

# Embedded C - Coursework - Final - JSR

*by* Joseph Samuel Roan

---

**Submission date:** 12-Apr-2026 09:27AM (UTC+0100)

**Submission ID:** 279548037

**File name:** 57738\_Joseph\_Samuel\_Roan\_Embedded\_C\_-\_Coursework\_-\_Final\_-\_JSR\_728569\_1560739904.docx  
(818.72K)

**Word count:** 2514

**Character count:** 12512

# ACDA4004 Embedded C Coursework

## Assignment

Joseph Samuel Roan

2411525

17/04/2026

## Abstract

The creation of an Automotive Cooling System Status Simulation has been completed, this has been completed using a circuit diagram and creating a code. The code created was used in conjunction with a simulation software to simulate the code working with a Microcontroller, which had previously been selected. The outcome was a working System, that functioned correctly to the outlined specification.

## Table of Contents

|                                                                 |    |
|-----------------------------------------------------------------|----|
| 1. Introduction.....                                            | 5  |
| 2. Background Research.....                                     | 5  |
| 2.1 What is an Automotive Cooling System Status Simulator?..... | 5  |
| 2.2 What is a PIC Microcontroller?.....                         | 5  |
| 3. Circuit Design.....                                          | 5  |
| 3.1 What components have been used?.....                        | 7  |
| 3.1.1 PIC16F877A Microcontroller.....                           | 7  |
| 3.1.2 Button.....                                               | 7  |
| 3.1.3 Capacitor.....                                            | 8  |
| 3.1.4 Crystal.....                                              | 8  |
| 3.1.5 LED.....                                                  | 8  |
| 3.1.6 LCD.....                                                  | 9  |
| 3.1.7 Motor.....                                                | 9  |
| 3.1.8 Pot.....                                                  | 10 |
| 3.1.9 Resistor.....                                             | 10 |
| 4. Flowchart.....                                               | 11 |
| 5. Screenshots.....                                             | 12 |
| 5.1 Stage 1 – System Ready.....                                 | 13 |
| 5.2 Stage 2 – Engine Started.....                               | 14 |
| 5.3 Stage 3 – Coolant Okay.....                                 | 15 |
| 5.4 Stage 4 – Fan Activated & Motor Running.....                | 16 |
| 5.5 Stage 5 – Alert.....                                        | 17 |
| 5.6 Stage 6 – Shutdown.....                                     | 19 |
| 6. Details of Testing.....                                      | 20 |
| 6.1 CCS C.....                                                  | 20 |
| 6.2 Compile.....                                                | 20 |
| 6.3 Input into proteus.....                                     | 20 |
| 6.4 Run and Testing.....                                        | 20 |
| 7. Conclusion.....                                              | 21 |
| 8. References.....                                              | 22 |
| 9. Appendices.....                                              | 24 |
| 9.1 Full Code.....                                              | 24 |

## List of Figures

|                                                              |    |
|--------------------------------------------------------------|----|
| Figure 1 - Circuit Design .....                              | 6  |
| Figure 2 - PIC16F877A Symbol .....                           | 7  |
| Figure 3 - Button Symbol .....                               | 7  |
| Figure 4 - Capacitor Symbol .....                            | 8  |
| Figure 5 - Crystal Symbol .....                              | 8  |
| Figure 6 - LED Symbol .....                                  | 8  |
| Figure 7 - LCD Symbol .....                                  | 9  |
| Figure 8 - Motor Symbol .....                                | 9  |
| Figure 9 - Pot Symbol .....                                  | 10 |
| Figure 10 - Resistor .....                                   | 10 |
| Figure 11 - Flowchart .....                                  | 11 |
| Figure 12 - Stage 1 'System Ready' .....                     | 13 |
| Figure 13 - Stage 2 'Engine Started' .....                   | 14 |
| Figure 14 - Stage 3 - Coolant Okay .....                     | 15 |
| Figure 15 - Stage 4 'Fan Activated & Motor Running' .....    | 16 |
| Figure 16 - Stage 5 'Alert', with Red LED Blinking on .....  | 17 |
| Figure 17 - Stage 5 'Alert', with Red LED Blinking off ..... | 18 |
| Figure 18 - Stage 6 'Shutdown' .....                         | 19 |

## 1. Introduction

This document details the creation of an Automotive Cooling System Status Simulation, including background research, the circuit schematic, the components used and their purpose, the code used to simulate the system, the Flowchart as well as details of testing. The outcome was a simulation that was operational and could perform the action of an automotive cooling system status. The details of this are covered more extensively below.

## 2. Background Research



### 2.1 What is an Automotive Cooling System Status Simulator?

This system is designed to operate vehicle control logic. This system is important, as they control various functions of a vehicle system [1], which from a safety and operational perspective is key, for the safety of the operators.



### 2.2 What is a PIC Microcontroller?

A Microcontroller is an integrated circuit, that will manage a specific operation within an embedded system [2], they are 'integral' and at the heart of all Embedded Systems, as they are efficient and reliable in managing embedded systems [3], they are widely used as they are compact and cost efficient [4]. The reasoning for the design to be as compact as practicable is for space efficiency [4] as it allows consumers to use more space for other components, or to make the size of the circuit smaller, which can keep production costs down. The 'PIC' stands for 'Peripheral Interface Controller', as this controls peripherals and interfaces within electronic devices [5].



## 3. Circuit Design

The circuit created for the system, is designed to work in conjunction with the code to create a simulation for an Automotive Cooling System Status. This circuit is shown below in Figure 1.

