<u>Group Number: D45</u>

A

MINI PROJECT REPORT ON

PWM BASED LIGHT INTENSITY CONTROLLER USING PIC

SUBMITTED TO SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE FULFILLMENT OF THE REQUIREMENTS
FOR THE COMPLETION OF MINI PROJECT

OF

THIRD YEAR ENGINEERING

IN

ELECTRONICS & TELECOMMUNICATION

BY

Vyankatesh Nilkanth Vibhute	**Exam No. T190233380**
Rushikesh Balasaheb Wakde	**Exam No. T190233386**
Ishan Rajendra Burchunde	**Exam No. T190233060**

UNDER THE GUIDANCE OF

Prof .T.V. Kafare

Sinhgad Institutes

DEPARTMENT OF ELECTRONICS & TELECOMMUNICATION ENGINEERING

SINHGAD COLLEGE OF ENGINEERING

S. No. 44/1, OFF SINHGAD ROAD, VADGAON BK, PUNE – 411041

APRIL 2024

(2023-24)

Sinhgad Institutes

CERTIFICATE

This is to certify that the Mini Project entitled

"PWM BASED LIGHT INTENSITY CONTROLLER USING PIC"

Submitted By

Vyankatesh Nilkanth Vibhute
Rushikesh Balasaheb Wakde
Ishan Rajendra Burchunde

is a bonafide work carried out by them under the supervision of **Prof .T.V. Kafare** and it is approved for the partial fulfillment of the requirements of T.E. E&TC Engineering submitted to Savitribai Phule Pune University, Pune.

The Mini Project work has not been earlier submitted to any other institute or university for the award of degree or diploma.

Prof .T.V. Kafare	**Dr. D. G. Ganage**	**Dr. M. B. Mali**	**Dr. S. D. Lokhande**
Guide	*Mini Project Coordinator*	*Head*	*Principal*
Department of E&TC	*Department of E&TC*	*Department of E&TC*	*SCOE, Pune*

Place: Pune

Date:

ACKNOWLEDGEMENT

We are feeling very humble in expressing my gratitude. It will be unfair to bind the precious help and support which we got from many people in few words. But words are the only media of expressing one's feelings and my feeling of gratitude is absolutely beyond these words. It would be my pride to take this opportunity to say the thanks.

Firstly, we would thank our beloved guide **Prof .T.V. Kafare** for his valuable guidance, patience and support; He was always there to force us a bit forward to get the work done properly and on time. He has always given us freedom to do mini project work and the chance to work under his supervision.

We would like to express our sincere thanks to **Dr. D. G. Ganage**, mini project coordinator, Department of E&TC, for his constant encouragement in the fulfillment of the mini project work. We would also like to express our sincere thanks to **Dr. M. B. Mali, Head,** Department of E&TC for his co-operation and useful suggestions. We would also like to thank **Dr. S. D. Lokhande, Principal, Sinhgad College of Engineering.** He always remains a source of inspiration for us to work hard and dedicatedly.

It is the love and blessings of our families and friends which drove us to complete this project work.

Thank you all!

Vyankatesh Nilkanth Vibhute
Rushikesh Balasaheb Wakde
Ishan Rajendra Burchunde

ABSTRACT

In this project, we present the design and implementation of a Pulse Width Modulation (PWM) based light intensity controller utilizing a PIC microcontroller. The primary aim of our project is to develop a versatile system capable of dynamically regulating the intensity of a light source by generating precise PWM signals through the PIC microcontroller. Our approach involves a comprehensive integration of hardware and software components, encompassing circuit design, microcontroller programming, and rigorous testing procedures.The project's foundation lies in a thorough exploration of PWM principles and the capabilities of PIC microcontrollers, underscoring their significance in modern electronics and automation. Through meticulous research and experimentation, we have devised a robust control scheme that effectively translates user input into corresponding adjustments in light intensity. Leveraging the flexibility of PWM, our system enables seamless transitions between different brightness levels, catering to diverse user preferences and environmental conditions.Furthermore, our project extends beyond mere technical implementation to encompass broader implications and potential applications. The ability to precisely modulate light intensity holds immense relevance across various domains, including residential, commercial, and industrial settings. Our controller's adaptability makes it suitable for diverse scenarios, ranging from mood lighting in homes to task lighting in workspaces, as well as specialized applications such as plant growth chambers and automotive lighting systems.By harnessing the synergies between PWM technology and PIC microcontrollers, our project seeks to contribute to the ongoing advancements in lighting control systems. We envision our work not only as a standalone endeavor but also as a stepping stone towards future innovations in smart lighting and energy-efficient solutions. Through collaborative efforts and interdisciplinary approaches, we strive to address contemporary challenges and drive progress in the field of electronics and automation.

CONTENTS

Chapter 1

Introduction

1.1 Introduction of the project

In an era where precision and efficiency are paramount in electronics and automation, the development of advanced control systems plays a pivotal role. Among these, the utilization of Pulse Width Modulation (PWM) techniques has emerged as a cornerstone for achieving dynamic and accurate control over various parameters. The ability to modulate the width of pulses within a digital signal offers unparalleled versatility, making PWM an indispensable tool in applications ranging from motor speed control to power regulation. In this context, our project focuses on harnessing the power of PWM for a specific application: light intensity control. Lighting plays a fundamental role in countless aspects of daily life, from creating ambiance in homes to facilitating productivity in workplaces. Traditionally, controlling light intensity has relied on analog methods, often lacking precision and flexibility. By contrast, PWM offers a digital approach that enables precise adjustment of light output with minimal energy wastage. The core objective of our project is to design and implement a PWM-based light intensity controller using a PIC microcontroller. This endeavor not only showcases the efficacy of PWM in achieving dynamic light control but also underscores the versatility of PIC microcontrollers in embedded systems. By interfacing hardware components with intelligent software algorithms, our controller aims to provide seamless and responsive adjustments to meet diverse lighting needs. Through meticulous research, experimentation, and collaboration, our project seeks to address several key challenges in light intensity control. These include achieving smooth transitions between different brightness levels, ensuring compatibility with various types of light sources, and optimizing energy efficiency. By overcoming these challenges, our controller promises to deliver an intuitive and efficient solution for applications requiring precise lighting control. Moreover, our project extends beyond technical implementation to explore broader implications and potential applications. From smart home automation to sustainable lighting solutions, the versatility of our PWM-based controller opens doors to innovative possibilities in diverse domains. By laying the groundwork for future advancements in lighting technology, we aim to make a meaningful contribution to the field of electronics and automation.

1.2 Aim and Objectives of the project

Aim: To design and develop an PWM (Pulse Width Modulation) Based Light Intensity Controller Using PIC Microcontroller.

Objectives:

1. To understand the principles of PWM modulation and its application in controlling light intensity.

2. To select and configure appropriate components, including the PIC microcontroller, push buttons, and LEDs.

3. To develop the necessary hardware circuitry to interface the PIC microcontroller with the push buttons and LEDs.

4. To program the PIC microcontroller to generate PWM signals and adjust the duty cycle based on user input from the push buttons.

5. To calibrate the system to achieve desired levels of light intensity control.

6. To test the functionality and performance of the PWM-based light intensity controller under various operating conditions.

7. To evaluate the energy efficiency and reliability of the controller in practical applications.

8. To document the design process, including circuit diagrams, code implementation, and testing procedures.

9. To analyze the results and identify areas for improvement or further development.

10. To demonstrate the project and its capabilities to stakeholders and potential users.

1.3 Hardware and Software Platform Used

Hardware Used:

1. PIC Microcontroller: Controls the PWM signals for light intensity adjustment and manages input and output interactions.

2. Input Devices: Utilizes push buttons for user input, facilitating adjustments to the light intensity controlled by the microcontroller.

3. Output Devices: Features an LED as the primary light source, with its brightness modulated by the microcontroller's PWM signals.

4. Power Supply Unit: Provides stable power to the system, ensuring consistent operation of the microcontroller, input devices, and LED.

5. Enclosure/Casing: Houses and protects the internal hardware components, maintaining a tidy appearance and user accessibility.

Software Platform Used:

1. Micro C Pro for PIC: Micro C Pro for PIC is a comprehensive IDE (Integrated Development Environment) designed specifically for programming PIC microcontrollers. It provides a user-friendly interface with features such as code editing, compiling, and debugging functionalities tailored for PIC-based firmware development. With its extensive library support and intuitive workflow, Micro C Pro for PIC simplifies the process of writing, testing, and optimizing code for PIC microcontrollers, making it an essential tool for embedded systems development.

2. Proteus 8 Professional: Enables simulation of electronic circuits and microcontroller systems, facilitating virtual testing and debugging before physical implementation.

3. PIC Kit 3: The PIC Kit 3 is a hardware programmer/debugger specifically designed for programming and debugging PIC microcontrollers. It connects to the target PIC microcontroller via In-Circuit Serial Programming (ICSP) and allows users to program, debug, and analyze code running on the microcontroller. It's commonly used in conjunction with MPLAB IDE, Microchip's integrated development environment, for PIC microcontroller development.

4. EasyEDA: EasyEDA is a cloud-based PCB design tool that simplifies the process of designing and prototyping electronic circuits. It offers a user-friendly interface with powerful schematic capture and PCB layout functionalities, allowing users to create professional-quality schematics and PCB designs effortlessly. With its extensive component library, real-time collaboration features, and seamless integration with manufacturing services, EasyEDA streamlines the PCB design workflow from concept to production, making it an ideal choice for engineers, students, and hobbyists alike.

1.4 Advantages and Applications

Advantages:

1. Reliability: The PWM-based light intensity controller offers reliable performance, ensuring consistent and accurate control over light output.

2. Low Power Consumption: By efficiently modulating the power delivered to the light source, the controller minimizes energy wastage, leading to lower power consumption and reduced operating costs.

3. Cost-Effectiveness: Implementing PWM control using a PIC microcontroller is a cost-effective solution, offering high performance at a relatively low cost compared to other control methods.

4. Versatility: The controller is versatile and can be adapted to various lighting applications, from residential and commercial lighting to automotive and industrial lighting, providing flexibility in usage.

5. Ease of Programming: Programming the PIC microcontroller to generate PWM signals and adjust the duty cycle is straightforward, thanks to the availability of user-friendly development tools and libraries.

