GSM BASED LPG LEAKAGE DETECTION USING MICROCONTROLLER

Thesis Submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

RAJAT RANA (121014)

PRASHANT BHARDWAJ (121017)

MUNISH SINGH (121026)

UNDER THE GUIDANCE OF

Dr. NEERU SHARMA

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

May, 2016

TABLE OF CONTENTS

DECLARATION

I hereby declare that the work reported in the B-Tech thesis entitled **"GSM BASED LPG LEAKAGE DETECTOR USING MICROCONTROLLER"** submitted at **Jaypee University of Information Technology, Waknaghat, India,** is an authentic record of my work carried out under the supervision of **Dr. Neeru Sharma**. I have not submitted this work elsewhere for any other degree or diploma.

Rajat Rana **Prashant Bhardwaj** **Munish Singh**

Department of Electronics and Communication Engineering

Jaypee University of Information Technology, Waknaghat, India

26th May, 2016

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY
WAKNAGHAT
SOLAN, HIMACHAL PRADESH

Date: 26th May 2016

CERTIFICATE

This is to certify that the work reported in the B-Tech. thesis entitled ***"GSM BASED LPG LEAKAGE DETECTION USING MICROCONTROLLER"***, submitted by **Rajat Rana, Prashant Bhardwaj and Munish Singh** at **Jaypee University of Information Technology, Waknaghat**, **India,** is a bonafide record of their original work carried out under my supervision. This work has not been submitted partially or wholly to any other university or institution for award of this or any other degree program.

Dr. Neeru Sharma

Assistant Professor

Electronics and Communication Engineering
JUIT Waknaghat

ACKNOWLEDGEMENT

We are highly indebted to **Dr. Neeru Sharma** for motivating and enlightening us for our project work .We thank you for being a constant support throughout, without whose valuable guidance and insights, this project would not be a complete one. We offer you our sincere gratitude for instructing and directing us through thick and thin.

We would also like to acknowledge **Mr. Mohan Sharma** for helping us in the labs. Thank you for being there whenever needed.

LIST OF FIGURES

ACRONYMS

ADC	Analog to Digital Convertor
AT	Attention
DAC	Digital to Analog Convertor
GSM	Global System for Mobile Communication
ISP	In System Programmable
LPG	Liquefied Petroleum Gas
SPI	Serial Peripheral Interface
TWI	Two Wire Interface
USART	Universal Synchronous and Asynchronous Receiver and Transmitter

ABSTRACT

The project is about the detection of LPG using a microcontroller, a sensor and GSM module. If there is a leakage nearby the sensor detects it and conveys the message to the microcontroller, which in turn controls all other required steps to be taken. It further conveys the message to different units like exhaust fan, GSM module. Exhaust fan will eject out the gas from the area. GSM module, after getting the leakage news will send a message to the user whose mobile number has been registered. And this way, the person can come to know about the leakage even if he or she is not at the location and take the necessary steps. The sensor can be calibrated according to the need of the user, and starts to detect the leakage at some particular threshold value.

CHAPTER 1

INTRODUCTION

While deciding about the major project, the main consideration was to make something that would be practically useful and could be utilized in our daily lives as well as various diverse fields. Theft has increased a lot in today's world and not everyone can afford costly detection system. We have tried to make cheap and effective detection system. Today cell phone has become an integral part of our life and through this security system we can get security alert on our mobile phone in case of any incident say LPG leak, theft, fire irrespective of where we are.

LPG consists of mixture of propane and butane which is highly flammable chemical. It is odourless gas due to which Ethanethoil is added as powerful odorant, so that leakage can be easily detected. There are other international standards like EN589, amyl mercaptane and tetrahydrothiophene which are most commonly used as odorants. LPG is one of the alternate fuels used now days. Sometimes liquefied petroleum gas is also known as LPG, LP gas, Auto gas etc. This gas is commonly used for heating appliances, hot water, cooking, and various other purposes also. LPG is also used as an alternate fuel in vehicles due to soaring in the prices of petrol and diesel. Some people have low sense of smell, may or may not respond on low concentration of gas leakage. In such a case, gas leakage security systems become an essential and help to protect from gas leakage accidents. A number of research papers have been published on gas leakage security system [1][2][3]. Embedded system for Hazardous gas detection and Alerting has been proposed in literature. Where the alarm is activate immediately, if the gas concentration exceeds normal level. Bhopal gas tragedy was an example of gas leakage accident in India. This was world's worst gas leakage industrial accident. Gas leakage detection is not only important but stopping leakage is equally essential. This project provides a cost effective and highly accurate system, which not only detect gas leakage but also alert (Beep) and turn off main power and gas supplies, and send an SMS. GSM module is used which alert the user by sending an SMS. In order to provide high accuracy gas sensor MQ-5 has been used.

For better understanding of heat flux sensor behavior, it can be modeled as a simple electrical circuit consisting of a resistance, *R*, and a capacitor, *C*. In this way it can be seen that one can

attribute a thermal resistance R_{sen}, a thermal capacity C_{sen} and also a response time τ_{sen} to the sensor. Gas sensors interact with a gas to initiate the measurement of its concentration. The gas sensor then provides output to a gas instrument to display the measurements.

There are mainly three units, in this circuit: sensor unit, microcontroller unit and GSM modem. For detecting dangerous & flammable gas leaks in any closed environment such as a car, house, service station or storage tank, a gas sensor is used which detects natural gas, LPG and coal gas. This sensor can also be used to sense other gases like iso-butane, propane and even cigarette smoke. This unit can easily be incorporated into an alarm unit to sound an alarm. GSM modem can be configured by standard GSM AT command set for sending and receiving SMS and getting modem status. Depending upon the gas sensor output, the microcontroller can send message to the authorized person.

CHAPTER 2

HARDWARE DESCRIPTION

2.1 MQ5 SEMICONDUCTOR SENSOR

Ideal sensor for use to detect the presence of a dangerous LPG leak in your car or in a service station, storage tank environment. This unit can be easily incorporated into an alarm unit, to sound an alarm or give a visual indication of the LPG concentration. The sensor has excellent sensitivity combined with a quick response time. The sensor can also sense iso-butane, propane, LNG and cigarette smoke. MQ-5 semiconductor sensor is Combustible Gas Sensitive. The MQ-5 gas sensor is made up of SnO2 which has lower conductivity in clean air. A simple electro-circuit is used here which is used to convert the changing conductivity into corresponding output signal of gas concentration. Both Methane and Propane can be detected easily by MQ-5 sensor because it has high sensitivity towards Methane, Propane and Butane. It is a low cost sensor suitable for different application. This sensor is used to sense the leakage of LPG. In normal conditions the output of this sensor is ‘high’ and it goes ‘low’, when the LPG is sensed.

Figure2.1: MQ-5 gas sensor module.

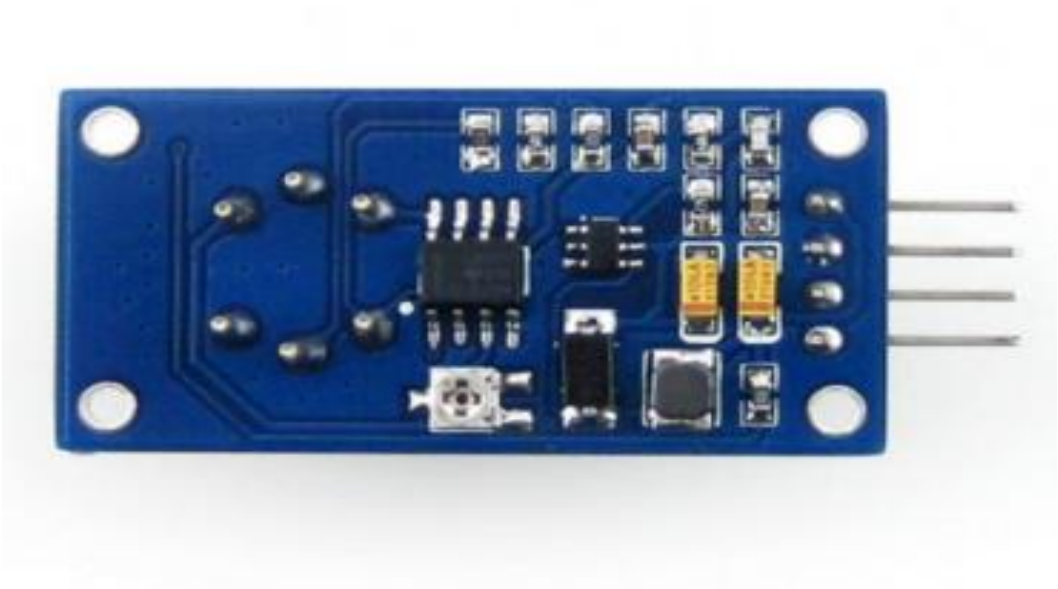


Figure2.2: Backside view of MQ-5 gas sensor module.

2.1.1 Overview

- Sensitive for LPG, natural gas, coal gas
- Output voltage boosts along with the concentration of the measured gases increases
- Fast response and recovery
- Adjustable sensitivity
- Signal output indicator

2.1.2 Specifications

- Power: 2.5V ~ 5.0V
- Dimension: 40.0mm * 21.0mm
- Mounting holes size: 2.0mm

2.1.3 Applications

- Domestic gas leakage detector
- Industrial Combustible gas detector
- Portable gas detector.

2.2 OPERATIONAL AMPLIFIER

An operational amplifier, often called an op-amp, is a DC-coupled high-gain electronic voltage amplifier with differential inputs and, usually, a single output. Typically the output of the op-amp is controlled either by negative feedback, which largely determines the magnitude of its output voltage gain, or by positive feedback, which facilitates regenerative gain and oscillation. High input impedance at the input terminals and low output impedance are important typical characteristics.

2.3 MICROCONTROLLER

It was electricity in the beginning that people were happy because they did not know that it was all around them and could be utilized. That was good. Then Faraday came and a stone has started to roll slowly.

The first machines using a new sort of energy appeared soon. A long time has passed since then and just when the people finally got used to them and stopped paying attention to what a new generation of specialists were doing, someone came to an idea that electrons could be a very convenient toy being closed in a glass pipe. It was just a good idea at first, but there was no return. Electronics was born and the stone kept on rolling down the hill faster and faster...

A new science - new specialists. Blue coats were replaced with white ones and people who knew something about electronics appeared on the stage. While the rest of humanity were passively watching in disbelief what was going on, the plotters split in two groups - “software-oriented” and “hardware-oriented”. Somewhat younger than their teachers, very enthusiastic and full of ideas, both of them kept on working but separate ways. While the first group was developing constantly and gradually, the hardware-oriented people, driven by success, threw caution to the wind and invented transistors.

