



**VEHICLE THEFT INTIMATION
AND READING VEHICLE'S
INFORMATION USING RFID**



A PROJECT REPORT

Submitted by

MOHANDOSS V S

Reg. No.:1110107054

NAGARAJAN D

Reg. No.:1110107058

SHEIK MOHAMMED M

Reg. No.:1110107323

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE-641049

APRIL 2015

(An Autonomous Institution Affiliated to Anna University, Chennai)



KUMARAGURU COLLEGE OF TECHNOLOGY



COIMBATORE-641049

(An Autonomous Institution Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report titled **“Vehicle theft intimation and reading vehicle’s information using RFID”** is the bonafied work of **“MOHANDOSS V S, NAGARAJAN D, SHEIK MOHAMMED M”** who carried out the project work under my supervision.

SIGNATURE

Ms. A. Amsaveni M.E,(Ph.D),
Ph.D.,
Associate Professor,
Department of ECE,
Kumaraguru College of Technology,
Coimbatore.

SIGNATURE

Dr. Rajeswari Mariappan M.E.,
Head Of Department,
Department of ECE,
Kumaraguru College of Technology,
Coimbatore.

The candidates with Register numbers 1110107054, 1110107058 and 1110107323 are examined by us in the project viva-voce examination held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

First we would like to express our praise and gratitude to the Lord, who has showered his grace and blessing enabling us to complete this project in an excellent manner. He has made all things in beautiful in his time.

We express our sincere thanks to our beloved Joint Correspondent, **Shri. Shankar Vanavarayar** for his kind support and for providing necessary facilities to carry out the project work.

We would like to express our sincere thanks to our beloved Principal **Dr.R.S.Kumar M.E., Ph.D.**, who encouraged us with his valuable thoughts.

We would like to express our sincere thanks and deep sense of gratitude to our HOD, **Dr. Rajeswari Mariappan M.E., Ph.D.**, for her valuable suggestions and encouragement which paved way for the successful completion of the project.

We are greatly privileged to express our deep sense of gratitude to the Project Coordinator **Ms.S.Nagarathinam M.E., (Ph.D)**, Assistant Professor (SRG), for her continuous support throughout the course.

In particular, We wish to thank and express our everlasting gratitude to the Supervisor **Ms.A.Amsaveni M.E., (Ph.D)**, Associate Professor for her expert counselling in each and every steps of project work and we wish to convey our deep sense of gratitude to all teaching and non-teaching staff members of ECE Department for their help and cooperation.

Finally, we thank our parents and our family members for giving us the moral support in all of our activities and our dear friends who helped us to endure our difficult times with their unflinching support and warm wishes.

ABSTRACT

The main scope of this project is to stop the engine of the vehicle automatically when a person trying to steal the vehicle. The person must need License and Vehicle Key to start the vehicle. The person needs to show the RFID license to start the vehicle. If RFID information doesn't match the owner's information, GSM Modem gives a missed call to the owner. Then GSM Modem sends the alert message to the owner. If the owner sends the message as 'START', then the engine of the vehicle will be turned on. If the owner sends the message as 'STOP', the engine will be turned off. Toggle switch is used to change the mode whether it is vehicle section or traffic section. Vehicle information will be read from the Number Plate of the vehicle and shown in the LCD display using RFID. The vehicle's information are like owner's name, date of birth, engine number, vehicle model and insurance expiry date.

TABLE OF CONTENTS

CHAPTER NO NO	TITLE	PAGE
	ABSTRACT	iv
	LIST OF ABBREVIATIONS	vii
1	INTRODUCTION	1
2	HARDWARE DESCRIPTION	4
	2.1 Peripheral interface controller	5
	2.1.1 Introduction	5
	2.1.2 Pin diagram	6
	2.1.3 Architecture of PIC16F877A	7
	2.2 Relay driver	20
	2.3 LCD Interfacing	25
	2.4 Power supply	27
	2.5 GSM modem	29
	2.6 RFID	32
	2.7 Block diagram	34
	2.8 Circuit diagram	35
	2.9 Working Principle	36

3	SOFTWARE DESCRIPTION	37
	3.1 Embedded C	38
	3.1.1 Introduction	38
	3.2 MP LAB IDE	41
	3.3 Proteus software	42
	3.4 PIC Boot loader	45
	3.4.1 Introduction	45
	3.5 Algorithm	46
4	CONCLUSION	47
	REFERENCE	49

LIST OF ABBREVIATIONS

PIC	Peripheral interface controller
RISC	Reduced Instruction Set Computer
EEPROM	Electrically Erasable Programmable Read Only Memory
SRAM	Static Random Access Memory
JTAG	Joint Test Access Group
LED	Light Emitting Diode
LCD	Liquid Crystal Display
VB	Visual Basic
IDE	Integrated Development Environment
API	Application Programmable Interface
GUI	Graphical User Interface
COM	Component Object Model
ICS	In Circuit Simulators
JVM	Java Virtual Machine

AVR	Advanced Virtual RISC
DTE	Data Terminal Equipment
DCE	Data Circuit-terminating Equipment

CHAPTER 1

INTRODUCTION

Introduction

Currently most of the public having an own vehicle, theft is happening on parking and sometimes while in driving insecurity places. The safety of vehicles is extremely essential for public. Hence Vehicle locking system is estimated to prevent vehicle theft.

Basic idea of our project is to provide GSM based security even if the owner is away from the restricted areas. For this wireless mode of transmission using GSM is adopted. Beside this there are many methods of wireless communication but GSM is selected because as compared to other techniques, this is an efficient and cheap solution also, GSM is much familiar technology and it is easily available.

This project is designed to provide ubiquitous access to the system for the security using extensive GSM technology for communication purposes and microcontroller for device control. The highlights of the proposed system are the long range of communication and security. The security is provided by sending a message to access number, controlling and acknowledgement is done through microcontroller.

This system consists of a GSM modem for sending and receiving the SMS, PIC 16F877A microcontroller which is controlling the entire system, LCD for the display purpose and supporting circuitries like rectifier for ac to dc conversion. In this system, the authorized person will have the entire control and will decide which people will be allowed to start the vehicle.

Radio frequency identification (RFID) is a general term that is used to describe a system that transmits the identity (in the form of a unique serial number) of an object wirelessly, using radio waves. RFID technologies are grouped under the more generic Automatic Identification (Auto ID) technologies.

Each person must have their own RFID license. The person needs to show the RFID license to start the vehicle which is mandatory, this will help to avoid vehicle driving without having license.

Additional feature in this project is reading the vehicle's information using RFID. This will give complete information about the vehicle like owner's name, owner's date of birth, vehicle insurance expiry date, vehicle name and engine number and etc. This will help the traffic police to check the details of the vehicle.

CHAPTER 2

HARDWARE DESCRIPTION

2.1 PERIPHERAL INTERFACE CONTROLLER

2.1.1 Introduction

The Peripheral Interface Controller (PIC) contains an ALU, which does arithmetic and logic operations, the RAM, the EEPROM (Flash Memory), the data EEPROM, and the “W” register. The “W” register is not a part of the register-file but is a stand-alone, working register (also called an “accumulator”). The ALU, the RAM, the “W” register, and the data EEPROM each manipulate and hold 8-bit-wide data, which ranges in value from zero to 255 (or, in hexadecimal, from 0x00 to 0xFF).

The program EEPROM (Flash Memory) works with 14-bit-wide words and contains each of the user’s instructions. It is not uncommon for microcontrollers to have different sizes of data memory and program memory (in the PIC: 8-bits for data and 14-bits for program words). More than that, the key is that the data and program memories occupy separate spaces. This allows access to each at the same time.

