

WIRELESS NOTICE BOARD USING GSM

Project Report submitted in partial fulfillment of the degree of
Bachelor of Technology



May 2014

SAURABH SENGER

101055

Under the Supervision of

DR. RAJIV KUMAR

**DEPARTMENT OF ELECTRONICS AND
COMMUNICATION ENGINEERING**

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

WAKNAGHAT

CERTIFICATE

This is to certify that project report entitled “Wireless Notice Board Using GSM”, submitted by Saurabh Senger 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.

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.

Date:

Supervisor’sName

Designation

Acknowledgement

We owe a great many thanks to a great many people who have helped and supported us during this project. Our deepest thanks to **Dr. Rajiv Kumar**, our project guide for his exemplary guidance, monitoring and constant encouragement throughout the course of this project work. He has taken great pains to go through the project and make necessary corrections as and whenever needed. We are also grateful to **Mr. Mohan (ECE Project lab)** for his practical help and guidance. We would also like to thank all the faculty members of ECE department without whom the progress of this project would have been a distant reality. We also extend our heartfelt thanks to our family and well-wishers.

Date:

Name of the student

CONTENTS

CHAPTER	TOPICS	Page No.
	CERTIFICATE	2
	ACKNOWLEDGEMENT	3
	ABSTRACT	9
1	INTRODUCTION	10
	1.1 INTRODUCTION TO PROJECT	
2	PCB DESIGNING AND FABRICATION	11
	2.1 INTRODUCTION	
	2.2 FLOW CHART FOR PCB DESIGNING	
	2.3 DESIGNING OF PCB LAYOUT	
3	ATMEGA 168	16
	3.1 INTRODUCTION	
	3.2 FEATURES	
	3.3 DESCRIPTIONS	

3.4	PIN DIAGRAM	
3.5	THINGS TO REMEMBER ABOUT ATMEGA MICROCONTROLLER	
3.6	SELECTION OF MICROCONTROLLER	
3.7	APPLICATION SCENERIO FOR MICROCONTROLLER	
3.8	IMPORTANT FACTS ABOUT MICROCONTROLLER	
4	GSM	25
4.1	INTRODUCTION	
4.2	CELLULAR MOBILE SYSTEM	
4.3	GSM TECHNOLOGY	
4.4	MOBILE STATION	
4.5	NETWORK SYSTEM AND SWITCHING	
4.6	LOCATION UPDATING	
4.7	AUTHENTICATION AND SECURITY	
4.8	CALL MANAGEMENT	
4.9	CALL CONTROL	
4.10	CALL ROUTING	
4.11	SUPPLEMENTARY SERVICE MANAGEMENT	
4.12	STRENGTH OF GSM	
4.13	SERVICE PROVIDED BY THE GSM	
4.14	APPLICATIONS	
4.15	WHAT APPLICATION ARE SUITABLE FOR GSM?	
4.16	INTRODUCTION TO GSM MODULE	
4.17	AT COMMANDS FORMAT	
4.18	GSM MODEM PC INTERFACING	

5	EXPRESS SCH	39
5.1	INTRODUCTION	
5.2	COMPONENTS AND SYMBOLS	
5.3	PLACING THE COMPONENTS	
5.4	PLACING THE SYMBOLS AND PORTS	
5.5	WIRING COMPONENTS	
5.6	WORKING WITH WIRES	
6	LCD	42
6.1	INTRODUCTION	
6.2	PIN DIAGRAM	
6.3	SEQUENCING OF WRITING TO THE LCD	
6.4	LCD INTERFACES DIAGRAM	
7	SOURCE CODE	46
	CONCLUSION	57
	REFRENCES	58

List of Figures

S.No.	Title	Page No.
1.	PCB DESIGNING PROCESS	10
2.	DRILLING	12
3.	PIN DIAGRAM	16
4.	ATMEGA 168	18
5.	GSM PLMN REFERENCE MODEL	28
6.	GSM SECURITY	30
7.	CALL ROUTING FOR A MOBILE TERMINATING CALL	31
8.	GSM MODULE	34
9.	EXPRESS SCH OVERVIEW	37
10.	LCD INTERFACING DIAGRAM	43

List of Tables

S.No.	Title	Page No.
1.	AT COMMAND SET OVERVIEW.	35.
2.	LCD PIN DESCRIPTION	41

Abstract

How a reliable and an authentic wireless communication could be easily developed between a mobile phone and microcontroller using GSM (Global System for Mobile Communication) MODEM (Modulator-Demodulator). This explains GSM based e-notice board which can be widely used for multitude of applications including educational sector, traffic control, banks, public advertisements, stock exchanges etc. Moreover we can also learn as well as modify some of the common applications of GSM MODEM as per the requirements and needs of the user. Here we will learn the hardware behind the picture. In addition to this I will tell you how to upgrade the software resulting low NRE (Non-Recurring Engineering) cost as well as low time-to-market. This technical paper plays a vital role in the state-of-the-art scenario where market window is relentlessly shrinking and really need pocket-friendly and authentic products.

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION TO THE PROJECT

The project mainly focuses on transmission of textual data through air interface by the use of the GSM through asynchronous serial communication. The data will be processed by the microprocessor on both ends. The data will be displayed on the lcd screen

Actually what happens is sending sms through phone has become very popular and if we can use this sms to control devices and in displaying data is possible. To receive or decode the sms globally by using gsm, by any part of the we can control and display data on lcd board.

The system required for the purpose is nothing but a microprocessor based sms box. The components are integrated with the display board and thus incorporates the wireless features.

The GSM modem receives the sms the AT commands are serially transferred to the modem. In return the modem transmit the stored message through COM port the microcontroller validates the sms and then display the message in the lcd board.

The main focus of this on displaying information to a dedicated lcd by any part of the world using gsm network which facilitates to control any message board globally from any location.

CHAPTER 2

PCB DESIGNING & FABRICATION

2.1 INTRODUCTION

A printed circuit board, or PCB, is used to mechanically support and electrically connect electronic components using conductive pathways, tracks or signal traces etched from copper sheets laminated onto a non-conductive substrate. It is also referred to as printed wiring board (PWB) or etched wiring board. Printed circuit boards are used in virtually all but the simplest commercially produced electronic devices.

A PCB populated with electronic components is called a printed circuit assembly (PCA), printed circuit board assembly or PCB Assembly (PCBA). In informal use the term "PCB" is used both for bare and assembled boards, the context clarifying the meaning.

Alternatives to PCBs include wire wrap and point-to-point construction. PCBs must initially be designed and laid out, but become cheaper, faster to make, and potentially more reliable for high-volume production since production and soldering of PCBs can be automated. Much of the electronics industry's PCB design, assembly, and quality control needs are set by standards published by the IPC organization.

PCBs are inexpensive, and can be highly reliable. They require much more layout effort and higher initial cost than either wire-wrapped or point-to-point constructed circuits, but are much cheaper and faster for high-volume production. Much of the electronics industry's PCB design, assembly, and quality control needs are set by standards that are published by the IPC organization.

PCB board design defines the electrical pathways between components. It is derived from a schematic representation of the circuit. When it is derived, or imported from a schematic design, it translates the schematic symbols and libraries into physical components and connections.

