HEALTH CARE MONITORING USING SMS BASED TELEMEDICINE SYSTEM

A PROJECT REPORT

Submitted by

R.G.BALAJI	71206106014
D.NITHYANANTH	71206106034
J.STANY JOSEPH	71206106050
S.SARAVANAN	71206106305

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

ANNA UNIVERSITY :: CHENNAI 600 025

APRIL 2010

ANNA UNIVERSITY : CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"HEALTH CARE MONITORING USING SMS BASED TELEMEDICINE SYSTEM"** is the bonafide work of **"BALAJI.R.G, NITHYANANTH.D, STANY JOSEPH.J, SARAVANAN.S"** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. RAJESWARI MARIAPPAN	Ms. M.SHANTHI
HEAD OF THE DEPARTMENT	**SUPERVISOR**
	ASST.PROFESSOR
Electronics and Communication Engineering Kumaraguru College of Technology Coimbatore – 641006.	Electronics and Communication Engineering Kumaraguru College of Technology Coimbatore – 641006.

The candidates with university register numbers **71206106014, 71206106034, 71206106050, 71206106305** were examined by us in the project viva-voce examination held on ..16/04/2010.

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

We acknowledge our thanks to **Dr. S.RAMACHANDRAN, PRINCIPAL**, Kumaraguru College of Technology, Coimbatore, for providing an opportunity to take up this project.

We take this opportunity to thank, **Dr. RAJESWARI MARIAPPAN, HEAD OF THE DEPARTMENT**, Electronics and Communication Engineering, Kumaraguru College of Technology, Coimbatore, for her extended support in completion of the project.

We wish to express our sincere thanks and deep sense of gratitude to **Ms. A.VASUKI, PROJECT COORDINATOR**, Department of Electronics and Communication Engineering, Kumaraguru College of Technology, Coimbatore, for her wise counsel with timely and inspiring directions during this project work.

We take this opportunity to express our profound thanks to our guide **Ms. M.SHANTHI, ASST.PROFESSOR**, Department of Electronics and Communication Engineering, Kumaraguru College of Technology, Coimbatore, for her valuable guidance and support for the successful completion of the project work.

Finally, we express our thanks to our parents and friends for their encouragement in completion of this project work successfully.

ABSTRACT

A mobile monitoring system utilizing short message service with low cost hardware equipment has been developed to enable transmission of the temperature and pulse information of a patient. Communication between the GSM modem in the patient side and multiple mobile numbers (for whom the patient information is to be transmitted) which are stored in the database is achieved through programming this system using attention commands. The proposed setup can be operated for monitoring the patient from anywhere covered by the cellular service by sending short message service (SMS) to remote mobile phones. Proposed implementation includes temperature sensor and pulse sensor to get temperature and pulse information of the patient. Low cost temperature sensor LM35 and pulse sensor 0628i are interfaced to 89S51 Microcontroller using analog to digital converter ADC0808. The AT commands are standard control tools to establish communication between GSM modem and mobile phone .The patient details are checked for abnormality and the information is displayed in a LCD screen. If there is any abnormality at once the SMS will be sent through GSM modem to the specific contacts maintained in the database.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 BRIEF OVERVIEW

According to the World Health Organization (WHO) estimate, cardiovascular disease kills almost seventeen million people around the globe each year, with around twenty million people at a risk of sudden heart failure. Some of these lives can often be saved if prompt emergency care and cardiac surgery are provided within the so-called golden hour. Therefore, patients who are at risk require that their cardiac health to be monitored frequently whether they are indoors or outdoors so that emergency treatment can be given if problems arise.

Telemedicine is widely considered to be part of the inevitable future of the modern practice of medicine. It is gaining more and more momentum as a new approach for patients' surveillance outside of hospitals (at home) to encourage public safety, facilitate early diagnosis, treatment, and for increased convenience. It is defined as the "use of advanced telecommunication technologies to exchange health information and provide health care services across geographic, time, social, and cultural barriers," telemedicine is currently being used by doctors, hospitals, and other health care providers around the world, Conventional telemedicine systems using Public Switched Telephone Network (PSTN) land lines are already available to enable a doctor to monitor a patient remotely for home care or emergency applications.

The mobile phone has been recognized as a possible tool for telemedicine since it became commercially available. Moreover, newer cellular access technologies, such as GPRS, EDGE, 3G, and WiMax, provide for much higher data transmission speeds (rates) than the basic 2G GSM cellular system, which offers future telemedicine solutions endless choices for highend designs. However, these relatively new wireless technologies are deployed mostly in or around crowded high income metropolitan areas. Therefore, the majority (80.5%) of the 3.7 billion cellular phone users in the world are still 2G GSM users.

Short Messaging Service (SMS) is an integral part of the original GSM system and its evolution. SMS is being used for many commercial and government applications, and this use is continuously increasing. This project describes an SMS based telemedicine system to transfer a patient's temperature and pulse informations, which can also be expanded to include other vital signs. Motivation to use SMS is that not only does SMS offer an alternative means of transmission in a cellular communication system like GSM, but at times it can be the only available or most efficient option.

