"GSM BASED CAR SECURITY SYSTEM"

Project Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology

In

Electronics and Communication Engineering

By

MAYANK GAUTAM 061073

NIMIT GUPTA 061082

VISHAL DOGRA 061144

Under the Supervision of

Mr. Vivek Kumar Sehgal

Jaypee University of Information Technology

Waknaghat, Solan - 173 234,

Himachal Pradesh

Certificate

This is to certify that the project report entitled **"GSM BASED CAR SECURITY SYSTEM"**, submitted by **Mayank Gautam, Nimit Gupta, and Vishal Dogra** in partial fulfillment for the award of degree of Bachelor of Technology in Electronics and Communication Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

Vivek Sehgal

Date: MAY 2010

Mr. VIVEK KUMAR SEHGAL

(Sr. LECTURAR)

Certified that this work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

1. **Mayank Gautam (061073)**

2. **Nimit Gupta (061082)**

3. **Vishal Dogra (061144)**

Acknowledgement

It has been a wonderful and intellectually stimulating experience working on the **"GSM BASED CAR SECURITY SYSTEM"**. Theft is a serious issue and this project can be of great use to the vehicles, which will enable us to keep a check on theft activities.

We wish to express our earnest gratitude to **Mr. Vivek Kumar Sehgal** for providing us invaluable guidance, suggestions and providing us with the finest details of the subjects, which allowed us to present our project in its final form.

We gratefully acknowledge all the staff members of Electronics and Communication Engineering Department of Jaypee University of Information Technology, Waknaghat, for providing us with all the facilities required for the completion of our project.

We would like to thank **Mr. Manoj Pandey** for letting us work in the electrical lab and providing us with certain basic hardware requirements as per need.

Date: May 2010

Mayank Gautam

Nimit Gupta

Vishal Dogra

TABLE OF CONTENTS

1. INTRODUCTION 1

1.1 Understanding current scenario of car security systems.....2

1.2 Aftermarket alarm triggers..... ..2

1.3 Effectiveness of Car Alarms.....3

2. BLOCK DIAGRAM 5

2.1 Door Switch.....6

2.2 Car Battery.....6

2.3 Microcontroller.....6

2.4 Opto-Isolator.....7

2.5 Cell phone.....7

3. CIRCUIT MODULING 8

3.1 Circuit Diagram.....9

3.2 Circuit Diagram Description.....10

4. HARDWARE REQUIREMENTS 11

4.1 The Basic Components.....11

4.1.1 Resistor.....11

4.1.2 Capacitor.....13

4.1.3 LED.....14

4.1.4 Diode.....15

4.1.5 Voltage Regulator.....16

5. COMPONENTS TESTING 18

5.1 Resistor..... 18

5.2 Capacitor..... ...19

5.3 Diode.....20

5.4 LED..... 20

6. THE FAMILY 8051 AND MICROCONTROLLER 89s51 22

6.1 Introduction.....22

6.2 Types of memory.....23

6.2.1 Code Memory.....23

6.2.2 External RAM.....24

6.2.3 On-Chip Memory.....25

6.3 Addressing Modes.....25

6.4 Microcontroller AT89s52 28

6.4.1 Features..... 28

6.4.2 DESCRIPTION.....29

6.4.3 Pin Description.....30

7. SOLDERING 35

7.1 Introduction.....35

7.2 Applications.....35

7.3 Soldering Techniques.....36

7.4 Precautions.....37

8. ASSEMBLY SOURCE FOR µC 38

8.1 Software requirement.....38

8.1.1 Introduction to Keil.....38

8.1.2 How to use Keil..... 40

8.2 ORCAD.....39
8.3 Conversion of assembly language into HEX file.....40
8.4 Source Code.....44

9. CONCLUSION 47

10. BIBLIOGRAPHY 48

Abstract

With rising technology a major challenge is poised in front of engineers, i.e. to be able to secure all possible technology in the most efficient manner. Accepting this challenge and motivating ourselves through the same, we decided to work in the field of security system designs.

Through our initial research it was evident that the automobile sector is way behind in blending the latest information and communication technologies in their security system. Hence in this project we have presented an advanced security system for cars. We have used the current technology of GSM. The reason being the present highly developed infrastructure for systems supporting GSM.

GSM based security system are much more stout then 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 high traffic zones like metros where these alarms go off unnecessarily. Thus, in our project, a new system with GSM techniques and a better decision making process is built to make our vehicle more secure. It is a unique wireless home/car 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 Phone Call" as soon as your burglar alarm system intercepts an intrusion.

This project proposes an 8 bit embedded controller interfaced with GSM model. The control mechanism is based on DTMF tones generated by mobile phone when the number keys are pressed.

Chapter 1

INTRODUCTION

Humans have always felt very possessive of their belongings. During ancient times, inhabitants used to live in secure caves so that they would be protected from unwanted intrusion and from deadly animals. The advancement of civilization witnessed better and improved means of home security system being implemented by humans. Technological revolution of modern age has resulted in concept of home security finding widespread popularity. Everyone worth his salt thinks of protecting his hard earned possessions. Urban population specifically has become very conscious of the important aspect of home security. India security system has several useful tools available to protect your beloved home.

One can find latest technological innovations being employed to protect not only houses but also offices, buildings, and warehouses etc. Nobody likes to be robbed of his precious possessions and hence opting for a secure system makes sense. Also the fact that, crime rate has kept increasing further emphasizes the need of upgrading the home security. With changing times, even petty thieves have acquired new skills to rob your belongings. Most of the urban cities and metros are witnessing increased rate of crimes and that requires for installation of a safe and secure system to protect your valuables, family and home.

One can look for several products pertaining to security systems India. Big corporate giants and many renowned companies have launched quality security products to safeguard homes and precious belongings. One can find burglar alarm, fire alarm, home security kits, new age doors and windows, security cameras, motion detectors, personal safety products, wire and wireless security products and other such products. These products are available in low end as well as high end range. Depending upon one's budget, he or she can go for the required security system suitable to the place to be secured. One

can also find numerous security experts who provide customized suggestions for home security.

1.1 Understanding current scenario of car security systems

The present technology can be understood through the following text.

A car alarm is an electronic device installed in a vehicle in an attempt to discourage theft of the vehicle itself, its contents, or both. Car alarms work by emitting high-volume sound (usually a siren, klaxon, pre-recorded verbal warning, the vehicle's own horn, or a combination thereof) when the conditions necessary for triggering are met, as well as by flashing some of the vehicle's lights, and (optionally) notifying the car's owner via a paging system and interrupting various electrical circuits necessary for the car to start.

Car alarms should not be confused with immobilizers; although the purpose of both may be to deter car theft, they operate in a dissimilar fashion. An immobilizer generally will not offer any audible or visual theft deterrence, nor require any additional input from the driver than the driver of a non-immobilizer car.

Car alarms can be divided into two categories:

- OEM (built-in to the vehicle at the factory)
- Aftermarket (installed at any time after the car has been built, such as by the new car dealer, an auto accessories store, or the vehicle's owner)

Alarms come with a mix of features. Remote car alarms typically consist of an additional radio receiver that allows the owner to wirelessly control the alarm from a key fob. Remote car alarms typically come equipped with an array of sensors along with immobilizers and motion detectors.

1.2 Aftermarket alarm triggers

The simplest aftermarket alarms are one-piece units with a siren and control module. Such a unit will typically contain a shock sensor and two wires (12 volt constant power and ground) which are connected to the car's battery. This type of alarm is triggered

by vibration transferred to the shock sensor, or by voltage changes on the input (the alarm assumes that a sudden change in voltage is due to a door or trunk being opened, or the ignition being turned on); however it is very prone to false triggers on a late-model vehicle with many electronic control modules, which can draw current with the car off. For this last reason these alarms are increasingly becoming obsolete.

More sophisticated aftermarket alarms are wired in to the vehicle's circuits individually. Typically, these alarms have inputs for power and ground, as well as for positive- and negative-switched door open circuits, negative trunk and/or hood circuits, and ignition-switched circuits to detect the vehicle being turned on; aftermarket alarms also usually have a shock sensor which may be built into the control module or external to it.

In addition, some aftermarket alarms have provisions for optional sensors which can sense the vehicle being tilted (this alerts against unauthorized towing), glass breakage (which can sometimes be done without an impact sufficient to trigger the shock sensor), or motion inside or immediately outside the vehicle (this is a concern on convertibles).

The sensors mentioned here are usually adjustable to in order to avoid false alarms - for example a shock sensor will sometimes vibrate due to a loud noise in the area, or an accidental bump to the car from a passerby. This can cause the alarm to falsely sense an attempted break-in.

Some alarms will bypass some or all of the inputs at times by design. For example, Directed Electronics alarms have a feature called "Nuisance Prevention Circuitry" which ignores any input which has triggered 3 times within 1 hour, unless the car owner turns the ignition on to reset it.

Other alarms can bypass some of their inputs via a button combination on the remote, or when remote starting (if the alarm supports this feature).

1.3 Effectiveness of Car Alarms

Since most car alarms are triggered accidentally (frequently because of high sensitivity settings), people often ignore alarms. The New York City Police Department claims that car alarms are actually making the crime problem worse because false alarms are so common that people simply ignore them.

Because of the large number of false alarms with car alarms, many vehicle manufacturers no longer factory-fit simple noise-making alarms, instead offering silent—but effective—immobilizers. Alternatively, an aftermarket vehicle tracking_system can enable the police to trace stolen vehicles. Most police tracking systems require the user to pay a recurring fee, whereas factory immobilizers are included in the purchase price of the vehicle. GPS locating systems enable the owner of the vehicle to lock and unlock, track, and disable the starter of the vehicle online.

Yet another class of security covers aftermarket car alarms that include 2-way paging controllers. Some 2-way systems have an LCD icon display that can pinpoint the actual part of the vehicle being threatened, and many alert the user with beeps or silent vibration.

How to synergize the latest technologies with our system?