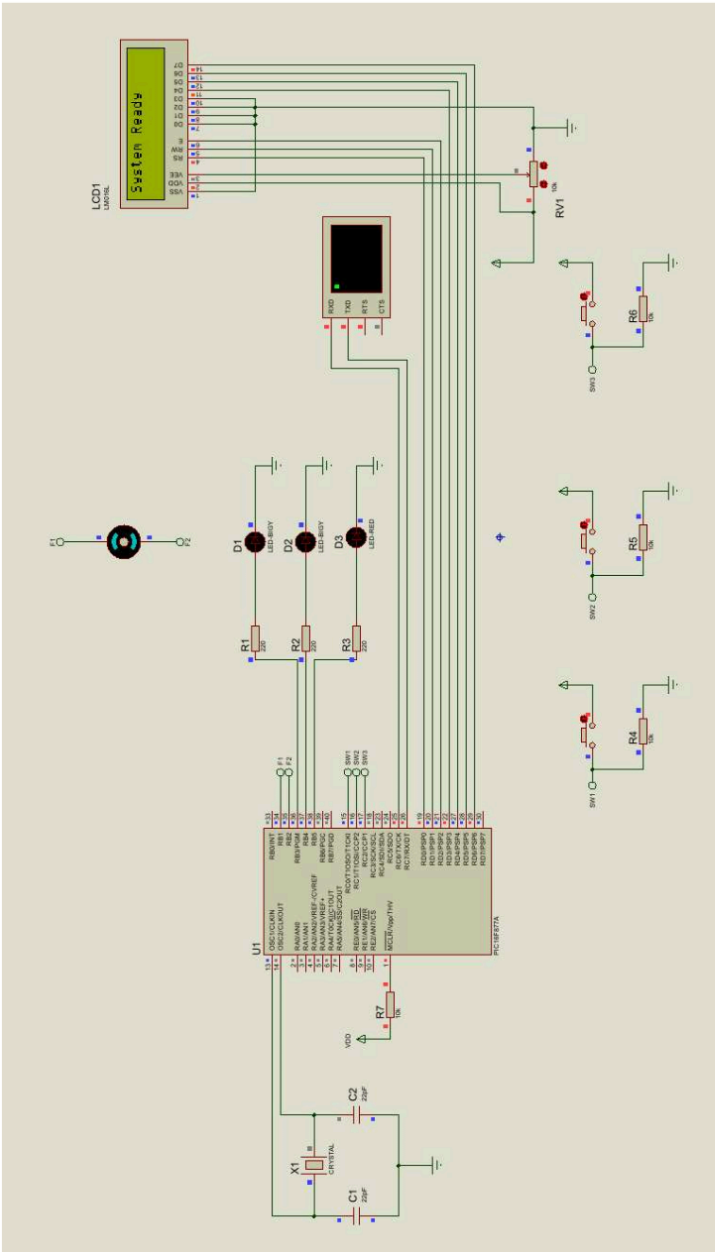


Figure 1 - Circuit Design

3.1 What components have been used?

To create the system, various components were used, their descriptions and what their purpose in the circuit are explained below:

3.1.1 PIC16F877A Microcontroller

This component is a PIC 16 Microcontroller, which is an 8-bit Microcontroller, described as being a balance between capability and stability [6], with the purpose being to act as the brain of the system and to make it work. This is set to 50MH. The symbol used to show this on the circuit is shown below in Figure 2.

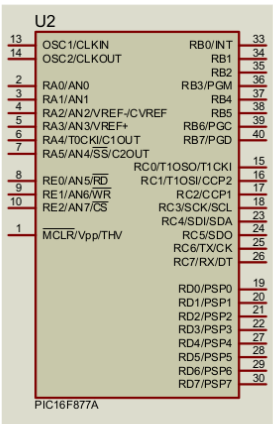


Figure 2 - PIC16F877A Symbol

3.1.2 Button

The purpose and use of the button is determined from the code, in this circuit there are 3 buttons, their purpose is to turn on the system when pressed in a certain order. When pressed they also send a signal to the LCD to display and luminate which stage it is at. The symbol used to show this on the circuit is shown below in Figure 3.



Figure 3 - Button Symbol

### 3.1.3 Capacitor

A capacitor is an electronic component that stores electrical energy over a large period, releasing the electrical energy as required [7]. The purpose of the two capacitors in this circuit is to store and release electrical energy as required, within the crystal component, in the system. The symbol used to show this on the circuit is shown below in Figure 4.



Figure 4 - Capacitor Symbol

### 3.1.4 Crystal

A crystal oscillator is an electronic oscillator circuit, that uses a vibrating crystals mechanical resonance, to create an electrical signal, which will have a precise frequency [8]. The purpose of the crystal oscillator in this circuit is to ensure that the circuit is at the correct frequency to work. The frequency is set to 20MH, the same as the Pic Microcontroller. The symbol used to show this on the circuit is shown below in Figure 5.



Figure 5 - Crystal Symbol

### 3.1.5 LED

Identified in the circuit from D1 to D3, LED is an acronym for 'Light Emitting Diode' [9], which is a 'solid -state lighting source' [9], being used in various different fields of operations. The purpose of an LED is to light up and luminate on command, how and when this occurs is determined by the code the designer creates. The LEDs in this circuit are to luminate when a system is in use, however the 'D3' LED will flash, as a warning before system shutdown, additionally the colour on the LED's are different as D3 is Red, whereas D1 & D2 are yellow. The symbol used to show this on the circuit is shown in Figure 6.