This document contains device specific information about the following devices:

- PIC16F874A
- PIC16F876A
- PIC16F877A

PIC16F873A/876A devices are available only in 28-pin packages, while PIC16F874A/877A devices are available in 40-pin and 44-pin packages. All

devices in the PIC16F87XA family share common architecture with the following differences:

- The PIC16F873A and PIC16F874A have one-half of the total on-chip memory of the PIC16F876A and PIC16F877A
- The 28-pin devices have three I/O ports, while the 40/44-pin devices have five I/O ports.
- The 28-pin devices have fourteen interrupts, while the 40/44-pin devices have fifteen interrupts.
- The 28-pin devices have five A/D input channels, while the 40/44-pin devices have eight A/D input channels.
- The Parallel Slave Port is implemented only on the 40/44-pin devices.

2.1.2 Pin Diagram

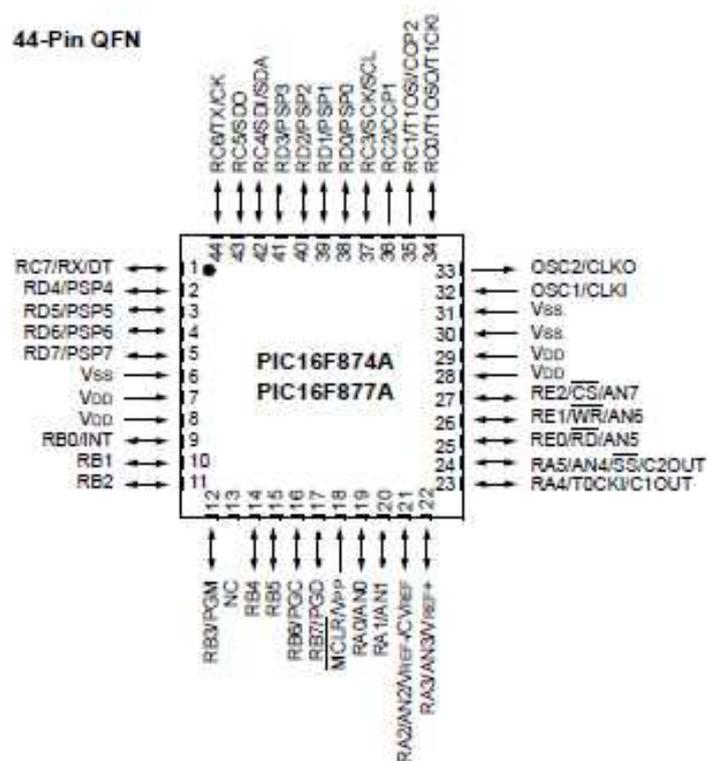


Fig 2.1.2 Pin Diagram for PIC 16F877A

2.1.3 Architecture PIC16F877A

The PIC(Peripheral Interfacing Controller) microcontrollers has been introduced by the microchip technology. while PIC16F874A/877A devices are available in 40-pin and 44-pin packages.

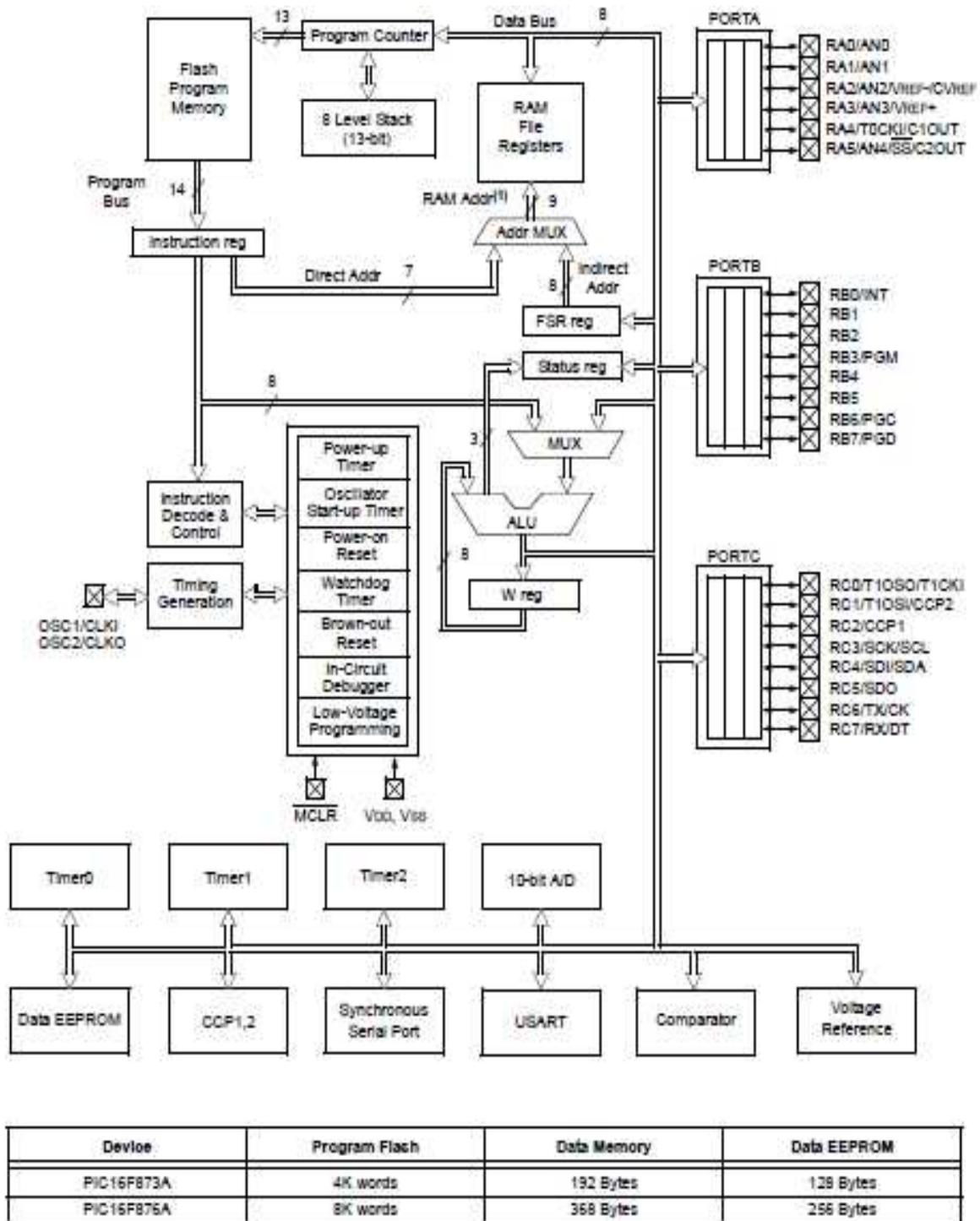


Fig 2.1.3 Architecture of PIC 16F877A

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then the user must follow a specific write sequence to initiate the write for each byte. The

write will not initiate if the write sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte.

We strongly recommend that interrupts be disabled during this code segment . Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set. At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

The steps to write to EEPROM data memory are:

1. If step 10 is not implemented, check the WR bit to see if a write is in progress.
2. Write the address to EEADR. Make sure that the address is not larger than the memory size of the device.
3. Write the 8-bit data value to be programmed in the EEDATA register.
4. Clear the EEPGD bit to point to EEPROM data memory.
5. Set the WREN bit to enable program operations.
6. Disable interrupts (if enabled).
7. Execute the special five instruction sequence:
8. Write 55h to EECON2 in two steps (first to W, then to EECON2)
9. Write AAh to EECON2 in two steps (first to W, then to EECON2)Set the WR bit
- 10.Enable interrupts (if using interrupts).