6. Compact Size: The compact size of the controller makes it suitable for integration into space-constrained environments and devices, enabling efficient and space-saving designs.

Applications:

1. Residential Lighting: PWM control allows for adjustable lighting levels in homes, enhancing comfort and ambiance.

2. Commercial Lighting: PWM enables energy-efficient lighting solutions in offices, retail spaces, and commercial buildings.

3. Automotive Lighting: PWM is used for controlling interior and exterior lighting in vehicles, improving safety and aesthetics.

4. Horticultural Lighting: PWM-based systems facilitate precise control over light intensity for optimized plant growth in indoor farming and greenhouse applications.

5. Entertainment Lighting: PWM control is used in stage and studio lighting to create dynamic effects and mood lighting.

6. Medical Lighting: PWM-based systems provide adjustable illumination for medical examination lights and surgical lighting.

Chapter 2

Literature Review

2.1 Introduction (Recent trends of work / State of Art systems)

In recent trade work focused on lighting technology, there has been a noticeable surge in the adoption of Pulse Width Modulation (PWM) control systems. These systems enable precise adjustment of light intensity, catering to diverse applications ranging from residential and commercial lighting to automotive and horticultural sectors. Manufacturers are increasingly integrating PWM technology into LED lighting solutions to offer customizable brightness levels and dynamic lighting effects, enhancing user experience and energy efficiency. Furthermore, advancements in PWM control algorithms and microcontroller integration have streamlined the development of smart lighting systems, paving the way for IoT-enabled lighting solutions that can be remotely monitored and managed.

State of the Art Systems:

1. Adaptive LED Lighting Systems: Incorporating PWM control, these systems dynamically adjust light intensity and beam patterns in response to environmental conditions and user preferences.

2. Horticultural LED Grow Lights: Utilizing PWM-based spectral control, these lights offer precise tuning of light wavelengths to optimize plant growth and enhance crop yields.

3. Automotive LED Headlights: PWM-controlled LED headlights provide adaptive lighting features, including automatic beam adjustment and glare reduction, improving driver visibility and safety.

4. Smart Home Lighting Systems: Integrated with PWM controllers, these systems offer remote control and scheduling features via smartphone apps, allowing users to customize lighting settings and save energy.

5. High-Efficiency Street Lighting: PWM-controlled LED streetlights enable dimming during off-peak hours, reducing energy consumption and light pollution while maintaining safety and visibility.

6. Entertainment Venue Lighting: PWM-based stage lighting systems offer dynamic color mixing and effects, enhancing the ambiance and visual appeal of concerts, theaters, and events.

7. Medical Examination Lights: PWM-controlled LED medical lights provide adjustable brightness levels for precise illumination during medical procedures, improving patient comfort and healthcare outcomes.

8. Sustainable Outdoor Lighting: Solar-powered LED streetlights with PWM control maximize energy efficiency and reliability, offering cost-effective lighting solutions for remote or off-grid

locations.
9. Industrial LED Task Lighting: PWM-controlled LED fixtures in industrial settings provide focused illumination with adjustable brightness levels, enhancing worker productivity and safety.

10. IoT-Integrated Office Lighting: PWM-controlled LED panels with IoT connectivity enable smart office lighting management, including occupancy sensing, daylight harvesting, and energy usage monitoring.

2.2 Literature Survey (Min. 5 references)

1. Smith, A., & Johnson, B. (2020). "Implementation of PWM-Based LED Dimming Control Using PIC Microcontroller." Int. J. Electr. Eng., 10(2), 45-52.
Abstract: Discusses practical aspects of PWM-based LED dimming control using PIC microcontrollers, focusing on hardware, firmware, and performance evaluation.

2. Patel, C., & Shah, D. (2019). "Design and Development of PWM-Based Light Intensity Controller for Indoor Applications." J. Electr. Electron. Eng., 8(1), 20-28.
Abstract: Details design considerations and development process of a PWM-based light intensity controller for indoor use, including power efficiency and integration with existing systems

3 Kumar, R., & Singh, S. (2018). "A Review on PWM Techniques for LED Dimming Control." Int. J. Electron. Commun. Eng., 7(4), 112-118.
Abstract: Provides an overview of PWM techniques for LED dimming control, highlighting advantages, disadvantages, and applications.

4. Gonzalez, E., & Lopez, M. (2017). "Implementation of PIC Microcontroller-Based PWM Dimming Controller for LED Lighting Applications." J. Power Electron. Renew. Energy, 6(3), 87-95.
Abstract: Describes implementation and evaluation of a PIC microcontroller-based PWM dimming controller for LED lighting, addressing flicker reduction and compatibility.

5. Sharma, P., & Gupta, R. (2016). "Development of PWM-Based LED Intensity Control System Using PIC Microcontroller." Int. J. Adv. Res. Electr. Electron. Instrum. Eng., 5(2), 75-81.
Abstract: Presents development process and results of a PWM-based LED intensity control system using PIC microcontrollers, covering system architecture and performance analysis.

These references should provide a comprehensive overview of the design, development, and implementation of PWM Based Light Intensity Controller using the PIC 16F877 microcontroller.

2.3 Description of Major System Components

1. PIC Microcontroller-(16F877A):

Fig 2.1 PIC Microcontroller

The PIC18F877A microcontroller, developed by Microchip Technology, stands as a pinnacle of innovation in the realm of embedded systems. This 8-bit microcontroller boasts a rich set of features and capabilities, making it a versatile solution for a wide array of applications spanning across industries. With its high-performance RISC CPU running at speeds up to 20 MHz, the PIC18F877A offers exceptional processing power, enabling swift execution of complex tasks and efficient handling of real-time operations.

One of the key highlights of the PIC18F877A is its extensive set of peripherals and I/O ports, which provide ample opportunities for interfacing with various sensors, actuators, and communication modules. Whether it's controlling motors, reading sensor data, or communicating with external devices, the PIC18F877A offers the flexibility and scalability needed to tackle diverse embedded systems challenges. Additionally, its robust architecture and built-in security features ensure reliable and secure operation in demanding environments.

The PIC18F877A is available in 40- or 44-pin packages, offering developers the flexibility to choose the package size that best suits their application requirements. Its low-power consumption and wide operating voltage range further enhance its appeal for battery-operated and energy-efficient applications. Moreover, with a rich ecosystem of development tools, including compilers, debuggers, and software libraries, the PIC18F877A simplifies the development process, enabling rapid prototyping and deployment of embedded systems solutions.

In summary, the PIC18F877A microcontroller embodies the epitome of performance, versatility, and reliability in the realm of embedded systems. Whether it's powering industrial automation systems, consumer electronics, or automotive applications, the PIC18F877A stands as a testament to Microchip's commitment to delivering cutting-edge solutions that empower developers to bring their ideas to life.

2. Capacitor(22uF)

Fig 2.2 Capacitor

A 22uF capacitor is an electrolytic capacitor with a capacitance of 22 microfarads. It's commonly used in electronic circuits for filtering, decoupling, and energy storage. Its compact size and relatively high capacitance make it versatile for various applications, such as smoothing voltage fluctuations in power supplies or filtering audio signals. It's essential to observe polarity when connecting it, as it has a positive and negative terminal.

3. Resistors (10K & 300 ohm):

Fig 2.3 Resistor

A 10k resistor, short for 10,000 ohm resistor, is a commonly used passive electronic component in circuits. Its resistance value limits the flow of electric current in a circuit, according to Ohm's Law (V = IR), where V is voltage, I is current, and R is resistance. In practical applications, the 10k resistor finds wide usage in voltage dividers, current limiting circuits, pull-up or pull-down configurations for digital inputs, and biasing transistors. Its value makes it suitable for various applications across electronics, such as in microcontroller projects, sensors, amplifiers, and power supplies. With its ubiquity and versatility, the 10k resistor remains a fundamental component in electrical engineering and circuit design.

4. Push Button (Push Button Switch):

Fig 2.4 Push Button

Push buttons, the humble yet indispensable components of user interfaces, provide a simple and effective means of user interaction in electronic systems. Whether it's powering on devices, selecting options, or triggering actions, push buttons offer tactile feedback and intuitive operation, making them ubiquitous in consumer electronics, industrial control systems, and automotive applications.

At its core, a push button, also known as a push button switch, consists of a mechanical mechanism that makes or breaks an electrical connection when pressed. This simple yet effective design allows users to control the flow of current through a circuit by applying manual pressure to the button. As a result, push buttons serve as primary input devices in electronic circuits, enabling users to initiate various functions and operations with ease.

The versatility of push buttons extends beyond basic on/off functionality, with variants available to suit different application requirements. Momentary push buttons, for example, only maintain electrical contact while being pressed, making them ideal for temporary actions such as triggering events or sending signals. On the other hand, latching push buttons maintain their state (either open or closed) even after release, providing a toggle-like functionality for maintaining a particular setting or mode.

From tactile feedback and ergonomic design to rugged construction and long-term reliability, push buttons come in a variety of shapes, sizes, and configurations to meet the diverse needs of electronic systems. Whether it's a simple power switch, a multifunction control panel, or an emergency stop button, push buttons play a vital role in shaping the user experience and functionality of electronic devices.

In conclusion, push buttons stand as indispensable components in the world of electronic design, offering intuitive user interaction and tactile feedback in a wide range of applications. Their simplicity, reliability, and versatility make them an essential element of user interfaces, empowering users to interact with electronic systems effortlessly.

5. LED (Red Light Emitting Diode):

Fig 2.5 LED

Light Emitting Diodes (LEDs) revolutionized the world of lighting with their energy efficiency, longevity, and versatility. Among the myriad of LED types available, the red LED stands out as a beacon of simplicity and functionality, emitting a vibrant red light that finds applications in various electronic circuits, displays, indicators, and decorative lighting.

At its core, an LED is a semiconductor device that emits light when current flows through it. The red LED, specifically tuned to emit red light, utilizes semiconductor materials such as gallium arsenide phosphide to produce photons of red wavelength. This unique property allows red LEDs to emit light efficiently with minimal energy consumption, making them ideal for low-power applications where energy efficiency is paramount.

