DESIGN AND CONSTRUCTION OF MICROCONTROLLER TIMER SOCKET OUTLET

DEPARTMENT OF ELECTRICAL/ELECTRONIC

ENGINEERING. FACULTY OF ENGINEERING

FEBUARY 2025

DESIGN AND CONSTRUCTION OF MICROCONTROLLER TIMER SOCKET OUTLET

BY

OKONKWO KOSISOCHUKWU JEFFERSON ENG1905366

OKUNZUWA OSARETIN FRANCIS ENG1905368

OMOROGIEVA EDOGIAWERIE DESINY ENG1905369

OZABOR MARVELOUS NOSA ENG1905390

PROJECT SUBMITTED TO THE DEPARTMENT OF ELECTRICAL/ELECTRONIC ENGINEERING. FACULTY OF ENGINEERING

UNIVERSITY OF BENIN, BENIN CITY

FEBUARY 2025

CERTIFICATION

We, the undersigned, certify that this work was carried out by Okonkwo Kosisochukwu Jefferson, Okunzuwa Osaretin Francis, Omorogieva Edogiawerie Desiny, Ozabor Marvelous Nosa in the Department of Electrical and Electronics Engineering, Faculty of Engineering, University of Benin, Benin city. For the requirements of the award of Bachelor of Engineering (B.ENG.) Degree in Electrical and Electronics Engineering

--- ---

Engr. G. O. Imhabekhai Date

(Project Supervisor)

--- ---

Prof. K.O. Ogbeide Date

(Head of Department)

DEDICATION

I dedicate this work to God, whose unwavering support, encouragement, and guidance have been invaluable throughout this journey.

ACKNOWLEDGMENTS

To God: Your divine grace and unwavering presence in our lives have been a constant source of strength and inspiration. We are profoundly thankful for the blessings, wisdom, and inner strength that you have provided, guiding us through the journey

To the Head of Department (HOD), Engr. Prof. K. O. Ogbeide. Your leadership, encouragement and commitment to academic excellence have been instrumental in shaping our academic and research path. Your guidance and support have helped us navigate the challenges and complexities of this project.

To our supervisor: we extend our sincere appreciation to our supervisor, Engr. G. O. Imhabekhai, for his invaluable mentorship, patience and expertise. Your guidance has been instrumental in refining our research skills and shaping the direction of this project. Your dedication to our success has been a driving force behind our achievements.

To our Friends: To our friends who stood by us, listened to our concerns, and provided words of encouragement during the challenging times of this project, your friendship means the world to us. Your belief in our abilities and your willingness to be a pillar of support have been immeasurable.

ABSTRACT

This project “Design and Construction of a microcontroller timer socket outlet, that takes over the task of human intervention in electrical and electronic appliances when connected to power. It can also be used as home automation system to ensure energy saving by shutting off from the main supply in such a way that it will switch off the loads once it counts down to zero.

The working principle is such that a preset time, usually between 1minute and 1440 minutes, is set using the appropriate buttons and made to start operation when the start button is hit. The preset time counts down to zero and disconnects automatically from the main supply to conserve power usage. The microcontroller does the countdown which is displayed on the LCD . At the completion of the task, audio and visual signals are indicated to signify completion.

The microcontroller-driven timer socket outlet project effectively showcased the capability to automate electrical devices according to set time intervals. Utilizing a microcontroller paired with a real-time clock (RTC) module, the system enabled users to program specific activation and deactivation times for connected appliances, promoting energy savings and ease of use. Tests verified precise timing functionality and consistent performance across different loads. In summary, the project successfully delivered an affordable and intuitive tool for automating electrical devices, with potential uses in smart homes, energy conservation, and IoT applications. Enhancements such as wireless features and mobile app compatibility could further expand its functionality in the future.

TABLE OF CONTENT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

In an era marked by rapid technological advancement and increasing energy demands, the need for efficient power management solutions has become more important. Many households use a variety of electrical appliances such as fans, heaters, and kitchen devices. Unfortunately, these appliances are often left on longer than necessary, leading to higher electricity bills and potential safety hazards. Addressing these challenges requires innovative solutions that combine automation, precision, and ease of use. One such solution is a microcontroller timer socket outlet.

A timer is a device used to measure the amount of time taken by a process or activities. It can also be defined as a device that starts or stops a machine working at a particular time (Oxford Dictionary, 2017). Timers are frequently used to run process/operation with predefined time interval. The most popular type of timer is the programmable timer that has various facilities to configure the timer by users. A Microcontroller plays a crucial role in fostering this innovation.

1.0 BACKGROUND OF STUDY

The increasing global concern for energy conservation and efficient power management has led to significant advancements in home electrical systems. The Traditional electrical outlets rely on manual switching, which often proves inconvenient, especially when appliances need to operate at specific times. For instance, users may forget to turn off devices, leading to energy wastage or potential safety hazards. The introduction of microcontroller-based timer sockets eliminates the need for human intervention by automating the power control process. These devices ensure that

appliances are turned on or off according to a predefined schedule, making them ideal for applications such as lighting systems, irrigation pumps, heating elements, and security appliances. This innovation allows users to set specific time durations for which their appliances remain powered, effectively preventing unnecessary power usage when devices are not in active use.

The concept of "vampire power," which refers to the energy consumed by appliances when they are plugged in but not actively in use, is particularly relevant here. This occurs because some devices, even when turned off, are not completely disconnected from the power source. However, simply switching off devices at the power point can effectively control vampire power, eliminating the need to unplug all devices entirely. The Microcontroller Timer Socket Outlet is a solution that helps conserve vampire power. It allows electrical and electronic appliances to be connected to the socket outlet and programmed to receive power only for a specific duration.

A typical microcontroller timer outlet socket comprises several key components, including a microcontroller unit (MCU), a real-time clock (RTC) module, a relay switch, an LCD display for user interaction, and control buttons for setting time parameters. The device operates by accepting user-defined time settings and triggering the relay to connect or disconnect power accordingly.

Energy efficiency has become a critical concern in modern society, with improper power management leading to significant energy wastage in both residential and commercial settings. In homes, this results in higher electricity bills and unnecessary wear and tear on appliances, while

in industries, it translates to increased operational costs. The development of intelligent power sockets that automate energy management offers a cost-effective and safe solution to these challenges. This research is centered on the design and construction of a microcontroller-based timer outlet socket, which aims to tackle energy conservation challenges by offering an automated, user-friendly, and efficient solution. The project seeks to minimize unnecessary power consumption in households and other environments, providing a practical approach to reducing energy waste.

1.1 PROBLEM STATEMENT

Energy wastage is a prevalent issue in many households and industries, primarily due to the continuous operation of electrical appliances beyond their required duration. This not only leads to increased electricity bills but also accelerates the wear and tear of appliances, reducing their lifespan. The absence of automated control mechanisms means that users must manually switch appliances on and off, which is often impractical and inefficient.

Moreover, leaving appliances running for extended periods poses significant safety risks, such as overheating and electrical fires. Many users forget to turn off devices, increasing the likelihood of such hazards. Existing timer switches often lack flexibility and modern features, such as real-time adjustments, remote control, or user-friendly interfaces, making them less effective in addressing these challenges.

For example, in agricultural settings, water pumps must operate at specific times to ensure proper irrigation. Manual control of these pumps is labor-intensive and prone to errors, leading to

inefficient water usage. Similarly, in commercial settings, security lighting needs to follow a fixed schedule, and manual operation may result in human errors, causing unnecessary power consumption. The lack of a reliable automatic power control solution means that users continue to experience inefficiencies in energy use, highlighting the need for a more advanced and user-friendly system.

To address these challenges, this research focuses on designing and constructing a microcontroller-based timer outlet socket. The system will allow users to set operational time limits, automate power management, and integrate remote monitoring capabilities to enhance safety and efficiency.

1.2 AIM

The aim of this project is to design and build an automatic socket outlet with an operating timer built into it using a microcontroller.

1.3 OBJECTIVES

The objectives of this project are as follows:

1. To promote energy conservation in residential settings.
2. To minimize and prevent electrical hazards within households.
3. To regulate and manage the consumption of electrical power in homes.
4. To lower electricity bills by conserving and controlling power usage.

1.4 SCOPE OF PROJECT

This project focuses on the design and construction of a timer-controlled outlet socket using a microcontroller. The project comprises of the following key aspects:

a) The ATmega328P microcontroller is used as the core control unit.

b) Integration of an RTC module for precise time management.

c) Development of a relay circuit for switching power on and off.

d) Incorporation of an LCD screen to display real-time status and user settings.

e) Use of push buttons for user input and control.

f) Implementation of a stable power supply unit to ensure constant operation.

1.5 METHODOLOGY

The methodology for this project is categorized into four key areas: energy conservation, electrical safety, power regulation, and cost efficiency.

DESIGN APPROACH FOR ENERGY CONSERVATION: To enhance energy conservation in residential and commercial settings, the system was designed with an automated power management feature. The following steps were undertaken:

a) Microcontroller Integration: The ATmega328P microcontroller was selected for its energy-efficient operation and processing capabilities.

b) Real-Time Clock (RTC) Module: Implemented to allow precise scheduling of appliance operation, reducing energy wastage.

c) User Input and Display Interface: Push buttons and an LCD screen were included for users to set countdown timers and monitor system status.

d) Automated Relay Circuit: A relay was integrated to control power flow, automatically disconnecting appliances after the preset time elapsed.

e) Performance Testing: The system was tested under different operational scenarios to quantify energy savings compared to conventional outlets.

DESIGN APPROACH FOR ELECTRICAL SAFETY: To minimize and prevent electrical hazards, the following safety mechanisms were incorporated into the design:

a) Automatic Shutdown Relay: The microcontroller was programmed to deactivate the relay in case of an electrical fault or prolonged operation.

b) Battery Backup System: A 5V backup battery was included to maintain system settings and ensure continuity during power outages.

c) Warning Indicators: An LCD indicator was implemented to notify users of system status of the timer.

d) Safety Compliance Testing: The prototype was tested under various fault conditions to ensure operational reliability and adherence to safety standards.

DESIGN APPROACH FOR POWER REGULATION: Efficient power regulation is essential for system stability and longevity. The following methodologies were implemented:

a) Load Monitoring and Optimization: The system was tested with different electrical loads to evaluate its efficiency in maintaining stable power delivery.

b) Efficient Circuit Design: Components were arranged to minimize energy losses and enhance overall power efficiency.

c) Firmware Optimization: The microcontroller firmware was designed to enter low-power mode when idle, minimizing unnecessary power consumption.

DESIGN APPROACH FOR REDUCING ELECTRICITY BILLS: In the aspect of lowering electricity bills by conserving and controlling power usage, the following methods were implemented:

Programmable Timer: Automatically disconnects appliances after a set duration to prevent energy waste.

• Microcontroller Automation: Ensures precise control of power cycles, reducing unnecessary consumption.

• Efficient Circuit Design: Uses low-power components to minimize energy losses.

• Real-Time Monitoring: LCD display provides status updates for better user control.

1.6 MODEL OF PROJECT

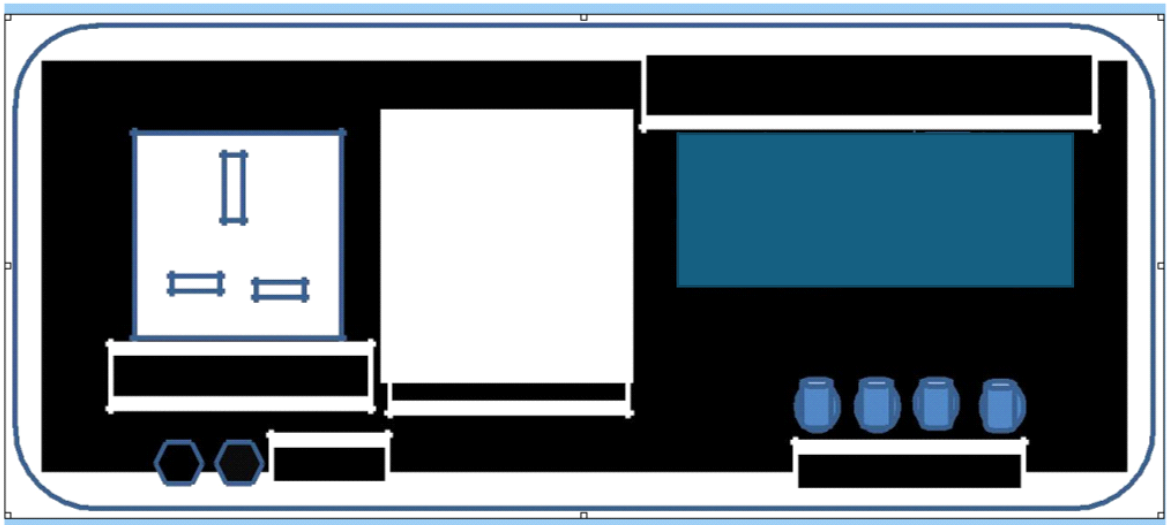


Figure 1.1 Model of the project

1.13-amp socket

2.LED

3.LCD

4. Buttons

The 13-amp socket is meant for appliances rated 13 amps and below.

