Master Thesis

Master's Programme in Electronics Design, 60 credits

ELECTRONIC WATER HEATER

15 credits

Halmstad 2018-03-01
Rincy Valsalan
ABSTRACT

The main aim of my project is to develop a hardware implementation of the electronic water heater by choosing different components and minimize the errors in the same. I have considered several options depending on the availability of components, cost, reliability, implementation, financial budget, specification and thinking about the professional technical skills required. In this project I designed and implemented, an AVR micro controller based water temperature measurement system using Atmega328p microcontroller.

The idea of the project came from a company called Relek production AB, Sweden and they develop and supply electrical equipment for heating: such as electric boilers, under floor heating boilers, IR heaters, emergency power plants, power monitors, etc. Now they want to develop a new version of electronic water heater and according to their specification.

The microcontroller (Atmega328p) based temperature control system is used in this project for providing better functioning of the system and will also serve the following purposes.

1) As there will be less usage of energy as it is more energy efficient.

2) The microcontroller along with temperature sensor decides when the heater should turn on/off.

With this project I have designed the schematic diagram by using Eagle Autodesk PCB CAD program. The seven-segment display is used in this project to show the current temperature. A temperature sensor (LM35) is used in this project to sense the temperature and give these measured values to the microcontroller. The temperature measurement and heater control are processed using C++ program.

I have connected the circuit as per the schematic diagram and programe the microcontroller, interfacing all the major components like 7 segment display, temperature sensor, 2 pushbuttons (for manually incrementing and decrementing the set point in the program), and optoisolator (to sense the output from microcontroller and control the heater through thyristor).
ACKNOWLEDGEMENTS

This thesis is the concluding part of master's program in Electronic Design at Halmstad University, Sweden. Firstly, I would like to thank my program director Dr. Pererik Andreasson for constantly supporting and guiding me throughout the program.

My sincere gratitude to my thesis supervisor Dr. Hans-Erik Eldemark, for his patience, unlimited explanations, valuable instructions, and support. I would also like to thank Dr. Maben Rabi for his valuable suggestions and guidance that helped me to complete my thesis.

Special thanks to my husband Aneesh Alosheyas and my parents for their support and faith in me. Finally, I would like to thank my friend Swetha Yanamandra for sharing ideas and constantly motivating me throughout the thesis work.

Rincy Valsalan
Halmstad University, Sweden.
INDEX

ABSTRACT...........................................................................................................................................................................ii

ACKNOWLEDGEMENTS.........................................................................................................................................................iii

List of Tables........................................................................................................................................................................vi

List of Figures.......................................................................................................................................................................vi

Chapter 1 Introduction..........................................................................................................................................................1

Chapter 2 Older version of electronic water heater

2.1 Main parts and drawbacks...............................................................................................................................................3

2.2 Block diagram (older version of the product) ...............................................................................................................5

2.2.1 Power Supply...........................................................................................................................................................7

2.2.2 Relay.........................................................................................................................................................................7

2.2.3 Atmega8.................................................................................................................................................................8

2.2.4 Overheat sensor.......................................................................................................................................................9

Chapter 3 New Version of Heater

3.1 Block diagram description.............................................................................................................................................10

3.1.1 Power supply block.................................................................................................................................................11

3.1.2 ATmega328p3.........................................................................................................................................................12

3.1.3 Seven Segment Display........................................................................................................................................13

3.1.4 Temperature Sensor.............................................................................................................................................14

3.1.5 Opto-isolator........................................................................................................................................................16

3.1.6 Thyristor...............................................................................................................................................................16

3.1.7 Wi-Fi Module, Esp8266....................................................................................................................................19

3.2 Flow chart.................................................................................................................................................................20

3.2.1 Flow chart description........................................................................................................................................21
Chapter 4 Technical proposal Diagram .......................................................... 22

4.1.1 Power Supply ......................................................................................... 23

4.1.2 Connection details of Atmega328p with other components .................. 23

4.2 Method ....................................................................................................... 25

Chapter 5 Software used

5.1 Schematic design ..................................................................................... 30

5.2 Software used for programming the microcontroller

5.2.1 Atmel Studio 7 ....................................................................................... 31

5.2.2 Arduino IDE ........................................................................................ 31

Chapter 6 Overheat Sensor Study

6.1 NTC .......................................................................................................... 34

6.2 Probe Thermostat ...................................................................................... 34

6.3 Overheat Protection Switch (EF06052) .................................................... 34

6.4 General method ....................................................................................... 35

Chapter 7 Results & Discussion .................................................................... 36

Chapter 8 Conclusion & Future work ........................................................... 43

References ....................................................................................................... 45

Appendix .......................................................................................................... 47
List of Tables

Table 3.1: Comparison between LM35 with other sensors
Table 3.2: The pin details of Esp8266 Wi-Fi module
Table 5.2: Technical specification of the board

List of Figures

Figure 2.1: An electronic water heater
Figure 2.2: Block diagram previous system
Figure 2.3: Single phase connection diagram
Figure 2.4: The connection diagram of relay with power supply, heater, and sensor
Figure 3.1: Block diagram of the new system
Figure 3.2: Power Supply Block detailed diagram with signals
Figure 3.3: Connection Diagram of LM35
Figure 3.4: Flowchart of the system
Figure 4.1: Seven segment display showing room temperature
Figure 4.2: Serial monitor output
Figure 4.3: Serial monitor output after changing the set point
Figure 4.4: The temperature display and optoisolator output after changing the set point
Figure 7.1: Figure 7.1 Room temperature displays on seven segment and mobile application
Figure 7.2: Figure 7.2 The LED turned OFF at 30 degree Celsius
Figure 7.3: LED turned ON when temperature is 26 degree Celsius
Figure 7.4: Hysteresis loop showing the ON/OFF process
Figure 7.5: Serial monitor output showing the set point is decreased
Figure 7.6: Serial plotter output showing status of temperature sensor and set point in each second
Figure 7.7: Graph showing the set point and sensor temperature
Figure 7.8: Graph shows the temperature at which heater is turning ON/OFF
Chapter 1

Introduction

Water heaters are devices used to heat the surrounding water. This is best suitable for industrial application where the thermostatic control will automatically turn on or off when it reaches the temperature you set. There are a lot of different water heaters available in the market, among them solar water heater, tank less water heater, and conventional storage tank are the important ones. These all need more installation cost, space, and power. And for solar water heater the biggest problem occurs in the winter season when there is less available solar energy. So, electronic water heater is always a good choice to eliminate all those difficulties which other ordinary water heater have.

The older version of the electronic water heater consists of immersion heater (which look like a coil or a metal loop, see Figure 2.1) along with a thermostat and a switchbox or terminal box with microprocessor-controlled temperature control. In new version also, the outer view is similar but there will be change in electronic components inside the switchbox. The main changes include: Atmega328p microcontroller instead of Atmega8, relays are replaced by thyristors, implementation of Wi-Fi module to operate the device remotely.

The high voltage part of the device includes thyristor, bridge rectifier, voltage regulator, and transformer. And the low voltage part includes microcontroller, 7 segment display (which is used to display the operating status such as current temperature), temperature sensor (to sense the temperature of water), and Wi-Fi module (to control a device remotely). The optocoupler or optoisolator is used to provide an isolation between microcontroller and thyristor, as it will damage the microcontroller (maximum input voltage: 5V DC) if we directly connect it to thyristor (maximum input voltage 400 V AC).

Improving the previous system can contribute greatly to reducing the usage of power, eg: mechanical relay draws more current than thyristor; hence by choosing thyristor a great reduction in power consumption as well as efficiency of the device can be improved. This electronic water heater which contains a microcontroller-based temperature control seeks to maintain a desired temperature that is optimal for the working of the heater. The main benefit of this kind of water heater is its convenience (i.e. easy to implement and operate) and energy saving.

The main components used in this project are Atmega328p microcontroller, seven segment display, temperature sensor (LM35), thyristor, optoisolator, heater (instead of this a bulb also can be used for demonstration purpose). The power supply unit also implemented but for breadboard connection it is not really required, because we can connect 5v directly form a power supply in lab.
But for the final product this power supply unit is necessary to convert the 400V AC to 5V DC. This is achieved by using delta connected step down transformer, diode and voltage regulator combination.

1. **Problem statement**

The problems that this thesis addresses are, (1) Change two phase to three phase, (2) a good temperature hysteresis should be provided, (3) A better electrical component should be used instead of relays to improve the life time of the device, (4) a Wi-Fi technology should be implemented to control the device remotely, (5) a better over heat protection should be implemented.

2. **Approach**

My thesis aim is to improve the electronic water heater with upgraded electronics, by using Atmega328p microcontroller instead of Atmega8A, thyristor instead of relay, Wi-Fi module for controlling the heater remotely. The main significant difference between Using thyristor in the place of relays will improve the life time of the device and it is also power efficient, as mechanical relay draw more power than thyristor. Thyristors can be used to phase control a load, which means, it can use to dim lights, control the speed of a motor. This is much impossible with relays.