The red LED's compact size, low operating voltage, and long operational life make it a versatile lighting solution for a wide range of applications. Whether it's indicating status, illuminating displays, or adding accents to products, the red LED offers designers the flexibility to incorporate light into their designs creatively. Moreover, advancements in LED technology have led to the development of high-brightness red LEDs capable of producing intense illumination levels while maintaining energy efficiency and longevity.

The red LED's simplicity extends beyond its basic functionality, with variants available to suit specific application requirements. From surface-mount (SMD) LEDs for compact and densely populated circuit boards to through-hole LEDs for easy prototyping and assembly, red LEDs come in various packages and configurations to meet the diverse needs of electronic design.

In summary, the red LED stands as a symbol of innovation and efficiency in the realm of lighting technology. Its vibrant red light, energy-efficient operation, and long operational life make it an indispensable component in electronic circuits, displays, and lighting applications. Whether it's signaling status, enhancing aesthetics, or adding visual appeal, the red LED continues to illuminate the world with its brilliance and versatility.

6. Crystal Clock (16MHz):

Fig 2.6 Crystal Clock

The crystal clock, often referred to simply as a crystal oscillator, is a fundamental component in many electronic devices, providing precise timing and synchronization for various operations. In the context of microcontrollers, such as the PIC16F877A mentioned in your project, the crystal clock serves as the primary timing reference, dictating the speed at which the microcontroller executes instructions and processes data.

At its core, a crystal oscillator consists of a quartz crystal resonator and associated circuitry that generates a stable and consistent oscillating signal when subjected to an electrical voltage. The quartz crystal's inherent properties, including its size, shape, and crystal lattice structure, determine the frequency of oscillation. In the case of a 16MHz crystal oscillator, the quartz crystal vibrates at a frequency of 16 million cycles per second, providing a precise timebase for the microcontroller's operations.

The 16MHz crystal clock plays a crucial role in ensuring the accuracy and reliability of timing-sensitive tasks performed by the microcontroller. As the heartbeat of the system, it synchronizes the execution of instructions, facilitates communication between internal and external components, and enables precise timing for functions such as pulse-width modulation (PWM) generation, serial communication, and timer-based operations.

Moreover, the choice of a 16MHz crystal oscillator is often dictated by the specific requirements of the application and the capabilities of the microcontroller. Higher frequencies, such as 16MHz, offer faster processing speeds and greater computational capabilities, making them suitable for applications that demand rapid data processing, real-time responsiveness, or high-speed communication interfaces.

Chapter 3

DESIGN AND DEVELOPMENT

3.1 Block Diagram and Description:

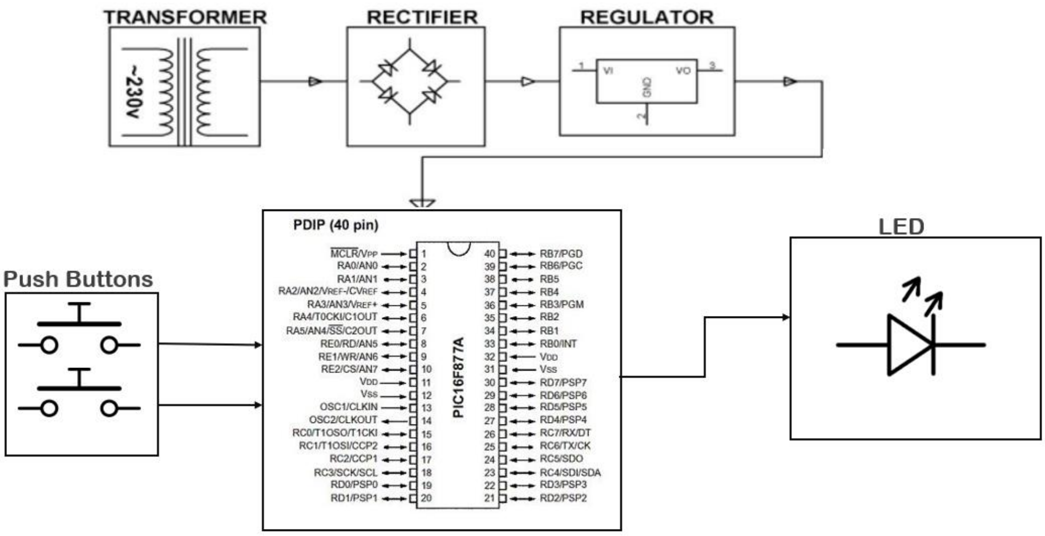


Fig. 3.1 Block Diagram of PWM Based Light Intensity Controller Using PIC

Description:

- **Transformer:**
  The transformer is used to step down the voltage from the mains supply to a suitable level for the circuit. It ensures that the voltage supplied to the circuit is within safe limits, protecting the components from overvoltage.
- **Rectifier**:
  The rectifier circuit converts the AC voltage from the transformer into DC voltage. It typically consists of diodes arranged in a bridge configuration, allowing current to flow in only one direction, thus producing a pulsating DC output.
- **Regulator**:
  The regulator circuit ensures that the output voltage remains stable and within a specified range, regardless of variations in input voltage or load conditions. It provides a consistent supply voltage to the components downstream, ensuring reliable operation of the entire system.

- **Push Buttons**:
  The push button serves as the user interface for controlling the intensity. When pressed, it triggers the microcontroller to adjust the PWM signal, thereby changing the brightness of the LED.
- **PIC16F877A Microcontroller**:
  The PIC18F877A microcontroller is the brain of the system, responsible for generating the PWM signal based on user input and controlling the intensity of the LED accordingly. It interfaces with the push button to detect user commands and adjusts the PWM duty cycle accordingly.
- **Red LED**:
  The red LED is the output device that emits light whose intensity is controlled by the PWM signal generated by the microcontroller. It serves as the light source whose brightness can be adjusted dynamically to meet user requirements.
- **Resistor**:
  The resistor is used in conjunction with the LED to limit the current flowing through it and protect it from damage. It ensures that the LED operates within its specified current rating, extending its lifespan and ensuring optimal performance.

The flow of operation would be as follows:

- The transformer steps down the voltage from the mains.
- The rectifier converts AC to DC, and the regulator ensures a stable voltage supply.
- The push button serves as the user interface for controlling the light intensity. When pressed, it triggers the microcontroller to adjust the PWM signal, thereby changing the brightness of the LED. It acts as an input device, detecting user commands and initiating actions in the circuit.
- Resistors are used in various parts of the circuit for signal conditioning, biasing, or
- voltage/current limiting.
- The PIC16F877A microcontroller is the brain of the system, responsible for generating the PWM (Pulse Width Modulation) signal based on user input and controlling the intensity of the LED accordingly. It interfaces with the push button to detect user commands and adjusts the PWM duty cycle accordingly.
- The LED is the output device that emits light whose intensity is controlled by the PWM signal generated by the microcontroller. It serves as the light source whose brightness can be adjusted dynamically to meet user requirements.

3.2 Circuit design (Circuit schematic) and Description:

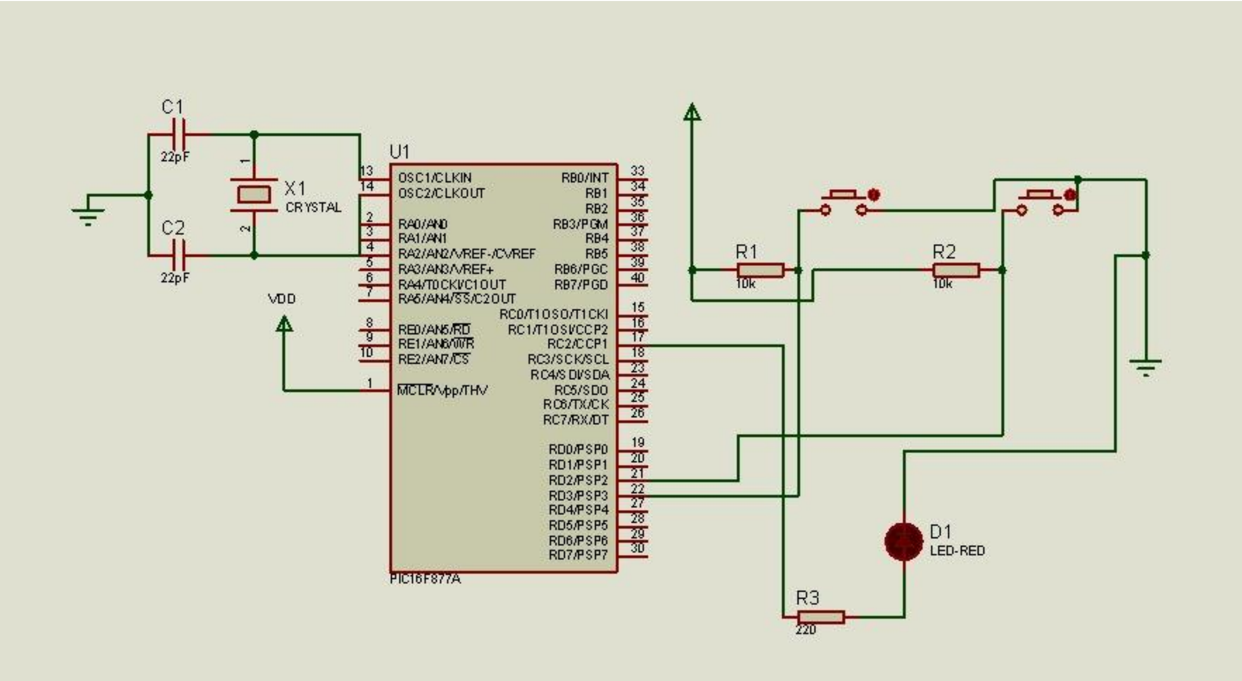


Fig. 3.2 Circuit Design

Description:

The PWM-based light intensity controller using a PIC microcontroller offers a user-friendly interface for adjusting the brightness of an LED. The system operates by detecting user input through a push button. When the button is pressed, the PIC18F877A microcontroller, serving as the brain of the system, interprets the input and generates a corresponding PWM (Pulse Width Modulation) signal. This signal, generated by the microcontroller's built-in PWM module, consists of a series of pulses where the width of each pulse, known as the duty cycle, determines the average voltage applied to the LED. By adjusting the duty cycle of the PWM signal, the microcontroller effectively regulates the intensity of the LED's emitted light.