Up till that moment, the things could be more or less kept under control, but a broad publicity was not aware of what was going on, which soon led to a fatal mistake! Being naive in belief that cheap tricks could slow down technology development and development of the world and retrieve the good all days, mass market opened its doors for the products of Electronics Industry, thus closing a magic circle. A rapid drop in prices made these components available for a great variety of people. The stone was falling freely.

The first integrated circuits and processors appeared soon, which caused computers and other products of electronics to drop down in price even more. They could be bought everywhere. Another circle was closed! Ordinary people got hold of computers and computer era has begun.

While this drama was going on, hobbyists and professionals, also split in two groups and protected by anonymity, were working hard on their projects. Then, someone suddenly put a question: Why should not we make a universal component? A cheap, universal integrated circuit that could be programmed and used in any field of electronics, device or wherever needed? Technology has been developed enough as well as the market. Why not? So it happened, body and spirit were united and the first integrated circuit was designed and called the **MICROCONTROLLER.**

2.3.1 8051 Microcontroller

The controller is used to do all the programming. A **microcontroller** (also MCU or µC) is a functional computer system-on-a-chip. It contains a processor core, memory, and programmable input/output peripherals. Microcontrollers include an integrated CPU, memory (a small amount of RAM, program memory, or both) and peripherals capable of input and output. It emphasizes high integration, in contrast to a microprocessor which only contains a CPU (the kind used in a PC). In addition to the usual arithmetic and logic elements of a general purpose microprocessor, the microcontroller integrates additional elements such as read-write memory for data storage, read-only memory for program storage, Flash memory for permanent data storage, peripherals, and input/output interfaces. At clock speeds of as little as 32 KHz, microcontrollers often operate at very low speed compared to microprocessors, but this is adequate for typical applications. They consume relatively little power (milli watts or even microwatts), and will generally have the ability to retain functionality while waiting for an event such as a button press or interrupt. Power consumption while sleeping (CPU clock and peripherals disabled) may be just nanowatts, making them ideal for low power and long lasting battery applications. Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, remote controls, office machines, appliances, power tools, and toys. By reducing the size, cost, and power consumption compared to a design using a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to electronically control many more processes.

The AT89S51 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 non-volatile 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 non-volatile memory programmer. By combining a versatile 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT89S51 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications. The AT89S51 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 AT89S51 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.

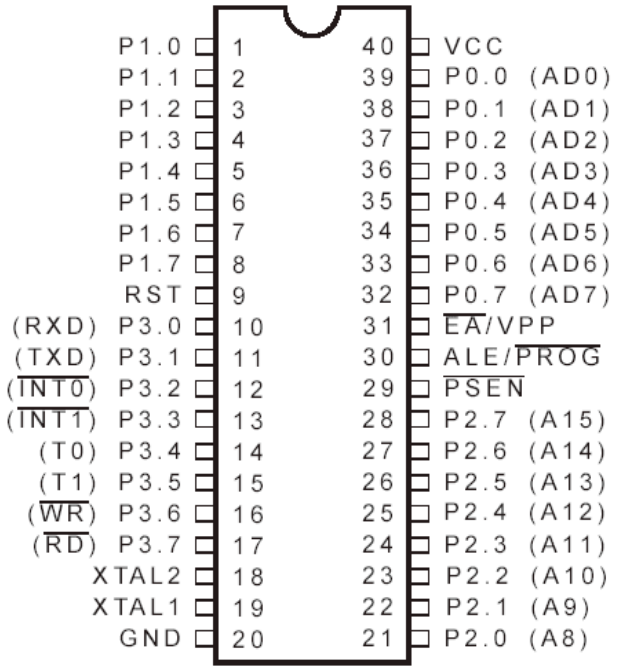


Figure 2.3: Pin Diagram of 8051

2.3.2 What are Microcontrollers and what are they Used for?

Like all good things, this powerful component is basically very simple. It is made by mixing tested and high- quality "ingredients" (components) as per following receipt:

1. The simplest computer processor is used as the "brain" of the future system.
2. Depending on the taste of the manufacturer, a bit of memory, a few A/D converters, timers, input/output lines etc. are added
3. All that is placed in some of the standard packages.
4. A simple software able to control it all and which everyone can easily learn about has been developed.

On the basis of these rules, numerous types of microcontrollers were designed and they quickly became man's invisible companion. Their incredible simplicity and flexibility conquered us a long time ago and if you try to invent something about them, you should know that you are probably late, someone before you has either done it or at least has tried to do it.

The following things have had a crucial influence on development and success of the microcontrollers:

- Powerful and carefully chosen electronics embedded in the microcontrollers can independently or via input/output devices (switches, push buttons, sensors, LCD displays, relays etc.), control various processes and devices such as industrial automation, electric current, temperature, engine performance etc.
- Very low prices enable them to be embedded in such devices in which, until recent time it was not worthwhile to embed anything. Thanks to that, the world is overwhelmed today with cheap automatic devices and various “smart” appliances.
- Prior knowledge is hardly needed for programming. It is sufficient to have a PC (software in use is not demanding at all and is easy to learn) and a simple device (called the programmer) used for “loading” ready-to-use programs into the microcontroller.

So, if you are infected with a virus called electronics, there is nothing left for you to do but to learn how to use and control its power!

2.3.3 HOW DOES THE MICROCONTROLLER OPERATE?

Even though there is a large number of different types of microcontrollers and even more programs created for their use only, all of them have many things in common. Thus, if you learn to handle one of them you will be able to handle them all. A typical scenario on the basis of which it all functions is as follows:

1. Power supply is turned off and everything is still…the program is loaded into the microcontroller, nothing indicates what is about to come…
2. Power supply is turned on and everything starts to happen at high speed! The control logic unit keeps everything under control. It disables all other circuits except quartz crystal to operate. While the preparations are in progress, the first milliseconds go by.
3. Power supply voltage reaches its maximum and oscillator frequency becomes stable. SFRs are being filled with bits reflecting the state of all circuits within the microcontroller. All pins are configured as inputs. The overall electronis starts operation in rhythm with pulse sequence. From now on the time is measured in micro and nanoseconds.
4. Program Counter is set to zero. Instruction from that address is sent to instruction decoder which recognizes it, after which it is executed with immediate effect.
5. The value of the Program Counter is incremented by 1 and the whole process is repeated...several million times per second.

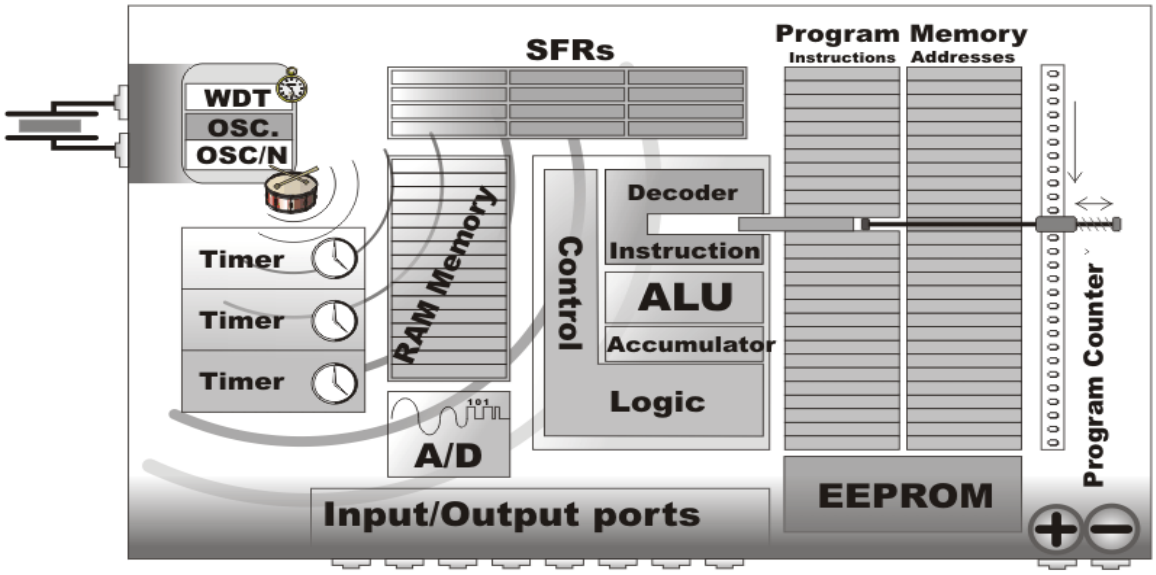


Figure 2.4: Hardware Organisation of 8051

2.3.4 WHAT IS IN THE MICROCONTROLLER?

As you can see, all the operations within the microcontroller are performed at high speed and quite simply, but the microcontroller itself would not be so useful if there are not special circuits which make it complete. In continuation, we are going to call your attention to them.

Read Only Memory (ROM)

Read Only Memory (ROM) is a type of memory used to permanently save the program being executed. The size of the program that can be written depends on the size of this memory. ROM can be built in the microcontroller or added as an external chip, which depends on the type of the microcontroller. Both options have some disadvantages. If ROM is added as an external chip, the microcontroller is cheaper and the program can be considerably longer. At the same time, a number of available pins is reduced as the microcontroller uses its own input/output ports for connection to the chip. The internal ROM is usually smaller and more expensive, but leaves more pins available for connecting to peripheral environment. The size of ROM ranges from 512B to 64kB.

Random Access Memory (RAM)

Random Access Memory (RAM) is a type of memory used for temporary storing data and intermediate results created and used during the operation of the microcontrollers. The content of this memory is cleared once the power supply is off. For example, if the program performs an addition, it is necessary to have a register standing for what in everyday life is called the “sum”. For that purpose, one of the registers in RAM is called the "sum" and used for storing results of addition. The size of RAM goes up to a few KBs.

Electrically Erasable Programmable ROM (EEPROM)

The EEPROM is a special type of memory not contained in all microcontrollers. Its contents may be changed during program execution (similar to RAM), but remains permanently saved even after the loss of power (similar to ROM). It is often used to store values, created and used during operation (such as calibration values, codes, values to count up to etc.), which must be saved after turning the power supply off. A disadvantage of this memory is that the process of programming is relatively slow. It is measured in milli seconds (ms).