3. **Goal**

1. Provide a better hysteresis to turn ON and OFF the heater through programming the microcontroller. Replacing mechanical relays with thyristor will enhance the product performance and life expectancy of the heater.
2. Check the device for getting expected results and compare with the previous product.
Chapter 2
Older version of electronic water heater

2.1 Main parts and drawbacks

This water heater consists of an immersion heater, which can be flanged, threaded or screw plug, over-the top-side, and it is available with different power ratings also. Flanged type immersion heaters are most commonly used and usually round shaped and welded onto pipe flanges. It is used for heating the water to a desired temperature. This immersion heater is attached to a switch box or terminal box where all the other controlling (such as relays, microcontroller, etc.) and measuring components (such as sensor) are placed. Each element of the immersion heater is connected to the main supply line.

The terminal box consists of a high voltage side which includes power supply components (which contain transformer, bridge rectifier, and voltage regulator) to give desired operating voltage to other components like microcontroller, led, sensor etc. (which are on the low voltage side). The connection between microcontroller and relays is achieved by using an optocoupler or optoisolator which transfer the signal from the microcontroller to the relay to make the heater on or off.

Usually home-based water heaters are a 4500 watt with a voltage of 240 V to run efficiently. Below this voltage the heater will have to work tougher that damages the heater. But most of the industries have three phase power supply and they preferred a device which can be worked on the same supply condition to reduce the wiring cost. The step-down transformer was used here only allows it to accommodate a 240V AC (two phase).

The main drawbacks in previous product are:

1. The power supply used in this project can only operate up to a voltage of 240 V and hence a separate wiring should be provided to make the heater to operate under such specification.

2. Relays are going to off/on for small variation in temperature.

3. Sometime the overheat sensor sensing the atmospheric temperature and making the device to turn off.

4. Each time user needs to go and fix the problem to get desired operating condition.

These are the four main reasons behind the idea for making a new device which can minimize all these drawbacks. More details about each component and its working are explained in section below.
Figure 2.1: An electronic water heater (adapted from [25]).
2.1 Block diagram (older version of the product)

The older version of the product includes a microcontroller (Atmega8A), relay, transformer, seven segment display, optoisolator and heater, as the main components. This product can only withstand up to a voltage of 230V. Three relays are used in this system to control the heater. L1, L2, L3 represents each power line of three phase supply and N means neutral. Each block in Figure 2.2 explained below.

For a small variation in water temperature the mechanical relays used here are going to on or off state in a manner and making the system to drain more current as well as affecting the life time of the product.
During summer the overheat sensor gives wrong information by sensing the outside temperature resulting in shutdown of the whole system. This is only rectified manually by restarting the system (see section 2.2.4 and chapter 6 for more details). Single phase connection diagram of the previous product is shown below in Figure 2.3, where H1 represent the connection to the one heating terminal of the heater, L1 is the power supply line, and N means neutral. Figure 2.4 shows the internal connection diagram of power supply lines, overheat sensor, heater elements with relay. L1, L2, L3 represents power supply lines in a three phase with neutral point N.

![Single phase connection diagram]

Figure 2.3: Single phase connection diagram

Technical specification as of older product as follows:

1. Power range: 1000W to 12000W
2. Connection voltage: 230 V 1N
3. Maximum system pressure: 9 bar
4. Temperature: STB 75 degree Celsius, 110 degree Celsius, 220 degree Celsius, depending on application
5. Temperature through software: 75 degree Celsius, 90 degree Celsius, 160 degree Celsius
6. Sensor: NTC 10 kilo ohm
7. Dry core: sensing at increase 1 degree per 3 second
8. Sensitivity: sensing after 60 minutes
9. External control: SPST potential free connector
10. Data reading: 7 segment display
11. Sample time: 0.5s
12. Ambient temperature: 5 degree Celsius to 50 degree Celsius
2.2.1 Power Supply

The transformer used here is also a step down one, but it only can used in a two-wire system, that means this type of transformer used to convert 230 V AC to 12 V AC. It cannot function above the usual voltage rating. This 12V AC is converted to 12 DC with the help of diode rectifier, then this 12 DC is down converted to 5V DC by the voltage regulator. This 5V is essential for the working of the microcontroller and other components. As it can only operate up to 230V, a different wiring should to be provided for the heater to make it turn ON.

2.2.2 Relay

Here mechanical relays are used to make the heater on or off, depending on the temperature from the sensor. These relays are electromagnetic switches operated by electric current. It works simply like a manual switch, but instead of pressing or pushing manually the switch, we must apply electric current to change the switch of the relay [15]. It is helpful when we must turn on or off a device automatically. Relay consists of electromagnets and mechanical switch and an electromagnet is device made of wire wound in a coil around a ferromagnetic material such as iron. Depends on the voltage rating we can choose the relay. Some relays can be only used in low voltage application and some others are for high voltage application.
If proper voltage is not applied, it will cause damage of the relay. Depending on time, the spring as well as linkage in the coil of the relay became weak and this will result in maloperation and false trips. It also does not have the directional feature. If we are planning to make a device for long run, then the mechanical relays are not a good choice.

### 2.2.3 Atmega8

Atmega8 is a 28 pin PDIP or 32 pin TQFP chip with a low power CMOS 8-bit microcontroller based on AVR enhanced RISC architecture. The CPU core ensures correct program execution. It has a throughput of 1MIPS per MHz and it will give a clear idea of power consumption versus processing speed. The TQFP chip consists of 32 general purpose working registers and it is directly connected to the arithmetic and logic unit. Atmel’s high-density non-volatile memory technology is used to manufacture this device. The full suite of program and system development tools, including C compilers, macro assemblers, program simulators, and evaluation kits are supported in ATmega8.

The main features of this microcontroller include [5]:

1. 8 Kbytes of In-System Programmable Flash
2. 512 bytes of EEPROM
3. 1 Kbyte of SRAM
4. 23 general purpose I/O lines
5. 32 general purpose working registers
6. Three flexible Timer/Counters with compare modes
7. Internal and external interrupts
8. A serial programmable USART
9. A byte oriented Two wire Serial Interface
10. A 6-channel ADC in PDIP with 10-bit accuracy
11. Eight channels in TQFP and QFN/MLF packages with 10-bit accuracy
12. A programmable Watchdog Timer with Internal Oscillator, etc.
2.2.4 Overheat sensor

Overheat sensors are used to sense and protect against over temperature and over current. In the previous version of the product, the thermostat or temperature switch, BH-B2D, is used for protecting the circuit from thermal overload. It is made up of a bimetal sealed firmly with small internal resistance, no noise, and it will reset automatically. These bimetals are sensitive to temperature and current. When the current passes through the bimetal, the status will be changed due to over temperature or over current. As current rises, heat also rises.

This will result in cut off of the circuit or turning on the circuit. This type of switch is safer than normal protectors, as there is no lag temperature sensing under the condition of sudden huge current.

The main specifications are [21]:

1. Electrical Rating: DC-12V at maximum 4A; DC-24V at maximum 3A; AC-125V at maximum 3A; AC-250V at maximum 2A
2. Open temperature rating: (30 to 150) ±5 degree Celsius
3. Dimension: Length (15 mm), Width (7.3mm), Height (3.9mm), and with a lead wire (normally nickel strip of length 70mm and it can vary according to customers' requirement)
4. Reset temperature: 2 by 3 of the action temperature and with tolerance of ±15 degree Celsius

These sensors can be used as an effective reliable security protector, in alarm devices. These are widely used in various type motors, lighting device, inverter welding machine, switch power system, rechargeable battery circuit, portable power tools, and electric appliances. It also used in other electric equipment to prevent from overheating and overcurrent due to the abnormal work status. It can withstand pressure of 3.5MPa are appropriate for build-in system of plastic-coated motor.

In older version this sensor sensing the outer atmospheric temperature and giving a feedback to the microcontroller, this resulting in turning off the system. So, in the new version of the product this error should be eliminated. A better solution should be developed to implement an overheat sensor which will work perfectly.
Chapter 3
New Version of Heater

1. Block Diagram

Figure 3.1: Block diagram of the system

Signal flow in the block diagram:

1. Analog temperature value from sensor is given to microcontroller.
2. Digital temperature value from microcontroller is given to 7 segment display.
3. Microcontroller sends a pulse to optoisolator; (4) it will make the optoisolator to turn on the thyristor; (5) then thyristor will make the heater to turn on.

(Note: the dotted line shows the components I have designed in this project)
3.1 Block diagram description

3.1.1 Power supply block

A power supply is used to provide electric energy to at least one electrical or electronic device. A device which controls the output voltage or current to a value is called regulated power supply. This is really an unavoidable part in circuit design if we are including different components with different current or voltage ratings. The regulated output is obtained in steps; first the AC current or voltage is down converted by transformer then it is rectified to DC by using a series and parallel combination of diodes or using a bridge rectifier. This is then filtered by using a capacitor combination to obtain a rippled current and after that it is regulated by a voltage regulator to obtain DC regulated power supply (to 5V). Here I have used 3 delta connected single phase transformers. Instead of that it better to use a three-phase transformer to eliminate the damage which can be caused in special cases like thunder and lighting. In 3 single phase transformer connection if any damage caused to any one of the transformers other two still work which causing damages to the winding as well as connected devices.

Figure 3.2  Power supply block diagram with signals.
3.1.2 ATmega328p

The microcontroller Atmega328p is the main component of this project with single integrated 
circuit containing a processor core, memory, and programmable input/output peripherals, which 
controls all the activities, such as controlling devices, comparing and monitoring the inputs, and 
sending outputs. It is in-built with an ADC (analog to digital converter) and therefore it does not 
need to interface external withy ADC. It used in automobile engine control systems, implantable 
medical devices, remote controls, office machines, appliances, power tools, toys and other 
embedded systems.