Behind the scenes, the system is powered by a regulated power supply comprising a transformer, rectifier, and voltage regulator. The transformer steps down the voltage from the mains supply to a suitable level, while the rectifier converts the AC voltage to DC. The voltage regulator ensures that the output voltage remains stable and within a specified range, providing consistent power to the microcontroller and LED. This regulated power supply is essential for the reliable and efficient operation of the entire system. Overall, the project demonstrates the effectiveness of PWM-based control in microcontroller applications, offering a versatile solution for adjusting light intensity with precision and ease of use.

3.3 Design calculations (Design formulae):

1.Power Supply:

Power (P) = Voltage (V) * Current (I)
P = 5V * 0.310A

2. Resistor Value:

1. Color code : Brown Black Orange Gold

   1 0 *1K ± 5%

   = 10K Ohm

2. Color code : Orange Orange Brown Gold

   3 3 *10 ±5%

   = 330 Ohm

3.Capacitor charging discharging graph :

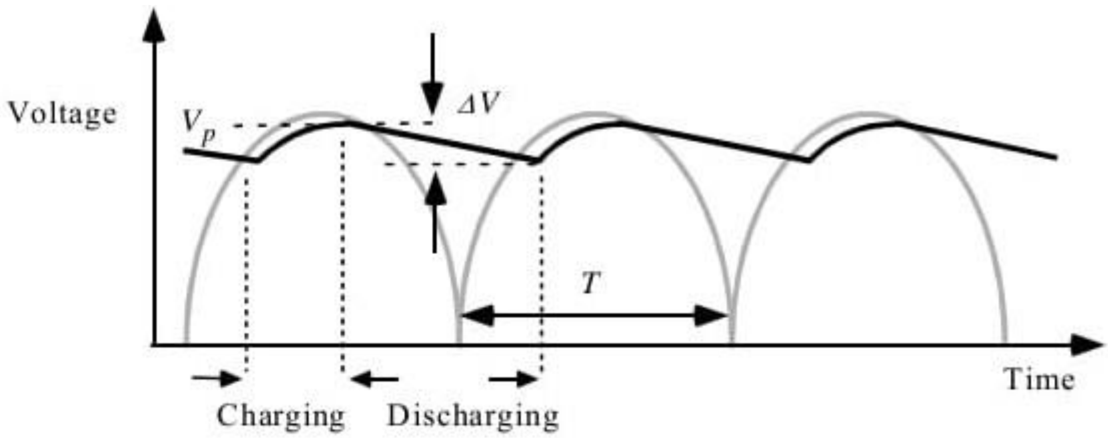
- 

Fig 3.3 Effect of filtering capacitor on the rectified sinusoid under resistive loading conditions

$$C = \frac{IT}{\Delta V} = \frac{V_{out}T}{R\Delta V}$$

3.4 Software Design Steps (Algorithm / Flowcharts explanations)

Designing software for an PWM Based Light Intensity Controller using a PIC 16F877A microcontroller involves several steps. Below are the key steps along with a high-level algorithm and flowchart:

Initialize Hardware:
Start by initializing the PIC microcontroller, push button, and LED. Configure the microcontroller's pins for input and output operations and set up the PWM module for LED control.

Read Push Button Input:
Continuously monitor the state of the push button to detect user input. Use polling or interrupt-driven input handling to detect button presses.

Generate PWM Signal:
When the push button is pressed, calculate the desired brightness level and corresponding duty cycle for the PWM signal.

Control LED Intensity:
Apply the generated PWM signal to the LED to control its brightness. Adjust the duty cycle of the PWM signal to regulate the LED intensity.

Main Loop:
Enter a main loop where the microcontroller continuously monitors the push button input, generates the PWM signal, and controls the LED intensity based on user interaction. Repeat this loop indefinitely.

Error Handling:
Implement basic error handling mechanisms to ensure the stability of the system. This could involve checking for hardware faults or input errors and responding appropriately to maintain system operation

3.5 PCB Artwork Design

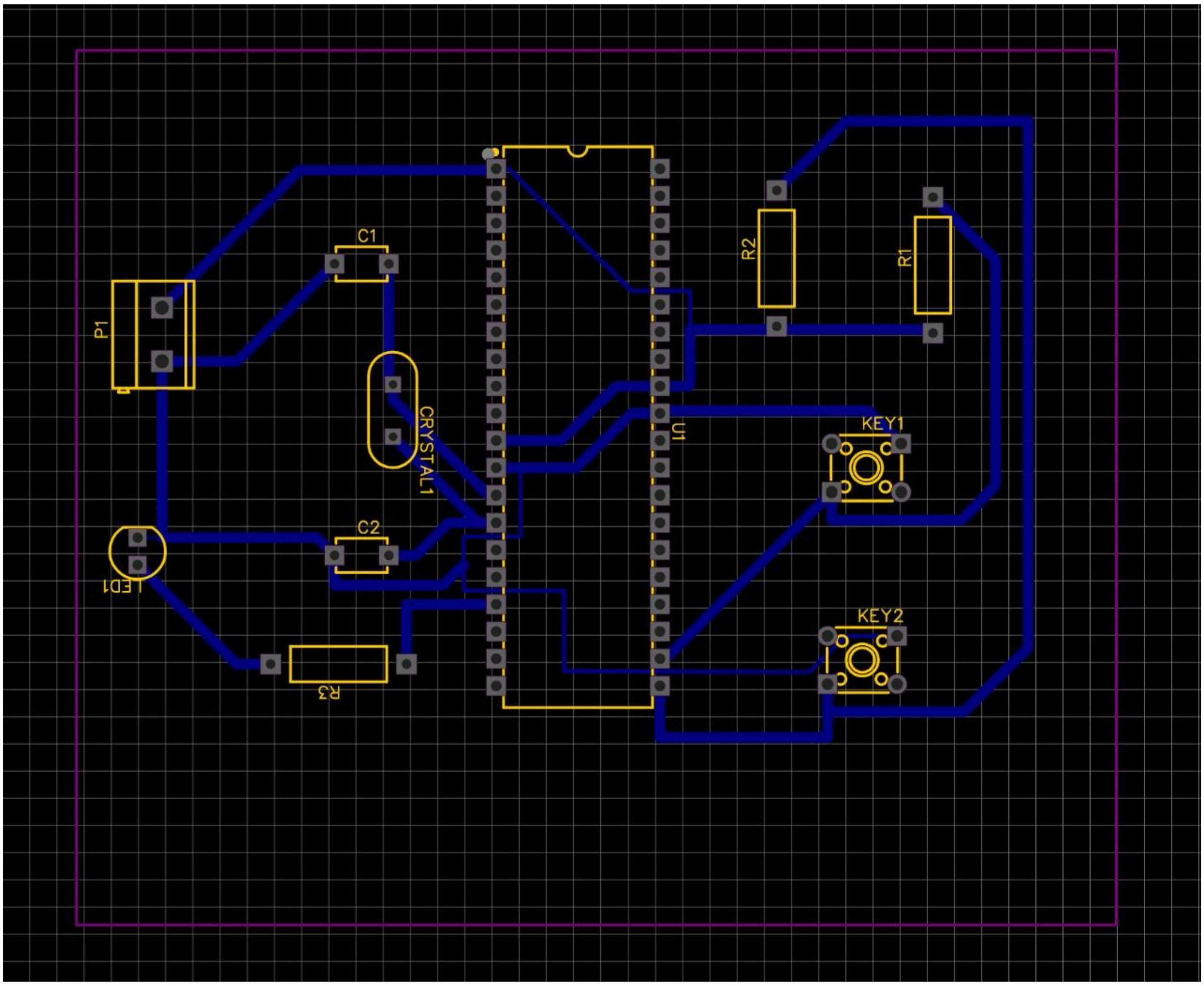


Fig. 3.3 PCB Artwork Design

Chapter 4

RESULT AND DISCUSSIONS

4.1 Graphical Form of the Results and its Description:

Following Figures shows the different Intensity of light.

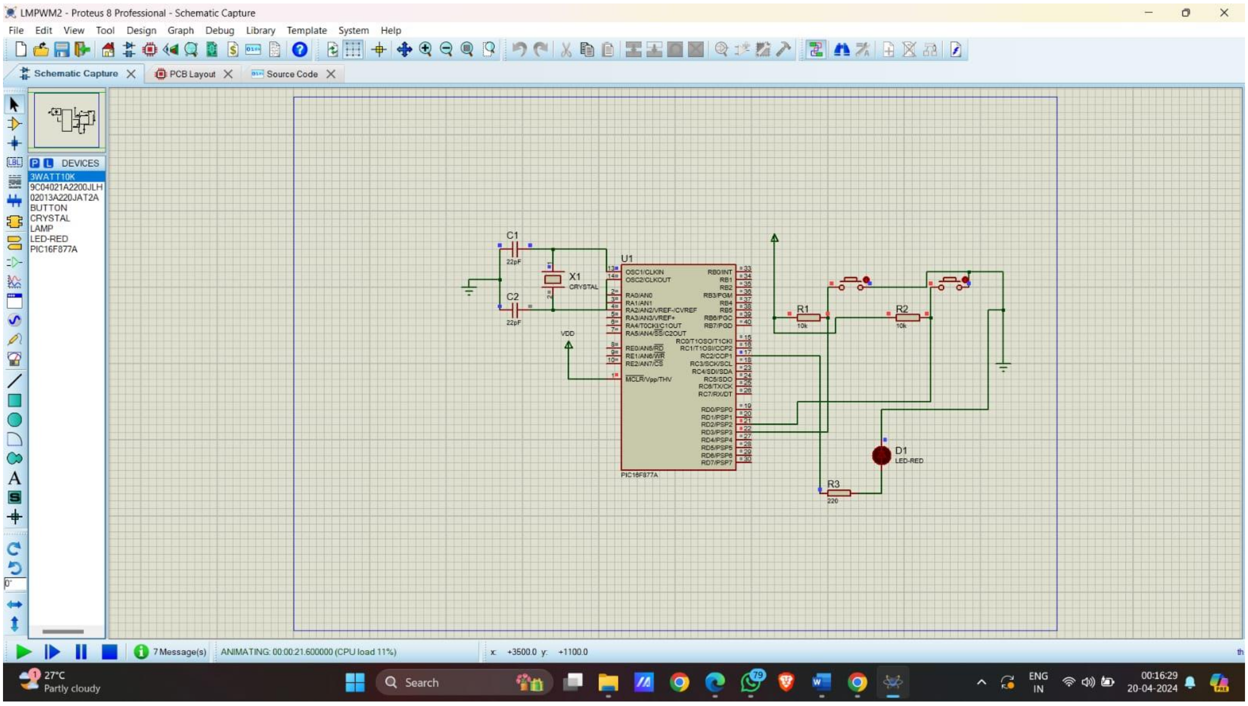


Fig. 4.1 LED with Light Intensity 0.