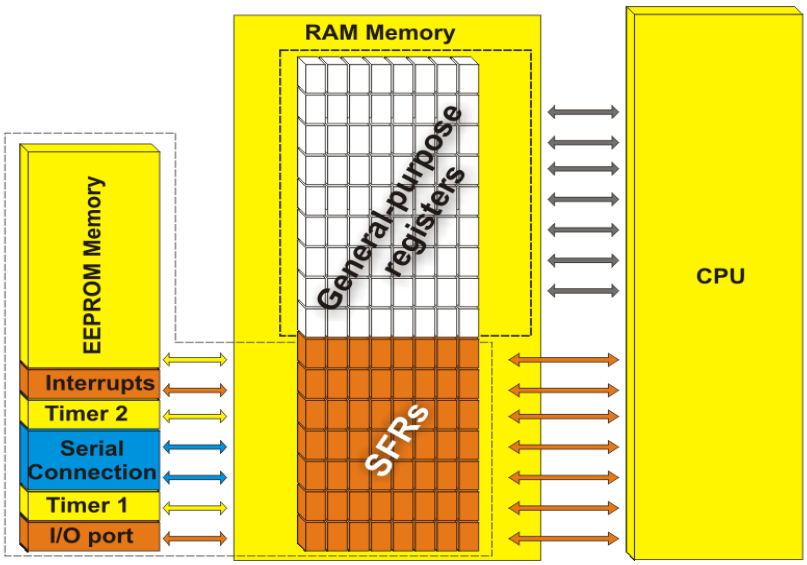


Figure 2.5: Internal structure of Microcontroller

Special Function Registers (SFR)

Special function registers are part of RAM memory. Their purpose is predefined by the manufacturer and cannot be changed therefore. Since their bits are physically connected to particular circuits within the microcontroller, such as A/D converter, serial communication module etc., any change of their state directly affects the operation of the microcontroller or some of the circuits. For example, writing zero or one to the SFR controlling an input/output port causes the appropriate port pin to be configured as input or output. In other words, each bit of this register controls the function of one single pin.

Program Counter

Program Counter is an engine running the program and points to the memory address containing the next instruction to execute. After each instruction execution, the value of the counter is incremented by 1. For this reason, the program executes only one instruction at a time just as it is written. However…the value of the program counter can be changed at any moment, which causes a "jump" to a new memory location. This is how subroutines and branch instructions are executed. After jumping, the counter resumes even and monotonous automatic counting +1, +1, +1…

Central Processor Unit (CPU)

As its name suggests, this is a unit which monitors and controls all processes within the microcontroller and the user cannot affect its work. It consists of several smaller subunits, of which the most important are:

- ***Instruction decoder*** is a part of the electronics which recognizes program instructions and runs other circuits on the basis of that. The abilities of this circuit are expressed in the "instruction set" which is different for each microcontroller family.
- ***Arithmetical Logical Unit (ALU)*** performs all mathematical and logical operations upon data.
- ***Accumulator*** is an SFR closely related to the operation of ALU. It is a kind of working desk used for storing all data upon which some operations should be executed (addition, shift etc.). It also stores the results ready for use in further processing. One of the SFRs, called the Status Register, is closely related to the accumulator, showing at any given time the "status" of a number stored in the accumulator (the number is greater or less than zero etc.).

A bit is just a word invented to confuse novices at electronics. Joking aside, this word in practice indicates whether the voltage is present on a conductor or not. If it is present, the appropriate pin is set to logic one (1), i.e. the bit's value is 1. Otherwise, if the voltage is 0 V, the appropriate pin is cleared (0), i.e. the bit's value is 0. It is more complicated in theory where a bit is referred to as a binary digit, but even in this case, its value can be either 0 or 1.

Input/output ports (I/O Ports)

In order to make the microcontroller useful, it is necessary to connect it to peripheral devices. Each microcontroller has one or more registers (called a port) connected to the microcontroller pins.

Why do we call them input/output ports? Because it is possible to change a pin function according to the user's needs. These registers are the only registers in the microcontroller the state of which can be checked by voltmeter **Oscillator.**

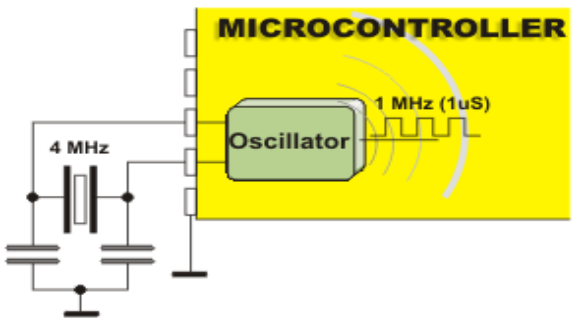


Figure 2.6: Oscillator

Even pulses generated by the oscillator enable harmonic and synchronous operation of all circuits within the microcontroller. It is usually configured as to use quartz-crystal or ceramics resonator for frequency stabilization. It can also operate without elements for frequency stabilization (like RC oscillator). It is important to say that program instructions are not executed at the rate imposed by the oscillator itself, but several times slower. It happens because each instruction is executed in several steps. For some microcontrollers, the same number of cycles is needed to execute any instruction, while it's different for other microcontrollers. Accordingly, if the system uses quartz crystal with a frequency of 20MHz, the execution time of an instruction is not expected 50ns, but 200, 400 or even 800 ns, depending on the type of the microcontroller.

Timers/Counters

Most programs use these miniature electronic stopwatches in their operation. These are commonly 8- or 16-bit SFRs the contents of which is automatically incremented by each coming pulse. Once the register is completely loaded, an interrupt is generated!

If these registers use an internal quartz oscillator as a clock source, then it is possible to measure the time between two events (if the register value is T1 at the moment measurement has started, and T2 at the moment it has finished, then the elapsed time is equal to the result of subtraction T2-T1). If the registers use pulses coming from external source, then such a timer is turned into a counter.

A register or a memory cell is an electronic circuit which can memorize the state of one byte. Besides 8 bits available to the user, each register has also a number of addressing bits. It is important to remember that:

- All registers of ROM as well as those of RAM referred to as general-purpose registers are mutually equal and nameless. During programming, each of them can be assigned a name, which makes the whole operation much easier.
- All SFRs are assigned names which are different for different types of the microcontrollers and each of them has a special function as their name suggests.

Watchdog timer

The Watchdog Timer is a timer connected to a completely separate RC oscillator within the microcontroller.

If the watchdog timer is enabled, every time it counts up to the program end, the microcontroller reset occurs and program execution starts from the first instruction. The point is to prevent this from happening by using a special command. The whole idea is based on the fact that every program is executed in several longer or shorter loops.

If instructions resetting the watchdog timer are set at the appropriate program locations, besides commands being regularly executed, then the operation of the watchdog timer will not affect the program execution.

If for any reason (usually electrical noise in industry), the program counter gets stuck at some memory location from which there is no return, the watchdog will not be cleared, so the register's value being constantly incremented will reach the maximum. Reset occurs.

Power Supply Circuit

There are two things worth attention concerning the microcontroller power supply circuit:

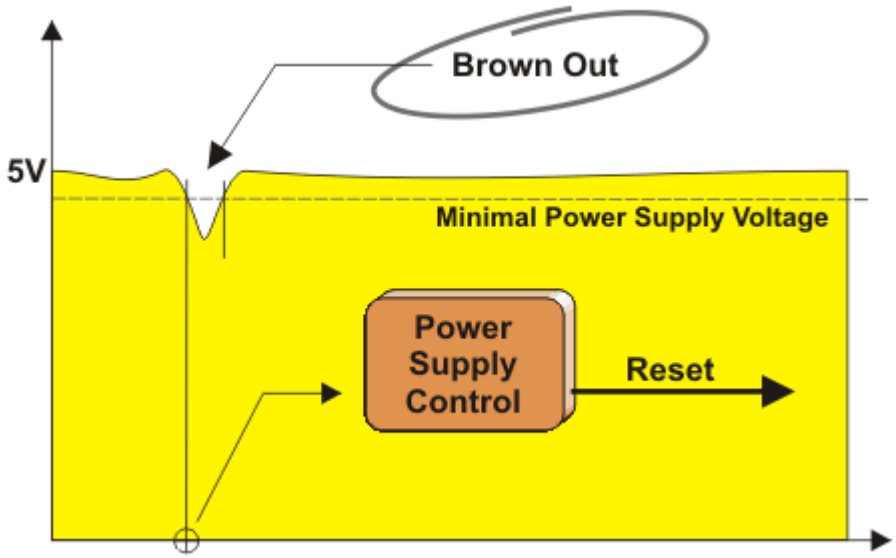


Figure 2.7: Brown out Condition

Brown out is a potentially dangerous state which occurs at the moment the microcontroller is being turned off or when power supply voltage drops to the lowest level due to electric noise. As the microcontroller consists of several circuits which have different operating voltage levels, this can cause it's out of control performance. In order to prevent it, the microcontroller usually has a circuit for brown out reset built-in. This circuit immediately resets the whole electronics when the voltage level drops below the lower limit.

Reset pin is usually referred to as Master Clear Reset (*MCLR*) and serves for external reset of the microcontroller by applying logic zero (0) or one (1) depending on the type of the microcontroller. In case the brown out is not built in the microcontroller, a simple external circuit for brown out reset can be connected to this pin.

Serial Communication

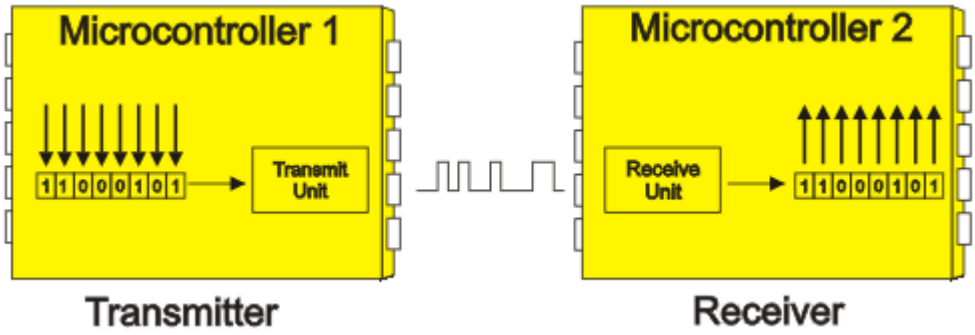


Figure 2.8: Serial Communication

Parallel connections between the microcontroller and peripherals established over I/O ports are the ideal solution for shorter distances up to several meters. However, in other cases, when it is necessary to establish communication between two devices on longer distances it is obviously not possible to use parallel connections. Then, serial communication is the best solution.

Today, most microcontrollers have several different systems for serial communication built in as a standard equipment. Which of them will be used depends on many factors of which the most important are:

- How many devices the microcontroller has to exchange data with?
- How fast the data exchange has to be?
- What is the distance between devices?
- Is it necessary to send and receive data simultaneously?

One of the most important things concerning serial communication is the Protocol which should be strictly observed. It is a set of rules which must be applied in order that devices can correctly interpret data they mutually exchange. Fortunately, the microcontrollers automatically take care of this, so the work of the programmer/user is reduced to a simple write (data to be sent) and read (received data).

