

Password Protected Vehicle Ignition System in Two wheelers

A.Gowrisankar, A V S Pradeep

Abstract—GSM based security system are much more stout than an ordinary security system. The ordinary systems are simply based on the concept of sensors. They sound an alarm on detecting movement. This system of technology has now lost its appeal as it has become a common sighting in metros where these alarms go off unnecessarily. We proposed with Keypad and GSM techniques with a better decision making process are built to make our vehicle more secure. It is a unique wireless two-wheeler security device that gives instant alerts on your mobile phone the moment a security breach is detected. It is designed to alert you wirelessly through a GSM modem. In this system Proposed an 8 bit embedded controller inter model. The control mechanism is based on the number keys are pressed.

Index Terms— AT89s52microcontrollers, Keypad, GSMmodem, Mobilehandset, Buzzer, LCDdisplay, Max232ic, Keilsoftware, prots simulator.

I. INTRODUCTION

The advances in the technologies related to wireless communication has led to the emergence of several engineering designs to aid the human requirements. Thus with the creeping interests in the wireless and GSM based projects, we came up with this idea of developing a simpler, multipurpose, cost-effective design to secure the two wheelers. The facility would require a tool that extends access to devices from his/her premises to anywhere the user might be provide a password by using keypad, if entered password is correct the controller send command to ignition system to start the vehicle and also send command to GSM modem to alert the owner of the vehicle. If an Unauthorized person accesses the vehicle the owner can control the system by sending a message. This is done through a mobile device making Commands are sent to vehicle ignition system. Providing a cost effective, reliable far reaching access to the user. The coded SMS is sent to the ignition system Applications base station controller that receives the messages, decodes the messages, initiates required automation operations and responds to the successful initiations by a reply to the user.

II. INTERFACING

A. LCD Interfacing

A program must interface with the outside world using input and output devices that communicate directly with a human being. One of the most common devices attached to a controller is an LCD display. Some of the most common

LCDs connected to the controllers are 16x1,16x2 and 20x2 displays. This means 16 characters per line by 1 line 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

Pin No	Symbol	Details
1	Vcc	Supply voltage +5v
2	GND	Ground
3	Vo	Contrast adjustment
4	rs	0-> Control input 1-> Data input
5	R/W	Read/Write
6	E	Enable
7 to 14	D0 to D7	Data
15	VB1	Backlight +5v
16	VB0	Backlight ground

TABLE I. LCD pin details

EN: Line is called “Enable.” This control line is used to tell the LCD that you are sending it data. To send data to the LCD, your program should make sure this line is low (0) and then set the other two control lines and/or put data on the data bus. When the other lines are completely ready, bring EN high (1) and wait for the minimum amount of time required by the LCD datasheet (this varies from LCD to LCD), and end by bringing it low (0) again.

RS: Line is the “Register Select” line. When RS is low (0), the data is to be treated as a command or special instruction (such as clear screen, position cursor, etc.). When RS is high (1), the data being sent is text data which should be displayed on the screen. For example, to display the letter “T” on the screen you would set RS high.

RW: Line is the “Read/Write” control line. When RW is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively querying (or reading) the LCD. Only one instruction (“Get LCD status”) is a read command. All others are write commands, so RW will almost always be low.

Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, and DB7.

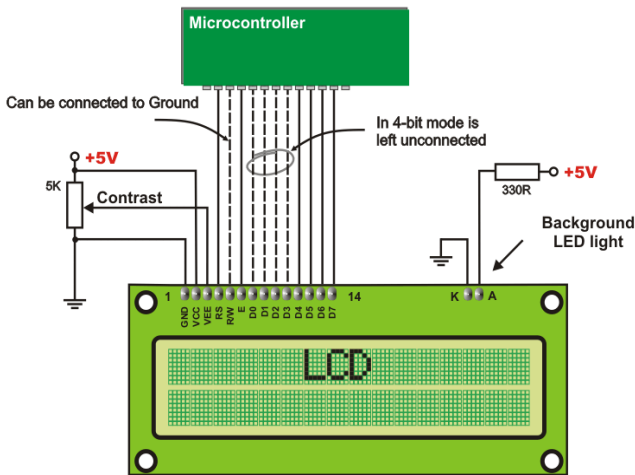


Figure2.1: LCD Interfacing

Logic status on control lines:

- E – 0 Access to LCD disabled
- 1 Access to LCD enabled
- R/W - 0 Writing data to LCD
- 1 Reading data from LCD
- RS - 0 Instructions
-1 Character

B. GSM Modem interfacing

GSM (Global System for Mobile communications) is an open, digital cellular technology used for transmitting mobile voice and data services. GSM (Global System for Mobile communication) is a digital mobile telephone system that is widely used in Europe and other parts of the world. GSM uses a variation of Time Division Multiple Access (TDMA) and is the most widely used of the three digital wireless telephone technologies (TDMA, GSM, and CDMA). GSM digitizes and compresses data. It operates at either the 900 MHz or 1,800 MHz frequency band. It supports voice calls and data transfer speeds of up to 9.6 Kbit/s, together with the transmission of SMS (Short Message Service).

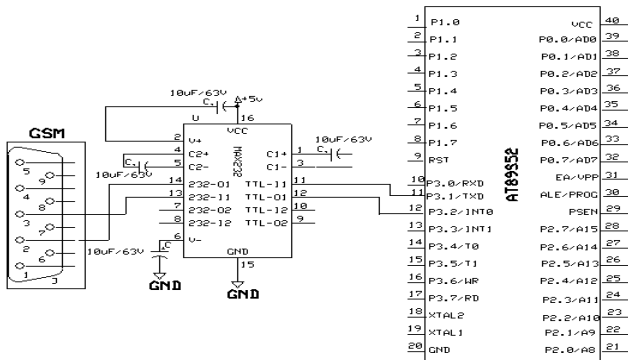


Figure2.2: GSM Interfacing

The controlling device controls the GSM engine by sending AT commands via its serial interface. AT command syntax:

The “AT” or “at” prefix must be set at the beginning of each command line. To terminate a command line enter <CR>. Commands are usually followed by a response that includes. “<CR><LF><response><CR><LF>”.

Throughout this document, only the responses are presented, <CR><LF> are omitted intentionally.

The AT Command	Description of Commands
AT	Check if serial interface and GSM modem is working.
ATE0	Turn echo off, less traffic on serial line.
AT+CNMI	Display of new incoming SMS.
AT+CPMS	Selection of SMS memory.
AT+CMGF	SMS string format, how they are compressed.
AT+CMGR	Read new message from a given memory location.
AT+CMGS	Send message to a given recipient.
AT+CMGD	Delete message.

TABLE II. AT-Command set overview

C. MAX232 ic interfacing

Max 232 is a communications device used mainly for serial commands to and from a flash ROM. The MAX232 is an integrated circuit that converts signals from an RS-232 serial port to signals suitable for use in TTL compatible digital logic circuits. The MAX232 is a dual driver/receiver and typically converts the RX, TX, CTS and RTS signals. The drivers provide RS-232 voltage level outputs (approx. ± 7.5 V) from a single + 5 V supply via on-chip charge pumps and external capacitors. This makes it useful for implementing RS-232 in devices that otherwise do not need any voltages outside the 0 V to + 5 V range, as power supply design does not need to be made more complicated just for driving the RS-232 in this case. The receivers reduce RS-232 inputs (which may be as high as ± 25 V), to standard 5 V TTL levels. These receivers have a typical threshold of 1.3 V, and a typical hysteresis of 0.5v, The later MAX232A is backwards compatible with the original MAX232 but may operate at higher baud rates and can use smaller external capacitors – 0.1 µF in place of the 1.0 µF capacitors used with the original device. The newer MAX3232 is also backwards compatible, but operates at a broader voltage range, from 3 to 5.5V.

Voltage levels:

It is helpful to understand what occurs to the voltage levels. When a MAX232 IC receives a TTL level to convert, it changes a TTL Logic 0 to between +3 and +25V, and changes TTL Logic 1 to between -3 to -25V, and vice versa for converting from RS232 to TTL. This can be confusing when you realize that the RS232 Data Transmission voltages at a certain logic state are opposite from the RS232 Control Line voltages at the same logic state. To clarify the matter, see the table below. For more information see Levels. Standard serial interfacing of microcontroller (TTL) with PC or any RS232C Standard device , requires TTL to RS232 Level converter . A MAX232 is used for this purpose. It provides 2-channel RS232C port and requires external 10uF capacitors. The driver requires a single supply of +5V.