2.2 FLOW CHARTS FOR STEPS OF PCB DESIGN:

PCB Designing includes the following steps:-

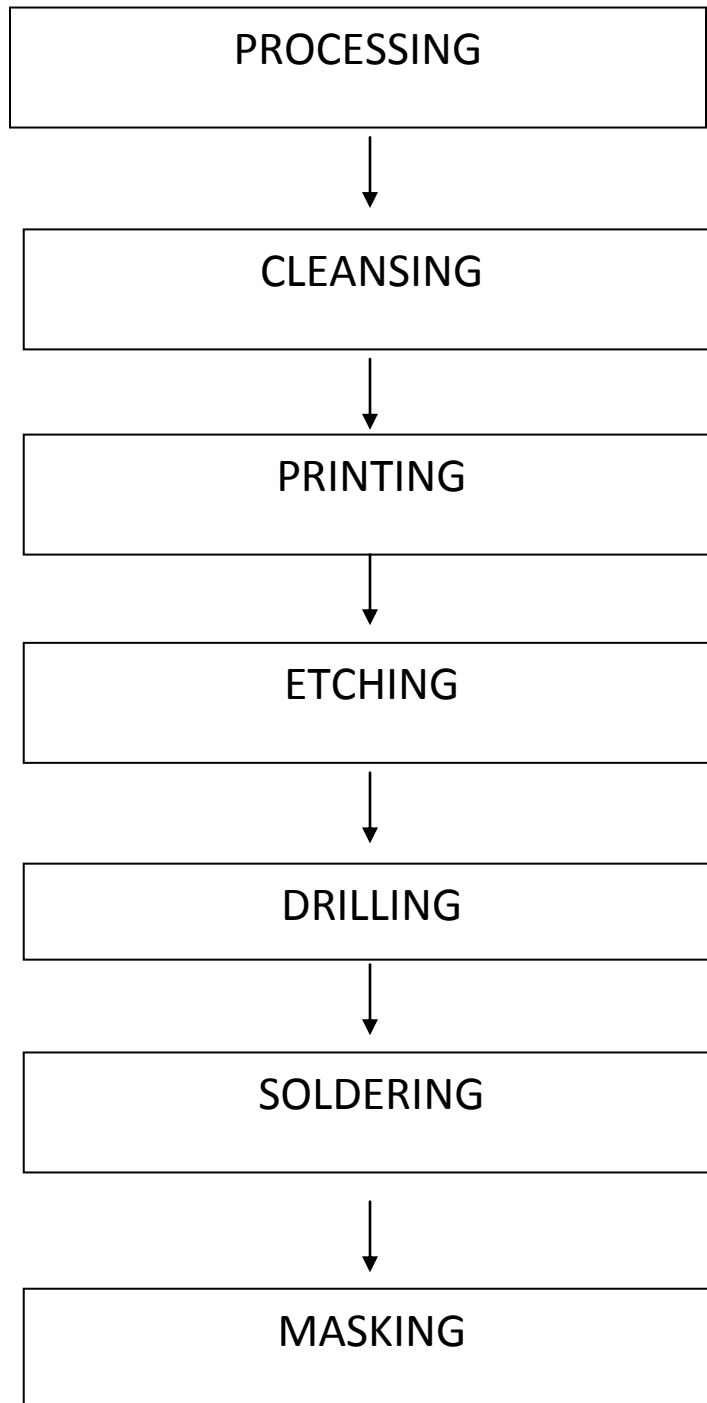


Fig:1 PCB Designing process

2.2.1 PROCESSING

The layout of a PCB has to incorporate all the information on the board before one can go on to the artwork preparation. This means that a concept that clearly defines all the details of the circuit and partly also of the final equipment, is a prerequisite before the actual layout can start. The detail circuit diagram is very important for the layout designer and he must also be familiar with the design concept and with the philosophy behind the equipment.

2.2.2 CLEANING

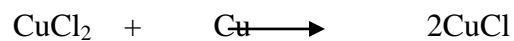
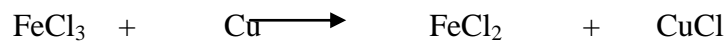
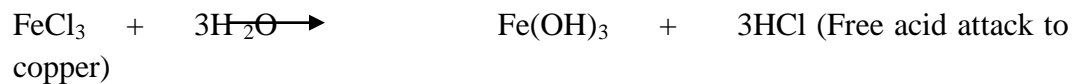
The cleaning of the copper surface prior to resist application is an essential step for any type of PCB process using etches or plating resist. After scrubbing with the abrasive, a water rinse will remove most of the remaining slurry.

2.2.3 ETCHING

It is of utmost importance to choose a suitable Etchant Systems. There are many factors to be considered:-

- ❖ Etching speed
- ❖ Copper solving capacity
- ❖ Etchant price
- ❖ Pollution character

Reactions Involved:-



2.2.4 DRILLING

Drilling is a cutting process that uses a drill bit to cut or enlarge a hole of circular cross-section in solid materials. The drill bit is a rotary cutting tool, often multipoint. The bit is pressed against the work-piece and rotated at rates from hundreds to thousands of revolutions per minute. This forces the cutting edge against the work-piece, cutting off chips from what will become the hole being drilled.

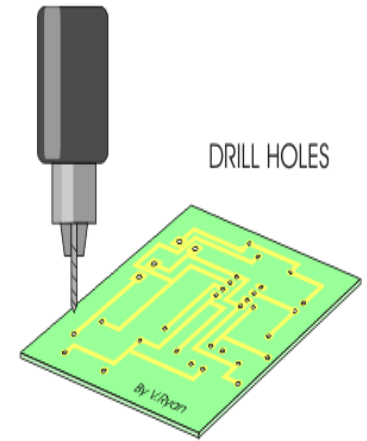


Fig 2:DRILLING

2.2.5 COMPONENTS PLACEMENT

- ❖ Component placement is an extremely important function of the designer
- ❖ Components should be placed according to their connections to other components, thermal considerations, mechanical requirements, as well as signal integrity and rout- ability
- ❖ Components which have connections to each other should be placed in the same vicinity
- ❖ For example, a processor should be placed very close to the RAM and Flash ICs on which it relies
- ❖ Components should also be placed on a grid, usually a 100 mil grid, in order to provide for a symmetric flow of routing where tracks and components are lined up

2.2.6 MASKING

It is done for the protection of conductor track from Oxidation. Solder mask or solder resist is a lacquer-like layer of polymer that provides a permanent protective coating for the copper traces of a printed circuit board (PCB) and prevents solder from bridging between conductors, thereby preventing short circuits. Solder mask was created primarily to facilitate wave soldering used in mass assembly. Solder mask is traditionally green but is now available in many colors.

Solder mask comes in different media depending upon the demands of the application. The lowest-cost solder mask is epoxy liquid that is silkscreened through the pattern onto the PCB. Other types are the liquid photo-image able solder mask (LPSM) inks and dry film photo-image able solder mask (DFSM). LPSM can be silkscreened or sprayed on the PCB, exposed to the pattern and developed to provide openings in the pattern for parts to be soldered to the copper

pads. DFSM is vacuum- laminated on the PCB then exposed and developed. All three processes go through a thermal cure of some type after the pattern is defined.

2.3 DESIGNING OF THE PCB LAYOUT

A PCB layout is required to place components on the PCB so that the component area can be minimized and the components can be placed in an efficient manner. The components can be placed in two ways, either manually or by software. The manual procedure is quiet cumbersome and is very inefficient. The other method is by the use of computer software. This method is advantageous as it saves time and valuable copper area. There are various software's available for this purpose like-

- Express PCB
- Pad2pad
- Protel PCB
- PCB design e.t.c.

Many of them are loaded with auto routing and auto placement facility. The software that we have used here is EXPRESS PCB. This software has a good interface, easy editing options and a wide range of components.

Express P.C.B.

Express PCB is a very easy to use Windows application for laying out printed circuit boards. There are two parts to Express PCB, Express SCH for drawing schematics and Express PCB for designing circuit boards.

There are lots of functions available in the software. This software is free of cost and it is very easy to use. The different layers of the PCB can be viewed by just a click of a button on the interface. And we easily get its print on paper which is utilized for further processing. We can design single sided PCB as well as Double Sided PCB with this Software.

CHAPTER 3

ATMEGA168

3.1 INTRODUCTION

The computer on one hand is designed to perform all the general purpose tasks on a single machine like you can use a computer to run a software to perform calculations or you can use a computer to store some multimedia file or to access internet through the browser, whereas the microcontrollers are meant to perform only the specific tasks, for e.g., switching the AC off automatically when room temperature drops to a certain defined limit and again turning it ON when temperature rises above the defined limit.

3.2 FEATURES

- RISC Architecture with CISC Instruction set
- Powerful C and assembly programming
- Scalable
- Same powerful AVR microcontroller core
- Low power consumption
- Both digital and analog input and output interfaces

3.3 DESCRIPTION

The Atmel ATmega48/88/168 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega48/88/168 achieves throughputs approaching 1 MIPS per MHz allowing the system designed to optimize power consumption versus processing speed.

The Atmel ATmega48/88/168 provides the following features: 4K/8K/16K bytes of In-System Programmable Flash with Read-While-Write capabilities, 256/512/512 bytes EEPROM, 512/1K/1K bytes SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte-oriented 2-wire Serial Interface, an SPI serial port, a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable Watchdog Timer

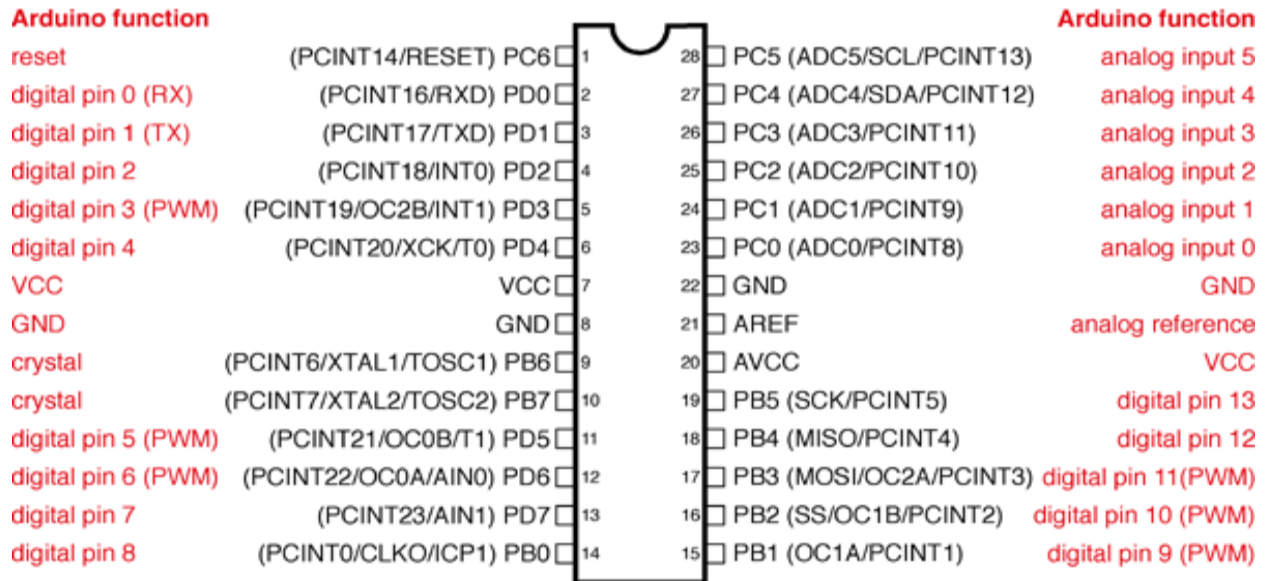
with internal Oscillator, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, USART, 2-wire Serial Interface, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption.

