vehicle type identification system using an 8051 microcontroller

Project Report

Submitted in the partial fulfillment of the requirements for the

Course Title: PROCESSORS &

CONTROLLERS

Course code: 22EC2106

submitted by

GANGA RAJ	2200033047

UNDER THE GUIDANCE OF

Dr.RAGHAVA PRASAD

Associate Professor

KL UNIVERSITY

Green fields, Vaddeswaram – 522 302

Guntur Dt., AP, India.

Declaration

The Project Report entitled "**Security System: Activation of alarm on unauthorized access using 8051** " is a record of Bonafede work of SATHWIK , NITHIN , GANGA RAJfollowed by IDNO 2200033027, 2200033047, 2200030054 submitted in partial fulfillment for the subject titled 22EC2106 - PROCESSORS AND CONTROLLERS in Dept of ECE, KL University. The results embodied in this reporthave not been copied from any other departments/University/ Institute.

Name:

2200033027 – SATHWIK.M
2200033047 - GANGA RAJ
2200033054 – NITHIN SAI

Certification

This is to certify that the Project Report entitled "**The Stepper Motor Control Using 8051 Microcontroller**" is being submitted by SATHWIK , NITHIN , GANGA RAJfollowed by IDNO 2200033027, 2200033047, 2200030054 in partial fulfillment for the subject titled (22EC2106 - PROCESSORS AND CONTROLLERS) in Dept of ECE, KL University is a record of Bonafede work carried out under our guidance andsupervision. The results embodied in this report have not been copied from any other departments/ University/ Institute.

Signature of Examiner **Signature of Supervisor**

Acknowledgement

It is a great pleasure for me to express my gratitude to our honorable President Sri.Koneru Satyanarayana, for giving me the opportunity and platform with facilities in accomplishing the project-based laboratory report.

I express my sincere gratitude to our principal Dr. T Rama Krishna for his administration towards our academic growth.

I record it as my privilege to deeply thank our pioneer HOD-ECE for providing us with the efficient faculty and facilities to realize our ideas.

I sincerely thank our project supervisor **"Dr. Prasad", Associate Professor, "Dept of ECE"** for his novel association of ideas, encouragement, appreciation, and intellectual zeal which motivated us to venture into this project successfully.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to making this project report successful.

Name:
2200033027 – SATHWIK.M
2200033047 - GANGA RAJ
2200033054 – NITHIN SAI

CONTENTS

ABSTRACT

The Security System utilizing the 8051 microcontroller is designed to swiftly respond to unauthorized access. Through strategically placed sensors, including proximity detectors and motion sensors, the system continuously monitors the environment. Once unauthorized access is detected, the 8051 microcontroller, acting as the system's core, triggers an alarm system. This alarm can be a blend of visual and auditory alerts, ensuring immediate attention and deterring intruders effectively. The system's programming logic defines the activation criteria, offering flexibility and customization options. The 8051 microcontroller's real-time processing capabilities, low power consumption, and seamless sensor integration make it an ideal choice. Additionally, the system can be expanded to include remote monitoring and event logging features for enhanced security management. Overall, this Security System with 8051 microcontroller integration stands as a reliable and scalable solution for bolstering security across various settings.

The integration of the 8051 microcontroller offers several advantages, including real-time processing, low power consumption, and ease of interfacing with various sensors and actuators. Additionally, the system can be expanded with additional features such as remote monitoring, logging of intrusion events, and integration with other security protocols.

In conclusion, the Security System: Alarm Activation on Unauthorized Access using 8051 Microcontroller provides an effective and scalable solution for enhancing security in residential, commercial, and industrial environments. Its ability to detect and respond to unauthorized access swiftly makes it a valuable asset in comprehensive security strategies.