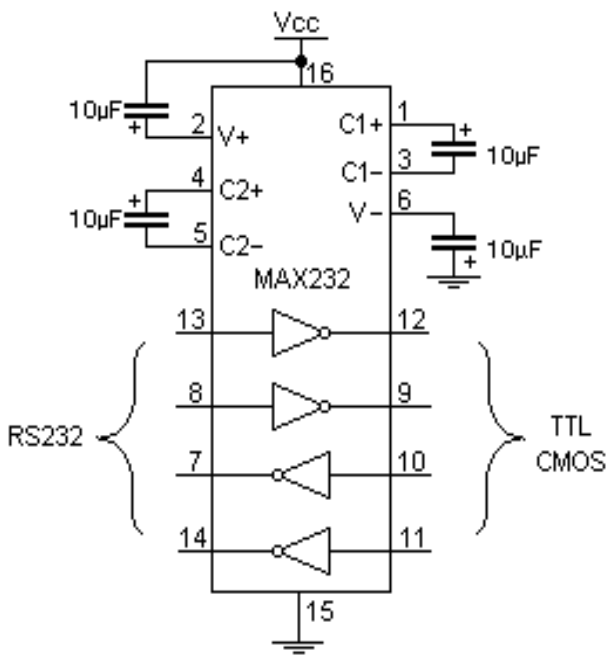


Figure2.3:MAX232 ic Internal Diagram

D. Relay interfacing

Relay is an electromagnetic switch. Relay is used to drive high voltage loads depending on the switching logic of a microcontroller (or) any digital device. Relay coil is connected to the collector of a transistor through 12V Vcc. emitter of the transistor is grounded. a biasing resistor is connected to the base of the transistor. This value can be varies between 220Ω to 1.5kΩ.

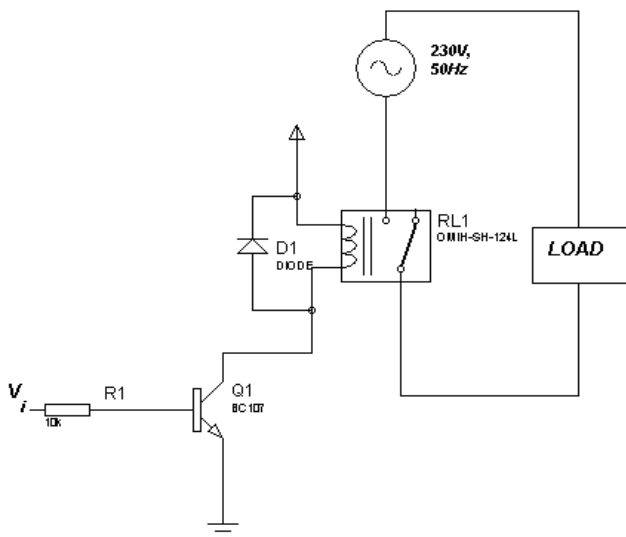


Figure2.4: Relay Interfacing

Inside the relay there are three terminals namely common, normally connected, normally open. By default the common terminal is connected to the normally connected terminal with a spring tension. Whenever we apply logic to the base of the transistor, the transistor acts as closed switch and makes the coil energized in to an electromagnet. Which in turn attracts the small iron strip of the common terminal, which makes a connection between common and normally opened terminal? A load can be connected between these two terminals and a source as shown in the figure 2.4

E. Power Supply Unit:

As the microcontroller operating voltage is +5V DC. Through this power supply circuit we have to create a +5V DC which is given to the micro controller. The below components are used to create the power supply.

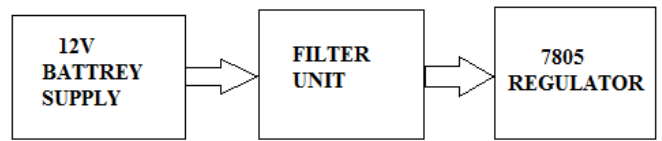


Figure2.5: Block diagram of power supply

Description:

12v dc power supply is given to filter unit That output DC Voltage is given to the positive voltage regulator LM7805 (i.e., 78 represents the positive series and 5 represent the output voltage it can provide). So the output of the regulator will be the regulated +5V DC. To indicate the condition of the circuit we place a LED at the end of the circuit.

