

LEVEL 4 ASSESSMENT SPECIFICATION

Student name:	Shaun Harrison	Student ID number:	1810504
Programme:			
Module:	Embedded Systems		
Module code:	PCEE4004	Contribution to Overall Module Assessment (%):	100%
Lecturer:	Sarah Williams	Internal Verifier:	
Assignment Title:	Office Environment Monitor	Word count (or equivalent):	2000
Submission deadline:	10th May 2019	Return date of provisional marks & written feedback:	24th May 2019
Submission method:	All written assessments, where practical and possible, must be submitted via Turnitin unless otherwise instructed by the Lecturer. (Please DO NOT put this assessment specification into Turnitin or it will match many similarities with other students' submissions.) Alternative submission method (if applicable): Late submission of the assessment will result in a late penalty mark. Penalties for late submission: Up to one week late, maximum mark of 40%. Over one week late, 0%. Only the Extenuating Circumstances Panel may approve a change to submission dates.		
Academic honesty / referencing:	Academic honesty is required. In the main body of your submission you must give credit to authors on whose research and ideas your work is based. Append to your submission a reference list that indicates the books, articles, etc. that you have used, cited or quoted in order to complete this assessment.		

STUDENT FEEDBACK FORM

This section details the *extent* to which the assessment criteria are demonstrated by you, which in turn determines your mark. The marks available for each category of skill are shown. Lecturers will use the space provided to comment on the achievement of the task(s), including those areas in which you have performed well and areas that would benefit from development/improvement.

Generic Assessment Criteria	Marks available	Marks awarded
1. Engagement with Literature Skills		
Evidence of extensive research provided.	10	7
2. Knowledge and Understanding Skills		
Evidence of planning and consideration of components to use. Hardware and software design provided, using standard conventions and demonstrate excellent understanding of the topic area.	20	16
3. Cognitive and Intellectual Skills		
Evidence is provided to show that the system has been tested effectively and demonstrated as working as required in class. Also further discussion provides even more ways in which the system might be further improved, despite the fact it is already highly functional with many excellent features.	25	20
4. Practical Application Skills		
Highly functional circuit which provides extensive features for the environmental monitor. Code reflects complexity of the system developed, is well written using functions, reflecting flowcharts provided at design stage. This improves readability of the code although comments are also used to support understanding. Overall your project demonstrates an exceptional deployment of both practical programming and electrical circuit design and construction.	35	30
5. Transferable Skills for Life and Professional Practice		
	10	8

Work is coherent, very fluent and is presented professionally. Throughout you have demonstrated that you can work effectively independently.

Assessment Mark (Assessment marks are subject to ratification at the Exam Board. These comments and marks are to give feedback on module work and are for guidance only until they are confirmed.)

Late Submission Penalties (tick if appropriate)

Up to 1 week late (40% Max)

Over 1 week late (0%)

81%

GENERIC ASSESSMENT CRITERIA

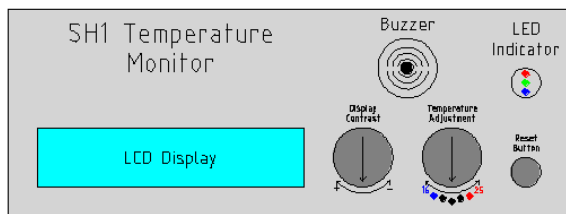
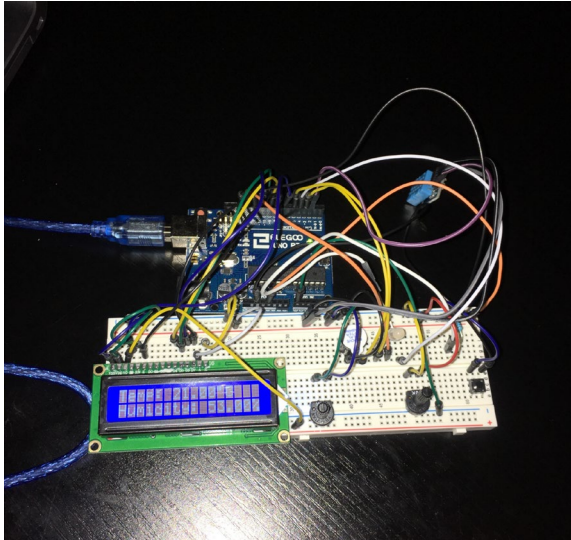
Level 4

In accordance with the Framework for Higher Education Qualifications, at the end of Level 4 students will be expected to have demonstrated knowledge of the basic underlying concepts and principles of a subject, and an ability to evaluate and interpret these within the context of that area of study. They should be able to present, evaluate and interpret qualitative and quantitative data in order to develop lines of argument and make sound judgements in accordance with basic theories and concepts of their subject(s) of study. They will have learned how to evaluate different approaches to solving problems, and will be able to communicate the results of their study/work accurately and reliably, and with structured and coherent arguments. They will be able to undertake further training and develop new skills within a structured and managed environment and will have the qualities and transferable skills necessary for employment requiring the exercise of some personal responsibility.

Level 4	FAIL	MARGINAL FAIL	SATISFACTORY (3 rd / Pass)	GOOD (2.2 / Pass)	VERY GOOD (2.1 / Merit)	EXCELLENT (1 st / Distinction)	EXCEPTIONAL (1 st / Distinction)
Category	0-29%	30-39%	40-49%	50-59%	60-69%	70-84%	85-100%
Engagement with literature (including reading, referencing, academic conventions and academic honesty)	Little or no evidence of reading and/or reliance on inappropriate sources. Views and findings mostly unsupported and non-authoritative. Referencing conventions used incoherently or largely absent.	Poor engagement with essential reading. No evidence of wider reading. Reliance on inappropriate sources, and/or indiscriminate use of sources. Heavily reliant on information gained through class contact. Inconsistent and weak use of referencing.	Engagement with a limited range of mostly relevant and credible sources but with some reliance on information gained through class contact. Some omissions and minor errors. Referencing conventions evident though not always applied accurately or consistently.	Engagement with an appropriate range of literature, including sources retrieved independently. Some over-reliance on texts rather than other sources. Referencing may show minor inaccuracies or inconsistencies.	Engagement with a wide range of literature, including sources retrieved independently. Selection of relevant and credible sources. Generally sound referencing, with no/very few inaccuracies or inconsistencies.	Engagement with an extensive range of relevant and credible literature. Consistently accurate application of referencing.	Exceptional engagement with an extensive range of relevant and credible literature. High-level referencing skills consistently applied.
Knowledge and understanding (Knowledge of the basic underlying concepts and principles of a subject.)	Major gaps in knowledge and understanding of the basic underlying concepts and principles of the subject matter. Inclusion of irrelevant material. Substantial inaccuracies.	Gaps in knowledge of the basic underlying concepts and principles, with flawed or superficial understanding. Some significant inaccuracies and/or irrelevant material.	Limited knowledge and understanding of the basic underlying concepts and principles within the subject area. Some elements may be missing.	Knowledge of the basic underlying concepts and principles is accurate with a good understanding of the field of study but lacks depth and/or breadth.	Competent knowledge of the basic underlying concepts and principles. Exhibits very good understanding.	Excellent knowledge and understanding of the basic underlying concepts and principles of the subject.	Exceptional, detailed knowledge and understanding of the basic underlying concepts and principles