Atmega328p is a 28 pin PDIP chip with a low power CMOS 8-bit microcontroller based on AVR 
enhanced RISC architecture. By using this microcontroller, a system designer can optimize the 
devise for power consumption versus processing speed as it achieves throughputs close to 1 
MIPS per MHz. For example, reducing the clock speed from 16 MHz to 8 MHz can drop the 
current usage from 12 mA to approximately 8.5 mA. On-chip boot program and conventional 
non-volatile memory programmer are used to change or reprogram the Flash program memory. 
The 328P alphanumeric at the end represents a Pico power processor, designed for low power 
consumption. The normal working of this microcontroller is achieved by providing a supply 
voltage of +5V DC. The microcontroller read the available data from the input and processes the 
same to provide output.

The microcontroller-based control system contains essentially four parts, i.e., the process, the 
analog to digital converter, the control algorithm, and the clock. The output from the process (eg: 
temperature sensor) is continuous. This output is converted into digital form by the analog to 
digital converter. It provides a cost effective and highly flexible solution to many embedded 
control applications. The ATmega328p supports program and system development tools 
including, C Compilers, Macro Assemblers, Program Debugger/Simulators, and in-Circuit 
Emulators. The main significant difference between Atmega328p and Atmega8 is the flash 
storage space or flash program memory of the chip, which is 8kb for Atmega8 and 32 kb for 
Atmega328p. In future if want to interface other components to this microcontroller then it will 
not affect the performance of the system as it contains more storage space.

The salient features of ATmega328p microcontroller are [1]: -

1. 32k bytes of in system programmable flash with read while write capabilities.
2. 1k byte EEPROM
3. 2k byte SRAM
4. 23 general purpose registers
5. 32 general purpose working registers
6. Real time counters (RTC)
7. Three flexible timers/counters with compare modes and PWM
8. One serial programmable USARTs
9. One byte-oriented 2-wire Serial Interface (I2C)
10. A 6-channel 10-bit ADC in PDIP
11. An 8 channels ADC in TQFP and QFN/MLF packages)
12. A programmable Watchdog Timer with internal Oscillator
13. An SPI serial port
14. six software selectable power saving modes
15. Timer/Counters
16. SPI port
17. interrupt system to continue functioning

3.1.3 Seven Segment Display

A seven-segment display is an electronic display used for displaying numbers from 0-9 and also characters. The n-type and p-type semiconductors joined together to form a pn junction which the basic architecture of a light-emitting section of an LED. When the pn junction is forward-biased, electrons in the n side are excited into the p side and where they combine with holes. As a result, photons are emitted. This is the main working principle of a LED. We can find the same displays in digital clocks, certain calculators and timers. Common Anode and Common Cathode are the two types of seven segment display. Common Cathode: negative terminals of all the 8 LEDs are connected, called as COM and all positive terminals are left alone. Common anode: positive terminals of all the 8 LEDs are connected together, called as COM and all negative terminals are left alone.

The 7-segment display consists of 8 LEDs and each LED is used to display each segment, for displaying the decimal point or dot the 8th LED is used. It contains 10 pins in which a, b, c, d, e, f, g and h/dp, are used for denoting one to 8 pins. The two middle pins are common anode/cathode of all the LEDs. We need to connect only one COM pin because common anode/cathode is internally shorted.

Two 7 segment displays are used in this project to obtain desired display condition and it is the same display used in previous product. It was the demand from the company to use the same display otherwise it is better to use an LCD (Liquid crystal display).
All are common cathode displays (HDSP-5503) and in red color. These displays are ideal for most applications and the digits can be viewed from 7 meters. It has the following features; average power per segment is 82 mW, DC forward current per segment is 25 mW, operating temperature range is -40 to 100 degree Celsius, and peak wave length of 635 nm [28].

3.1.4 Temperature Sensor

The sensor used in is LM35 and these particular series are precision integrated-circuit temperature sensors. The reason behind using LM35 is; it is a popular, less expensive temperature sensor, and it gives a voltage reading which are linear to the temperature readings in degree Celsius.

And, it provides an output voltage of 10.0mV for each rise of temperature in degree Centigrade. The output value of the sensor can be fed to any analog to digital converting pin of the microcontroller for reading and displaying the output to any display unit. Figure 3.3 shows the connection diagram of LM35, where Vs pin is for the applying supply voltage for working of the sensor, Vout pin for getting analog output value from the sensor, and the last pin is connected to the ground. The comparison between LM35 with another sensor is given in Table 3.1.

Figure 3.3 Connection Diagram of LM35 (adapted from [14]).
<table>
<thead>
<tr>
<th>Sensor</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS18B20</td>
<td>It is a one wire digital thermometer provides 9-bit to 12-bit measurement of temperature in Celsius. Typical sensing temperature is -55 degree Celsius to 120 degree Celsius.</td>
</tr>
<tr>
<td>PT100</td>
<td>It is a platinum temperature sensor. These sensors are resistance temperature detecting (RTD) and having a resistance of 100 ohms at 0 degree Celsius. It provides a linear increase in resistance according to temperature.</td>
</tr>
<tr>
<td>TMP35 / TMP36 / TMP37</td>
<td>It functions similar to LM35 and it also provides a voltage output that is linearly proportional to the Celsius. Without the use of external calibration, it provides an accuracy of ±1°C at 25°C and ±2 degree Celsius over the −40 to 125-degree Celsius temperature range.</td>
</tr>
<tr>
<td>LM35</td>
<td>It is a precision temperature sensor and has an advantage over other temperature sensor as it provides voltage readings which are linear to the temperature readings in degree Celsius. It does not need any external calibration and can operate over a -55 °C to 150 °C temperature range.</td>
</tr>
</tbody>
</table>

Table 3.1 Comparison between LM35 with other sensors

The main features of this sensor include [14]: Calibrated specifically in degree Celsius, Linear 10.0 mV for each degree Celsius scale factor, −55° to +150°C range, Suitable for remote applications, Low cost, Operates at 4 to 30 volts, Less than 60 µA current drain, Low self-warming (0.08 degree Celsius in still air), Nonlinearity just ±0.25 degree Celsius typical. With using this sensor, temperature can be measured more accurately than with a thermistor. It also possesses low self-heating and will not cause more than 0.1-degree temperature rise in still air. The LM35 contains a reference voltage generator, a temperature dependent diode, and a buffer.
3.1.5 **Optoisolator**

Optocoupler or optoisolator are devices used to isolate the electronic and electrical circuits. This specific device isolates the sensitive electronics from robust electronics like motors, with keeping the load in control over the source.

It primarily used in this project to provide electrical isolation between a high voltage component (thyristor) and low voltage component (microcontroller). It can also provide electrical isolation between an input source and an output load using just light. The fundamental structure of an optocoupler consists of an LED that produces infra-red light and a semiconductor photo-sensitive device, such as photo-transistor, photo-resistor, photo-diode, photo-SCR, or a photo-triac; these are utilized to detect the emitted infra-red light [20].

In this project I have used a triac output optocoupler (MO3032-M). It consists of a GaAs infrared emitting diode optically coupled to monolithic silicon detector which performs the function of a bilateral triac driver. This type of optocoupler are designed to interface with components such as solid-state relays, industrial controllers, printers, motors, solenoids and consumer appliances, which are operating at a voltage of 115 VAC. More detail of this optoisolator is provided in the appendix.

3.1.6 **Thyristor**

It is a solid-state device which is used to switch electric networks or devices, such as lamps, motors, heaters etc. It is a four-layer device made of p-n-p-n and having three p-n junctions. The outer p-layer is called the anode and that to the outer n-layer is called the cathode. The thyristor is a unidirectional device, which means it passes current in only one direction that is from anode to cathode. It is appropriate for changing streams extending from a couple of milliamps at several volts to a huge number of amps at a huge number of volts. It will conduct current in one way only and used to control AC load [30].

The thyristor is essentially a switch, which can be exchanged on whenever, however can only be turned off when the current moving through it is zero. In sinusoidal AC switching circuits, it only conducts during one half of the cycle (like a half-wave rectifier), that is when the Anode is positive irrespective of whatever the gate signal is. In AC heating applications it happens twice per cycle. Heating is controlled by switching the thyristor on and off and altering the relative extent of on-time and off-time. In older version of the device each relay is used to control each element of the heater. Instead of relays, thyristor is connected to each element of the heater in new version.
Advantages of thyristor over relay:

1. Thyristor are best for switching an AC signal as they will shut themselves off around zero crossing.
2. If we have to control a dc device in future it is also possible by using thyristor and it is relay hard for relays to do the same as relays always open up the contacts under load.
3. Thyristor can replace relays in almost every application, as they are less expensive, and their function is similar to that of a relay
4. They are less noisy when compared to relay.

In this project I used a 3 pin 2N6507TG SCR THYRISTOR. The main features of this triac output thyristor are; Peak Repetitive Off-State Voltage of 400 V, maximum operating temperature 120 degree Celsius, average input current of 16 A, maximum trigger voltage 1.5 V, glass passivized junctions with center gate fire mainly for greater parameter uniformity as well as stability, small size, rugged, constructed for low thermal resistance, high heat dissipation, durability, blocking voltage up to 800 volts, 300 a surge current capability.