Figure 6 - LED Symbol

3.1.6 LCD

LCD, identified as LCD1 stands for 'Liquid Crystal Display' [10], which is a Flat Panel device that uses liquid crystals to control light [11], using the lights polarisation it shows objects and using ambient lights to luminate the display with the characters selected [11]. The purpose of an LCD within this circuit is to show the operator which stage the Automotive cooling system status simulator is at. The symbol used to show this on the circuit is shown below in Figure 7.

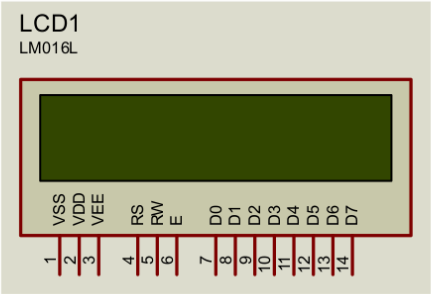


Figure 7 - LCD Symbol

3.1.7 Motor

A motor is a component that converts energy, for instance electrical energy, into mechanical motion [12], which can be used in a system or machine [13]. The purpose of the motor in this circuit is to simulate a radiator fan, within an Automotive Cooing System. The symbol used to show this on the circuit is shown below in Figure 8.

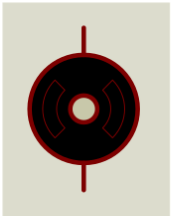


Figure 8 - Motor Symbol

### 3.1.8 Pot

A pot is short for 'Potentiometer' [14], which is a three-terminal variable resistor [15], which adjusts resistance to control the flow of electric current [16]. The pot in the circuit is identified as RV1, the purpose of the pot in this circuit is to control the flow of electric current within the LCD component used in the system. The symbol used to show this on the circuit is shown below in Figure 9.

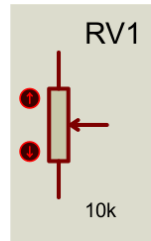


Figure 9 - Pot Symbol

### 3.1.9 Resistor

A Resistor is an electronic component that's primary function is to add resistance to the flow of the electric current, which will regulate the electric current [17], there use in the circuit is to do this. The resistors are identified in the circuit schematic as R1 to R7, using the 'R' to identify the component as a resistor. The symbol used to show this on the circuit is shown below in Figure 10.



Figure 10 - Resistor



#### 4. Flowchart

The flowchart for the Automotive cooling system status simulator, is shown below, refer to Figure 11.

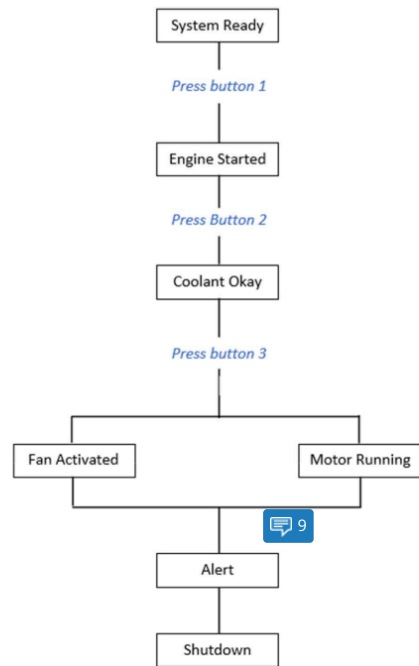


Figure 11 - Flowchart

## 5. Screenshots

There are several different stages within this simulation, the stages are shown below in the screenshots.

### 5.1 Stage 1 – System Ready

Stage 1 is when the system is ready to start, however none of the buttons have been pressed yet, the 'System Ready' is displayed on both the UART and the LCD. The Screenshot of Stage 1 is shown below in Figure 12.

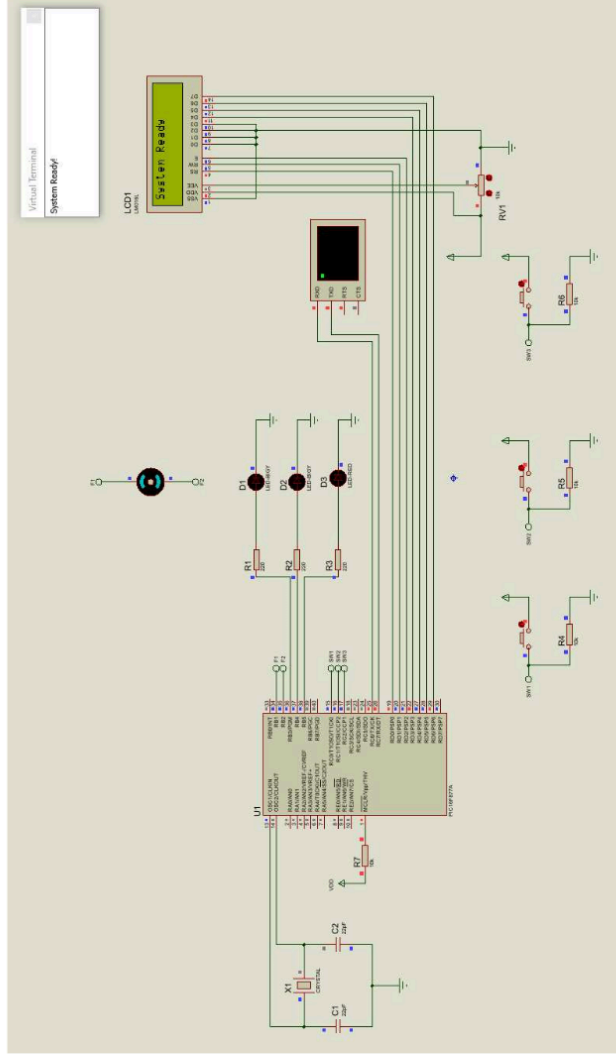


Figure 12 - Stage 1 'System Ready'

5.2 Stage 2 – Engine Started

Stage 2 is when the engine has been started, this is from pressing the first button, the first LED, described as D1 on the Circuit Schematic, is illuminated from this too, to show that it is turned on. The 'Engine Started' is displayed on both the UART and the LCD. The Screenshot of Stage 2 is shown below in Figure 13.