Level 4	FAIL	MARGINAL FAIL	SATISFACTORY (3 rd / Pass)	GOOD (2.2 / Pass)	VERY GOOD (2.1 / Merit)	EXCELLENT (1 st / Distinction)	EXCEPTIONAL (1 st / Distinction)
Category	0-29%	30-39%	40-49%	50-59%	60-69%	70-84%	85-100%
Cognitive and intellectual skills (Evaluate underlying concepts and principles of a subject and interpret qualitative and quantitative data in order to develop lines of argument and make sound judgements.)	Wholly or almost wholly descriptive work. Little or no evaluation of the underlying concepts and principles. Failure to develop arguments, leading to illogical or invalid judgements. Minimal or no use of evidence to back up views.	Largely descriptive work, with superficial evaluation of the underlying concepts and principles. Weak interpretation of data, flawed development of arguments and judgements. Information accepted uncritically, uses generalised statements made with scant evidence and unsubstantiated opinions. Ideas sometimes illogical and contradictory.	Limited attempt at evaluation of the underlying concepts and principles, tending towards description. Can interpret qualitative and quantitative data but with some errors. Some evidence to support emerging arguments and judgements but these may be underdeveloped or with a little inconsistency / mis-interpretation.	Good evaluation of the underlying concepts and principles. Can interpret qualitative and quantitative data, with minor errors. An emerging ability to use evidence to support the argument. Mostly valid arguments and logical judgements.	Sound evaluation of the underlying concepts and principles. Can interpret qualitative and quantitative data accurately. Ability to devise arguments using evidence to make mostly appropriate and valid judgements.	Excellent evaluation of the underlying concepts and principles. Can interpret qualitative and quantitative data accurately and with some insight. Excellent ability to devise arguments using evidence and make appropriate and valid judgements.	Exceptional evaluation of the underlying concepts and principles based evidence. Outstanding interpretation of qualitative and quantitative data. Exceptional ability to devise arguments using evidence and make wholly appropriate and valid judgements.
Practical skills (Different approaches to solving problems in particular contexts.)	Limited or no use of taught, basic methods, materials, tools and/or techniques. Little or no appreciation of the context of the application. Very weak evidence of different approaches to problem-solving in particular contexts.	Rudimentary application of taught, basic methods, materials, tools and/or techniques but without consideration and competence. Flawed appreciation of the context of the application. Weak evidence of different approaches to problem-solving in particular contexts.	An adequate awareness and mostly appropriate application of basic methods, materials, tools and/or techniques. Basic appreciation of the context of the application. Can identify problems in particular contexts and propose basic alternative approaches or solutions though there may be errors.	A good and appropriate application of basic methods, materials, tools and/or techniques. Clear appreciation of the context of the application. Good evidence of different approaches to problem-solving in particular contexts and proposes mostly appropriate solutions.	A very good application of a range of basic methods, materials, tools and/or techniques. Very good consideration of the context of the application. Very good evidence of different approaches to problem-solving in particular contexts and proposes appropriate solutions.	An advanced application of a range of taught, basic methods, materials, tools and/or techniques. The context of the application is well considered, and insightful. Excellent evidence of different approaches to problem-solving in particular contexts and proposes appropriate solutions.	Exceptional levels of application and deployment skills in particular practical contexts. Outstanding identification of problems in particular contexts and formulation of wholly appropriate, thoughtful solutions / different approaches.

Transferable skills for life and professional practice (Communicate the results of their study/work accurately and reliably, and with structured and coherent arguments; the qualities needed for employment requiring the exercise of some personal responsibility within a structured and managed environment.)	Work is poorly structured, disorganised, inaccurate and/or confusingly expressed. Very weak use of language and/or very inappropriate style. Failure to work effectively individual or as part of a group. Little or no evidence of the skills for employment requiring the exercise of some personal responsibility.	Work is poorly presented in a disjointed manner. It is loosely, and at times incoherently, structured, with information and ideas often poorly expressed. Weak use of language and/or inappropriate style. Flawed approach to individual or group work, meeting only partial obligations to others. Limited evidence of the skills for employment requiring the exercise of some personal responsibility.	Mostly ordered presentation and structure in which relevant ideas / concepts are reasonably expressed. Work may lack coherence and/or accuracy in places. Can work as part of a group, meeting most obligations to others but perhaps with limited involvement in group activities. Demonstrates the basic skills for employment requiring the exercise of some personal responsibility, with some areas of minor weakness.	Mostly coherent, organised and accurate work, in a suitable structure and is for the most part clearly expressed. Can work effectively independently and/or as part of a team, with clear contribution to group activities. Demonstrates the skills for employment requiring the exercise of some personal responsibility, with some areas of strength and some of minor weakness.	Work is accurate, coherent, fluent, well-structured and organised. Can work effectively independently and/or as part of a team, with very good contribution to group activities. Demonstrates very good skills for employment requiring the exercise of some personal responsibility, with just occasional minor weakness.	Work is coherent, very fluent and is presented professionally. Can work effectively independently and/or as part of a team, with an excellent contribution to group activities. Demonstrates excellent skills for employment requiring the exercise of some personal responsibility with an appetite for further development.	Work is accurate, exceptionally coherent, very fluent and presented well. Can work effectively independently and/or as part of a team, with an exceptional contribution to group activities. Demonstrates exceptional skills for employment requiring the exercise of some personal responsibility with an appetite for further development.
---	--	--	--	---	---	--	---



EMBEDDED SYSTEMS ASSIGNMENT

Temperature Monitoring Device

TEMPERATURE MONITOR

The 'SH1 Temperature Monitor' is a user adjustable temperature monitor, that alerts the user to higher or lower temperatures than the users selected comfortable temperature.

Shaun Harrison

HNC Electrical and Electronic Engineering

Contents

Embedded Systems Assignment	2
Introduction	2
Hardware Design.....	2
Main Components.....	2
Wiring Diagram	3
Pictures.....	4
Software Design	4
Testing	4
Documentation	10
System Explanation.....	10
Setup	10
Main.....	11
Discussion, Evaluation and Conclusions	12
References	14
Bibliography	15
Appendices;	16
Data Sheets	16
Shaun Harrison – Appendix 1 - Code.....	17
Appendix 2 – Flow Charts	21
Appendix 4 – Quick Start Guide	25

Embedded Systems Assignment

Introduction

There have been concerns raised about the effectiveness of the heating and air conditioning system in the office. To assess the validity of the claims it has been decided the temperature in the room will be monitored, a monitoring device is to be designed and built, the monitoring device should raise an alarm when temperature either drops below or rises above the 'normal' temperature.

'Normal' temperature is not quantified, but the Health & Safety Executive website, and Energy Saving Trust website, along with various other online discussions and articles, suggest a range from 16°C to 22°, depending on the working environment and work itself.

Hardware Design

In order to make a system that can measure the temperature of its environment, display the temperature, and alert a user to temperatures higher or lower than their preferred temperature, the following main components will be used.

Main Components

DHT11 Temperature and Humidity Sensor

This component measures and reports temperature and humidity, it is required to compare the actual temperature against the set 'normal' temperature. See also Appendix 6 – DHT11 Sensor Data Sheet.

16*2 LCD Screen

This component allows the microprocessor to display information to the device user, such as a welcome message, instructions, or information. See also Appendix 7 – 16*2 LCD Data Sheet.

RGB LED

An Red Green Blue Light Emitting Diode is used to visually inform the user if the temperature is lower than acceptable, higher than acceptable, or acceptable. See also Appendix 8 – RGB LED Data Sheet.

Active Buzzer

An active buzzer will buzz when connected to a DC power supply, used as an alarm where temperature drops further below or rises further above 'normal'

Elegoo UNO Microcontroller

This is the programmable microcontroller that will run the program and decide on action to be taken after each temperature read. See also Appendix 9 – UNO Data Sheet.

10k Ohm Potentiometer

2 Potentiometers are used in the system;

One is used to adjust the contrast on the LCD screen.

One is used for the user to adjust the 'normal' temperature before the program starts.