There is a chance of turning ON of thyristor accidently, due to sudden rise in current or voltage, especially in situations like thunder and lightning. This can create a lot of problem and even can result in destruction of the device. Simple snubber circuit is the best way to minimize this problem. Snubber circuit means a combination of resistor, capacitor, or diode connected in-between the load and the thyristor (See figure 3.4). To avoid the risk of false triggering of thyristor an RC snubber circuit must be used. It is used to damp the oscillatory voltage to a suitable value and hence it can be used for protection as well as to improve the performance.

3.1.7 Wi-Fi Module Esp8266

Esp8266 is a low power consumption Wi-Fi module with an embedded microcontroller with integrated TCP/IP protocol stack that can give any microcontroller access to Wi-Fi network, which means that it can directly access any Wi-Fi network. It is really cost effective and easy to implement. It is in-build with calibrated RF allowing it to work under all operating conditions hence it requires no external RF parts. The remarkable thing is that this module has a Microcontroller Unit integrated in it.
It also gives the opportunity to control input output digital pins via simple programming language. The normal operating voltage is 3.3 vdc so it should not connect directly to a 5 v supply board. The main advantage of using Wi-Fi compared to other, like Bluetooth and Zigbee is Wi-Fi has large range and comparatively more data bandwidth capability. The pin details of Esp8266 Wi-Fi module I given on Table 3.2.

The main features of this module are [8]:

1. 32-bit RISC CPU and running at 80 MHz
2. 64 KiB of instruction RAM, 96 KiB of data RAM
3. External QSPI flash – 512 KiB to 4 MiB and up to 16 MiB is supported
4. IEEE 802.11 b/g/n Wi-Fi
5. Integrated TR switch with balun, low noise amplifier, power amplifier and matching network
6. WEP or WPA/WPA2 authentication, or open networks
7. 16 GPIO pins
8. SPI, I²C,
9. I²S interfaces with DMA (sharing pins with GPIO)
10. UART on dedicated pins
11. one 10-bit Analog to Digital Converter

(Note: MiB is known as mebibytes and equivalent to $2^{20}$ bytes; KiB is known as kibibyte and equivalent to $2^{10}$ bytes)
<table>
<thead>
<tr>
<th>PIN</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTX D</td>
<td>UART Transmit Data</td>
</tr>
<tr>
<td>URX D</td>
<td>UART Receive Data: Input should be 3.3V compatible</td>
</tr>
<tr>
<td>CH_PD</td>
<td>Power-down: Low input powers the chip down, high input powers it up; tie high for normal operation or the module will not function.</td>
</tr>
<tr>
<td>GPIO0</td>
<td>At boot: Must be high to enter flash or normal boot; low enters special boot modes.</td>
</tr>
<tr>
<td>GPIO2</td>
<td>At boot: Low causes boot loader to enter flash upload mode; high causes normal boot.</td>
</tr>
<tr>
<td>RST</td>
<td>Reset; active low</td>
</tr>
<tr>
<td>GND</td>
<td>Ground</td>
</tr>
<tr>
<td>VCC</td>
<td>Power/3.3V</td>
</tr>
</tbody>
</table>

Table 3.2 The pin details of Esp8266 Wi-Fi module
2. Flow chart of the heater controller

Figure 3.5: Flowchart of the heater controller.
3.2.1 Flow chart description

The flow chart presented above is the logic developed to make a responsive system that would react to change in temperature with the help of microcontroller and sensor. By this way we can easily understand the logic behind the temperature control. The user needs to start the water heater by plugging it to the power source, and then it can be operated to a desired temperature level by entering the temperature value (T1) in degree Celsius, using the push button.

There are two push buttons implemented in this project, one is for incrementing the set value and other is used to decrement the set value. In this project I used a set value of 30. So, the user can choose which temperature is needed up to which the heater will be turned on. By fixing the Wi-Fi module the user can also have the facility to adjust the operating condition through smart phone, tablet, or other such type of devices.

When the device turned on and at the same time the temperature sensor sends information about the current water temperature (T2) to the microcontroller. So, if the temperature of the water is still the same as provided by the user the microcontroller will not turn the heater on. It is better to provide a maximum temperature, and above which the heater should not work. That means the temperature should vary between the set value 30 degree Celsius and its maximal value 70 degree Celsius. It should not a compulsory one if we are not planning to increase the temperature above 50 degree Celsius. This is the method by which we set a hysteresis.

The temperature value from sensor will be displayed on the seven-segment display. Then the microcontroller compares the two temperatures (user set temperature, T1 and current water temperature, T2). If the answer is “yes,” that means the current water temperature from temperature sensor (T2) is greater than or equal to the set temperature (T2), then the heater will turn off. If the answer is “no” then the heater will be in on condition until the current water temperature from temperature sensor (T2) is greater than or equal to the set temperature (T2).

This is exactly same algorithm I have applied in the programming part. The whole program I have given in the appendix.
Chapter 4
Technical proposal Diagram
4.1.1 Power Supply

The power supply should deliver constant output regulated supply for successful working of the project. Three delta connected single phase transformers are used in this project, so that it can be used in high voltage as well as in low voltage condition. This is done by choosing the main line L1, L2, and L3. Through this it can accommodate an input voltage of 400V.

The primary of the three transformers are connected to the main line (L1, L2, L3), respectively, which step down the supply voltage to 12V AC. Secondary sides of each transformer are connected to each bridge rectifier or else it can be connected by using a combination of diodes (both gives same result). The bridge rectifier will convert 12V AC to 12V DC. The capacitors are used for filtering purpose to remove ripples and to get pure dc voltage. The 12V DC is used as the input to 7805 voltage regulator for getting output voltage of +5V, which is needed for the operation of microcontroller, 7 segment display, and temperature sensor etc.

Low power voltage regulator offers the advantage of good regulation, current limitation and short circuit protection at 100mA. These particular type of regulators can be damaged is by an excessive input voltage. This regulator can withstand the input voltage up to 35V. For bread board connection this power supply unit is not necessary. This unit is really needed when converting the bread board connections and making it into a PCB.

4.1.2 Connection details of Atmega328p with other components

The connection details of each components and the pin specification are given below. Atmega328p contains three input output ports; Port B, Port C, Port D. Both port B and D are 8-bit bi-directional I/O port with internal pull-up resistors, while port C is a 7-bit bi-directional I/O port with internal pull-up resistors.

A 16 MHz crystal oscillator is connected to PB6 and PB7, with the help of two 22pF capacitor to provide necessary frequency for the working of the microcontroller. Port B and D are occupied with the connection for the two seven segment display and two push buttons. The first seven segment display is connected to port D starting from PD2 (pin4) and second seven segment display connected to port B starting from PB2 (pin16).

The output from the temperature sensor is connected to third pin (PC2) of the Port C. The other two pin are correspondingly connected to ground and 5v supply. PD0 and PD1 are occupied with the two push buttons. The output to the 3 optoisolator from microcontroller is given from port C through PC3, PC4, and PC5.
The output signals from the three optoisolator are connected to the gate of the three thyristors, respectively. The anode of the thyristor is connected to the main supply voltage through a heater. It is better to provide a fuse in between the heater and the thyristor to save the thyristor from the damage due to high voltage or current during some special situation. LM35 temperature sensor will give an output which is linearly proportional to the temperature in Celsius.
4.2 Method

The block diagram of the older as well as new version has been drawn first and analyzed each block carefully. The schematic diagram is drawn with the help of Eagle software with the reference of the block diagram. The components have been chosen by considering the basic idea of the older version of the device and the specifications need to meet. Schematic diagram has been drawn by carefully reading and analyzing a lot of internet resources and through discussions. After finishing the schematic diagram, I have programmed the microcontroller using Arduino IDE.

I have started programming by interfacing temperature sensor and the 7 segments with microcontroller. Each port is assigned for each seven-segment display and according to the schematic diagram the pin number of each component is provided in the program. Input and output are defined correctly in the program. A program for converting the analog value from the temperature sensor to digital value has been implemented. The seven-segment display also interfaced properly so that exact digit will display on each seven segments.

After interfacing temperature sensor and the 7 segments, the program is checked by using compile command in Eagle software and corrected the errors. Then the hardware is connected to software using a cable and uploaded the program to the microcontroller, then checked whether the hardware as well as the program is working properly or not. After uploading the program to the microcontroller, the seven-segment display started displaying values. From serial monitor output it was visible that the seven-segment display is exactly showing the same value given by the temperature sensor. Figure 4.1 shows the 7-segment display showing the room temperature of 23 degree Celsius sensed by the temperature sensor.

In this project LM35 temperature sensor is used to provide the appropriate voltage which is equivalent to the temperature of the medium. But the output of LM35 temperature sensor is analog in nature and microcontroller cannot process the analog signal directly. So, first it will convert the analog output of LM35 temperature sensor to digital values using its analog to digital converter and then it processes the digital value to convert the digital value in degree centigrade value.
Below shows how we can convert analog value of temperature sensor to digital when it should be displayed on the 7-segment display.