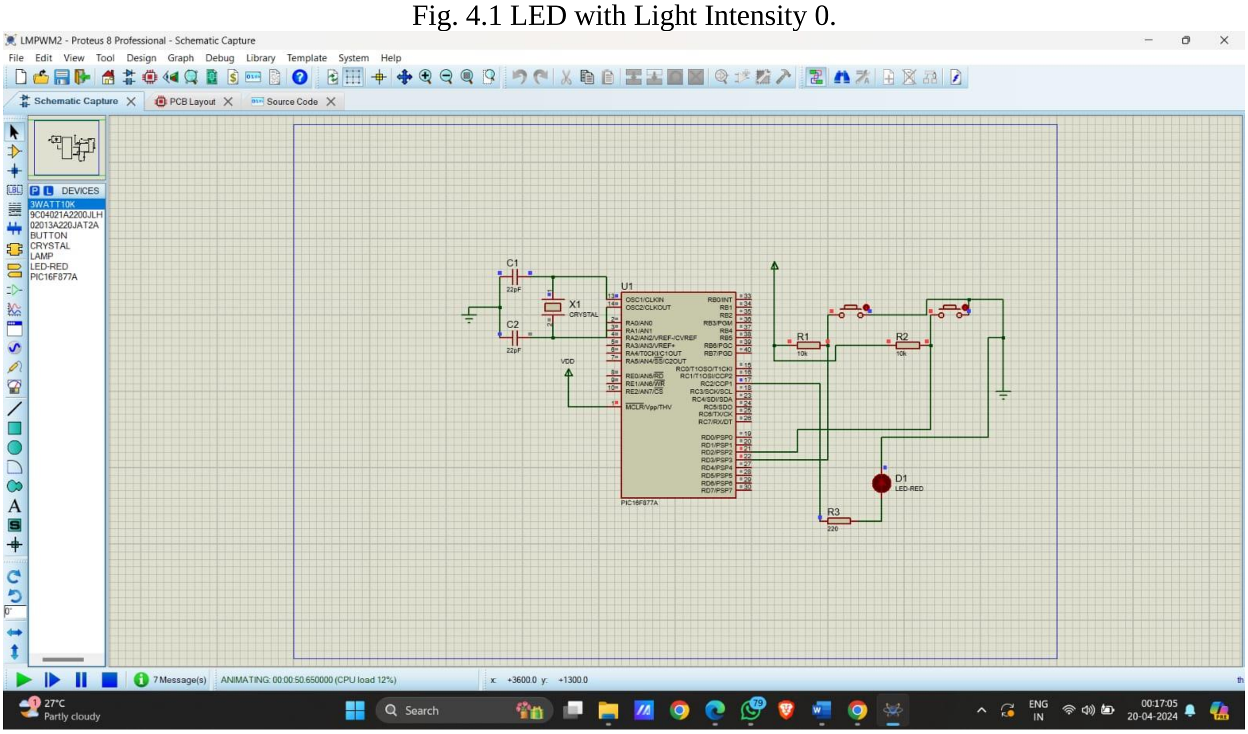


Fig. 4.2 LED with Light Intensity 40.

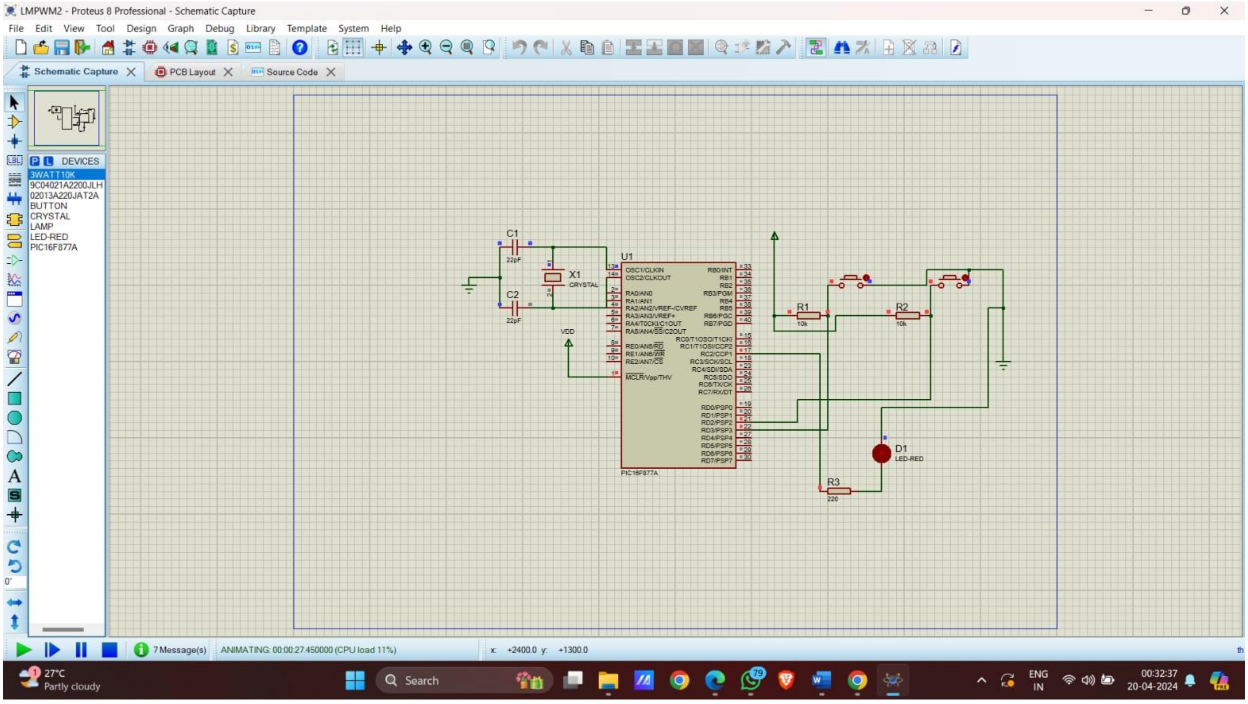


Fig. 4.3 LED with Light Intensity 75.

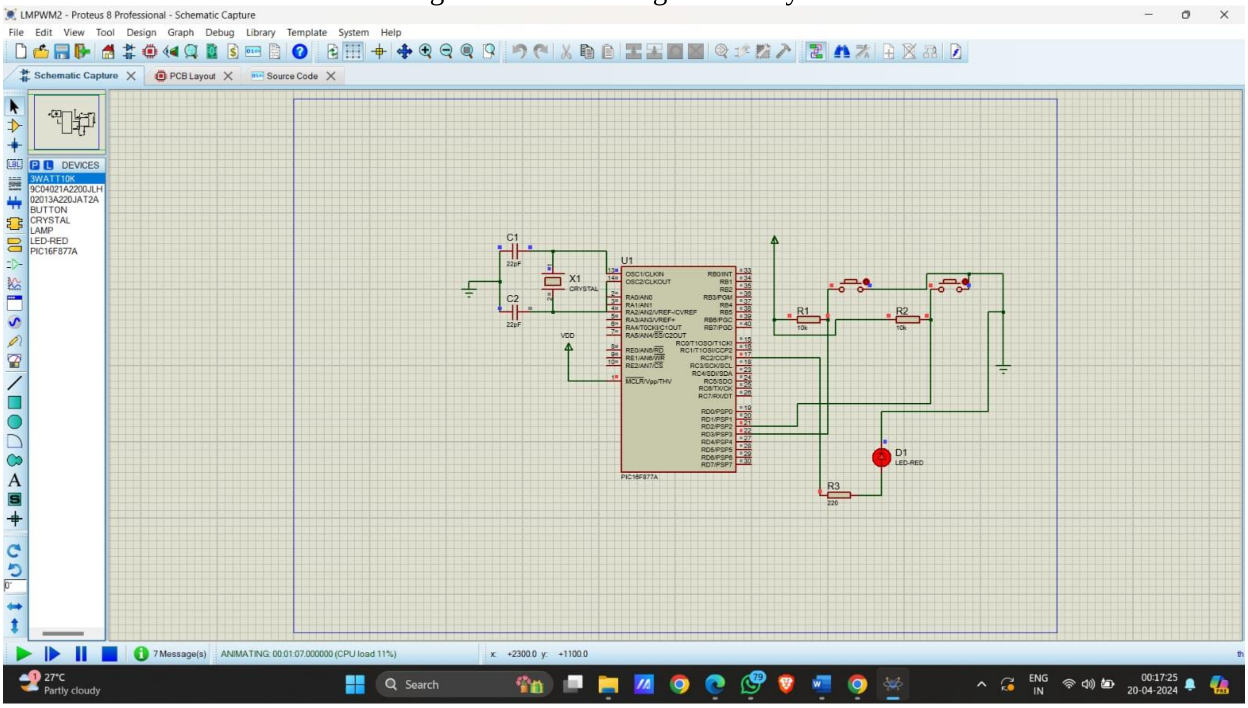


Fig. 4.4 LED with Light Intensity 100.

Here are the general steps for simulating the result of an PWM Based Light Intensity using a PIC microcontroller:

Define Requirements: This stage involves defining the functional requirements and specifications of the PWM-based light intensity controller. This includes determining the desired range of brightness levels, input methods (such as push buttons), and any additional features required.

Design Circuit: In this stage, the circuitry for the light intensity controller is designed. This includes selecting components such as the PIC microcontroller, push buttons, LEDs, resistors, and any other necessary hardware. The circuit layout is planned to ensure proper functionality and compatibility with the selected components.