330 Ohm Resistor

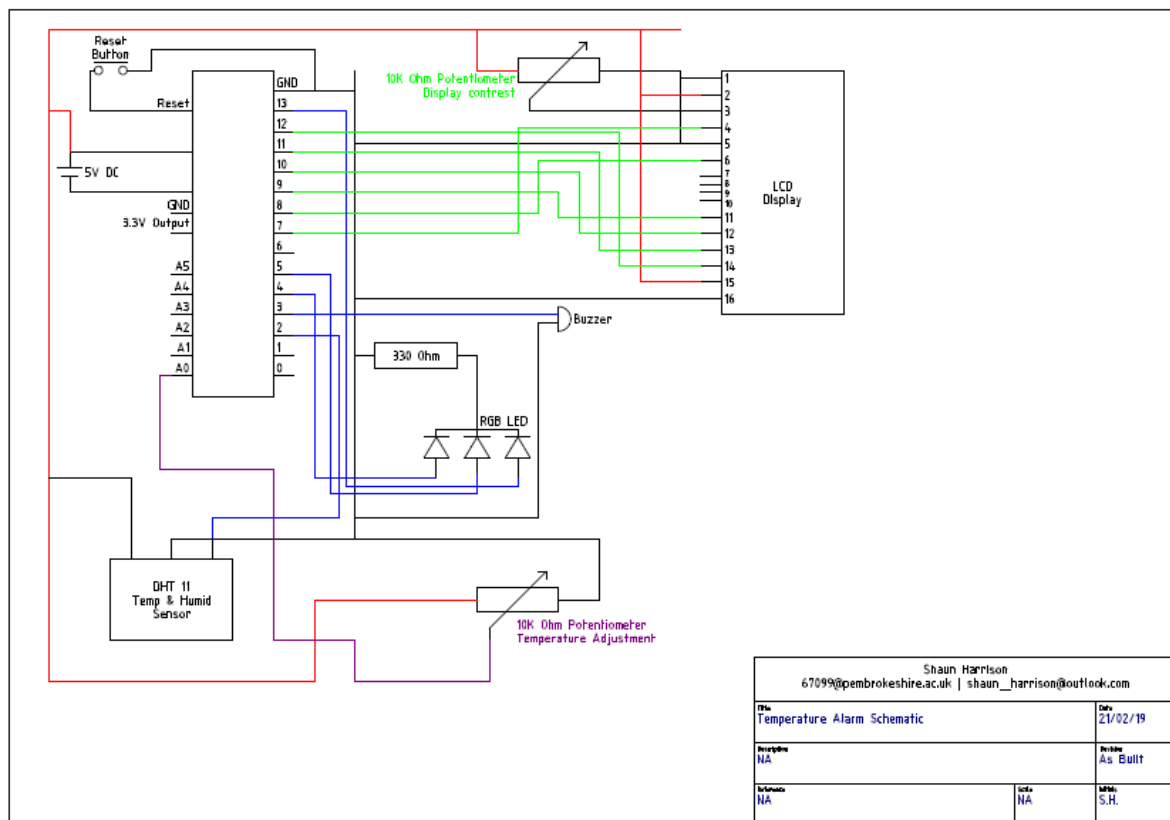
Used to reduce Current in the LED – From the datasheet it's determined the maximum forward current in the green LED should be 25 mA, in the red/blue LED's it's 30 mA. As $I = \frac{V}{R}$ and the supply voltage from the microcontroller is 5V, a 330 ohm resistor limits current to 15.2 mA.

Push Button Switch

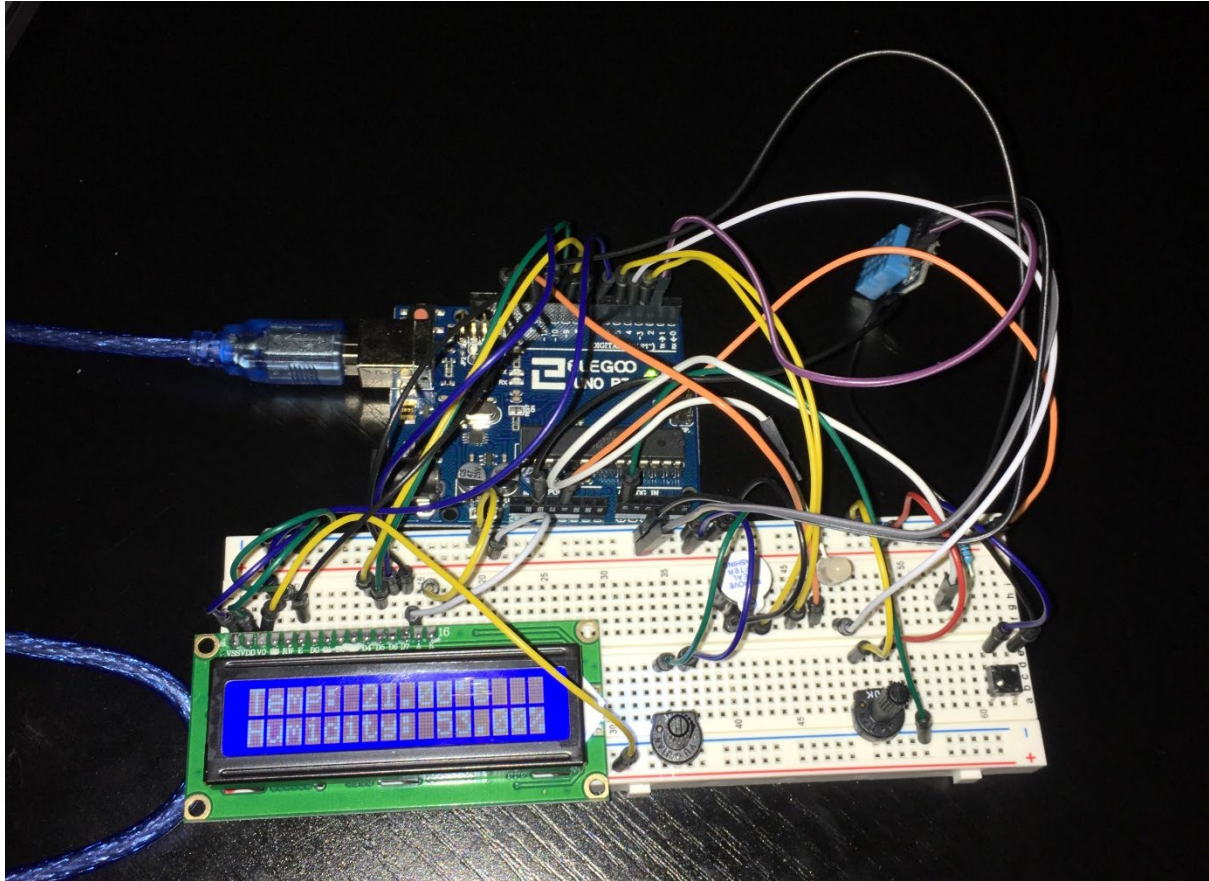
A push button switch is installed as a reset button. This allows the user to select a new temperature should they wish.

Wiring Diagram

Also attached as Appendix 3 – Wiring Diagram.



Pictures



Software Design

See Appendices 1 and 2, Code and Comments and Flow Chart.

Testing

All functions have been tested and work as expected.

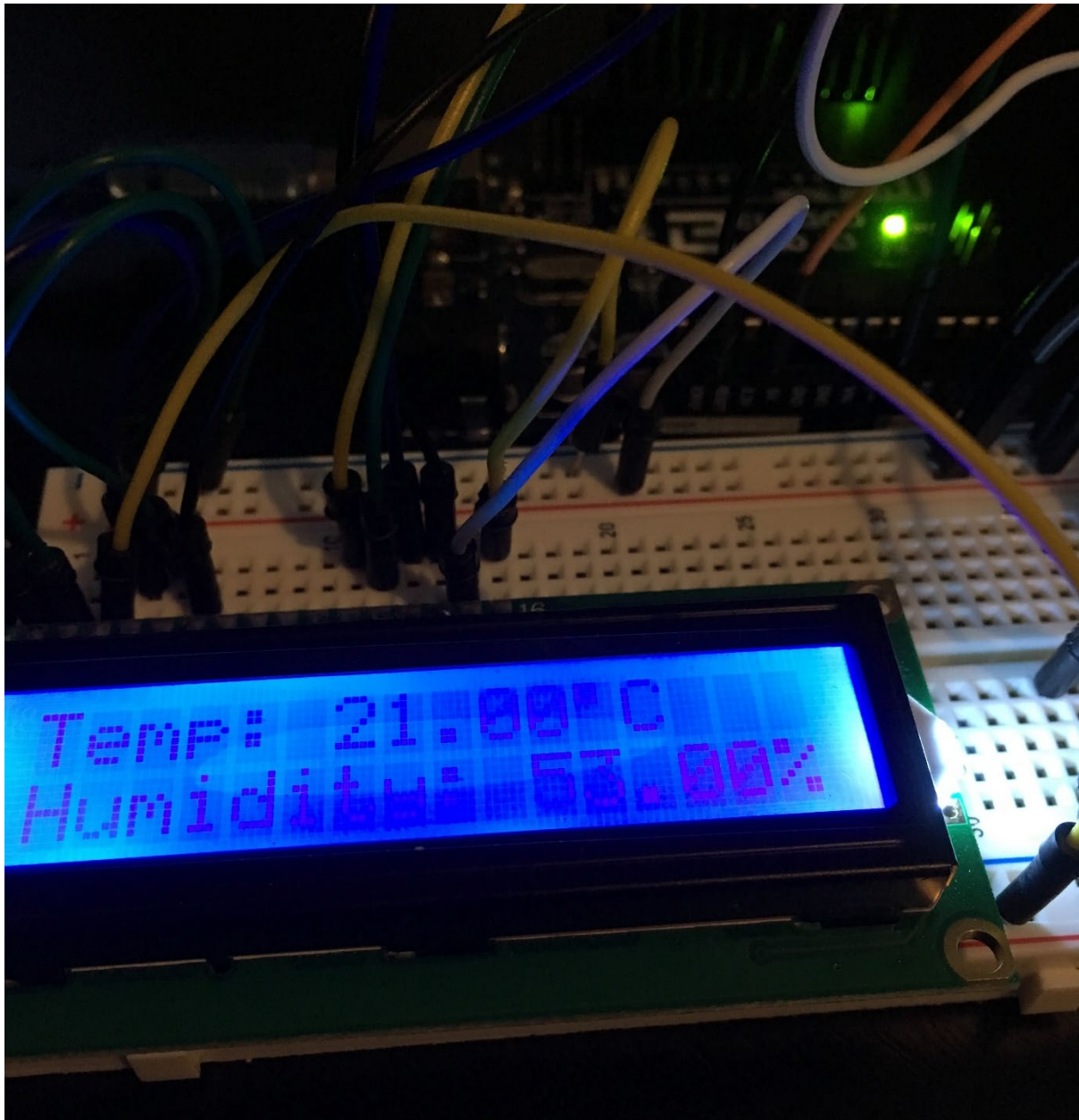
When powered up a welcome message is displayed on the LCD screen, as pictured below:



After a brief pause the setup code continues, and there is some time to set the users preferred temperature. By adjusting the potentiometer it is possible to select every value from 16 to 25.