By using the following formula, the analog value can be converted to digital

\[ \frac{e}{V_{\text{max}}} = \frac{d}{2^{n-1}} \]

where, \( V_{\text{max}} \) the maximum voltage used by temperature sensor, \( n \) is the number of bits available, \( d \) is the digital value, \( e \) is the analog value from temperature sensor.

Example if we have 30 °C temperature, scale factor = 10.0 mV/°C or 0.01V/°C

Then sensor analog value = 30 * 0.01 = 0.3V

so, \( e = 0.30V \)

\( V_{\text{max}} = 5V \)

\[ d = \frac{e * 2^{n-1}}{V_{\text{max}}} \]

\[ d = 0.30 * 1023/5 \]

\[ d = 61.38 \]
That means every time we measure the digital value, it is almost double to the temperature. During programming it should be really considered to display the accurate temperature on the seven-segment display. So, the digital value must be divided by 2 for getting the approximate value of the temperature.

Once the device is start running, first check the set point in the program and analyze the actual performance of the temperature sensor is meeting or not. Two push buttons are used to allow the user to set a desired set point manually. A set point is the value which is already implemented or stored in the microcontroller while programming the same. Each button is assigned for incrementing and decrementing the set point. So, the user can increment or decrement the set point using the push button to reach a desired temperature they want. Once the temperature is set to the desired temperature, the microcontroller will compare both the set value and the current temperature of the water with the help of the temperature sensor. An opto-isolated thyristor is used to control the heater. Two seven segments are used to display the output i.e. it will display the water temperature from temperature sensor in degree centigrade. At the same time by analyzing the values from the temperature sensor the microcontroller gives input to the thyristor through optoisolator, thereby turning on/off the heater automatically, depending up on the set point from user. In this manner the microcontroller monitors and controls the temperature.

The voltage from temperature sensor is given to microcontroller through pin PC2. According to the program the microcontroller processes the analog signal into digital and forms a specific voltage level for a particular temperature. When the device started by providing proper supply voltages, the temperature sensor sends temperature measurement of water to the microcontroller and then the microcontroller compare the set temperature by user (if the user did not set any temperature then the microcontroller use the set point which is already saved in microcontroller while programming the same) and the temperature reading from temperature sensor. If the temperature from the temperature sensor is less than the set point, this will make the microcontroller to send an output to optoisolator. Then the optoisolator sends an output signal to the gate of the thyristor and making the thyristor to turn ON the heater. The microcontroller will not send any output signal to optoisolator, if the temperatures is greater than or equal the set temperature. There by turning OFF the heater and this makes the immersion heater to work until the desired user set temperature.

For demonstration purpose I am not using the power supply block which I have already designed in this project. The required voltages for the working of each component are applied from the power supply available in the laboratory. I initially gave a set temperature of 25 while programming the microcontroller.
So, if the temperature from the temperature sensor is less than 25 degree Celsius then the heater will turn on. Otherwise it will not turn on. As you can see in Figure 4.2, the outputs from the serial monitor showing that the heater turned ON when the temperature reached 23 degree Celsius. When the temperature is above the set point (that is 25 degree Celsius) the heater is turned OFF. So, from this it is noticeable that the system working accurately as desired.

![Figure 4.2 Serial monitor output](image)

Figure 4.2 Serial monitor output

Figure 4.3 shows the serial monitor output after changing the set point and with slight modification in program. Now the set point is changed to 30 degree Celsius and it is visible that the heater is ON when the temperature from temperature sensor is 23 degree Celsius. The output to heater is still on as the sensor temperature is not going above or equal to the set point.
Output is analyzed after changing the set point to 30 degree Celsius and provided little bit modification in program, like taking 100 sample values from the temperature sensor and taking average of it to get a precise value of temperature. I have also connected an LED as the output from optoisolator to check if the microcontroller giving output at a particular temperature or not. From the serial monitor output itself visible that the microcontroller sensing the input from the temperature sensor and according to that turning ON or OFF the LED. In breadboard connected circuit also it is observable by the glowing of the LED. Figure 4.4 shows the seven-segment display and optoisolator output through LED. From this it is clear that when the temperature is 27 the microcontroller sending signals and making the LED to glow.
Chapter 5

Hardware and software used

This project consists of two major parts, which are hardware or schematic design and software used for programming the microcontroller.

5.1 Schematic design

For designing the hardware or schematic of my project I used EAGLE (Easily Applicable Graphical Layout Editor) Autodesk PCB CAD program. It is a PCB design software platform consisting of a schematic's editor, a printed circuit board editor and an auto router unit. The software originates with an extensive library of components. With the use of library editor is also possible to design new schematic or modify existing ones [7].

Eagle is prepared by CadSoft and is also available in three versions. The light version is restricted to one sheet of schematics and half Eurocard format. It can be used under with freeware license for non-commercial use. This software can be downloaded for Windows or Linux. The schematic has designed and drawn with the help of tools provided in the software. Same software can be used to make Printed Circuit Board (PCB) using layout editor. According to the design I have connected the circuit by using the standard component symbols.

First, I have created the schematic sheet and then added the components. I have added all of the components which I need in this project by using the ‘Add’ icon from the toolbox left-side and then selected the ‘NET’ tool from the toolbox to make connections between each component. In Eagle it is easy to make the connections between each component by using the ‘NET’ tool. If we want to modify a group of components it is also possible with the use of group tool. By simply clicking the components it is possible to change their values to a desired value.

Electrical rule check (ERC) and design rule check (DRC) are used to check whether the connection is correct and to make sure the board will function as expected. That means for example it will warn about unconnected input-pins and also warns if we have connected the supply voltage inappropriately. So, it is important to run the electronic rule check after creating schematic. This can be done by the ERC and DRC command from the toolbar. All the components used in this schematic can be viewed as a list for this the ULP command should be clicked and the bill of material should be selected.
5.2 Software used for programming the microcontroller

5.2.1 Atmel Studio 7

During the staring stage of the project I used Atmel Studio 7 as a programing platform to write and debug the program to my microcontroller. The program is written in embedded C. To program Atmel AVR microcontroller like Atmega328p we have to use Atmel Studio software. It supports all AVR microcontrollers by Atmel and it is also a platform for new AVR or ARM devices. Generally, it provides the same platform for 8-bit and 32-bit microcontroller. It includes the editor, C compiler, assembler, file downloader, and a microcontroller emulator. It can be freely downloaded from company website. There are many example projects available in it, which is helpful for beginners [4].

First Start the Atmel Studio 7 program by clicking its icon, then select File then click New Project. Then we have to select in which language format we need the program, such as simple C or C++. After giving the project name and location to save the file, click OK button. Then device selection dialog box will appear, there we have to choose which microcontroller we needed, in my case I chose Atmege328p, then click OK. Then the project file will be created, and we can add different instructions or C code to create a program as needed.

To compile the code, we have to click Build Solution from build menu. If there is any error, then it will show the line numbers in which error occurred. The compiler can generate different types of files with the extension such as “.exe,” “. elf,” “.map,” “.eprom” and “.hex.” For microcontroller it is recommended to use a file of HEX type. Debugging is the process of detecting and resolving errors or defects in a program or software code. It prefers to do the same to prevent any unexpected behavior of software as well as system. Debugging can be done by selecting ‘Start Debugging and Break’ from Debug menu. While debugging it is possible to check the content of each register and change the same. As per the requirement it can be used to build any kind of application. Then the generated code can be used to upload to the microcontroller.

5.2.2 Arduino IDE

I have also used Arduino IDE and Arduino UNO for testing my program and writing the program in a simpler way with a smaller number of instructions as compared to other programming platform. Arduino IDE is a powerful, feature rich development tool for AVR microcontrollers. It’s making to provide the programmer with the easiest possible solution of developing applications for embedded system [2].
Arduino Uno is an open-source stage utilized for building several devices within a short period of time. Arduino contains both physical programmable circuit board and a bit of programming, known as Integrated Development Environment (IDE). It is easy to compose and transfer the code to the physical board. The Arduino programming platform has turned out to be very prevalent and it helpful to individuals who are beginner in programming.

The Arduino Uno is a microcontroller board based on the ATmega328 and it has 14 digital input/output pins of which 6 can be used as PWM outputs. Other than this it also facilitates the following, 6 analog inputs, a 16 MHz crystal oscillator, a power jack, an ICSP header, a USB connection, and a reset button. The Arduino Uno can be powered with the help of the USB connection or with an external power supply. The power source can be selected automatically. AC-to-DC adapter or battery can be used as an external power supply. The external power supply of 6 to 20 volts can be used to operate the board. If we use more than 12V I will cause the voltage regulator to overheat and results in damage of the board. So, the most recommended range is 7 to 12 volts. The technical detail of the Arduino Uno board is given on Table 5.2.

The power pins details are as follows [3]:

**Vin**- It is input voltage to the Arduino board. There two are ports on the bored where we can connect the out external power to board.