11. Clear the WREN bit to disable program operations.
12. At the completion of the write cycle, the WR bit is cleared and the EEIF interrupt flag bit is set. (EEIF must be cleared by firmware.) If step 1 is not implemented, then firmware should check for EEIF to be set, or WR to clear, to indicate the end of the program cycle.

```

BSF    STATUS,RP1    ;
BSF    STATUS,RP0
BTFS   EECON1,WR     ;Wait for write
GOTO   $-1           ;to complete
BCF    STATUS,RP0    ;Bank 2
MOVF   DATA_KE_ADDR,W ;Data Memory
MOVWF  EEADR         ;Address to write
MOVF   DATA_KE_DATA,W ;Data Memory Value
MOVWF  EECDATA       ;to write
BSF    STATUS,RP0    ;Bank 3
BCF    EECON1,EEPGD ;Point to DATA
                           ;memory
BSF    EECON1,WREN   ;Enable writes

BSF    INTCON,GIE    ;Disable INTs.
MOVLW  55h           ;
MOVWF  EECON2        ;Write 55h
MOVLW  AAh           ;
MOVWF  EECON2        ;Write AAh
BSF    EECON1,WR     ;Set WR bit to
                           ;begin write
BSF    INTCON,GIE    ;Enable INTs.
BCF    EECON1,WREN   ;Disable writes

```

Fig 2.1.3.1 Coding to Write Data in EEPROM

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKI OSC1 CLKI	13	14	30	32	I I	ST/CMOS ⁽⁴⁾	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS. External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins).
OSC2/CLKO OSC2 CLKO	14	15	31	33	O O	—	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
MCLR/VPP MCLR VPP	1	2	18	18	I P	ST	Master Clear (input) or programming voltage (output). Master Clear (Reset) input. This pin is an active low Reset to the device. Programming voltage input.
RA0/AN0 RA0 AN0 RA1/AN1 RA1 AN1 RA2/AN2/VREF-/CVREF RA2 AN2 VREF- CVREF RA3/AN3/VREF+ RA3 AN3 VREF+ RA4/T0CKI/C1OUT RA4 T0CKI C1OUT RA5/AN4/SS/C2OUT RA5 AN4 SS C2OUT	2	3	19	19	I/O I I/O I I/O I I O I/O I I I O I/O I I I O	TTL TTL TTL TTL ST TTL	PORTA is a bidirectional I/O port. Digital I/O. Analog input 0. Digital I/O. Analog input 1. Digital I/O. Analog input 2. A/D reference voltage (Low) input. Comparator VREF output. Digital I/O. Analog input 3. A/D reference voltage (High) input. Digital I/O – Open-drain when configured as output. Timer0 external clock input. Comparator 1 output. Digital I/O. Analog input 4. SPI slave select input. Comparator 2 output.

Legend: I = Input O = output I/O = Input/output P = power
 — = Not used TTL = TTL Input ST = Schmitt Trigger Input

Table 2.3.1.2 PIC16F877A Pin out Descriptions

2.1.4 Memory Organization

A microcontroller is a complete computer system on a single chip. It is more than just a microprocessor: It also contains a Read-Only Memory (ROM), a Read-Write Memory (RAM), some input/output ports, and some peripherals, such as, counters/timers, analog-to-digital converters, digital-to-analog converters, and serial communication ports.

The internal view of a typical microprocessor and is composed of three things: an arithmetic/logic unit (ALU) which performs calculations on data. A set of registers which hold the user's data and the system's data; and a control unit which orchestrates everything and interprets and executes the user's instructions. As far as the microprocessor is concerned, it assumes that there are sets of data memories and program memories (RAM and ROM) in the system.

The only thing the microprocessor has to do is run a cycle of getting new instructions and executing them from the memories. Both the RAM and the ROM are organized as indexed sets of data words, where each "index" is the "address" of its corresponding data. Both the data and its address codes are numbers represented in binary or hexadecimal.

The RAM is a read-write memory which can rapidly read and write the data. It is a volatile memory which means that it loses its memory when power is removed (turned off). The ROM is for program memory and is "read-only" except in modern variants, such as Electrically Erasable Programmable Read Only Memory (EEPROM) and Flash Memory, which allow data words to be written as well as read.

The writing of an EEPROM is not the same as a RAM since the data-writing time of the EEPROM is about ten thousand times as long as the data-writing time of the RAM. The ROM and its variants are non-volatile memories that preserve their memories when the power is removed (turned off).

2.1.5 Special Features

- Timer0: 8-bit timer/counter with 8-bit pre scalar
- Timer1: 16-bit timer/counter with pre scalar, can be incremented during Sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, pre scalar and post scalar
- Two Capture, Compare, PWM modules

Capture is 16-bit, max. Resolution is 12.5 ns

Compare is 16-bit, max. Resolution is 200 ns

PWM max. Resolution is 10-bit

- Synchronous Serial Port (SSP) with SPI™ (Master mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) – 8 bits wide with external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

Key Features	PIC16F877A
Operating Frequency	DC – 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)
Flash Program Memory (14-bit words)	8K
Data Memory (bytes)	368
EEPROM Data Memory (bytes)	256
Interrupts	15
I/O Ports	Ports A, B, C, D, E
Timers	3
Capture/Compare/PWM modules	2
Serial Communications	MSSP, USART
Parallel Communications	PSP
10-bit Analog-to-Digital Module	8 input channels
Analog Comparators	2
Instruction Set	35 Instructions
Packages	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN

Table 2.1.5 PIC 16F877A Features

2.1.5.1 High-Performance RISC CPU:

- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches, which are two-cycle
- Operating speed: DC – 20 MHz clock input DC – 200 ns instruction cycle
- Up to 8K x 14 words of Flash Program Memory, Up to 368 x 8 bytes of Data Memory (RAM), Up to 256 x 8 bytes of EEPROM Data Memory
- Pin out compatible to other 28-pin or 40/44-pin PIC16F877A.

2.1.6 SPECIAL REGISTERS

2.1.6.1 Status Register

The Status register contains the arithmetic status of the ALU, the Reset status and the bank select bits for data memory. The Status register can be the destination for any instruction, as with any other register. If the Status register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the TO and PD bits are not writable, therefore, the result of an instruction with the Status register as destination may be different than intended. For example, CLRFSSTATUS will clear the upper three bits and set the Z bit. This leaves the Status register as 000u u1uu (where u = unchanged). It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the Status register because these instructions do not affect the Z, C or DC bits from the Status register.

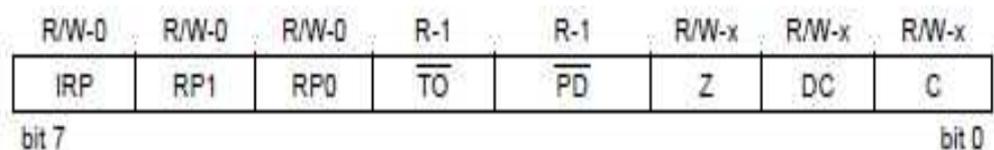


Fig 4.5 Status Register (ADDRESS 03h, 83h, 103h, 183h)

Bit 7 **IRP**: Register Bank Select bit (used for indirect addressing)

1 = Bank 2, 3 (100h-1FFh)

0 = Bank 0, 1 (00h-FFh)

Bit 6-5 **RP1:RP0**: Register Bank Select bits (used for direct addressing)

11 = Bank 3 (180h-1FFh)

10 = Bank 2 (100h-17Fh)

01 = Bank 1 (80h-FFh)

00 = Bank 0 (00h-7Fh)

Each bank is 128 bytes.

Bit 4 **TO**: Time-out bit

1 = after power-up, CLRWDT instruction or SLEEP instruction

0 = A WDT time-out occurred

Bit 3 **PD**: Power-down bit

1 = after power-up or by the CLRWDT instruction

0 = by execution of the SLEEP instruction

Bit 2 **Z**: Zero bit

1 = the result of an arithmetic or logic operation is zero

0 = the result of an arithmetic or logic operation is not zero

Bit 1 **DC**: Digit carry/borrow bit

1 = A carry-out from the 4th low order bit of the result occurred

0 = No carry-out from the 4th low order bit of the result

Bit 0 **C**: Carry/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)

1 = A carry-out from the Most Significant bit of the result occurred

0 = No carry-out from the Most Significant bit of the result occurred

2.1.7 I²C BUS (Inter Integrated Circuit)

The I²C module provides an interface between the TCI648x/C6472 device and other devices compliant with the I²C-bus specification and connected by way of an I²C-bus. External components attached to this 2-wire serial bus can transmit and receive up to 8-bit wide data to and from the device through the I²C module.