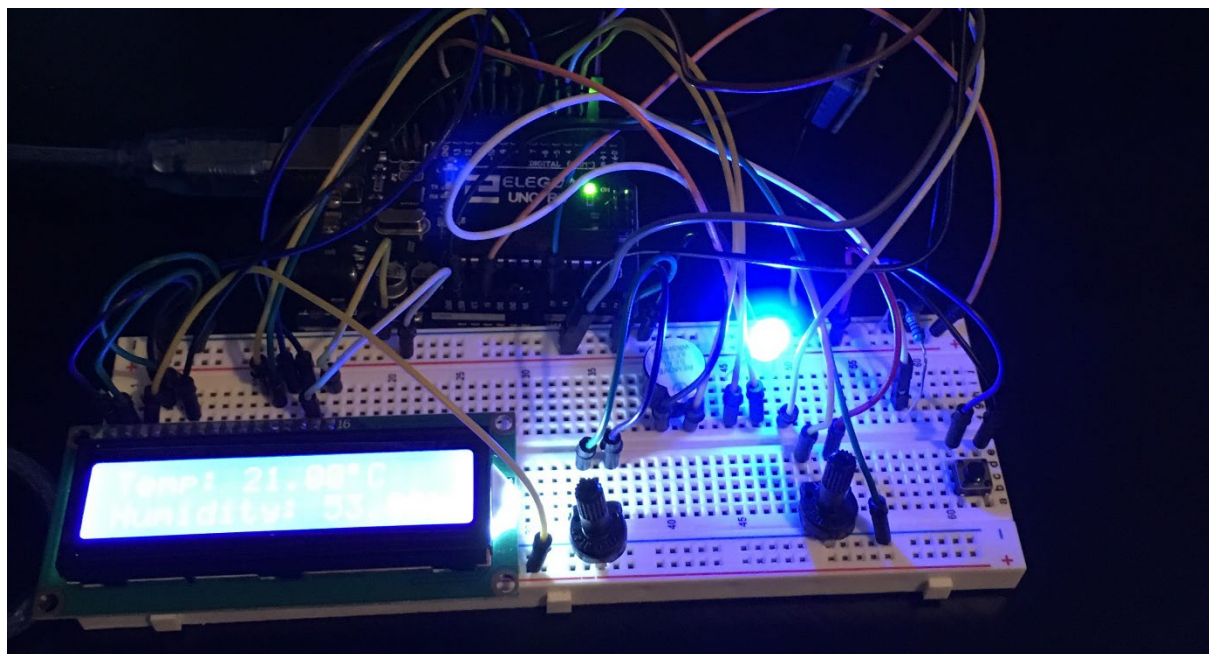
The reset button was tested, and allowed the user to go through the setup stage again, selecting a new temperature. Once the main loop was running the LCD screen continuously displays the temperature and humidity in the room.

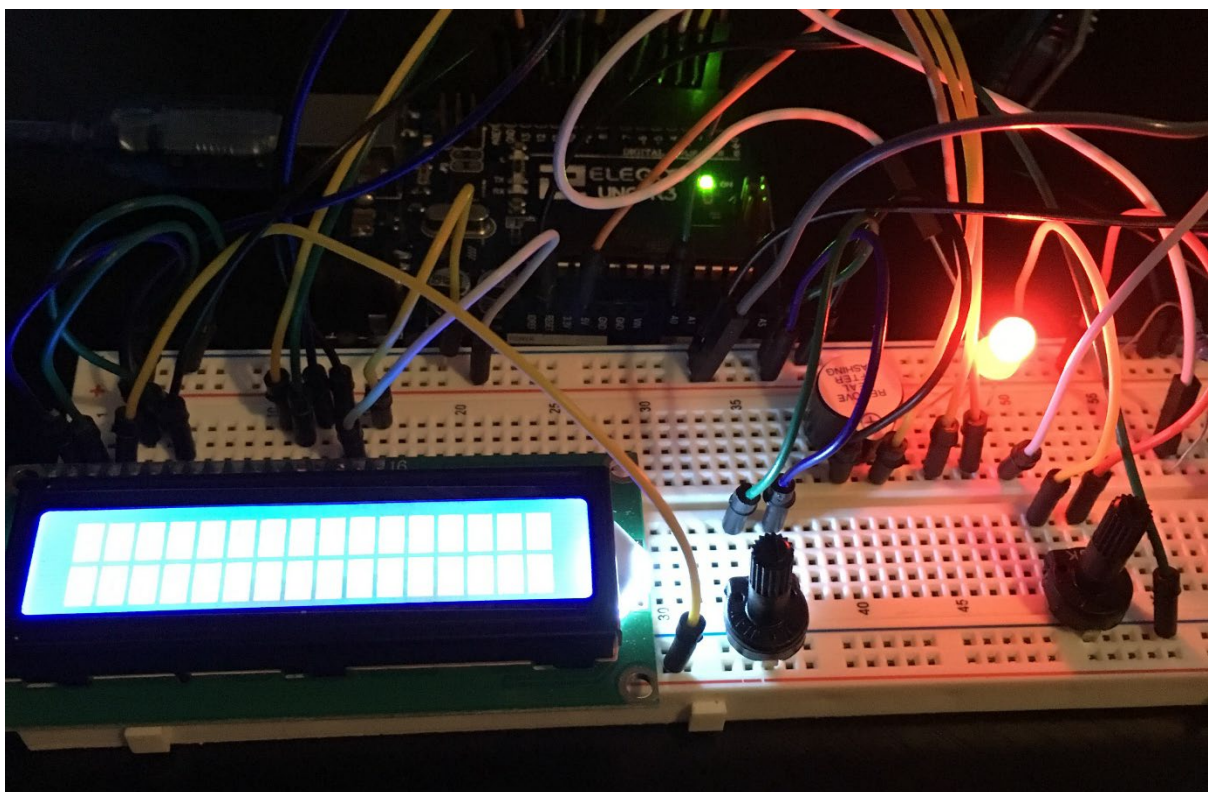
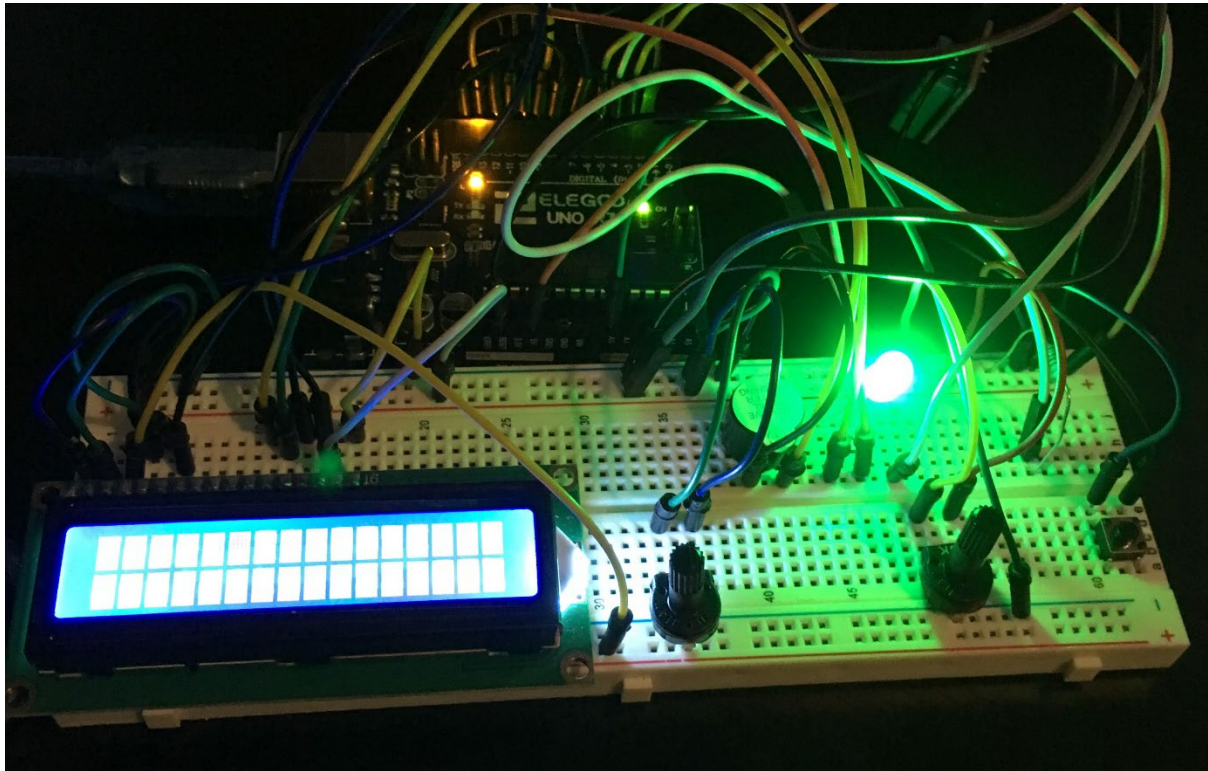