A byte consists of 8 bits grouped together. If a bit is a digit then it is logical that bytes are numbers. All mathematical operations can be performed upon them, just like upon common decimal numbers, which is carried out in the ALU. It is important to remember that byte digits are not of equal significance. The largest value has the leftmost bit called the most significant bit (MSB). The rightmost bit has the least value and is therefore called the least significant bit (LSB). Since 8 digits (zeros and ones) of one byte can be combined in 256 different ways, the largest decimal number which can be represented by one byte is 255 (one combination represents zero).

Program

Unlike other integrated circuits which only need to be connected to other components and turn the power supply on, the microcontrollers need to be programmed first. This is a so called "bitter pill" and the main reason why hardware-oriented electronics engineers stay away from microcontrollers. It is a trap causing huge losses because the process of programming the microcontroller is basically very simple.

In order to write a program for the microcontroller, several "low-level" programming languages can be used such as Assembly, C and Basic (and their versions as well). Writing program procedure consists of simple writing instructions in the order in which they should be executed. There are also many programs running in Windows environment used to facilitate the work providing additional visual tools.

Interrupt

Electronics is usually faster than physical processes it should keep under control. This is why the microcontroller spends most of its time waiting for something to happen or execute. In other words, when some event takes place, the microcontroller does something. In order to prevent the microcontroller from spending most of its time endlessly checking for logic state on input pins and registers, an interrupt is generated. It is the signal which informs the central processor that something attention worthy has happened. As its name suggests, it interrupts regular program execution. It can be generated by different sources so when it occurs, the microcontroller immediately stops operation and checks for the cause. If it is needed to perform some operations, a current state of the program counter is pushed onto the Stack and the appropriate program is executed. It's the so called interrupt routine.

Stack is a part of RAM used for storing the current state of the program counter (address) when an interrupt occurs. In this way, after a subroutine or an interrupt execution, the microcontroller knows from where to continue regular program execution. This address is cleared after returning to the program because there is no need to save it any longer, and one location of the stack is automatically availale for further use. In addition, the stack can consist of several levels. This enables subroutines' nesting, i.e. calling one subroutine from another.

CHAPTER 3

MICROCONTROLLER ARCHITECTURE

3.1 WHAT IS 8051 STANDARD?

Microcontroller manufacturers have been competing for a long time for attracting choosy customers and every couple of days a new chip with a higher operating frequency, more memory and upgraded A/D converters appeared on the market.

However, most of them had the same or at least very similar architecture known in the world of microcontrollers as 8051 compatible. What is all this about?

The whole story has its beginnings in the far 80s when Intel launched the first series of microcontrollers called the MCS 051. Even though these microcontrollers had quite modest features in comparison to the new ones, they conquered the world very soon and became a standard for what nowadays is called the **microcontroller.**

The main reason for their great success and popularity is a skilfully chosen configuration which satisfies different needs of a large number of users allowing at the same time constant expansions (refers to the new types of microcontrollers). Besides, the software has been developed in great extend in the meantime, and it simply was not profitable to change anything in the microcontroller's basic core. This is the reason for having a great number of various microcontrollers which basically are solely upgraded versions of the 8051 family. What makes this microcontroller so special and universal so that almost all manufacturers all over the world manufacture it today under different name?

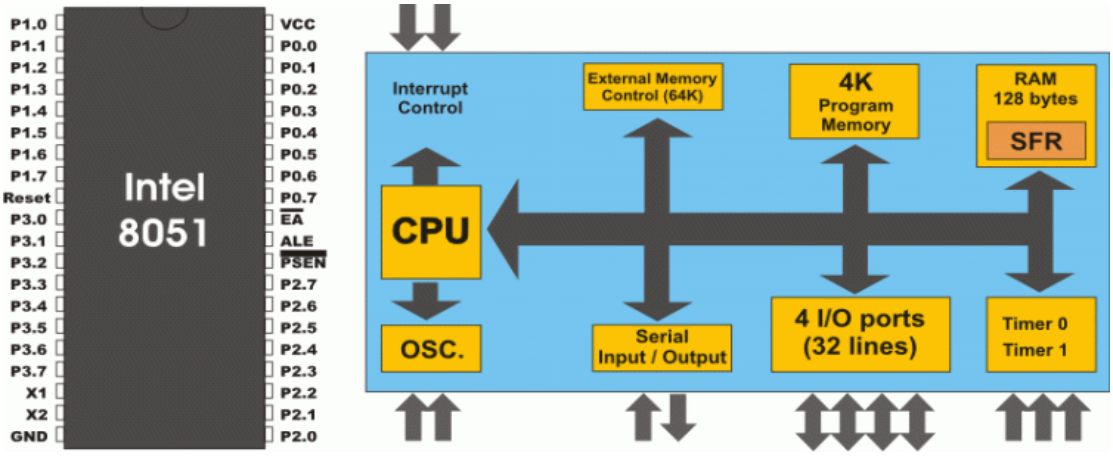


Figure 3.1: 8051 Block Diagram

As seen in figure above, the 8051 microcontroller has nothing impressive in appearance:

- 4 Kb of ROM is not much at all.
- 128b of RAM (including SFRs) satisfies the user's basic needs.
- 4 ports having in total of 32 input/output lines are in most cases sufficient to make all necessary connections to peripheral environment.

The whole configuration is obviously thought of as to satisfy the needs of most programmers working on development of automation devices. One of its advantages is that nothing is missing and nothing is too much. In other words, it is created exactly in accordance to the average user‘s taste and needs. Another advantages are RAM organization, the operation of Central Processor Unit (CPU) and ports which completely use all recourses and enable further upgrade.

3.2 PINOUT DESCRIPTION

Pins 1-8: Port 1 Each of these pins can be configured as an input or an output.

Pin 9: RS A logic one on this pin disables the microcontroller and clears the contents of most registers. In other words, the positive voltage on this pin resets the microcontroller. By applying logic zero to this pin, the program starts execution from the beginning.

Pins10-17: Port 3 Similar to port 1, each of these pins can serve as general input or output. Besides, all of them have alternative functions:

Pin 10: RXD Serial asynchronous communication input or Serial synchronous communication output.

Pin 11: TXD Serial asynchronous communication output or Serial synchronous communication clock output.

Pin 12: INT0 Interrupt 0 input.

Pin 13: INT1 Interrupt 1 input.

Pin 14: T0 Counter 0 clock input.

Pin 15: T1 Counter 1 clock input.

Pin 16: WR Write to external (additional) RAM.

Pin 17: RD Read from external RAM.

Pin 18, 19: X2, X1 Internal oscillator input and output. A quartz crystal which specifies operating frequency is usually connected to these pins. Instead of it, miniature ceramics resonators can also be used for frequency stability. Later versions of microcontrollers operate at a frequency of 0 Hz up to over 50 Hz.

Pin 20: GND Ground.

Pin 21-28: Port 2 If there is no intention to use external memory then these port pins are configured as general inputs/outputs. In case external memory is used, the higher address byte, i.e. addresses A8-A15 will appear on this port. Even though memory with capacity of 64Kb is not used, which means that not all eight port bits are used for its addressing, the rest of them are not available as inputs/outputs.

Pin 29: PSEN If external ROM is used for storing program then a logic zero (0) appears on it every time the microcontroller reads a byte from memory.

Pin 30: ALE Prior to reading from external memory, the microcontroller puts the lower address byte (A0-A7) on P0 and activates the ALE output. After receiving signal from the ALE pin, the external register (usually 74HCT373 or 74HCT375 add-on chip) memorizes the state of P0 and uses it as a memory chip address. Immediately after that, the ALU pin is returned its previous logic state and P0 is now used as a Data Bus. As seen, port data multiplexing is performed by means of only one additional (and cheap) integrated circuit. In other words, this port is used for both data and address transmission.

Pin 31: EA By applying logic zero to this pin, P2 and P3 are used for data and address transmission with no regard to whether there is internal memory or not. It means that even there is a program written to the microcontroller, it will not be executed. Instead, the program

written to external ROM will be executed. By applying logic one to the EA pin, the microcontroller will use both memories, first internal then external (if exists).

Pin 32-39: Port 0 Similar to P2, if external memory is not used, these pins can be used as general inputs/outputs. Otherwise, P0 is configured as address output (A0-A7) when the ALE pin is driven high (1) or as data output (Data Bus) when the ALE pin is driven low (0).

Pin 40: VCC +5V power supply.

3.3 INPUT/OUTPUT PORTS (I/O PORTS)

All 8051 microcontrollers have 4 I/O ports each comprising 8 bits which can be configured as inputs or outputs. Accordingly, in total of 32 input/output pins enabling the microcontroller to be connected to peripheral devices are available for use.

Pin configuration, i.e. whether it is to be configured as an input (1) or an output (0), depends on its logic state. In order to configure a microcontroller pin as an input, it is necessary to apply logic zero (0) to appropriate I/O port bit. In this case, voltage level on appropriate pin will be 0.

Similarly, in order to configure a microcontroller pin as an input, it is necessary to apply a logic one (1) to appropriate port. In this case, voltage level on appropriate pin will be 5V (as is the case with any TTL input). This may seem confusing but don't lose your patience. It all becomes clear after studying simple electronic circuits connected to an I/O pin.

Port 0

The P0 port is characterized by two functions. If external memory is used then the lower address byte (addresses A0-A7) is applied on it. Otherwise, all bits of this port are configured as inputs/outputs.

The other function is expressed when it is configured as an output. Unlike other ports consisting of pins with built-in pull-up resistor connected by its end to 5 V power supply, pins of this port have this resistor left out. This apparently small difference has its consequences:

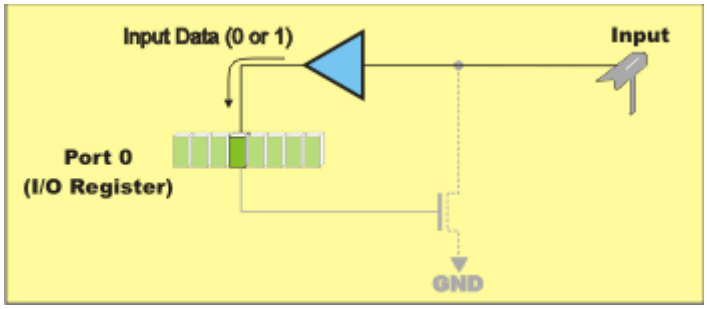


Figure 3.2: Input Data flow in Port 0

If any pin of this port is configured as an input then it acts as if it "floats". Such an input has unlimited input resistance and in determined potential.