1.2 BLOCK DIAGRAM

The outputs of Pulse sensor & Temperature sensor are connected to any two input pins of Analog to Digital Converter. The Digitalized output of ADC is connected to the microcontroller AT89S51. The LCD display is connected to one port of the microcontroller. The serial communication pins of the microcontroller are connected to the GSM modem. The Block Diagram is shown in figure 1.1

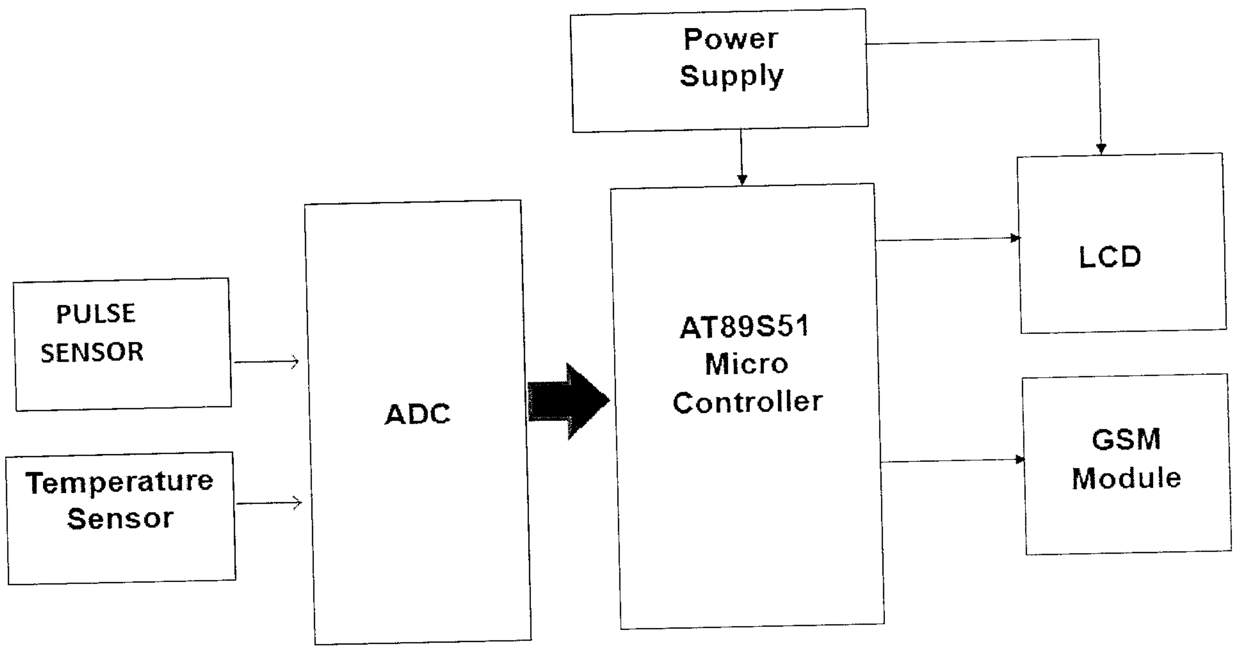


FIGURE 1.1 : BLOCK DIAGRAM

1.3 FUNCTIONAL DESCRIPTION

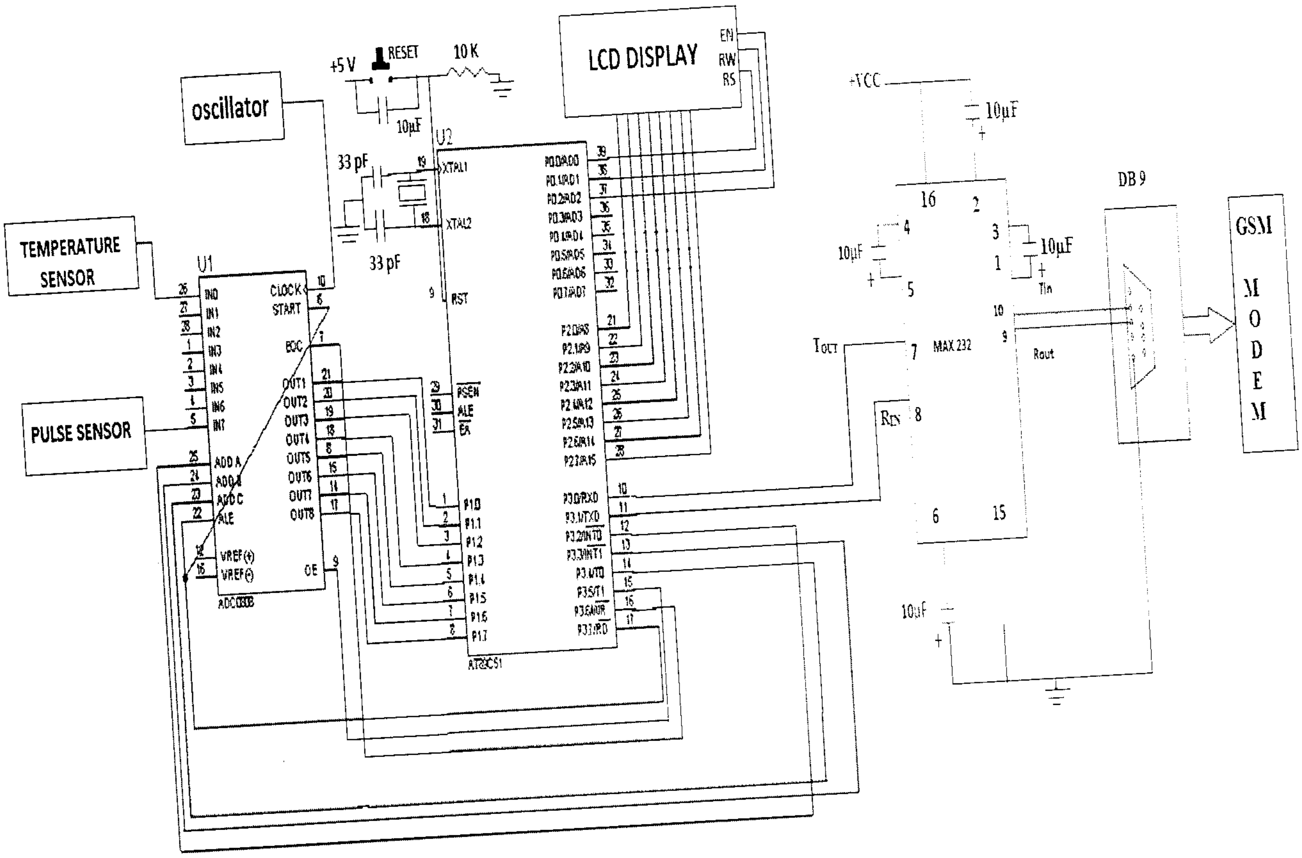


FIGURE 1.2 : CIRCUIT DIAGRAM

Figure 1.2 gives the Circuit Diagram. The temperature sensor is connected to the channel 0 of the ADC & the pulse sensor is connected to the channel 7 of the ADC. Any one of these sensors can be enabled by giving proper input to channel selection pins of ADC. The Start conversion, End of conversion & Output Enable functions can be controlled by Read, Write & Timer 1 pins of microcontroller.

The digitalized output of the ADC is connected to port1 of the microcontroller. The microcontroller compares the digitalized temperature sensor output with the threshold value. If it exceeds the threshold value, the temperature will be considered as abnormal.

The microcontroller will perform six successive analog to digital conversion for the pulse sensor output. If any one of this digitalized value exceeds the threshold value, then the pulse rate will be considered as normal. Otherwise pulse rate will be considered as abnormal.