The ATmega48, ATmega88 and ATmega168 differ only in memory sizes, boot loader support, and interrupt vector sizes. Table 2-1 summarizes the different memory and interrupts vector sizes for the three devices.

ATmega88 and ATmega168 support a real Read-While-Write Self-Programming mechanism. There is a separate Boot Loader Section, and the SPM instruction can only execute from there. In ATmega48, there is no Read-While-Write support and no separate Boot Loader Section. The SPM instruction can execute from the entire Flash.

3.4 PIN DIAGRAM AND DESCRIPTION

Atmega168 Pin Mapping



Digital Pins 11, 12 & 13 are used by the ICSP header for MISO, MOSI, SCK connections (Atmega168 pins 17, 18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

FIG3:PIN DIAGRAM

PIN DESCRIPTION:

VCC: Digital supply voltage.

GND: Ground.

Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source Capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running. Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit. Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier. If the Internal

Calibrated RC Oscillator is used as chip clock source, PB7.6 is used as TOSC2.1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5.0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

PC6/RESET:

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is un-programmed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guarantee to generate a reset.

Port D (PD7:0):

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

AVCC:

AVCC is the supply voltage pin for the A/D Converter PC3:0, and ADC7:6. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter. Note that PC6.4 use digital supply voltage, VCC.

AREF: AREF is the analog reference pin for the A/D Converter.

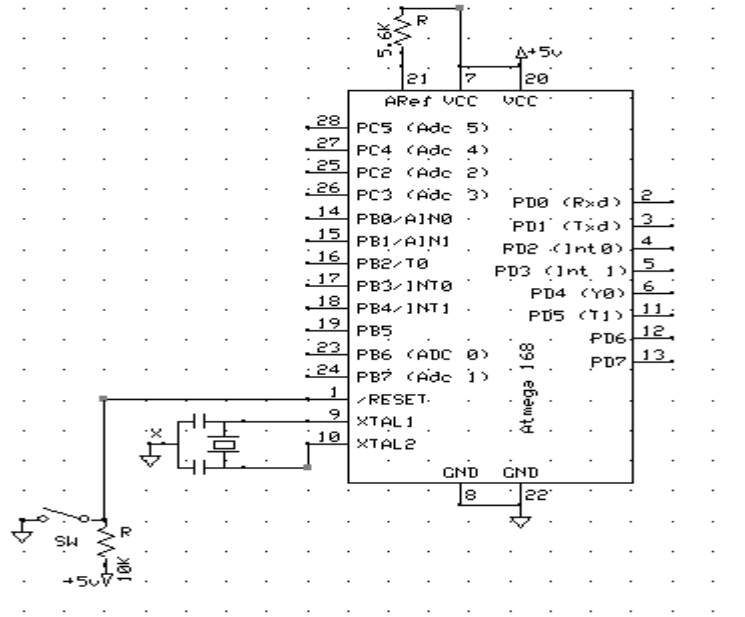


FIG 4:ATMEGA 168

3.5 THINGS TO REMEMBER ABOUT ATMEGA168 CONTROLLERS

The ATMEGA controllers are strong controllers but you have to take some small points in mind always like:

- When you go for the programming of atmega168 consider the pin no. as configures in red color in Pin diagram shown before (like controllers pin number 2 is digital pin number 0 for input or output ,pin23 is analog pin A0). Sao you will address the pin according to that number.
- Use the proper pin for proper input output interface that analog input should be configured at analog pin analog output should be configured on PWM pins and likewise the digital inputs and outputs

3.6 SELECTION OF MICROCONTROLLER

There are lots of factors which govern which microcontroller should be selected

Technical factors:

- 8-bit or 16-bit or 32-bit ALU(type of data acquisition you need)
- Program storage architecture—Harvard or Princeton memory and the required total external and internal memory up to or more

- Intensive computations at a fast rate requirement if any
- Internal EEPROM/ Flash/ROM/EPROM.
- Type of input/output system to be interfaced (digital or analog).
- If the advance features is available in the controller which you need like UART (for serial communication, ADC & PWM channels (for analog input output).

Additional factors:

- Cost when single chip and when MCU (micro-controller unit) interfaces to circuit with some features externally added.
- Major building hardware blocks and their cost and availability.
- Major building blocks of software, their cost and availability and required Hardware–software tradeoff.
- Ease of integration
- Availability of design team expertise
- Availability and cost of development software (compilers) and hardware tools
- Ease and availability of testing and debugging facilities
- Easy and reliable availability of the MCU, development tools and building blocks for hardware and software

Keeping these factors in mind the most common controller on which the student can work easily are 8051 family and AVR family of microcontrollers.

3.7 APPLICATION SCENERIO FOR MICROCONTROLLER

Of all the improvements and changes made in the field of electronics over the past few decades, microcontroller technology might be the most satisfying. These remarkable devices are used in everything from microwave ovens and cars to televisions and computer printers. In fact, it would be difficult to put together a list of microcontroller uses without missing a few.

Some modern devices, such as cars and more complex machinery use multiple microcontrollers. If you have heard the terms “brain” or “computer” applied to the overall operation of a car’s

engine, you have heard about the microcontroller that keeps most automobile systems in operation.

The best way to get a vision of a microcontroller is to think of it as a small computer. In fact, this tiny but important object has many of the same features and benefits of larger, more complex computers. They have a central processing unit (referred to as the CPU) in which the software programs work. Computers execute the programs to accomplish various tasks. Each computer has a storage device, such as a hard drive, a flash drive or a CD. Information from this larger storage device is stored in part in the random-access memory. A microcontroller has all of these components in some form. It's designed to be small and it's designed to be specific to a certain task or use. They are also solid-state, integrated circuits. This separates them from larger units such as a personal computer that can perform a number of different jobs for the user. Because they are so small, microcontrollers can be part of a larger device, such as the television, car or appliance mentioned earlier. They are programmed to control specific activity within that larger device. Their memory storage and programs are also very specific to the task at hand.

Most microcontrollers use very low amounts of electrical power and don't have moving parts, large input devices such as a keyboard or big display screens. In some cases, the microcontroller shows that it is working by powering a small light-emitting diode (LED) or simply by the expected activity taking place.

One good example might be an entertainment center in the home. If a person operates the CD player, television set or other device by remote control a microcontroller (tiny computer) receives information from the remote control device and transfers that information to the right place within the larger electronic system. This same type of activity takes place in a car when the key is turned, when one of the accessories is switched etc. A microcontroller takes input from these points and takes the input on to the next step.

Smaller devices might use simpler microcontrollers that have been in efficient use for decades. In more complicated systems, the microcontroller might be a bit more powerful, yet still available at low cost and using very little electrical power to perform their tasks. While larger computer chips and storage drives might have millions of bytes of capability, some microcontrollers used in modern electronic devices have only 1,000-byte storage capability. But with such specific tasks this is all that's needed.

The microcontrollers are now a very important part of the electronic world starting from mini gadgets to large scale industrial automations

3.8 IMPORTANT FACT ABOUT THE MICROCONTROLLER

There are certain microcontrollers in which for running the program first you have to transfer a boot-loader to the microcontroller. Generally, when you want to program a microcontroller, you need a programmer for that particular microcontroller. Apart from being expensive, it might have other disadvantages, like long programming times or too many interconnections between the micro and the programmer itself; some programmers are not capable of programming in circuit. Many of these problems are addressed by a boot-loader. So, the boot-loader has to be programmed in the program memory of the microcontroller just once, using a conventional programmer. After this, the microcontroller can be programmed without a programmer. Once in the microcontroller, the boot-loader is such programmed that each time after reset it starts running like any conventional program. What it does however is different from a regular program. First of all, depending on what type of boot-loader it is, it starts “listening” for incoming bytes via a specific interface. For instance, a UART boot-loader will listen to the UART buffer of the micro, checking for incoming bytes. If the bytes start arriving, the boot-loader will grab them and write them in the program memory in the sequence it receives them and at predefined locations. Once all bytes have been received, the boot-loader executes a jump at the start of the memory zone it has received and then the “normal” program starts running.

Programming or burning: Burning means transfer of the code from PC to the microcontroller through different communication medium

- Programming or burning a microcontroller means to transfer the program from the compiler to the memory of the microcontroller. A compiler is software which provides an environment to write, test and debug a program for the microcontroller. The program for a microcontroller is generally written in C or assembly language. Finally the compiler generates a hex file which contains the machine language instruction understandable by a microcontroller. It is the content of this hex file which is transferred to the memory of the microcontroller. Once a program is transferred or written in the memory of the microcontroller, it then works in accordance with the program.
- In order to program a microcontroller we need a device called a burner/programmer. A programmer is a hardware device with dedicated software which reads the content of the hex file stored on the PC or the laptop and transfers it to the microcontroller to be burned. It reads the data of the hex file by connecting itself to the PC via a serial or USB cable and transfers the data to the

memory of the microcontroller to be programmed in accordance with the protocols as described by the manufacturer in the datasheet.