We have included the following technologies in our system:

1. DTMF: Dual Tone Multiple Access.
2. Microcontroller
3. Programming language

Chapter 2

BLOCK DIAGRAM

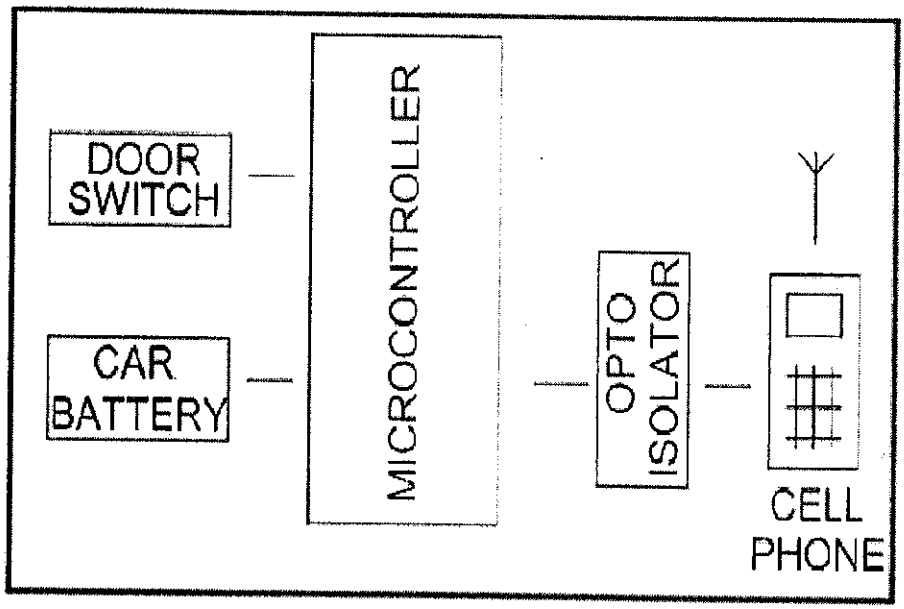


Figure 2.1: Block Diagram

BLOCK DIAGRAM DESCRIPTION

2.1. DOOR SWITCH: - Door Switch is push-to-on switch. When someone tries to open the door it signals the controller.

2.2. CAR BATTERY: - Provides the battery voltage of 12V to the circuit.

2.3. MICROCONTROLLER: - The controller will check the IC as per the truth table of the IC which is stored in its ROM. It will check each and every pin of the IC and produce the Output detail. Like "Gate 1 is good", "Gate is bad", "Counter 1 is good" etc.

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 4K bytes of In-System Programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry

Standard 80C51 instruction set and pin out. The on-chip Flash allows the program Memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S52 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and Clock circuitry. In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes.

The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and Interrupt system to continue functioning. The Power-down mode saves the RAM

contents but freezes the oscillator, disabling all other chip functions until the next external Interrupt or hardware reset.

2.4. OPTO-ISOLATOR: - A device that uses a short optical transmission path to transfer an electronic signal between elements of a circuit, typically a transmitter and a receiver, while keeping them electrically isolated

2.5. CELL PHONE: - The cell phone we are using for this project is based on the GSM technology having uplink frequency (890-915MHz) and downlink frequency (935-960MHz).

Chapter 3

CIRCUIT MODULING

The circuit diagram of the hardware is explained in this chapter. In this chapter, we will explain the major components used n the circuit.

3.1 Circuit Working

The microcontroller AT89s52 is the brain of the car security system. The signal from the car is constantly being sent to the controller which is being sensed. If there is any intrusion or any breakage then the signal breaks and then no signal is coming to the controller. As soon as the signal stops the controller send a signal to the opto-isolator which is attached to the cell phone which makes a call to other cell phone whose number is stored in the emergency dialing of the phone. The LM7805 converts the 9V supply to the 5V so as to drive the controller. The opto-isolator protects the circuit from any sudden high voltage which could damage the circuit. The opto-isolator itself blows out and thus saves the remaining circuit.

3.2 The Circuit Diagram

The circuit diagram shows in the figure 5.1. It shows all the components used in the sequence in they should have been.

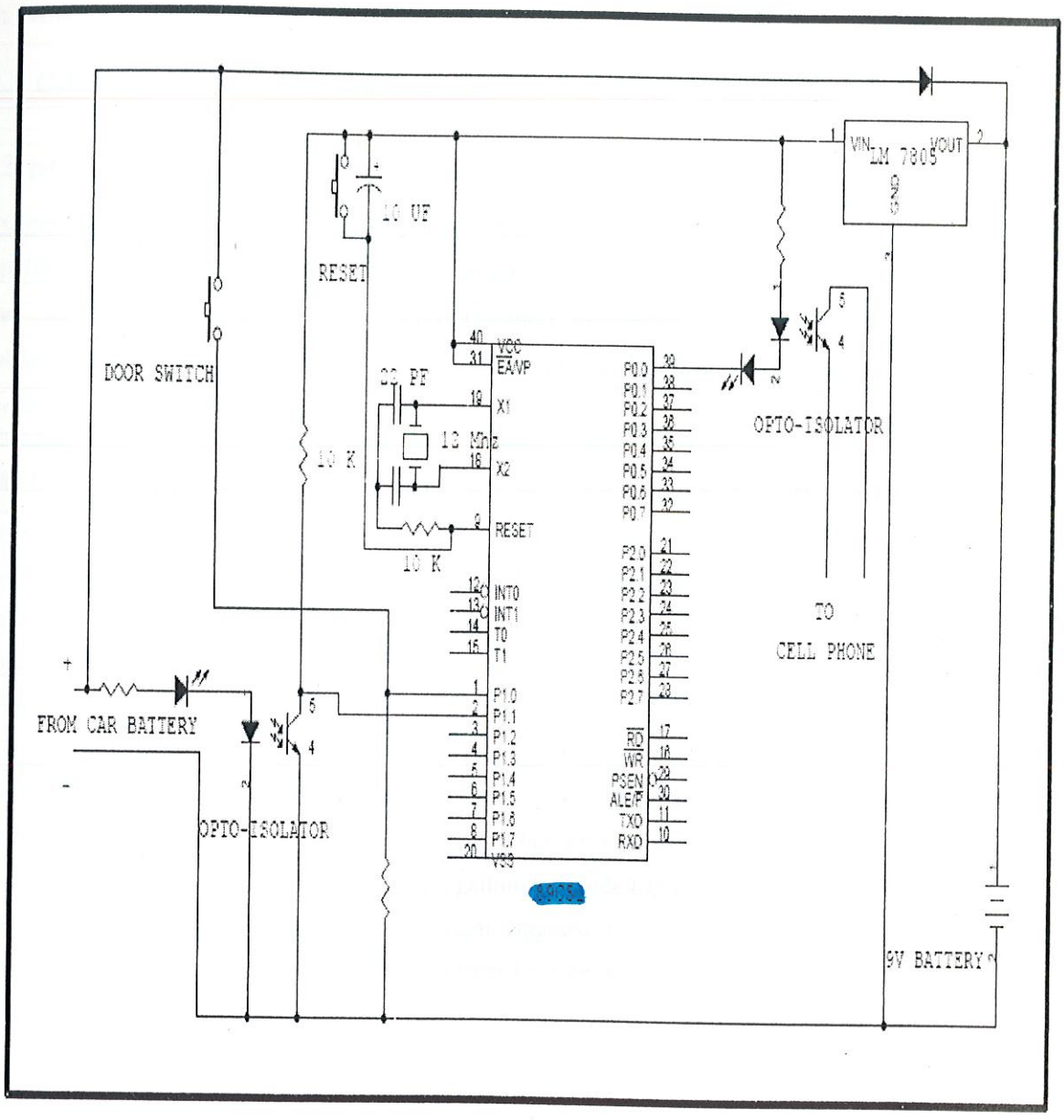


Figure 3.1: The Circuit Diagram

3.2.1 DOOR SWITCH: - Door Switch is push-to-on switch. When someone tries to open the door, it signals the controller.

3.2.2 CAR BATTERY: - Provides the battery voltage of 12V to the circuit.

3.2.3 µC AT89s52: - The AT89s52 is a low-power, high performance CMOS 8-bit microcontroller with 4K bytes of In-System Programmable (ISP) Flash memory. It is compatible with the industry-standard 80c52 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. It can be both serially and parallely programmed. It is cost-effective (may cost just Rs.55).

3.2.4 Opto-Isolator: - There are many situations where signals and data need to be transferred from one subsystem to another within a piece of electronics equipment, or from one piece of equipment to another, without making a direct 'ohmic' electrical connection. Often this is because the source and destination are (or may be at times) at very different voltage levels, like a microprocessor which is operating from 5V DC but being used to control a triac which is switching 240V AC. In such situations the link between the two must be an isolated one, to protect the microprocessor/microcontroller from over voltage damage. These use a beam of light to transmit the signals or data across an electrical barrier, and achieve excellent isolation. Optocoupler typically come in a small 6-pin or 8-pin IC package, but are essentially a combination of two distinct devices: an optical transmitter, typically a gallium arsenide LED (light-emitting diode) and an optical receiver such as a phototransistor or light-triggered diac. The two are separated by a transparent barrier which blocks any electrical current flow between the two, but does allow the passage of light.

Chapter 4

HARDWARE REQUIREMENTS

This chapter basically deals with all the hardware components required for putting it into a working mode. Each component is explained with its basic theory which helps us in getting the detailed insight of as to how that particular component is contributing in the circuit. Hence a defined role of every component is described in this chapter.

4.1 The Basic Components

This very section gives the detailed insight about each and every component like resistor, capacitor, diode etc. used in the basic modules of the circuit designing process.

4.1.1 Resistor

The resistor's function is to reduce the flow of electric current.

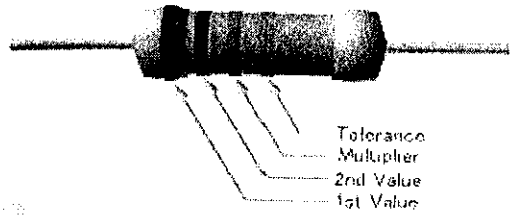
This symbol —WW— is used to indicate a resistor in a circuit diagram, known as a schematic.

Resistance value is designated in units called the "Ohm". A 1000 Ohm resistor is typically shown as 1K-Ohm (kilo Ohm), and 1000 K-Ohms is written as 1M-Ohm (mega ohm).

The first three colored rings define the resistance value while fourth ring gives the tolerance of the resistor.