The temperature value & pulse information will be displayed on the LCD display. If both or any one of this two parameter is abnormal then the intimation SMS will be sent to multiple mobile numbers which are stored in the database.

These short messages are sent by GSM modem which is connected to the microcontroller through Max 232 & controlled by means of AT commands.Figure 1.3 shows the flow chart of this project.

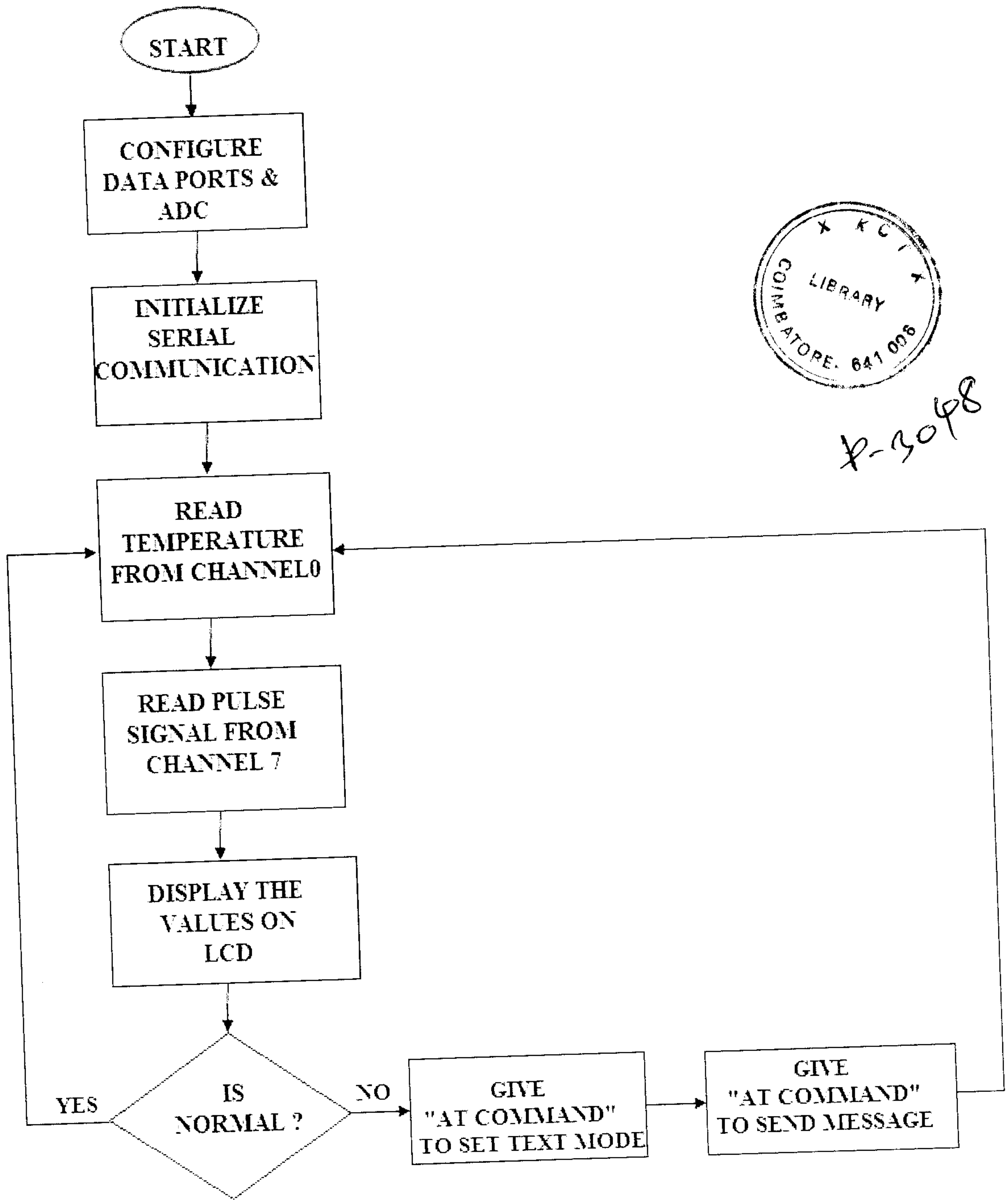


FIGURE 1.3 : FLOW CHART

CHAPTER 2

MICROCONTROLLER

2.1 INTRODUCTION TO AT89S51

Microcontroller is a general purpose device, which integrates a number of components of a microprocessor system on to single chip. It has inbuilt CPU, memory and peripherals to make it as a mini computer. A microcontroller combines on to the same microchip:

- The CPU core
- Memory (both ROM and RAM)
- Some parallel digital i/o

MICROCONTROLLERS ARE :

- Smaller in size
- Consumes less power
- inexpensive

Microcontroller is a standalone unit, which can perform functions on its own without any requirement for additional hardware like I/O ports and external memory. The heart of the microcontroller is the CPU core. In the past, this has traditionally been based on a 8-bit Microprocessor unit. For example Motorola uses a basic 6800 Microprocessor core in their 6805/6808 microcontroller devices.