The 7-segment display serves as a time-spent/remaining indicator. The buttons are meant to initiate/set/reset the circuit.

The LEDS acts as indicators showing the action has been completed.

1.7 RELEVANCE OF WORK

This project holds significant relevance in the fields of home automation and industrial control systems. The microcontroller-based timer outlet socket offers numerous benefits, including:

Energy Efficiency: By automating the operation of appliances, the system reduces unnecessary power consumption, contributing to energy conservation.

Convenience: The elimination of manual switching enhances user comfort and ensures that appliances operate according to a predefined schedule.

Cost Savings: The system helps lower electricity bills by ensuring that devices operate only when needed.

Safety Enhancement: The inclusion of safety features minimizes the risk of electrical hazards caused by prolonged use of appliances.

CHAPTER TWO

LITERATURE REVIEW

2.1 BRIEF HISTORY OF MICROPROCESSORS

The first microprocessor was created by Intel, a relatively small company at the time (short for Integrated Electronics), in the early 1970s. Initially, the chipset was developed for a Japanese company named Busicon, but they decided not to purchase it. Left with development costs and no buyer, Intel chose to market the chipset as a "general-purpose" micro processing system, targeting applications that traditionally used digital logic chips. This move proved successful, and soon after, Intel introduced the 4004, a general-purpose 4-bit microprocessor.

In 1974, Intel unveiled the second-generation microprocessor, the 8008, which was manufactured as a single chip and offered greater power than its predecessor. Shortly after, the Intel 8080 was introduced. Both the 8008 and 8080 utilized NMOS technology and were designed to operate on a single +5V power supply.

Around the same period, Motorola introduced its first microprocessor, the 6800, which was also an 8-bit processor with processing capabilities comparable to the Intel 8080. However, the architectures of the Intel 8080 and the Motorola 6800 differed significantly. The Intel 8080 used a register-based architecture with 16-bit registers called AX, BX, CX, DX, and HL. These registers could also function as 8-bit register pairs, allowing the AX register to be split into AH (higher byte) and AL (lower byte). Similarly, the BX, CX, DX, and HL registers could be divided into BH, BL, CH, CL, DH, DL, H, and L, providing flexibility in handling 8-bit data.

Another notable feature of the 8080 was its distinct I/O map. This meant that for byte-wide input and output operations with hardware, specific instructions were required: **IN** for reading from byte-wide input ports and **OUT** for writing to byte-wide output ports. Accessing memory, on the other hand, involved a separate memory map and typically utilized the **MOV** instruction.

The Motorola 6800 microprocessor used a mechanism known as "Memory Mapped I/O," in which memory and byte-wide input/output operations used the same memory map. Its register set was rather small, consisting of two 8-bit accumulators (known as A and B) and a 16-bit index register called X. Despite having fewer registers, the 6800 supported a variety of addressing modes, which effectively compensated for the limited number of registers and also simplified programming tasks.

To read data from memory or I/O, the LDAA instruction is used, while writing data to memory or I/O requires the STAA instruction. Access to the X register is managed through its own set of instructions, specifically LDX and STX.

Intel and Motorola have preserved the core architectural differences in their subsequent microprocessor developments.

In the Intel series, the 8080 evolved into the 8085 (an 8-bit processor similar to the 8080), followed by the third-generation 16-bit 8086 microprocessor. The 8088, an 8086 variation with

an 8-bit data bus, was used in the earliest IBM PCs, making it simple to connect to the 8-bit peripheral devices that were widespread at the time. This progression continued with the 80186, 80286, 80386 (a 32-bit processor), and 80486, culminating in the Pentium series of microprocessors (64-bit processors) available today. The 80x86 and Pentium processors were designed primarily for personal computers and support extensive memory maps .

Similarly, Motorola's microprocessor range evolved from the 6800 to the 6809 (8-bit), then to the 68000 (16-bit), followed by the 68010, 68020, and 68030, which were used in many workstations and Apple Macintosh computers .

Over time, the Intel 8080 core processor was adapted for use in various microcontrollers, such as the 8048 and 8051. The 8051 microcontroller is still in production today, now produced by Philips. Intel has since developed more advanced microcontrollers that surpass the complexity of the original designs .

Motorola followed a similar path, creating microcontrollers based on the 6800 architecture, including the 6805, 6808, and 6811, which are still in use today. As a result, many of today's popular microcontrollers are based on two core architectures: 8080 and 6800. Other manufacturers, like Rockwell, produced microprocessors based on the 6800 architecture (e.g., the 6502), while Zilog developed the Z80, which was based on the 8080 architecture. The emergence of newer architectures, such as Harvard architecture and Reduced Instruction Set Computers (RISC), has led to the development of microcontrollers like the Microchip PIC.

2.2 BRIEF HISTORY OF SOCKET OUTLET

2.2.1 THE BRITISH RING FINAL CIRCUIT SYSTEM AND BS 1363:

The 13 A plugs, socket-outlets, connection units, and adaptors plug and socket-outlet system were introduced in the UK in 1947 after years of deliberation. The Minister of Works and Planning established 'The Electrical Installations Committee,' which met for the first time in June 1942. The committee was tasked with conducting a thorough and coordinated review of building techniques to guide post-war construction efforts. Its mandate included reviewing existing information and proposing improvements. Between 1942 and 1944, the committee held 22 meetings, culminating in the publication of 'Post-War Building Study No. 11 - Electrical Installations' in January 1944, with a final report in July 1944.

This study played a pivotal role in shaping the UK's ring final circuit system and led to the development of the BS 1363 13-ampere fused, flat-pin plug and socket-outlet system by BEAMA. The study is highly detailed, covering all aspects of electrical installations, not just plugs and sockets. It provides clear insights into the reasoning behind the development of the current plug and socket system. Remarkably forward-thinking, the study addresses topics such as energy efficiency and environmental protection. It also proposes innovative ideas, including single-pole fusing, a compact consumer control unit design, and the cooker control unit that remains familiar today. Additionally, it criticizes the haphazard placement of equipment of

various types and sizes on walls, emphasizing the need for convenience and aesthetics—a recommendation that remains relevant in many premises today.

2.3 PROGRAMMABLE CONTROL TIMER

2.3.1 INTRODUCTION

With rising energy costs and increasing power shortages, conserving electricity has become more critical than ever. Consumers can adopt various strategies to reduce their electricity consumption, particularly with household appliances. For any economy to progress, effective time management and the responsible use of electrical power are essential. This can be achieved by minimizing waste in the limited power supply.

The programmable control timer allows users to automatically turn electrical appliances on or off, providing convenience and efficiency. Users can preset the device to a specific time using the set button, which activates the time input buttons. Once the desired time is set and displayed on the digital screen, pressing the count button initiates the countdown. When the timer reaches zero, the connected appliance will either turn on or off, depending on the user's settings.

2.3.2 Review of Past Work

The development of automated socket outlets and timer systems has been the subject of numerous studies aimed at improving energy efficiency, safety, and convenience in electrical appliance usage. Many researchers have explored different approaches to designing and implementing such systems, often integrating microcontrollers like Arduino and PIC to automate the power control process. One notable study on automated timer socket systems was conducted by *G. Chakra Mahesh Babu et al.*, where the researchers developed a microcontroller-based

socket outlet that allows users to predetermine the ON/OFF duration for connected electrical devices. Their system incorporated an Arduino UNO, a relay module, a Real-Time Clock (RTC), and an LCD display, enabling users to program appliances to operate for a specific time before automatically turning off. The project was tested on various household appliances, such as mobile phone chargers, toasters, and electric cookers, and successfully demonstrated its ability to reduce energy waste and enhance convenience. A similar project focused on the design and construction of a microcontroller-based timer socket outlet. This research explored the importance of integrating automation into electrical socket outlets to prevent unnecessary power consumption. The system allowed users to program appliances to run for specific durations, mitigating the risk of electrical hazards caused by prolonged use. This study emphasized the need for energy conservation in both residential and industrial settings, particularly in cases where users forget to turn off appliances manually. The implementation involved a microcontroller, a keypad for input, an LCD for displaying timer settings, and a relay for switching the socket outlet. Another relevant research study, conducted by *N.O. Adelakun a*nd *S.A. Omolola*, examined the use of Arduino-based automated socket outlets ran by real-time power. Their project proposed a system that monitors and regulates the power supply to socket outlets, ensuring that electrical devices are only powered for the required duration. The study highlighted the importance of not just generating electricity efficiently but also implementing conservation techniques to minimize energy wastage. By integrating Arduino programming, relay switching, and time-based automation, their system provided a cost effective and user-friendly solution for managing household and industrial electricity consumption Collectively, these studies reinforce the importance of automated timer sockets in energy conservation and safety enhancement. They highlight the widespread use of microcontrollers, RTC modules, LCD displays, and relays in

designing these systems, ensuring precision and ease of use. The research also emphasizes how automation can help reduce human error, such as forgetting to turn off appliances, while also contributing to lower electricity bills and reduced carbon emissions. The continuous advancements in microcontroller technology suggest that future designs may incorporate wireless control, IoT integration, and remote monitoring, making these systems even more efficient and adaptable for smart home applications.

This study focuses on the design and construction of a programmable control timer that can count down from a predetermined time to zero and turn the associated load on or off. The programmable function allows it to accept a variety of time inputs, with this prototype enabling a maximum of 86,400 seconds (1440 minutes).

The design utilizes a 4060B CMOS integrated circuit, which operates using a16,000 Hz crystal oscillator. The input frequency (16,000 Hz) is divided internally by the 4060B circuit to produce a 2 Hz signal. An additional flip-flop is incorporated to generate a 1 Hz pulse (equivalent to a 1-second interval). This pulse is then processed through control logic before being fed into a down-counter unit. The counter unit consists of three BCD (Binary-Coded Decimal) down counters, each preloaded with a specific value via the control logic. The control unit, which consists of latches or flip-flops, is coupled to external switches that offer precise input controls. The counter unit counts down from the preset value, and when it reaches zero, a zero-detection circuit signals the control unit, triggering the connected load to turn on or off. The timer controls the power supply to the load through an external relay circuit, ensuring efficient switching and the ability to handle higher loads.

A key feature of this design is the use of a crystal-based CMOS oscillator. Unlike a 555 timer, which lacks precision, a crystal oscillator ensures accurate and stable timing. In this design, the oscillator operates at 16,000Hz, and through the internal frequency dividers of the integrated circuit, a precise 1 Hz clock signal is generated for the counters. This timing technique reduces frequency instability and ensures consistent operation.

The design is based on readily available and cost-effective electronic components, making the overall construction economical.

The programmable control timer's unique feature is its ability to automatically control the electrical load connected to it after a preset time. The system is divided into six functional units, as illustrated in the block diagram in Figure 2.1.

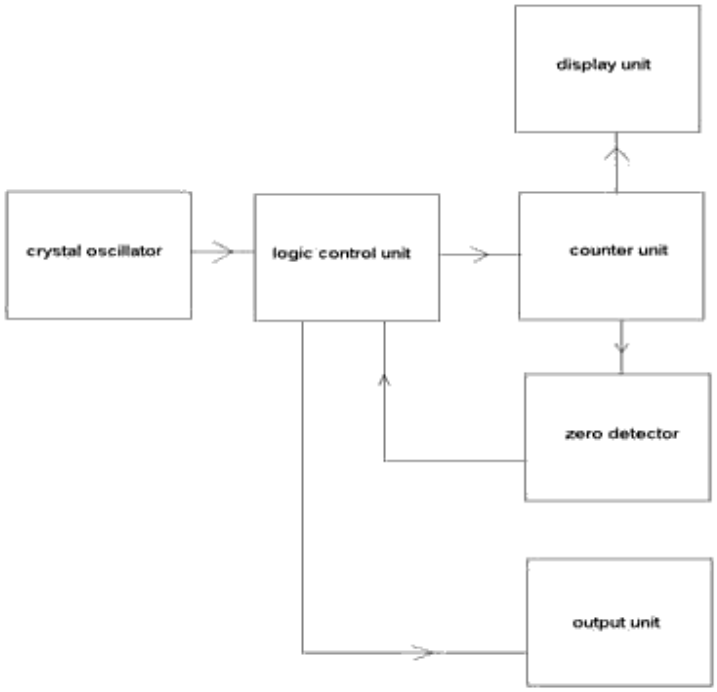


figure 2.1 The block diagram of the programmable control timer.

2.3.3 CRYSTAL OSCILLATOR

Figure 2.3 shows the logic diagram of the logic control unit, which is the foundation of the system architecture. This setup ensures reasonable frequency stability.

The 4060B has an internal 14-stage counter/divider logic that splits the primary frequency into smaller numbers. This enables the 4060B to generate ten distinct frequencies at once. All outputs are buffered to provide for high load driving capability. Pin 12 must be grounded so that the device can function. The frequency of the ten outputs (fax) is defined by the formula below:

$fax = fm/2x$

Here, x represents the Q value of a specific output terminal, and fm is the crystal frequency.

For this project, the crystal frequency is 16000 Hz. Therefore, the frequency output at pin 3 (Q14) is calculated as:

$FQ14 = 16000/8000 = 2Hz$

Also, pin 1's (Q12) frequency output is: $FQ12 = 16000/2000 = 8$ Hz

The circuit design makes use of these two output terminals (pins 1 and 3). The precise frequencies of 2 Hz and 8 Hz produced by this unit are necessary for this design to function.