Resistors are the passive components used to oppose the flow of current in electronic circuits. The opposition offered to current depends on the resistance of the resistor. The current flowing through a circuit can be controlled as per requirement with the help of the resistor of proper resistance.

Resistor color code

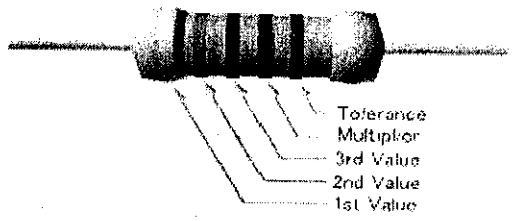


Example 1

(Brown=1),(Black=0),(Orange=3)

10×10^3 = 10k ohm

Tolerance(Gold) = ±5%



Example 2

(Yellow=4),(Violet=7),(Black=0),(Red=2)

470×10^2 = 47k ohm

Tolerance(Brown) = ±1%

Color	Value	Multiplier	Tolerance (%)
Black	0	0	-
Brown	1	1	±1
Red	2	2	±2
Orange	3	3	±0.05
Yellow	4	4	-
Green	5	5	±0.5
Blue	6	6	±0.25
Violet	7	7	±0.1
Gray	8	8	-
White	9	9	-
Gold	-	-1	±5
Silver	-	-2	±10
None	-	-	±20

Figure 4.1: color coding

4.1.2 Capacitor

Non-Polarized capacitor

Electrolytic Capacitor

Figure 4.2: Types of capacitors

Capacitors are the passive components excessively used in electronic circuits. A capacitor construction consists of two metallic plates or film separated by a dielectric. The dielectric used can be paper, polyester, mica, ceramic or electrolyte.

The capacitors are also rated for maximum value of voltage which they can withstand. This maximum operating voltage is decided on basis of the breakdown voltage of the dielectric used between the plates. Breakdown voltage of a capacitor is the voltage at which the dielectrics cease to be a dielectric, and start conducting.

Capacitor is another passive component used in electronic circuits. A capacitor construction consists of two metallic plates or films separated by a dielectric. The dielectric used can be paper, polyester, mica, ceramic or electrolyte. It is the dielectric used in the capacitors on basis of which the capacitors are categorized. Therefore, capacitor can be of following types: Paper Capacitor, Polyester Capacitor, Mica Capacitor, Ceramic Capacitor and Electrolytic Capacitor. The capacity ranges of these capacitors vary from type to type. Ceramic capacitors have usually very low capacitance with few pico farad rating. The electrolytic capacitors are high capacity ones, with capacity up to few thousand micro farads

The capacitors are also rated for maximum value of voltage which they can withstand. This maximum operating voltage is decided on basis of the breakdown voltage of the dielectric used between the plates. Breakdown voltage of a capacitor is the voltage at which the dielectric used ceases to be a dielectric, and starts conducting. This gives rise to short – circuiting of the plates, and hence, the capacitor.

4.1.3 LED

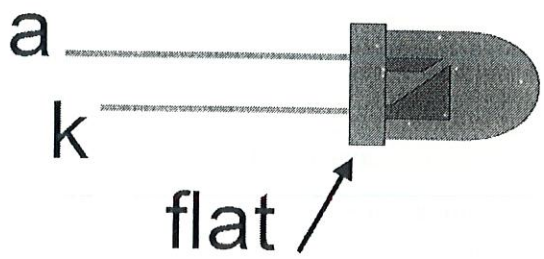


Figure 4.3: LED

Light Emitting Diodes (LED'S) are specially constructed of semiconductor material which radiate energy in form of light when the electron – hole recombination take place during its forward biasing. Such semiconductor materials are GaAsP, GaP etc. Different materials radiate light energy in different frequencies, which our eyes perceive or see as different colors. LED'S are widely used as indicators, where visual indication of current is required in the form of light, and also as decorative lights.

Testing a Light Emitting Diode (LED) is similar to testing like that of a regular diode, except that LED'S have higher forward voltage drops or higher forward resistance than a regular diode. This is because these LED'S are not made up of regular semiconductors like Silicon or Germanium, but of different semiconductors named earlier. These semiconductors have higher forward break over voltage than that in case of silicon or germanium. Further, this forward voltage drop also varies for different colors of LED'S, because they are further different in terms of the specialized semiconductors used.

Forward voltage drop can vary typically from about 1.5 volt for red LED to up to 3 volts for blue LED.

4.1.4 Diode IN4007

A diode is a semiconductor device which allows current to flow through it in only one direction. Although a transistor is also a semiconductor device, it does not operate the way a diode does. A diode is specifically made to allow current to flow through it in only one direction.

Some ways in which the diode can be used are listed here.

- A diode can be used as a rectifier that converts AC (Alternating Current) to DC (Direct Current) for a power supply device.
- Diodes can be used to separate the signal from radio frequencies.
- Diodes can be used as an on/off switch that controls current.

This symbol —▶|— is used to indicate a diode in a circuit diagram.
The meaning of the symbol is (Anode) —▶|— (Cathode).
Current flows from the anode side to the cathode side.

Although all diodes operate with the same general principle, there are different types suited to different applications. For example, the following devices are best used for the applications noted.

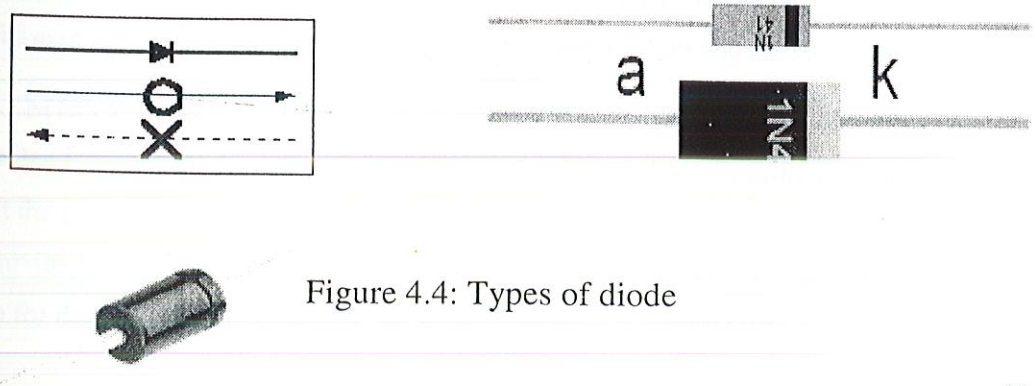


Figure 4.4: Types of diode

4.1.5 IC7805 (Series voltage regulator)

Voltage Regulator (regulator), usually having three legs, converts varying input voltage and produces a constant regulated output voltage. They are available in a variety of outputs.

The most common part numbers start with the numbers 78 or 79 and finish with two digits indicating the output voltage. The number 78 represents positive voltage and 79 negative one. The 78XX series of voltage regulators are designed for positive input. And the 79XX series is designed for negative input.

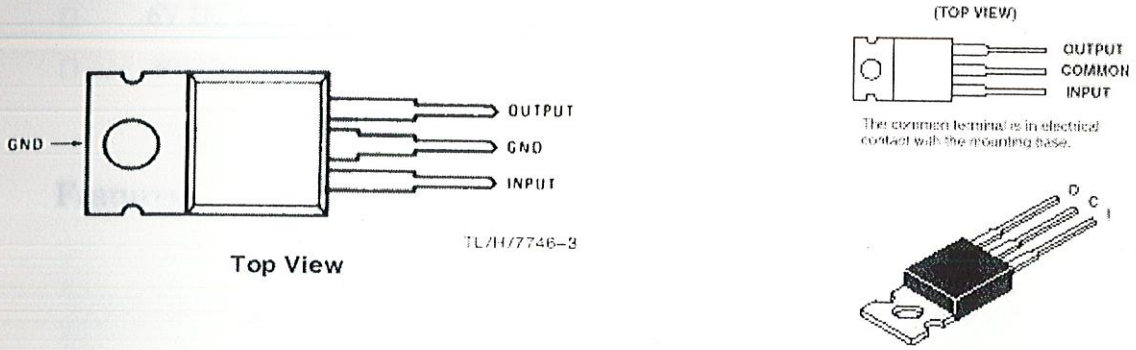


Figure 4.5: Voltage Regulator

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the

IC from overheating. Considerable effort was expanded to make the LM78XX series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

Examples:

- 5V DC Regulator Name: LM7805 or MC7805
- -5V DC Regulator Name: LM7905 or MC7905
- 6V DC Regulator Name: LM7806 or MC7806
- -9V DC Regulator Name: LM7909 or MC7909