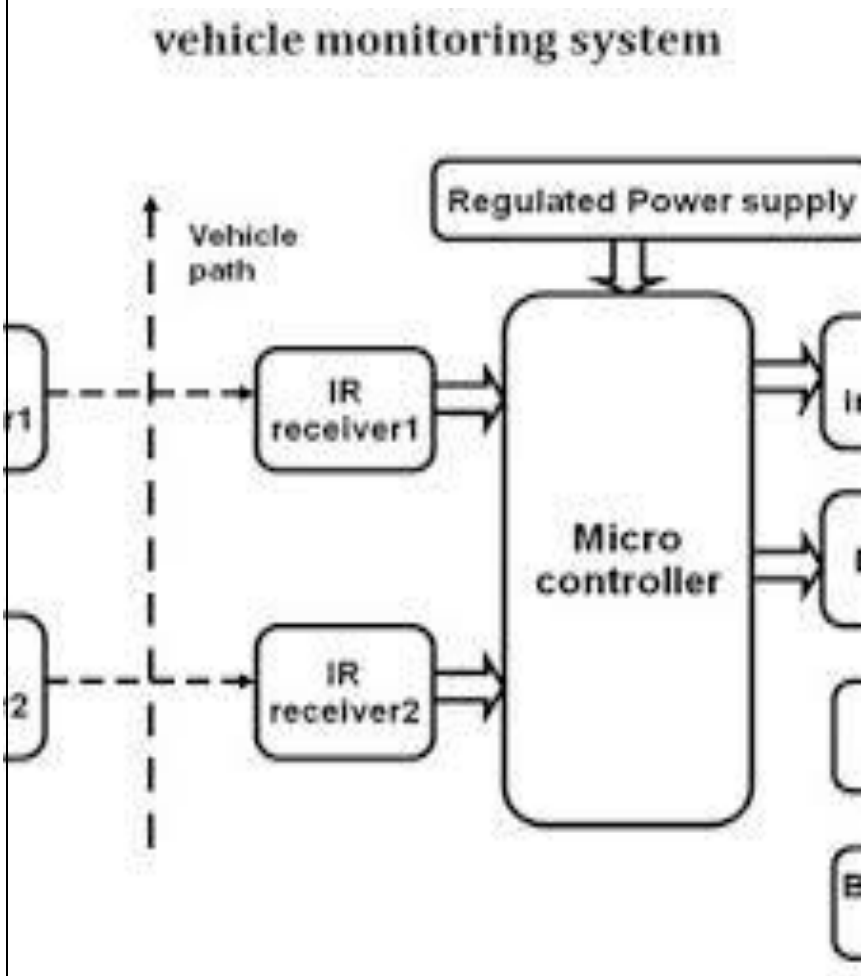
CHAPTER 1: INTRODUCTION

The Vehicle Type Identification System using an 8051 microcontroller is designed to automatically detect and classify different types of vehicles passing through a designated area. This system aims to streamline traffic monitoring and management by providing real-time data on the types of vehicles on the road. By leveraging the capabilities of the 8051 microcontroller, the system offers a cost-effective and efficient solution for traffic analysis.

Chapter 2: Block Diagram

Here is the circuit diagram and working of simple stepper motor control using 8051 microcontroller

Fig.1. block diagram



Chapter 3: Requirements

SlNo	Name of component	quantity
1	8051 microcontroller	1
2	sensors	2
3	display	1
4	IDE, compiler,	1
5	connectors, cables,	1