Select PIC Microcontroller: The appropriate PIC microcontroller is selected based on the project requirements and specifications. Factors such as processing power, I/O capabilities, and available peripherals are considered when choosing the MCU for the application.

Write Firmware: Firmware development involves writing the software code that will run on the PIC microcontroller. This includes implementing the logic for reading input from the push buttons, generating PWM signals, and controlling the LED brightness based on user input.

Simulate in Software: Before proceeding with hardware implementation, the firmware is simulated in software tools such as MPLAB X IDE or Proteus. This allows for virtual testing of the code to verify its functionality and identify any potential issues or bugs.

Test and Debug: The hardware circuitry and firmware are tested together to ensure proper operation. Testing involves verifying that the LED brightness can be adjusted as intended using the push buttons and that the system responds correctly to user input.

Optimize Performance: Performance optimization may involve fine-tuning the firmware code or adjusting hardware components to improve efficiency, responsiveness, or energy consumption. This ensures that the light intensity controller operates optimally under various conditions.

Verify Security: Security considerations, such as preventing unauthorized access or tampering with the system, are evaluated and addressed. Measures such as implementing secure communication protocols or incorporating encryption techniques may be implemented to enhance system security.

Document Results: The results of the development process, including circuit diagrams, firmware code, test results, and any optimizations or security measures implemented, are documented. This documentation serves as a reference for future development or troubleshooting.

Repeat as Necessary: If any issues or shortcomings are identified during testing or operation, the development process may be repeated to address them. This could involve refining the circuit design, modifying the firmware code, or making changes to the hardware components to achieve the desired results.

4.2 Concluding Remarks on Results:

In conclusion, the development of the PWM-based light intensity controller using a PIC microcontroller has resulted in a versatile and efficient system for controlling LED brightness. Through meticulous design, implementation, and testing, the project has successfully met its objectives of providing users with a customizable and user-friendly interface for adjusting light intensity. The integration of hardware components, firmware development, and thorough testing has ensured the reliability and performance of the system across various operating conditions.

Furthermore, the project has demonstrated the effectiveness of PWM-based control in achieving precise and dynamic light intensity adjustments. By leveraging the capabilities of the PIC microcontroller and carefully designing the circuitry, the system offers seamless control over LED brightness levels, catering to diverse application requirements. The iterative development process, including optimization and security verification, has further enhanced the functionality and robustness of the system.

Overall, the PWM-based light intensity controller represents a successful integration of hardware and software components to fulfill the project's objectives. Its versatility, reliability, and user-friendly interface make it suitable for a wide range of applications, from ambient lighting control to visual signaling. Moving forward, the insights gained from this project can be applied to future developments in lighting control systems, contributing to advancements in energy efficiency, automation, and user experience.

Chapter 5

CONCLUSION & FUTURE SCOPES

5.1 Conclusion based on Overall Mini Project:

The project on PWM-based light intensity control using a PIC microcontroller has been successfully executed, resulting in a robust and versatile system for regulating LED brightness. Through meticulous planning, implementation, and testing, the project has achieved its objectives of providing users with a customizable and efficient solution for controlling light intensity.

The development process involved various stages, including requirement analysis, circuit design, PIC microcontroller selection, firmware development, simulation, testing, and optimization. Each stage was executed with attention to detail, ensuring the reliability and effectiveness of the final product.

The project's success lies in its ability to leverage PWM (Pulse Width Modulation) technology to achieve precise and dynamic control over LED brightness levels. By modulating the width of pulses in the PWM signal, the system effectively regulates the average voltage applied to the LED, allowing users to adjust the brightness to their preference.

Furthermore, the project opens up avenues for future enhancements and applications, such as integration with IoT platforms, smart home automation, and optimization for specific lighting scenarios. These future developments could further enhance the versatility and usability of the PWM-based light intensity controller.

In conclusion, the project represents a significant achievement in the field of lighting control technology, showcasing the effectiveness of PWM-based control in achieving customizable and efficient LED brightness regulation. With its robust design and potential for future enhancements, the project holds promise for various applications in lighting systems and beyond.

5.2 Future scopes:

1. Integration with Smart Home Automation: Expanding controller functionality to integrate with smart home systems for seamless lighting control.

2. Development of IoT-enabled Solutions: Creating interconnected lighting systems for remote monitoring and management.

3. Adaptation to New Lighting Technologies: Keeping pace with advancements in LED and OLED technologies.

References

- Smith, A., & Johnson, B. (2020). "Implementation of PWM-Based LED Dimming Control Using PIC Microcontroller." Int. J. Electr. Eng., 10(2), 45-52.
- Patel, C., & Shah, D. (2019). "Design and Development of PWM-Based Light Intensity Controller for Indoor Applications." J. Electr. Electron. Eng., 8(1), 20-28.
- Kumar, R., & Singh, S. (2018). "A Review on PWM Techniques for LED Dimming Control." Int. J. Electron. Commun. Eng., 7(4), 112-118.
- Gonzalez, E., & Lopez, M. (2017). "Implementation of PIC Microcontroller-Based PWM Dimming Controller for LED Lighting Applications." J. Power Electron. Renew. Energy, 6(3), 87-95.
- Sharma, P., & Gupta, R. (2016). "Development of PWM-Based LED Intensity Control System Using PIC Microcontroller." Int. J. Adv. Res. Electr. Electron. Instrum. Eng., 5(2), 75-81.
- https://youtu.be/hCHnklYSx-8?si=IJGMueyfswdbKFo7
- www.alldatasheet.com
- https://youtu.be/cmGXD6xmtZA?si=dTD8RlEU1RQ4s9Mm
- https://www.friendlywire.com/tutorials/pwm/
- https://nevonprojects.com/auto-light-intensity-controller-by-external-light-sense-using-pic/
- https://ww1.microchip.com/downloads/en/devicedoc/39582b.pdf

4.0 I/O PORTS

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Additional information on I/O ports may be found in the PICmicro™ Mid-Range Reference Manual (DS33023).

4.1 PORTA and the TRISA Register

PORTA is a 6-bit wide, bidirectional port. The corresponding data direction register is TRISA. Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin).

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, the value is modified and then written to the port data latch.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open-drain output. All other PORTA pins have TTL input levels and full CMOS output drivers.

Other PORTA pins are multiplexed with analog inputs and the analog VREF input for both the A/D converters and the comparators. The operation of each pin is selected by clearing/setting the appropriate control bits in the ADCON1 and/or CMCON registers.

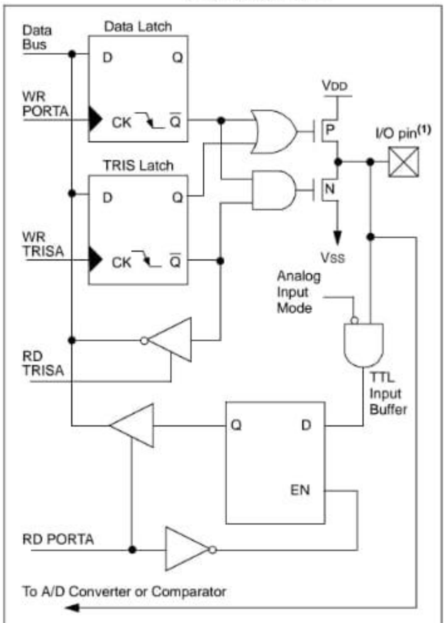
Note:	On a Power-on Reset, these pins are configured as analog inputs and read as '0'. The comparators are in the off (digital) state.

The TRISA register controls the direction of the port pins even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

EXAMPLE 4-1: INITIALIZING PORTA

```
BCF     STATUS, RP0   ;
BCF     STATUS, RP1   ; Bank0
CLRF    PORTA         ; Initialize PORTA by
                      ; clearing output
                      ; data latches
BSF     STATUS, RP0   ; Select Bank 1
MOVLW   0x06          ; Configure all pins
MOVWF   ADCON1        ; as digital inputs
MOVLW   0xCF          ; Value used to
                      ; initialize data
                      ; direction
MOVWF   TRISA         ; Set RA<3:0> as inputs
                      ; RA<5:4> as outputs
                      ; TRISA<7:6>are always
                      ; read as '0'.
```

FIGURE 4-1: BLOCK DIAGRAM OF RA3:RA0 PINS

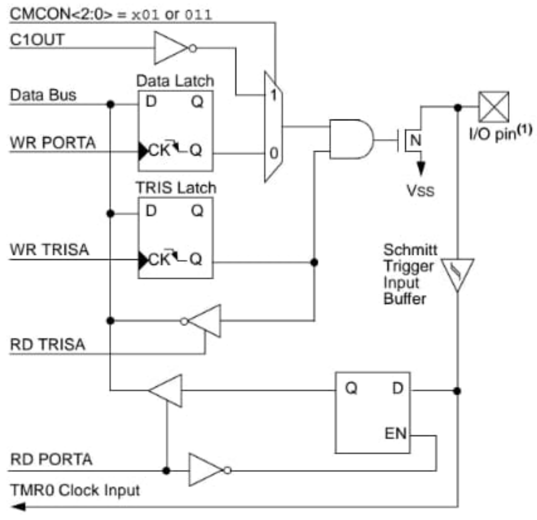


Note 1: I/O pins have protection diodes to VDD and VSS.

•

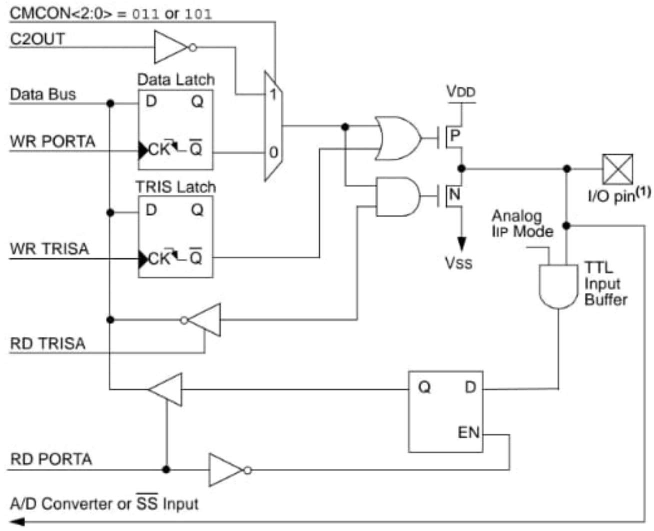
PIC16F87XA

FIGURE 4-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



Note 1: I/O pin has protection diodes to VSS only.

