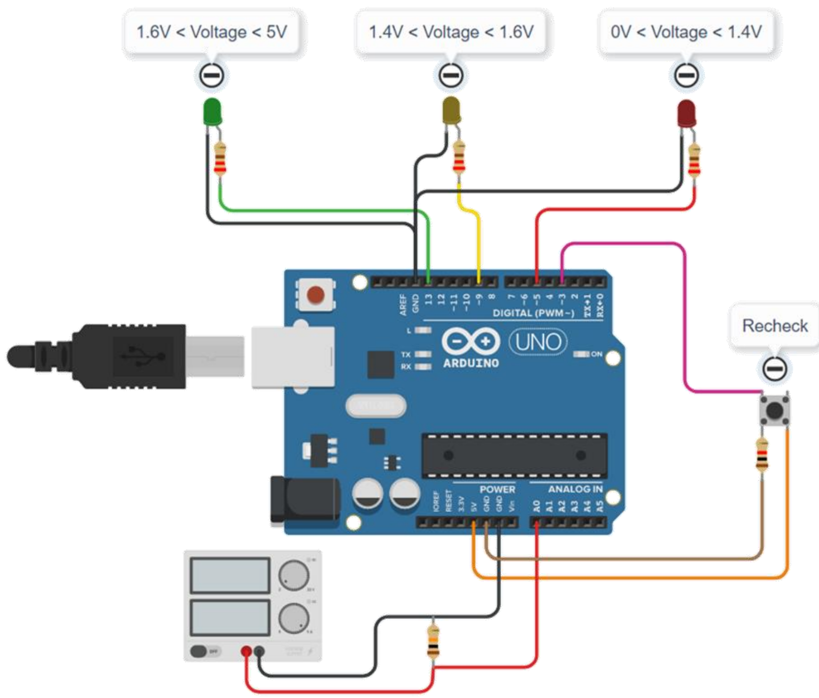


Figure 8 Schematic Representation of The Circuit of Mini Project 3

It is worthy to note that the resistor connected in parallel to the power supply is present to produce a voltage of zero when no battery cell is connected.

3.6 Arduino Code

Since there is possibility for an input voltage of 5V, there is no need for a reference voltage.

Hence, the transfer equation becomes:

$$Q_{Level} = \frac{V - V_{min}}{V_{max} - V_{min}} \times Q_{Level,max} = \frac{V-0}{5-0} \times 1023 = 204.6 \times V \quad (6)$$

Therefore, a voltage of 1.6V produces a quantization level of 327 and that of 1.4V produces a quantization level of 286.

Finally, the code is the following.

```
int G_LED = 13; // a green LED is connected to pin 13
int Y_LED = 9; // a yellow LED is connected to pin 9
int R_LED = 5; // a red LED is connected to pin 5
int recheck_button = 3; // to repeat the test.
int time_delay = 2000; //so that any LED lights up briefly (for
2 seconds).
float reading, recheck_state, g, y, r; // g, y, and r ensure
the green, yellow, and red LEDs respectively only once turn on
per test.


void setup() {
 pinMode(G_LED, OUTPUT);
 pinMode(Y_LED, OUTPUT);
 pinMode(R_LED, OUTPUT);
 pinMode(recheck_button, INPUT);
 pinMode(A0, INPUT);
}

void loop() {
 reading = analogRead(A0);
 recheck_state = digitalRead(recheck_button);

 //a reading of 327 corresponds to a voltage of 1.6V.
 while(g == 0 && reading >= 327 && reading <= 1023){
   digitalWrite(Y_LED, LOW);
   digitalWrite(R_LED, LOW);
   digitalWrite(G_LED, HIGH);
   delay(time_delay);
   digitalWrite(G_LED, LOW);
   g++;
 }
```

```
  //a reading of 286 corresponds to a voltage of 1.4V.
  while(y == 0 && reading >= 286 && reading < 327){
    digitalWrite(G_LED, LOW);
    digitalWrite(R_LED, LOW);
    digitalWrite(Y_LED, HIGH);
    delay(time_delay);
    digitalWrite(Y_LED, LOW);
    y++;
  }

  while(r == 0 && reading > 0 && reading < 286){
    digitalWrite(Y_LED, LOW);
    digitalWrite(G_LED, LOW);
    digitalWrite(R_LED, HIGH);
    delay(time_delay);
    digitalWrite(R_LED, LOW);
    r++;
  }

  if (g == 1 || y == 1 || r == 1){
    if (recheck_state == HIGH || reading == 0){
      g = 0;
      y = 0;
      r = 0;
    }
  }
}
```

3.7 Simulation Results

The following screenshots, figures 9, 10, and 11, show the different stages of simulation.