Features

- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

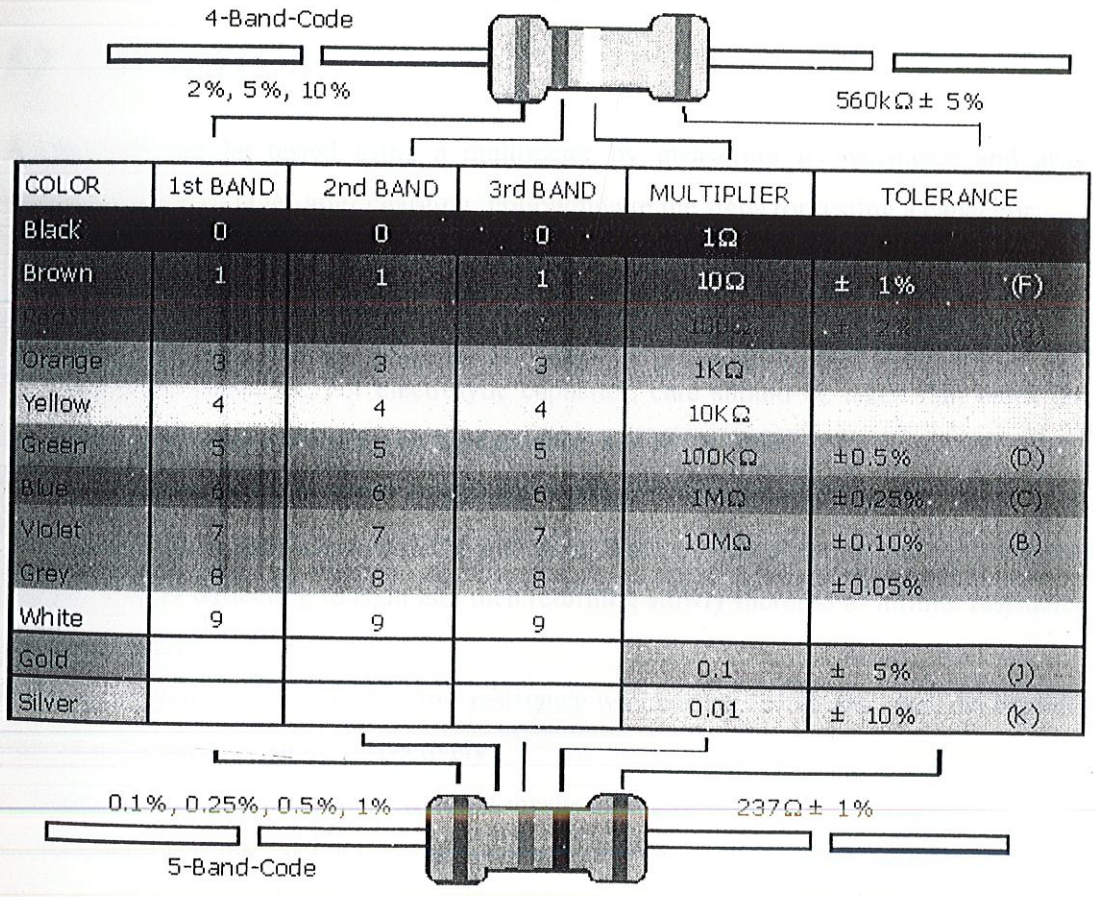
Chapter 5

COMPONENT TESTING

5.1 RESISTORS

Resistors can be easily tested using multimeter in resistance range.

The idea is to measure the resistance of the resistor and compare this measured value with the co lour coded value. If the measured value matches the co lour coded value, resistor is OK, and otherwise it is not. The steps involved in measurement of the resistance are as follows.



COLOR	1st BAND	2nd BAND	3rd BAND	MULTIPLIER	TOLERANCE
Black	0	0	0	1Ω	
Brown	1	1	1	10Ω	± 1% (F)
Red	2	2	2	100Ω	± 2% (G)
Orange	3	3	3	1KΩ	
Yellow	4	4	4	10KΩ	
Green	5	5	5	100KΩ	±0.5% (D)
Blue	6	6	6	1MΩ	±0.25% (C)
Violet	7	7	7	10MΩ	±0.10% (B)
Grey	8	8	8		±0.05%
White	9	9	9		
Gold				0.1	± 5% (J)
Silver				0.01	± 10% (K)

Switch 'ON' the multimeter and point selector switch in Ω section.

- Connect the leads to "V/Ω "and "COM "sockets.
- Select the lowest range in 'Ω' section with the selector switch.
- Connect the tips of the leads to both the leads of the resistor.
- If the resistance of the resistor is in the selected range, then the resistance value will be displayed, otherwise "over range" will be displayed.
- If "over range" is displayed, switch the selector switch to next higher value and repeat till value is displayed.
- For an open resistor, "over range" will be displayed in even the highest selected range.

The range can also be directly selected depending on the co lour code value.

5.2 CAPACITOR

A capacitor can be tested using a multimeter by measuring its resistance and also, by measuring their voltages after charging .Following are the steps for testing a capacitor.

- Set the multimeter to highest resistance range using the selector switch.
- Connect the probe tips to capacitor leads. Any lead can be connected to any lead except for the electrolytic capacitor. For electrolytic capacitor, care should be taken that black lead is connected to the lead marked (-) on the capacitor, and red to other.
- The multimeter should show increasing resistance, which increases ultimately to attain "over range" for a good capacitor. On analog multimeter, this phenomenon can be seen as – first the pointer deflecting to right and then returning slowly increase to infinite resistance for a good capacitor.
- If the multimeter shows zero or low resistance which is not increasing, then the capacitor is fully short or partially short, respectively.
- If the multimeter shows open circuit, it means capacitor is open.
- If resistance of the capacitor increases, but do not attain 'over-range' or infinite resistance, that means capacitor is leaky.

5.3 DIODE

Following are the steps for testing of a pn–junction diode:

- Select the lowest resistance range in the multimeter. Connect the leads to "V/Ω" and "COM" sockets. Red to "V/Ω" and black to "COM".
- Connect red lead to 'p' side of diode and black lead to 'n' side. The multimeter should show a low resistance. This resistance can vary from tens of ohms to few hundreds of ohms depending on the voltage in the leads. This resistance will be further less for a germanium diode.
- Now, connect red lead to 'n' side and black lead to 'p' side. The diode gets reverse biased. It should show open – circuit even in highest range.
- If the diode shows low resistance in both directions, it is "short". And if it shows "open" in both directions, it is open. If the forward bias resistances is low, and reverse bias resistance is quite less than infinity that too means diode is faulty.

Some digital multimeter have diode checking facility in them, which is marked with the help of symbol of diode on multimeter .If multimeter is switched to this position , then the multimeter should show 0.5 to 0.8V in forward bias and "open" in reverse bias .In this position , the multimeter reads the forward voltage drop across the diode . The variation from 0.5 to 0.8V is there due to variation in voltages in leads. The diode resistance is actually voltage dependent, because it is a non–ohmic device.

5.4 IR LED

The testing procedure for testing a LED is as follows:

- Select a lowest resistance range of the multimeter.
- Connect red lead of multimeter to anode of LED (longer lead) and black lead to the cathode (shorter lead) so as to forward bias the LED.
- The multimeter should read low resistance. This resistance will be higher than that of a regular diode. If the multimeter is showing "open", then switch to next higher range. This is

because the voltage in the multimeter leads may not be enough to forward bias the LED. At higher range, this voltage increases, and diode gets forward biased. When this happens, the diode also starts glowing.

- Now reverse bias the diode by connecting red lead to cathode and black lead anode. The multimeter should show "open" at all ranges, or nearly infinite resistance.

- If the multimeter has diode testing facility, then, during forward biasing, a good LED should show a forward voltage drop ranging from 1.5 volt to 3 volts. And during reverse biasing, it should show open.

Chapter 6

MICROCONTROLLER 89s52

This chapter gives a detailed description of the heart of the circuit i.e. THE MICROCONTROLLER used in our project. We have used 8051 microcontroller.

6.1 Introduction

Despite it's relatively old age, the 8051 is one of the most popular microcontrollers in use today. Many derivative microcontrollers have since been developed that are based on—and compatible with--the 8051. Thus, the ability to program an 8051 is an important skill for anyone who plans to develop products that will take advantage of microcontrollers. Many web pages, books, and tools are available for the 8051 developer. I hope the information contained in this document/web page will assist you in mastering 8051 programming. While it is not my intention that this document replaces a hardcopy book purchased at your local book store, it is entirely possible that this may be the case. It is likely that this document contains everything you will need to learn 8051 assembly language programming. Of course, this document is free and you get what you pay for so if, after reading these documents, you still are lost you may find it necessary to buy a book. This document is both a tutorial and a reference tool. The various chapters of the document will explain the 8051 step by step. The chapters are targeted at people who are attempting to learn 8051 assembly language programming. The appendices are a useful reference tool that will assist both the novice programmer as well as the experienced professional developer. This document assumes the following:

- A general knowledge of programming.
- An understanding of decimal, hexadecimal, and binary number systems.
- A general knowledge of hardware.

That is to say, no knowledge of the 8051 is assumed--however, it is assumed you've done some amount of programming before, have a basic understanding of hardware, and a firm grasp on the three numbering systems mentioned above. The concept of converting a

number from deciminal to hexadecimal and/or to binary is not within the scope of this document--and if you can't do those types of conversions there are probably some concepts that will not be completely understandable. This document attempts to address the need of the typical programmer. For example, there are certain features that are nifty and in some cases very useful--but 95% of the programmers will never use these features.

6.2 Types of Memory

The 8051 has three very general types of memory. To effectively program the 8051 it is necessary to have a basic understanding of these memory types.

1. On-Chip Memory: It refers to any memory (Code, RAM, or other) that physically exists on the microcontroller itself. On-chip memory can be of several types, but we'll get into that shortly.

2. External Code Memory: It is code (or program) memory that resides off-chip. This is often in the form of an external EPROM.

3. External RAM: It is RAM memory that resides off-chip. This is often in the form of standard static RAM or flash RAM.

6.2.1 Code Memory

Code memory is the memory that holds the actual 8051 program that is to be run. This memory is limited to 64K and comes in many shapes and sizes: Code memory may be found on-chip, either burned into the microcontroller as ROM or EPROM. Code may also be stored completely off-chip in an external ROM or, more commonly, an external EPROM. Flash RAM is also another popular method of storing a program. Various combinations of these memory types may also be used--that is to say, it is possible to have 4K of code memory on-chip and 64k of code memory off-chip in an EPROM.

When the program is stored on-chip the 64K maximum is often reduced to 4k, 8k, or 16k. This varies depending on the version of the chip that is being used. Each version offers specific capabilities and one of the distinguishing factors from chip to chip is how much ROM/EPROM space the chip has. However, code memory is most commonly

implemented as off-chip EPROM. This is especially true in low-cost development systems and in systems developed by students.

Programming Tip: Since code memory is restricted to 64K, 8051 programs are limited to 64K. Some assemblers and compilers offer ways to get around this limit when used with specially wired hardware. However, without such special compilers and hardware, programs are limited to 64K.

6.2.2 External RAM

As an obvious opposite of Internal RAM, the 8051 also supports what is called External RAM. As the name suggests, External RAM is any random access memory which is found off-chip. Since the memory is off-chip it is not as flexible in terms of accessing, and is also slower. For example, to increment an Internal RAM location by 1 requires only 1 instruction and 1 instruction cycle. To increment a 1-byte value stored in External RAM requires 4 instructions and 7 instruction cycles. In this case, external memory is 7 times slower! What External RAM loses in speed and flexibility it gains in quantity. While Internal RAM is limited to 128 bytes the 8051 supports External RAM up to 64K.