The I²C allows connection of up to 128 individually addressable devices using only two bi-directional lines: clock (SCL) and data (SDA). The only additional hardware required is a pull-up resistor for each of the lines. Each of the connected devices can be either a master or slave device. Only master devices are allowed to drive the clock line. The I²C protocol and the EFM32 I2C module feature several mechanisms for handling bus conflicts and contention

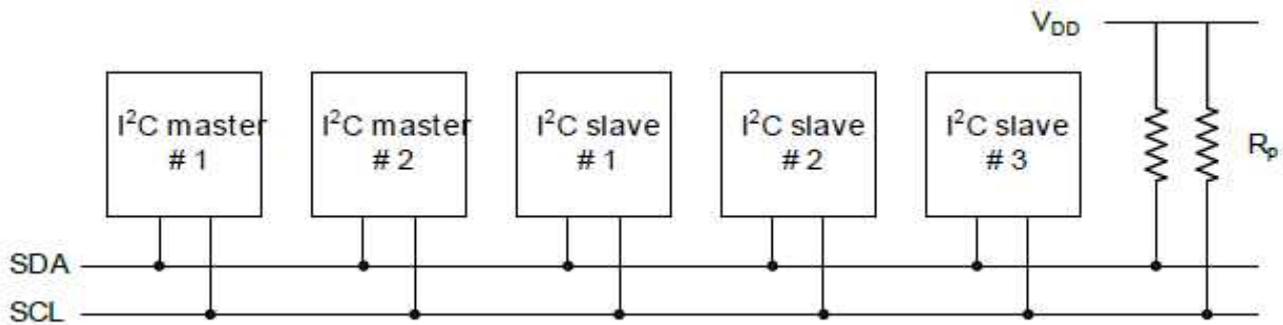


Fig 4.6 I²C Connection Scheme

At the physical layer both SCL and SCA lines are in open-drain, hence the pull-up resistors. Increasing the number of devices on the I2C bus will also increase the line capacitance and thus reduce the slew rate. The slew-rate can be controlled by changing the drive strength in the GPIO module for the I2C pins. The size of the pull-up resistors can be calculated as a function of the maximum rise time allowed for the given bus speed and the estimated bus capacitance.

2.1.7.1 Functional Block diagram

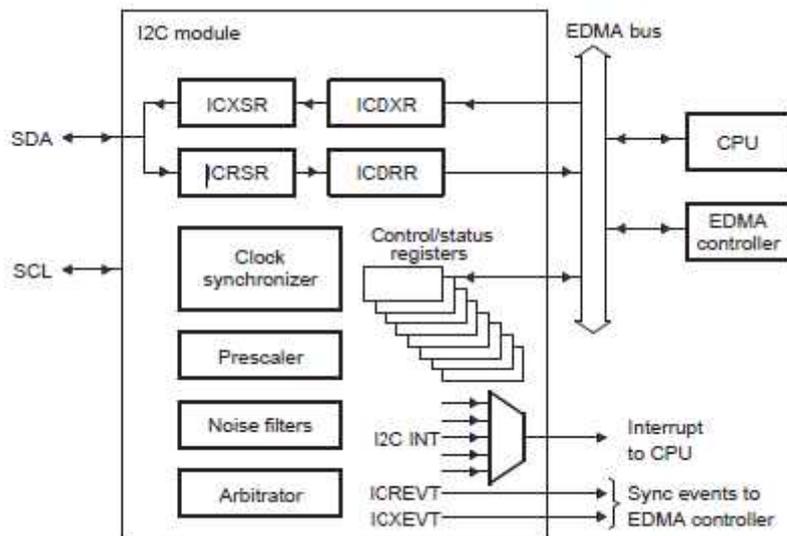


Fig 4.7.1 Block diagram of I²C

2.2 RELAY DRIVER

A relay is a switch worked by an electromagnet. It is useful if we want a small current in one circuit to control another circuit containing a device such as a lamp or electric motor which requires a large current, or if we wish several different switch contacts to be operated simultaneously.

When the controlling current flows through the coil, the soft iron core is magnetized and attracts the L-shaped soft iron armature. This rocks on its pivot and opens, closes or changes over, the electrical contacts in the circuit being controlled it closes the contacts.

The current needed to operate a relay is called the pull-in current and the dropout current in the coil when the relay just stops working. Relay driver circuit is used for on / off control of relay, it acts as a switch, normally open relay is used. Relay function is derived by controller unit. Relay working current is 40 ma.

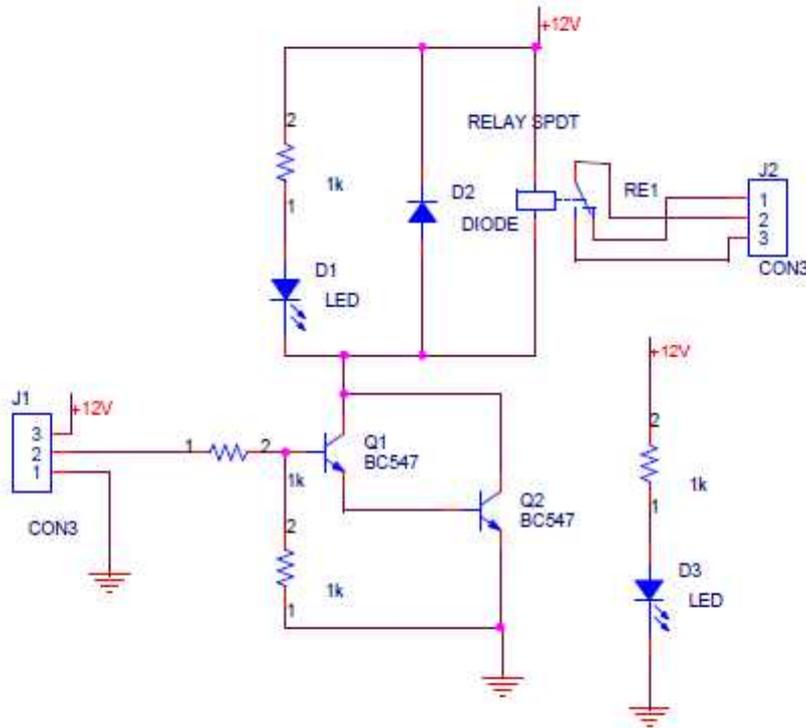
When the controller unit output is low relay should be in off condition. When the controller unit output is high, relay should be ready for doing specified function.

For example a low voltage battery circuit can use a relay to switch a 230V AC mains circuit. There is no electrical connection inside the relay between the two circuits; the link is magnetic and mechanical.

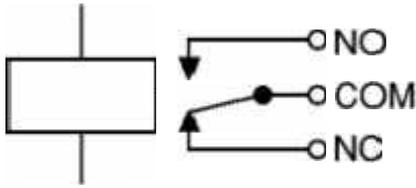
The coil of a relay passes a relatively large current, typically 30mA for a 12V relay, but it can be as much as 100mA for relays designed to operate from lower voltages. Most ICs (chips) cannot provide this current and a transistor is

usually used to amplify the small IC current to the larger value required for the relay coil. The maximum output current for the popular 555 timer IC is 200mA so these devices can supply relay coils directly without amplification.

RELAY DRIVER-1



Relays are usually SPDT or DPDT but they can have many more sets of switch contacts, for example relays with 4 sets of changeover contacts are readily available. Most relays are designed for PCB mounting but you can solder wires directly to the pins providing you take care to avoid melting the plastic case of the relay. The animated picture shows a working relay with its coil and switch contacts. You can see a lever on the left being attracted by magnetism when the coil is switched on. This lever moves the switch contacts. There is one set of contacts (SPDT) in the foreground and another behind them, making the relay DPDT.