**5V**- This is the regulated power supply needed for almost all components on the board including microcontroller

**3V3**- It is an on-board regulator generated supply voltage of 3.3 volts and it draws a maximum current of 50 mA.

**GND**- It is used provide ground connections to all the components.

The Arduino Uno can be programmed with the Arduino software called Arduino IDE. It is really easy to check the program and implement the same to the microcontroller we are using. The bootloader allows you to upload new code to the microcontroller without the use of an external hardware programmer and it communicates using the original STK500 protocol. By using Arduino Uno, we can also bypass the boot loader and program the microcontroller with the use of the ICSP (In-Circuit Serial Programming) header.
<table>
<thead>
<tr>
<th>Microcontroller</th>
<th>ATmega328</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Voltage</td>
<td>5V</td>
</tr>
<tr>
<td>Input Voltage (recommended)</td>
<td>7-12V</td>
</tr>
<tr>
<td>Input Voltage (limits)</td>
<td>6-20V</td>
</tr>
<tr>
<td>Digital I/O Pins</td>
<td>14 (6 of which are PWM output)</td>
</tr>
<tr>
<td>Analog Input Pins</td>
<td>6</td>
</tr>
<tr>
<td>DC Current per I/O Pin</td>
<td>40 mA</td>
</tr>
<tr>
<td>DC Current for 3.3V Pin</td>
<td>50 mA</td>
</tr>
<tr>
<td>Flash Memory</td>
<td>32 KB (0.5 KB is used for boot loading)</td>
</tr>
<tr>
<td>SRAM</td>
<td>2 KB</td>
</tr>
<tr>
<td>EEPROM</td>
<td>1 KB</td>
</tr>
<tr>
<td>Clock Speed</td>
<td>16 MHz</td>
</tr>
</tbody>
</table>

Table 5.1 Technical specification of the board [2]

The 14 digital pins can be used as input as well as output, using functions pinMode, digitalWrite, and digitalRead. Each pin has internal pull-up resistors and in addition some pins have special functions. The serial pins, pin0 (RX) and pin1 (TX) are used to receive (RX) and transmit (TX) serial data. These pins are connected to the matching pins of the ATmega328p.

The pin 3, 5, 6, 9, 10, and 11 can be assigned to provide 8-bit PWM output with the use of analogWrite function. The SPI pins, such as pin 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK) are used to support SPI communication. The pin 13 has an additional feature, which is a built-in LED is connected to this digital pin. When the pin13 became HIGH, the LED will glow and when the pin13 is LOW, then the LED will turn off.
Chapter 6

Overheat Sensor Study

In this thesis I am not designed and implemented a overheat sensor because of limitation of time. But here I am describing little bit about sensors and what type of sensors can be used in future with this device to protect it from overheat. This should implemented in-between microcontroller and the thyristor. The probe can be inserted inside the heating element to sense the overheat condition [17].

6.1 NTC

NTC (negative temperature coefficient) thermistors are the ones used to sense and protect a device or a circuit from excess of heat without causing damage to them. It has a typical temperature range of -80 degree Celsius to 150 degree Celsius. They are actually solid-state temperature sensors which act like temperature sensitive electrical resistors. They are most commonly used to reimburse for deviations in temperature in solenoids and coil. They are comparatively cheap and easy to use in any type of devices, especially in water heaters.

6.2 Probe Thermostat

It consists of bi-metallic quick action switch that is sealed in a brass, stainless steel or other materials depending on the request from the customer. Probes are used to sense liquid and are fixed by using a NPT thread or a flange. These types of sensors are typically used as an over temperature protection device. It can also be used to open or close an electrical device with a pre-set temperature. If we have to provide a proper hysteresis in certain application, then it is best choice. It has typical temperature range of 0 to 175 degree Celsius with a maximum current of 16 amperes at a nominal voltage of 250 V AC.

6.3 Overheat Protection Switch (EF06052)

This switch is a form of usually closed switch for temperature control. It is used for preventing devices from overheating, switching the circuit and playing the role in protection. The switching temperature usually ranging from 45 to 150 degree Celsius, but sometimes this switch shows error of plus or minus 5 degrees. It is used in all kind of electronic products as well as home appliances applicable to all kinds of home appliances and electronics products, such as Battery Chargers, Solenoids, Transformers, Electric Motors, Heating pads, Transformers, etc.
6.4 General method

The same sensor used in older version can also use here (BH-B2D) but providing little bit modification in program and the circuit. In previous system it was giving a wrong measurement of temperature causing complete stoppage of the system. It is better to provide a separate circuit to control the over heat is a best option, by this if the sensor sense wrong temperature (such as sensing atmospheric temperature instead water temperature) it will not result in shutdown of the whole system. A circuit should design that will only stop the heater under such circumstances and turn it ON when normal temperature is reached.

Another option is giving some instructions in program, so the microcontroller able to check the temperature sensor value as well as the overheat sensor value and comparing both to take a decision. That means if both the sensors are giving the same value then the microcontroller can make decision according to that. In this way it is possible to minimize errors caused in the previous system.
Chapter 7

Results & Discussion

In this project the temperature sensor accurately identifying the temperature and displaying the same on 7 segment display. The thyristor switches the heater on or off by receiving signal from the opto-isolator. An LED is connected instead of heater to examine the output from the microcontroller. The proper temperature measurement is achieved by LM35 temperature sensor and suitable hysteresis for turning ON or OFF the heater is provided on the program. The temperature measured by the temperature sensor is checked and verified by using a room temperature sensing application on my mobile. Figure 7.1 shows the room temperature application and seven segment display exactly showing the same temperature.

![Figure 7.1 Room temperature displays on seven segment and mobile application.](image_url)

Figure 7.2 shows when the temperature sensor measurement (30 degree Celsius) is reaching set point (30 degree Celsius), the thyristor going OFF consequently the LED connected across it is going OFF.
Figure 7.2 The LED turned OFF at 30 degree Celsius.

Figure 7.3 shows when the temperature sensor measurement (26 degree Celsius) going below the set point (30 degree Celsius) the thyristor turning ON consequently the LED connected across it is going ON.

Figure 7.3 LED turned ON when temperature is 26 degree Celsius.
A hysteresis has been generated to start the heater when the temperature is below 30°C and stop it when it is above 30°C. The microcontroller will take the average of 100 values of current water temperate and according to that only give directions to operate the heater. Two temperatures needed to be given for safety reason: the components in the switch box will damage if the temperature reaches 150°C, or if the water in the tank start boiling. So, providing hysteresis will avoid such type of damages. When heater is on, the set temperature defines at which temperature the heater should turned off. Hysteresis value specifies a point of temperature which should not be exceeded while heater turned off. Microcontroller will turn the heater on when its temperature drops under set point. This hysteresis process is described in the Figure 7.4.

![Hysteresis loop showing the ON/OFF process.](image)

Figure 7.4 Hysteresis loop showing the ON/OFF process.

The commands which I have used to program the microcontroller to provide proper hysteresis is given below.

```c
if(temperatureC < setpoint){
    digitalWrite(PIN_OPTO1,HEATER_ON);
    Serial.println("Heater ON");
    delay (100);
}
if (temperatureC >= setpoint && b){
    digitalWrite(PIN_OPTO1,HEATER_OFF);
    Serial.println("Heater OFF");
}
```
The microcontroller will compare the set temperature and current water temperature. If the current water temperature is greater than or equal to the set temperature, then the heater will turn OFF. And if the current water temperature is less than the set temperature, then the heater will turn ON. More detailed description of hysteresis is given on section 3.2.

Figure 7.5 shows the serial monitor screen output which displays temperature sensed by the sensor, set point, and the status of the heater. The set value can be changed by pressing each time the push button to a desired value. From the serial monitor output it is visible that the set temperature is decreased to 28 from 30 by pressing twice the push button. Same as this it is possible to set any value as set point below 70 degree Celsius (it is the maximum limit of temperature which is already implemented in the program).

![Serial Monitor Output](image)

Figure 7.5 Serial monitor output showing the set point is decreased.

The serial plotter output showing the set point (red line) is 30 degree Celsius and current temperature sensor (blue line) value as 29 degree Celsius (see Figure 7.6). The set point is decreased to 29 degree Celsius, making the heater to turn OFF and hence the temperature sensor value also going down to 25 degree Celsius. After some time the set point is increased to 30 and then further increased it to 32. This will make the heater to turn ON as the temperature sensor value is below the set point.
Figure 7.6 Serial plotter output showing status of temperature sensor and set point in each second.

The graph is provided on Figure 7.7 which is plotted using matlab with different temperature values. It gives a clear idea of the set point and temperature sensor reading. Starting the set point was 30 and then increased the set point to 31 using push button. After some time it is reduced to 29 and then further reduced to 26, which making the heater permanently going to OFF state (represented by the horizontal line).

In Figure 7.8 we can see the exact temperature at which heater is going ON/OFF state. Suppose if the user set a desired temperature, let say 28 degree Celsius is the desired temperature. After setting the same the sensor will start checking the temperature of water. If the temperature of water is equal to 28 degree Celsius then the microcontroller will stop sending input to the optoisolator thereby stopping the heater.

Figure 7.7 Graph showing the set point and sensor temperature
Same procedure will take place if the temperature of the water is greater than 28 degree Celsius. For temperature values which are less than 28 degree Celsius the heater will continuously work until the set temperature is reached. Through all these data it can be estimated that the system is working as desired and it is responding with the change in temperature.

![Graph showing temperature at which heater is turning ON/OFF](image)

Figure 7.8 Graph shows the temperature at which heater is turning ON/OFF.