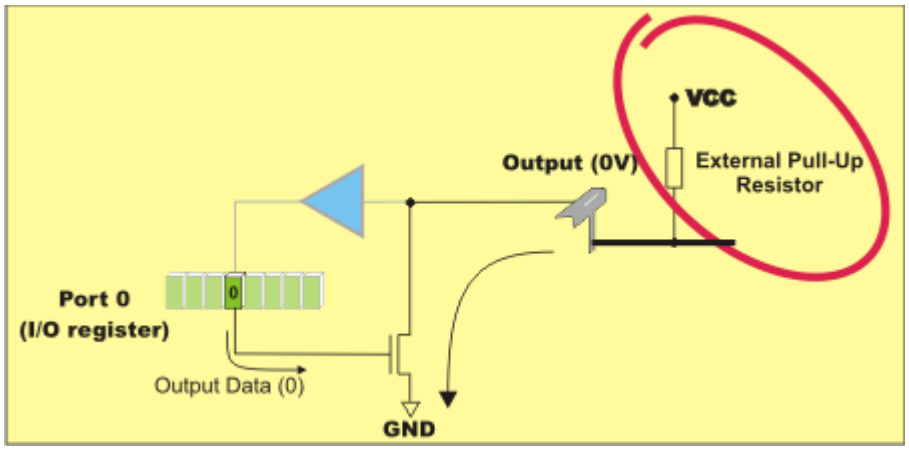


Figure 3.3: Output Data flow in Port 0

When the pin is configured as an output, it acts as an "open drain". By applying logic 0 to a port bit, the appropriate pin will be connected to ground (0V). By applying logic 1, the external output will keep on "floating". In order to apply logic 1 (5V) on this output pin, it is necessary to build in an external pull-up resistor.

Port 1

P1 is a true I/O port, because it doesn't have any alternative functions as is the case with P0, but can be configured as general I/O only. It has a pull-up resistor built-in and is completely compatible with TTL circuits.

Port 2

P2 acts similarly to P0 when external memory is used. Pins of this port occupy addresses intended for external memory chip. This time it is about the higher address byte with

addresses A8-A15. When no memory is added, this port can be used as a general input/output port showing features similar to P1.

Port 3

All port pins can be used as general I/O, but they also have an alternative function. In order to use these alternative functions, a logic one (1) must be applied to appropriate bit of the P3 register. In terms of hardware, this port is similar to P0, with the difference that its pins have a pull-up resistor built-in.

Pin's Current limitations

When configured as outputs (logic zero (0)), single port pins can receive a current of 10mA. If all 8 bits of a port are active, a total current must be limited to 15mA (port P0: 26mA). If all ports (32 bits) are active, total maximum current must be limited to 71mA. When these pins are configured as inputs (logic 1), built-in pull-up resistors provide very weak current, but strong enough to activate up to 4 TTL inputs of LS series.

3.4 MEMORY ORGANIZATION

The 8051 has two types of memory and these are Program Memory and Data Memory. Program Memory (ROM) is used to permanently save the program being executed, while Data Memory (RAM) is used for temporarily storing data and intermediate results created and used during the operation of the microcontroller. Depending on the model in use (we are still talking about the 8051 microcontroller family in general) at most a few Kb of ROM and 128 or 256 bytes of RAM is used.

All 8051 microcontrollers have a 16-bit addressing bus and are capable of addressing 64 kb memory. It is neither a mistake nor a big ambition of engineers who were working on basic core development. It is a matter of smart memory organization which makes these microcontrollers a real programmers friendly.

3.4.1 Program Memory

The first models of the 8051 microcontroller family did not have internal program memory. It was added as an external separate chip. These models are recognizable by their label beginning with 803 (for example 8031 or 8032). All later models have a few Kbyte ROM embedded. Even though such an amount of memory is sufficient for writing most of the

programs, there are situations when it is necessary to use additional memory as well. A typical example are so called lookup tables. They are used in cases when equations describing some processes are too complicated or when there is no time for solving them. In such cases all necessary estimates and approximates are executed in advance and the final results are put in the tables (similar to logarithmic tables).

EA=0 In this case, the microcontroller completely ignores internal program memory and executes only the program stored in external memory.
EA=1 In this case, the microcontroller executes first the program from built-in ROM, then the program stored in external memory.
In both cases, P0 and P2 are not available for use since being used for data and address transmission. Besides, the ALE and PSEN pins are also used.

3.4.2 Data Memory

As already mentioned, Data Memory is used for temporarily storing data and intermediate results created and used during the operation of the microcontroller. Besides, RAM memory built in the 8051 family includes many registers such as hardware counters and timers, input/output ports, serial data buffers etc. The previous models had 256 RAM locations, while for the later models this number was incremented by additional 128 registers. However, the first 256 memory locations (addresses 0-FFh) are the heart of memory common to all the models belonging to the 8051 family. Locations available to the user occupy memory space with addresses 0-7Fh, i.e. first 128 registers. This part of RAM is divided in several blocks.

The first block consists of 4 banks each including 8 registers denoted by R0-R7. Prior to accessing any of these registers, it is necessary to select the bank containing it. The next memory block (address 20h-2Fh) is bit- addressable, which means that each bit has its own address (0-7Fh). Since there are 16 such registers, this block contains in total of 128 bits with separate addresses (address of bit 0 of the 20h byte is 0, while address of bit 7 of the 2Fh byte is 7Fh). The third group of registers occupy addresses 2Fh-7Fh, i.e. 80 locations, and does not have any special functions or features.

3.4.3 Additional RAM

In order to satisfy the programmers' constant hunger for Data Memory, the manufacturers decided to embed an additional memory block of 128 locations into the latest versions of the

8051 microcontrollers. However, it's not as simple as it seems to be… The problem is that electronics performing addressing has 1 byte (8 bits) on disposal and is capable of reaching only the first 256 locations, therefore. In order to keep already existing 8-bit architecture and compatibility with other existing models a small trick was done.

What does it mean? It means that additional memory block shares the same addresses with locations intended for the SFRs (80h- FFh). In order to differentiate between these two physically separated memory spaces, different ways of addressing are used. The SFRs memory locations are accessed by direct addressing, while additional RAM memory locations are accessed by indirect addressing.

3.4.4 Memory expansion

In case memory (RAM or ROM) built in the microcontroller is not sufficient, it is possible to add two external memory chips with capacity of 64Kb each. P2 and P3 I/O ports are used for their addressing and data transmission.

From the user's point of view, everything works quite simply when properly connected because most operations are performed by the microcontroller itself. The 8051 microcontroller has two pins for data read RD# (P3.7) and PSEN#. The first one is used for reading data from external data memory (RAM), while the other is used for reading data from external program memory (ROM). Both pins are active low. A typical example of memory expansion by adding RAM and ROM chips (Hardware architecture), is shown in figure above.

Even though additional memory is rarely used with the latest versions of the microcontrollers, we will describe in short what happens when memory chips are connected according to the previous schematic. The whole process described below is performed automatically.

- When the program during execution encounters an instruction which resides in external memory (ROM), the microcontroller will activate its control output ALE and set the first 8 bits of address (A0-A7) on P0. IC circuit 74HCT573 passes the first 8 bits to memory address pins.
- A signal on the ALE pin latches the IC circuit 74HCT573 and immediately afterwards 8 higher bits of address (A8-A15) appear on the port. In this way, a desired location of additional program memory is addressed. It is left over to read its content.

- Port P0 pins are configured as inputs, the PSEN pin is activated and the microcontroller reads from memory chip.

Similar occurs when it is necessary to read location from external RAM. Addressing is performed in the same way, while read and write are performed via signals appearing on the control outputs RD (is short for read) or WR (is short for write).

CHAPTER 4

GSM MODULE

GSM (Global System for Mobile) / GPRS (General Packet Radio Service) TTL-Modem is SIM900 Quad-band GSM / GPRS device, works on frequencies 850 MHZ, 900 MHZ, 1800 MHZ and 1900 MHZ. It is very compact in size and easy to use as plug in GSM Modem. The Modem is designed with 3V3 and 5VDC TTL interfacing circuitry, which allows User to directly interface with 5V Microcontrollers (PIC, AVR, Arduino, 8051, etc.) as well as 3V3 Microcontrollers (ARM, ARM Cortex XX, etc.). The baud rate can be configurable from 9600- 115200 bps through AT (Attention) commands. This GSM/GPRS TTL Modem has internal TCP/IPstack to enable User to connect with internet through GPRS feature. It is suitable for SMS as well as DATA transfer application in mobile phone to mobile phone interface. The modem can be interfaced with a Microcontroller using USART (Universal Synchronous Asynchronous Receiver and Transmitter) feature (serial communication).

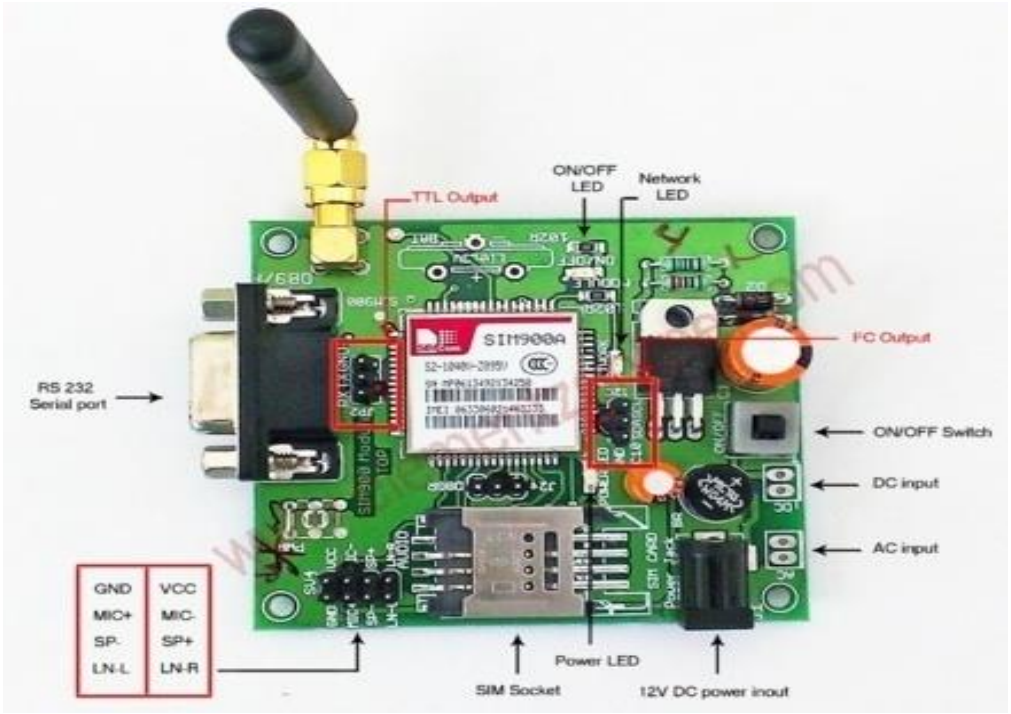


Figure 4.1: GSM module SIM900 model

4.1 Features

- Quad Band GSM/GPRS: 850 / 900 / 1800 / 1900 MHz
- Built in RS232 to TTL or vice versa Logic Converter (MAX232)
- Configurable Baud Rate
- SMA (Sub Miniature version A) connector with GSM L Type Antenna
- Built in SIM (Subscriber Identity Module) Card holder
- Built in Network Status LED
- Inbuilt Powerful TCP / IP (Transfer Control Protocol / Internet Protocol) stack for internet data transfer through GPRS (General Packet Radio Service)
- Audio Interface Connectors (Audio in and Audio out)
- Most Status and Controlling pins are available
- Normal Operation Temperature : -20 °C to +55 °C
- Input Voltage : 5V to 12V DC
- LDB9 connector (Serial Port) provided for easy interfacing
- SIM Card Holder
- Buzzer
- Supply Port
- Audio Connector
- Communication PORT
- Antenna

GSM Module works with AT COMMANDS. **AT commands** are used to control MODEMs. AT is the abbreviation for Attention.
AT commands with a GSM/GPRS MODEM or mobile phone can be used to access following information and services:

1. Information and configuration pertaining to mobile device or MODEM and SIM card.
2. SMS services.

3. MMS services.

4. Call services.

5. Data and Voice link over mobile network.

4.2 Explanation of commonly used commands

1) AT - This command is used to check communication between the module and the computer.