- The programmer and the compiler differ for microcontrollers from different companies. In some cases the compiler has programmer software inbuilt in it. You simply need to connect the programmer hardware and the microcontroller can be programmed from the compiler itself. Examples of some compiler are MIKRO-C, KEIL, AVR STUDIO, ARDUINO and many more.

CHAPTER 4

GSM

4.1 INTRODUCTION

GSM (Groupe Spéciale Mobile, now named Global System for Mobile Communications) is a 2nd Generation cellular mobile system innovated in Europe by ETSI (European Telecommunications Standardization Institute). It is now a “family name” of a number of systems including 2.5G and 3G systems such as:

- GPRS (General Packet Radio Services) – GSM enhancement in cell-phone software and network hardware and
- software to support packet switching (the one used by the Internet)
- EDGE (Enhanced Data rates for GSM Evolution) – a technique to achieve better ‘compression’ of data on the airinterface
- EGPRS (EDGE based GPRS)

UMTS (Universal Mobile Telecommunications Service) – 3G System based on W-CDMA (Wideband Code Division Multiple Access)

4.2 CELLULAR MOBILE SYSTEMS

Cellular radio was devised in order to make better use of limited resource of Radio Spectrum. Each Megahertz of spectrum will only support a comparatively a small number of simultaneous conversations and the same frequency must be reused many times in order to meet the capacity needed for national or regional service. Cellular radio achieves this by creating a honeycomb of cells over the region and assigning frequencies and power levels in such a way that the same frequency can be reused for different conversation two or three cells away. Although most of the countries in Europe had their own analogue cellular system, they were not compatible with each other. So with the development in Political, Commercial and Industrial areas there arose a necessity for uniformity in cellular communication Cellular mobile communication has generations:

1. **First generation:** Analog radio; mostly telephony only, virtually no data capability other than special device with analog modem. This generation is now mostly obsolete.
2. **Second Generation:** Digital radio and short messaging; this is now the main stream system. Recently a variety of technique has been innovated and employed to enhance data capability of 2G systems. The data capability includes Internet access and picture sharing. These systems are called 2.5G systems. High end cell phones and services are now available with 2.5G capabilities.
3. **Third Generation:** Digital system with multimedia services including video phone and relatively higher speed (say up to 1 Mbps) Internet access. A slow roll off of 3G system has been started in advanced networks of developed and rapidly developing countries.
4. **Fourth Generation:** Digital system with voice-over-IP (VOIP) technology (Note that the voice for G1, 2 and 3 are circuit-switched). That is, the services are integrated into all IP network. This is expected to be future network and not coming any time soon.

4.3 GSM TECHNOLOGY

One of the most important conclusions from the early tests of the new GSM technology was that the new standard should employ Time Division Multiple Access (TDMA) technology. This ensured the support of major corporate players like Nokia, Ericsson and Siemens, and the flexibility of having access to a broad range of suppliers and the potential to get product faster into the marketplace. After a series of tests, the GSM digital standard was proven to work in 1988.

With global coverage goals in mind, being compatible with GSM from day one is a prerequisite for any new system that would add functionality to GSM. As with other 2G systems, GSM handles voice efficiently, but the support for data and Internet applications is limited. A data connection is established in just the same way as for a regular voice call; the user dials in and a circuit-switched connection continues during the entire session. If the user disconnects and wants to re-connect, the dial-in sequence has to be repeated. This issue, coupled with the limitation that users are billed for the time that they are connected, creates a need for packet data for GSM.

The digital nature of GSM allows the transmission of data (both synchronous and asynchronous) to or from ISDN terminals, although the most basic service support by GSM is telephony. Speech, which is inherently analog, has to be digitized. The method employed by ISDN, and by current telephone systems for multiplexing voice lines over high-speed trunks and optical fiber lines, is Pulse Coded Modulation (PCM). From the start, planners of GSM wanted to ensure

ISDN compatibility in services offered, although the attainment of the standard ISDN bit rate of 64 Kbit/s was difficult to achieve, thereby belying some of the limitations of a radio link. The 64 Kbit/s signal, although simple to implement, contains significant redundancy.

Since its inception, GSM was destined to employ digital rather than analog technology and operate in the 900 MHz frequency band. Most GSM systems operate in the 900 MHz and 1.8 GHz frequency bands, except in North America where they operate in the 1.9 GHz band. GSM divides up the radio spectrum bandwidth by using a combination of Time- and Frequency Division Multiple Access (TDMA/FDMA) schemes on its 25 MHz wide frequency spectrum, dividing it into 124 carrier frequencies (spaced 200 KHz apart). Each frequency is then divided into eight time slots using TDMA, and one or more carrier frequencies are assigned to each base station. The fundamental unit of time in this TDMA scheme is called a 'burst period' and it lasts 15/26 ms (or approx. 0.577 ms). Therefore the eight 'time slots' are actually 'burst periods', which are grouped into a TDMA frame, which subsequently form the basic unit for the definition of logical channels. One physical channel is one burst period per TDMA frame

The development of standards and systems spans well beyond the technical realm and often into the political; this is best exemplified by what happened with GSM. Shortly after the suitability of TDMA for GSM was determined, a political battle erupted over the question of whether to adopt a wide-band or narrow-band TDMA solution. Whereas France and Germany supported a wide-band solution, the Scandinavian countries favored a narrow-band alternative. These governmental preferences were clearly a reflection of the respective countries' domestic equipment manufacturers as German and French manufacturers SEL and Alcatel had invested substantially into wide-band technology, whereas their Scandinavian counterparts Ericsson and Nokia poured resources into the narrow-band alternative. Italy and the UK, in turn, were the subjects of intense lobbying on behalf of the two camps with the result of frequently changing coalitions

The culmination of this controversy between the two camps was a CEPT (Conference des Administrations Europeenes des Posts et Telecommunications) Meeting in Madeira in February 1987. The Scandinavian countries finally convinced Italy, the UK and a few smaller states of the technical superiority of narrow-band technology and left Germany and France as the only proponents of the wide-band alternative. Since CEPT followed purely intergovernmental procedures, however, decisions had to be taken unanimously, and Germany and France were able to veto a decision that would have led to the adoption of narrow-band TDMA as the technology underlying the GSM project.

A unique feature of GSM is the Short Message Service (SMS), which has achieved wide popularity as what some have called the unexpected 'killer application' of GSM. SMS is a bi-directional service for sending short alphanumeric message in a store-and-forward process. SMS can be used both 'point-to-point' as well as in cell-broadcast mode Supplementary services are

provided on top of tele-services or bearer services, and include features such as, inter alia, call forwarding, call waiting, caller identification, three-way conversations, and call-barring.

The most novel and far-reaching feature of GSM is that it provides most of Europe's cellular phone users with a choice – choice of network and choice of operator. Also, international roaming was and continues to be the cornerstone of GSM. For this to be possible, all networks and handsets have to be identical. With many manufacturers creating many different products in many different countries, each type of terminal has been put through a rigorous approval regime. However, at the time, no approval process was available, and it took nearly a year before the handheld terminals were tested and fit for market entry.

Another of GSM's most attractive features is the extent to which its network is considered to be secure. All communications, both speech and data, are encrypted to prevent eavesdropping, and GSM subscribers are identified by their Subscriber Identity Module (SIM) card (which holds their identity number and authentication key and algorithm). While the choice of algorithm is the responsibility of individual GSM operators, they all work closely together through the Memorandum of Understanding (MOU) (to be described in greater detail in section 2.2.2) to ensure security of authentication. This smartcard technology minimizes the necessity for owning terminals - as travelers can simply rent GSM phones at the airport and insert their SIM card. Since it's the card rather than the terminal that enables network access, feature access and billing, the user is immediately on-line.

SYSTEM ARCHITECTURE

The functional architecture of a GSM system can be broadly classified into

- Mobile Station (MS)
- Base Station Subsystem (BSS)
- Network and Switching Subsystem (NSS)
- Operation Subsystem (OSS)

The MS and the BSS communicate via the Um interface or radio link. The BSS communicates with Mobile Service Switching Center across the A interface.

4.4 MOBILE STATION:

This may be a standalone piece of equipment for certain services or support the connection of external terminals. The MS consists of the Mobile Equipment (ME) and a Subscriber Identity Module (SIM). The ME is uniquely identified by the International Mobile Equipment Identity (IMEI), but it need not be personally assigned to one subscriber, The SIM which is a smart card

provides personal mobility and the user can access the subscriber services. The subscriber can operate on any terminal just by inserting the SIM card in that GSM terminal. SIM card contains the International Mobile Subscriber Identity (IMSI) used to identify the subscriber to the system, a secret key for authentication and other security information's. SIM card may be protected against unauthorized use by a password.

BASE STATIONSUBSYSTEM: The BSS is composed of three parts, Base Transceiver Station (BTS) and Base Station Controller (BSC). These two communicate across the standardized Abis interface. The third part is Transponder and Rate Adaptation Unit (TRAU).

Base Transceiver Station: This provides the GSM radio coverage within a cell. It comprises of radio transmitting and receiving equipment and associated signal processing units. This complements the radio features of ME.

Base Station Controller: This manages the radio resources for one or more BTS's. It handles radio channel set-up, Handovers and frequency hopping. Handovers between BTS's belonging to different BSC's however can involve MSC's but are still managed by the original serving BSC. It controls the transmission of information about Local Area Codes (LAC), signaling channel configuration and information's about neighboring cells.