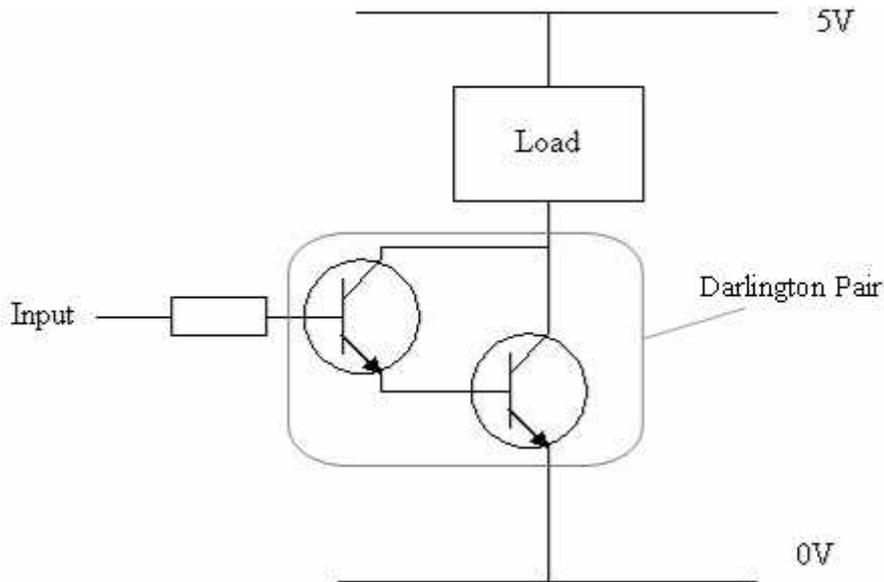
The relay's switch connections are usually labeled COM, NC and NO:

- **COM** = Common, always connect to this, it is the moving part of the switch.
- **NC** = Normally Closed, COM is connected to this when the relay coil is **off**.
- **NO** = Normally Open, COM is connected to this when the relay coil is **on**.

Circuit description:

Generally the relay circuit is designed to control the load. The load may be motor or any other load. The load is turned ON and OFF through relay. The relay ON and OFF is controlled by the pair of switching transistors (BC 547). The relay is connected in the Q2 transistor collector terminal. A Relay is nothing but electromagnetic switching device which consists of three pins. They are Common, Normally close (NC) and Normally open (NO).

The relay common pin is connected to supply voltage. The normally open (NO) pin connected to load. When high pulse signal is given to base of the Q1 transistors, the transistor is conducting and shorts the collector and emitter terminal and zero signals is given to base of the Q2 transistor. So the relay is turned OFF state.



When low pulse is given to base of transistor Q1 transistor, the transistor is turned OFF. Now 12v is given to base of Q2 transistor so the transistor is conducting and relay is turned ON. Hence the common terminal and NO terminal of relay are shorted. Now load gets the supply voltage through relay.

Darlington transistor:

Here we are compound two transistors to get more current gain, this compound structure is called Darlington pair. The **Darlington transistor** (often called a **Darlington pair**) is a compound structure consisting of two bipolar transistors (either integrated or separated devices) connected in such a way that the current amplified by the first transistor is amplified further by the second one.

Transistors have a characteristic called current gain. This is referred to as its **hFE**.

The amount of current that can pass through the load when connected to a transistor that is turned on equals the **input current x the gain of the transistor (hFE)**. The current gain varies for different transistor and can be looked up in the data sheet for the device. Typically it may be 100. This would mean that the current available to drive the load would be 100 times larger than the input to

the transistor. In some application the amount of input current available to switch on a transistor is very low. This may mean that a single transistor may not be able to pass sufficient current required by the load. As stated earlier this equals the **input current x the gain of the transistor (hFE)**.

If it is not be possible to increase the input current then we need to increase the gain of the transistor. This can be achieved by using a Darlington Pair. A Darlington Pair acts as one transistor but with a current gain that equals:

Total current gain (**hFE total**) = current gain of transistor 1 (**hFE t1**) x current gain of transistor 2 (**hFE t2**)

So for example if you had two transistors with a current gain (**hFE**) = 100:

$$(\mathbf{hFE\ total}) = 100 \times 100$$

$$(\mathbf{hFE\ total}) = 10,000$$

You can see that this gives a vastly increased current gain when compared to a single transistor.

Therefore this will allow a very low input current to switch a much bigger load current

Normally to turn on a transistor the base input voltage of the transistor will need to be greater than 0.7V. As two transistors are used in a Darlington Pair this value is doubled. Therefore the base voltage will need to be greater than $0.7V \times 2 = 1.4V$. It is also worth noting that the voltage drop across collector and emitter pins of the Darlington Pair when the turn on will be around 0.9V Therefore if the supply voltage is 5V (as above) the voltage across the load will be around 4.1V ($5V - 0.9V$)

2.3 LCD INTERFACING

LCD INTERFACING WITH MICROCONTROLLER:

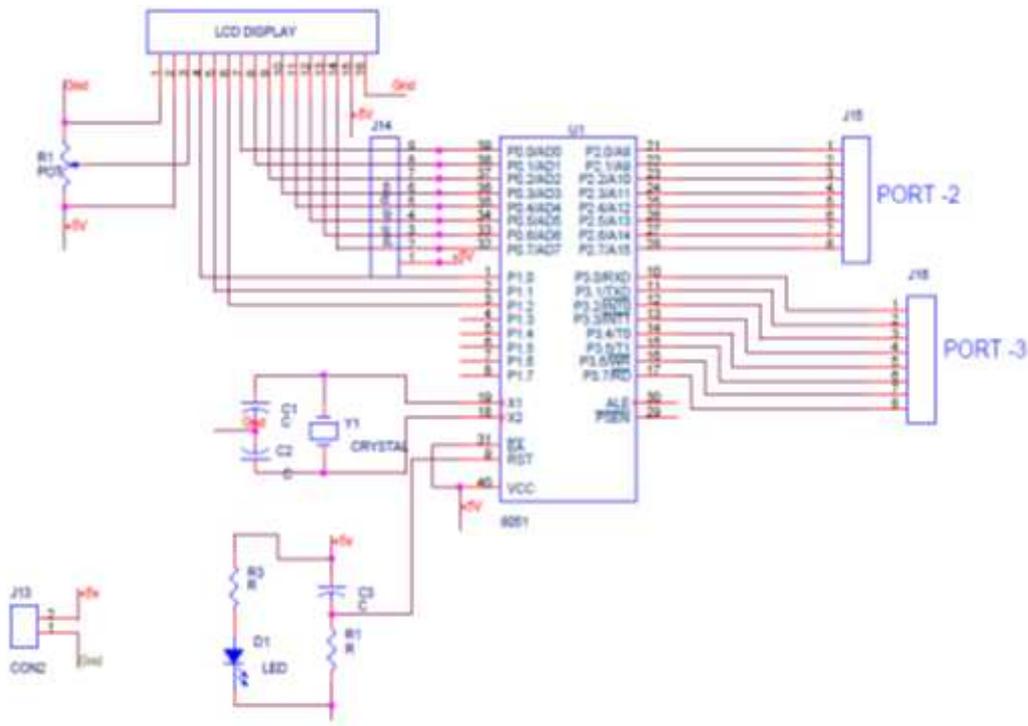


Diagram of LCD interfacing

Operation:

The power supply is given to the controller is +5v, it will operated in +5v only. Here we are interfacing LCD display with the controller. It is 40 pin controllers. It consists of 4 ports namely port0, port1, port2 and port3. Each ports have 8pins.