Hideyuki Suzuki and Kazuyuki Aihara [27], presents a system with two sensors, one on the upper level and other on the lower level. The values from these two sensors are used to control the water level of the tank. That means the valve will be turn ON/OFF with hysteresis according to the information from the sensors. Here in this thesis only one sensor is used to send information and thereby controlling the heater by comparing with set point.

Stevan et al. [31], present a detailed study on electronic water heater. In this the designing, operating, and simulation of an immersion water heater is provided. The system shown in section 2.1 is almost similar to the one implemented in this thesis.

As stated in Martin Pfeiffer [16], section 3.2, and here also two temperature values (one from sensor and other is the set point) will determine at what temperature the heater should turn ON/OFF. It is clearly given in the flow chart section as well as in the program (which is given on the appendix).
As per the goal of the thesis implementation of an electronic water heater using Atmega328p microcontroller has been done. Programed the same and tested with different set points. In older version if there is any 0.1-degree change in temperature making the heater to turn ON or OFF, affecting the efficiency of the device. As compared with the older version the output from the temperature sensor in new product is not fluctuating more. It is maintaining a smooth ON/OFF procedure through the proper hysteresis setting in the program.

The Wi-Fi module details are provided, but not executed in the system. It also needs to be interfaced with the microcontroller through a well-defined program. The mechanical relays are replaced by thyristor. A study of overheat sensors is carried out and need to implement in future work. Because, it is also needed some more time to test the overheat sensors to analyze whether it can be function without any error. So, three of the five problems in the previous system are fixed and implemented presently in this project.
Chapter 8

Conclusion & Future work

The aim of the project was to design an electronic water heater, and it has been implemented by using Atmega328p, LM35 temperature sensor, thyristors and optoisolator. The connection of the circuit on a breadboard from the schematic diagram has been executed and the microcontroller is programmed. The output to heater is checked and verified by setting different temperatures. This product is easy to install and have minimized the general plumbing and wiring cost. It is less expensive and uses lesser energy than most the other types of water heater. There are still options for improvement of the system and thereby the business value of the product can be increased. The electronic water heater can be used in industry according to the demand and specification. The temperature hysteresis can be changed to a value especially in industries where a desired temperature for heating the liquid is needed.

![Figure: 8.1 Hysteresis in new system and old system.](image)

From figure 8.1 it is clear that the new system is fluctuating less than old system. The old system hysteresis pattern made the system to turn ON or OFF for minute variation in temperature. The temperature variations which results in the tuning ON or OFF the heater are comparatively stable in new system.
Advantages:

1. This is useful for fast heating of the water and easy to operate.
2. It can be used in home as well as industry.
3. The monitoring of the temperature allows anyone realize the current temperature of the water.
4. Monitoring and controlling the temperature using microcontroller save energy and makes it energy efficient product.
5. This can be employed to support people who can’t switch on or off the device manually, especially in the case of disabled people.
6. By fixing the Wi-Fi module the user can control and monitor the device through a smart phone or with another such type of gadget.

Future work

In future there is a lot of opportunity to develop this electronic water heater by interfacing it with other components, such as advanced sensors. The advance sensors can be used to measure the over temperature and over current values of the heater. It will really advantageous when the heater became broken or stopped working. Then by the use of these sensors we can identify the exact problem which forced the heater to stay unresponsive.

The power supply I have designed can be used during the PCB implementation. Even there is a chance to change the power supply which I have designed in this project to more power efficient and with using fewer components.

The Wi-Fi module needs to be interfaced with the microcontroller. This will not a hard task but will take some time to program and interface the same. Through this the wireless monitoring and control of the device can be achieved. This will help really in setting the temperature of the heater, turning on or of the device, and for rectifying the false temperature measurement without causing damage to the device. A GSM module also can be introduced to enhance the performance of the system. Through this it is possible to control a device by sending a message to it.

The same circuit can be used to control any other devices, such as a fan in industry. So similarly, in future there is option to use it for home automation by providing necessary changes in circuit as well as in program.
7. References

1. 8-bit AVR Microcontrollers, ATmega328/P DATASHEET SUMMARY, Retrieved from: http://www.atmel.com
12. Li W., Li Y. and Xiao F. The design and implementation of digital temperature measurement and automatic control system, IEEE International Conference on Computer Application and System Modeling, 2010.
Appendix

1 Program

#define PIN_DEC A3
#define PIN_INC A4
#define PRESSED LOW
#define PIN_OPTO1 A5
#define HEATER_ON HIGH
#define HEATER_OFF LOW

unsigned long start, finished, elapsed, total;

int setpoint=30;
int reading;
int average =0;
int count=0;
int tempReading;
int temp;
int delayTime=500;
int tempPin = A2;
int x,y,r,k;
int b=70;

int bcd_array[10][7] = {
    { 1,1,1,1,1,0 }, // 00
    { 0,1,1,0,0,0 }, // 01
    { 1,1,0,1,1,0 }, // 02
    { 1,1,1,1,0,1 }, // 03
    { 1,1,1,1,0,0,1 }, // 04
    { 1,1,1,0,1,0 }, // 05
    { 1,1,0,1,0,1 }, // 06
    { 1,0,1,1,0,1 }, // 07
    { 0,1,1,1,0,0 }, // 08
    { 1,1,1,0,0,0 }, // 09
};
void BCD0(int);
void BCD1(int);

void setup() {
    // We'll send debugging information via the Serial monitor
    Serial.begin(9600);
    // If you want to set the aref to something other than 5V
    pinMode(PIN_INC, INPUT_PULLUP);
    pinMode(PIN_DEC, INPUT_PULLUP);
    pinMode(2, OUTPUT);
    pinMode(3, OUTPUT);
    pinMode(4, OUTPUT);
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
    pinMode(A2, INPUT);

    { 0,1,1,0,0,1,1 }, // = 04
    { 1,0,1,1,0,1,1 }, // = 05
    { 1,0,1,1,1,1,1 }, // = 06
    { 1,1,1,0,0,0,0 }, // = 07
    { 1,1,1,1,1,1 }, // = 08
    { 1,1,1,0,0,1,1 }; // = 09
pinMode(PIN_OPTO1, OUTPUT);
pinMode(9, OUTPUT);
pinMode(10, OUTPUT);
pinMode(11, OUTPUT);
pinMode(12, OUTPUT);
pinMode(13, OUTPUT);
pinMode(14, OUTPUT);
pinMode(15, OUTPUT);
}

void loop(void) {
    for(int i = 0; i < 100; i++) {
        average = average + analogRead(tempPin);
        delay(1);
    }
    delay(100);
    average = average / 100;
    temp = average;
    temp = temp * 0.5;
    x = (temp / 10);
    y = (temp - ((temp / 10) * 10));
    delay(1000);
    BCD0(x);
    BCD1(y);
// now print out the temperature
float temperatureC = ((x*10)+y);
Serial.println("Sensor Temp: is "); Serial.print(temperatureC); Serial.println(" degree C");

unsigned long oldSetpoint = setpoint;

// if inc button pressed
if (digitalRead(PIN_INC) == PRESSED){
    setpoint= setpoint+1;
}
if (digitalRead(PIN_DEC) == PRESSED) {
    setpoint= setpoint-1;
}
Serial.println("setpoint is"); Serial.print(setpoint); Serial.println(" degree C");

if(temperatureC <setpoint){
    digitalWrite(PIN_OPTO1,HEATER_ON);
    Serial.println("Heater ON");
    delay (100);
}
if (temperatureC >=setpoint && b ){
    digitalWrite(PIN_OPTO1,HEATER_OFF);
    Serial.println("Heater OFF");
    delay(100);
void BCD0(int number)
{
  int pin = 9;
  for (int j = 0; j < 7; j++) {
    digitalWrite(pin, bcd_array[number][j]);
    pin++;
  }
}

void BCD1(int number)
{
  int pin = 2;
  for (int j = 0; j < 7; j++) {
    digitalWrite(pin, bcd_array[number][j]);
    pin++;
  }
}
1. Schematic diagram of the Prototype
2. List of components used in circuit

