Electrical Water Heater Embedded System Using AVR ATMEGA32

Introduction

This presentation focuses on the specialized software developed for embedding within an AVR ATMEGA32 microcontroller. The primary purpose of this software is to effectively regulate and control the operations of an electric water heater, ensuring optimal performance and efficiency.

System Overview

The software serves as the central interface between the AVR ATMEGA32 microcontroller (including ADC and TWI/I2C peripherals) and various hardware components such as the LM35 sensor, LCD, 7 Segments display, and EEPROM. It is intricately designed to manage the operational logic of an electric water heater.

System Processes

- **Temperature Regulation:** Users can set a desired temperature for the water. The system actively cools the water if it's above the set point and heats it if below, maintaining a range within 5 degrees of the desired temperature.
- **Display and Monitoring:** Both the LCD and the 7 Segments display continuously show the current water temperature.
- **Memory and Data Handling:** The microcontroller (MC) records the current temperature every 300 milliseconds. This allows for the immediate display of the last recorded temperature on the LCD when the heater is restarted.
- **Communication Protocol:** Utilizes the TWI (I2C) communication protocol for efficient data transmission to and from the EEPROM, ensuring seamless operation and control.

Hardware Used

1. LM35 Temp Sensor

LM35 is a precession integrated circuit temperature sensor whose output voltage varies based on the temperature around it. It can be used to measure temperature anywhere between -55°C to 150°C. It doesn't require any calibration. LM35 provides analog voltage that is linearly proportional to the Celsius temperature. Each increase in one degree Celsius cause increase of 10mV in output voltage. Follow the next equation to calculate the temperature:

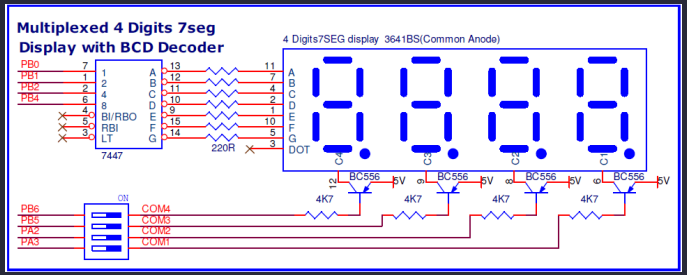
- Temp(0C) = Output Voltage (mV) / 10

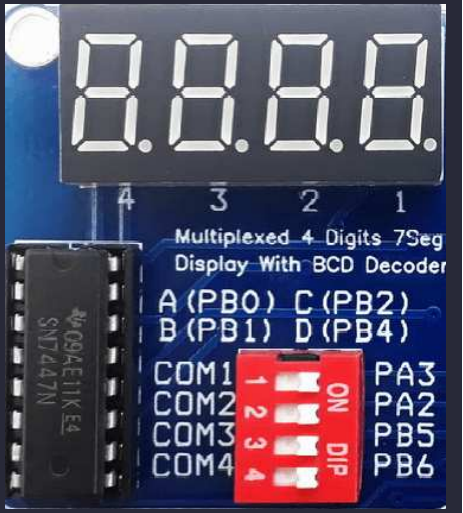
Analog Input	microcontroller Pin	Arduino Pin
Variable resistor (VR1)	PA0	A0
LDR		
Variable resistor (VR2)	PA1	A1
Temperature sensor (LM35)		

Hardware Used

2. Seven Segment Display

7segment display is used to indicate numerical data. It can display digits from 0 to 9. 7segment display is very popular and has many applications. Eta32 includes multiplexed common anode four digits 7segment display in addition to 7447 BCD decoder to simplify firmware.

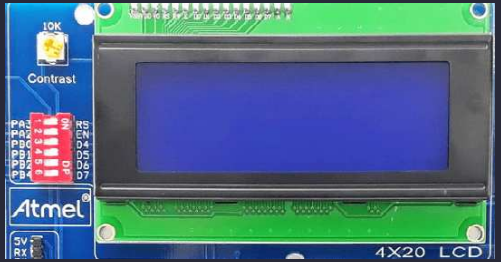




Hardware Used

3. LCD Panel 4x20

Eta32 kit contains 4X20 character LCD. Four lines of characters each has 20 characters line length is quite sufficient to show quite amount of information. LCD is configured in 4 bit mode and connected to microcontroller via DIP switch as shown in table:

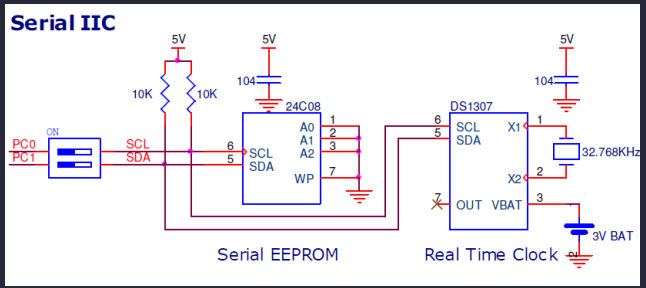


LCD pin number	LCD pin symbol	Microcontroller pin	Arduino Pin
4	RS	PA3	A3
6	EN	PA2	A2
11	D4	PB0	0
12	D5	PB1	1
13	D6	PB2	2
14	D7	PB4	4