In the recent years, Microcontroller have been developed around specifically designed CPU cores, for example the microchip PIC range of microcontroller. The AT89S51 is a low-power, high performance CMOS 8-bit microcomputer with 4K bytes of Flash programming .The device is manufactured using ATMEL's high-density nonvolatile memory technology and is compatible to standard 80S51 instruction set and pin out.

The on-chip Flash allow the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining versatile 8-bit CPU with Flash on a monolithic chip, the ATMEL AT89S51 is a powerful microcomputer which provides a high-flexibly and cost-effective solution to many embedded control applications.

2.2 PIN DIAGRAM :

All the four ports of the Microcontroller AT89S51 are 8-bit bi-directional I/O ports. In port 0 each pin can sink eight TTL inputs & other three ports can sink/source four TTL inputs .Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pull-ups are required during program verification. Port 3 also serves the functions of various special features of the AT89S51. It also receives some control signals for Flash programming and verification. XTAL1 and XTAL2 are the input and output respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator. Either quartz crystal or ceramic resonator may be used. Pin diagram of Microcontroller AT 89S51 is given in figure 2.1

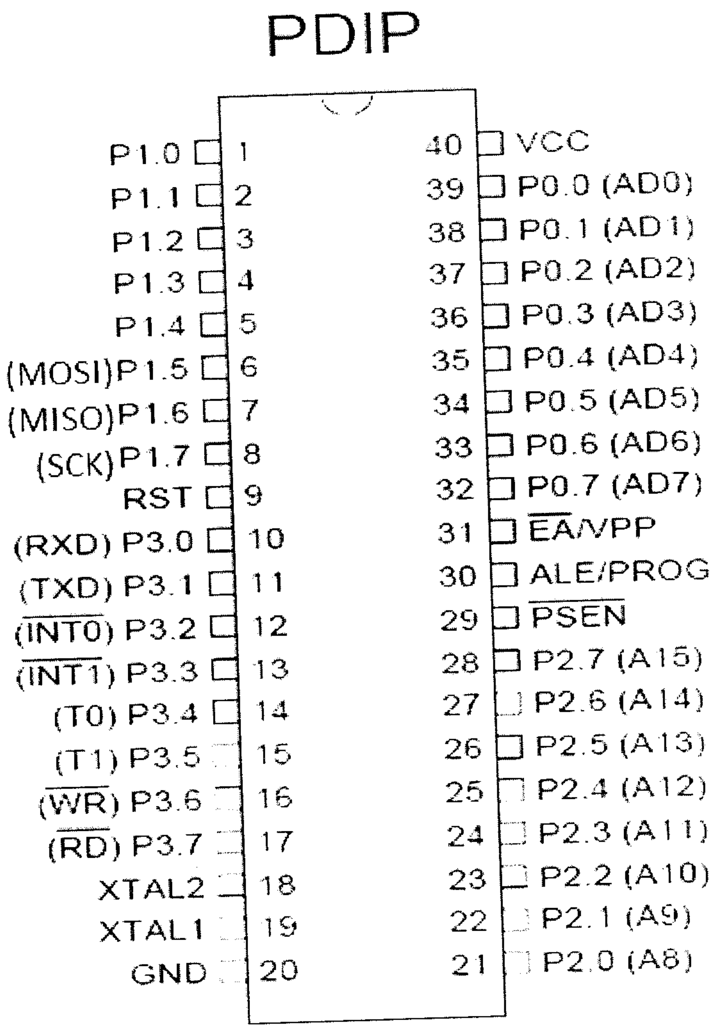


FIGURE 2.1 : PIN DIAGRAM OF AT89S51

2.3 CORE FEATURES

- Compatible with MCS-51™ Products
- 4K Bytes of In-System Programmable Flash Memory Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- 128 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-bit Timer/Counters
- Six Interrupt Sources
- Full Duplex UART Serial Channel
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Fast Programming Time
- Flexible ISP Programming (Byte and Page Mode)

CHAPTER 3

SENSORS & ANALOG TO DIGITAL CONVERTER

3.1 TEMPERATURE SENSOR

Most commonly, a temperature sensor is used to convert temperature value to an electrical value. Temperature sensors are the key to read temperatures correctly and to control temperature in industries and medical applications.A large distinction can be made between temperature sensor types. Sensors differ a lot in properties such as contact-way, temperature range, calibrating method and sensing element. The temperature sensors contain a sensing element enclosed in housings of plastic or metal. With the help of conditioning circuits, the sensor will reflect the change of body temperature.