Figure 9 shows that for a voltage input of 1.6V, the green LED turns on, and this is expected.

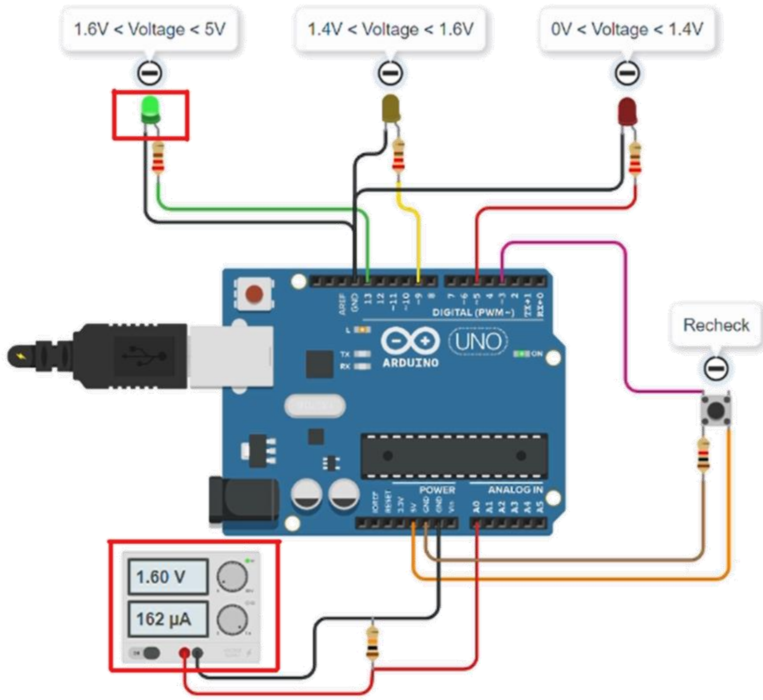


Figure 9 Supplying a Voltage of 1.6V

In figure 10, the yellow LED turns on upon supplying a voltage of 1.4V.

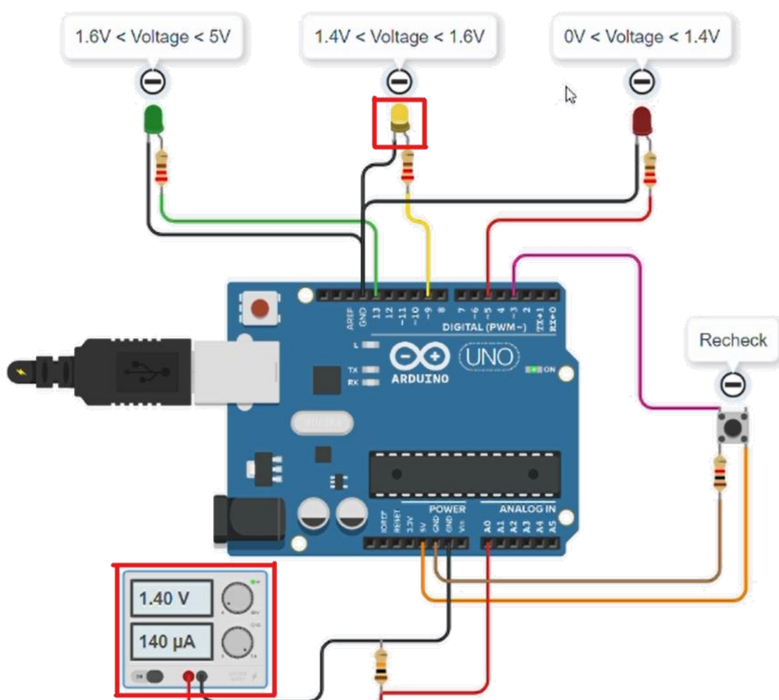


Figure 10 Supplying a Voltage of 1.4V

Lastly, upon supplying a voltage of 1V, the red LED turns on as it can be seen in figure 11.