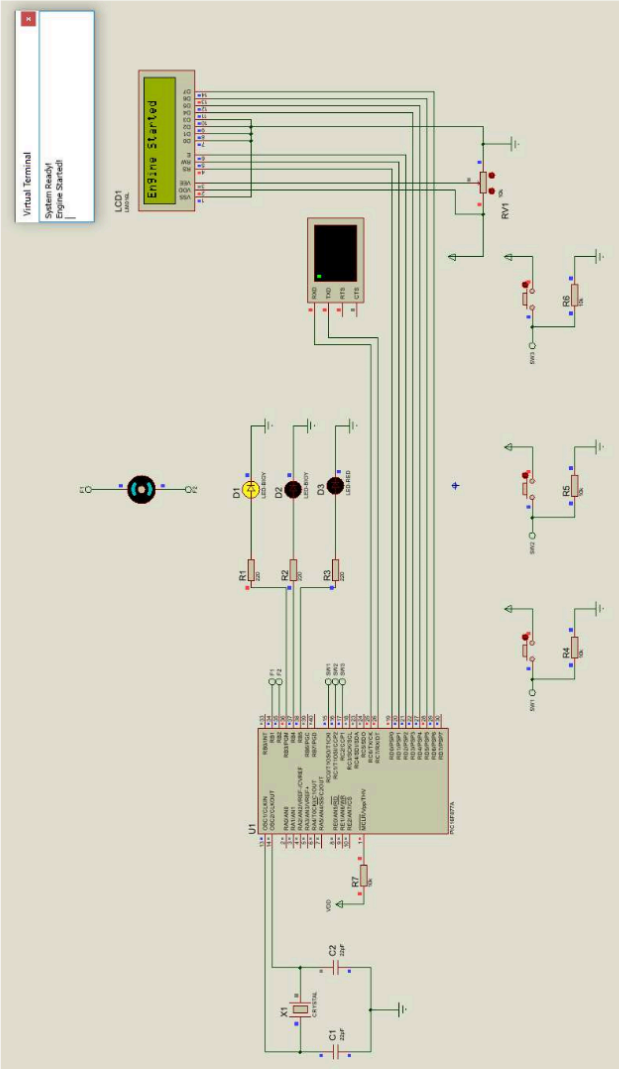


Figure 13 - Stage 2 'Engine Started'

**Stage 3 – Coolant Okay**

Stage 3 is when the fan has been activated, this happens after flowing stage 2, pressing the second button, the second LED, described as D2 on the Circuit Schematic, is illuminated from this too, to show that it is turned on. The 'Coolant Okay' is displayed on both the UART and the LCD. The Screenshot of Stage 3 is shown below in Figure 14.



5.4 Stage 4 – Fan Activated & Motor Running

Stage 4 is when the Fan is activated & the motor starts running, this happens after flowing stage 3, pressing the third button, the third LED, described as D3 on the Circuit Schematic, is illuminated from this too, to show that it is turned on. However, this light is red. 'Fan Activated' and 'Motor Running' is displayed on both the UART and the LCD. The motor will run for a total of 3 seconds. The Screenshot of Stage 4 is shown below in Figure 15.

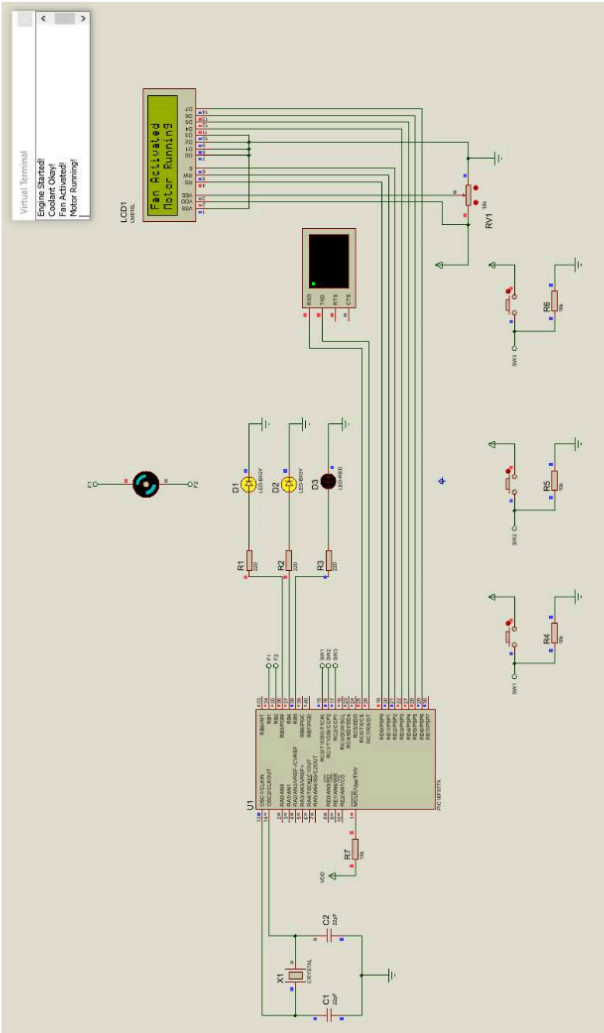


Figure 15 - Stage 4 'Fan Activated & Motor Running'

5.5 Stage 5 – Alert

Stage 5 is when the 'Alert' occurs. This is to alert the operator, that the system is about to shut down. This happens following stage 4, with no actions undertaken by the operator from stage 4. The third LED, described as D3 on the Circuit Schematic, will start to blink (turn on and off), the colour of which will be red. 'Alert' is displayed on both the UART and the LCD. Figure 16 below shows the light when it is blinking on, and Figure 17 below shows the light when the blinking is off.