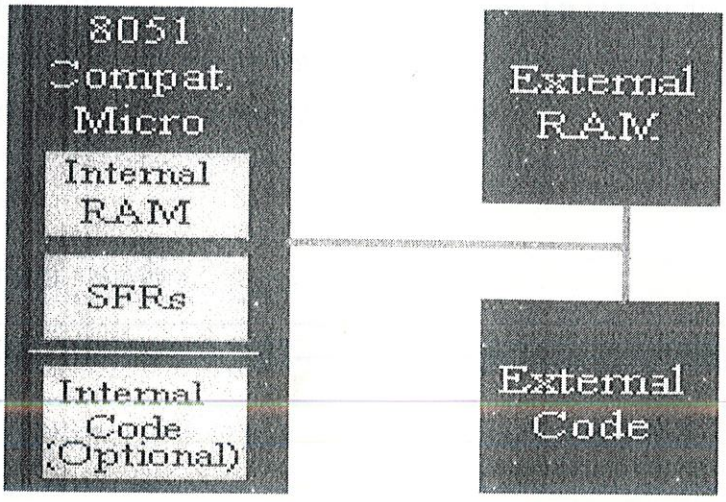


Figure 6.1: External RAM

Programming Tip: The 8051 may only address 64k of RAM. To expand RAM beyond this limit requires programming and hardware tricks. You may have to do this "by hand" since many compilers and assemblers, while providing support for programs in excess of 64k, do not support more than 64k of RAM. This is rather strange since it has been my experience that programs can usually fit in 64k but often RAM is what is lacking. Thus if you need more than 64k of RAM, check to see if your compiler supports it-- but if it doesn't, be prepared to do it by hand.

6.2.3 On-Chip Memory

The 8051 includes a certain amount of on-chip memory. On-chip memory is really one of two memories. As is illustrated in this map, the 8051 has a bank of 128 bytes of Internal RAM. This Internal RAM is found on-chip on the 8051 so it is the fastest RAM available, and it is also the most flexible in terms of reading, writing, and modifying its contents. Internal RAM is volatile, so when the 8051 is reset this memory is cleared. The 128 bytes of internal ram is subdivided as shown on the memory map. The first 8 bytes (00h - 07h) are "register bank 0". By manipulating certain SFRs, a program may choose to use register banks 1, 2, or 3. These alternative register banks are located in internal RAM in addresses 08h through 1Fh. For now it is sufficient to know that they "live" and are part of internal RAM. Bit Memory also lives and is part of internal RAM. We'll talk more about bit memory very shortly, but for now just keep in mind that bit memory actually resides in internal RAM, from addresses 20h through 2Fh.

6.3 Addressing Modes

An Addressing mode refers to how you are addressing a given memory location.

In summary, the addressing modes are as follows, with an example of each:

Immediate Addressing	**MOV A,#20h**
Direct Addressing	**MOV A,30h**
Indirect Addressing	**MOV A,@Ro**
External Direct	**MOVX A,@DPTR**
Code Indirect	**MOVC A,@A+DPTR**

Table 6.1: Addressing Modes

1. **Immediate Addressing**: It is so named because the value to be stored in memory immediately follows the operation code in memory. The instruction itself dictates what value will be stored in memory.
   Mov A, #20h
   This instruction uses Immediate Addressing because the Accumulator will be loaded with the value that immediately follows; in this case 20 (hexadecimal). Immediate addressing is very fast since the value to be loaded is included in the instruction. However, since the value to be loaded is fixed at compile-time it is not very flexible.

2. **Direct Addressing**: It is so-named because the value to be stored in memory is obtained by directly retrieving it from another memory location.
   Mov A, 30h
   This instruction will read the data out of Internal RAM address 30 (hexadecimal) and store it in the Accumulator it is fast, although the value to be loaded isn't included in the instruction, it is important to note that when using direct addressing any instruction which refers to an address between 00h and 7Fh is referring to Internal Memory. Any instruction which refers to an address between 80h and FFh is referring to the SFR control registers that control the microcontroller itself.

3. **Indirect Addressing**: Indirect addressing is a very powerful addressing mode which in many cases provides an exceptional level of flexibility.
   Indirect addressing is also the only way to access the extra 128 bytes of Internal RAM found on an 8052. Indirect addressing appears as follows:
   Mov A, R0
   This instruction causes the 8051 to analyze the value of R0 register. The 8051 will then load the Accumulator with the value from Internal RAM which is found at the address indicated by R0. It never refers to SFR but Internal RAM.
   MOV R0, #99h; Load the address of the serial port.

4. **External Direct**: External Memory is accessed using a suite of instructions which use "External Direct" addressing. It's called so as it appears to be direct addressing, but it is use51d to access external memory rather than internal memory.
   MOVX A, DPTR

5. **External Indirect**: External memory can also be accessed using a form of indirect addressing called as External Indirect addressing. This form of addressing is usually used in relatively small projects that have a very small amount of external RAM.
   An example of this addressing mode is:
   MOVX R0, A

6.4 Microcontroller AT89s52

It is an 8-bit microcontroller with 4k bytes. In System Programmable flash memory. It can be both serially and parallel programmed. The details are discussed below.

6.4.1 Features:-

1. Compatible with MCS-51 Products
2. 4k Bytes of In-System Programmable (ISP) Flash Memory
3. Endurance: 1000 write/Erase Cycles
4. Fully tactic Operation: 0 Hz to 33MHZ
5. Three-level Program Memory Lock
6. 128 * 8-bit Internal RAM
7. 32 Programmable I/O lines
8. Two 16-bit Timer/Counters
9. Six Interrupt Sources
10. Full Duplex UART Serial Channel
11. Low-Power Idle and Power-down Modes
12. Interrupt Recovery from Power-down Mode

13. Watchdog Timer

14. Dual Data Pointer

15. Power-off Flag

16. Fast Programming Time

17. Flexible ISP Programming (Byte and Page Mode)

6.4.2 Description

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 4K bytes of In-System Programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry standard 80C51 instruction set and pin out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications. The AT89S52 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

6.4.3 Pin Description

- **VCC:** Supply voltage (all packages except 42-PDIP).

- **GND:** Ground (all packages except 42-PDIP: for 42-PDIP GND connects only the logic core and the embedded program memory).

- **VDD:** Supply voltage for the 42-PDIP which connects only the logic core and the embedded program memory.

- **PWRVDD:** Supply voltage for the 42-PDIP which connects only the I/O Pad Drivers. The application board must connect both VDD and PWRVDD to the board supply voltage.

- **PWRGND:** Ground for the 42-PDIP which connects only the I/O Pad Drivers. PWRGND and GND are weakly connected through the common silicon substrate, but not through any metal link. The application board must connect both GND and PWRGND to the board ground.

- **Port 0:** Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs. Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pull-ups. Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pull-ups are required during program verification.

- **Port 1:** Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs.

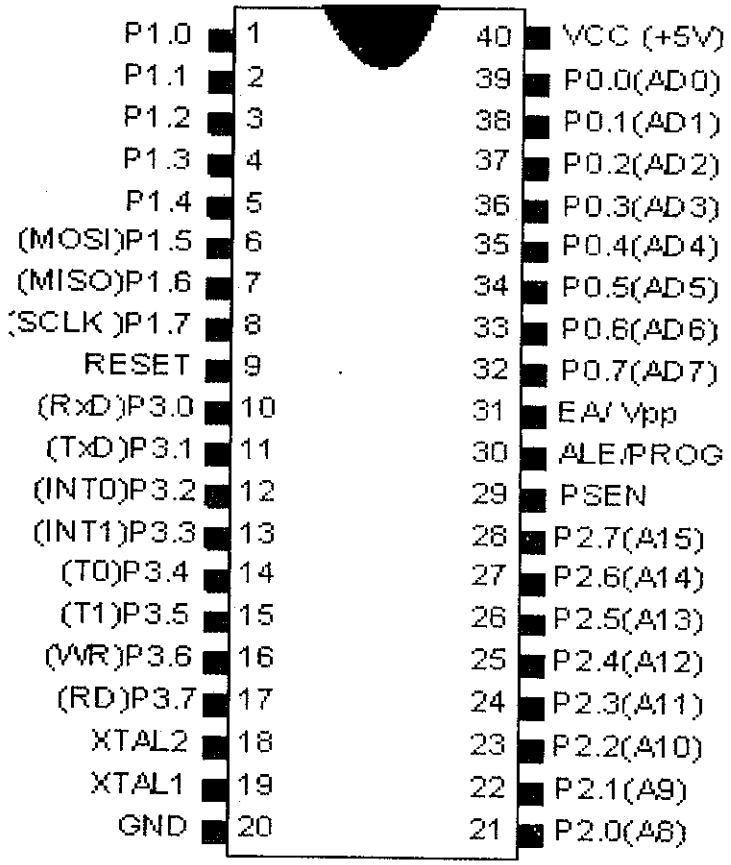


Figure 6.2: PIN Diagram 89s52

Port Pin	Alternate Function
P1.5	MOSI (used for In-system Programming)
P1.6	MISO (used for In-system Programming)
P1.7	SCK (used for In-system Programming)

Table 6.2: Port 1 Functions

- The internal pull-ups. Port 1 also receives the low-order address bytes during Flash programming and verification.

- **Port 2:** Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (IIL) because of the internal pull-ups. Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that uses 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that uses 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

- **Port 3:** Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (IIL) because of the pull-ups. Port 3 receives some control signals for Flash

programming and verification. Port 3 also serves the functions of various special features of the AT89S52, as shown in the following table.

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

Table 6.3: Port 3 Functions

- **RST:** Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives High for 98 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.
- **ALE/PROG:** Address Latch Enable (ALE) is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during Flash programming. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory. If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

- **PSEN:** Program Store Enable (PSEN) is the read strobe to external program memory. When the AT89S52 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.
- **EA/VPP:** External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, EA will be internally latched on reset. EA should be strapped to VCC for internal program executions. This pin also receives the 12-volt programming enable voltage (VPP) during Flash programming.
- **XTAL1:** Input to the inverting oscillator amplifier and input to the internal clock operating circuit.
- **XTAL2:** Output from the inverting oscillator amplifier

Chapter 7

SOLDERING

7.1 Introduction

Soldering is a process in which two or more metal items are joined together by melting and flowing a filler metal into the joint, the filler metal having a relatively low melting point. Soft soldering is characterized by the melting point of the filler metal, which is below 400 °C (800 °F). The filler metal used in the process is called solder.