At this point it was determined the temperature in the room was 21°C, it was possible to test each function by varying the users preferred temperature. The following tests and observations were completed:

Set Temp (°C)	Room Temp (°C)	LED	Buzzer	Matches Function	Matches Expected Result
21	21 (+0)	Green – Short Flash	Off	normal	Yes
20	21 (+1)	Red – Short Flash	Off	high	Yes
19	21 (+2)	Red – Long Flash	On	highhigh	Yes
18	21 (+3)	Red – Long Flash	On	highhigh	Yes
22	21 (-1)	Blue – Short Flash	Off	low	Yes
23	21 (-2)	Blue – Long Flash	On	lowlow	Yes
24	21 (-3)	Blue – Long Flash	On	lowlow	Yes

Some pictures show the LED flashing all 3 colours.





Some final tests were completed to prove the device responded correctly to changing temperature.

The DHT11 temperature sensor was held to simulate an increase in room temperature, the following results were observed and recorded as measured temperature increased and subsequently decreased:

Set Temp (°C)	Measured Temp (°C)	LED	Buzzer	Matches Function	Matches Expected Result
21	21 (+0)	Green – Short Flash	Off	normal	Yes
21	22 (+1)	Red – Short Flash	Off	high	Yes
21	23 (+2)	Red – Long Flash	On	highhigh	Yes
21	24 (+3)	Red – Long Flash	On	highhigh	Yes
21	23 (+2)	Red – Long Flash	On	highhigh	Yes
21	22 (+1)	Red – Short Flash	Off	high	Yes
21	21 (+0)	Green – Short Flash	Off	normal	Yes

The DHT11 sensor was then held again, and the normal temperature set at 24°C, this time as the sensor cooled it would simulate a falling room temperature, results were observed and recorded:

Set Temp (°C)	Measured Temp (°C)	LED	Buzzer	Matches Function	Matches Expected Result
24	24 (+0)	Green – Short Flash	Off	normal	Yes
24	23 (-1)	Blue – Short Flash	Off	low	Yes
24	22 (-2)	Blue – Long Flash	On	lowlow	Yes
24	21 (-3)	Blue – Long Flash	On	lowlow	Yes

After cooling, the reset button was pressed, and the current room temperature (21°C) was selected again, the LED displayed short green flashes as expected, all eventualities and changes were tested and results were as expected.

Documentation

See Appendix 4 – Quick start guide for a brief instructional leaflet.

System Explanation

Setup

When the system is first powered up, it will first run through some setup code. This code has been programmed to show a welcome message on the LDC screen. Then it will display a

message instructing the user to set their temperature, and display the current temperature setting below.

Function SetTemp

A function is repeated to allow the user time to select the temperature they want to use as 'normal'.

The temperature is set by adjusting one of the potentiometers, analogue pin A0 reads a voltage input that is adjusted by the potentiometer, the value read could range from 0 to 1023, although testing determined the extreme upper and lower value's weren't actually achievable even at maximum adjustment. The input range from the potentiometer is set to 60-1000, an integer value is set between 16 and 25, based on the value read on pin A0 using the 'map' command, saved as a variable called 'settemp'.

The screen refreshes every ½ second to update the current temperature, and will refresh 10 times (giving a total of 5 seconds) before the setup code completes, and the program enters the main loop.

Main

When the main loop starts the microcontroller will read from the DHT11 sensor, the microcontroller will check for 4 scenarios, as soon as conditions for one of the scenarios are met it will run a function, then repeat the main loop. If the conditions do not meet the conditions required for any scenario, a function is run, after a 1 second delay, the main loop will repeat.

1st Scenario: If the temperature is more than 'settemp'+1, 2° or more over the users temperature selection, a function called 'highhigh' is run.

2nd Scenario: If the temperature is more than 'settemp', 1° over the users temperature selection, a function called 'high' is run.

3rd Scenario: If the temperature is lower than 'settemp'-1, 2° or more less than the users selected temperature, a function called 'lowlow' is run.

4th Scenario: If the temperature is lower than 'settemp', 1° under the users temperature selection, a function called 'low' is run

No Scenario: If none of the scenarios above are true, a function called 'normal' is run

Function: highhigh

If the function 'highhigh' is run the microcontroller will write the current temperature and humidity values to the LCD display, and will turn on the red LED pin, and the buzzer pin, wait 5 seconds, then turn both pins off and exit the function.

Function: high

If the function 'high' is run the microcontroller will write the current temperature and humidity values to the LCD display, and will turn on the red LED pin, wait 1 second before turning the pin off again, then exit the function.

Function: lowlow

If the function 'highhigh' is run the microcontroller will write the current temperature and humidity values to the LCD display, and will turn on the blue LED pin, and the buzzer pin, wait 5 seconds, then turn both pins off and exit the function.

Function: low

If the function low is run the microcontroller will write the current temperature and humidity values to the LCD display, and will turn on the blue LED pin, wait 1 second before turning the pin off again, wait 1 second before turning the pin off again, then exit the function.

Function: normal

If the function normal is run the microcontroller will write the current temperature and humidity values to the LCD display, and will turn on the green LED pin, wait 1 second before turning the pin off again, wait 1 second before turning the pin off again, then exit the function.

Discussion, Evaluation and Conclusions

The results observed throughout testing are consistent with expectations. The hardware and software exceed the design criteria and requirements, and the project can be considered a success. The device can successfully alarm a user to temperatures either lower or higher than the temperature they have selected as normal.

There are numerous ways the device could have been made more useful, including;

- Using a different microcontroller, with more pins. Additional inputs and outputs would have allowed for further hardware to be added, such as a keypad to set the temperature, or additional sensors.
- Using additional sensors. By having more than one temperature sensor located in different positions in the room a more accurate representation of the room temperature could be obtained, or the sensors could be used to identify wider problems with the heating system or building fabric. Cold spots could be identified, whereas the device built would mainly confirm correct operation of the thermostat.
- Consider humidity. Although measured and displayed the device does not consider humidity, an additional alarm could have been added if humidity is at an uncomfortable level.
- Recording of results. The device will inform the user if the temperature is normal, low, or high, but does not record these results. If connected to a computer the high, high, low, low, and low functions could contain code that would write a message to the serial monitor using the Serial.println command, for example, if the user has selected 21°C, and the temperature is measured at 20°C, the following code would display a message on the serial monitor "Required Temperature: 21°C – Measured Temperature: 20°C – LOW"

```
Serial.print("Required Temperature: "); // Serial.print prints the message
                                        // contained in "" to the serial monitor
Serial.print(settemp);                 // Serial.print continues the message on
                                        // the same line, and will display the
                                        // variable "settemp"
Serial.print((char)223);               // prints the ° symbol
Serial.print("C – Measured Temperature: "); // prints the message contained in ""
Serial.print(DHT.temperature);         // prints the measured temperature
Serial.print((char)223);               // prints the message contained in ""
Serial.println("C - LOW");             // prints the message contained in "",
                                        // Serial.println will move to the next line
                                        // after the message, ready for the next
                                        // message
```

Similar code could be placed in each function to record “HIGH HIGH”, “HIGH”, “LOW” and “LOW LOW” scenarios. If this was to be recorded rather than monitored by a user the delay could be set longer, maybe 15 minutes instead of just 1 second, and a clock could also be incorporated into the system if required to timestamp the messages.