For example,

AT

OK

The command returns a result code OK if the computer (serial port) and module are connected properly. If any of module or SIM is not working, it would return a result code ERROR.

2) +CMGF - This command is used to set the SMS mode. Either text or PDU mode can be selected by assigning 1 or 0 in the command.

SYNTAX:

AT+CMGF=<mode>

0: for PDU mode

1: for text mode

The text mode of SMS is easier to operate but it allows limited features of SMS. The PDU (protocol data unit) allows more access to SMS services but the operator requires bit level knowledge of TPDUs. The headers and body of SMS are accessed in hex format in PDU mode so it allows availing more features.

3) +CMGW - This command is used to store message in the SIM.

SYNTAX: AT+CMGW=" Phone number"> Message to be stored Ctrl+z

As one types AT+CMGW and phone number, > sign appears on next line where one can type the message. Multiple line messages can be typed in this case. This is why the message is terminated by providing a "Ctrl+z" combination. As Ctrl+z is pressed, the following information response is displayed on the screen.

+CMGW: Number on which message has been stored

4) +CMGS - This command is used to send a SMS message to a phone number.

SYNTAX: AT+CMGS= serial number of message to be send.

As the command AT+CMGS and serial number of message are entered, SMS is sent to the particular SIM.

For example,

AT+CMGS=1

OK

5) ATD - This command is used to dial or call a number.

SYNTAX: ATD<Phone number > (Enter)

For example,

ATD123456789

6) ATA - This command is used to answer a call. An incoming call is indicated by a message "RING" which is repeated for every ring of the call. When the call ends „NO CARRIER" is displayed on the screen.

SYNTAX: ATA (Enter)

As ATA followed by enter key is pressed, incoming call is answered.

For example,

RING

RING

ATA

7) ATH - This command is used to disconnect remote user link with the GSM module.

SYNTAX: ATH (Enter)

4.3 SIMCom SIM900AGSM Module

This is actual SIM900 GSM module which is manufactured by SIMCom. Designed for global market, SIM900 is a quad-band GSM/GPRS engine that works on frequencies GSM 850MHz, EGSM 900MHz, DCS 1800MHz and PCS 1900MHz. SIM900 features GPRS multi slot class 10/ class 8 (optional) and supports the GPRS coding schemes CS-1, CS-2, CS-3 and CS-4. With a tiny configuration of 24mm x 24mm x 3mm, SIM900 can meet

almost all the space requirements in User's applications, such as M2M, smart phone, PDAand other mobile devices.

4.3.1 Power Supply Socket

This power supply socket which actually named as AC/DC Socket provides the functionality to user to connect external power supply from Transformer, Battery or Adapter through DC jack. User can provide maximum of 12V AC/DC power supply through AC/DC socket. This is power supply designed into maximum protection consideration so that it can even prevent reverse polarity DC power supply as well as DC conversion from AC power Supply. It also includes LM317 Voltage Regulator which provides an output voltage adjustable over a1.2V to 37V.

4.3.2 Power On/Off and GSM on Switch

Power On/Off switch is type of push-on push-off DPDT switch which is used for only make power supply on/off provided through AC/DC Socket indicated by 'Power LED'. GSM On Switch is type of Push on DPST tactile switch which is used for only to make GSM module 'On' indicated by 'Module On/Off LED' while initiating with Network indicated by 'Network Indication LED'.

4.3.3 SIM (Subscriber Identity Module) Card Slot

This on board SIM card slot provide User functionality of insert a SIM (GSM only) card of any service provider. Process of inserting and locking SIM card into SIM card slot is given in this manual. While inserting in and removing out SIM card from SIM card slot, User needs to take precaution that power supply should be OFF so that after making Power supply ON it will be easy to reinitialize with SIM for this module.

4.3.4 Indicator LEDs

Indicator LEDs just used to indicate status accordingly. These are three LEDs represents Power On/Off Status, Network Status and Module On/Off Status respectively. Power LED will keep on until the power supply is enable to this board by using push-on push-offswitch. Network Status LED will show whether inserted SIM card successfully connected to service provider's Network or not, in short signal strength. Module On/Off indicator LED will show status of GSM module.

4.3.5 Serial Communication (Connecting GSM Module with Serial to USB converter through RXD, TXD and GND)

This module is designed in a way so that User can connect this module without Serial cable, this module can be connected to any of Serial to USB converter module or cable. Here we have shown demo how to connect this interfacing board with CP2102 Serial to USB converter Module through RXD, TXD and GND. Connect CP2102 Serial to USB converter module to PC through USB cable, connect one end of USB cable to PC's USB connector and connect another end of USB to CP2102 module's USB connector. Connect three Single Berg Wires to CP2102 modules' RXD, TXD and GND pin. Then connect RXD wire to TXD of GSM module and TXD wire to RXD of GSM module. Make GND common by connecting GND wire to GND pin of GSM module.

4.3.6 Testing GSM Module on Terminal Software

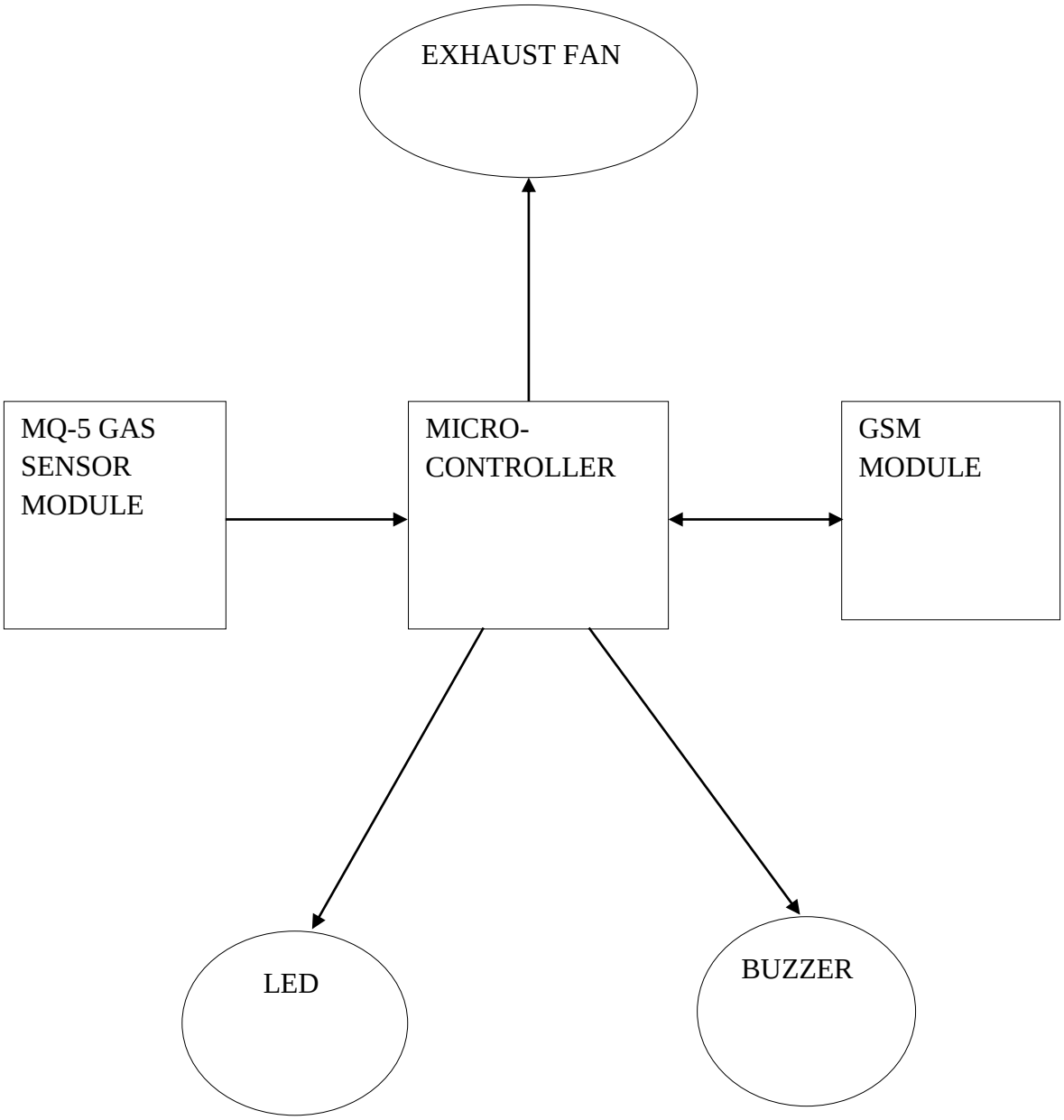
Terminal software used share the data through Serial port. Hence here also User need this software to test GSM module. For demo purpose we are going to show demo for how to send SMS and how to dial a call through Realterm software. First of all install Realterm, it will create its own shortcut on Desktop. Then double click on Realterm icon.