Chapter 4: Theoretical Analysis

- Detailed explanation of the algorithm used for vehicle type identification.
- Description of the sensors employed for detecting vehicles and their types.
- Explanation of the decision-making process in identifying vehicle types.
- Consideration of factors like speed, size, weight, etc., for classification.Code:
- Detailed explanation of the algorithm used for vehicle type identification.
- Description of the sensors employed for detecting vehicles and their types.
- Explanation of the decision-making process in identifying vehicle types.
- Consideration of factors like speed, size, weight, etc., for classification.

```
; Include 8051 definitions
#include <8051.h>

; Define I/O ports for sensors, LCD, keypad, and buzzer
sbit sensor1 = P1^0;  ; Connect first sensor to P1.0
sbit sensor2 = P1^1;  ; Connect second sensor to P1.1
sbit rs = P2^0;       ; LCD RS pin
sbit rw = P2^1;       ; LCD RW pin
sbit en = P2^2;       ; LCD EN pin
sfr lcd_data = 0x80;  ; LCD data port
sbit keypad = P3^0;   ; Keypad input pin
sbit buzzer = P3^1;   ; Buzzer output pin

; Define delay function
delay_ms PROC
   mov R7, DPL
DELAY_1:
   mov R6, #221
DELAY_2:
   djnz R6, DELAY_2
   djnz R7, DELAY_1
   ret
delay_ms ENDP

; Initialize LCD
lcd_init PROC
   mov A, #38h  ; Function set: 2 lines, 5x8 character size
   call lcd_cmd
   mov A, #0Ch  ; Display control: display on, cursor off, blink off
   call lcd_cmd
   mov A, #06h  ; Entry mode set: increment cursor, no display shift
   call lcd_cmd
   mov A, #01h  ; Clear display
   call lcd_cmd
   ret
lcd_init ENDP

; Send command to LCD
lcd_cmd PROC
   clr rs     ; Select command register
   clr rw     ; Write mode
   setb en    ; Enable LCD
   clr en
   ret
lcd_cmd ENDP
```

```
; Send data to LCD
lcd_data PROC
   setb rs     ; Select data register
   clr rw      ; Write mode
   setb en     ; Enable LCD
   clr en
   ret
lcd_data ENDP

; Write string to LCD
lcd_write PROC
   mov R0, #0
   mov R1, DPL  ; Load address of string
LCD_LOOP:
   mov A, @R1  ; Load character from string
   jz LCD_END  ; If end of string, exit
   call lcd_data  ; Send character to LCD
   inc R1      ; Move to next character
   inc R0      ; Increment counter
   cjne R0, #16, LCD_LOOP  ; Continue until 16 characters displayed
LCD_END:
   ret
lcd_write ENDP

; Main program
MAIN:
   mov P1, #0FFh  ; Set P1 as input
   mov P2, #0FFh  ; Set P2 as output
   mov P3, #0FFh  ; Set P3 as input/output
   acall lcd_init  ; Initialize LCD

   ; Main loop
   LOOP:
      ; Check if sensor 1 is triggered
      jb sensor1, START_TIMER

      ; Check if sensor 2 is triggered
      jb sensor2, STOP_TIMER

      ; If no sensor is triggered, continue looping
      sjmp LOOP

   ; Start timer
   START_TIMER:
      mov TMOD, #01h  ; Timer 0, Mode 1
      mov TH0, #0     ; Initialize timer
      mov TL0, #0
```

```
    setb TR0        ; Start timer
    sjmp LOOP

; Stop timer and calculate time
STOP_TIMER:
    clr TR0         ; Stop timer
    mov A, TL0      ; Get timer value
    mov R1, #0      ; Clear R1
    mov R2, #25     ; Set threshold for person walking

    ; Compare time taken with thresholds
    cjne A, R1, CHECK_BICYCLE
    ; Display result for person walking
    mov DPTR, #PERSON_WALKING_MSG
    acall lcd_write
    sjmp LOOP

CHECK_BICYCLE:
    cjne A, R2, CHECK_BIKE
    ; Display result for bicycle
    mov DPTR, #BICYCLE_MSG
    acall lcd_write
    sjmp LOOP

CHECK_BIKE:
    mov R1, #20     ; Set threshold for bike
    cjne A, R1, CHECK_TRUCK
    ; Display result for bike
    mov DPTR, #BIKE_MSG
    acall lcd_write
    sjmp LOOP

CHECK_TRUCK:
    mov R1, #15     ; Set threshold for truck
    cjne A, R1, CHECK_CAR
    ; Display result for truck
    mov DPTR, #TRUCK_MSG
    acall lcd_write
    sjmp LOOP

CHECK_CAR:
    mov R1, #10     ; Set threshold for car
    cjne A, R1, LOOP
    ; Display result for car
    mov DPTR, #CAR_MSG
    acall lcd_write
    sjmp LOOP
```

```
; Define messages
PERSON_WALKING_MSG: DB "Person Walking", 0
BICYCLE_MSG: DB "Bicycle", 0
BIKE_MSG: DB "Bike", 0
TRUCK_MSG: DB "Truck", 0
CAR_MSG: DB "Car", 0

END
```

Chapter 5: Simulation and Results

- Simulation of the system using appropriate software tools (if applicable).
- Analysis of simulated results to validate the system's functionality.
- Discussion on the accuracy and efficiency of the proposed algorithm.