Although the design exceeds the criteria in the design brief many enhancements could be made to accomplish a more complex objective, or make the device more useful.

References

Bird, J. (2007) *Electrical and Electronic Principles and Technology*, 3rd Edition, Oxford, Elsevier

Burgess, M. (2003) *C Programming Tutorial*, 4th Edition, Free Software Foundation

<http://www.hse.gov.uk/temperature/law.htm> 23/02/19

<https://www.energysavingtrust.org.uk/home-energy-efficiency/thermostats-and-controls>
23/02/19

<https://forum.arduino.cc/index.php?topic=206852.0> 02/03/19

Bibliography

Bird, J. (2007) *Electrical and Electronic Principles and Technology*, 3rd Edition, Oxford, Elsevier

Kernighan, B. and Ritchie, D. (1988) *The C Programming Language*, 2nd Edition, New Jersey, Prentice Hall

Burgess, M. (2003) *C Programming Tutorial*, 4th Edition, Free Software Foundation

<https://www.arduino.cc/reference/en/language/structure/control-structure/else/> 18/02/19

<http://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-an-arduino/>
18/02/19

<http://www.circuitbasics.com/how-to-set-up-an-lcd-display-on-an-arduino/> 18/2/19

<https://electronics.stackexchange.com/questions/67103/can-i-use-the-analog-pins-on-the-arduino-for-my-project-as-digital> 20/02/19

<https://dronebotworkshop.com/using-ir-remote-controls-with-arduino/> 20/02/19

<http://www.hse.gov.uk/temperature/law.htm> 23/02/19

<https://www.energysavingtrust.org.uk/home-energy-efficiency/thermostats-and-controls>
23/02/19

<https://www.pgi.com/blog/2016/05/the-optimal-office-temperature-for-workplace-productivity/>
23/02/19

<https://programmingelectronics.com/tutorial-14-5-switch-case-statement-old-version/>
24/02/19

<https://study.com/academy/lesson/flowchart-symbols-in-programming-definition-functions-examples.html> 24/02/19

<https://www.programiz.com/article/flowchart-programming> 24/02/19

<https://forum.arduino.cc/index.php?topic=206852.0> 02/03/19

Appendices;

Appendix 1 – Code & Comments (follows)



Appendix 1 - Shaun
Harrison - Code & Co

Appendix 2 – Flow Chart (follows)



Appendix 2 - Flow
Chart Page 1 - Setup.i



Appendix 2 - Flow
Chart Page 2 - Main.p



settemp function flow
chart.pdf



highhigh function
flow chart.pdf



high function flow
chart.pdf



normal function flow
chart.pdf



low function flow
chart.pdf



lowlow function flow
chart.pdf

Appendix 3 – Wiring Diagram



Appendix 3 - Wiring
Diagram - User Adjus

Appendix 4 – A4 'Quick Start Guide (follows)



Appendix 4 - User
Manual.pdf

Appendix 5 – Indicative control panel if built



Appendix 5 -
Indicative Panel Cover

Data Sheets

Appendix 6 – DHT11 Sensor Data Sheet

Appendix 7 – 16*2 LCD Data Sheet

Appendix 8 – RGB LED Data Sheet

Appendix 9 – UNO Data Sheet



Appendix 6 -
Datasheet - DHT11.pc



Appendix 7 -
Datasheet - LCD1602/



Appendix 8 -
Datasheet - LED RGB.i



Appendix 9 -
Datasheet - UNO.pdf

Shaun Harrison – Appendix 1 - Code

```
// Temperature monitoring/alarm program using DHT11 temperature and humidity sensor
// Allows for user to set their desired temperature
// Created by Shaun Harrison, Embedded Systems, Year 1 HNC Electronic and Electrical
// Engineering
// Pembrokeshire College

// Pins in use
// 4 red
// 13 green
// 5 blue
// 3 buzzer PWM
// 2 DHT11 Input
// A0 Potentiometer to adjust temperature setting
// 7, 8, 9, 10, 11, 12 - Pins for LCD display - Data

#include <dht.h> // include dht (temp sensor) library
#include <LiquidCrystal.h> // include LCD display library

LiquidCrystal lcd(7,8,9,10,11,12); // LCD display data pins

int redLED = 4; // set pin names and define pin number
int greenLED = 13; // Allows to change pin number here
int blueLED = 5; // And automatically change throughout the code
int buzz = 3;
int tempinput = A0;

int settemp; // creates an integer for use later

dht DHT; // temp sensor referred to as DHT

#define DHT11_PIN 2 // temp sensor on pin 2

void TempSet() // Sets a function that can be called repeatedly
{
  lcd.setCursor(0,1); // set the cursor position position 0, row 1
  lcd.print("Current: "); // prints text "Current "
  settemp = map(analogRead(tempinput),60,1000,16,25); //variable set temp is mapped
                                                    from analog input range (60-
                                                    1000, recorded min & max) to
                                                    output range (16-25, desired
                                                    range)

  lcd.print(settemp); // display the value of settemp on the LCD display
  lcd.print((char)223); // degrees symbol
  lcd.print("C"); // tells degrees C
  delay(500); // wait 1/2 second before continuing
} // end of function
```

```
void highhigh() // sets a function to run if temperature is high high
{digitalWrite(redLED,HIGH); // turn on red LED
 analogWrite(buzz, 60); // turn on buzzer, analogue write to reduce voltage/volume
 lcd.setCursor(0,0); //sets top row 1st character
 lcd.print("Temp: "); // print temp on LCD display
 lcd.print(DHT.temperature); // insert temp
 lcd.print((char)223); // degrees symbol
 lcd.print("C"); // tells degrees C
 lcd.setCursor(0,1); // set second row
 lcd.print("Humidity: "); // print humidity on LCD display
 lcd.print(DHT.humidity); // print humidity value
 lcd.print("%"); // print %
 delay(5000); // wait 5 seconds
 digitalWrite(redLED,LOW); // turn off LED
 digitalWrite(buzz,LOW); // turn off Buzzer
}
```

```
void high() // sets a function to run if temperature is high
{digitalWrite(redLED,HIGH); // turn on red LED
 lcd.setCursor(0,0); //print temp & humidity details to LCD as above
 lcd.print("Temp: ");
 lcd.print(DHT.temperature);
 lcd.print((char)223);
 lcd.print("C");
 lcd.setCursor(0,1);
 lcd.print("Humidity: ");
 lcd.print(DHT.humidity);
 lcd.print("%");
 delay(1000); // wait 1 second
 digitalWrite(redLED,LOW); // turn off LED
}
```

```
void lowlow () // sets a function to run if temperature is low low
{digitalWrite(blueLED,HIGH); // turn on blue LED
 analogWrite(buzz, 60); // turn on buzzer, analogue write to reduce voltage/volume
 lcd.setCursor(0,0); //print temp & humidity details to LCD as above
 lcd.print("Temp: ");
 lcd.print(DHT.temperature);
 lcd.print((char)223);
 lcd.print("C");
 lcd.setCursor(0,1);
 lcd.print("Humidity: ");
 lcd.print(DHT.humidity);
 lcd.print("%");
 delay(5000); // wait 5 seconds
 digitalWrite(blueLED,LOW); // turn off LED
 digitalWrite(buzz,LOW); // turn off buzzer
}
```

```
void low() // sets a function to run if temperature is low
{digitalWrite(blueLED,HIGH); // turn on blue LED
```

```
lcd.setCursor(0,0); //print temp & humidity details to LCD as above
lcd.print("Temp: ");
lcd.print(DHT.temperature);
lcd.print((char)223);
lcd.print("C");
lcd.setCursor(0,1);
lcd.print("Humidity: ");
lcd.print(DHT.humidity);
lcd.print("%");
delay(1000); // wait 1 second
digitalWrite(blueLED,LOW); // turn off blue LED
}
```

```
void normal() // sets a function to run if temperature is normal
{digitalWrite(greenLED,HIGH); // turn on green LED
lcd.setCursor(0,0); //print temp & humidity details to LCD as above
lcd.print("Temp: ");
lcd.print(DHT.temperature);
lcd.print((char)223);
lcd.print("C");
lcd.setCursor(0,1);
lcd.print("Humidity: ");
lcd.print(DHT.humidity);
lcd.print("%");
delay(1000); // wait 1 second
digitalWrite(greenLED,LOW); // turn off LED
}
```

```
void setup() // setup code before entering the main loop
{
  pinMode(redLED,OUTPUT); // Red LED, Pin is output
  pinMode(greenLED, OUTPUT); // Green LED
  pinMode(blueLED, OUTPUT); // Blue LED
  pinMode(buzz, OUTPUT); // Buzzer, PWM
  pinMode(tempinput, INPUT_PULLUP); // pin for temperature input defined as an input
}
```