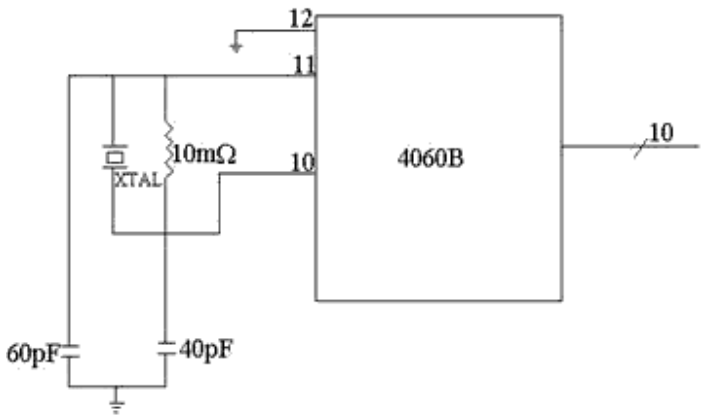


figure 2.2 Configuration of the 4060B crystal oscillator.

2.3.4 LOGIC CONTROL UNIT

Figure 2.3 illustrates the logic diagram of the logic control unit, which serves as the core of the system design. This unit is responsible for further dividing the frequency outputs from the crystal oscillator and controlling the counters. It also incorporates latch logic functions to manage operations effectively.

The logic control unit consists of four latches. With our choices, the 4013B 1 and 4013B 2 are setup as divide-by-2 latches. Specifically, the 4013B 1 divides the 2 Hz pulse from the crystal oscillator to produce a 1 Hz signal. This resulting 1 Hz signal is used for counting purposes, with each pulse representing a 1-second interval.

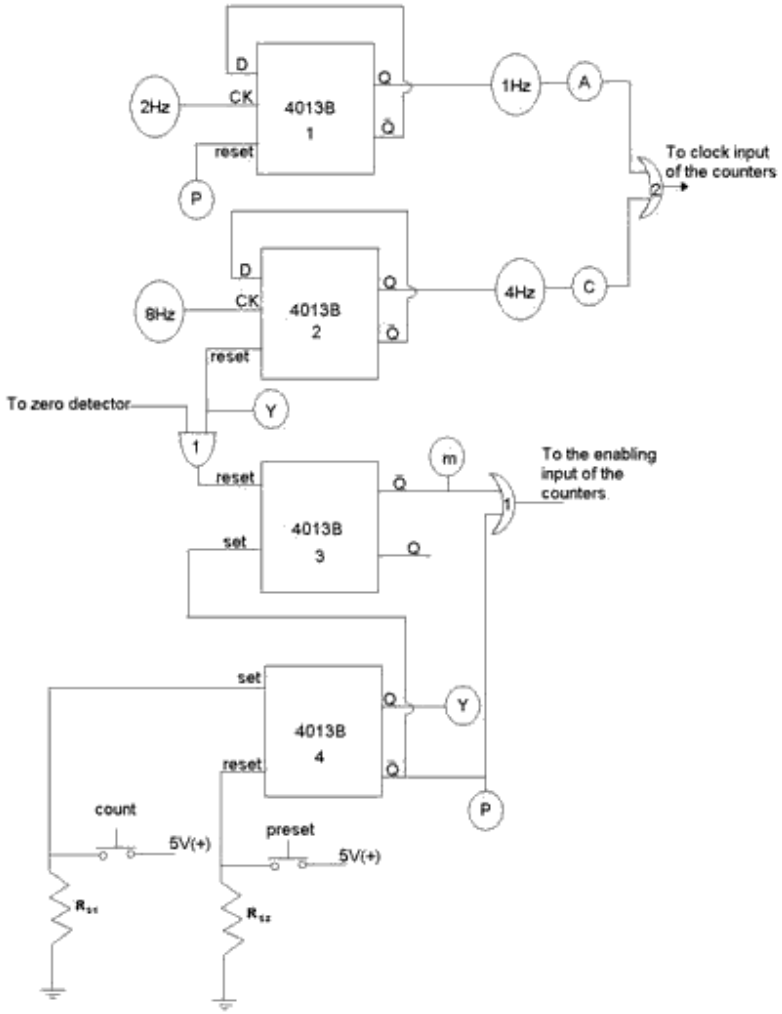


Figure 2.3 The logic control unit.

This timing mechanism provides a 1-second interval for the counter unit. The 4013B 2 latch separates the 8 Hz signal from the crystal oscillator into a 4 Hz pulse that is utilized to preset the counter. Their reset terminals play a key role in this control, as raising a reset input to a HIGH logic level disables the latch.

The 4013B 3 and 4013B 4 latches are responsible for device control. The 4013B 3 latch manages output control and responds immediately when the counter reaches "000." The 4013B 4 latch is the primary control unit, overseeing the entire logic system. It includes external inputs connected to two soft-touch buttons. When either button is pressed, a HIGH logic signal is sent to the corresponding latch input. One button is designated for COUNT control, while the other handles PRESET functions. These buttons operate based on the basic SR latch or flip-flop truth table.

When the PRESET button is pressed, the 4013B 4 latch is reset. This results in the Q- output becoming HIGH, while the Q output remains LOW. The terminal P output disables 4013B 1 and enables 4013B 2. Consequently, OR gate 2 receives the 4 Hz signal to clock the counter inputs, while the 1 Hz signal is interrupted. OR gate 1 remains HIGH due to the terminal P output.

During the PRESET action, AND gate 1 is disabled, preventing accidental triggering of the output by the zero detector. When you hit the COUNT button, the 4013B 4 latch is set. Its Q output switches from logic 0 to 1, and Q- becomes LOW. In this state, OR gate 1's output is LOW. Now, 4013B 1 is enabled and 4013B 2 is disabled. OR gate 2 produces a 1 Hz signal, which gives the counter unit a 1-second interval to count down.

The output of AND gate 1 becomes HIGH when terminal Y is HIGH and a HIGH signal is received from the zero detector. Terminal Y is HIGH when 4013B 4 is in COUNT mode, meaning the Q output is HIGH .

The PRESET mode is used to input a specific digit or time into the counter unit at a faster rate (0.25-second intervals). In contrast, the COUNT mode allows the counter to operate freely, decrementing every one second.

2.3.5 COUNTER UNIT

The counter unit, as shown in Figure 2.4, consists of three 4029B BCD down counters. These counters are systematically and synchronously cascaded, allowing the system to count up to a maximum of 86,400 seconds (1,440 minutes). The cascading is designed so that each counter can be individually preset to a specific digit using dedicated buttons. Each counter has its own preset button.

The 4029B counters are configured such that pin 7 (carry-out) goes LOW when a counter reaches its lowest digit, i.e., "0." This indicates the end of the counting cycle for that particular counter.

To preset each counter, a 2-input AND gate is used. The circuit includes three such AND gates (AND gate 2, 3, and 4), one for each counter. These gates are connected as illustrated in Figure 4. One input of each AND gate is connected to pin 7 of the appropriate 4029B counter, and the other input is coupled to a positive voltage (5V) via resistors, ensuring a HIGH logic state. This configuration allows for the regular flow of carry-in to carry-out signals, resulting in flawless counter cascading.

When the 4013B 4 latch is in PRESET mode, terminal Y is LOW. This causes the outer terminals of the switches to also be LOW. Pressing a switch sends a LOW logic signal to the corresponding AND gate, activating pin 5 (carry-in or enabling input) of the respective counter. Since the counters are synchronously cascaded and share a common clock signal, the counter associated with the active switch begins counting down at a speed of 4 Hz. This process has been previously explained in the control logic unit section. The PRESET action is disabled when the COUNT button is pressed, as terminal Y transitions to a HIGH logic level.

In COUNT mode, the necessary LOW logic signal at the AND gates are no longer possible, preventing further presetting. The display unit receives the BCD output from the counter unit, and the three carry-out terminals from this unit are connected to the zero detection unit.

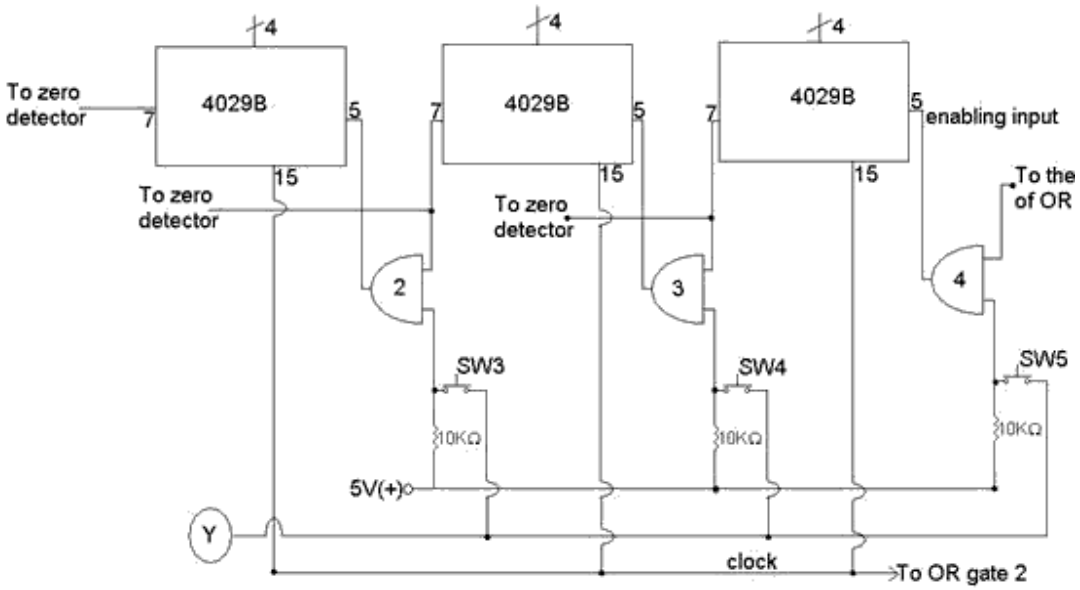


figure 2.4 The counter unit

2.3.6 DISPLAY UNIT

The display unit consists of an LCD (Liquid Crystal Display) operating at 3.7V, as illustrated in the display circuit shown in Figure 2.5. The LCD is integrated into the circuit to directly display the BCD (Binary-Coded Decimal) outputs from the counters.

Under normal operating conditions, the LCD operates at 3.7V, which is compatible with the logic levels of the microcontroller or other driving circuitry. The LCD's low power consumption eliminates the need for additional current-limiting resistors, simplifying the circuit design.

To ensure proper operation of the LCD, the control pins of the display are configured according to the manufacturer's specifications. For example, the enable (E), read/write (R/W), and register select (RS) pins are connected to the microcontroller or control logic to manage data and command transfers. The VSS pin is connected to ground, while the VDD pin is supplied with 3.7V for power.

Configuration of Control Pins:

i. Pins 3 and 4 (Enable and Register Select): Set to a logical high state (positive) during data/command transfers.

ii. Pin 5 (Read/Write): Set to a logical low state to enable writing to the LCD.

This configuration ensures proper communication between the control logic and the LCD, enabling accurate and reliable display of the countdown timer's output.

Under normal operating conditions, the expected voltage across each LED segment is 2.7V, with a current of 9.5mA to achieve this, the voltage drop across the series resistor is calculated as follows:

5V - 2.7V = 2.3V

The resistance value R5 is determined using Ohm's Law:

R5 = 2.3/ 9.5 x 10 = 242.1 Ω

A standard resistor value of 220Ω is used in the circuit, as it is the closest available value to the calculated 242.1Ω.

Additionally, pins 3 and 4 of the 4511B IC are set to a logical high state (positive), while pin 5 is set to a logical low state, as specified in the device's datasheet. This configuration ensures proper operation of the LCD

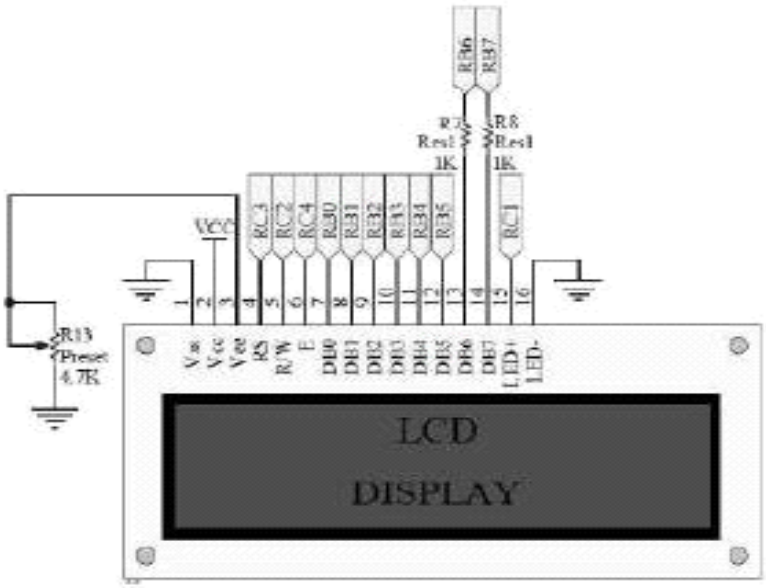


figure 2.5 The circuit of a liquid crystal display(LCD)/display unit

2.3.7 ZERO DETECTOR

The zero detector is a circuit designed to signal when the counters reach a zero state. It is constructed using a simple 3-input OR gate and an inverter, as shown in the logic diagram in Figure 2.6.