CHAPTER- 5

DESIGN OVERVIEW

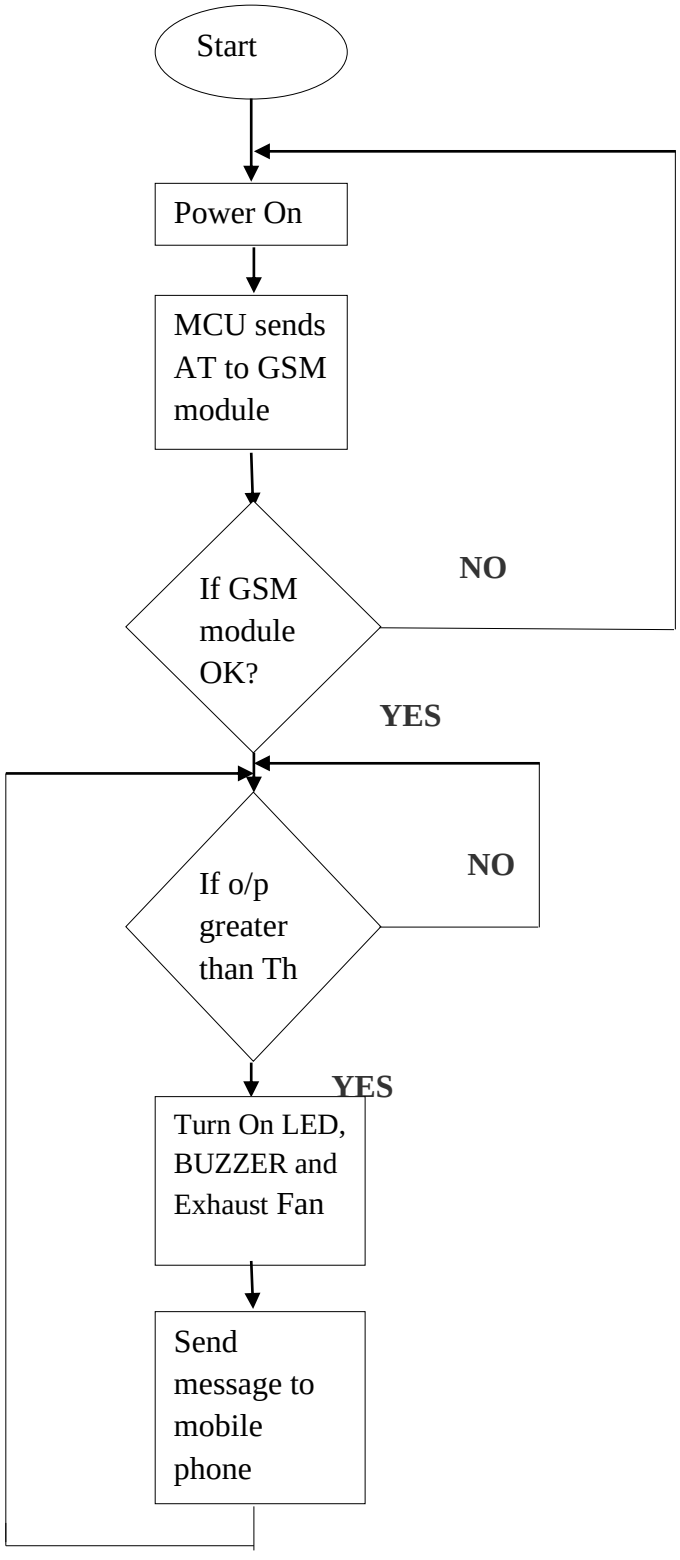
5.1 BLOCK DIAGRAM

Gas Sensor senses the leakage and intimate the same to the Microcontroller. Microcontroller controls the whole system. It interacts with all the components of the project. It receives signal from the sensor and sends signals to other components. It receives OK signal from GSM module and sends signal of leakage to it. Buzzer, LED, Fan also receive signal from it.



5.2 FLOWCHART

This flowchart will describe the procedure from the point of turning ON the system and further how the system works in order.



First of all, the systems power is turned ON. As a result the sensor gets heated and gets ready to operate. Then Microcontroller send AT command to GSM module to check whether it is working properly or not. If No then the loop continues until the answer comes out to be Yes. After the affirmative signal from the GSM module, the system is in ready state for detection. If leakage occurs and the amount is above a certain threshold value, them the Microcontroller will do its jobs to intimate other components of the system. It will signal the Buzzer to beep and LED to glow. Also Exhaust Fan will be signalled to get started. GSM module will be signalled to send an SMS to the predefined mobile number.

CHAPTER - 6

SOFTWARE DESCRIPTION

6.1 INTRODUCTION TO KEIL

Keil Micro Vision is an integrated development environment used to create software to be run on embedded systems (like a microcontroller). It allows for such software to be written either in assembly or C programming languages and for that software to be simulated on a computer before being loaded onto the microcontroller.

µVision3 is an IDE (Integrated Development Environment) that helps write, compile, and debug embedded programs. It encapsulates the following components:

- A project manager.
- Make facility.
- Tool configuration
- Editor
- A powerful debugger.

6.2 TO CREATE A NEW PROJECT IN UVISION3:

1. Select Project - New Project.
2. Select a directory and enter the name of the project file.
3. Select Project = Select Device and select a device from Device Database.
4. Create source files to add to the project
5. Select Project - Targets, Groups, and Files. Add/Files, select Source Group1, and add the source files to the project.
6. Select Project - Options and set the tool options. Note that when the target device is selected from the Device Database all-special options are set automatically. Default memory model settings are optimal for most applications.

7. Select Project - Rebuild all target files or Build target.

To create a new project, simply start Micro Vision and select Project=>New Project from the pull down menus. In the file dialog that appears, choose a name and directory for the project. It is recommended that a new directory be created for each project, as several files will be generated. Once the project has been named, the dialog shown in the figure below will appear, prompting the user to select a target device. In this lab, the chip being used is the AT89C52, which is listed under the heading Atmel.

Windows for choosing target device:

Next, Micro Vision must be instructed to generate a HEX file upon program compilation. A HEX file is a standard file format for storing executable code that is to be loaded onto the microcontroller. In the Project Workspace pane at the left, right click on Target 1 and select Options for 1 .Under the Output tab of the resulting options dialog, ensure that both the Create Executable and Create HEX File options are checked. Then click OK.

Next, a file must be added to the project that will contain the project code. To do this, expand the Target 1 heading, right click on the Source Group 1 folder, and select Add files. Create a new blank file (the file name should end in .asm), select it, and click Add. The new file should now appear in the Project Workspace pane under the Source Group 1 folder. Double-click on the newly created file to open it in the editor. All code for this lab will go in this file. To compile the program, first save all source files by clicking on the Save All button, and then click on the Rebuild All Target Files to compile the program as shown in the figure below. If any errors or warnings occur during compilation, they will be displayed in the output window at the bottom of the screen. All errors and warnings will reference the line and column number in which they occur along with a description of the problem so that they can be easily located. Note that only errors indicate that the compilation failed, warnings do not (though it is generally a good idea to look into them anyway).

When the program has been successfully compiled, it can be simulated using the integrated debugger in Keil Micro Vision. To start the debugger, select Debug=>Start/Stop Debug Session from the pull down menus.

At the left side of the debugger window, a table is displayed containing several key parameters about the simulated microcontroller, most notably the elapsed time (circled in the figure below). Just above that, there are several buttons that control code execution. The Run

button will cause the program to run continuously until a breakpoint is reached, whereas the Step Into button will execute the next line of code and then pause (the current position in the program is indicated by a yellow arrow to the left of the code).

6.3 PROGRAMMER

The programmer used is a powerful programmer for the Atmel 89 series of microcontrollers that includes 89C51/52/55, 89S51/52/55 and many more.

It is simple to use & low cost, yet powerful flash microcontroller programmer for the Atmel 89 series. It will Program, Read and Verify Code Data, Write Lock Bits, Erase and Blank Check. All fuse and lock bits are programmable. This programmer has intelligent on board firmware and connects to the serial port. It can be used with any type of computer and requires no special hardware. All that is needed is a serial communication port which all computers have.

All devices also have a number of lock bits to provide various levels of software and programming protection. These lock bits are fully programmable using this programmer. Locks bits are useful to protect the program to be read back from microcontroller only allowing erase to reprogram the microcontroller.

Major parts of this programmer are Serial Port, Power Supply and Firmware microcontroller. Serial data is sent and received from 9 pin connector and converted to/from TTL logic/RS232 signal levels by MAX232 chip. A Male to Female serial port cable, connects to the 9 pin connector of hardware and another side connects to back of computer. All the programming intelligence is built into the programmer so you do not need any special hardware to run it. Programmer comes with window based software for easy programming of the devices.

CHAPTER – 7

WORKING

Initially, the microcontroller sends signal to the GSM module and if the GSM module is connected properly with the microcontroller it sends an acknowledgement signal back to the microcontroller. Then if there is any gas leakage in the atmosphere it is detected by the gas sensor unit using MQ-5 sensor. After the sensor unit detects the gas leakage, a signal is sent to the ADC unit of the microcontroller which then sends activation signal to other external devices connected to it such as buzzer, GSM module, and exhaust fan. The GSM module gets activated which sends a warning SMS to the user and turns on the exhaust fan. At the end, when the gas leakage is successfully stopped then with the help of reset button the whole system is made to reach its initial stage. The MQ-5 Gas Sensor is a semiconductor type gas sensor which detects gas leakage by comparing the concentration of ethanol which is present as a mixture in the LPG with air. It then gives analogue voltage as output. MQ-5 is a SnO2 sensor. Tin oxide sensors are generally operated in air in the temperature range between 200 and 400◦C. At these temperatures it is generally accepted that the conduction is electronic; it is also accepted that chemisorption of atmospheric gases takes place at the surface of the tin oxide. The overall conduction in a sensor element, which determines the sensor resistance, is determined by the surface reactions, the resulting charge transfer processes with the underlying semi conducting material and the transport mechanism from one electrode to the other through the sensing layer. For example, it is well known that oxygen ionosorption as O−2or O− will result in the building of a negative charge at the surface and the increase of the surface resistance. It is also considered that reducing gases like ethanol react with the surface oxygen ions, freeing electrons—the sensing step— that can return to the conduction band. The transduction step, i.e. the actual translation of this charge transfer into a decrease of the sensor resistance, depends on the morphology of the sensing layer. The result is that, even for exactly the same surface chemistry, the dependence of the sensor resistance on the concentration of ethanol can be very different for compact and porous sensing layers. In our case, the sensing layer consists of single crystalline grains with a narrow size distribution. Due to the fact that the final thermal treatment is performed at 700◦C, the grains are just loosely connected. Accordingly, the best way to describe the conduction process is to consider that the free charge carriers (electrons for SnO2) have to overcome the surface barriers appearing at the surface of the grains. Due to the narrow size distribution it is also

quite probable that a mean-field treatment suffices. One can easily model the dependence of the resistance on the ethanol concentration by making the following assumptions supported by the already established knowledge in this field.