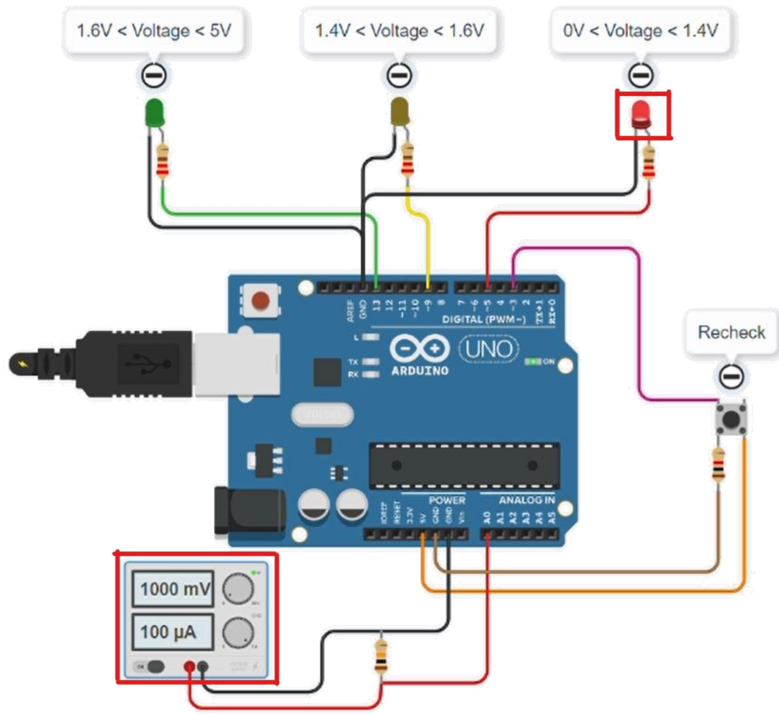


Figure 11 Supplying a Voltage of 1V

The action of pressing the momentary push button cannot be shown using screenshots of the simulation, but a screen recording of the simulation can do so. The file of said screen recording has been uploaded alongside this document; it is named "Single-Cell Battery Tester Simulation Video". Moreover, a prototype of this project has been constructed. The testing video of this prototype is named "Single-Cell Battery Tester Prototype Test Video".

Appendix

A) LM35DZ Datasheet

Product Folder | Order Now | Technical Documents | Tools & Software | Support & Community

TEXAS INSTRUMENTS

LM35

SNIS159H – AUGUST 1999 – REVISED DECEMBER 2017

LM35 Precision Centigrade Temperature Sensors

1 Features

- Calibrated Directly in Celsius (Centigrade)
- Linear + 10-mV/°C Scale Factor
- 0.5°C Ensured Accuracy (at 25°C)
- Rated for Full −55°C to 150°C Range
- Suitable for Remote Applications
- Low-Cost Due to Wafer-Level Trimming
- Operates From 4 V to 30 V
- Less Than 60-μA Current Drain
- Low Self-Heating, 0.08°C in Still Air
- Non-Linearity Only ±¼°C Typical
- Low-Impedance Output, 0.1 Ω for 1-mA Load

2 Applications

- Power Supplies
- Battery Management
- HVAC
- Appliances

3 Description

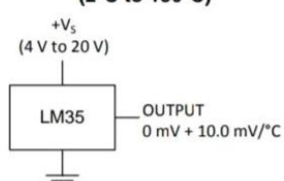
The LM35 series are precision integrated-circuit temperature devices with an output voltage linearly-proportional to the Centigrade temperature. The LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient Centigrade scaling. The LM35 device does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C over a full −55°C to 150°C temperature range. Lower cost is assured by trimming and calibration at the wafer level. The low-output impedance, linear output, and precise inherent calibration of the LM35 device makes interfacing to readout or control circuitry especially easy. The device is used with single power supplies, or with plus and minus supplies. As the LM35 device draws only 60 μA from the supply, it has very low self-heating of less than 0.1°C in still air. The LM35 device is rated to operate over a −55°C to 150°C temperature range, while the LM35C device is rated for a −40°C to 110°C range (−10° with improved accuracy). The LM35-series devices are available packaged in hermetic TO transistor packages, while the LM35C, LM35CA, and LM35D devices are available in the plastic TO-92 transistor package. The LM35D device is available in an 8-lead surface-mount small-outline package and a plastic TO-220 package.

Device Information(1)

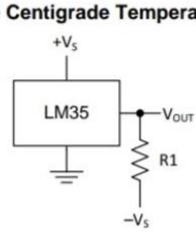
PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM35	TO-CAN (3)	4.699 mm × 4.699 mm
	TO-92 (3)	4.30 mm × 4.30 mm
	SOIC (8)	4.90 mm × 3.91 mm
	TO-220 (3)	14.986 mm × 10.16 mm

(1) For all available packages, see the orderable addendum at the end of the datasheet.