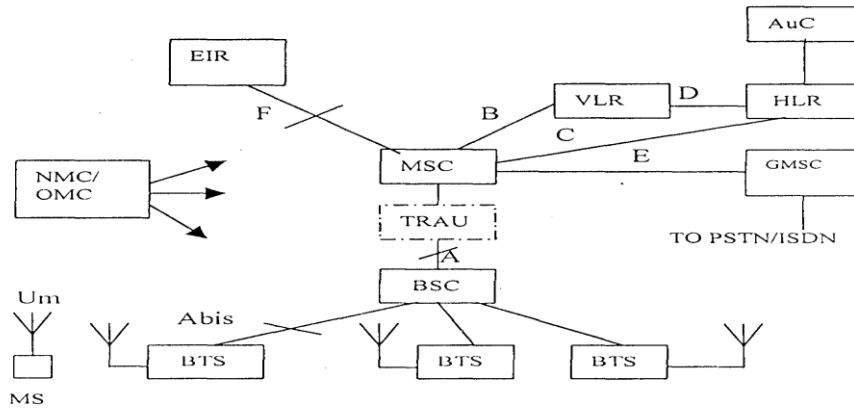


Fig. 1. GSM PLMN reference model

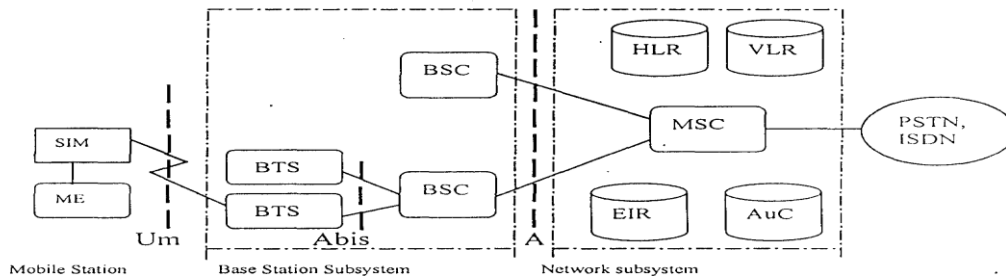


FIG 5: GSM PLMN REFERENCE MODEL

Trans-coder and Rate Adaptation Unit: This is responsible for trans-coding between GSM encoded speech at 13KPS and fixed network speech at 64KPS. Similarly it performs rate adaptation of GSM data services. Although it is a part of BSS, it is located at MSC Sites. This is to benefit from the lower rate coding and consequent saving in transmission costs.

4.5 NETWORK SYSTEM AND SWITCHING

NSS in GSM uses an Intelligent Network (The central component of NSS is the Mobile Service Switching Center (MSC). It is supported by Interworking functions (JWF), Home Location Register (HLR), Visitor Location Register (VLR), Gateway MSC (GMSC) and Signal Transfer Point (STP).

Mobile Service Switching Center: It acts like a switching node and additionally provides all the functionality needed to handle a mobile subscriber such as registration, authentication, location updating. Handovers and call routing to a roaming subscriber. These functions are provided in conjunction with several functional entities. An MSC controls several BSC's.

Inter Working Function, (IWF): A gateway for MSC to interface with external networks for communications with users outside GSM. The role of IWF depends upon the type of user data and the network to which it interfaces.

Home Location Register (HLR,): It consists of a computer without switching capabilities. It is a database, which contains subscriber information related to the subscriber's current locations but not the actual location. HLR has two divisions Authentication Center (AuC) and Equipment Identity Register (EIR). The AuC manages the security data for subscriber authentication. The EIR database carry information about certain ME's. The security procedure is discussed later.

Visitor Location Register VLR,):: It links to one or more MSC's, temporarily storing subscription data currently served by its corresponding MSC. VLR holds more current subscriber location than I—ILR. Although VLR is an independent unit, it is always implemented together with the MSC.

Gateway MSC GMSC): In order to set-up a requested call, the call is initially routed to a GMSC which finds the correct HLR.GMSC has an interface with external network for gate - waying and the network operates the full signaling system 7 (SS7) between NSS Machines.

SIGNALING TRANSFER POINT: IT ACTS AS A STANDALONE NODE TO OPTIMIZE THE COST OF THE SIGNALING TRANSPORT AMONG MSC/VLR, GMSC AND HLR> MOBILITY MANAGEMENT:

This layer handles the functions that arise from the mobility of the subscribers as well as the authentication and security aspects.

4.6 LOCATION UPDATING: The mobile phone receives constantly information send by the network. This information includes identification of the area where the mobile is currently located. In order to keep track of its location, the mobile stores the ID of the area in which it is currently registered. Every time the network broadcasts the ID of the area, the mobile compares this, information to the area ID stored in its memory. When the two ID's are no longer the

same, the mobile sends the network a request, i.e. Registration inquiry to the area it has entered. The network receives the request and registers the mobile in the new area, a new VLR. Simultaneously, a subscriber HLR is informed about the new location and the data of the subscriber is cleared from the previous VLR.

4.7 AUTHENTICATION AND SECURITY: Authentication involves two functional entities: -SIM and AuC. Each subscriber is given a secret key, one copy of which is stored in SIM card and the other in AuC. During Authentication, the AuC, a 128 bit random number RAND) that is send o lie Mobile. Both mobile and the AuC then uses the RAND in conjunction with the subscribers secret key and a ciphering algorithm cal lcd A3, to generate a 32 bit long signal response (SRES) that is send back to AuC. If both numbers match the subscriber is authenticated.

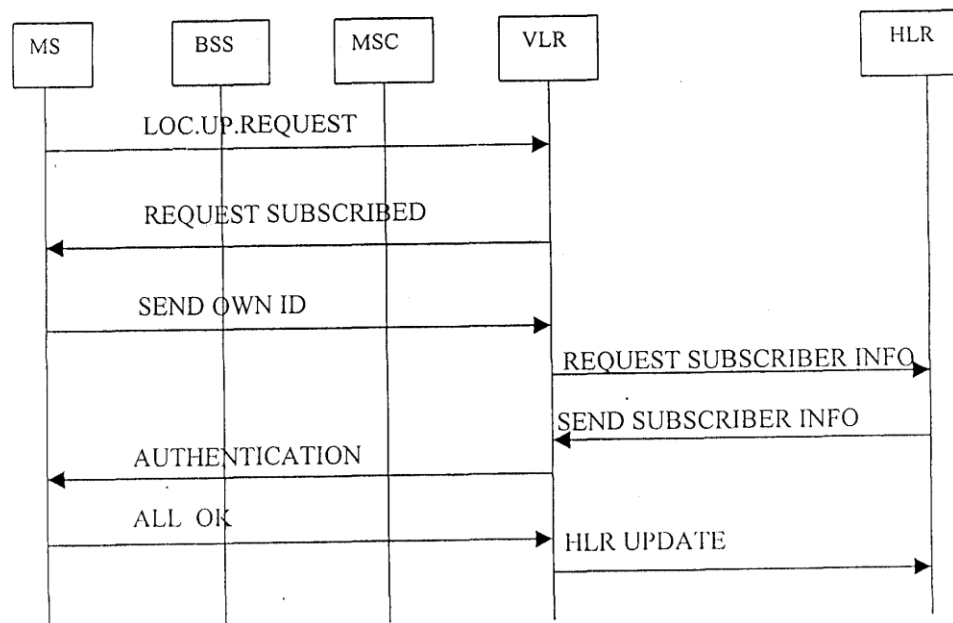


FIG 6: SECURITY IN GSM

4.8 CALL MANAGEMENT

This layer is responsible for call control, supplementary service management and short message service management.

4.9 CALL CONTROL: Call control functions set up calls, maintain calls and release calls.

4.10 CALL ROUTING: The directory number dialed to reach a mobile subscriber is called the mobile subscriber ISDN (MSISDN) which is defined by E.164 numbering plan. This number includes country code and national destination code, which identifies the subscriber's operator. The first few digits of the remaining subscriber number may identify the subscriber's HLR.

An incoming mobile terminating call is directed to the GMSC. GMSC contains a table linking MSISDN to their corresponding HLR. The GMSC queries the called subscriber's HLR for a Mobile Station Roaming Number (MSRN). The HLR typically stores only the SS7 address of the subscriber's current VLR and so HLR queries the current VLR, which will temporarily allocate an MSRN from its pool for the call. This MSRN is returned to the HLR and back to GMSC, which can then route the call to the new MSC. At the new MSC the IMSI corresponding to the MSRN is looked up and the mobile is paged in its current location area.

4.11 SUPPLEMENTARY SERVICE MANAGEMENT

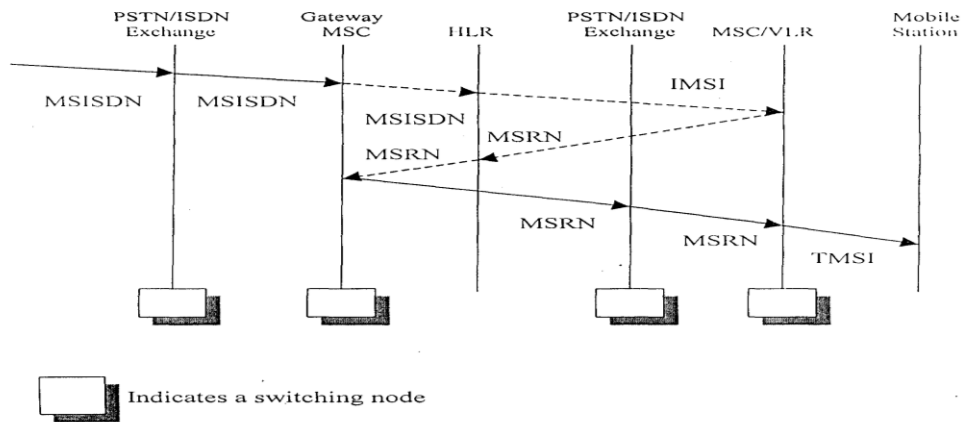


Fig.4. Call routing for a mobile terminating call

FIG 7 :CALL ROUTING FOR A MOBILE TERMINATING CALL

4.12 STRENGTH OF GSM

- Compatibility with ISDN using rate adaptation box.
- Use of SIM cards.
- Control of transmission power.
- Frequency hopping.