Hardware Used

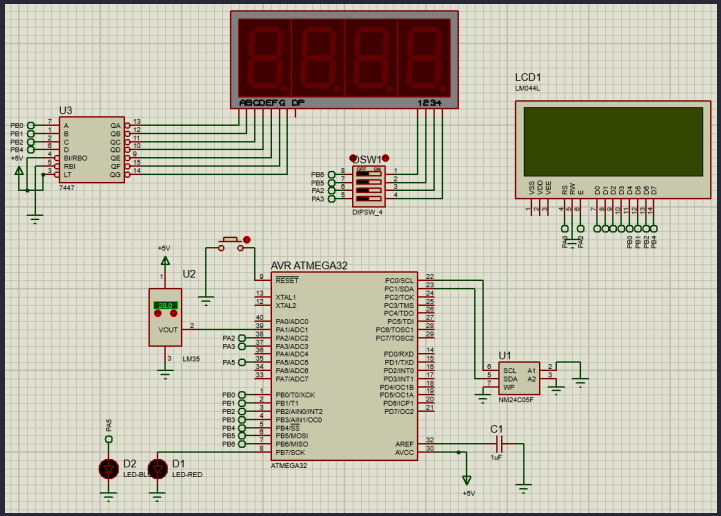
4. EEPROM

The AT24C08 memory provides 8192 bits of serial electrically erasable and programmable read only memory (EEPROM) organized as 1024 bytes of 8 bit. The device address byte of the chip (AT24C08) is &HA0

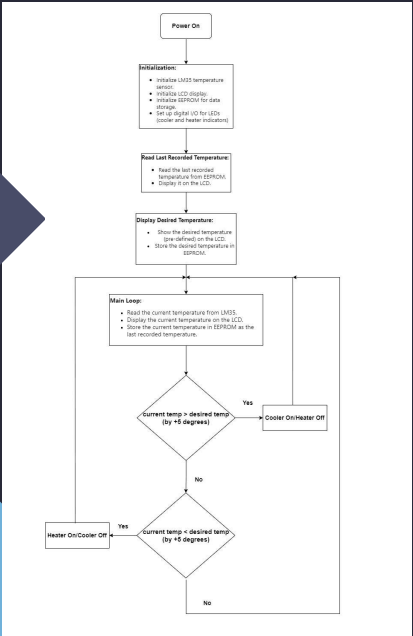


Proteus Schematic

This Schematic and Connection aims to mimic the ETA32 kit's DIO.

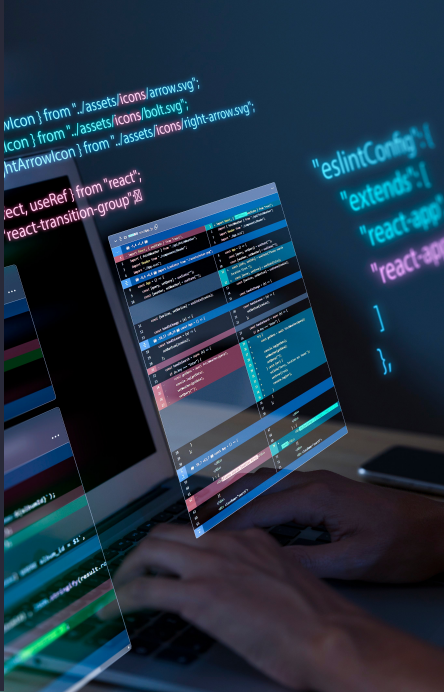


Code Flowchart



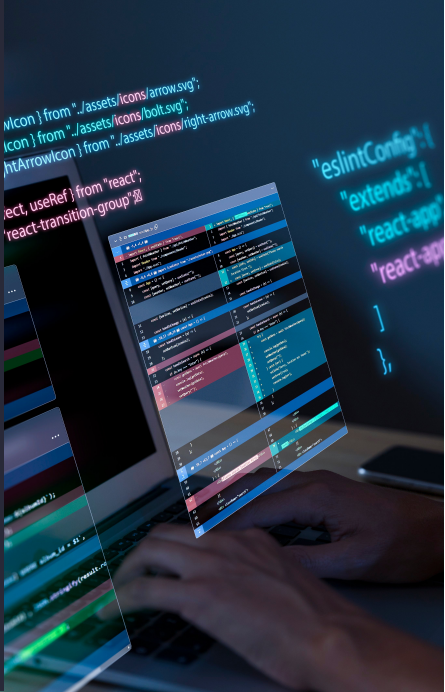
Limitations

- **LEDs were used to represent the heater/cooler**
- **The temperature was manually changed through the LM35 buttons in the simulation**
- **The seven segment display does not work properly on Proteus, however, on the ETA32 Kit it works just fine.**
- **LCD was used instead of the seven segment display and it functions correctly on Proteus.**
- **The EEPROM driver doesn't seem to interface well with the version on Proteus.**



Future Developments

- **Physical implementation to be done instead of a simulation**
- **Develop a properly working driver for the EEPROM**
- **Allow user to enter desired temp via keypad instead of hard coding the desired temperature in the code**



Conclusion

The project successfully implemented an embedded system for controlling an electric water heater, demonstrating effective integration of various hardware components and software logic.

Thanks!

Do you have any questions?

elaraby.mohammed@live.com