<table>
<thead>
<tr>
<th>Part</th>
<th>Value</th>
<th>Device</th>
<th>Package</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C8</td>
<td>100μF</td>
<td>C-EU025-025X050</td>
<td>C025-025X050</td>
<td>CAPACITOR, European symbol</td>
</tr>
<tr>
<td>C11</td>
<td>22pF</td>
<td>C-EU025-025X050</td>
<td>C025-025X050</td>
<td>CAPACITOR, European symbol</td>
</tr>
<tr>
<td>C12</td>
<td>22pF</td>
<td>C-EU025-025X050</td>
<td>C025-025X050</td>
<td>CAPACITOR, European symbol</td>
</tr>
<tr>
<td>C13</td>
<td>100μF</td>
<td>C-EU025-025X050</td>
<td>C025-025X050</td>
<td>CAPACITOR, European symbol</td>
</tr>
<tr>
<td>IC1</td>
<td>LM35</td>
<td>LM50</td>
<td>SOT23</td>
<td>SOT-23 Single-Supply Centigrade Temperature Sensor</td>
</tr>
<tr>
<td>LED1</td>
<td>HDSM-5503</td>
<td>S4301B</td>
<td>S4301B</td>
<td>Single Digit Seven-Segment Display (cathode common)</td>
</tr>
<tr>
<td>LED2</td>
<td>HDSM-5503</td>
<td>S4301B</td>
<td>S4301B</td>
<td>Single Digit Seven-Segment Display (cathode common)</td>
</tr>
<tr>
<td>LED3</td>
<td>Red</td>
<td>LEDR155</td>
<td>Q62902-155</td>
<td>LED</td>
</tr>
<tr>
<td>LED4</td>
<td>Green</td>
<td>LEDB155</td>
<td>Q62902-155</td>
<td>LED</td>
</tr>
<tr>
<td>LED5</td>
<td>Yellow</td>
<td>LEDB155</td>
<td>Q62902-155</td>
<td>LED</td>
</tr>
<tr>
<td>OC1</td>
<td>TIL191</td>
<td>TIL191</td>
<td>DIL04</td>
<td>TEXAS INSTRUMENTS OPTO COUPLER</td>
</tr>
<tr>
<td>OC2</td>
<td>TIL191</td>
<td>TIL191</td>
<td>DIL04</td>
<td>TEXAS INSTRUMENTS OPTO COUPLER</td>
</tr>
<tr>
<td>OC3</td>
<td>TIL191</td>
<td>TIL191</td>
<td>DIL04</td>
<td>TEXAS INSTRUMENTS OPTO COUPLER</td>
</tr>
<tr>
<td>Q1</td>
<td>16MHz</td>
<td>XTAL/S</td>
<td>Q5</td>
<td>CRYSTAL</td>
</tr>
<tr>
<td>R1</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R2</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R3</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R4</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R5</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R6</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R7</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R8</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R9</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R10</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R11</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R12</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R13</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R14</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R15</td>
<td>330</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R16</td>
<td>330</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R17</td>
<td>330</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R18</td>
<td>330</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R19</td>
<td>330</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R20</td>
<td>330</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R21</td>
<td>330</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R22</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>R23</td>
<td>430</td>
<td>R-EU_0204/5</td>
<td>0204/5</td>
<td>RESISTOR, European symbol</td>
</tr>
<tr>
<td>SW1</td>
<td>Reset</td>
<td>SWITCH_PUSHBUTTONQ-PE</td>
<td>TACT_PANA-EVQ</td>
<td>Buttons</td>
</tr>
<tr>
<td>SW2</td>
<td>Down</td>
<td>SWITCH_PUSHBUTTONQ-PE</td>
<td>TACT_PANA-EVQ</td>
<td>Buttons</td>
</tr>
<tr>
<td>SW3</td>
<td>Up</td>
<td>SWITCH_PUSHBUTTONQ-PE</td>
<td>TACT_PANA-EVQ</td>
<td>Buttons</td>
</tr>
<tr>
<td>T4</td>
<td>BT151</td>
<td>BT151</td>
<td>TO220CL</td>
<td>THYRISTOR</td>
</tr>
<tr>
<td>T5</td>
<td>BT151</td>
<td>BT151</td>
<td>TO220CL</td>
<td>THYRISTOR</td>
</tr>
<tr>
<td>T6</td>
<td>BT151</td>
<td>BT151</td>
<td>TO220CL</td>
<td>THYRISTOR</td>
</tr>
<tr>
<td>U1</td>
<td>328P</td>
<td>ATMEGA328P_PDIP</td>
<td>DIP28</td>
<td>Atmel 328P</td>
</tr>
</tbody>
</table>
Introduction

The Atmel® picopower® ATmega328/P is a low-power CMOS 8-bit microcontroller based on the AVR® enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328/P achieves throughputs close to 1 MIPS per MHz. This empowers system designers to optimize the device for power consumption versus processing speed.

Feature

High Performance, Low Power Atmel®AVR® 8-Bit Microcontroller Family

- Advanced RISC Architecture
  - 131 Powerful Instructions
  - Most Single-Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20MHz
  - On-chip 2-cycle Multiplier

- High Endurance Non-volatile Memory Segments
  - 32kBytes of In-System Self-Programmable Flash program Memory
  - 1kBytes EEPROM
  - 2kBytes Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data Retention: 20 years at 85°C/100 years at 25°C
  - Optional Boot Code Section with Independent Lock Bits
    - In-System Programming by On-chip Boot Program
    - True Read-While-Write Operation
    - Programming Lock for Software Security
Prerequisites

- Hardware Prerequisites
  - ATmega328PB Xplained Mini board
  - Two Micro-USB Cables (Type-A/Micro-B)
  - Atmel Power Debugger kit
  - Three female to male wires. One male to male wire.

- Software Prerequisites
  - Atmel Studio 7.0

- Estimated Completion Time: 60 minutes

Introduction

This hands-on will demonstrate how to develop AVR® applications in Atmel Studio 7 along with the rich user interface and other great development tools that it provides.

The ATmega328PB Xplained Mini evaluation kit is a hardware platform for evaluating the Atmel ATmega328PB microcontroller. A fully integrated embedded debugger is included in the kit, which provides seamless integration with Atmel Studio 6.2 or later. Easy access to the features of the ATmega328PB is enabled by the kit, facilitating easy integration of the device in a custom design.

This training module also demonstrates how to save power in applications. The power consumption will be measured using the power debugger board and the “Data Visualizer”, which is a new feature in Atmel Studio 7.

The training will start with creating an application on the ATmega328PB Xplained Mini board. The initial power consumption of the microcontroller will be measured. After this, different peripherals will be turned off, and the power consumption will be measured to show how much power is saved.
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 ±2/4°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 ±0.5°C at room temperature and ±1°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.

<table>
<thead>
<tr>
<th>PART NUMBER</th>
<th>PACKAGE</th>
<th>BODY SIZE (NOM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM35</td>
<td>TO-CAN (3)</td>
<td>4.699 mm x 4.699 mm</td>
</tr>
<tr>
<td></td>
<td>TO-92 (3)</td>
<td>4.30 mm x 4.30 mm</td>
</tr>
<tr>
<td></td>
<td>SOT-23 (6)</td>
<td>4.90 mm x 3.91 mm</td>
</tr>
<tr>
<td></td>
<td>TO-220 (3)</td>
<td>2.996 mm x 10.16 mm</td>
</tr>
</tbody>
</table>

(1) For all available packages, see the orderable addendum at the end of the datasheet.
6-Pin DIP Random-Phase Optoisolators Triac Driver Output
(400 Volts Peak)

The MOC3020 Series consists of gallium arsenide infrared emitting diodes, optically coupled to a silicon bilateral switch.

- To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option.
- They are designed for applications requiring isolated triac triggering.

Recommended for 115V/240 Vac(rms) Applications:
- Solenoid/Valve Controls
- Lamp Ballasts
- Interfacing Microprocessors to 115 Vac Peripherals
- Motor Controls
- Static ac Power Switch
- Solid State Relays
- Incandescent Lamp Dimmers

MAXIMUM RATINGS ($T_A = 25^\circ$C unless otherwise noted)

<table>
<thead>
<tr>
<th>Rating</th>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFRARED EMITTING DIODE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reverse Voltage</td>
<td>$V_{BR}$</td>
<td>3</td>
<td>Volts</td>
</tr>
<tr>
<td>Forward Current Continuous</td>
<td>$I_F$</td>
<td>60</td>
<td>mA</td>
</tr>
<tr>
<td>Total Power Dissipation @ $T_A = 25^\circ$C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Negligible Power in Triac Driver</td>
<td>$P_D$</td>
<td>100</td>
<td>mW</td>
</tr>
<tr>
<td>Derate above 25°C</td>
<td>$P_D$</td>
<td>1.33</td>
<td>mW/°C</td>
</tr>
<tr>
<td>OUTPUT DRIVER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Off-State Output Terminal Voltage</td>
<td>$V_{DRM}$</td>
<td>400</td>
<td>Volts</td>
</tr>
<tr>
<td>Peak Repetitive Surge Current</td>
<td>$I_{SM}$</td>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>Derate above 25°C</td>
<td>$P_D$</td>
<td>300</td>
<td>mW</td>
</tr>
<tr>
<td>TOTAL DEVICE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Isolation Surge Voltage(1)</td>
<td>$V_{DO}$</td>
<td>7500</td>
<td>Volts</td>
</tr>
<tr>
<td>(Peak ac Voltage, 60 Hz, 1 Second Duration)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Power Dissipation @ $T_A = 25^\circ$C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Derate above 25°C</td>
<td>$P_D$</td>
<td>330</td>
<td>mW</td>
</tr>
<tr>
<td>Junction Temperature Range</td>
<td>$T_J$</td>
<td>-40 to +100</td>
<td>°C</td>
</tr>
<tr>
<td>Ambient Operating Temperature Range</td>
<td>$T_A$</td>
<td>-40 to +85</td>
<td>°C</td>
</tr>
<tr>
<td>Storage Temperature Range</td>
<td>$T_{ES}$</td>
<td>-40 to +150</td>
<td>°C</td>
</tr>
<tr>
<td>Soldering Temperature (5 s)</td>
<td>$T_s$</td>
<td>260</td>
<td>°C</td>
</tr>
</tbody>
</table>

---

1. Isolation surge voltage, $V_{DO}$, is an internal device dielectric breakdown rating.
   For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
I am Rincy Valsalan and this thesis is the concluding part of Master's program in Electronic Design at Halmstad University, Sweden. The main aim of my project is to develop a hardware implementation of the electronic water heater by using microcontroller Atmega328p.