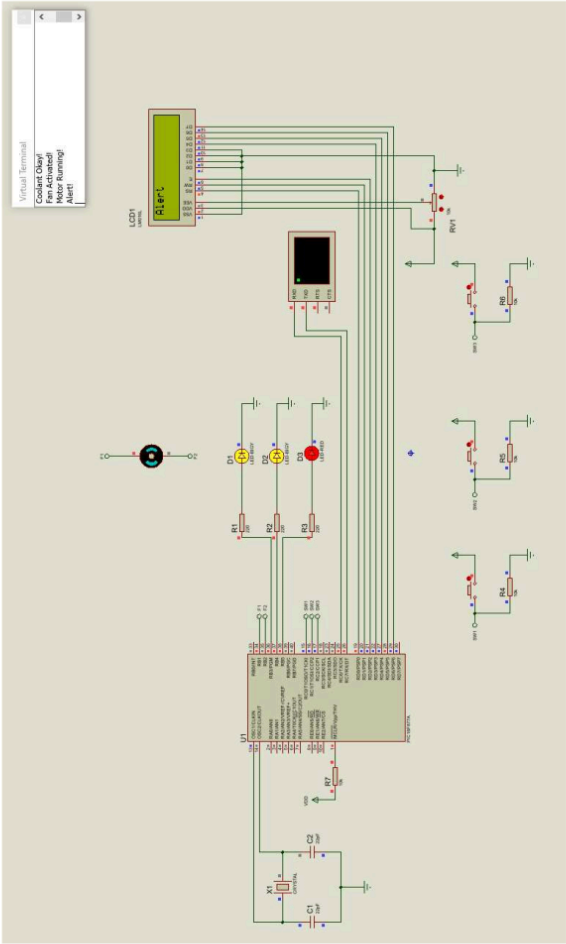


Figure 16 - Stage 5 'Alert', with Red LED Blinking on

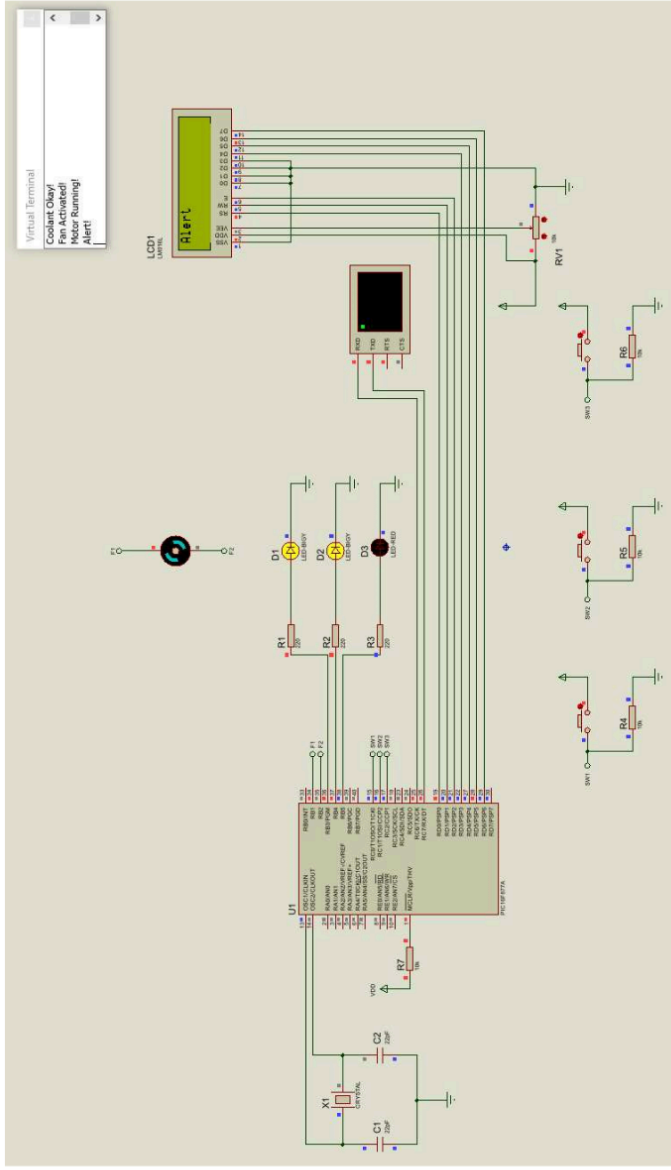


Figure 17- Stage 5 'Alert', with Red LED Blinking off

5.6 Stage 6 – Shutdown

Stage 6 is the final stage; it is when the system is shutdown. This occurs after all the LED'S are turned off and the system is 'shutdown' and turned off. This stage required no actions undertaken by the operator following stage 5. 'Shutdown' is displayed on both the UART and the LCD. The screenshot showing stage 6 is shown below in Figure 18.