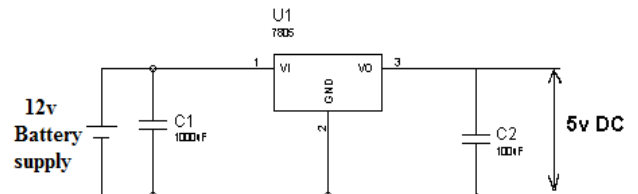


Figure2.6: power supply unit circuit diagram

F. keypad interfacing



Figure2.7: Keypad Interfacing

Keypad Properties:

The keypad has a 3-columns and 4-rows matrix orientation as seen in figure 2. If a key is pushed then the circuit would be shorted for those specific key pins. The short circuit would always be between a row pin and a column pin. For this specific keypad the rows 1-4 are represented by pins 1-4 and columns 1-3 are represented by pins 5-7. For example button 1 would be represented by pins 1 and 5, so if a voltage is applied to one of the pins and a voltmeter is connected to the other pin, when the button is pushed the voltmeter would read the input voltage. It would be easy to construct a table that would show each button represents what character,

and which pins are shorted if that button is pressed, which is provided in Table III

Key Number	Character	Column Pin	Row Pin
1	'1'	5	1
2	'2'	6	1
3	'3'	7	1
4	'4'	5	2
5	'5'	6	2
6	'6'	7	2
7	'7'	5	3
8	'8'	6	3
9	'9'	7	3
10	'*'	5	4
11	'0'	6	4
12	'#'	7	4

TABLE III.

5. Keypad Communication with the Microcontroller:

In this project, the used microcontroller is AT89s52, so some other microcontrollers might have some different requirements or setting in the code, however the method is set to be general so it could be used in any microcontroller that have the appropriate pins and settings. Also, some important parameter should be set in advance to find which key is pushed. First the microcontroller pins that are connected to the rows should be set to be high output pin, and the column pins should be set to be input pins. Then in order to detect if a pin is pushed the microcontroller would set one row output to be low and then try to detect if it receives a low input from the column of the keypad. Then, it would set that row back to high and set another row to low, that would happen until it detects the low signal and convert that signal into a key character. This loop would happen in a matter of microseconds so if a button is pushed it would detect it easily.

For example: if button '8' is pushed the microcontroller would set the first row to low, then detect if an input goes low, which none would because '8' is connected to row 3 which is high and then the shorted output between column 2 and row 3 would still be high. Then the microcontroller will reset row 1 to high and set row 2 to low, which would not change any input. Then, the microcontroller would reset row 2 to high and set row 3 to low, and try to detect if one of the columns inputs is low, which the 2nd column would be low, because when '8' is pressed column 2 would be shorted with row 3 and since row 3 is low, and column 2 is connected to it and to a pull up resistor, the resistor would pull the voltage away from the pin of column 2 which makes the microcontroller detects a low input in column 2 when row 3 is set to low, then it would now that '8' is pressed.

G. Buzzer interfacing

What is a Buzzer?

A buzzer is a mechanical, electromechanical, magnetic, electromagnetic, electro-acoustic or piezoelectric

audio signaling device. A piezo electric buzzer can be driven by an oscillating electronic circuit or other audio signal source. A click, beep or ring can indicate that a button has been pressed.

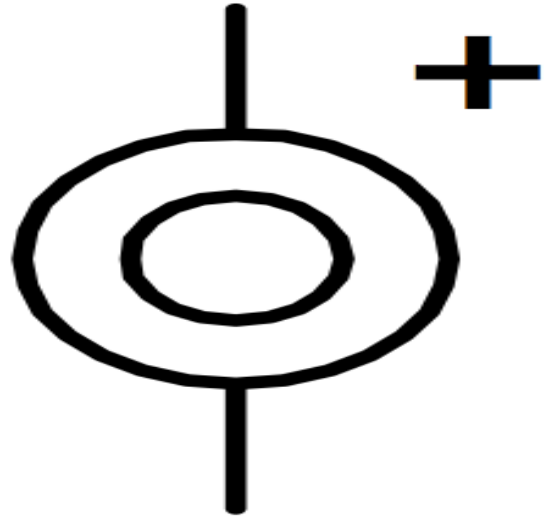


Figure2.8: Buzzer symbol

There are several different kinds of buzzers. At Future Electronics we stock many of the most common types categorized by Type, Sound Level, Frequency, Rated Voltage, Dimension and Packaging Type. The parametric filters on our website can help refine your search results depending on the required specifications. The most common sizes for Sound Level are 80 dB, 85 dB, 90 dB and 95 dB. We also carry buzzers with Sound Level up to 105 dB. There are several types available including Electro-Acoustic, Electromagnetic, and Electro mechanic, Magnetic and Piezo, among others.