For LCD interfacing, LCD data line is taken from the port0 (0-7) for data line given to LCD. For LCD control line is taken from the port1(0-2). Here we are using trim pot variable resistor. This resistor is used for adjustment of LCD brightness.

Crystal oscillator is used for producing clock pulse. There are different type of oscillators like RC,LC etc.,Some of the oscillator produce clock pulse which is vary with voltage, temperature fluctuations. Only crystal oscillator produce stable clock pulse, it does not vary with the voltage fluctuations. We never get accurate operation of controller if clock pulse is not stable.

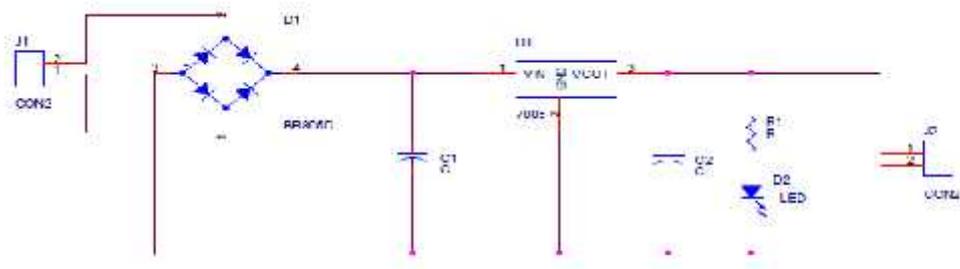
When power ON, the R1,C3 is useful for reset the controller.Then the port2, port3 is used for input/output purpose .Input/output is taken from the port. Data are given to the controller through input ports.

2.4 POWER SUPPLY

Power Supply

Power supply is a reference to a source of electrical power. A device or system that supplies electrical or other types of energy to an output load or group of loads is called a power supply unit. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others. This typically involves converting 240 volt AC supplied by a utility company to a well-regulated lower voltage (+5V) DC for electronic devices.

Diagram:



Blocks of Power Supply

Potential transformer

Rectifier

Filter

Regulator

All electronic circuits need DC power supply either from battery or power pack units. It may not be economical and convenient to depend upon battery power supply. Hence, much electronic equipment contains circuit which converts AC supply voltage into DC voltage at the required level. Transformer involves converting 240 volt AC supply into step downed required AC voltage. Rectifier is defined as an electronic device used for converting AC voltage into unidirectional voltage. A rectifier utilizes unidirectional conduction device like a vacuum diode or PN junction diode. It also consists of filter to remove the pulsating AC component.

Then it will feed into regulator. The regulation of power supplies is done by incorporating circuitry to tightly control the output voltage and/or current of the power supply to a specific value. The specific value is closely maintained despite variations in the load presented to the power supply's output, or any reasonable voltage variation at the power supply's input. We can add more than one regulator according to our needed voltage to get the different output voltages. From this, we can get the regulated +5V and +/-12V DC supply.

Advantage:

- Small size and less weight.
- Less expensive
- High reliable and versatile
- Reduced man power

2.5 GSM

GSM

GSM (Global System for Mobile communications: originally from *Group Special Mobile*) is the most popular standard for mobile phones in the world. Its promoter, the GSM Association, estimates that 80% of the global mobile market uses the standard GSM is used by over 3 billion people across more than 212 countries and territories. Its ubiquity makes international roaming very common between mobile phone operators, enabling subscribers to use their phones in many parts of the world. GSM differs from its predecessors in that both signaling and speech channels are digital, and thus is considered a *second generation* (2G) mobile phone system. This has also meant that data communication was easy to build into the system. GSM EDGE is a 3G version of the protocol.

Cellular radio network

GSM is a cellular network, which means that mobile phones connect to it by searching for cells in the immediate vicinity. There are five different cell sizes in a GSM network—macro, micro, pico, femto and umbrella cells. The coverage area of each cell varies according to the implementation environment. Macro cells can be regarded as cells where the base station antenna is installed on a mast or a building above average roof top level. Micro cells are cells whose antenna height is under average roof top level; they are typically used in urban areas. Pico cells are small cells whose coverage diameter is a few dozen meters; they are mainly used indoors.

Cell horizontal radius varies depending on antenna height, antenna gain and propagation conditions from a couple of hundred meters to several tens of kilometers. The longest distance the GSM specification supports in practical use

is 35 kilometers (22 mi). There are also several implementations of the concept of an extended cell where the cell radius could be double or even more, depending on the antenna system, the type of terrain and the timing advance. Indoor coverage is also supported by GSM.

GSM frequencies

GSM networks operate in a number of different frequency ranges (separated into GSM frequency ranges for 2G and UMTS frequency bands for 3G). Most 2G GSM networks operate in the 900 MHz or 1800 MHz bands. Some countries in the Americas (including Canada and the United States) use the 850 MHz and 1900 MHz bands because the 900 and 1800 MHz frequency bands were already allocated. Most 3G GSM networks in Europe operate in the 2100 MHz frequency band. The rarer 400 and 450 MHz frequency bands are assigned in some countries where these frequencies were previously used for first-generation systems.

GSM-900 uses 890–915 MHz to send information from the mobile station to the base station (uplink) and 935–960 MHz for the other direction (downlink), providing 125 RF channels (channel numbers 1 to 124) spaced at 200 kHz. Duplex spacing of 45 MHz is used. In some countries the GSM-900 band has been extended to cover a larger frequency range. This 'extended GSM', E-GSM, uses 880–915 MHz (uplink) and 925–960 MHz (downlink), adding 50 channels (channel numbers 975 to 1023 and 0) to the original GSM-900 band. Time division multiplexing is used to allow eight full-rate or sixteen half-rate speech channels per radio frequency channel. There are eight radio timeslots (giving eight burst periods) grouped into what is called a TDMA frame. Half rate channels use alternate frames in the same timeslot. The channel data rate for all 8 channels is 270.833 k bit/s, and the frame duration is 4.615 ms. The

transmission power in the handset is limited to a maximum of 2 watts in GSM850/900 and 1 watt in GSM1800/1900.

Network structure

The network behind the GSM seen by the customer is large and complicated in order to provide all of the services which are required. It is divided into a number of sections and these are each covered in separate articles.

- The Base Station Subsystem (the base stations and their controllers).
- The Network and Switching Subsystem (the part of the network most similar to a fixed network). This is sometimes also just called the core network.
- The GPRS Core Network (the optional part which allows packet based Internet connections).
- All of the elements in the system combine to produce many GSM services such as voice calls and SMS.

Subscriber Identity Module (SIM)

One of the key features of GSM is the Subscriber Identity Module, commonly known as a SIM card. The SIM is a detachable smart card containing the user's subscription information and phone book. This allows the user to retain his or her information after switching handsets. Alternatively, the user can also change operators while retaining the handset simply by changing the SIM. Some operators will block this by allowing the phone to use only a single SIM, or only a SIM issued by them; this practice is known as SIM locking, and is illegal in some countries.

2.6 RFID SYSTEM

Radio frequency identification (RFID) is a general term that is used to describe a system that transmits the identity (in the form of a unique serial number) of an object wirelessly, using radio waves. RFID technologies are grouped under the more generic Automatic Identification (Auto ID) technologies. The barcode labels that triggered a revolution in identification systems long time ago, are inadequate in an increasing number of cases. They are cheap but the stumbling block is their low storage capacity and the fact that they cannot be reprogrammed.

A feasible solution was putting the data on silicon chips. The ideal situation is contactless transfer of data between the data carrying device and its reader. The power required to operate the electronic data carrying device would also be transferred from the reader using contactless technology. These procedures give RFID its name.

Before RFID can be understood completely, it is essential to understand how Radio Frequency communication occurs. RF (Radio Frequency) communication occurs by the transference of data over electromagnetic waves. By generating a specific electromagnetic wave at the source, its effect can be noticed at the receiver far from the source, which then identifies it and thus the information. In an RFID system, the RFID tag which contains the tagged data of the object generates a signal containing the respective information which is read by the RFID reader, which then may pass this information to a processor for processing the obtained information for that particular application.