Soldering is distinguished from brazing by use of a lower melting-temperature filler metal; it is distinguished from welding by the base metals not being melted during the joining process. In a soldering process, heat is applied to the parts to be joined, causing the solder to melt and be drawn into the joint by capillary action and to bond to the materials to be joined by wetting action. After the metal cools, the resulting joints are not as strong as the base metal, but have adequate strength, electrical conductivity, and water-tightness for many uses. Soldering is an ancient technique mentioned in the Bible and there is evidence that it was employed up to 5000 years ago in Mesopotamia.

7.2 Applications

One of the most frequent applications of soldering is assembling electronic components to printed circuit boards (PCBs). Another common application is making permanent but reversible connections between copper pipes in plumbing systems. Joints in sheet metal objects such as food cans, roof flashing, rain gutters and automobile radiators have also historically been soldered, and occasionally still are. Jewelry components are assembled and repaired by soldering. Small mechanical parts are often soldered as well. Soldering is also used to join lead came and copper foil in stained glass work. Soldering can also be used to effect a semi-permanent patch for a leak in a container cooking vessel.

7.3 Soldering Techniques

Bad solder joints are often the cause of annoying intermittent faults. They can often be hard to find a cause circuit failure at the most inappropriate time. It's much better to learn to make a good solder joints from day one.

- Preparing the soldering iron:
- Wipe the tip clean on the wetted sponge provided.
- Bring the resin cored solder to the iron and 'tin' the tip of the iron.
- Wipe the excess solder of the tip using the wet sponge.

Repeat until the tip is properly 'tinned'.

Soldering components into PCB

- Bend the component leads at right angles with both bends at the same distance apart as the PCB pad holes.
- Ensure that both component leads and the copper PCB pads are clean and free of oxidization.
- Insert component leads into holes and bend leads at about 30 degrees from vertical.
- Using small angle cutters, cut the leads at about 0.1 - 0.2 of an inch (about 2 - 4 mm) above copper pad.
- Bring tinned soldering iron tip into contact with both the component lead and the PCB pad. This ensures that both surfaces undergo the same temperature rise.
- Bring resin cored solder in contact with the lead and the copper pad. Feed just enough solder to flow freely over the pad and the lead without a 'blobbing' effect. The final solder joint should be shiny and concave indicating good 'wetting' of both the copper pad and the component lead. If a crack appears at the solder to metal interface then the potential for forming a dry joint exists. If an unsatisfactory joint is formed, suck all the solder off the joint using a solder sucker or solder wick (braid) and start again.

7.4 Precautions

1. Mount the components at the apron places before soldering. Follow the circuit description and components details, leads identification etc. Do not start soldering before making it confirm that all the components are mounted at the right place.

2. Do not use a spread solder on the board, it may cause short circuit.

3. Do not sit under the fan while soldering.

4. Position the board so that gravity tends to keep the solder where you want it.

5. Do not over heat the components at the board. Excess heat may damage the components or board.

6. The board should not vibrate while soldering otherwise you have a dry or a cold joint.

7. Do not put the kit under or over voltage source. Be sire about the voltage either is D.C. or a.c. while operating the gadget.

8. Do spare the bare ends of the components leads otherwise it may short circuit with the other components. To prevent this use sleeves at the component leads or use sleeved wire for connections.

9. Do not use old dark colour solder. It may give dry joint. Be sure that all the joints are clean and well shiny.

10. Do make loose wire connections especially with cell holder, speaker, probes etc. Put knots while connections to the circuit board, otherwise it may get loose.

Chapter 8

ASSEMBLY SOURCE FOR µC

8.1 Software Requirement

The software platform used for the programming of the microcontroller 89s51 of the 8051 family is KEIL. With this software, assembly language programming can be actualized.

8.1.1 Introduction to Keil

The Keil C51 C Compiler for the 8051 microcontroller is the most popular 8051 C compiler in the world. It provides more features than any other 8051 C compiler available today. The C51 Compiler allows you to write 8051 microcontroller applications in C that have the efficiency and speed of assembly language. Language extensions in the C51 Compiler give you full access to all resources of the 8051. C51 translates C source files into a relocatable object module. When the DEBUG control is used, the object file contains full symbolic information for debugging with the µVision3 Debugger or an in-circuit emulator. In addition to the object file, the C51 Compiler generates a listing file which optionally may include symbol table and cross-reference information.

Keil was founded in 1986 to market add-on products for the development tools provided by many of the silicon vendors. Keil implemented the first C compiler designed from the ground-up specifically for the 8051 microcontroller. Keil provides a broad range of development tools like ANSI C compiler, macro assemblers, debuggers and simulators, linkers, IDE, library managers, real-time operating systems and evaluation boards for 8051, 251, ARM and XC16x/C16x/ST10 families. In October 2005, Keil (Keil Elektronik GmbH in Munich, Germany and Keil Software, Inc. in Plano, Texas) was acquired by ARM.

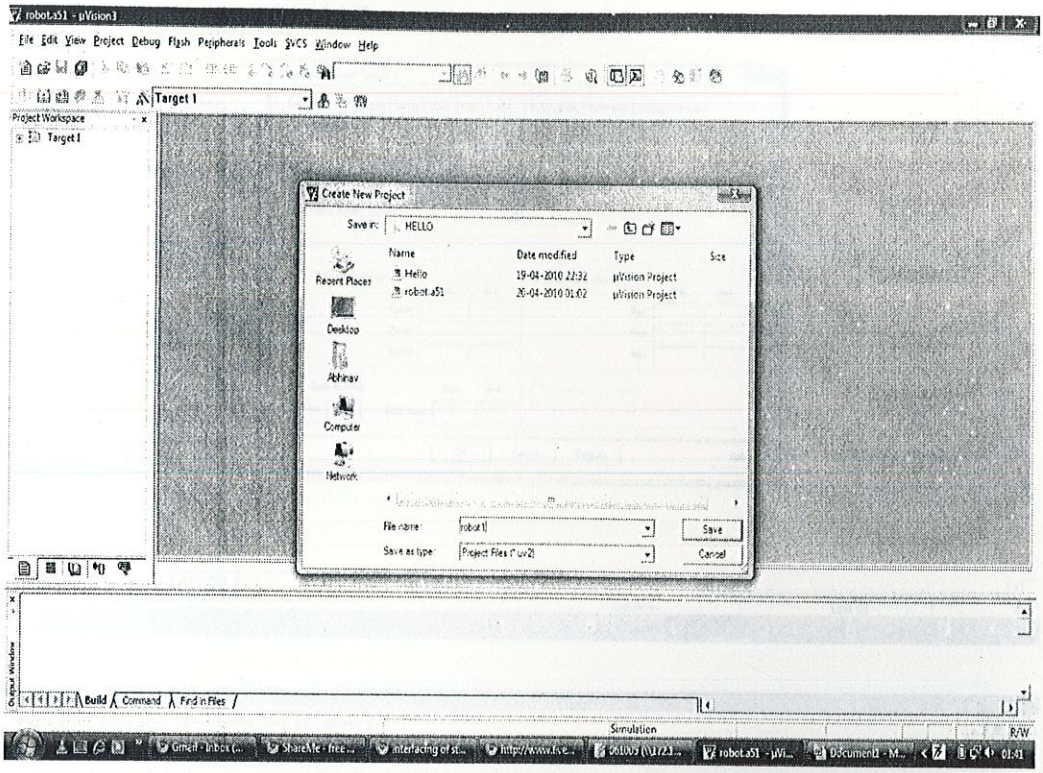
Keil development tools for the 8051 Microcontroller Architecture support every level of software developer from the professional applications engineer to the student just

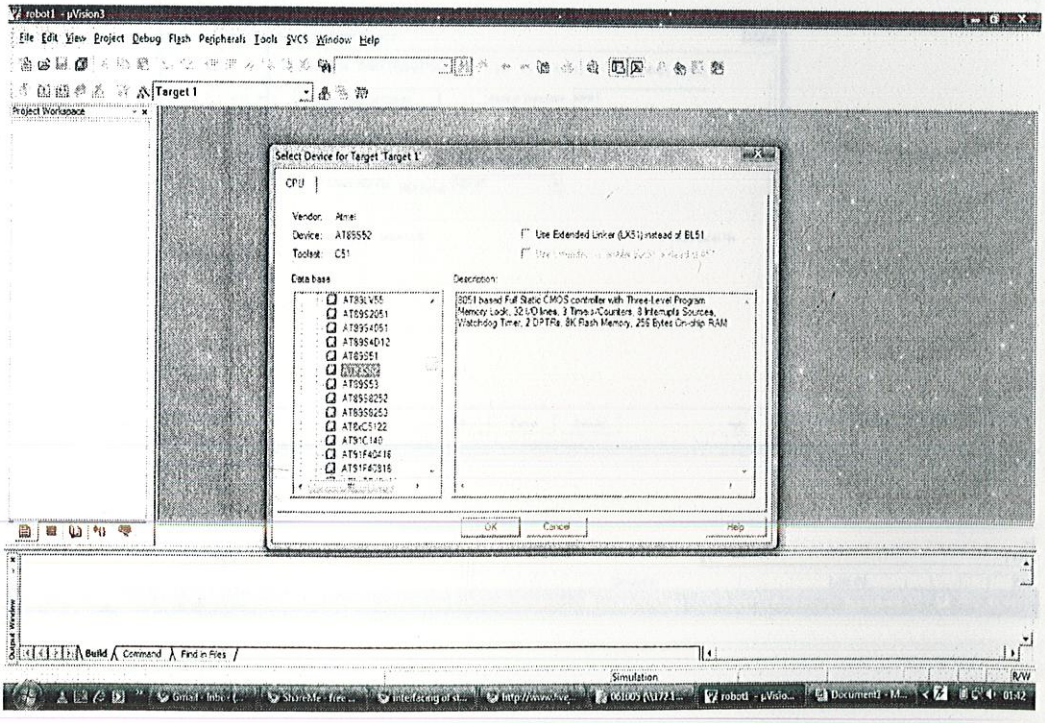
learning about embedded software development. The industry –standard Keil C compilers, Macro Assemblers, Debuggers, Real-time Kernels, Single-board Computers, and Emulators support all 8051 derivatives and help to get the projects completed on schedule. The Keil 8051 Development Tools are designed to solve the complex problems facing embedded software developers.