// Set up code introduces the program on the LCD display, and allows the user to set a temperature

```
lcd.begin(16,2); // LCD is 16 characters b 2 rows
lcd.setCursor(0,0); // set cursor position 0 row 0
lcd.print("SH1 Temp Monitor"); // program name
lcd.setCursor(0,1); // set cursor
lcd.print("Shaun Harrison."); // my name
delay(3000); // wait 3000 mS (3 Seconds)
lcd.clear(); // clear the LCD Display
lcd.setCursor(0,0); // set the cursor position 0, row 0
lcd.print("Set Temp 16-25"); // print message to LCD screen
lcd.print((char)223); // degrees symbol
lcd.print("C"); // tells degrees C
TempSet(); // runs the function defined earlier as TempSet, allows 0.5 second to set
```

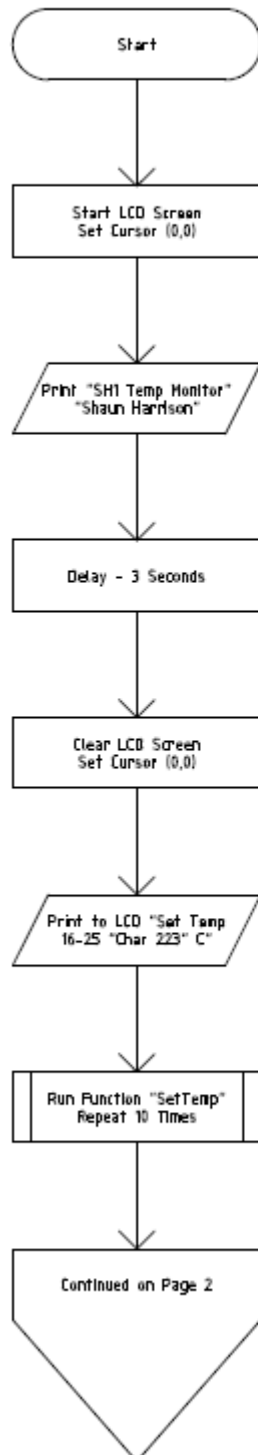
```
TempSet(); // repeat - allows additional 0.5 second - 1.0 second to set
TempSet(); // 1.5 second to set
TempSet(); // 2.0 second to set
TempSet(); // 2.5 second to set
TempSet(); // 3.0 second to set
TempSet(); // 3.5 second to set
TempSet(); // 4.0 second to set
TempSet(); // 4.5 second to set
TempSet(); // 5.0 second to set

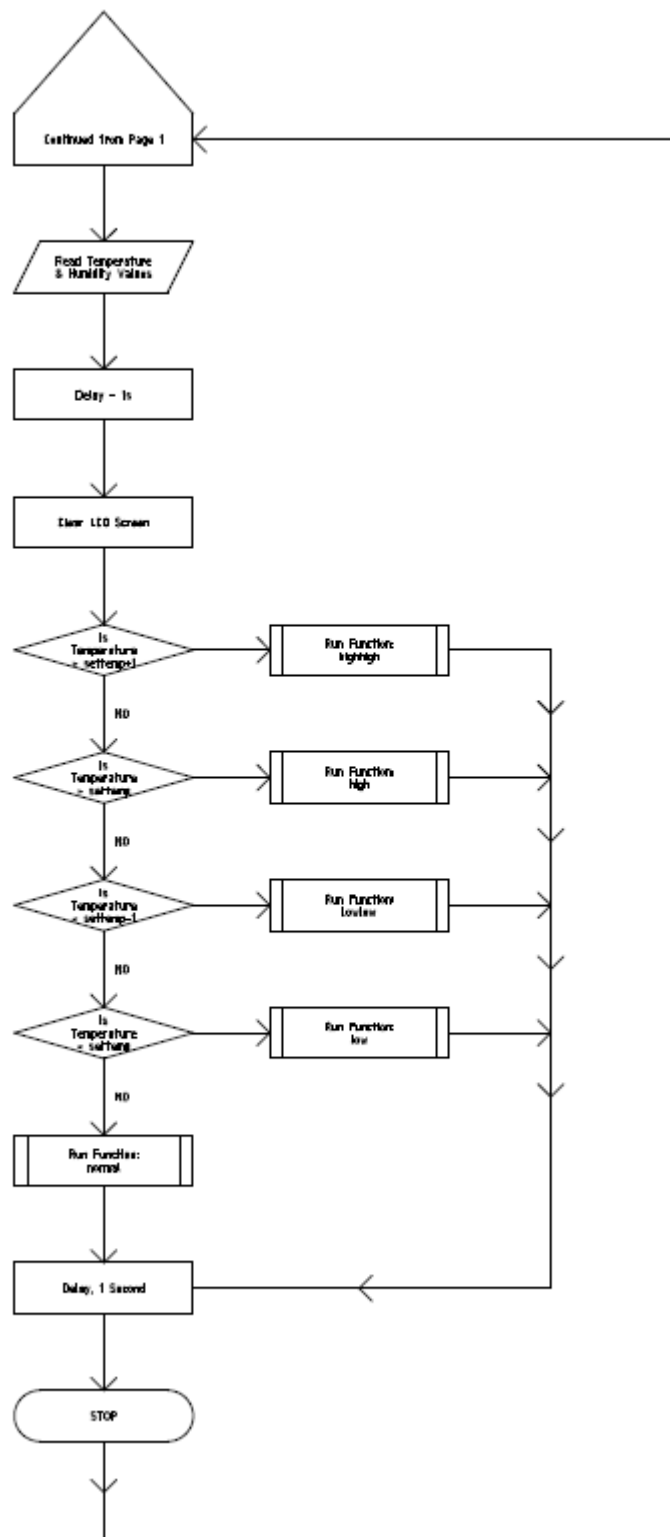
} // Exit set up

void loop() // program begins
{
  int chk = DHT.read11(DHT11_PIN); // read DHT temp/humidy values
  delay(1000); // delay 1000 mS
  lcd.clear(); // clears the LED display
  if(DHT.temperature > settemp+1) // if temp is 2deg or more over settemp then
  {
    highhigh(); // runs the highhigh function defined earlier
  }
  else if(DHT.temperature > settemp) // otherwise, if 1 deg over settemp
  {
    high(); // runs the high function defined earlier
  }
  else if(DHT.temperature < settemp-1) // otherwise, if temp 2 or more degrees under
                                     settemp
  {
    lowlow(); // runs the lowlow function defined earlier
  }
  else if(DHT.temperature < settemp) // otherwsie if temp 1 degree under settemp
  {
    low(); // runs the low function defined earlier
  }
  else // if no conditions met, do this
  {
    normal(); // runs the normal function defined earlier
  }
  delay(1000); // wait 1 second start again
} // end of main loop
```