Basic Centigrade Temperature Sensor (2°C to 150°C)



Full-Range Centigrade Temperature Sensor



Choose $R_1 = -V_S / 50\ \mu A$
V_{OUT} = 1500 mV at 150°C
V_{OUT} = 250 mV at 25°C
V_{OUT} = −550 mV at −55°C

An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

6 Specifications

6.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)(1)(2)

		MIN	MAX	UNIT
Supply voltage		–0.2	35	V
Output voltage		–1	6	V
Output current			10	mA
Maximum Junction Temperature, $T_J$max			150	°C
Storage Temperature, T_{stg}	TO-CAN, TO-92 Package	–60	150	°C
	TO-220, SOIC Package	–65	150	

(1) If Military/Aerospace specified devices are required, please contact the Texas Instruments Sales Office/ Distributors for availability and specifications.
(2) Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its rated operating conditions.

6.2 ESD Ratings

			VALUE	UNIT
$V_{(ESD)}$	Electrostatic discharge	Human-body model (HBM), per ANSI/ESDA/JEDEC JS-001(1)	±2500	V

(1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.

6.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

		MIN	MAX	UNIT
Specified operating temperature: T_{MIN} to T_{MAX}	LM35, LM35A	–55	150	°C
	LM35C, LM35CA	–40	110	
	LM35D	0	100	
Supply Voltage (+V_S)		4	30	V

6.4 Thermal Information

THERMAL METRIC(1)(2)		LM35				UNIT
		NDV	LP	D	NEB	
		3 PINS		8 PINS	3 PINS	
$R_{\theta JA}$	Junction-to-ambient thermal resistance	400	180	220	90	°C/W
$R_{\theta JC(top)}$	Junction-to-case (top) thermal resistance	24	—	—	—	

(1) For more information about traditional and new thermal metrics, see the *IC Package Thermal Metrics* application report, SPRA953.
(2) For additional thermal resistance information, see *Typical Application*.

6.7 Electrical Characteristics: LM35, LM35C, LM35D Limits

Unless otherwise noted, these specifications apply: $-55°C \le T_J \le 150°C$ for the LM35 and LM35A; $-40°C \le T_J \le 110°C$ for the LM35C and LM35CA; and $0°C \le T_J \le 100°C$ for the LM35D. $V_S = 5$ Vdc and $I_{LOAD} = 50$ µA, in the circuit of Full-Range Centigrade Temperature Sensor. These specifications also apply from 2°C to T_{MAX} in the circuit of Figure 14.

PARAMETER	TEST CONDITIONS	LM35			LM35C, LM35D			UNIT
		TYP	TESTED LIMIT(1)	DESIGN LIMIT(2)	TYP	TESTED LIMIT(1)	DESIGN LIMIT(2)	
Accuracy, LM35, LM35C(3)	$T_A = 25°C$	±0.4	±1		±0.4	±1		°C
	$T_A = -10°C$	±0.5			±0.5		±1.5	
	$T_A = T_{MAX}$	±0.8	±1.5		±0.8		±1.5	
	$T_A = T_{MIN}$	±0.8		±1.5	±0.8		±2	
Accuracy, LM35D(3)	$T_A = 25°C$				±0.6	±1.5		°C
	$T_A = T_{MAX}$				±0.9		±2	
	$T_A = T_{MIN}$				±0.9		±2	
Nonlinearity(4)	$T_{MIN} \le T_A \le T_{MAX}$, $-40°C \le T_J \le 125°C$	±0.3		±0.5	±0.2		±0.5	°C
Sensor gain (average slope)	$T_{MIN} \le T_A \le T_{MAX}$, $-40°C \le T_J \le 125°C$	10	9.8		10		9.8	mV/°C
		10	10.2		10		10.2	
Load regulation(5) $0 \le I_L \le 1$ mA	$T_A = 25°C$	±0.4	±2		±0.4	±2		mV/mA
	$T_{MIN} \le T_A \le T_{MAX}$, $-40°C \le T_J \le 125°C$	±0.5		±5	±0.5		±5	
Line regulation(5)	$T_A = 25°C$	±0.01	±0.1		±0.01	±0.1		mV/V
	$4\text{ V} \le V_S \le 30$ V, $-40°C \le T_J \le 125°C$	±0.02		±0.2	±0.02		±0.2	
Quiescent current(6)	$V_S = 5$ V, 25°C	56	80		56	80		µA
	$V_S = 5$ V, $-40°C \le T_J \le 125°C$	105		158	91		138	
	$V_S = 30$ V, 25°C	56.2	82		56.2	82		
	$V_S = 30$ V, $-40°C \le T_J \le 125°C$	105.5		161	91.5		141	
Change of quiescent current(5)	$4\text{ V} \le V_S \le 30$ V, 25°C	0.2	2		0.2	2		µA
	$4\text{ V} \le V_S \le 30$ V, $-40°C \le T_J \le 125°C$	0.5		3	0.5		**3**	
Temperature coefficient of quiescent current	$-40°C \le T_J \le 125°C$	0.39		0.7	0.39		0.7	µA/°C
Minimum temperature for rate accuracy	In circuit of Figure 14, $I_L = 0$	1.5		2	1.5		2	°C
Long term stability	$T_J = T_{MAX}$, for 1000 hours	±0.08			±0.08			°C