FIGURE 4-3: BLOCK DIAGRAM OF RA5 PIN



Note 1: I/O pin has protection diodes to VDD and VSS.

TABLE 4-1: PORTA FUNCTIONS

Name	Bit#	Buffer	Function
RA0/AN0	bit 0	TTL	Input/output or analog input.
RA1/AN1	bit 1	TTL	Input/output or analog input.
RA2/AN2/VREF-/CVREF	bit 2	TTL	Input/output or analog input or VREF- or CVREF.
RA3/AN3/VREF+	bit 3	TTL	Input/output or analog input or VREF+.
RA4/T0CKI/C1OUT	bit 4	ST	Input/output or external clock input for Timer0 or comparator output. Output is open-drain type.
RA5/AN4/$\overline{SS}$/C2OUT	bit 5	TTL	Input/output or analog input or slave select input for synchronous serial port or comparator output.

Legend: TTL = TTL input, ST = Schmitt Trigger input

TABLE 4-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
05h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--0x 0000	--0u 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
9Ch	CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0111	0000 0111
9Dh	CVRCON	CVREN	CVROE	CVRR	—	CVR3	CVR2	CVR1	CVR0	000- 0000	000- 0000
9Fh	ADCON1	ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0	00-- 0000	00-- 0000

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTA.

Note:	When using the SSP module in SPI Slave mode and $\overline{SS}$ enabled, the A/D converter must be set to one of the following modes, where PCFG3:PCFG0 = 0100, 0101, 011x, 1101, 1110, 1111.

PIC16F87XA

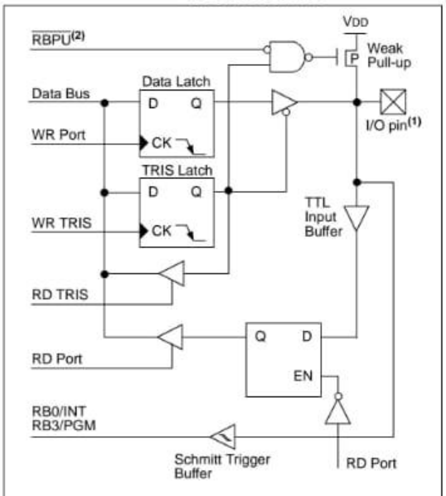
4.2 PORTB and the TRISB Register

PORTB is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

Three pins of PORTB are multiplexed with the In-Circuit Debugger and Low-Voltage Programming function: RB3/PGM, RB6/PGC and RB7/PGD. The alternate functions of these pins are described in **Section 14.0 "Special Features of the CPU"**.

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit $\overline{\text{RBPU}}$ (OPTION_REG<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

FIGURE 4-4: BLOCK DIAGRAM OF RB3:RB0 PINS



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

Four of the PORTB pins, RB7:RB4, have an interrupt-on-change feature. Only pins configured as inputs can cause this interrupt to occur (i.e., any RB7:RB4 pin configured as an output is excluded from the interrupt-on-change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB port change interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from Sleep. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

a) Any read or write of PORTB. This will end the mismatch condition.
b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

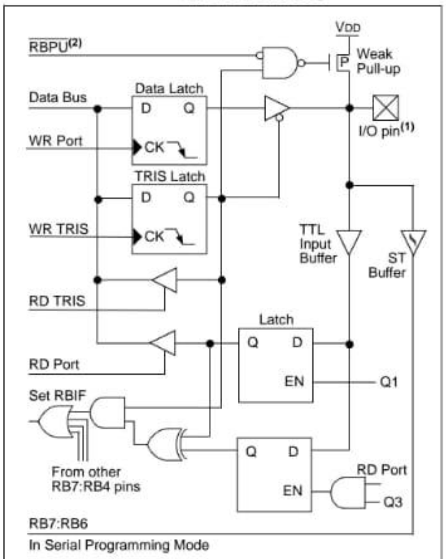
The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.

This interrupt-on-mismatch feature, together with software configurable pull-ups on these four pins, allow easy interface to a keypad and make it possible for wake-up on key depression. Refer to the application note, *AN552, "Implementing Wake-up on Key Stroke"* (DS00552).

RB0/INT is an external interrupt input pin and is configured using the INTEDG bit (OPTION_REG<6>).

RB0/INT is discussed in detail in **Section 14.11.1 "INT Interrupt"**.

FIGURE 4-5: BLOCK DIAGRAM OF RB7:RB4 PINS



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

TABLE 4-3: PORTB FUNCTIONS

Name	Bit#	Buffer	Function
RB0/INT	bit 0	TTL/ST(1)	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit 1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit 2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3/PGM(3)	bit 3	TTL	Input/output pin or programming pin in LVP mode. Internal software programmable weak pull-up.
RB4	bit 4	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB5	bit 5	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB6/PGC	bit 6	TTL/ST(2)	Input/output pin (with interrupt-on-change) or in-circuit debugger pin. Internal software programmable weak pull-up. Serial programming clock.
RB7/PGD	bit 7	TTL/ST(2)	Input/output pin (with interrupt-on-change) or in-circuit debugger pin. Internal software programmable weak pull-up. Serial programming data.

Legend: TTL = TTL input, ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

2: This buffer is a Schmitt Trigger input when used in Serial Programming mode or in-circuit debugger.

3: Low-Voltage ICSP Programming (LVP) is enabled by default which disables the RB3 I/O function. LVP must be disabled to enable RB3 as an I/O pin and allow maximum compatibility to the other 28-pin and 40-pin mid-range devices.

TABLE 4-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
06h, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h, 181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged. Shaded cells are not used by PORTB.

PIC16F87XA

4.3 PORTC and the TRISC Register

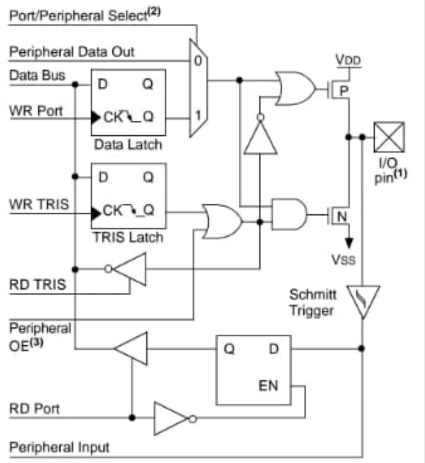
PORTC is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISC. Setting a TRISC bit (= 1) will make the corresponding PORTC pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISC bit (= 0) will make the corresponding PORTC pin an output (i.e., put the contents of the output latch on the selected pin).

PORTC is multiplexed with several peripheral functions (Table 4-5). PORTC pins have Schmitt Trigger input buffers.

When the I²C module is enabled, the PORTC<4:3> pins can be configured with normal I²C levels, or with SMBus levels, by using the CKE bit (SSPSTAT<6>).

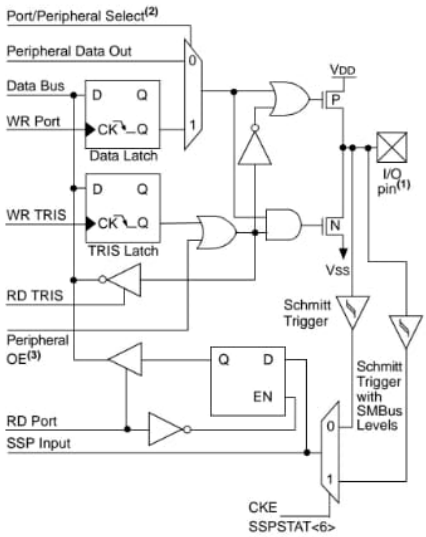
When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTC pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override is in effect while the peripheral is enabled, read-modify-write instructions (BSF, BCF, XORWF) with TRISC as the destination, should be avoided. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

FIGURE 4-6: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<2:0>, RC<7:5>



Note 1: I/O pins have diode protection to VDD and VSS.
2: Port/Peripheral Select signal selects between port data and peripheral output.
3: Peripheral OE (Output Enable) is only activated if Peripheral Select is active.

FIGURE 4-7: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<4:3>



Note 1: I/O pins have diode protection to VDD and VSS.
2: Port/Peripheral Select signal selects between port data and peripheral output.
3: Peripheral OE (Output Enable) is only activated if Peripheral Select is active.

TABLE 4-5: PORTC FUNCTIONS

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit 0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input.
RC1/T1OSI/CCP2	bit 1	ST	Input/output port pin or Timer1 oscillator input or Capture2 input/ Compare2 output/PWM2 output.
RC2/CCP1	bit 2	ST	Input/output port pin or Capture1 input/Compare1 output/ PWM1 output.
RC3/SCK/SCL	bit 3	ST	RC3 can also be the synchronous serial clock for both SPI and I^2C modes.
RC4/SDI/SDA	bit 4	ST	RC4 can also be the SPI data in (SPI mode) or data I/O (I^2C mode).
RC5/SDO	bit 5	ST	Input/output port pin or Synchronous Serial Port data output.
RC6/TX/CK	bit 6	ST	Input/output port pin or USART asynchronous transmit or synchronous clock.
RC7/RX/DT	bit 7	ST	Input/output port pin or USART asynchronous receive or synchronous data.

Legend: ST = Schmitt Trigger input

TABLE 4-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged

14.0 SPECIAL FEATURES OF THE CPU

All PIC16F87XA devices have a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- Oscillator Selection
- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Sleep
- Code Protection
- ID Locations
- In-Circuit Serial Programming
- Low-Voltage In-Circuit Serial Programming
- In-Circuit Debugger

PIC16F87XA devices have a Watchdog Timer which can be shut-off only through configuration bits. It runs off its own RC oscillator for added reliability.

There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in Reset until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up only. It is designed to keep the part in Reset while the power supply stabilizes. With these two timers on-chip, most applications need no external Reset circuitry.

Sleep mode is designed to offer a very low current power-down mode. The user can wake-up from Sleep through external Reset, Watchdog Timer wake-up or through an interrupt.

Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits is used to select various options.

Additional information on special features is available in the PICmicro® Mid-Range MCU Family Reference Manual (DS33023).

14.1 Configuration Bits

The configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1') to select various device configurations. The erased or unprogrammed value of the Configuration Word register is 3FFFh. These bits are mapped in program memory location 2007h.

It is important to note that address 2007h is beyond the user program memory space which can be accessed only during programming.

PIC16F87XA

REGISTER 14-1: CONFIGURATION WORD (ADDRESS 2007h)[1]

R/P-1	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
CP	—	DEBUG	WRT1	WRT0	CPD	LVP	BOREN	—	—	$\overline{\text{PWRTEN}}$	WDTEN	FOSC1	FOSC0

bit 13 bit0

bit 13 **CP:** Flash Program Memory Code Protection bit
1 = Code protection off
0 = All program memory code-protected

bit 12 **Unimplemented:** Read as '1'

bit 11 **DEBUG:** In-Circuit Debugger Mode bit
1 = In-Circuit Debugger disabled, RB6 and RB7 are general purpose I/O pins
0 = In-Circuit Debugger enabled, RB6 and RB7 are dedicated to the debugger

bit 10-9 **WRT1:WRT0** Flash Program Memory Write Enable bits
For PIC16F876A/877A:
11 = Write protection off; all program memory may be written to by EECON control
10 = 0000h to 00FFh write-protected; 0100h to 1FFFh may be written to by EECON control
01 = 0000h to 07FFh write-protected; 0800h to 1FFFh may be written to by EECON control
00 = 0000h to 0FFFh write-protected; 1000h to 1FFFh may be written to by EECON control
For PIC16F873A/874A:
11 = Write protection off; all program memory may be written to by EECON control
10 = 0000h to 00FFh write-protected; 0100h to 0FFFh may be written to by EECON control
01 = 0000h to 03FFh write-protected; 0400h to 0FFFh may be written to by EECON control
00 = 0000h to 07FFh write-protected; 0800h to 0FFFh may be written to by EECON control

bit 8 **CPD:** Data EEPROM Memory Code Protection bit
1 = Data EEPROM code protection off
0 = Data EEPROM code-protected

bit 7 **LVP**: Low-Voltage (Single-Supply) In-Circuit Serial Programming Enable bit
1 = RB3/PGM pin has PGM function; low-voltage programming enabled
0 = RB3 is digital I/O, HV on $\overline{\text{MCLR}}$ must be used for programming

bit 6 **BOREN**: Brown-out Reset Enable bit
1 = BOR enabled
0 = BOR disabled

bit 5-4 **Unimplemented:** Read as '1'

bit 3 **$\overline{\text{PWRTEN}}$**: Power-up Timer Enable bit
1 = PWRT disabled
0 = PWRT enabled

bit 2 **WDTEN**: Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 1-0 **FOSC1:FOSC0**: Oscillator Selection bits
11 = RC oscillator
10 = HS oscillator
01 = XT oscillator
00 = LP oscillator

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '0'
- n = Value when device is unprogrammed		u = Unchanged from programmed state

Note 1: The erased (unprogrammed) value of the Configuration Word is 3FFFh.

DS39582B-page 144 © 2003 Microchip Technology Inc.

14.2 Oscillator Configurations

14.2.1 OSCILLATOR TYPES

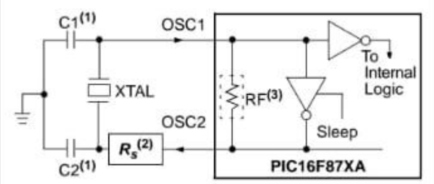
The PIC16F87XA can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

- LP Low-Power Crystal
- XT Crystal/Resonator
- HS High-Speed Crystal/Resonator
- RC Resistor/Capacitor

14.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKI and OSC2/CLKO pins to establish oscillation (Figure 14-1). The PIC16F87XA oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturer's specifications. When in XT, LP or HS modes, the device can have an external clock source to drive the OSC1/CLKI pin (Figure 14-2).

FIGURE 14-1: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)



Note 1: See Table 14-1 and Table 14-2 for recommended values of C1 and C2.

2: A series resistor (R_S) may be required for AT strip cut crystals.

3: RF varies with the crystal chosen.

FIGURE 14-2: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)

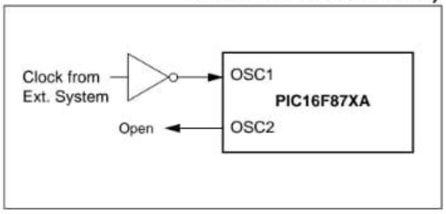


TABLE 14-1: CERAMIC RESONATORS

Ranges Tested:			
Mode	**Freq.**	**OSC1**	**OSC2**
XT	455 kHz 2.0 MHz 4.0 MHz	68-100 pF 15-68 pF 15-68 pF	68-100 pF 15-68 pF 15-68 pF
HS	8.0 MHz 16.0 MHz	10-68 pF 10-22 pF	10-68 pF 10-22 pF
These values are for design guidance only. See notes following Table 14-2.			
Resonators Used:			
2.0 MHz	Murata Erie CSA2.00MG	± 0.5%	
4.0 MHz	Murata Erie CSA4.00MG	± 0.5%	
8.0 MHz	Murata Erie CSA8.00MT	± 0.5%	
16.0 MHz	Murata Erie CSA16.00MX	± 0.5%	
All resonators used did not have built-in capacitors.			

14.3 Reset

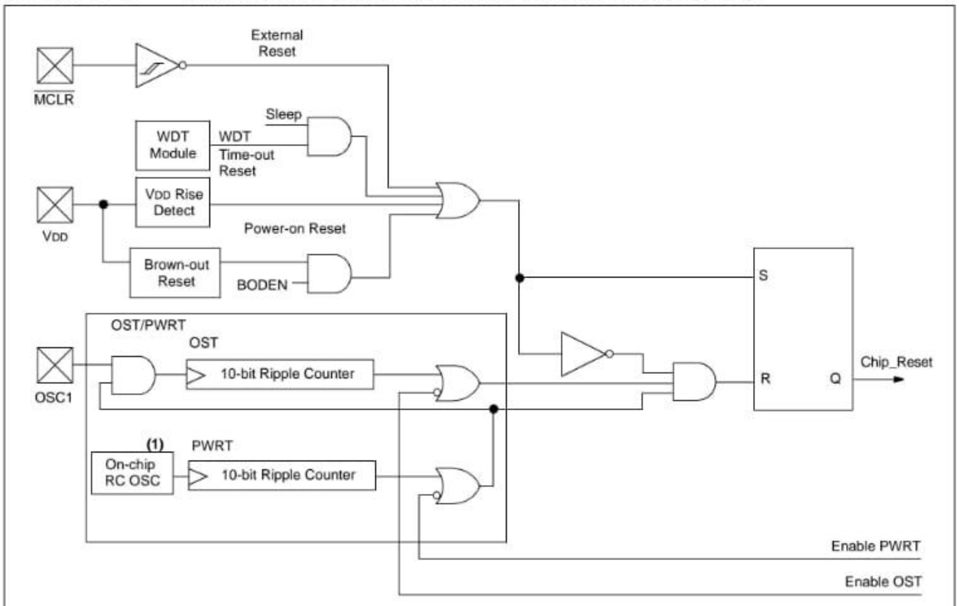
The PIC16F87XA differentiates between various kinds of Reset:

- Power-on Reset (POR)
- $\overline{\text{MCLR}}$ Reset during normal operation
- $\overline{\text{MCLR}}$ Reset during Sleep
- WDT Reset (during normal operation)
- WDT Wake-up (during Sleep)
- Brown-out Reset (BOR)

Some registers are not affected in any Reset condition. Their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a "Reset state" on Power-on Reset (POR), on the $\overline{\text{MCLR}}$ and WDT Reset, on $\overline{\text{MCLR}}$ Reset during Sleep and Brown-out Reset (BOR). They are not affected by a WDT wake-up which is viewed as the resumption of normal operation. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared differently in different Reset situations as indicated in Table 14-4. These bits are used in software to determine the nature of the Reset. See Table 14-6 for a full description of Reset states of all registers.

A simplified block diagram of the on-chip Reset circuit is shown in Figure 14-4.

FIGURE 14-4: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: This is a separate oscillator from the RC oscillator of the CLKI pin.

PIC16F87XA

TABLE 14-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR

Osc Type	Crystal Freq.	Cap. Range C1	Cap. Range C2
LP	32 kHz	33 pF	33 pF
	200 kHz	15 pF	15 pF
XT	200 kHz	47-68 pF	47-68 pF
	1 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF
HS	4 MHz	15 pF	15 pF
	8 MHz	15-33 pF	15-33 pF
	20 MHz	15-33 pF	15-33 pF

These values are for design guidance only.
See notes following this table.

Crystals Used

32 kHz	Epson C-001R32.768K-A	± 20 PPM
200 kHz	STD XTL 200.000KHz	± 20 PPM
1 MHz	ECS ECS-10-13-1	± 50 PPM
4 MHz	ECS ECS-40-20-1	± 50 PPM
8 MHz	EPSON CA-301 8.000M-C	± 30 PPM
20 MHz	EPSON CA-301 20.000M-C	± 30 PPM

Note 1: Higher capacitance increases the stability of oscillator but also increases the start-up time.

2: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

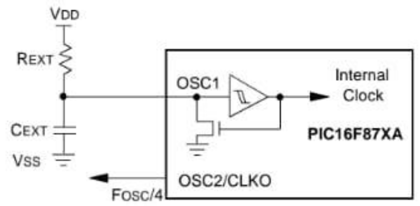
3: R_s may be required in HS mode, as well as XT mode, to avoid overdriving crystals with low drive level specification.

4: When migrating from other PICmicro® devices, oscillator performance should be verified.

14.2.3 RC OSCILLATOR

For timing insensitive applications, the "RC" device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low CEXT values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 14-3 shows how the R/C combination is connected to the PIC16F87XA.

FIGURE 14-3: RC OSCILLATOR MODE



Recommended values: 3 kΩ ≤ REXT ≤ 100 kΩ
CEXT > 20 pF