The inputs of the OR gate are connected to the carry-out signals from the three counters, as depicted in Figure 4. When all three carry-out signals are low, the output of the OR gate remains low. The inverter then converts this low logic level into a high logic level. This high signal is used to trigger the output through a 4013B flip-flop, as described in.

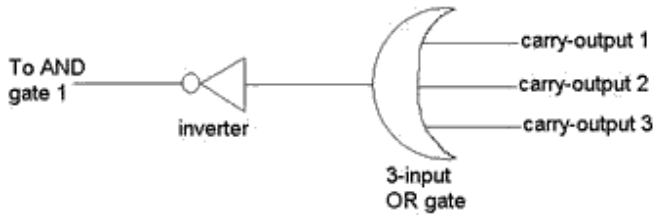


figure 2.6 The zero detector unit

2.3.8 OUTPUT UNIT

The output unit consists of a **relay switching circuit** that responds to signals from the control logic unit, specifically from the **4013B IC** (refer to Figure 2.3). The output unit, as shown in Figure 2.7, includes the following components:

1. NPN Switching Transistor (BC548): Used to control the relay.

2. Diode: Protects the transistor from voltage spikes.

3. 12V Relay: Switches the load based on the transistor's state.

Operation of the Output Unit

i. The base of the transistor is connected to point **m** of the **4013B IC**.

ii. When terminal **m** is **HIGH**, the transistor is switched **ON**, energizing the relay. This connects contact **O2** to **O3**.

iii. When terminal **m** is **LOW**, the transistor is switched **OFF**, de-energizing the relay. This connects contact **O2** to **O1**.

iv. The **relay** has a coil resistance of **400 Ω**.

v. The transistor has a typical **current gain** (hfe) of **100** .

Current Calculations

1. **Collector Current (IC)** :

The collector current is calculated using Ohm's law:

IC = VCC/Rrelay = 12V/400 Ω = 0.03A = 30mA

2. **Base Current (IB):**

The base current is determined using the transistor's current gain:

IB = IC/hfe = 30mA/100= 0.3mA

3. **Base Resistor (RB):**

The base resistor is calculated to ensure proper transistor operation:

RB = VBB - VBE/IB = 5V - 0.6V/0.3 x 10^-3A = 14.67 kΩ

A **preferred value of 10 kΩ** is used for RB to protect the transistor and ensure reliable operation.

Protection Mechanisms

Diode:

i. The diode is connected across the relay coil to protect the transistor from back electromotive force (**back EMF**) generated by the relay during switching.

ii. This back EMF can damage the transistor, so the diode acts as a **flyback diode** to cancel out the induced voltage spike.

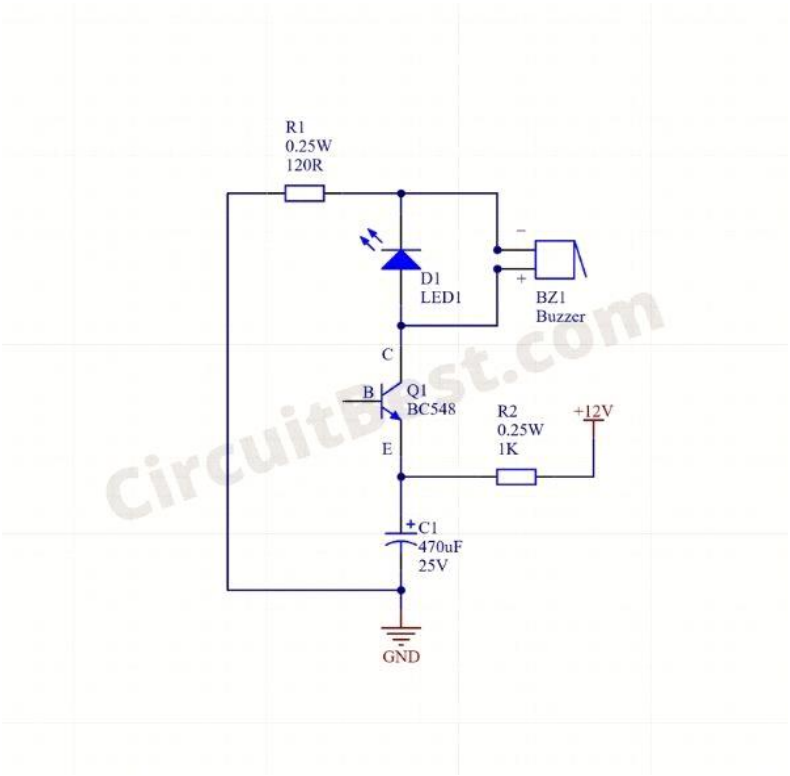


Figure 2.7 The output Unit

2.4 THE ATMEGA328P MICROCONTROLLER

The ATmega328P is a popular microcontroller known for its affordability, high performance, and energy efficiency. It is a single-chip computer that integrates a processor core, memory, and programmable input/output (I/O) ports, making it ideal for embedded systems. Microcontrollers like the ATmega328P are designed specifically for embedded applications, providing both computational and control capabilities.

The ATmega328P is built on the AVR architecture with a Reduced Instruction Set Computing (RISC) design. This implies it executes instructions in a straightforward, single-operation way, as opposed to Complex Instruction Set Computing (CISC) architectures, which integrate numerous operations into a single instruction. The ATmega328P has a Harvard architecture**, in which program and data memory are accessed via different buses. This separation enhances performance and bandwidth compared to the von Neumann architecture, where program and data share the same memory bus.

In this project, the ATmega328P is used as an embedded system to control an LCD display, manage the actions of LEDs, and handle input from push buttons.

Pin Configuration and Ports

The ATmega328P is commonly packaged as a 28-pin PDIP (Plastic Dual In-line Package) with a 32-pin TQFP (Thin Quad Flat Package). It features 23 programmable I/O pins, which are divided into three ports: Port B, Port C, and Port D

i. Port B: 8-bit port (PB0 to PB7)
ii. Port C: 7-bit port (PC0 to PC6)
iii. Port D: 8-bit port (PD0 to PD7)

The direction of each pin (input or output) is determined by the corresponding Data Direction Register (DDRx), where x represents the port (B, C, or D). Setting a 0 to a bit in the DDRx register sets the corresponding pin as an input, whereas putting a 1 makes it an output.
Pins configured as outputs can drive LEDs and other low-power devices directly.

Key Features of the ATmega328P

The ATmega328P offers several features that make it suitable for this project:

Flash Program Memory: 32 KB (16,000 lines of code at 16 bits per word).

SRAM Data Memory: 2 KB for temporary data storage.

EEPROM Memory: 1 KB for non-volatile data storage.

Clock Speed: Up to 20 MHz (commonly operated at 16 MHz in Arduino boards).

I/O Pins: 23 programmable pins, divided into three ports.

Timers/Counters: Three timers (two 8-bit and one 16-bit) for precise timing operations.

Communication Interfaces: Supports USART, SPI, and I2C for serial communication.

Analog-to-Digital Converter (ADC): 10-bit resolution with 6 channels for analog input.

Instruction Set and Performance

The ATmega328P has 131 instructions, most executed in a single clock cycle. This tremendous efficiency is a distinguishing feature of the RISC design. However, program branching needs two cycles due to the microcontroller's pipeline topology.

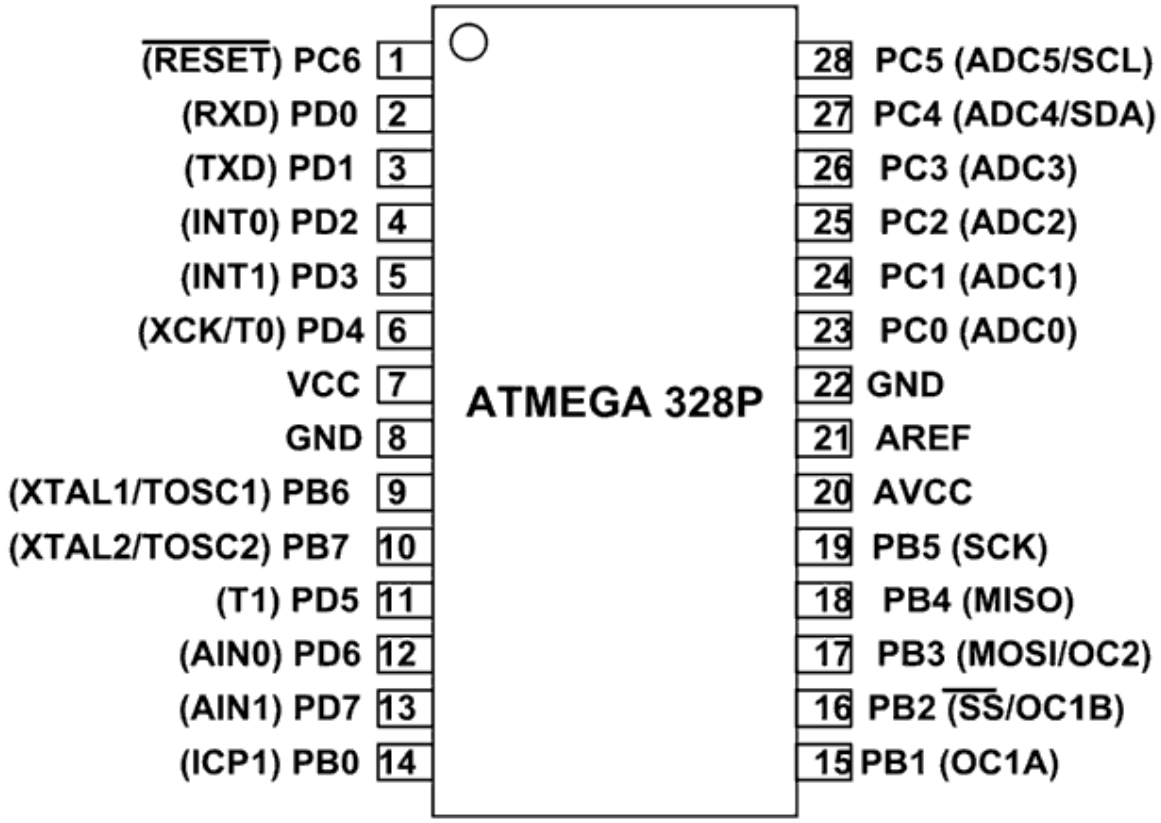


Figure 2.9 ATmega 328P pin diagram

Integration with LCD Display

In this project, the ATmega328P is used to control an LCD display. The LCD provides a versatile and user-friendly interface, capable of displaying alphanumeric characters, symbols, and custom graphics. The microcontroller communicates with the LCD using its I/O pins, typically through a

4-bit or 8-bit parallel interface. The LCD's contrast can be adjusted using a potentiometer connected to the **V0** pin, ensuring optimal visibility under different lighting conditions.

Table 2.1 ATmega328P features

Feature	**ATmega328P Specification**
Flash Program Memory	32 KB (16,000 lines of code at **16-bit words**)
Data Memory (RAM)	2 KB
EEPROM Data Memory	1 KB
Operating Speed	Up to **20 MHz** (commonly operated at 16 MHz in Arduino boards)
Timers	3 timers (two 8-bit timers with one 16-bit timer)
Voltage Range	**1.8V to 5.5V** (operates at 5V in most applications)
Interrupt Sources	**26 interrupt sources** (including external, timer, and serial communication interrupts)
I/O Pins	**23 programmable I/O (input and output) pins** (Port B: 8, Port C: 7, Port D: 8)
Instruction Set	**131 instructions** (most executed in a single clock cycle)
Serial Communication	Supports **USART**, **SPI**, and **I2C** protocols
Parallel Communication	Not natively supported (requires external components or bit-banging using I/O pins)

CHAPTER THREE

DESIGN AND CONSTRUCTION

3.1. INTRODUCTIION

In this chapter, we'll delve into the methodology behind designing and analyzing the various units within a circuit. The design specifications are determined using component datasheets and mathematical relationships derived from relevant textbooks. The circuit is divided into four main sections:

1. Power Supply Unit
2. Display Unit
3. Control Unit
4. Connection/Operation Unit

3.2. DESIGN OF THE POWER SUPPLY UNIT

The power supply unit is a critical component in any electronic system, as it provides the necessary energy for the circuit to function. Without it, no equipment can operate. This unit is constructed using a step-down transformer, fuse, filters, rectifiers, and voltage regulators. The process begins with an AC voltage, which is converted into a steady DC voltage through rectification, filtering, and regulation to achieve the desired fixed DC output.

The power supply is designed to draw electricity from two sources: the mains and a backup battery. The input voltage of 220V/50Hz is stepped down to 12V using a rectifier and filter. This voltage is partially used to charge the backup battery and is further regulated to 5V to ensure the

system retains functionality during power outages. The AC voltage is first connected to a transformer, which reduces the voltage to the required level for the desired DC output. The power supply unit is illustrated in Figure 3.1.