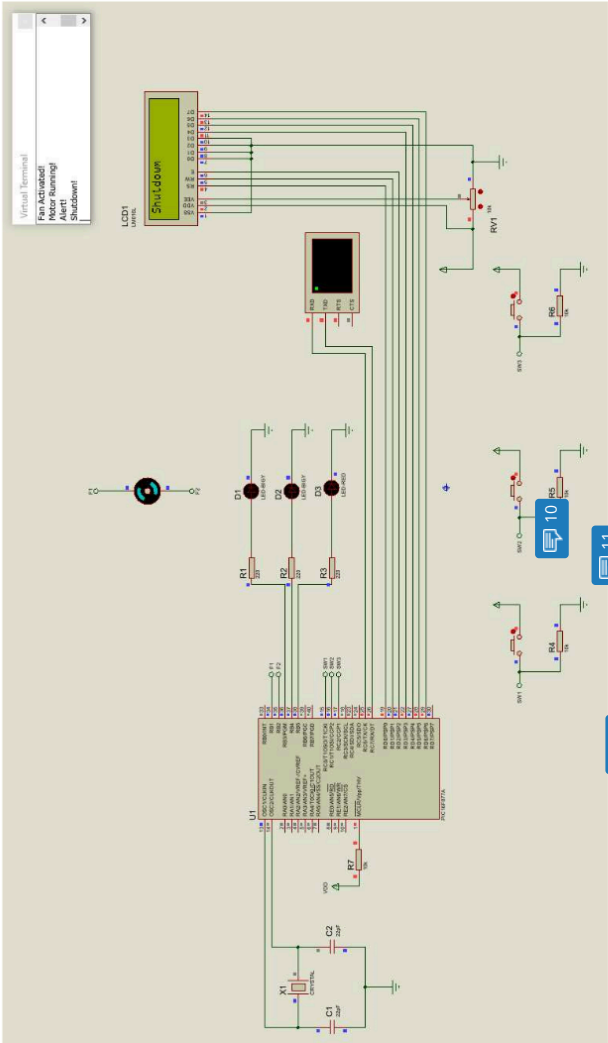


Figure 18 – Stage 6 'Shutdown'

## **6. Details of Testing**

The details of the testing are shown below:

### **6.1 CCS C**

CCS C was used to create the code initially, this was done using 'C' coding language.

### **6.2 Compile**

After the code was created in CCS C, the software has a compile tool, this tool was used to check for errors, if the code had errors or warnings, following this the code was fixed in conjunction with the corrections.

### **6.3 Input into proteus**

After the code was compiled correctly with no errors, the code was inputted into proteus, this was done within the 'source code' section of the software. The files inputted was the 'C' file, the 'Hex' file, and the file for the microcontroller.

### **6.4 Run and Testing**

After the files were inputted into proteus, the software was run, if anything was incorrect, for instance the code gave the wrong instructions then this was amended, the fault finding occurs at this stage heavily and it was the testing stage that took the longest period time, as in some instances not everything was correct, for example the motor starting before the motor command said it should on the brief, or the lights not luminating correctly, or at the correct point. After there was no errors and the system matched the descriptor, that was when the system was finished.

## 7. Conclusion

In conclusion, the outcome is that the system works for its function, meeting the specifications outlined by the coursework descriptor, however with all things there is room for improvement, for instance, to restart the operation the stop button must be pressed before the start button is pressed again (to restart the operation). It would be recommended for future use that this be added to the circuit, to allow an easier job for the operator.