Figure2.8: Buzzer module

III. CIRCUIT DESCRIPTION

Port3 (pins 10 & 11) of Microcontroller 8051 is used for the purpose of serial communication (transmitter & receiver). In port3 (pin 10) can be used as input port by writing 1's to the

BC547 at Port P3.6. the transistor will be ON condition and coil get energized and ignition is activated. Microcontroller displays ACCESS GRANTED IGNITE YOUR VEHICLE string on LCD screen and sends acknowledgement message to our mobile handset as YOUR VEHICLE IS ACCESSED. if unknown person accessed you vehicle simply send OFF to controller. In controller circuit the GSM modem receives a message and transmits the information to Micro controller through MAX 232. When the Micro controller receives the information then it sends LOGIC 1 to base of the transistor BC547 at Port P3.7. the transistor will be ON condition and Buzzer is activated. and microcontroller sends LOGIC 0 to the base of the transistor at P3.6 and it will be OFF condition. The relay coil get de-energized ignition system is deactivated. The microcontroller displays “YOUR VEHICLE IS IN OFF MODE” on LCD screen and sends acknowledgement message to our mobile handset as YOUR VEHICLE IS IN OFF MODE & BUZZER IS ACTIVATED. If entered password is wrong then the Microcontroller receives the information then it sends LOGIC 1 to base of the transistor BC547 at Port P3.7. the transistor will be ON condition and Buzzer is activated. and micro controller sends LOGIC 0 to the base of the transistor at P3.6 and it will be OFF condition. The microcontroller displays “ACCESS DENIED WRONG PASSWORD” on LCD screen and sends acknowledgement message to our mobile handset as UNAUTHORIZED PERSON ACCESSED YOUR VEHICLE BUZZER IS IN ON.

VI. CONCLUSION

We have successfully implemented password protected Vehicle Ignition system in two wheelers. In this the microcontroller collects the data from the keypad and verifying password. if the password correct Vehicle Ignition System is activated. its alerts owner by sending Sms to their handset This system is fully controlled by the microcontroller depending on the information received from the GSM modem. For the displaying purpose we are using LCD.

REFERENCES

- [1] Muhammad Ali Mazzidi “The 8051 Microcontroller and Embedded systems”2006.
- [2] RAJ KAMAL “EMBEDDED SYSTEMS”
- [3] The Keil u Vision4 Integrated Development Environment (IDE), www.keil.com/uv4
- [4] <http://www.engineersgarage.com/electroniccomponents/at89s52-microcontroller>
- [5] <http://datasheetreference.com/atmel-89s52datasheet.html>
- [6] KennethJ.Ayala”The8051MicrocontrollerArchitecture, Programming& Applications “
- [7] SIMCON SIM300 command set hardware interface description
- [8] Proteus 8 Professional simulator.
- [9] N. Kaushik, M. Veralkar, P. Parab, and K. Nadkarny, “Anti-Theft vehicle security system,” *International Journal for ScientificResearch and Development*, vol. 1, no. 12, pp. 2845-2848, March 2014.
- [10] R.K. Jugren, “Automotive Electronics Handbook.” 2nd Ed. McGrawHill,NewYork,1999.
- [11] Kenneth J. Ayala, “The 8051 micro-controller Architecture, programming & applications”, Penram International, India, 1996



A V S PRADEEP, Btech,Mtech, Asst.professor
Department of ECE, Miracle educational society group of institutions



A.Gowrisankar is pursuing his master of Technical in Embedded systems in Miracle education society group of institutions,vizianagaram. He has completed his B.Tech in Electronics and communication engineering from prasiddha college of engineering and technology.His area of interest in research in Embeddedsystems and robotics