(1) Tested Limits are ensured and 100% tested in production.
(2) Design Limits are ensured (but not 100% production tested) over the indicated temperature and supply voltage ranges. These limits are not used to calculate outgoing quality levels.
(3) Accuracy is defined as the error between the output voltage and 10 mv/°C times the case temperature of the device, at specified conditions of voltage, current, and temperature (expressed in °C).
(4) Non-linearity is defined as the deviation of the output-voltage-versus-temperature curve from the best-fit straight line, over the rated temperature range of the device.
(5) Regulation is measured at constant junction temperature, using pulse testing with a low duty cycle. Changes in output due to heating effects can be computed by multiplying the internal dissipation by the thermal resistance.
(6) Quiescent current is defined in the circuit of Figure 14.

B) TMP36 Datasheet

ANALOG DEVICES

Low Voltage Temperature Sensors

Data Sheet **TMP35/TMP36/TMP37**

FEATURES

Low voltage operation (2.7 V to 5.5 V)
Calibrated directly in °C
10 mV/°C scale factor (20 mV/°C on TMP37)
±2°C accuracy over temperature (typ)
±0.5°C linearity (typ)
Stable with large capacitive loads
Specified −40°C to +125°C, operation to +150°C
Less than 50 μA quiescent current
Shutdown current 0.5 μA max
Low self-heating
Qualified for automotive applications

APPLICATIONS

Environmental control systems
Thermal protection
Industrial process control
Fire alarms
Power system monitors
CPU thermal management

GENERAL DESCRIPTION

The TMP35/TMP36/TMP37 are low voltage, precision centigrade temperature sensors. They provide a voltage output that is linearly proportional to the Celsius (centigrade) temperature. The TMP35/TMP36/TMP37 do not require any external calibration to provide typical accuracies of ±1°C at +25°C and ±2°C over the −40°C to +125°C temperature range.

The low output impedance of the TMP35/TMP36/TMP37 and its linear output and precise calibration simplify interfacing to temperature control circuitry and ADCs. All three devices are intended for single-supply operation from 2.7 V to 5.5 V maximum. The supply current runs well below 50 μA, providing very low self-heating—less than 0.1°C in still air. In addition, a shutdown function is provided to cut the supply current to less than 0.5 μA.

The TMP35 is functionally compatible with the LM35/LM45 and provides a 250 mV output at 25°C. The TMP35 reads temperatures from 10°C to 125°C. The TMP36 is specified from −40°C to +125°C, provides a 750 mV output at 25°C, and operates to 125°C from a single 2.7 V supply. The TMP36 is functionally compatible with the LM50. Both the TMP35 and TMP36 have an output scale factor of 10 mV/°C.

FUNCTIONAL BLOCK DIAGRAM

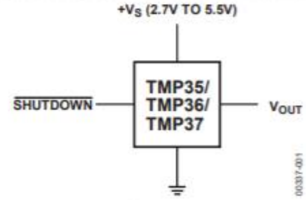


Figure 1.

PIN CONFIGURATIONS

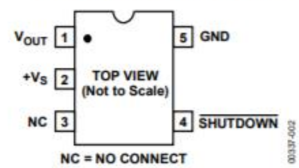


Figure 2. RJ-5 (SOT-23)

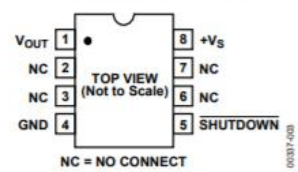
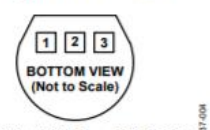


Figure 3. R-8 (SOIC_N)



PIN 1, $+V_S$; PIN 2, V_{OUT}; PIN 3, GND

Figure 4. T-3 (TO-92)

The TMP37 is intended for applications over the range of 5°C to 100°C and provides an output scale factor of 20 mV/°C. The TMP37 provides a 500 mV output at 25°C. Operation extends to 150°C with reduced accuracy for all devices when operating from a 5 V supply.

The TMP35/TMP36/TMP37 are available in low cost 3-lead TO-92, 8-lead SOIC_N, and 5-lead SOT-23 surface-mount packages.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700
Technical Support www.analog.com