Chapter 6: Hardware implementation

Sensor Setup:

Install ultrasonic sensors at strategic locations to cover the area where vehicles will pass.

Place infrared sensors at suitable heights and positions to detect different vehicle types.

Connect sensors to the appropriate input pins of the microcontroller.

Calibrate sensors to ensure accurate distance measurement and detection.

Microcontroller Setup:

Connect the 8051 microcontroller to the power supply.

Connect the microcontroller to the display unit for output.

Interface the microcontroller with sensors using GPIO pins.

Write the firmware code to initialize the microcontroller, read sensor data, and implement the vehicle identification algorithm.

Display Unit Integration:

Connect the display unit (LCD or LED) to the microcontroller.

Write code to display the identified vehicle type on the display.

Ensure proper communication between the microcontroller and the display unit.1. Gather Required Components:

Fingerprint scanner module

89S52 microcontroller

LCD display

Keypad

Solenoid or motor for locker mechanism

Power supply

Breadboard or PCB (Printed Circuit Board)

2. Design the Circuit:

Plan the layout of the components on the breadboard or design the PCB layout.

Connect the components according to the block diagram.

Ensure proper wiring and connections between the components.

3. Connect Fingerprint Scanner:

Connect the fingerprint scanner module to the microcontroller.

Interface the fingerprint scanner according to its datasheet or user manual.

Verify the communication between the microcontroller and the fingerprint scanner.

4. Interface LCD Display and Keypad:

Connect the LCD display and keypad to the microcontroller.

Use appropriate interface protocols such as GPIO, SPI, or I2C.

Write code to control the LCD display and read input from the keypad.

5. Control Locker Mechanism:

Connect the solenoid or motor to the microcontroller.

Ensure proper power supply and voltage levels for the locker mechanism.

Write code to control the locker mechanism based on authentication results.

6. Power Supply Setup:

Connect the power supply to all components.

Ensure the power supply meets the voltage and current requirements of the components.

Use proper voltage regulation and filtering to prevent damage to the components.

7. Write Embedded Software:

Develop firmware for the 89S52 microcontroller using Embedded C.

Implement algorithms for fingerprint recognition, authentication, and control logic for locker mechanism.

Test the software modules individually and then integrate them into a complete system.

8. Test and Debug:

Power up the system and test each component and functionality.

Verify fingerprint scanning, LCD display output, keypad input, and locker mechanism control.

Debug any issues encountered during testing and make necessary adjustments to the hardware or software.

9. Final Assembly:

Once all components are working correctly, finalize the assembly of the system.