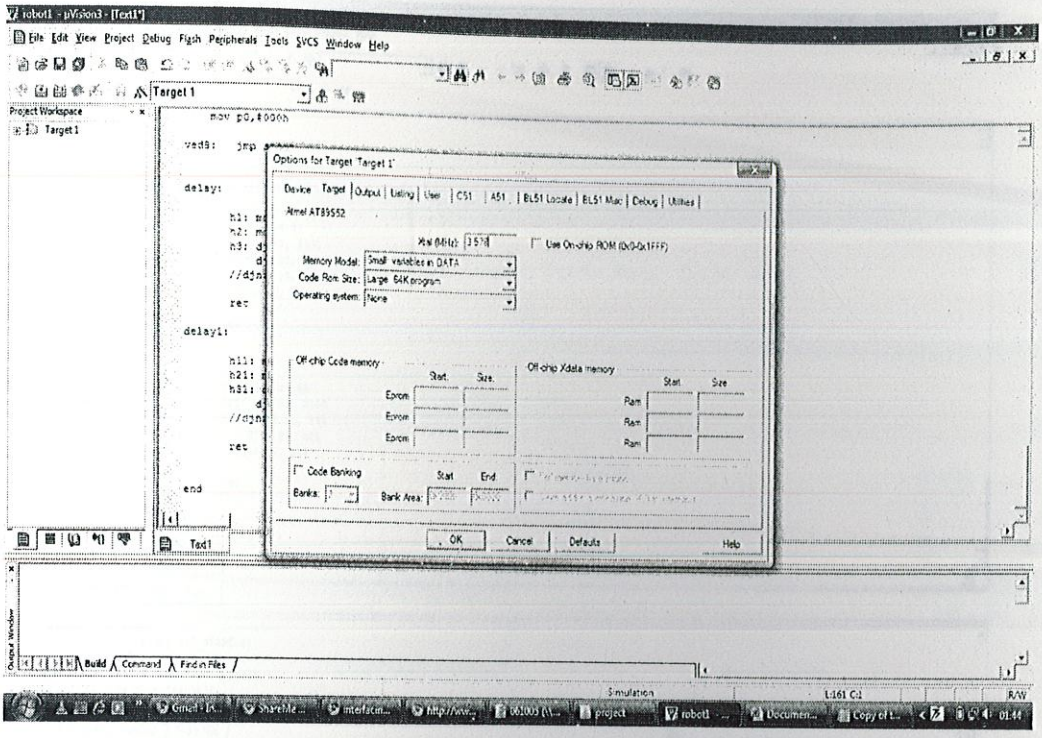
Features

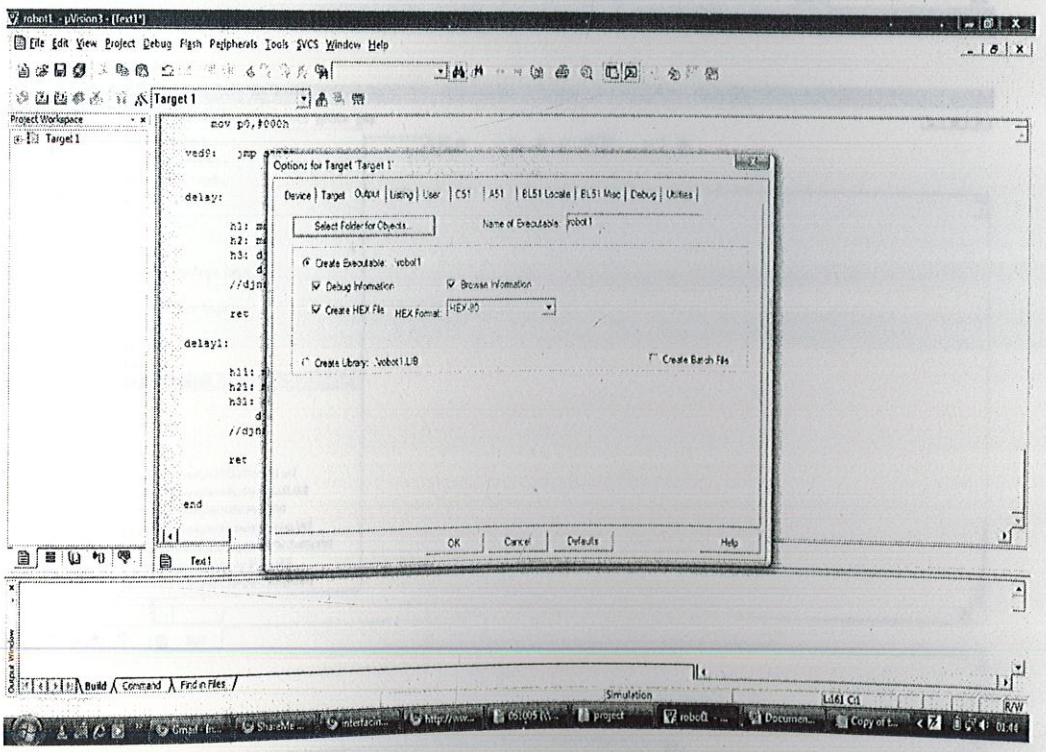
- Nine basic data types, including 32-bit IEEE floating-point
- Interrupt functions may be written in C
- Full use of the 8051 register banks
- Complete symbol and type information for source-level debugging
- Use of AJMP and ACALL instructions
- Bit-addressable data objects
- Support for the Philips 8xC750, 8xC751, and 8xC752 limited instruction sets
- Support for the Infineon 80C517 arithmetic unit

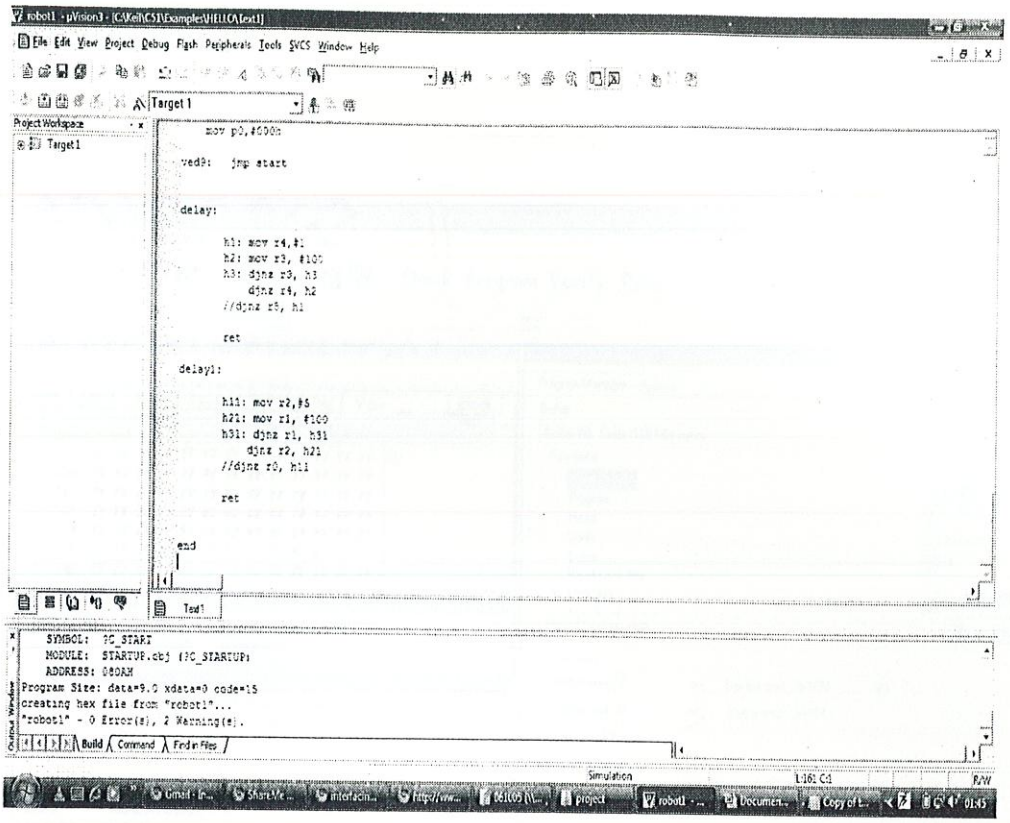
8.3 Conversion of assembly language into HEX file