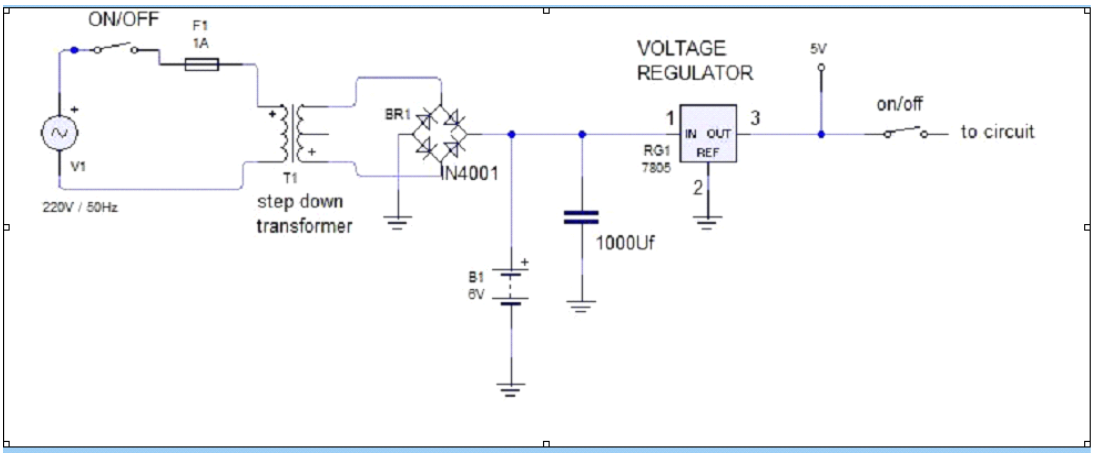


Figure 3.1 The power supply section

A 220v AC/50Hz source is used together with a 5v backup battery. Where the primary supply is 220V/50Hz, the transformer used is a 220/12V.

3.2.1. TRANSFORMER SELECTION

A transformer is a static device that converts electrical power from one circuit to another with the same frequency. It can either step up or step down the incoming voltage, depending on the turns ratio. It consists of two coils, known as the primary coil and the secondary coil (windings) .

The step-down transformer used in this project has the following specifications:

Input Primary Voltage (V1):220 V

Output Secondary Voltage (V2): 12 V

Frequency:50 Hz

Maximum Current Rating:1 A

The transformation constant (K) is given by the formula:

$\frac{E2}{E1} = \frac{N1}{N2} = \frac{I1}{I2} = \frac{V2}{V1} = K$ ……..equation 3.1

Where:

K = Transformation constant

V1 = Primary voltage of the transformer

V2 = Secondary voltage of the transformer

V3 = Voltage from the bridge rectifier before filtering

Vdc= DC output from the rectifier after filtering

For a 220V/12V transformer:

K = 12/220 = 0.05455

The relationship between the root mean square (RMS) voltage and peak voltage is given by:

$Vrms = \frac{VP}{\sqrt{2}}$ .....Equation 3.2

Where Vp is the peak voltage. For the secondary side of the transformer:

Vp2 = 12 x √2 = 12 x 1.4142 = 16.97 V

3.2.2. Rectifier Selection

For a full-wave rectifier, the peak inverse voltage (PIV) must remain within a safe limit for all circuit components. It should not exceed any component's breakdown voltage.

According to Kirchhoff's law, the PIV for a full-wave rectifier is given by:

PIV = 2 x Vp2.....Equation 3.3

Using the peak voltage calculated earlier:

PIV = 2 x 16.97 = 33.94 V

This ensures that the rectifier components can safely handle the voltage levels without risk of damage.

3.2.3. FILTER SELECTION

In a power supply unit, filtration is essential to remove ripples and produce a stable DC voltage. A filter is used at the output of the full-wave rectifier to smooth out the pulsating DC voltage. The primary goal is to provide a steady DC voltage for electronic circuits, which act as a load on the power supply.

Capacitor Filter Operation

In a capacitor filter, the capacitor stores charge during the peak voltage (conduction period) and discharges it to the load during the non-conducting period. This process prolongs the time during which current flows through the load, significantly reducing the ripple voltage.

Ripple Voltage and Ripple Factor

The ripple voltage (Vr) and ripple factor (γ) are critical parameters in filter design. The ripple factor indicates the effectiveness of the filter in smoothing the DC output. The equations for ripple voltage and ripple factor are as follows:

1. Ripple Voltage (Vr):

$Vr = Iload/f.C$

Where:

Iload = Load current (25 mA in this case).

f = Frequency of the AC supply (50 Hz).

C = Filtering capacitor (in farads).

2. Ripple Factor (γ):

$\gamma = Vr/Vdc$

Where:

Vdc = DC output voltage (5V in this case).

The ripple factor could also be expressed as:

$\gamma = 1/4\sqrt{3} \,.\, f C \,.\, RL$

Where:

RL = Load resistance (200 Ω in this case).

Assumptions for Filter Design,

To get effective filtering in the device, the following assumptions are made:

Ripple Factor (γ): Must be less than 2%. For this design, a ripple factor of 1.5% is selected.

Frequency (f): 50 Hz (standard mains frequency).

Load Resistance (RL):Calculated using Ohm's law:

RL = Vdc/Iload = 5V/25 x 10^-3A = 200 Ω

Calculating the Filtering Capacitor (C)

Using the ripple factor equation:

$\gamma = 1/4 \times \sqrt{3} \times f \times C \times RL$

Rearranging for C :

$C = 1/4 \times \sqrt{3} \times f \times RL \times \gamma$

Substituting the known values:

$C = 1/4 \times \sqrt{3} \times 50 \times 200 \times 0.015$

$C = 1/4 \times 1.732 \times 50 \times 200 \times 0.015$

$C = 1/1039.2 \approx 962\ \mu F$

Selected Filtering Capacitor

A standard capacitor value close to the calculated value is **1000 µF**, which is commonly available and suitable for this application.

Summary of Filter Design Parameters

Ripple Factor (γ):1.5%

Load Resistance (RL):200 Ω

Filtering Capacitor (C): 1000 µF

Frequency (f):50 Hz

DC Output Voltage (Vdc): 5V

Load Current (Iload): 25 mA

Key Notes

The 1000 µF capacitor effectively reduces the ripple voltage to ensure a stable 5V DC output.

The ripple factor of 1.5% meets the design requirement of being less than 2%.

The filter design ensures that the power supply unit provides a clean and stable DC voltage for the connected electronic circuits.

3.2.4. REGULATOR SELECTION

A voltage regulator is designed to produce a fixed output voltage that remains stable regardless of variations in the input voltage or changes in load conditions. In this circuit, the LM7805 voltage regulator is used, which operates under the following conditions:

Input Voltage (Source Voltage): +12V

Output Voltage: +5V

Voltage Drop Across LM7805: 12V - 5V = 7V

The LM7805 ensures a consistent 5V output, making it ideal for powering low-voltage electronic circuits that require a stable power supply.

3.3. THE VISUAL DISPLAY UNIT

The display unit consists of a seven-segment displays and BJT transistors functioning as current switches. This design uses two types of BJT transistors and two types of seven-segment screens. The unit incorporates four seven-segment displays, which are multiplexed together. All seven segments are connected to Port B of the microcontroller (pins RB0, RB1, etc.).

The LEDs in the seven-segment displays are configured such that they glow when the microcontroller pin is driven LOW and turn OFF when the pin is set HIGH. This configuration is chosen because microcontrollers typically have a current-sinking capability in the range of tens of milliamperes, while their current-sourcing capabilities limited to microamperes. To protect the LEDs and ensure proper operation, current-limiting resistors are used to control the current flowing through each LED.

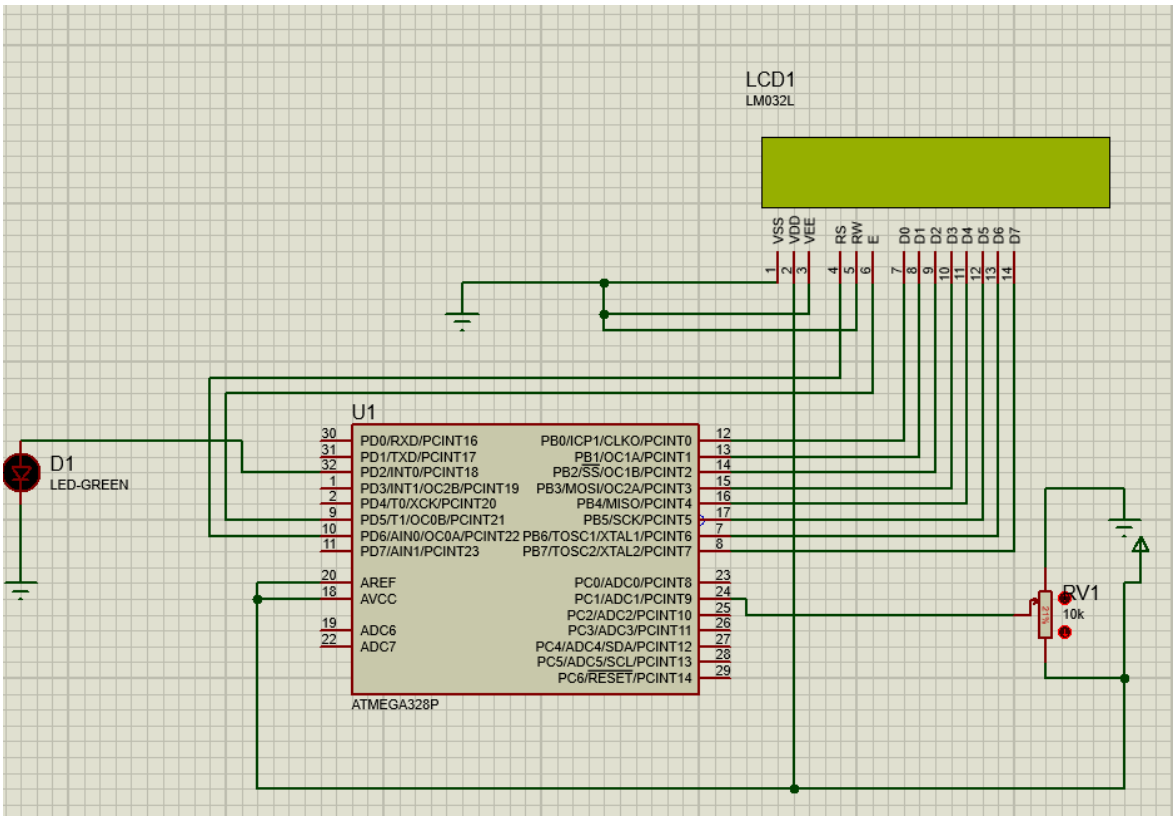


Figure 3.2 Visual display section with LED

3.3.1 COMPONENTS USED

Components used include:

- ATmega 328P microcontroller.
- LCD.
- NPN transistors.
- 15A Fuse.
- 5V LM7805 Regulator.
- 220V/12V Transformer.

- ON/OFF Switch.
- 1N4001 Diodes.
- SPDT Relay.
- 1/4W Resistors.
- Ceramic Capacitors.
- Push-buttons.
- Light Emitting Diodes.
- 13-amp socket.
- 5V battery.

3.3.2. DESIGN CALCULATION FOR LEDS

Transistor Base-Emitter Voltage (VBE)

For an NPN transistor, the minimum proper base-emitter voltage VBE is as follows :

For a silicon (Si) transistor, (VBEQ= 0.7V).

For a germanium (Ge) transistor, (VBEQ= 0.3V).

In this design, a silicon transistor is used, so (VBE = 0.7V). Additionally, the voltage on any pin of the ATmega382P microcontroller (except for (VDD), (MCLR), and (RA4)has a maximum value of (Vbb = 0.3V). Since the supply voltage = 5V , the voltage supplied to the base of the NPN transistor from the ATmega382P is:

VBB = 5V - 0.3V = 4.7V

The maximum output current sourced by any input/output pin of the microcontroller is 25mA.

LED Voltage and Current Parameters;

LED Forward Voltage VLED: 2V (from the datasheet).

Collector-Emitter Voltage VCE:0.2V.

Output Voltage VOUT :4.7V.

The voltage across the collector resistor VRC is calculated as:

VRC= VOUT - VLED - VCE .....Equation 3.4

Substituting the values:

VRC = 4.7V - 2V - 0.2V = 2.5V

The current through the collector resistor IRC is equal to the LED current ILED, which is set to 20mA. The value of the collector resistor RC is calculated as:

RC = VRC/ILED.....Equation 3.15

Substituting the values:

RC = 2.5V/20 x10^-3A= 125 Ω

While (RC = 125) is the minimum value required to provide the LED with a forward voltage of 2V, this value would cause the LEDs to burn out quickly. Therefore, a higher value of (RC) is chosen to ensure safe operation. A value of **330 Ω** is typically preferred for better reliability.