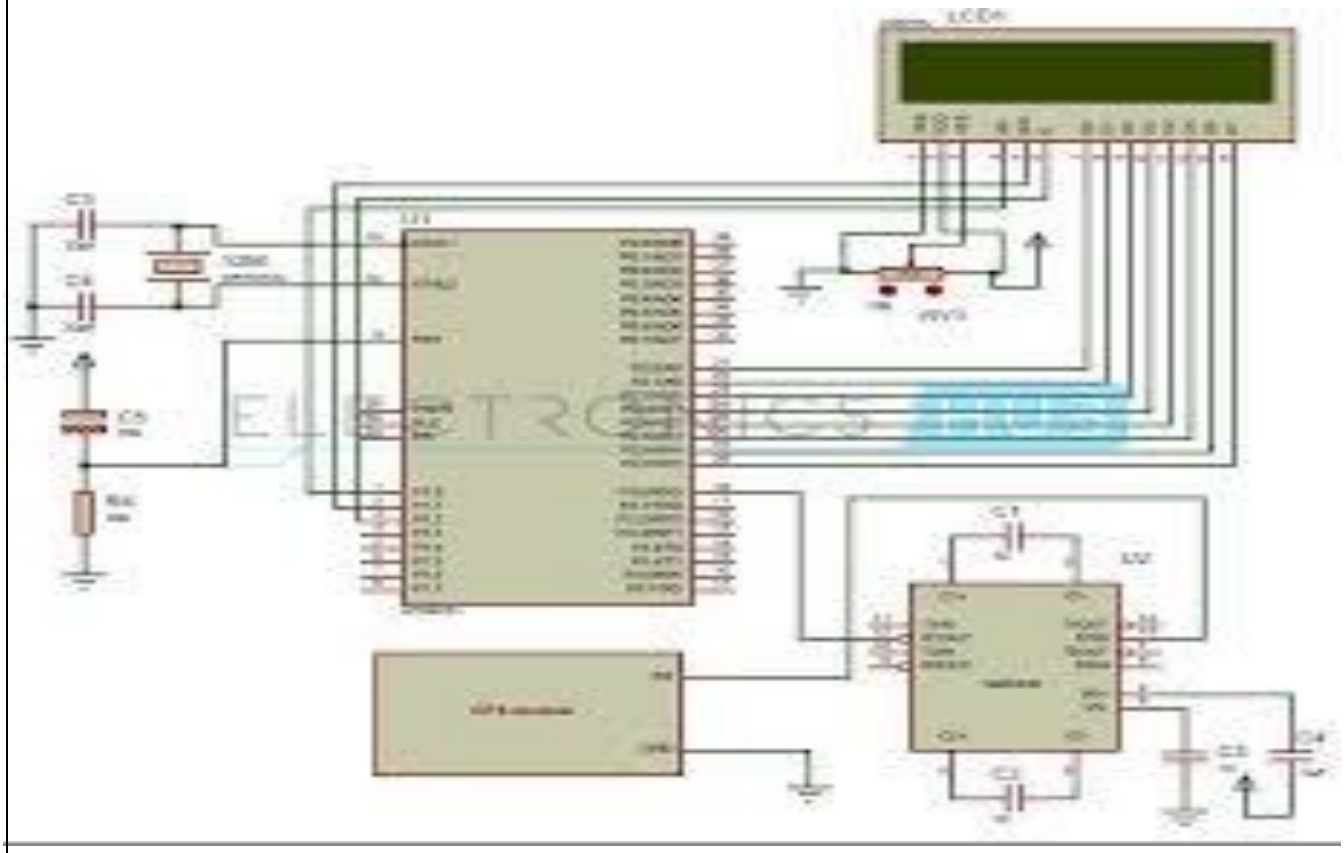
Securely mount the components on a suitable enclosure or platform.

Ensure proper cable management and protection against environmental factors.

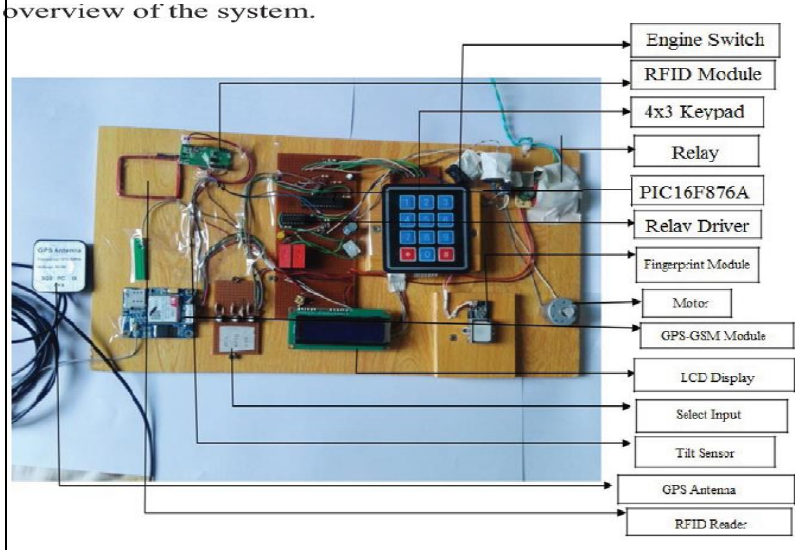
10. System Integration and Deployment:

Integrate the fingerprint bank locker system into the bank's infrastructure.

Train bank staff and users on how to use the system effectively.



overview of the system.



Chapter 7: Conclusion and Future scope

- Summary of the project's objectives and achievements.
- Evaluation of the system's performance against the initial goals.
- Discussion on potential improvements and enhancements.
- Future scope for expanding the system's capabilities or integrating it with other systems.
- Final thoughts on the significance and applicability of the developed vehicle type identification system.