- Discontinuous transmission.
Mobile assisted handover

4.13 SERVICES PROVIDED BY THE GSM

Basic service supported by GSM is telephony. The speech is digitized and then transmitted. There is also an emergency service. Data services offered by GSM are- Data transfer at 9.6KBPS, ISDN, PSPDN, CSPDN, group3 facsimile and short message service. In addition to voice calls, GSM networks also provide many other services, and we are familiar with most of them. There's the ubiquitous SMS-the capability to send short text messages from one phone to another. There are the data services, which exploit GSM's digital nature. This allows one to access the Internet from a GSM phone, and many service providers are offering this service. You can send email from the handset and lots more. One limitation of GSM is that currently it has a maximum data transfer rate of only 9 kbps but new development like HSCSD (high-speed circuit switched data) and GPRS (general packet-switched radio) are pushing this up quite a bit.

4.14 APPLICATIONS

Access control devices: Now access control devices can communicate with servers and security staff through SMS messaging. Complete log of transaction is available at the head-office Server instantly without any wiring involved and device can instantly alert security personnel on their mobile phone in case of any problem.

Transaction terminals: EDC machines, POS terminals can use SMS messaging to confirm transactions from central servers. The main benefit is that central server can be anywhere in the world. Today you need local servers in every city with multiple telephone lines. You save huge infrastructure costs as well as per transaction cost.

Supply Chain Management: Today SCM require huge IT infrastructure with leased lines, networking devices, data centre, workstations and still you have large downtimes and high costs. You can do all this at a fraction of the cost with GSM M2M technology. A central server in your head office with GSM capability is the answer, you can receive instant transaction data from all your branch offices, warehouses and business associates with nil downtime and low cost.

4.15 WHAT APPLICATION ARE SUITABLE FOR GSM ?

If your application needs one or more of the following features, GSM will be more cost-effective than other communication system

Short Data Size: Your data size per transaction should be small like 1-3 lines. e.g. banking transaction data, sales/purchase data, consignment tracking data, updates. These small but important transaction data can be sent through SMS messaging which cost even less than a local telephone call or sometimes free of cost worldwide. Hence with negligible cost you are able to send critical information to your head office located anywhere in the world from multiple points. You can also transfer faxes, large data through GSM but this will be as or more costly compared to landline networks.

Multiple Remote Data Collection Points: If you have multiple data collection points situated all over your city, state, country or worldwide you will benefit the most. The data can be sent from multiple points like your branch offices, business associates, warehouses, and agents with devices like GSM modems connected to PCs, GSM electronic terminals and Mobile phones. Many a times some places like warehouses may be situated at remote location may not have landline or internet but you will have GSM network still available easily.

High Uptime: If your business require high uptime and availability GSM is best suitable for you as GSM mobile networks have high uptime compared to landline, internet and other communication mediums. Also in situations where you expect that someone may sabotage your communication systems by cutting wires or taping landlines, you can depend on GSM wireless communication.

Large Transaction Volumes: GSM SMS messaging can handle large number of transaction in a very short time. You can receive large number SMS messages on your server like e-mails without internet connectivity. E-mails normally get delayed a lot but SMS messages are almost instantaneous for instant transactions. Consider situation like shop owners doing credit card transaction with GSM technology instead of conventional landlines. many a time you find local transaction servers busy as these servers use multiple telephone lines to take care of multiple transactions, whereas one GSM connection is enough to handle hundreds of transaction per minute.

Mobility, Quick installation: GSM technology allows mobility, GSM terminals, modems can be just picked and installed at other location unlike telephone lines. Also you can be mobile with GSM terminals and can also communicate with server using your mobile phone. You can just purchase the GSM hardware like modems, terminals and mobile handsets, insert SIM cards, configure software and your are ready for GSM communication. GSM solutions can be implemented within few weeks whereas it may take many months to implement the infrastructure for other technologies.

4.16 INTRODUCTION TO GSM MODULE



FIG 8:GSM MODULE

Designed for global market, SIM300 is a Tri-band GSM/GPRS engine that works on frequencies EGSM 900 MHz, DCS 1800 MHz and PCS 1900 MHz SIM300 features GPRS multi-slot class10/ class 8 (optional) and supports the GPRS coding schemes CS-1, CS-2, CS-3 and CS-tiny configuration of 40mm x 33mm x 2.85mm, SIM300 can fit almost all the spacerequirements in our applications, such as smart phone, PDA phone and other mobile devices. In this hardware SIM300 is only interfaced with RS232, Regulated power Supply 4.0V SIM Tray Antenna with LED indications.

A: SIM300 module interface

B: SIM card interface

C: headset interface

D: Download switch, turn on or off download function

E: VBAT switch, switch the voltage source from the adaptor or external battery

F: PWRKEY key, turn on or turn off SIM300

G: RESET key

H: expand port, such as keypad port, main and debug serial port, display port

I: MAIN serial port for downloading, AT command transmitting, data exchanging

J: DEBUG serial port

K: hole for fixing the antenna

- L: source adapter interface
- M: light
- N: buzzer
- O: headphones interface
- P: hole for fixing the SIM300

4.17 AT COMMANDS FORMATS

A command line is a string of characters sent from a DTE to the modem (DCE) while the modem is in a command state. A command line has a prefix, a body, and a terminator. Each command line (with the exception of the A/ command) must begin with the character sequence AT and must be terminated by a carriage return. Commands entered in upper case or lower cases are accepted, but both the A and T must be of the same case, i.e., “AT or “at. The default terminator is the ENTER key <CR> character. Characters that precede the AT prefix are ignored. The command line interpretation begins upon receipt of the ENTER character.

Characters

the command lines are parsed as commands with associated parameter values. The basic commands consist of single ASCII characters, or single characters preceded by a prefix character (e.g., “&” or “+”), followed by a decimal parameter. Missing decimal parameters are evaluated as 0.

The following section describes the AT-Command set. The commands can be tried out by connecting a GSM modem to one of the PC’s COM ports. Type in the test-command, adding CR + LF (Carriage return + Line feed = \r\n) before executing. Also

Table 2-1. AT-Command set overview

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

TABLE 1: AT COMMANDS SET OVERVIEW

4.18 GSM MODEM-PC Interfacing

GSM Modem is used to receive message from the authorized user. This GSM modem requires a SIM card from a wireless carrier in order to operate. This SIM number is contact number of the receiving section. PCs use AT commands to control modems. Although GSM modem is interfaced with PC through Bluetooth using a VB program, a GSM modem can be tested before actually implementing into the system. The MS HyperTerminal is a handy tool when it comes to testing the GSM device.

It can be found at

*Start -> Programs -> Accessories ->
Communications -> HyperTerminal.*

Various parameters like connection name, icon, comm port, and correct port settings are specified for our GSM modem. Finally in the hyper terminal main window, AT commands are sent and responds received from GSM modem can be seen. Screenshot of dialog box showing sending and receiving of AT commands is shown in figure below (Here mobile phone connection

is the connection name) .The screenshot of MS HyperTerminal's main window in Windows 98. We will first have to type "AT" A response "OK" should be returned from the mobile phone or GSM modem. Now we will type "AT+CPIN?". The AT command "AT+CPIN?" is used to query whether the mobile phone or GSM modem is waiting for a PIN (personal identification number, i.e. password). If the response is "+CPIN: READY", it means the SIM card is ready for use. After this, various AT commands depending on the used modem" s instruction set are send to modem and responses are received. This process is very useful in testing GSM modems.