## 8. References

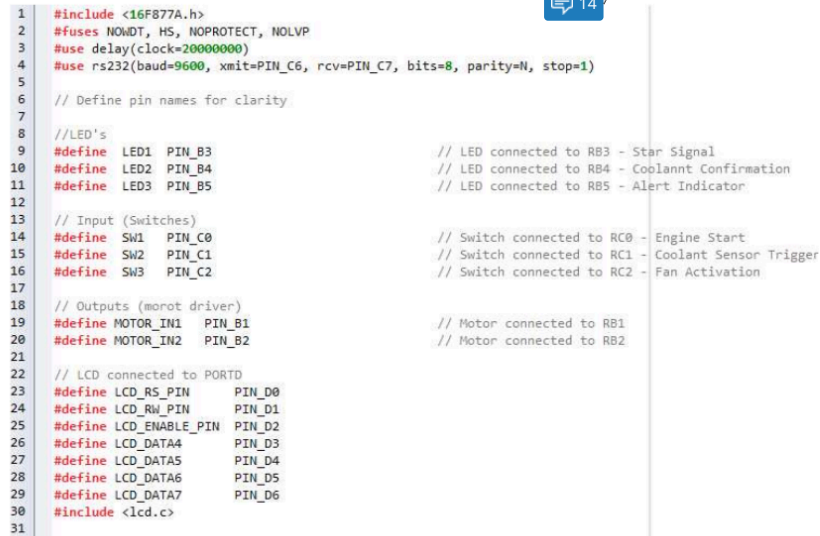
- [1] L. Cope, "How a Vehicle Control System Works," Engineer Fix, 9 November 2025. [Online]. Available: <https://engineerfix.com/how-a-vehicle-control-system-works/>. [Accessed 31 March 2026].
- [2] C. Hashemi-Pour and B. Lutkevich, "What is a microcontroller (MCU)?," TechTarget, 12 August 2024. [Online]. Available: <https://www.techtarget.com/iotagenda/definition/microcontroller>. [Accessed 10 March 2026].
- [3] Elegancespark Innovations, "Why Microcontrollers Are the Heart of Embedded Systems," Elegancespark Innovations, 12 February 2025. [Online]. Available: <https://elegancesparkinnovations.com/why-microcontrollers-are-the-heart-of-embedded-systems/>. [Accessed 10 March 2026].
- [4] P. Agarwal, "Advantages and Disadvantages of Microcontroller," Geeks for Geeks, 23 July 2025. [Online]. Available: <https://www.geeksforgeeks.org/computer-organization-architecture/advantages-and-disadvantages-of-microcontroller/>. [Accessed 10 March 2026].
- [5] "PIC Microcontroller Guide: PIC16F877A, PIC18F, dsPIC33, and More," RAYPCB, [Online]. Available: <https://www.raypcb.com/pic-microcontrollers/>. [Accessed 10 March 2026].
- [6] "Essential PIC Guide: PIC12, PIC16 & PIC18 Deep Dive," Bytewires. [Online]. [Accessed 10 March 2026].
- [7] W. Sarjeant, "Capacitor fundamentals," in *Proceedings of the 19th Electrical Electronics Insulation Conference*, Chicago , 1989.
- [8] R. Gobato, M. R. R. Gobato and A. Heidari, "Rhodochrosite as Crystal Oscillator," *American Journal of Biomedical Science & Research*, vol. 3, no. 2, p. 187, 2019.
- [9] M.-H. Chang, D. Das , P. V. Varde and M. Pecht, "Light emitting diodes reliability review," *Microelectronics Reliability*, vol. 52, no. 5, pp. 762-782, 2012.
- [10] R. H. Chen, "Preface," in *Liquid Crystal Displays: Fundamental Physics and Technology*, John Wiley & Sons, 2011.
- [11] A. Sinha, "What is LCD?," Electronicsforu, 27 August 2024. [Online]. Available: <https://www.electronicsforu.com/technology-trends/learn-electronics/lcd-liquid-crystal-display-basics>. [Accessed 17 March 2026].
- [12] A. Hughes and B. Drury, "Introduction," in *Electric Motors and Drives: Fundamentals, Types and Applications*, Newnes, 2019, p. 1.
- [13] N. Sclater and N. P. Chironis, *Mechanisms and Mechanical Devices Sourcebook*, New York: Mcgraw-hill, 2001.
- [14] M. Gasperi and P. Hurbain, "Potentiometer Sensors," in *Extreme NXT: Extending the LEGO MINDSTORMS NXT to the Next Level*, Apress, 2009.

- [15] C. May, "Non-linear Resistors," in *Passive Circuit Analysis with LTspice*, 2020, pp. 101-141.
- [16] Y. Guoqing, S. Zhanyou and A. Guochen, "Application of Digitally Controlled Potentiometers in the Intelligent Sensor's Measure and Control System," in *8th International Conference on Electronic Measurement and Instruments*, Xi'an, 2007.
- [17] P. L. G. B. C. S. N. A. Michael Pecht, "Passive Components," in *Circuits, Signals, and Speech and Image Processing*, CRC Press, 2006.

## 9. Appendices

### 9.1 Full Code

The code for the Automotive Cooling System Status Simulator, was created on CCS C, compiled, and extracted to proteus. Using screenshots from the CCS C software, the full code for the Automotive Cooling System Status Simulator is shown below:



```
1  #include <16F877A.h>
2  #fuses NOWDT, HS, NOPROTECT, NOLVP
3  #use delay(clock=2000000)
4  #use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7, bits=8, parity=N, stop=1)
5
6  // Define pin names for clarity
7
8  //LED's
9  #define LED1 PIN_B3           // LED connected to RB3 - Star Signal
10 #define LED2 PIN_B4           // LED connected to RB4 - Coolant Confirmation
11 #define LED3 PIN_B5           // LED connected to RB5 - Alert Indicator
12
13 // Input (Switches)
14 #define SW1 PIN_C0             // Switch connected to RC0 - Engine Start
15 #define SW2 PIN_C1             // Switch connected to RC1 - Coolant Sensor Trigger
16 #define SW3 PIN_C2             // Switch connected to RC2 - Fan Activation
17
18 // Outputs (motor driver)
19 #define MOTOR_IN1 PIN_B1       // Motor connected to RB1
20 #define MOTOR_IN2 PIN_B2       // Motor connected to RB2
21
22 // LCD connected to PORTD
23 #define LCD_RS_PIN PIN_D0
24 #define LCD_RW_PIN PIN_D1
25 #define LCD_ENABLE_PIN PIN_D2
26 #define LCD_DATA4 PIN_D3
27 #define LCD_DATA5 PIN_D4
28 #define LCD_DATA6 PIN_D5
29 #define LCD_DATA7 PIN_D6
30 #include <lcd.c>
31
```

```

33 void main() {
34
35     set_tris_b(0b1111001);
36
37     // All RC pins as inputs (SW1-SW3 use RC0-RC2)
38     set_tris_c(0b1111111);
39
40     // Initialize motor outputs LOW (motor off)
41     output_low(MOTOR_IN1);
42     output_low(MOTOR_IN2);
43
44     set_tris_d(0x00);           // Set PORTD as output
45     lcd_init();
46
47     lcd_putc("\fSystem Ready"); // Display System Ready
48     printf("System Ready!\r\n"); // Display System Ready
49
50     while(!input(SW1));         // Start Signal
51     delay_ms(50);
52
53     lcd_putc("\fEngine Started"); // Display Engine Started
54     printf("Engine Started!\r\n"); // Display Engine Started
55     output_high(PIN_B3);         // Turn on the LED
56     delay_ms(1500);
57
58     while(!input(SW2));         // Coolant Confirmation
59     output_high(PIN_B4);         // Turn on the LED
60     delay_ms(50);
61
62     lcd_putc("\fCoolant Okay"); // Display Fan Activated
63     printf("Coolant Okay!\r\n"); // Display Fan Activated
64     delay_ms(1500);
65