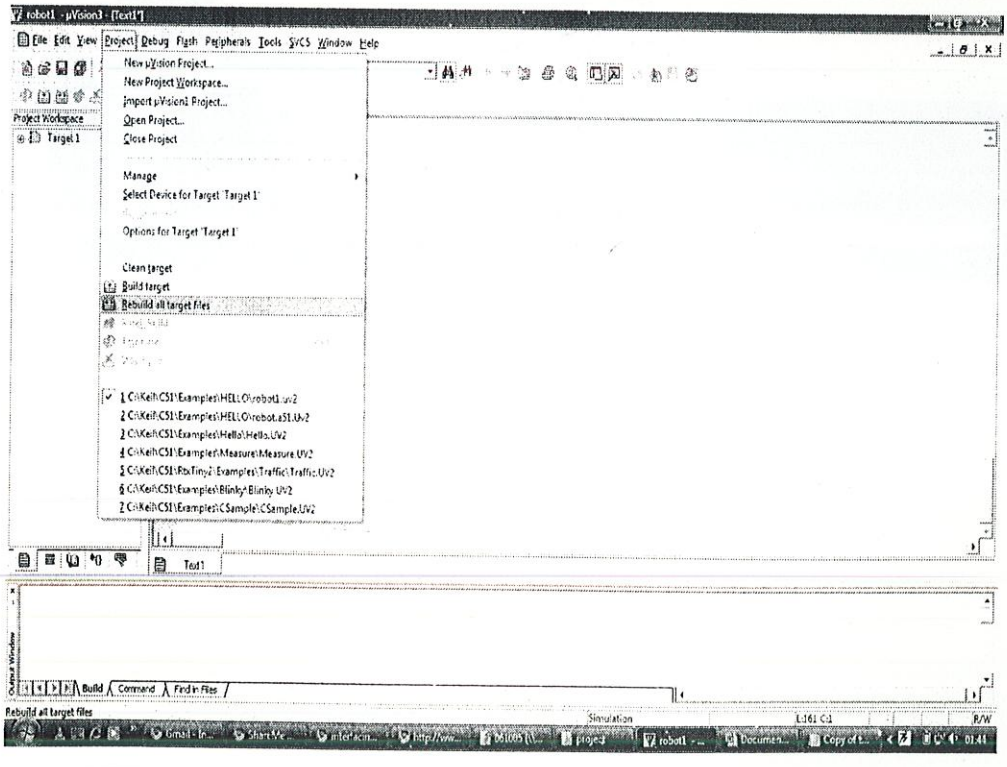




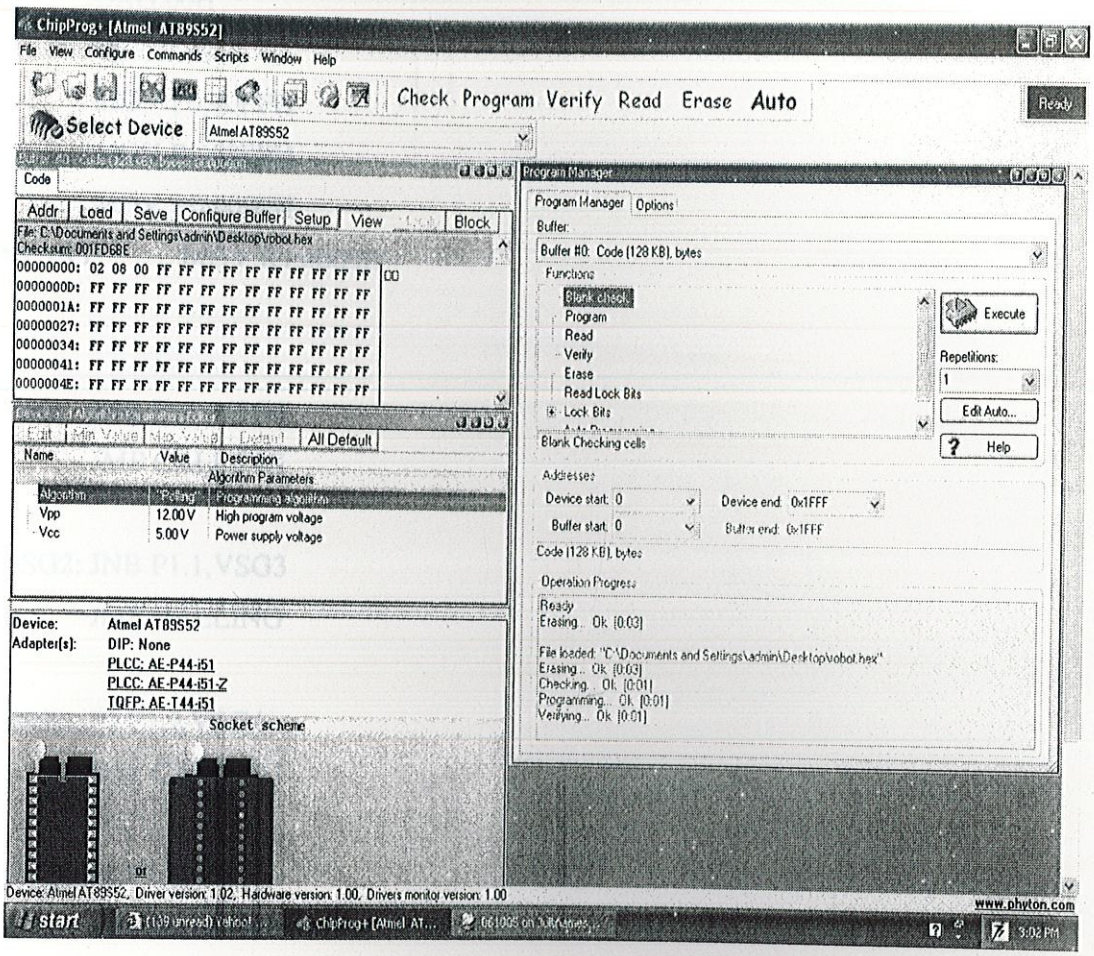

robot1 - µVision3 - [Text1*]
File Edit View Project Debug Flash Peripherals Tools SVCS Window Help
Target 1
Project Workspace
Target 1
mov p0,#000h
vad9: jmp
delay:
ret
delay1:
ret
end
Options for Target 'Target 1'
Device Target Output Listing User C51 A51 BL51 Locate BL51 Misc Debug Utilities
Atmel AT89S52
Xtal (MHz): 35.0
Use On-chip ROM (0x0-0x1FFF)
Memory Model: Small: variables in DATA
Code Rom Size: Large: 64K program
Operating system: None
Off-chip Code memory
Off-chip Xdata memory
Start Size
Eprom
Ram
Code Banking
Start End
Banks:
Bank Area:
OK Cancel Defaults Help
Text1
Build Command Find in Files
Simulation
L:161 C:1
R/W


robot1 - µVision3 - [Text1*]
File Edit View Project Debug Flash Peripherals Tools SVCS Window Help
Target 1
Project Workspace
Target 1
mov p0,#000h
vad9: jmp
delay:
ret
delay1:
ret
end
Options for Target 'Target 1'
Device Target Output Listing User C51 A51 BL51 Locate BL51 Misc Debug Utilities
Select Folder for Objects...
Name of Executable: robot1
Create Executable: robot1
Debug Information
Browse Information
Create HEX File
HEX Format: HEX-80
Create Library: .\robot1.LIB
Create Batch File
OK Cancel Defaults Help
Text1
Build Command Find in Files
Simulation
L:161 C:1
R/W


robot1 - µVision3 - [C:\Keil\C51\Examples\HELLO\Text1]
File Edit View Project Debug Flash Peripherals Tools SVCS Window Help
Target 1
Project Workspace
Target1
mov p0,#000h
wed9: jmp start
delay:
h1: mov r4,#1
h2: mov r3, #10
h3: djnz r3, h3
djnz r4, h2
//djnz r5, h1
ret
delay1:
h11: mov r2,#5
h21: mov r1, #100
h31: djnz r1, h31
djnz r2, h21
//djnz r0, h11
ret
end
Text1
SYMBOL: ?C_START
MODULE: STARTUP.obj (?C_STARTUP)
ADDRESS: 000RM
Program Size: data=9.0 xdata=0 code=15
creating hex file from "robot1"...
"robot1" - 0 Error(s), 2 Warning(s).
Build Command Find in Files
Simulation
L:161 C:1
R/W


robot1 - µVision3 - [Text1]
File Edit View Project Debug Flash Peripherals Tools SVCS Window Help
New µVision Project...
New Project Workspace...
Import µVision1 Project...
Open Project...
Close Project
Manage
Select Device for Target 'Target 1'
Options for Target 'Target 1'
Clean target
Build target
Rebuild all target files
1 C:\Keil\C51\Examples\HELLO\robot1.uv2
2 C:\Keil\C51\Examples\HELLO\robot.a51.Uv2
3 C:\Keil\C51\Examples\Hello\Hello.Uv2
4 C:\Keil\C51\Examples\Measure\Measure.Uv2
5 C:\Keil\C51\RxTiny2 Examples\Traffic\Traffic.Uv2
6 C:\Keil\C51\Examples\Blinky\Blinky.Uv2
7 C:\Keil\C51\Examples\CSample\CSample.Uv2
Text1
Build Command Find in Files
Rebuild all target files
Simulation
L:161 C:1
R/W

Burning of code



8.5 Source Code

```
        ORG 00H
        CLR P2.0
        MOV P0,#0FFH
        MOV P1,#0FFH
        MOV R0,#096H
START:  MOV R1,#003H

VSG:  JNB P1.0,VSG2
      JMP CALLING

VSG2: JNB P1.1,VSG3
      JMP CALLING

VSG3: JB P1.2,VSG4
      JMP CALLING

VSG4: JB P1.3,VSG5
      JMP CALLING

VSG5: JMP VSG


CALLING:

      SETB P2.0
```

```
        CALL DELAY
        CALL DELAY

        CLR P0.0
        CALL DELAY
        SETB P0.0
        CALL DELAY

        CLR P0.0
        CALL DELAY
        SETB P0.0
        CALL DELAY


XXX1:DJNZ R0,XXX
        JMP XX

XXX:  CALL DELAY

JMP XXX1


XX:    DJNZ R1,CALLING


JMP    START



delay:
```

```
H1: MOV R4,#10
H2: MOV R3, #100
H3: DJNZ R3, H3
    DJNZ R4, H2
        DJNZ R5, H1
    •
    RET


END
```

CONCLUSION

Working on a project that will be of great help to the security for vehicles has been interesting, challenging and a thrilling experience. We have developed a low cost and technologically efficient system comprising the following feature:

1. MICROCONTROLLER - The controller will check the IC as per the truth table of the IC which is stored in its ROM.

2. CELL PHONE - The cell phone we are using for this project is based on the GSM technology having uplink frequency (890-915MHz) and downlink frequency (935-960MHz).

3. OPTO-ISOLATOR - A device that uses a short optical transmission path to transfer an electronic signal between elements of a circuit, typically a transmitter and a receiver, while keeping them electrically isolated

4. ASSEMBLY SOURCE FOR MICROCONTROLLER - The software platform used for the programming of the microcontroller 89s52 of the 8051 family is KEIL. With this software, assembly language programming has been actualized.

Our final designed system is advantageous over other similar systems because:

1. The system moves a step ahead of other current security systems available in the market such as AUTOCOP etc. Our security system has real time application; i.e. the security system doesn't merely create noise but informs the owner wherever the person may be.
2. Cost effective.
3. It is sellable. Any product designed must be lucrative for consumers to buy. Such a smart system has the potential to be sold and attract customers.

The project has various future applications and can still be modified with certain new features which can be cost effective when manufactured at large scale. Also, our designed system is flexible and can accommodate new technologies such as CDMA and other code encrypted highly sophisticated techniques to further enhance the efficacy.

BIBLIOGRAPHY

www.electrogugs.com

www.robotics.com

www.dataarchive.com

www.wikipedia.com

www.google.com

www.atmel.com

www.allegromicro.com