CHAPTER 5

EXPRESS SCH

5.1 INTRODUCTION

Express SCH is a very easy to use Windows application for drawing schematics. While not required, we suggest that you draw a schematic for your circuit before designing the PC board

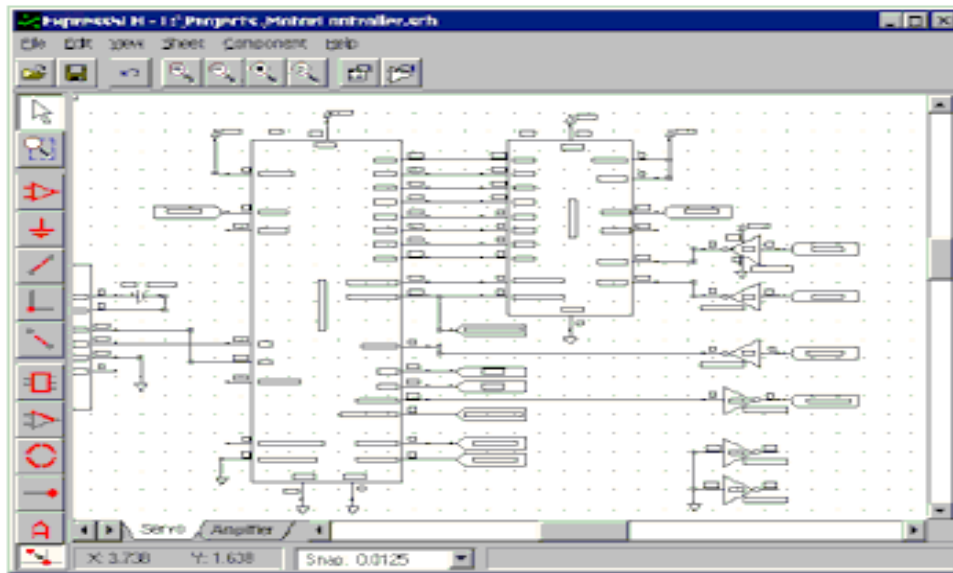


FIG 9:EXPRESS SCH OVERVIEW

5.2 COMPONENTS AND SYMBOLS

You build a schematic by first placing components and symbols on the page and then wiring them together. In Express SCH, components and symbols are different. Components represent parts that you add to your circuit such as resistors and ICs.

Symbols, such as these, are inserted into your schematic just like components. However they differ from components in that they don't have a physical part associated with them.

Symbols are used to make electrical connections to common buses (such as ground) without having to draw wires. They are also used to make connections from one page of a schematic to another. Here is how they work: All symbols are assigned a Net Name (such as GND). Every symbol placed on a schematic with the same Net Name is electrically connected together. For example, every component pin that is wired to a ground symbol (which has the Net Name GND) is connected.

5.3 PLACING THE COMPONENTS

Components represent parts that you will add to your circuit such as resistors and ICs.

The easiest way to place components in your board layout is to use the Component Manager.

Tip: Use the Find button to search for a component by its name. To place a component:

1. Click the button located on the top toolbar to display the Component Manager.
2. Select one of these categories:
 - Library components - Components that are included with the program
 - Custom components - Components that you have drawn
 - Favorite components - Components or symbols that you have book-marked
3. From the list box, choose the item to insert. Note that the components list is organized by the prefixes:
 - Connector - connectors
 - IC - integrated circuits
 - Misc - batteries, buzzers and motors
 - Passive - resistors and capacitors
 - Semiconductors - transistors and diodes
4. Select how the component and its accompanying text are rotated by clicking one of: or to flip right-to-left pick.
5. Press the Insert into schematic button, then drag the component to the desired location.
6. Double click on the item just inserted to display its properties dialog box. Here you will assign the component its Part ID (such as R1 or U1) and Part name (such as 74LS00 or 3.3K).

5.4 PLACING THE SYMBOLS,PORTS

Symbols such as these are inserted into your schematic just like components. However symbols are different from components in that they don't have a physical part associated with them. To insert a symbol in to your schematic:

1. Select the button from the side toolbar.
2. From the top toolbar, choose the symbol in the list box

3. Move the mouse into the main window, press the left mouse button and drag the symbol to the desired location.
4. After placing a port symbol, assign it a Net Name. Do this by double clicking on it to display the Symbol properties dialog box. The port symbol shown here has been assigned the Net Name: 20 MHz

Keep in mind that all symbols given the same Net Name are electrically connected together on the schematic.

5.5 WIRING COMPONENTS

From the side toolbar, select desired tab or press the W shortcut key.

Move the mouse to the wire's first end point and click left. Typically you will start on the pin of a component. Drag the wire to the second end point, then click left again. Continue placing wire segments until you have reached the final end point.

While dragging the wire, the Del key deletes the previous segment, the + and - keys zoom in and out, the G key toggles the snap-to-grid on and off, the Spacebar sets the wire, and the Esc key cancels it. To complete the operation, press the Spacebar or click right.

5.6 WORKING WITH WIRES :

Wires themselves cannot be moved. The path of a wire is determined by the straight line between its two connections. Therefore, to move a wire, you need to connect it to something different or to move what it is connected to. Corners in wires allow them to bend. They are displayed as small square blocks at the ends of wires. A wire with two corners is shown here. When you print your schematic, the corners will not be visible.

Corners can be dragged, inserted or deleted to change a wire's path. To insert a corner, select desired tab then click on the wire at the point where you want to insert it. To disconnect a wire and reconnect it elsewhere, select desired tab from the side toolbar. Next, click on the wire near the point you want to disconnect, and then drag the wire to a new pin.

CHAPTER 6

LCD

6.1 INTRODUCTION

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data. The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

Features:-

- 5 x 8 dots with cursor
- Built-in controller (KS 066 or Equivalent)
- + 5V power supply (Also available for + 3V)
- 1/16 duty cycle
- B/L to be driven by pin 1, pin 2 or pin 15, pin 16 or A.K (LED)
- N.V. optional for + 3V power supply

6.2 PIN DESCRIPTION

PIN NUMBER	SYMBOL	FUNCTION
1	V _{SS}	GND
2	V _{DD}	+ 3V or + 5V
3	V ₀	Contrast Adjustment
4	RS	H/L Register Select Signal
5	R/W	H/L Read/Write Signal
6	E	H?L Enable Signal
7	DB0	H/L Data Bus Line
8	DB1	H/L Data Bus Line
9	DB2	H/L Data Bus Line
10	DB3	H/L Data Bus Line
11	DB4	H/L Data Bus Line
12	DB5	H/L Data Bus Line
13	DB6	H/L Data Bus Line
14	DB7	H/L Data Bus Line
15	A _V EE	+ 4.2V for LED/Negative Voltage Output
16	K	Power Supply for B/L (OV)

TABLE 2:LCD PIN DESCRIPTION

6.3 SEQUENCING OF WRITING TO THE LCD

The sequence for writing to the LCD is:

1. To begin, E is low
2. Select the register to write to by setting RS high (data) or low (control)
3. Write the eight bits of data to the LCD
4. Set the Enable signal high then low again.

There are certain minimum timing requirements that must be followed when writing to the LCD, such as data setup times and Enable signal pulse width. These are in the order of tens and hundreds of nanoseconds. LCD displays have a ‘busy’ flag that is set while it is executing a control command. This flag is not accessible because the R/W line has been tied low (write). This is not such a problem because all commands have a maximum execution time. We simply wait for this time to pass before accessing the LCD again, eliminating the need to check the flag and thereby saving on an I/O line. For example: The ‘Clear Display’ command has an execution time of approx. 1.6mS. After sending this command to the LCD we simply wait 2mS before continuing on. This ensures that the command has finished. Characters to be displayed are

written into the LCD's 'data' RAM memory. The amount of RAM available depends on the type of LCD. Let's us consider a LCD having 80 bytes of RAM. An internal address counter holds the address of the next byte to write to. These 80 bytes are broken up into two blocks of 40 bytes. The address range of the first block is from 00h to 27h and the second block from 40h to 67h.

After power up and initialization address 00h is the first character of the top line and address 40h is the first character of the bottom line. The address counter is set to address 00h and is automatically incremented after each byte is written. Only the first 16 bytes of each line are visible. If we now start writing data to the LCD it will be stored from address 00h on but only the first 16 characters will be visible. To display the rest of the characters we need to 'scroll' the display.

Scrolling simply mean changing the start address of each line. If we scroll left one position address 01h becomes the first character of the top line and address 41h the first character of the bottom line. Scrolling right does the opposite – addresses 27h and 67h become the first characters of the top and bottom lines. Each block of addresses 'wrap' around in a circular manner. The next address after 27h is 00h; the next address after 67h is 40h. If we scroll left 40 times we will bring address 00h back into view as the last character on the top line.

The address ranges of each block are not continuous. There is a gap of 24 bytes between the ends of the first block (27h) and the start of the second (40h). So, if we keep writing characters to the display the first 40 will be stored in addresses 00h to 27h, the next 24 will be 'lost' and the next 40 will be stored at addresses 40h to 67h.