Transistor Current Calculations

The total collector current (IC) is the sum of the currents through all seven segments of the display:

IC = ILED x 7 = 20mA x 7 = 140mA

The base current IB is calculated using the transistor's current gain hfe:

IB = IC/hfe.....Equation 3.5

For the **S9014 transistor**, the current gain hfe = 33 . Substituting the values:

IB = 140mA ≈ 4.24mA

This base current of **4.24mA** is well below the **maximum output current** of 25mA that the microcontroller can source, ensuring that the microcontroller is not overloaded. The NPN transistor acts as a **buffer**, providing both current and power gain for the microcontroller.

Base Resistor RB Calculation

The voltage across the base resistor VRB is calculated as:

VRB = VBB - VBE.....Equation 3.6

Substituting the values:

VRB= 4.7V - 0.7V = 4V

The base resistor R_B is then calculated as:

RB = VRB/IB = 4V/4.24mA ≈ 942.86 Ω

A **standard resistor value** of **1000 Ω (1kΩ)** is chosen for RB to ensure reliable operation.

Summary of Design Parameters

i. Collector Resistor (RC): 330 Ω (chosen for safety).

ii. Base Resistor (RB): 1000 Ω (1kΩ).

iii. LED Forward Voltage (VLED):2V.

iv. Collector-Emitter Voltage (VCE): 0.2V.

v. Base-Emitter Voltage (VBE): 0.7V.

vi. Total Collector Current (IC): 140mA.

vii. Base Current (IB): 4.24mA.

This design ensures that the LEDs operate safely and efficiently, with the transistor acting as a buffer to protect the microcontroller from excessive current draw.

3.3.3 CONTROL UNIT

The control unit is primarily the microcontroller itself. it is the central processing unit that controls the entire system's operation. The microcontroller's primary function is to manage the timing process : It receives input from the switches (SET TIME, START, RESET, and STOP), and it then counts down the preset time, and then sends turn off signal to the relay when the time reaches zero.

3.4 SOFTWARE DESIGN

The Software design is the process of planning, developing, and implementing the software that operates the microcontroller. The microcontroller was Programmed using MicroC to implement logic for timing and control operations.

3.4.1 PSEUDOCODE OF THE PROGRAM

- Turn on the switch.
- Turn off the output relay.
- Input the set-time as time1.
- If reset button is hit Clear set-time.

5. If the start button is pressed, it turns the relay on.

   Begin the countdown with active_time = time1.

   While active_time > 0:

   Decrement active_time by 1.

   If active_time = 0:

   Shut the relay off, Turn on the red LED, Activate the buzzer for one minute

   If active_time > 0 and the relay is shut off:

   Turn on the blue LED and buzzer, indicating a power failure.

   The backup battery saves the system state.

6. At power restoration:

   If the start button is pressed, resume the operation.

- If the reset button is pressed, go to step 4.
- If the stop button is pressed, halt the entire operation.
- If the switch is turned off, shut the relay off, and ends the operation.

The pseudocode above outlines the logical flow of the program, ensuring proper control of the relay, LEDs, and buzzer based on user inputs and system conditions.

3.4.2 PROGRAM FLOWCHART

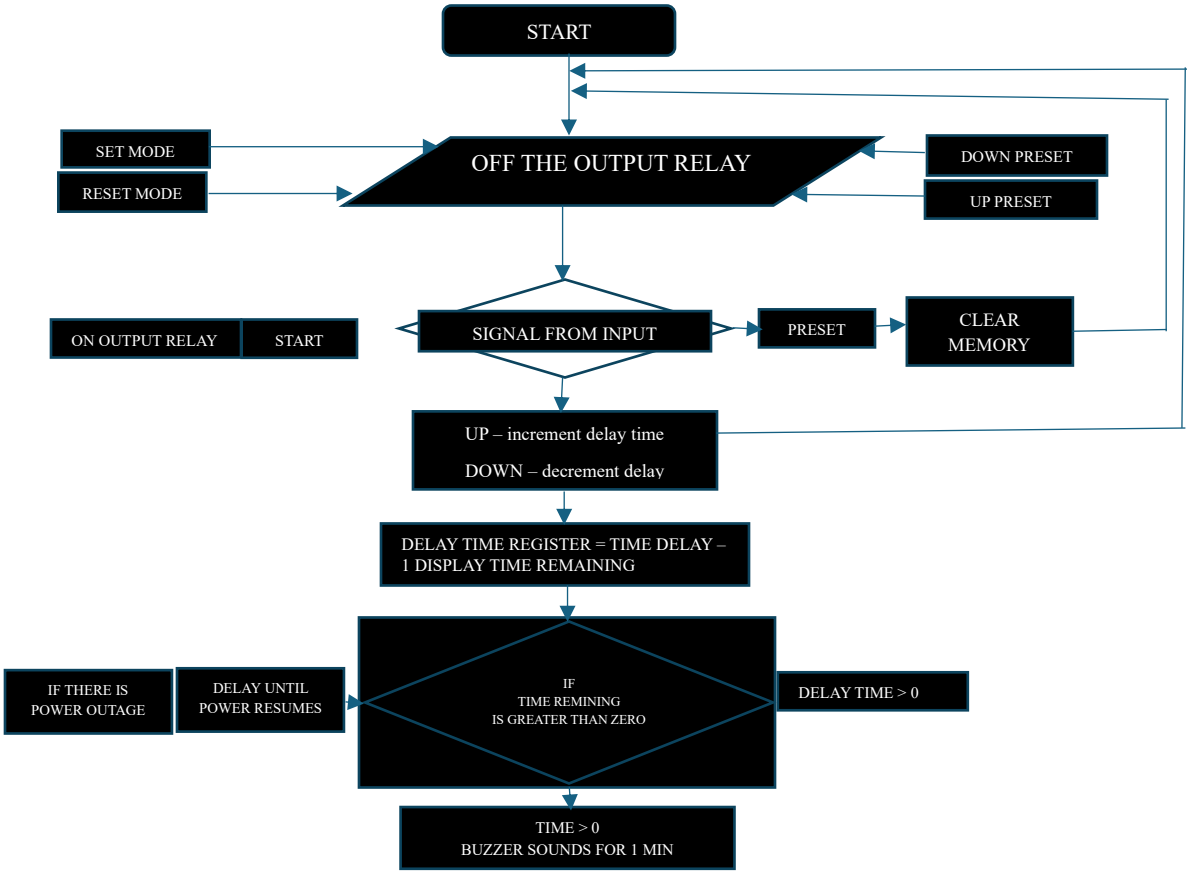


Figure 3.3 Flowchart of the program.

3.5. CONNECTION/OPERATION UNIT

The Connection/Operation Unit is a critical part of the Microcontroller Timer Socket Outlet, as it is where the user connects their electrical appliances to be controlled by the timer. This unit ensures that the appliances receive power for the specified duration and are disconnected from the mains supply once the timer counts down to zero, thereby conserving energy and reducing potential hazards.

Key Components and Functionality:

13-Amp Socket Outlet:

Function: The 13-amp socket outlet is a standard electrical receptacle designed to accommodate appliances that have a fuse rating of 13A or lower. This includes a wide range of common household devices such as phone chargers and televisions.

Role: It provides a convenient and safe connection point for the user's appliances, allowing them to be powered and controlled by the timer.

Relay:

Function: The relay is an electromechanical switch that is controlled by the microcontroller. It operates as a link between the microcontroller and the power supply to the socket outlet.

Role: When the timer is active and counting down, the microcontroller keeps the relay in a closed position, allowing current to flow through to the socket outlet and power the connected appliance. Once the timer reaches zero, the microcontroller sends a signal to the relay to open the circuit, disconnecting the appliance from the power supply.

3.6 OPERATION OF THE CONNECTION UNIT (MICROCONTROLLER TIMER SOCKET OUTLET)

Connection: The user plugs their appliance into the 13-amp socket outlet.

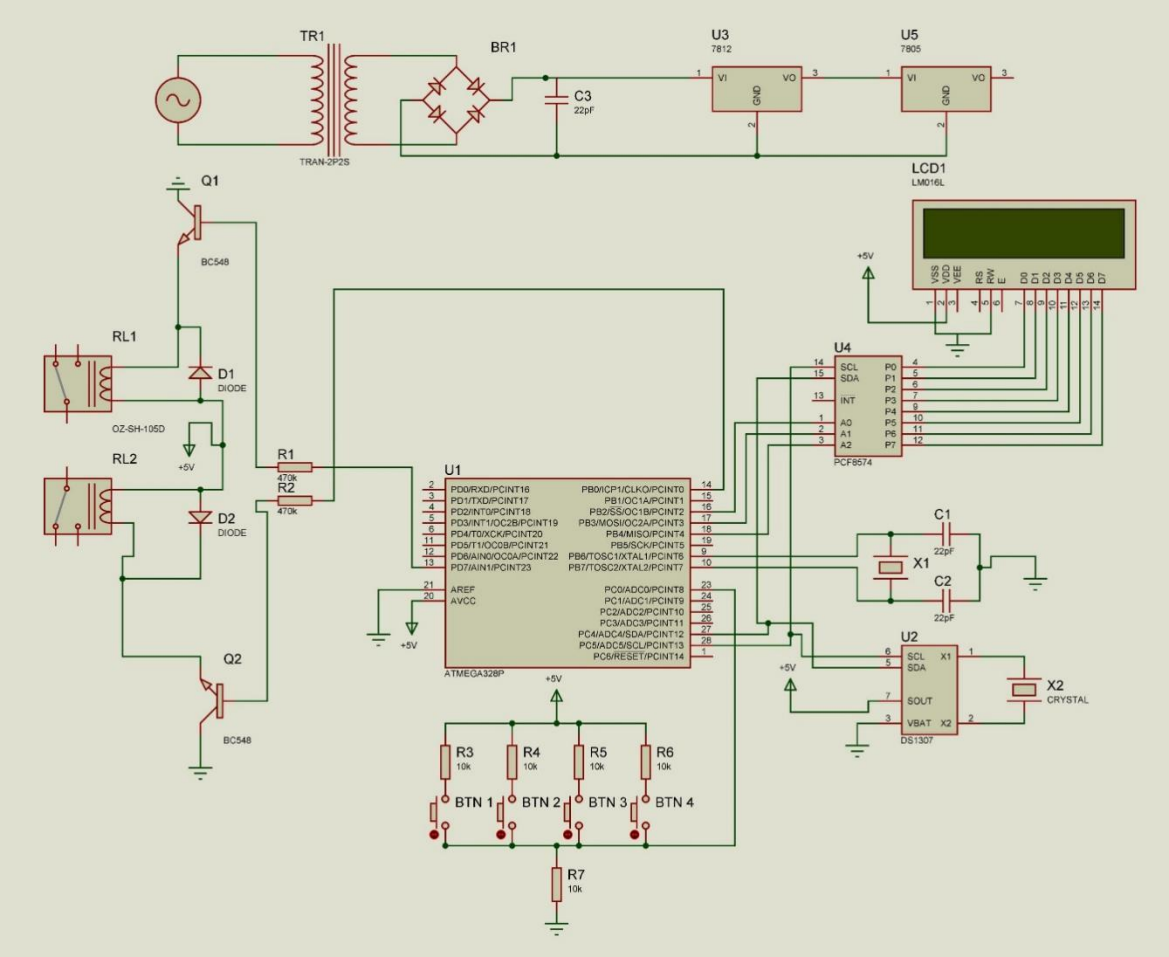
Timing Initiation: The user sets the desired time using the SET TIME button, presses the START button.

Power Supply: The microcontroller activates the relay, allowing power to flow to the socket outlet and energize the appliance.

Countdown: The microcontroller counts down the preset time, displaying the remaining time on the LCD.

Disconnection: Once the timer reaches zero, the microcontroller deactivates the relay, cutting off the power supply to the socket outlet and the connected appliance.

Completion Indication: The buzzer and LED serve as audio and visual indicators to signal the completion of the timing process, confirming that the appliance has been disconnected from the power supply.



CIRCUIT DIAGRAM

CHAPTER FOUR

CONSTRUCTION AND TESTING

4.1 INTRODUCTION

This chapter provides an overview of the testing procedures for various components, the standard and calculated values of the components used, and details on the construction, system operation, and casing design. Additionally, it highlights the software program development process, including program debugging, simulation, and loading.

4.2 COMPONENT TESTING

Each stage of the design was systematically tested, beginning with the power supply unit and proceeding to the connection/operation unit. The transformer was tested to ensure a 12V output on the secondary side, while the voltage regulator was verified to provide a stable 5V output for powering the circuit.

The pulse generated by the timer circuit was observed to align with the design specifications. The counting sequence of each decade counter was monitored using a seven-segment display and confirmed to be accurate.

The following instruments/devices were used for component testing:

i. Ammeter

ii. Voltmeter

iii. Ohmmeter

The following tests were conducted on each component:

Continuity and Short Circuit Tests: Performed on the transformer bridge, rectifier, transistor, and other components.

Mechanical Failure Tests: Conducted to identify broken or damaged pins on components.

Value Comparison: The measured values of the various components were compared with the values stated in the schematic diagram.

Initial confirmatory tests were carried out on a breadboard before finalizing the circuit on a vero-board. All connections on the vero-board were thoroughly inspected before soldering to ensure accuracy and functionality.