The reaction of ethanol takes place just with the previously adsorbed oxygen ions (well documented for the temperature and pressure range in which the gas sensors operate).

The adsorption of ethanol is proportional to the ethanol concentration in the gas phase.

On the basis of the above assumptions one can combine quasi-chemical reaction formalism with semiconductor physics calculations and one obtains power-law dependences of the form

$$R \sim p\ n\ \text{ethanol} \qquad \ldots\ldots\ldots (1)$$

where the value of n depends on the morphology of the sensing layer and on the actual bulk properties of the sensing materials. The relationship described by equation (1) is well supported by experiments.

For the effect of water vapour on the resistance of tin oxide based gas sensors there are a couple of ideas, briefly presented below. There are three types of mechanisms to explain the experimentally proven increase of surface conductivity in the presence of water vapour. Two, direct mechanisms, are proposed by Heiland and Kohl and the third, indirect, is suggested by Morrison and by Henrich and Cox.

The first mechanism of Heiland and Kohl attributes the role of the electron donor to the 'rooted' OH group, the one including lattice oxygen. The equation proposed is

$$(Sn\delta+\square H2Ogas+ SnSn+ OO\ Sn{-}OH\delta-) + (OH) + O+ e- \qquad \ldots\ldots\ldots (2)$$

where ($Sn\delta+\ Sn{-}OH\delta-$) is referred to as an isolated hydroxyl or OH group (dipole) and (OH) + Ois the rooted one. In the first equation, the donor is already ionized. The reaction implies the homolytic dissociation of water and the reaction of the neutral H atom with the lattice oxygen. The latter is normally fixing two electrons and then consequently being in the (2−) state. The built-up rooted OH group, having a lower electron affinity, can become ionized and become a donor. The second mechanism takes into account the possibility of the reaction between the hydrogen atom and the lattice oxygen and the binding of the resulting hydroxyl group to theSn atom. The resulting oxygen vacancy will produce, by ionization, the additional electrons. The equation proposed by Heiland and Kohl is

$$2(Sn\delta+\square H2Ogas+ 2SnSn+ OO\ Sn{-}OH\delta-) + V\ 2+ O+ 2e- \qquad \ldots\ldots\ldots (3)$$

Morrison, as well as Henrich and Cox, consider an indirect effect more probable. This effect could be the interaction between either the hydroxyl group or the hydrogen atom originating from the water molecule with an acid or basic group, which are also acceptor surface states. Their electronic affinity could change after the interaction. It could also be the influence of the co-adsorption of water on the adsorption of another adsorbate which could be an electron acceptor. Henrich and Cox suggested that the pre- adsorbed oxygen could be displaced by water adsorption. In any of these mechanisms, the particular state of the surface plays a major role, due to the fact that it is considered that steps and surface defects will increase the dissociative adsorption. The surface dopants could also influence these phenomena; all studies showed that the oxygen adsorbates are rearranged in the presence of adsorbed water. The rearrangement was different in the case of Ag and Pd surface doping. In choosing between one of the proposed mechanisms, one has to keep in mind that:

In all reported experiments, the effect of water vapour was the increase of surface conductance.

The effect is reversible, generally with a time constant of the order of around 1 h.

It is not easy to quantify the effect of water adsorption on the charge carrier concentration. For the first mechanism of water interaction proposed by Heiland and Kohl (rooted equation (2)), one could include the effect of water by considering the effect of an increased background of free charge carriers on the adsorption of oxygen. For the second mechanism proposed by Heiland and Kohl (isolated equation (3)) one can examine the influence of water adsorption as an electron injection combined with the appearance of new sites for oxygen chemisorption's; this is valid if one considers oxygen vacancies as good candidates for oxygen adsorption. In this case one has to introduce the change in the total concentration of adsorption sites.

$$[St] = [St0] + k0pH2O \quad \ldots\ldots\ldots (4)$$

obtained by applying the mass action law to equation (3). [St0] is the intrinsic concentration of adsorption sites and k0is the adsorption constant for water vapour. In the case of interaction with surface acceptor states, not related to oxygen adsorption, one can proceed as in the case of the first mechanism proposed by Kohl. In the case of an interaction with oxygen adsorbates, one can consider that the dissociation of oxygen ions is increased and examine the implications. The MQ-5 sensor has a sensing range of 300-1000ppm. The response time for measuring LPG gas content is quick. Whenever there is a gas leakage, the ethanol present in the air is oxidized to acetic acid, which is an organic acid. The resulting chemical reaction will produce an electrical current. The difference of potential produced by

this reaction is measured, processed, and displayed as an approximation of overall gas content in the atmosphere. The MQ-5 has six contacts. There is no polarization on the sensor so any of the two contacts, A or B, can be used interchangeably as Vcc and Ground. The contacts labelled as H are the contacts for the internal heating system. The internal heating system is a small tube made of aluminium oxide and tin dioxide. Inside this tube, there are heating coils which produce the heat. These coils can draw up to 150mA of current. The alumina tube is covered with tin dioxide, SnO2. Embedded between SnO2 and alumina tube is an aurum electrode. When heated, the SnO2 becomes a semiconductor and produces movable electrons. These movable electrons allow the flow of more current. When LPG gas molecules contact the electrode, the ethanol present in the LPG chemically changes into acetic acid and produces a flow of current within the tube. The more LPG gas present the more current is produced. The current, however, is not what is measured when measuring the output, what is measured is the voltage between the output of the sensor and the load resistor. Also, inside the sensor there is a variable resistor across contacts A and B. The resistance between the contacts A and B will vary depending on the amount of LPG present. As the amount of LPG increases, the internal resistance will decrease and thus, the voltage at the output will increase. This voltage is the analogue signal transmitted to the ADC of the microcontroller. The GSM module is used to send an SMS to the user‘s cell phone number. When gas leakage is detected by the gas sensor, the microcontroller sends a signal to the GSM module which then sends a message to the user. These SMSs are saved in the microcontroller memory. Multiple SMSs can also be sent to the user, police, fire station etc.

7.1 Connection between Microcontroller and GSM Module

Transmitter Pin (Tx) of Microcontroller is connected to the Receiver Pin (Rx) of GSM Module and Receiver Pin (Rx) of Microcontroller is connected to the Transmitter Pin (Tx) of GSM Module.

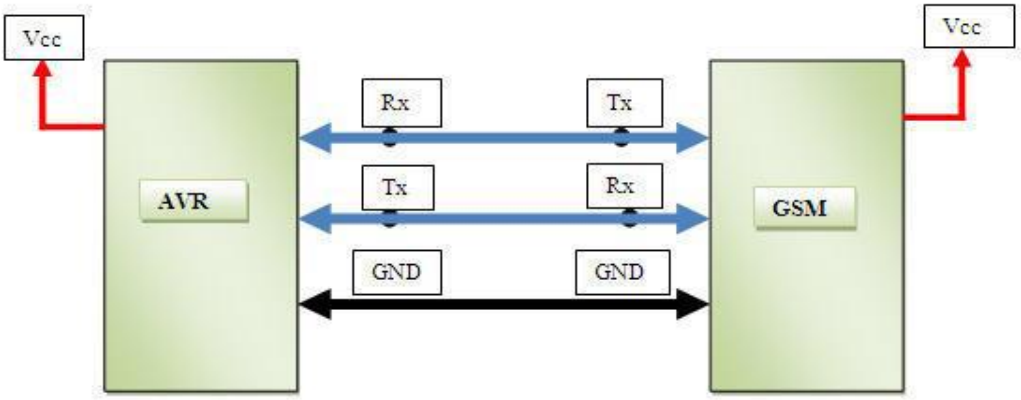


Figure 7.1: Connection of Microcontroller with GSM module.

7.2 Power Supply

With the help of step down transformer of 230V AC primary to 0-12V, 500mA secondary power supply is taken from main supply. Full-wave rectifier and a capacitor filter provide the output voltage and then fed to 5-volt regulator (LM7805) whose output is used as power supply for IC's and microcontroller.

CHAPTER 8
RESULTS AND CONCLUSION

8.1 APPLICATIONS

This project is applicable in following fields:

1. Gas leakage detector (Domestic)
2. Combustible gas detector (Industrial)
3. Gas detector (Portable)
4. Homes
5. Factories
6. LPG storage
7. Gas cars etc.

8.2 FUTURE WORK

Voice feedback system can be included in GSM based LPG weight and LPG leakage detection system. User will get intimation through pre-recorded voice messages. Further how much amount of gas has been leaked so far can be measured and informed to the user.

8.3 RESULTS

The designed system prevents gas leakage inside homes as well as outside homes. The system detects the LPG gas concentration in the air if it exceeds a safety level and then responds by using GSM to send an SMS to the consumer. The LED and Buzzer are activated to alert the consumer in case of gas leakage and the system displays the message on LCD display.

8.4 CONCLUSION

LPG Gas Leakage is a major problem in many industries and households. We have designed such a system which can detect Gas Leakages effectively using a gas sensor and alert people either by using GSM to send a message to their mobile phones or by activating the LED, Buzzer. Hence our Project will definitely prove to be a boon for households and industries in preventing future gas leakages.

This project work has many advantages which are as follows:

-The Research Work is easy to use and it gives remote indication to the user.

-The Sensor used in this Research Work has excellent sensitivity combined with a quick fast response time.

-The system is highly reliable, tamper-proof and secure.

-In the long run the maintenance cost is very less when compared to the present systems.

-It is possible to get instantaneous results and with high accuracy.

REFERENCES

[1] H.Huang, H.BainandS.Zhu,"A Greenhouse Remote Monitoring System Based on GSM," in Proc. of IEEE International Conference on information management , pp. 357-360,2011.

[2] "A Wireless Home Safety Gas Leakage Detection System": LuayFriwan, KhaldonLweesy ,AyaBani-Salma , Nour Mani. IEEE 2011.

[3] "Automatic LPG Booking, Leakage Detection and Real Time Gas Measurement Monitoring System": B. D. Jolhe, P. A. Potdukhe, N. S. Gawai.

[4] Hanwei Electronics Co. Ltd, MQ 5 sensor Datasheet.

[5] "GSM: Architecture, protocols and services" by JorgEberspacher, Christian, Hansjoergvogel, Christian Hartmann, John Wiley Son Ltd, 2009.

[6] Rajkamal, "Embedded Systems: Architecture, Programming and Design", Tata McGraw-Hill Education.

[7] M. Yuchun, H. Yinghong, Zhang Kun and Li Zhuang, "General Application Research on GSM Module," IEEE 8th International Conference on Internet Computing and Information Services (IPEC), 2007.

[8] International Journal of Technical Research and Applications eISSN: 2320-8163, www.ijtra.com Volume 1, Issue 2 (may-june 2013), PP. 42-45.