```

```

66 while(!input(SW3)); // Blink Alert LED
67 delay_ms(50);
68 lcd_putc("\fFan Activated\nMotor Running"); // Display Fan Activated
69 printf("Fan Activated!\r\n"); // Display Fan Activated
70 printf("Motor Running!\r\n"); // Display Motor Running
71 output_high(MOTOR_IN1); // Turn on the Motor
72 output_low(MOTOR_IN2); // Turn on the Motor
73 delay_ms(3000); // Wait for 3000 milliseconds (3 Seconds)
74
75 printf("Alert!\r\n"); // Display Alert
76 lcd_putc("\fAlert"); // Display Alert
77 output_high(PIN_B5); // Turn on the LED
78 delay_ms(100); // Wait for 100 milliseconds
79 output_low(PIN_B5); // Turn off the LED
80 delay_ms(100); // Wait for 100 milliseconds
81 output_high(PIN_B5); // Turn on the LED
82 delay_ms(100); // Wait for 100 milliseconds
83 output_low(PIN_B5); // Turn off the LED
84 delay_ms(100); // Wait for 100 milliseconds
85 output_high(PIN_B5); // Turn on the LED
86 delay_ms(100); // Wait for 100 milliseconds
87 lcd_putc("\fShutdown"); // Display Alert
88 printf("Shutdown!\r\n"); // Display Alert
89
90 output_low(PIN_B3); // Turn off the LED
91 output_low(PIN_B4); // Turn off the LED
92 output_low(PIN_B5); // Turn off the LED
93 output_low(MOTOR_IN1); // Turn off the Motor
94 output_low(MOTOR_IN2); // Turn off the Motor
95
96 while(TRUE) {
97
98 }
99
100

```

# Embedded C - Coursework - Final - JSR

## GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

74/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5



**Comment 1** | Engagement with Literature(10)

The introduction briefly explains the vehicle cooling system.



**Comment 2** | Engagement with Literature(10)

The report should critically explain the concepts of Automatic Cooling System by referring different journals and books.



**Comment 3** | Knowledge & Understanding(20)

It can be observed that features such as compatibility and lower cost are highlighted in the image. However, it is also important to compare factors like performance, speed, memory, and the advantage of having more selectable pins.

PAGE 6

PAGE 7



**Comment 4** | Knowledge & Understanding(20)

It's helpful to include the components required for the design, but you could also briefly explain which I/O interfaces are used.

PAGE 8



## Comment 5 | Knowledge & Understanding(20)

Well done, the sequence of operation is explained.

PAGE 9

---



## Comment 6 | Knowledge & Understanding(20)

What mode is used in LCD? 4-bit or 8-bit? and why should be included.

PAGE 10

---



## Comment 7 | Knowledge & Understanding(20)

The report does not explain the purpose or use of the UART, so including this would make it better.



## Comment 8 | Knowledge & Understanding(20)

The report explains how the system works, but it could be improved by clarifying how the timing delays are configured in the design.

PAGE 11

---



## Comment 9 | Cognitive & Intellectual (25)

The flowchart is neat and covers all decision points in the task, which is appreciable.

PAGE 12

---

PAGE 13

---

PAGE 14

---

PAGE 15

---

PAGE 16

---

PAGE 17

---

PAGE 18

---

PAGE 19

---



## Comment 10

no comment



## Comment 11 | Practical Skills (30)

**Practical observations:** The design works as intended, with no additional innovation observed.

The sequence follows a logical pattern, and all I/Os function as expected. The circuit is clearly presented with proper labelling and accurate values, and it has been implemented effectively.



**Comment 12** | Cognitive & Intellectual (25)

The justification for selecting certain components over others is addressed in some aspects, but not comprehensively.

PAGE 20

---

PAGE 21

---



**Comment 13** | Cognitive & Intellectual (25)

The limitations and further recommendations are appropriately addressed.

PAGE 22

---

PAGE 23

---

PAGE 24

---



**Comment 14** | Transferable Skills (15)

Explaining certain parts of the program is also necessary.

PAGE 25

---

PAGE 26

---

GRADING FORM: EMBEDDED C

JOSEPH SAMUEL ROAN

74

## ENGAGEMENT WITH LITERATURE(10)

*Research supports embedded automotive cooling logic and PIC-based simulation. Uses scholarly/technical sources and correct referencing.*

 Good range of relevant sources.

**7.5**

## KNOWLEDGE & UNDERSTANDING(20)

*Demonstrates knowledge of sequencing logic, timers/delays, I/O interfacing, LCD 4-bit mode, UART monitoring and real-world cooling workflow.*

 Satisfactory understanding; basic explanations provided but limited depth/clarity in some areas.

**12.5**

## COGNITIVE & INTELLECTUAL (25)

*Evaluates design approach and makes justified decisions using logical reasoning. Demonstrates structured thinking and reflective judgement.*

 Good analytical reasoning; justifies most choices; some evaluation of limitations.

**18**

## PRACTICAL SKILLS (30)

*Implements and tests a PIC16F877A Proteus simulation meeting : switch sequencing, LED states, fan motor control, alert blink, LCD and UART messaging.*

 Mostly functional system; Core requirements met; good integration and testing evidence.

**25**

## TRANSFERABLE SKILLS (15)

*Communicates work clearly and professionally through a structured report and well-documented code. Submits required files and evidence reliably.*



Well-structured report; clear communication; good code readability and presentation.

**11**