Thus, an RFID System can be visualized as the sum of the following three components:

- RFID tag or transponder
- RFID reader or transceiver
- Data processing subsystem

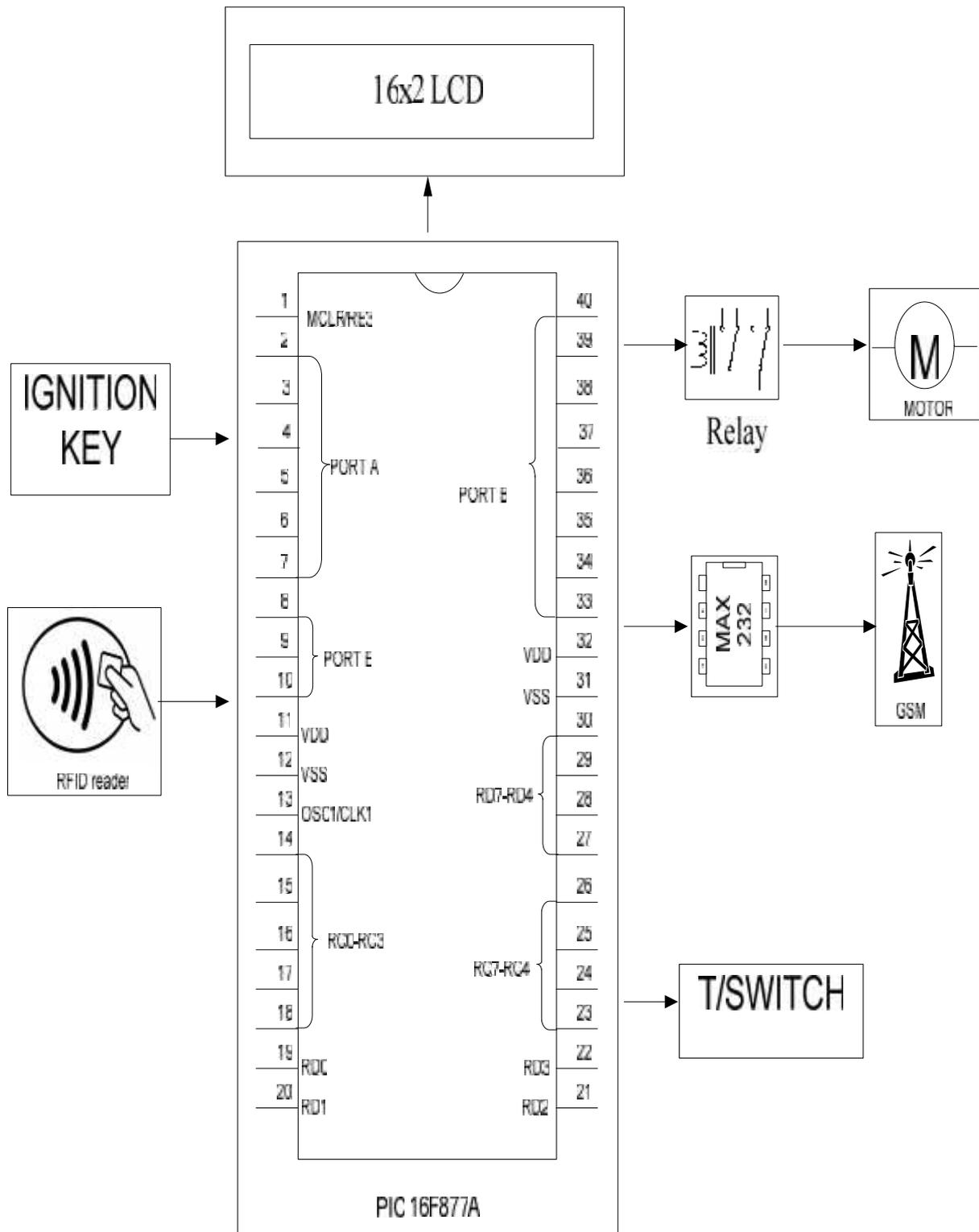
An RFID tag is composed of an antenna, a wireless transducer and an encapsulating material. These tags can be either active or passive. While the active tags have on-chip power, passive tags use the power induced by the magnetic field of the RFID reader. Thus passive tags are cheaper but with lower range (<10mts) and more sensitive to regulatory and environmental constraints, as compared to active tags. An RFID reader consists of an antenna, transceiver and decoder, which sends periodic signals to inquire about any tag in vicinity. On receiving any signal from a tag it passes on that information to the data processor.

The data processing subsystem provides the means of processing and storing the data.

RFID systems can also be differentiated based on the frequency range it uses. The common ranges are Low-Frequency (LF: 125 - 134.2 kHz and 140 - 148.5 kHz), High-Frequency (HF: 13.56 MHz) and Ultra-High-Frequency (UHF: 868 MHz - 928 MHz).

Low-frequency systems have short reading ranges and lower system costs. They are most commonly used in security access, asset tracking, and animal identification applications. High-frequency systems, offering long read ranges (greater than 90 feet) and high reading speeds, are used for such applications as railroad car tracking and automated toll collection. However, the higher performance of high-frequency RFID systems incurs higher system costs.

2.7 BLOCK DIAGRAM



Block Diagram of proposed system

2.9 WORKING PRINCIPLE

Working Principle

First select the mode of operation, vehicle section or traffic section.

In the vehicle section the person needs to insert the key in the keyhole and show the license having RFID tag in the RFID reader to start the vehicle.

If RFID information is matched with the the owner's information, the relay will be closed and the engine willbe turned on.

Else microcontroller orders the GSM modem to give the missed call and send alert message to the owner.

If owner sends the message as 'stop', the relay will be opened and motor will not be turned on.

If owner sends the message as 'start', the relay will be closed and motor will be turned on.

In the traffic section Vehicle information is stored in the Number Plate RFID tag. This information is read by the traffic police using RFID reader.

The vehicle's informations are like owner's name and date of birth, engine number, vehicle model and insurance expiry date.

CHAPTER 3

SOFTWARE DESCRIPTION

3.1 Embedded C

3.1.1 Introduction:

Embedded C is a set of language extensions for the C programming language by the C Standards committee to address commonality issues that exist between C extensions for different embedded systems. Historically, embedded C programming requires nonstandard extensions to the C language in order to support exotic features such as fixed-point arithmetic, multiple distinct memory banks, and basic I/O operations. In 2008, the C Standards Committee extended the C language to address these issues by providing a common standard for all implementations to adhere to. It includes a number of features not available in normal C, such as fixed-point arithmetic, named address spaces, and basic I/O hardware addressing.

Embedded C uses most of the syntax and semantics of standard C, e.g., main() function, variable definition, data type declaration, conditional statements (if, switch, case), loops (while, for), functions, arrays and strings, structures and union, bit operations, macros, etc.

3.1.2 Necessity:

During infancy years of microprocessor based systems, programs were developed using assemblers and fused into the EPROMs. There used to be no mechanism to find what the program was doing. LEDs, switches, etc. were used to check for correct execution of the program. Some 'very fortunate' developers

had In-circuit Simulators(ICEs), but they were too costly and were not quite reliable as well. As time progressed, use of microprocessorspecific assembly-only as the programming language reduced and embedded systems moved onto C as the embedded programming language of choice. C is the most widely used programming language for embedded processors/controllers. Assembly is also used but mainly to implement those portions of the code where very high timing accuracy, code size efficiency, etc. are prime requirements. As assembly language programs are specific to a processor, assembly language didn't offer portability across systems. To overcome this disadvantage, several high level languages, including C, came up. Some other languages like PLM, Modula-2, Pascal, etc. also came but couldn't find wide acceptance. Amongst those, C got wide acceptance for not only embedded systems, but also for desktop applications. Even though C might have lost its sheen as a mainstream language for general purpose applications, it still is having a strong hold in embedded programming. Due to the wide acceptance of C in the embedded systems, various kinds of support tools like compilers & cross-compilers, ICE, etc. came up and all this facilitated development of embedded systems using C. Assembly language seems to be an obvious choice for programming embedded devices. However, use of assembly language is restricted to developing efficient codes in terms of size and speed. Also, assembly codes lead to higher software development costs and code portability is not there. Developing small codes are not much of a problem, but large programs/projects become increasingly difficult to manage in assembly language. Finding good assembly programmers has also become difficult nowadays. Hence high level languages are preferred for embedded systems programming.