The internal address counter is directly accessible so we can set the next address to write to. This is how the characters are written to the second line of the LCD. For example, we can send 5 characters to the first line, set the address counter to 40h and then start writing characters to the second line

6.4 LCD INTERFACE DIAGRAM

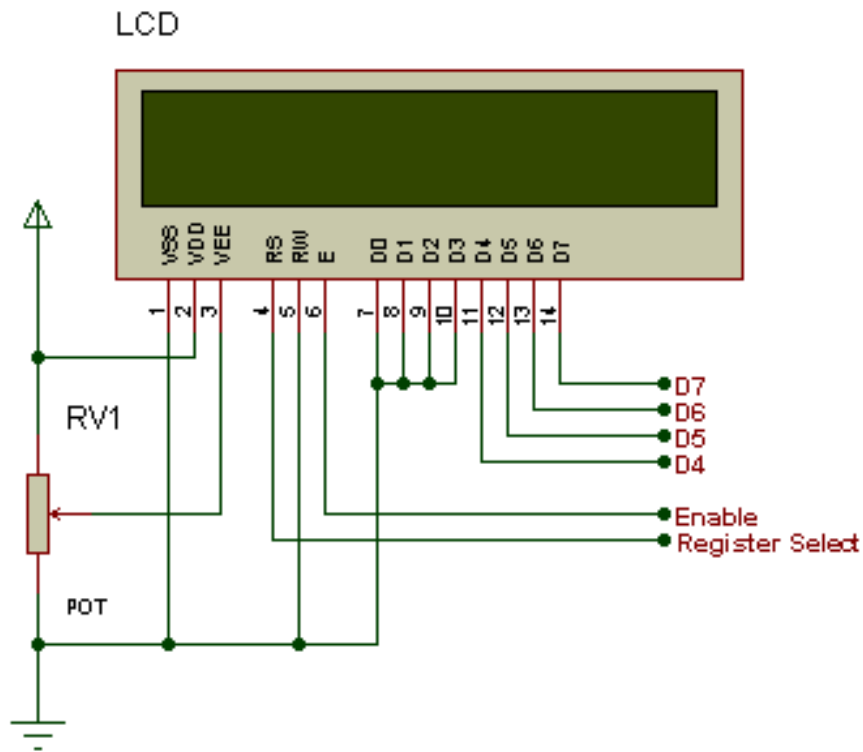


FIG 10:LCD INTERFACING DIAGRAM

Above is the connection diagram of LCD in 4-bit mode, where we only need 6 pins to interface an LCD. D4-D7 is the data pins connection and Enable and Register select are for LCD control pins. We are not using Read/Write (RW) Pin of the LCD, as we are only writing on the LCD so we have made it grounded permanently. If you want to use it, then you may connect it on your controller but that will only increase another pin and does not make any big difference. Potentiometer RV1 is used to control the LCD contrast. The unwanted data pins of LCD i.e. D0-D3 are connected to ground.

CHAPTER 7

SOURCE CODE

```
#include <LiquidCrystal.h>

#define led_pin 9

#define buzz_pin 8

LiquidCrystal lcd(2, 3, 4,5, 6, 7);

NewSoftSerial gsm(18,19);

char sms[50];

byte sms_length=0;

void setup()
{
  pinMode(led_pin,OUTPUT);
  pinMode(buzz_pin,OUTPUT);
  lcd.begin(16, 2);
  lcd.print("GSM Based System");
  gsm.begin(2400);
  byte s=search_modem(); // search for modem to be connected. blocking
  s=configure_modem(); // configure the modem //
  s=del_sms(); // keep the location 1 clear.//
  lcd.clear();
  lcd.print("NOTHING 2 DISPLAY");
  indicate(100);
```

```

}
void loop( )
{
  if(gsm.available(>2)// if data arrived from gsm
  {
    indicate(1000);// indication
    int k=check_sms();// check for sms
    if(k==1) //if sms arrived
      k=read_sms();//read the sms
    if(k==1)
      k=del_sms(); // clear the location for next sms to arrive
    lcd.clear();
    lcd.print("**Display Board**");
    lcd.setCursor(0,1);
    lcd.print(sms);
    for(int i=0;i<50;i++)
    {
      sms[i]=' '; //clear the sms
    }
  }
}
byte search_modem() // will keep on searching for modem unless it is found
{
  lcd.clear();
  lcd.print(" Testing modem....");

```

```

lcd.setCursor(0,1);
char ch[50];
int n=0;
gsm.println("AT"); // activate the modem
while(1)
{
  if(gsm.available())
  {
    ch[n]=gsm.read();
    if((ch[n-1]=='O')&&(ch[n]=='K')) //ok if modem is there
    {
      lcd.print("Testing Done...");
      return 1;
    }
    n++;
  }
  else{
    lcd.clear();
    lcd.print("Connect Modem");
    lcd.setCursor(0,1);
    gsm.println("AT"); // activate the modem
    delay(2000);
  }
}
} // search ends

```



```

byte configure_modem()
{
  char ch[50];
  int n=0;
  lcd.clear();
  lcd.print("Configuring modem....");
  gsm.println("ATE0"); // turn off echo
  delay(500);
  gsm.println("ATE0"); // turn off echo
  delay(500);
  gsm.println("ATE0"); // turn off echo
  delay(500);
  gsm.println("AT+CMGF=1"); //set the sms format as text
  delay(100);
  gsm.println("AT+CNMI=2,2,0,1,0"); // indicates the arrival of sms
  delay(1000);
  gsm.flush();
  gsm.println("AT");
  while(1)
  {
    if(gsm.available())
    {
      ch[n]=gsm.read();

      if((ch[n-1]=='0')&&(ch[n]=='K')) //ok if modem is there

```

```

    {
        lcd.clear();
        lcd.print("Cofigured.....");
        return 1;
    }
    n++;
}
else
{
    n=0;
    lcd.clear();
    lcd.print("Confi Failed..");
    lcd.setCursor(0,1);
    lcd.print("Please Reset");
}
}
}

```

```
int check_sms()
```

```

{
    char ch[20];
    int n=0;
    byte flag=0; // indicates the arrival of sms

    if(gsm.available())

```

```

{
    delay(1000);// wait for all serial data to arrive.
    while(gsm.available() // read all serial data.
    {
        ch[n]=gsm.read();
        if((ch[n-3]=='C')&&(ch[n-2]=='N')&&(ch[n-1]=='T')&&(ch[n]=='T'))
        {
            flag=1;
            lcd.clear();
            lcd.print("SMS ARRIVED");
        }
        n++;
    }
    if(flag==1) // IF SMS ARRIVES
        return 1;
    else
        return 0;

}
}

byte read_sms()
{
    char ch[100];
    int n=0;
    int m=0;

```

```

byte temp=0;
lcd.clear();
lcd.print("Reading SMS....");
lcd.setCursor(0,1);
gsm.println("AT+CMGR=1"); // command to read the sms at loction 1
while(1)
{
  if(gsm.available())
  {
    ch[n]=gsm.read();
    if((ch[n-1]=='O')&&(ch[n]=='K')) // successful reading of SMS
    {
      for(int i=0;i<=n;i++) // extract sms part
      {
        if(ch[i]=='*') // terminating character stop reading
        {
          temp=0;
          sms_length=m-1;
        }
        if(temp==1) // store sms part in sms variable
        {
          sms[m]=ch[i];
          m++;
        }
        if(ch[i]=='#') //starting character start reading

```

```

        temp=1;
    }// end of for loop
    lcd.print("Reading successful..");
    return 1;
}
n++;
}
}
}
byte del_sms()
{ char ch[10];
  int n=0;
  lcd.clear();
  lcd.print("Removing SMS..");
  lcd.setCursor(0,1);
  gsm.println("AT+CMGD=1"); // command to delete the sms at loction 1
  while(1)
  {
    if(gsm.available())
    {
      delay(1000);
      while(gsm.available())
      {
        ch[n]=gsm.read();
        if((ch[n-1]=='O')&&(ch[n]=='K'))

```

```

    {
        lcd.print("SMS Deleted..");
        return 1;

        if(ch[i]=='*') // terminating character stop reading
        {
            temp=0;
            sms_length=m-1;
        }
        if(temp==1) // store sms part in sms variable
        {
            sms[m]=ch[i];
            m++;
        }
        if(ch[i]=='#') //starting character start reading
            temp=1;
    } // end of for loop
    lcd.print("Reading successful..");
    return 1;
}
n++;
}
}
}
byte del_sms()

```

```

{
  char ch[10];
  int n=0;
  lcd.clear();
  lcd.print("Removing SMS..");
  lcd.setCursor(0,1);
  gsm.println("AT+CMGD=1"); // command to delete the sms at loction 1
  while(1)
  {
if(gsm.available())
  {
    delay(1000);
    while(gsm.available())
    {
      ch[n]=gsm.read();
if((ch[n-1]=='O')&&(ch[n]=='K'))
      {
        lcd.print("SMS Deleted..");
        return 1;
      }
      n++;
    }
    lcd.print("Error...");
    return 0;
  }
}

```

```
    }  
  }  
  void indicate(int d)  
  {  
    digitalWrite(buzz_pin,HIGH);  
    digitalWrite(led_pin,HIGH);  
    delay(d);  
    digitalWrite(buzz_pin,LOW);  
    digitalWrite(led_pin,LOW);  
  }
```


CONCLUSION

This project is based on GSM Modem infrastructure. So, all the operations involve the GSM system also. As we send any SMS, it goes through the GSM system. Any sent SMS can be received if we use a SIM card and GSM module. To operate any GSM modem, we have to use the AT commands to operate them. The format for sending message is for example, if any SMS arrives the GSM modem sends the serial data in ASCII format. We can read these data if we connect the modem with the serial port of the microcontroller at the baud rate of 9600. As the microcontroller comes to know that a SMS has been arrived, it can sent a proper AT command to read the SMS. The reading of SMS returns the mobile no of sender, the time and much more information. We have to select the SMS part of the message. The starting string of the SMS is used as the password. As the password is matched, then the SMS arrival is assumed to be valid by the microcontroller otherwise, it ignores the SMS. On the basis of proper initiator and terminator the controller (ATMEGA168) decodes the SMS (#&* in our case).Now the retrieved SMS is displayed on the LCD module. Proper LCD display is ensured through programming and the LCD interface design.

REFERENCES

1. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July GSM based e-Notice Board: Wireless Communication by Pawan Kumar, Vikas Bhrdwaj, Kiran Pal, Narayan Singh Rathor, Amit Mishra.
2. Remote Household Appliance Control System Using GSM
R. Chutia, D. Sonowal and S. Sharma
Department of Electronics and Communication Engineering
Tezpur Central University
Tezpur, Assam, India