Table 4.1 Components used

	COMPONENTS	QUANTITY	LABEL	CALCULATED VALUE	STANDARD VALUE
	Microcontroller	1	ATmega 328P	-	ATmega328P, 28-pins, 8MHz
	LCD	1	DS1,DS4,DS5,D	-	Common cathode
	Display	1	S6	-	-
	NPN Transistor	5	Q3,Q4,Q5,Q6,Q7	-	-
	Fuse	1	F1	-	1A
	LM7805	1	RG1	-	5V
	Regulator	1			-
	Step-down	1	T1	-	220V/12V, 1A 50Hz
	Switch	1	on/off	-	-

	IN4001 Diodes	4	BR1	-	-
	SPDT Relay	1	RL1	-	-
	Resistor	1	R3	-	10kΩ
	Resistor	1	R12	-	4.7kΩ
	Resistor	9	R1,R2,R3,R4,R5, R6,R7,R8,R9	942.56	1kΩ
	Resistor	7	RP1	125Ω	330Ω
	Capacitor	1	C5	-	22pF
	Capacitor	1		962µF	1000µF
	Push buttons	5	reset, start, up	-	-
			preset, down	-	-
			preset, stop	-	-
	LED	4	D1, D2, D3, D4	-	-
	13-amp socket	1	-	-	-
	Battery	1	B1	-	6V

4.3 THE CONSTRUCTION

After completing the design and calculations for each stage, the components were acquired and assembled on a breadboard. The connections were tested to ensure proper functionality before transferring the circuit to a vero-board for soldering. The construction process was carried out step by step, with each stage being tested upon completion. The construction began with the power supply unit and progressed through to the visual display unit, followed by the casing.

To facilitate smooth soldering, the surface of the vero-board was scraped to expose the copper tracks, ensuring the molten solder could flow easily. During soldering, precautions were taken to avoid dry joints by ensuring the soldering iron was sufficiently hot and the components were thoroughly cleaned beforehand.

The materials used for construction included:

i. Soldering Iron
ii. Soldering Lead
iii. Vero-Board
iv. Cutting Pliers
v. Connecting Wires

The Integrated Circuit (IC) sockets were first mounted and soldered onto the vero-board. Other components were then connected and soldered in their respective positions as per the circuit diagram . The use of IC sockets allowed for easy access and replacement of the ICs when necessary.

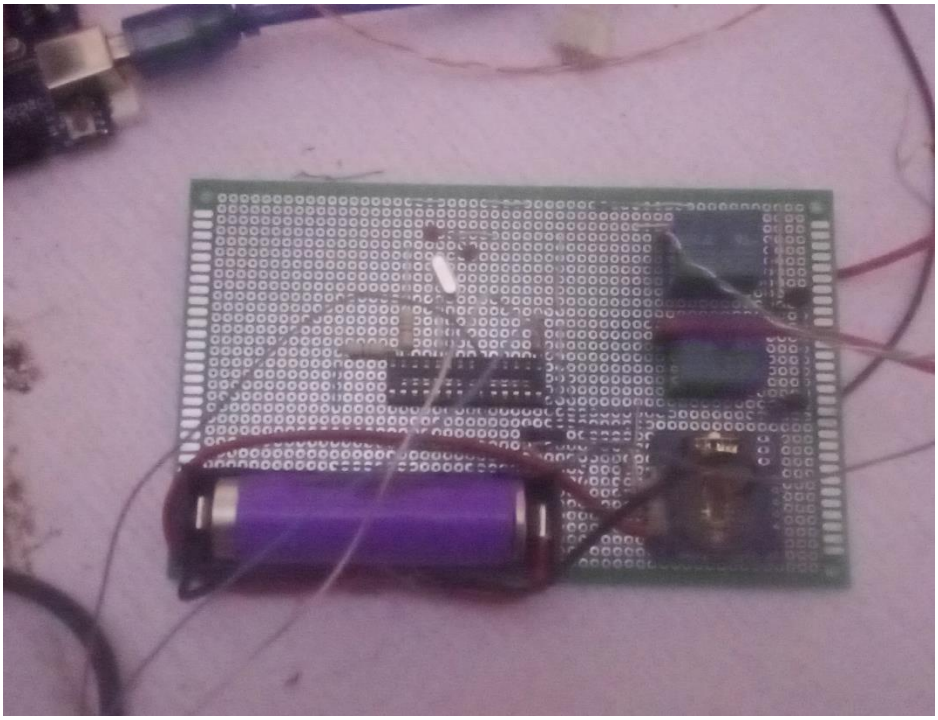


Figure 4.1a Circuit under construction

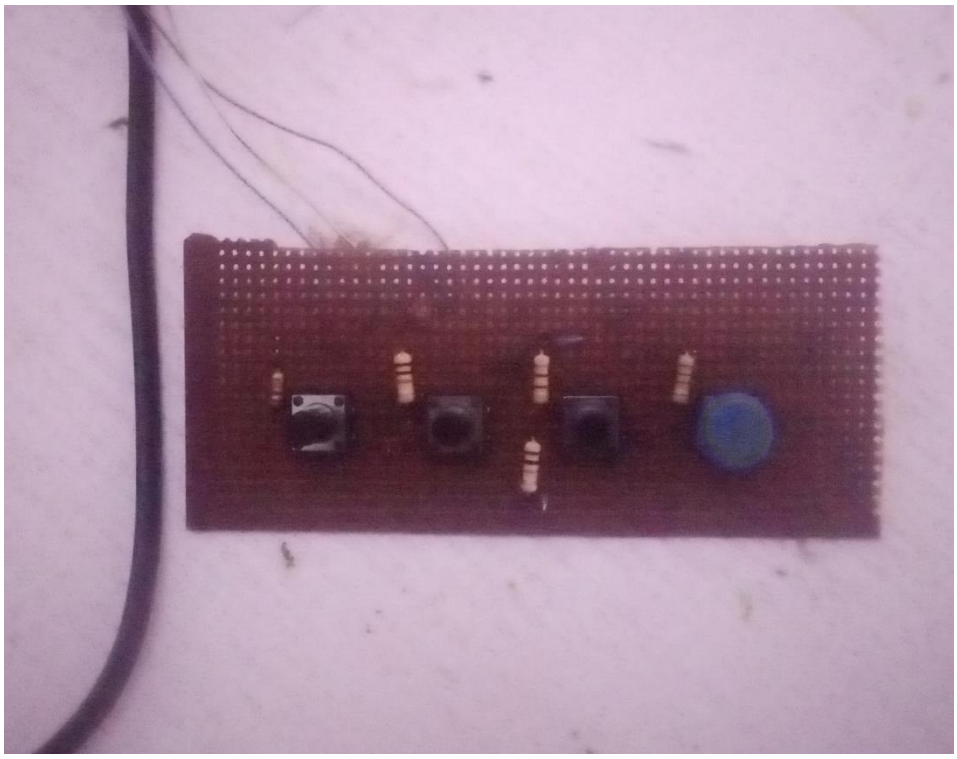


Figure 4.1b Circuit under construction(buttons)

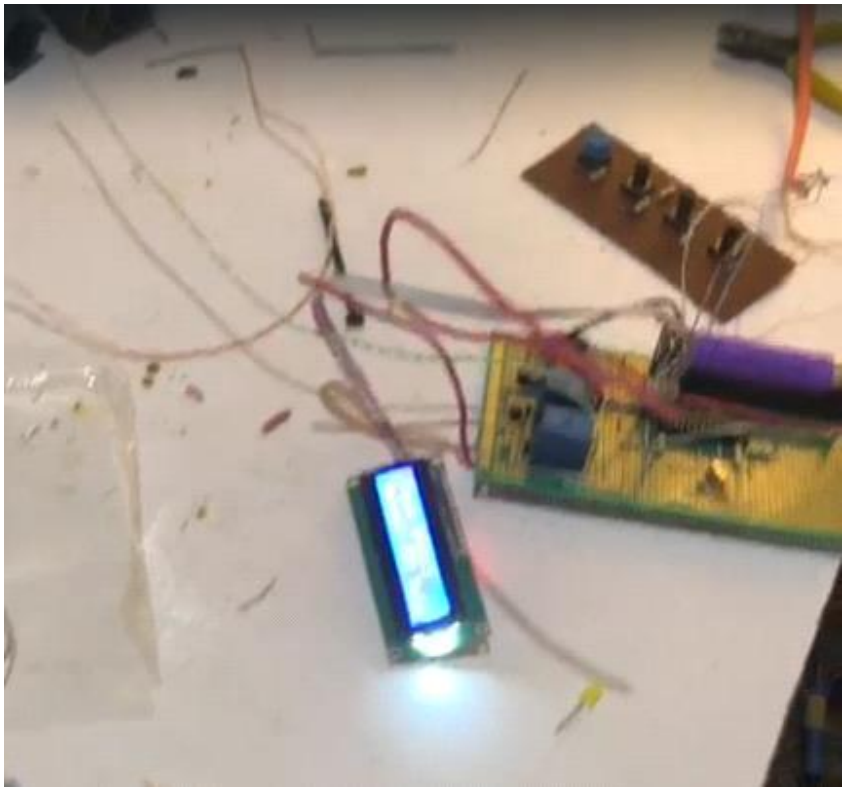


Figure 4.1c Circuit under construction(LCD, relay, buttons etc.)

4.3.1 SOLDERING PROCEDURES AND PRECAUTIONS

The electronic components outlined in the circuit diagram were soldered onto the vero-board to create a continuous path for signal flow. The point-to-point soldering method was carefully employed to prevent dry joints between the component leads and the copper tracks. A soldering lead with a low melting point was used to minimize the risk of damaging the components. Excessive heat or prolonged application of the soldering iron was avoided to protect the components from damage.

The following precautions were observed during soldering:

i. A 60-W soldering iron was used.

ii. The surfaces of the components and the soldering iron were kept clean to ensure proper soldering.

iii. Components were not overheated to prevent damage.

4.4 OVERALL SYSTEM TESTING

After all components were soldered onto the vero-board, the circuit was tested again to verify the correctness of the soldering and overall functionality. During testing, it was discovered that the push buttons were not functioning correctly. Upon inspection, it was found that the pins of the stop button were improperly soldered. The issue was resolved by re-soldering the pins, after which the buttons worked perfectly. Once this correction was made, the circuit operated as intended and was prepared for casing.

4.5 THE SYSTEM OPERATION

This project is designed to control power usage and time the operation of connected appliances for a specified duration. The maximum operating time for the socket is set using the timer's up or down buttons. Once the timer is set, the circuit begins operation, and the display continuously updates the remaining time. The reset button resets the timer and time memory to zero. At the end of the timed operation, a buzzer sounds, and the display shows an "end of operation" message.

4.6 CASING CONSTRUCTION

Several factors were considered when designing the casing for this project, including:

i. Strength of the material

ii. Cost of the materials

iii. Durability

iv. Size of the circuitry

Figure 4.2 Casting outlook of the project.

4.7 PROGRAM DEBUGGING

Debugging is the process of identifying and fixing errors or bugs that affect the functionality of a program. During debugging, all relevant elements are examined, including input data, output data, the source code, library routines, and any other factors that could contribute to the issue. An open-minded approach is essential, meaning that if a suspected section of the code appears error-free, other areas are investigated instead.

During the debugging of the software program, the following aspects were carefully reviewed:

1. **Initialization of Variable**s: All variables used in the program were properly initialized.

orrect Use of Datatypes: System-defined datatypes were implemented accurately.

1. **Proper Implementation of Functions**: System-defined functions were used correctly.

Case Sensitivity: Attention was paid to case sensitivity in the code.

1. **Understanding Microcontroller Ports**: Adequate study and comprehension of the microcontroller ports were ensured.

4.8 PROGRAM SIMULATION

The program was simulated using Proteus. The simulation process involved the following steps:

1. **Set Debugger Mode**: The debugger was configured to "Software Simulator."
2. **Start Debugging**: The debugger was initiated by pressing F9, which opened a window displaying a list of variables.
3. **Add Variables**: Variables were added to the list by clicking "Add."
4. **Select Functions**: An icon appeared next to the variables, allowing the selection of the intended function.
5. **Simulate Program**: The program was simulated to model its actual performance.

4.9 PROGRAM LOADING

Loading the program into the microcontroller involved several steps. A USBTinyISP programmer was used to transfer the program into the microcontroller's memory. Additionally, the WinAVR programming environment was installed on the computer to write and compile the program before transferring it to the microcontroller.

To ensure proper functionality, AVRDUDE was used to test the programmer while it was connected to both the computer and the microcontroller. The program was assigned to specific memory locations on the microcontroller, with most of the code directed to PORTB

For instance, the code snippet that follows was used:

DDRB = 0b00000001; • DDRB: Indicates the PORTB Data Direction Register.

- DDRB: Refers to the Data Direction Register for PORTB.
- The following number is in binary format, as indicated by the symbol 0b.
- 1: Specifies the pin position.

Once the program was finalized, it was loaded into the microcontroller and soldered onto the vero-board. The complete program code can be found in Appendix II.

4.10 RESULT

The construction of the microcontroller-based timer socket outlet was a time-intensive process due to its complexity and the high level of precision required. Despite challenges such as manufacturer defects and unreliable components, the project was successfully completed.