3.1.2 Advantages:

- It is small and simpler to learn, understand, program and debug.
- Compared to assembly language, C code written is more reliable and scalable, more portable between different platforms.
- C compilers are available for almost all embedded devices in use today, and there is a large pool of experienced C programmers.
- Unlike assembly, C has advantage of processor independence and is not specific to any particular microprocessor/microcontroller or any system. This makes it convenient for a user to develop programs that can run on most of the systems.
- As C combines functionality of assembly language and features of high level languages, C is treated as a 'middle-level computer language' or 'high level assembly language'.
- It is fairly efficient.
- It supports access to I/O and provides ease of management of large embedded projects.
- Java is also used in many embedded systems but Java programs require the Java Virtual Machine (JVM), which consumes a lot of resources. Hence it is not used for smaller embedded devices.

3.2 MP LAB IDE

3.2.1 Introduction

Integrated Development Environment (IDE) is an application that has multiple functions for software development. MPLAB IDE an executable program that integrates a compiler, an assembler, a project manager, an editor, a debugger, simulator, and an assortment of other tools within one Windows application. A user developing an application should be able to write code, compile, debug and test and application without leaving the MPLAB IDE desktop.

3.2.2 Features:

- Provides a new Call Graph for navigating complex code
- Supports Multiple Configurations within your projects
- Supports Multiple Versions of the same compiler
- Support for multiple Debug Tools of the same type
- Supports Live Parsing
- Import existing MPLAB[®] 8 IDE projects and use either IDE for the same source
- Supports hyperlinks for fast navigation to declarations and includes
- Supports Live Code Templates

3.3 PROTEUS SOFTWARE

3.3.1 Introduction

Proteus 8 is best simulation software for various designs with microcontroller. It is mainly popular because of availability of almost all microcontrollers in it. So it is a handy tool to test programs and embedded designs for electronics hobbyist. You can simulate your programming of microcontroller in Proteus 8 Simulation Software.

After simulating your circuit in Proteus 8 Software you can directly make PCB design with it so it could be a all in one package

3.3.2 Features:

- SIS Schematic Capture an easy to use yet and extremely powerful tool for entering your design
- PROSPICE Mixed mode SPICE Simulation industry standard SPICE3F5 simulator upgradeable to our unique virtual system modeling technology
- Modern Graphical User Interface standardized across all modules
- ARES PCB Layout
- Runs on Windows 98/ME/2000/XP or Later
- Technical Support direct form the author

Intelligent Schematic Input System (ISIS):

ISIS lies right at the heart of the **PROTUES** system and is far more than just another schematic package. It has powerful environment to control most aspects of the drawing appearance. Whether your requirement is the rapid entry of complex design for simulation & PCB layout, Or the creation of attractive Schematic for publication **ISIS** is the right tool for the job Product Features

- Produces publication quality schematic
- Style templates allow customization of supplied library
- Mouse driven context sensitive user interface
- Automatic wire routing and junction dot placement
- Full support for buses including sub- circuit ports and bus pins
- Large and growing component library of over 8000 parts

VSM (Virtual System Modeling):

Proteus VSM is an extension of the PROSPICE simulator that facilities co-simulation of microprocessor based design including all the associated electronics. Furthermore, you can interact with the microcontroller software through the use of animated keypads, switches, buttons, LEDs, lamps and even LCD displays.

Features:

- CPU models available for many popular microcontrollers including PIC, AVR, HC11 and 8051
- Interactive device models include LED and LCD displays, RS232 terminal, universal keypad plus a range of switches, buttons, pots, LEDs, 7 segment displays and more.
- Extensive debugging facilities including register and memory contents, breakpoints and single step modes.
- Source level debugging for selected development tools including IAR C-SPY and KeilVision 2.

3.4 PIC BOOT LOADER

3.4.1 Introduction

A bootloader is a program that stays in the microcontroller and communicates with the PC (usually through the serial interface). The bootloader receives a user program from the PC and writes it in the flash memory, then launches this program in execution. Bootloaders can only be used with those microcontrollers that can write their flash memory through software. The bootloader itself must be written into the flash memory with an external programmer. In order for the bootloader to be launched after each reset, a "goto bootloader" instruction must exist somewhere in the first 4 instructions; There are two types of bootloaders, some that require that the user reallocate his code and others that by themselves reallocate the first 4 instructions of the user program to another location and execute them when the bootloader exits.

- It is the smallest boot loader, taking less than 100 words of program space;
- Supports families of PIC devices: 16F, 18F, dsPIC30 (those devices that support self-programming)

3.4.2 Features:

- Size of only 100 words; (all versions, for 16F,18F,dsPIC occupy less than 100 words);
- Can write flash, EEPROM and configuration bytes(18F);
- On reset, waits 1 second (adjustable) for a message from the PC, if not received, launch user application;

- The .ASM file can be easily modified and adapted for any frequency (or baud rate);

3.5 ALGORITHM

Algorithm

- 1). Start the program
- 2). Insert the key
- 3). Show the license RFID tag to the RFID reader
- 4). If both key and license are present, go to step 6.
- 5). Else stop the program.
- 6). Check the license information.
- 7). If license information matches with the owner's information, the relay will be closed.
- 8). Else give the missed call and alert message to the owner.
- 9). If the owner sends the message as 'stop', the relay will be opened.
- 10). If the owner sends the message as 'start', the relay will be closed and motor will be turned on.
- 11). Stop the program.

CHAPTER 4

CONCLUSION

CONCLUSION

Vehicle tracking system is becoming increasingly important in large cities and it is more secured than other systems. Nowadays vehicle Stealing is rapidly increasing. Currently GPS tracking system is used in the vehicle. But this system is not enough to provide security. Due to this reason, this project is introducing the wireless technology effectively for the automotive environments by using the GSM Modem. It is used in sending sms intimation to owner in case of theft of vehicle. So the vehicle is protected by sending the message to GSM modem. The extra feature in this project is to avoid vehicle driving without having the license. Reading the vehicle's information will help the traffic police to check the details of the vehicle.

REFERENCES

1. edgfix.com projects, <http://www.edgfixkits.com/theft-intimation-of-the-vehicle-over-gsm-by-sms-with-user-programable-number-features-to-owner-who-can-stop-the-engine-remotely>. [2]
2. Muhammad Ali Mazidi, Janice Gillspie, Mckinlay, Rolin D., “TheMicrocontroller in Embedded System: using Assembly and C,” 2Ndedition, Pearson Education, 2007.[4]
3. Mouly, Michel, Pautet, M. Bernadette (2002). The GSM System for Mobile Communications. Telecom Publishing, ISBN:0945592159[5]
4. 8051 projects, <http://www.8051projects.info/content/projects/7-vehicle-tracking-system-using-gps-gsm-modem.html>. [6]
5. MarkoWolf, AndreWeimerskirch, and ThomasWollinger, “State of theArt: Embedding security in vehicles”, Journal on Embedded Systems, Volume 2007, Article ID 74706.[7]
6. P. Hubaux, S. C Apkun, and J. Luo, “The security and privacy of smartvehicles,” IEEE Security & Privacy Magazine, vol. 2, no. 3, pp. 49–55,2004.[8]