LM35 temperature sensor series is precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the centigrade temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in degrees Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient centigrade scaling. The LM35 does not require any external calibration or trimming. The LM35 is rated to operate over a -55° to +150°C temperature range. The pin diagram of LM 35 temperature sensor is shown in figure 3.1

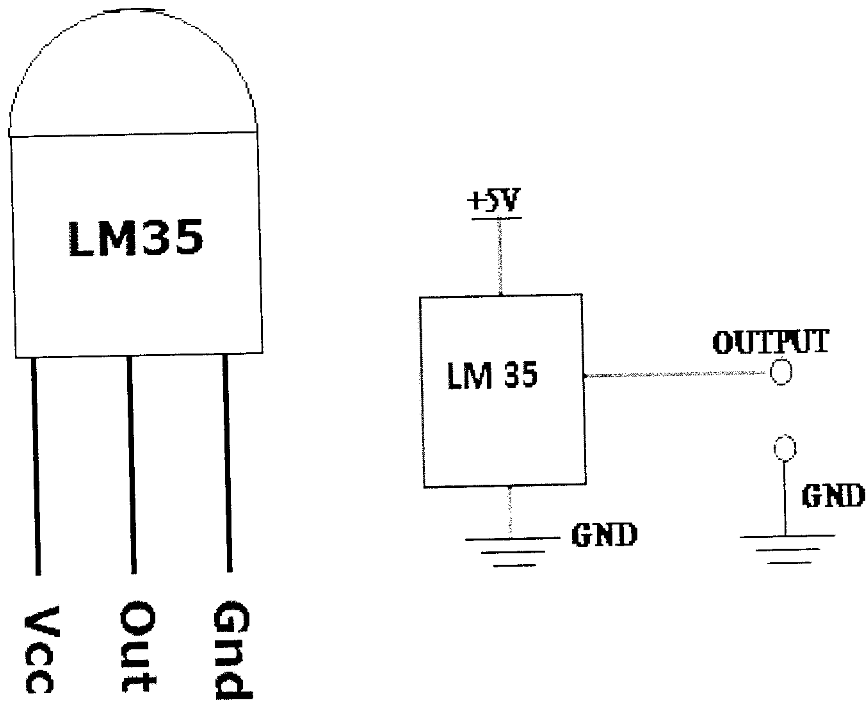


FIGURE 3.1 : TEMPERATURE SENSOR

It is easy to include the LM35 series in a temperature measuring application. The output voltage of LM35 is linearly proportional to the centigrade temperature, it has a Linear +10.0 mV/°C scale factor. n*10.0 mV output voltage will be obtained for the environment temperature being n°C.

The temperature sensor functional module consists of two parts: the functional module box shown in figure 3.2 and the probe head shown in figure 3.3. The LM35 temperature sensor is mounted on the probe head.

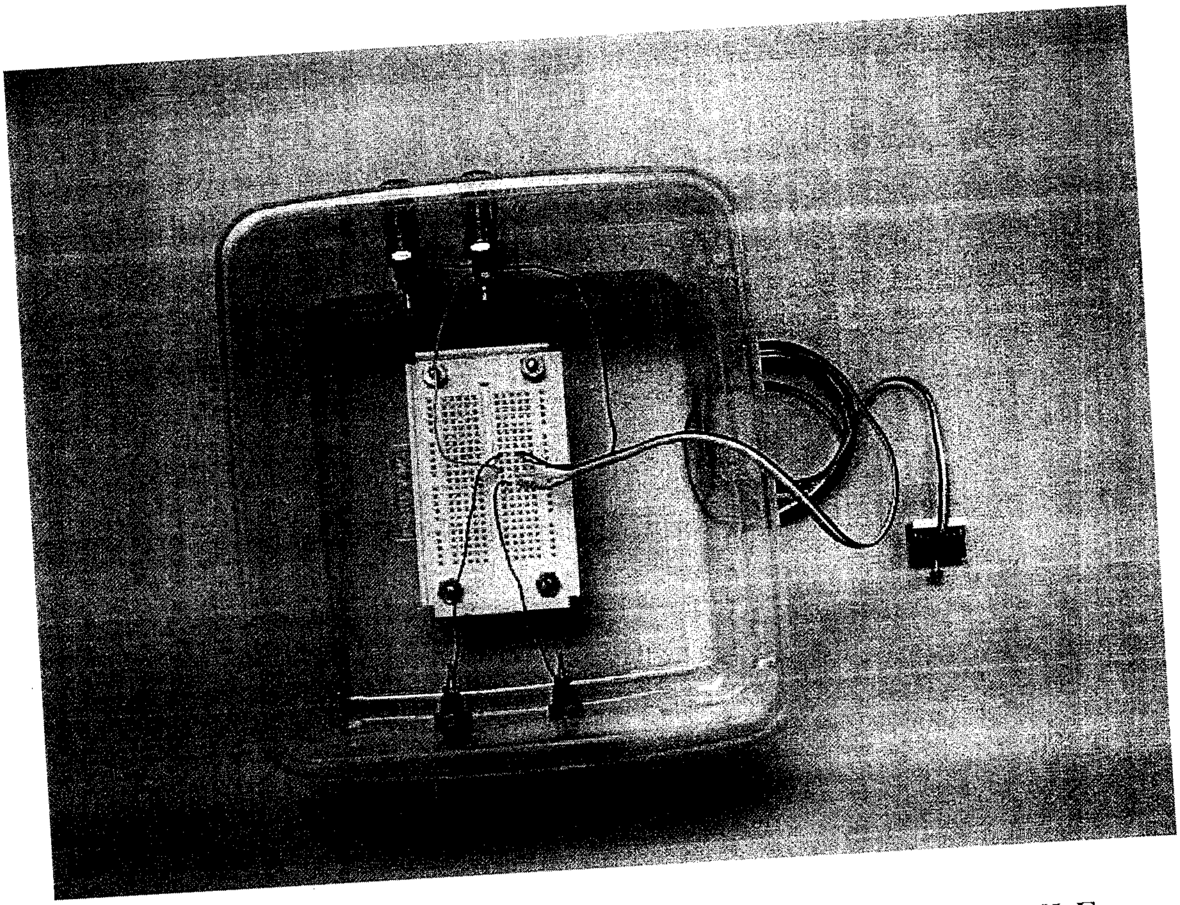


FIGURE 3.2 : TEMPERATURE SENSOR MODULE.