The microcontroller was configured to communicate with other hardware components, and the control circuit was implemented to drive the display unit. When an appliance is connected to the timer socket and a time input is set, pressing the start button activates the socket. A green LED indicates that the system is operational, and the LCD screen shows the elapsed or remaining time.

Upon completion of the countdown, a red LED and a buzzer provide visual and audio signals, respectively, to indicate the end of the operation.

CHAPTER FIVE

CONCLUSION, LIMITATIONS AND RECOMMEDATION

5.1 CONCLUSION

This particular project has taken one through some of the acts of engineering practice which involve; design, construction and testing. The project demonstrates that through the use of devices such as ATmega 328P, 7805 voltage regulator and other components used, electrical power usage can be monitored and controlled.

5.2 LIMITATION

This project has the following limitations:

i. It only supports 13-amp socket.

ii. The buzzer continues to buzz until the system is attended to. This can weaken the backup battery if allowed to buzz for a long period of time.

5.3 RECOMMENDATION

The following are recommended for future improvement:

i. The buzzer should be programmed to buzz for maximum of ten (10) seconds.

REFERENCES

1. Ahmed M. S., Mohammed A. S., Onimole T. G., and Attah P. O. (2006)

Design and construction of a remote Controlled fan regulator, Leonardo Electronic Journal of Practices and Technologies, Vol. 9, pp. 55-62* [online]. Available at: http://www.lejpt.academicdirect.org/A09/055_062.pdf [2nd January, 2015].

2. Datasheet Search System [online]; ©2000-2006 Datasheet4u.com

Available at: http://www.datasheet4u.com/ [2nd January, 2015].

3. Howstuffworks [online]; ©1998-2006 HowStuffWorks

Available at: http://www.howstuffworks.com/ [2nd January, 2015].

4. IEE Wiring Matters | Spring 2006 | [online]

Available at: http://www.iee.org/ [2nd January, 2015].

5. Inderpreet, Kaur (2010)

Microcontroller Based Home Automation System With Security, International Journal of Advanced Computer Science and Applications, Vol. 1, No. 6, India.

6. Integrated Electronics, M. C. Grewhill book by Robert B. S. Company (2008)

7. Kolo J. G. and Dauda U. S. (2008)

Development of a Simple Programmable Control Timer, Leonardo Journal of Sciences, ISSN 1583-0233, pp. 175-186.

8. Microcontrollers – A Brief History of Microprocessors – PIC Vietnam [online] Available at: http://www.picvietnam.com/ [2nd January, 2015].

9. Morgan J. (1999)
Time Management, Mood House Press, California.

10. Paul H. & Winfield H (2000)
The Art of Electronics, 2nd edition. Cambridge University Press, UK.

11. Raphael, Jr. (2008)
"Unplug for Dollars: Stop 'Vampire Power' Waste | PCWorld", PCWorld - Reviews and News on Tech Products, Software and Downloads. Available at: http://www.pcworld.com/article/153245/unplug_for_dollars_stop_vampire_power_waste.html [5th December, 2014].

12. Robert Boylestad and Louise Nashelsky (2004)
Electrical Devices and Circuit Theory, Prentice Hall Inc. Englewood Cliff, New Jersey.

13. Theraja B. L. and Theraja A. K. (2002)

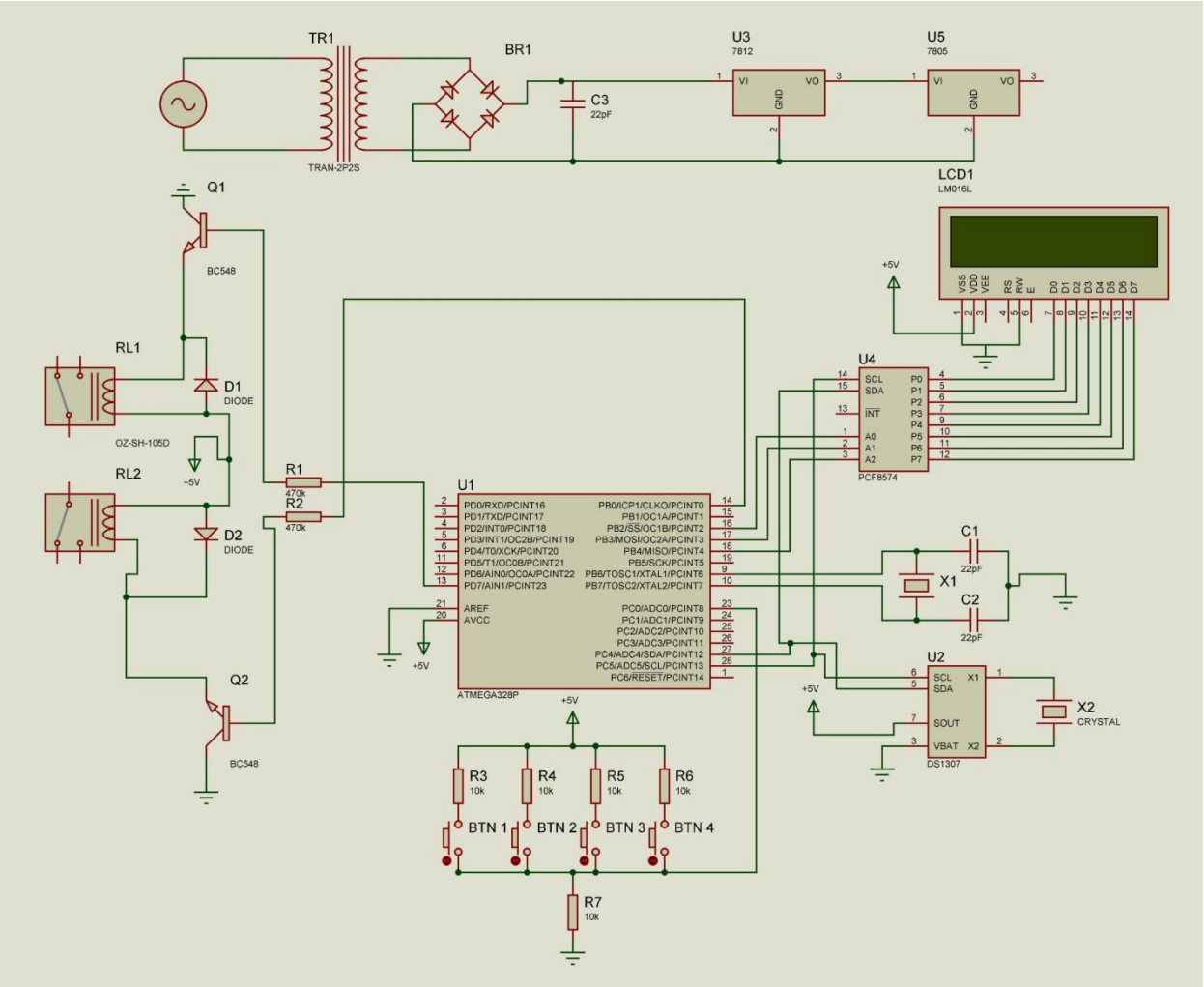
A Textbook of Electrical Technology, Revised 23rd Ed, Chand Company Ltd, New Delhi.

14. Theraja B. L. and Theraja A. K. (2005).

This alphabetical arrangement ensures proper organization and easier referencing.

APPENDIX I

CIRCUIT DIAGRAM



APPENDIX II

PROGRAM CODE

```
#include <Wire.h>              // Library for I2C communication
#include <LiquidCrystal_I2C.h>   // Library for I2C LCD
#include <RTClib.h>            // Library for RTC (Real-Time Clock)


// Initialize LCD and RTC
LiquidCrystal_I2C lcd(0x27, 16, 2); // Create LCD object with I2C address 0x27, 16 columns,
2 rows
RTC_DS1307 rtc;                // Create RTC object


// Relay Pins
#define RELAY1 6  // Define pin for Relay 1
#define RELAY2 7  // Define pin for Relay 2


// Button input pin
#define BUTTON_PIN A0  // Define analog pin for button input


// Button voltage ranges (ADC values)
#define BTN1_MIN 200  // Voltage range for Button 1 (~1-1.5V)
#define BTN1_MAX 350
```

```
#define BTN2_MIN 400  // Voltage range for Button 2 (~2-2.5V)
#define BTN2_MAX 550
#define BTN3_MIN 600  // Voltage range for Button 3 (~3-3.5V)
#define BTN3_MAX 750
#define BTN4_MIN 800  // Voltage range for Button 4 (~4-4.5V)
#define BTN4_MAX 900

// Variables to store relay on/off times
int relay1OnHour = 0, relay1OnMin = 0, relay1OffHour = 0, relay1OffMin = 0;
int relay2OnHour = 0, relay2OnMin = 0, relay2OffHour = 0, relay2OffMin = 0;

// Variables for menu navigation and settings
int menuState = 0, selectedRelay = 1, settingIndex = 0;
bool settingMode = false;  // Flag to indicate if in setting mode

void setup() {
  Serial.begin(9600);  // Initialize serial communication at 9600 baud rate
  pinMode(RELAY1, OUTPUT);  // Set Relay 1 pin as output
  pinMode(RELAY2, OUTPUT);  // Set Relay 2 pin as output
  lcd.init();              // Initialize the LCD
  lcd.backlight();         // Turn on the LCD backlight
  rtc.begin();             // Initialize the RTC
  Serial.println("Initializing RTC...");
```

```
  if (!rtc.isrunning()) {   // Check if RTC is running
    Serial.println("RTC not running, setting time...");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));  // Set RTC to compile time if not running
  }
}
void loop() {
  DateTime now = rtc.now();  // Get current time from RTC
  Serial.print("Current Time: ");
  Serial.print(now.hour()); Serial.print(":");
  Serial.print(now.minute()); Serial.print(":");
  Serial.println(now.second());  // Print current time to serial monitor
  handleButtons();              // Handle button inputs
  if (!settingMode) {           // If not in setting mode
    displayIdleScreen(now);    // Display idle screen with current time and relay status
    controlRelays(now);        // Control relays based on/off times
  }
}
void handleButtons() {
  int buttonValue = analogRead(BUTTON_PIN);  // Read analog value from button pin
  if (buttonValue >= BTN1_MIN && buttonValue <= BTN1_MAX) {  // Button 1 pressed
    if (!settingMode) settingMode = true;  // Enter setting mode if not already
    else validateSetting();              // Validate setting if in setting mode
```

```
  } else if (buttonValue >= BTN2_MIN && buttonValue <= BTN2_MAX) {  // Button 2
pressed
      adjustValue(1);  // Increase value
  } else if (buttonValue >= BTN3_MIN && buttonValue <= BTN3_MAX) {  // Button 3
pressed
      adjustValue(-1); // Decrease value
  } else if (buttonValue >= BTN4_MIN) {  // Button 4 pressed
      goBack();  // Go back in menu or exit setting mode
  }
  delay(200);  // Debounce delay
}
void displayIdleScreen(DateTime now) {
  lcd.clear();  // Clear the LCD screen
  lcd.setCursor(0, 0);  // Set cursor to first row
  lcd.print("Time: ");  // Display current time
  lcd.print(now.hour(), DEC);
  lcd.print(":");
  lcd.print(now.minute(), DEC);
  lcd.print(":");
  lcd.print(now.second(), DEC);
  lcd.setCursor(0, 1);  // Set cursor to second row
  lcd.print("Relay1: ");  // Display Relay 1 status
  digitalRead(RELAY1) ? lcd.print("ON  ") : lcd.print("OFF ");
```

```
    lcd.print("Relay2: ");  // Display Relay 2 status
    digitalRead(RELAY2) ? lcd.print("ON  ") : lcd.print("OFF ");
    delay(1000);  // Refresh delay
}
void controlRelays(DateTime now) {
    // Turn Relay 1 on/off based on set times
    if (now.hour() == relay1OnHour && now.minute() == relay1OnMin) digitalWrite(RELAY1,
HIGH);
    if (now.hour() == relay1OffHour && now.minute() == relay1OffMin) digitalWrite(RELAY1,
LOW);
    // Turn Relay 2 on/off based on set times
    if (now.hour() == relay2OnHour && now.minute() == relay2OnMin) digitalWrite(RELAY2,
HIGH);
    if (now.hour() == relay2OffHour && now.minute() == relay2OffMin) digitalWrite(RELAY2,
LOW);
}
void validateSetting() {
    settingIndex++;  // Move to next setting index
    if (settingIndex > 3) {  // If all settings are done
        settingIndex = 0;    // Reset index
        settingMode = false; // Exit setting mode
    }
}
```

```
void adjustValue(int change) {
  // Adjust relay 1 on/off times based on setting index
  if (settingIndex == 0) relay1OnHour = constrain(relay1OnHour + change, 0, 23);
  else if (settingIndex == 1) relay1OnMin = constrain(relay1OnMin + change, 0, 59);
  else if (settingIndex == 2) relay1OffHour = constrain(relay1OffHour + change, 0, 23);
  else if (settingIndex == 3) relay1OffMin = constrain(relay1OffMin + change, 0, 59);
}
void goBack() {
  if (settingMode && settingIndex > 0) settingIndex--;  // Go back to previous setting
  else settingMode = false;  // Exit setting mode
}
```