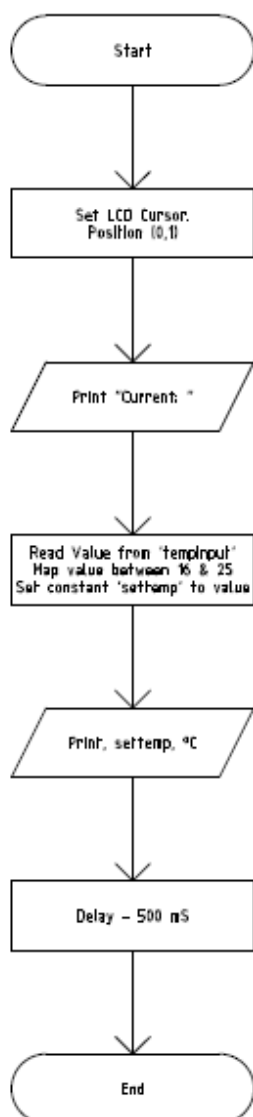
Appendix 2 – Flow Charts

Page 1 of 2 Setup

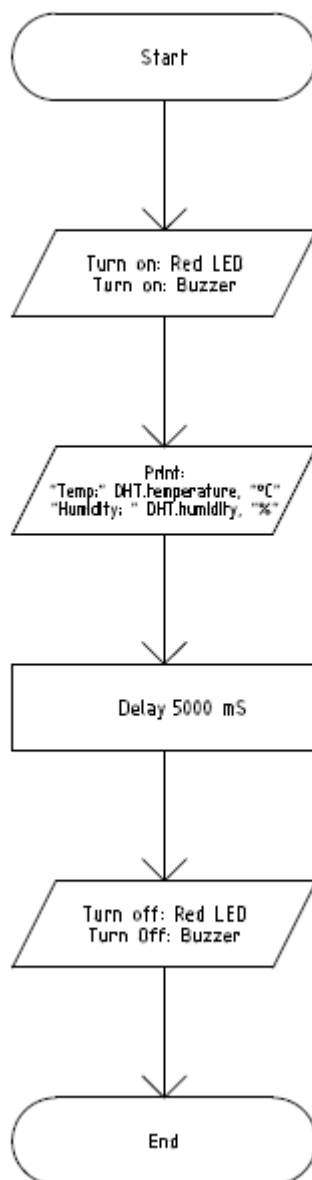




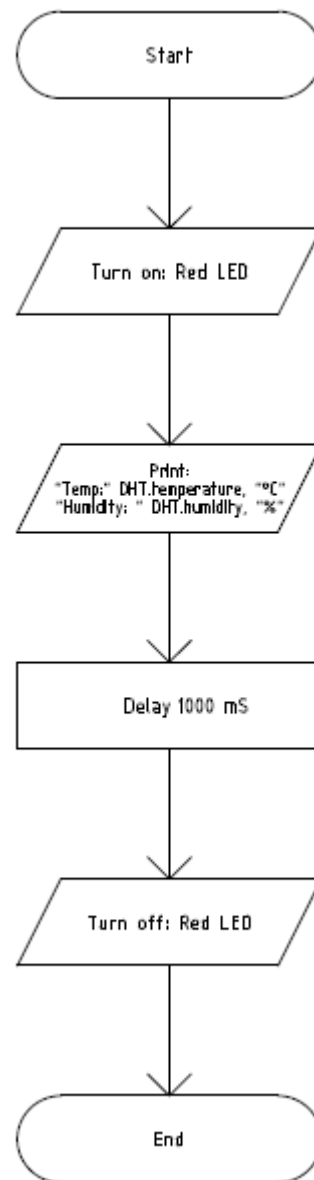
Set Temp Function

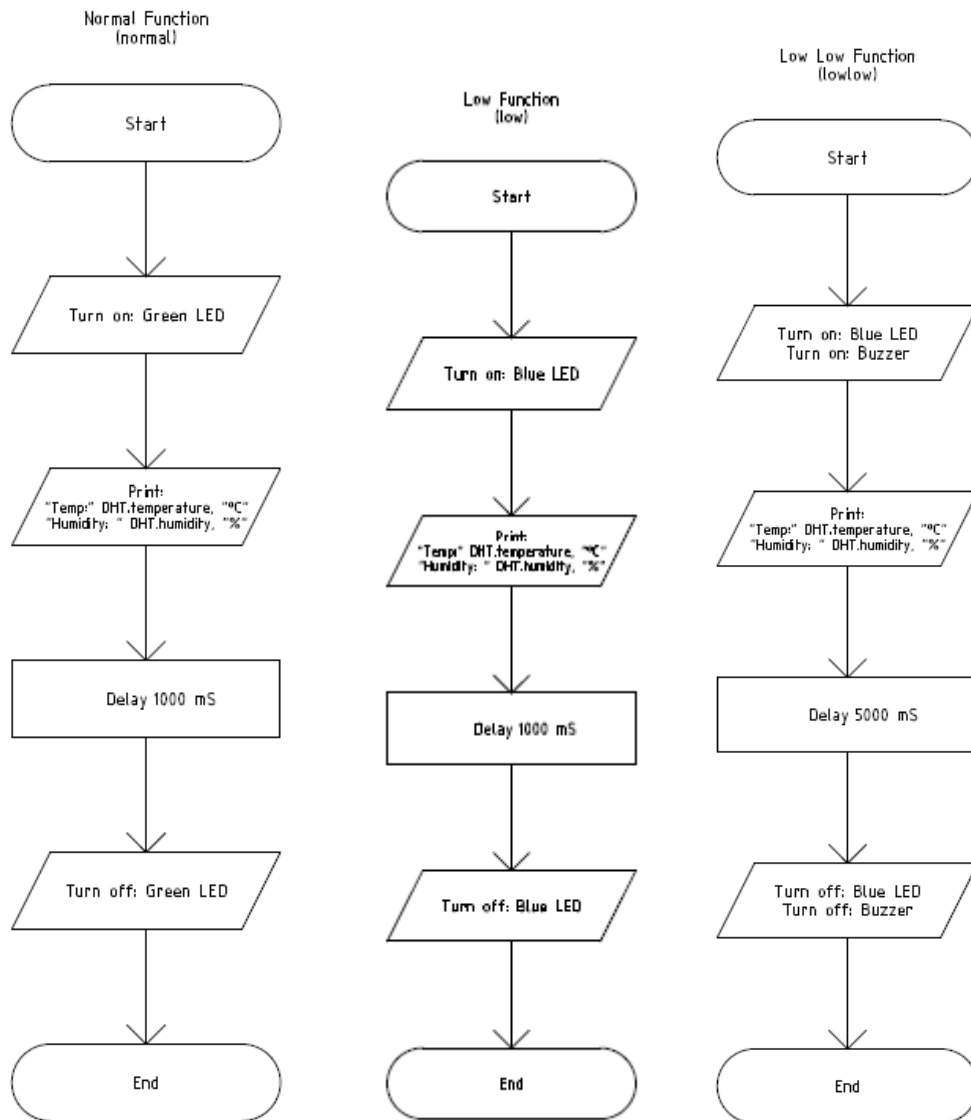


High High Function
(highhigh)



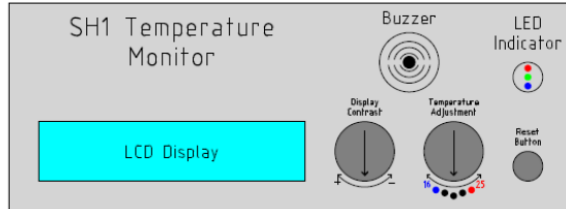
High Function
(high)





Appendix 4 – Quick Start Guide

SH1 Temperature Monitor Quick Start Guide



Overview

The SH1 Temperature Monitor is an electronic device that will alarm the user if the room temperature drops below or rises above the set temperature.

The device can be powered by a USB cable, a 9V battery, or a 9V DC power supply.

A multicoloured LED will allow you to easily determine if the room is at the set temperature, too hot, or too cold. Additionally, the built-in buzzer will alarm you of more significant temperature changes.

Powering up & setting the temperature

The SH1 Temperature Monitor should automatically power up once a power source is connected. You will see a welcome message on the screen before you are asked to set your desired temperature.

Following the welcome message, there is a 5-second window for you to select your desired temperature. Simply turn the temperature control knob anti-clockwise to increase the temperature, or clockwise to decrease the temperature.

After 5 seconds, the monitoring program will start, and further adjustments cannot be made. If you haven't reached your desired temperature, press the reset button to start again.

You can also press the reset button to check the correct temperature is set.

SH1 Temperature Monitor Quick Start Guide

The LCD Display

During normal operation, the LCD display will display the current temperature, and humidity.

You can adjust the contrast of the display with the "Display Contrast" control knob.

LED and Buzzer

The monitor will alert you to one of 5 outcomes with the built-in LED and buzzer.

If the temperature in the room matches the temperature set, the LED will flash green for 1 second, then off for 1 second and repeat.

If the temperature is 1°C above the set temperature, the LED will flash red for a second, then off for one second. If 1°C below, the LED will flash blue for 1 second then off for 1 second and repeat.

If the temperature is more than 1°C above the set temperature, the LED will flash red for 5 seconds, and the buzzer will sound, then off for 1 second. If more than 1°C below, the LED will flash blue for 5 seconds, and the buzzer will sound, then off for 1 second and repeat.

Parameter	Value(s)
Supply Voltage	9V DC
Operating Voltage	5V DC
Temperature Control Range	16-25°C
Operating Temperature Range	10-40°C
Temperature reading increment	1°C
Accuracy	+/- 0.2°C at 25°C