Wiring of the LM35 temperature functional module consists of wiring of functional module box and wiring of the sensor probe. The sensor is mounted on the probe head and the probe is connected to the module box. For LM35 in metal can package (TO-46), the small tab on the sensor indicates the position of "+Vs" leading pin.

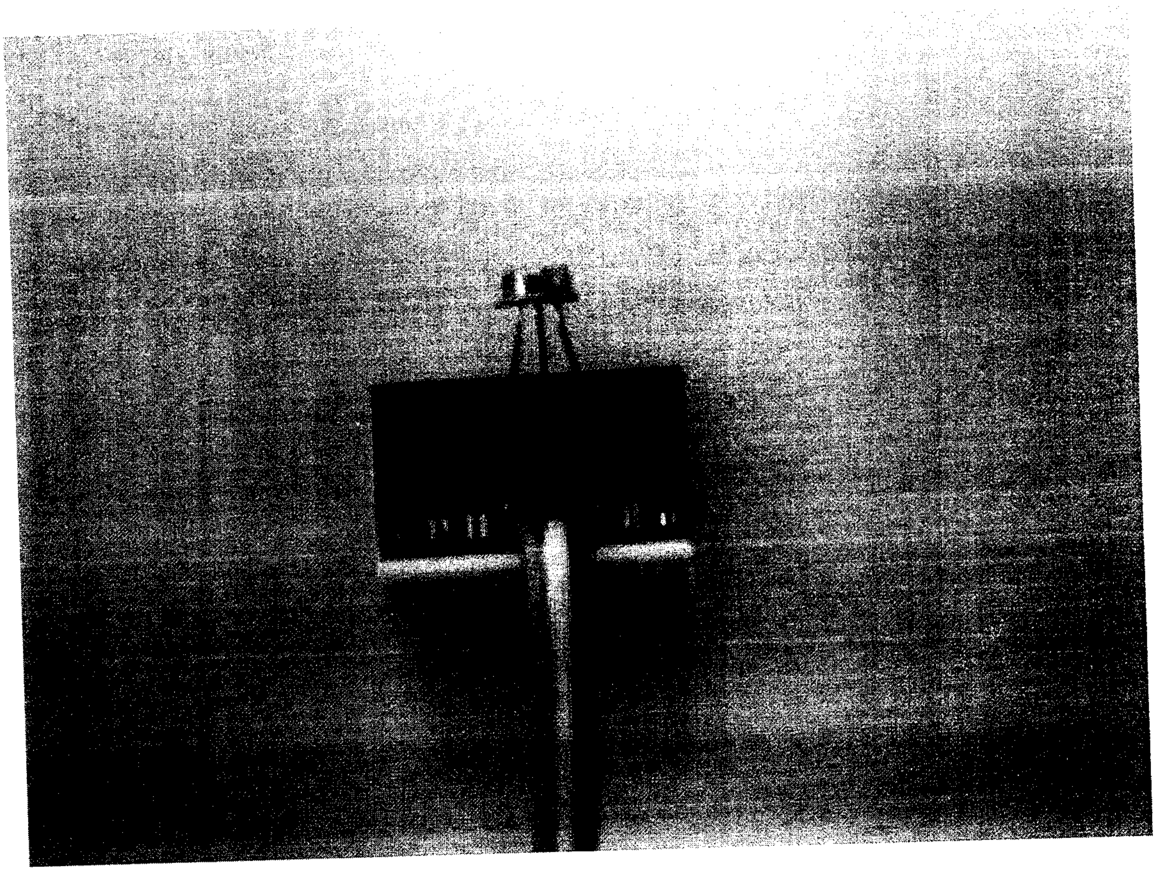


FIGURE 3.3 : PROBE HEAD.

3.2 PULSE SENSOR (0628i)

Every (normal) heartbeat is composed of a P wave, a QRS complex and a T wave. The P wave is caused by depolarization of the atrial tissue prior to contraction. The QRS waves are generated by currents when the ventricular tissue depolarizes, prior to contraction. The T wave is caused by currents generated as the ventricle recovers from depolarization. The P-R interval is between 0.12 and 0.20 seconds for most persons. The length of this interval is independent of the heartbeat rate. Specifically, the height of the P wave and the length of the S wave may differ from the pattern shown in medical books. This might be due to the method of measuring (the electrodes are placed on the wrists instead of on the breast).

The pulse sensor measures voltages that are produced by the heart. These small voltages can be measured at the skin of the wrists and elbow through electrodes. The voltages are amplified by the sensor and filtered and transferred through an optical coupler to a measurement interface.

3.3 ANALOG TO DIGITAL CONVERTER (ADC 0808)

The ADC is a device which accepts the analog input voltage V_a and produces an output binary word d1,d2,....dn. An ADC usually has two additional control lines: the START input to tell the ADC when to start conversion and the EOC output to announce when the conversion is complete.

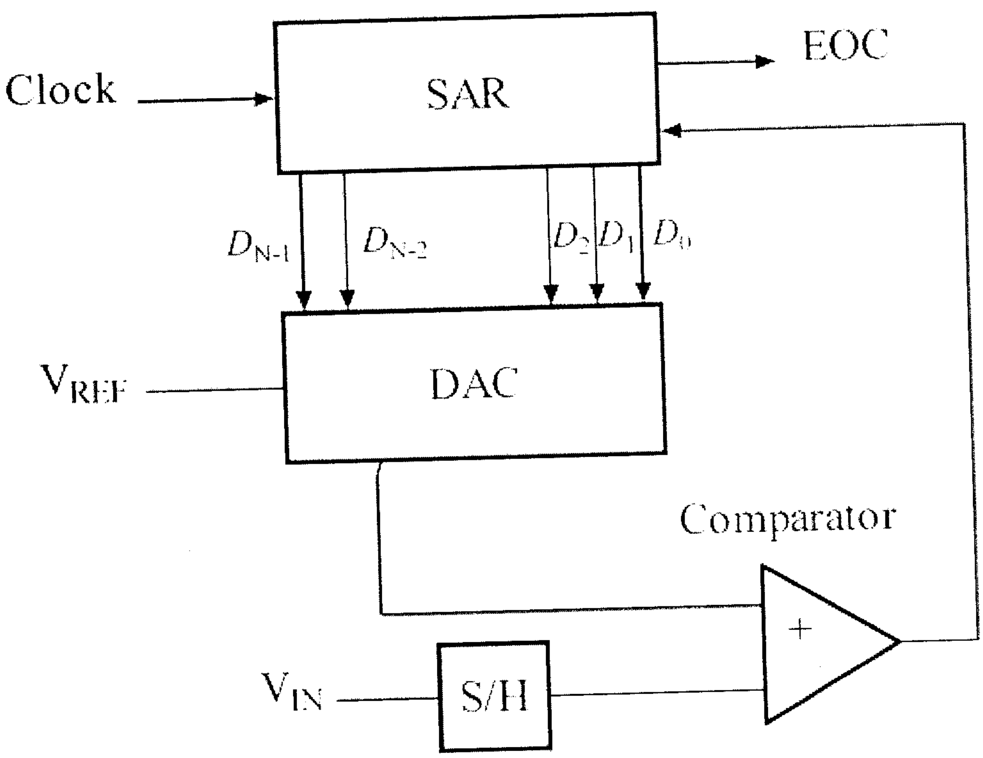


FIGURE 3.4 : SUCCESSIVE APPROXIMATION ADC

The successive approximation uses a very efficient code search strategy to complete n-bit conversion in just n clock periods. The block diagram of successive approximation ADC is shown in figure 3.4. An 8-bit converter would require eight clock pulses to obtain a digital output. The ADC 0808 uses a successive approximation register to find the required value for each bit by trial and error. The circuit operates as follows. With the arrival of the START command, the SAR sets the MSB d1 to 1and all other bits to zero so that the trial code is 10000000.

The output V_d of the DAC is now compared with analog input V_a. If V_a is greater than ADC output V_d then 10000000 is less than the correct digital representation. The MSB is left at 1 and the next lower significant bit is made 1 and further tested. However, if V_a is less than the DAC output, then 10000000 is less than the correct digital representation. So reset MSB to zero and go on to the next lower significant bit. This procedure is repeated for all subsequent bits, one at a time, until all bit positions have been tested.

The ADC0808 data acquisition component is a monolithic CMOS device with an 8-bit analog-to-digital converter, 8-channel multiplexer and Microprocessor compatible control logic. The 8-bit A/D converter uses successive approximation as the conversion technique. The converter features a high impedance chopper stabilized comparator, a 256R voltage divider with analog switch tree and a successive approximation register. The 8-channel multiplexer can directly access any of 8-single-ended analog signals. The pin diagram of ADC 0808 is shown in figure 3.5.

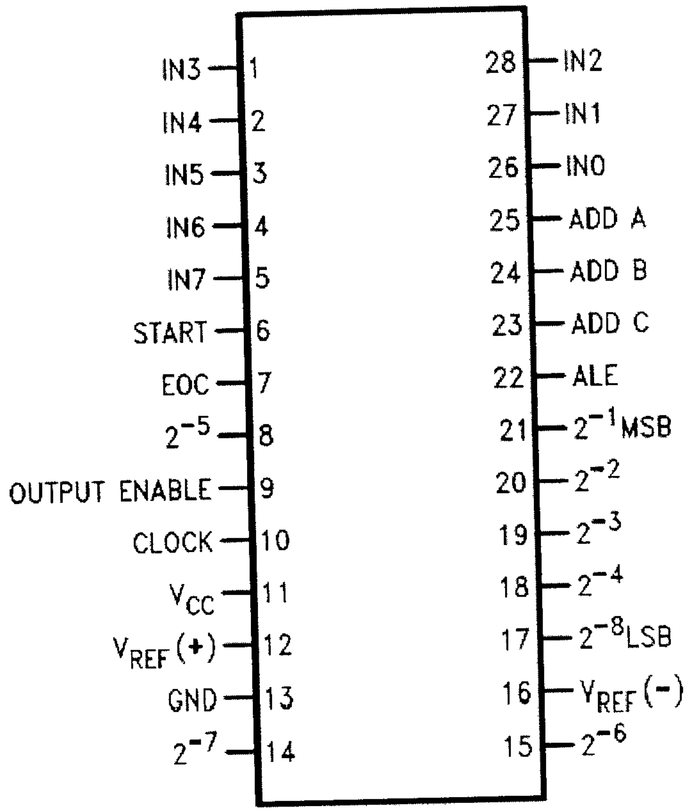


FIGURE 3.5 : PIN DIAGRAM OF ADC 0808

CHAPTER 4

GSM MODEM & ITS INTERFACING

4.1 INTRODUCTION TO GSM MODEM

Global system for mobile communication (GSM) is a globally accepted standard for digital cellular communication. GSM is the name of a standardization group established in 1982 to create a common European mobile telephone standard that would formulate specifications for a pan-European mobile cellular radio system operating at 900 MHz.

Originally it had been intended that GSM would operate on frequencies in the 900 MHz cellular band. In September 1993, the British operator Mercury One-to-One launched a network. Termed DCS 1800 it operated at frequencies in a new 1800 MHz band. By adopting new frequencies new operators and further competition was introduced into the market apart from allowing additional spectrum to be used and further increasing the overall capacity. This trend was followed in many countries, and soon the term DCS 1800 was dropped in favour of calling it GSM as it was purely the same cellular technology but operating on a different frequency band. In view of the higher frequency used the distances the signals travelled was slightly shorter but this was compensated for by additional base stations.

In the USA as well a portion of spectrum at 1900 MHz was allocated for cellular usage in 1994. The licensing body, the FCC, did not legislate which technology should be used, and accordingly this enabled GSM to gain a foothold in the US market. This system was known as PCS 1900 (Personal Communication System).

A GSM modem is a wireless modem that works with a GSM wireless network. A wireless modem behaves like a dial-up modem. The main difference between them is that a dial-up modem sends and receives data through a fixed telephone line while a wireless modem sends and receives data through radio waves.

A GSM modem can be an external device or a PC Card / PCMCIA Card. Typically, an external GSM modem is connected to a computer through a serial cable or a USB cable. A GSM modem in the form of a PC Card / PCMCIA Card is designed for use with a laptop computer. It should be inserted into one of the PC Card / PCMCIA Card slots of a laptop computer. Like a GSM mobile phone, a GSM modem requires a SIM card from a wireless carrier in order to operate.

In addition to the standard AT commands, GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. With the extended AT commands, you can do things like:

- Reading, writing and deleting SMS messages.
- Sending SMS messages.
- Monitoring the signal strength.
- Monitoring the charging status and charge level of the battery.
- Reading, writing and searching phone book entries.

4.2 GSM NETWORK ELEMENTS

The structure of GSM Network is shown in figure 4.1

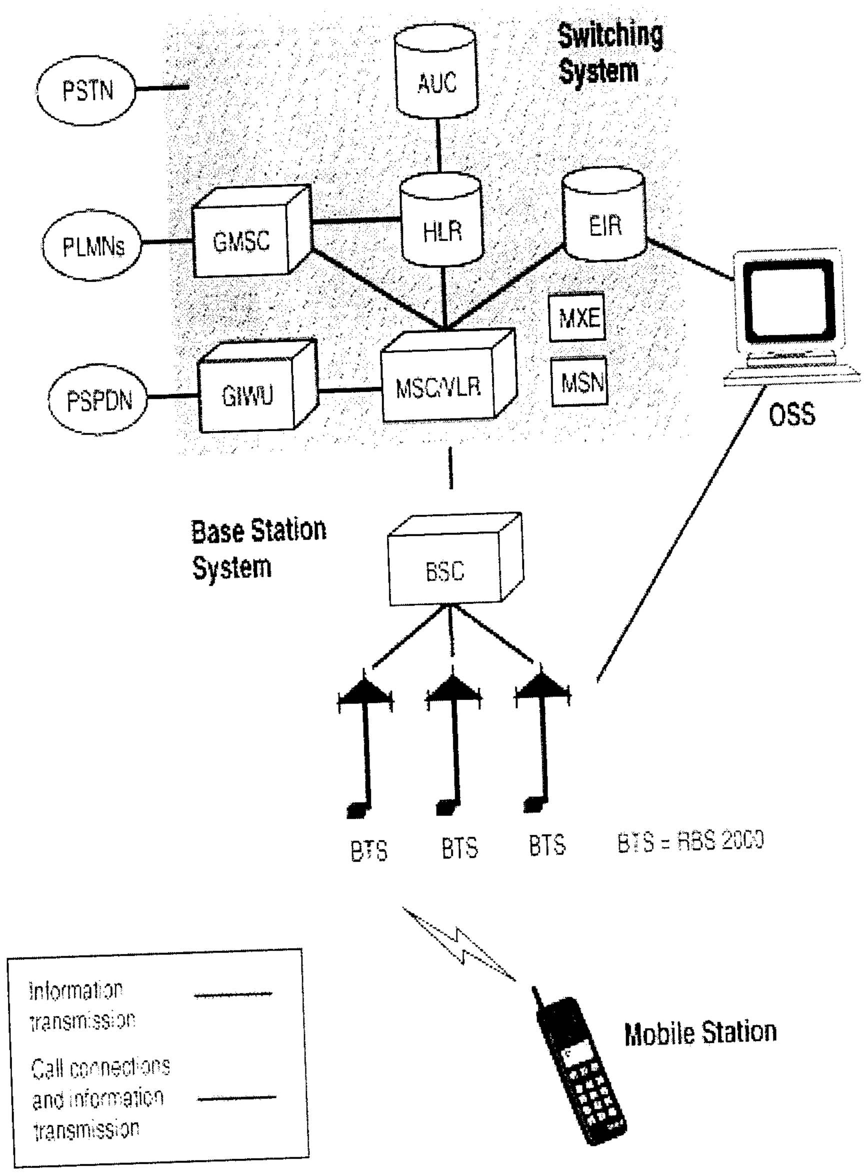


FIGURE 4.1 : GSM NETWORK ELEMENTS

THE SWITCHING SYSTEM

The switching system (SS) is responsible for performing call processing and subscriber-related functions. The switching system includes the following functional units.

- Home Location Register (HLR)—The HLR is a database used for storage and management of subscriptions. The HLR is considered the most important database, as it stores permanent data about subscribers, including a subscriber's service profile, location information, and activity status. When an individual buys a subscription from one of the PCS operators, he or she is registered in the HLR of that operator.
- Mobile Services Switching Center (MSC)—The MSC performs the telephony switching functions of the system. It controls calls to and from other telephone and data systems. It also performs such functions as toll ticketing, network interfacing, common channel signaling, and others.
- Visitor Location Register (VLR)—The VLR is a database that contains temporary information about subscribers that is needed by the MSC in order to service visiting subscribers. The VLR is always integrated with the MSC. When a mobile station roams into a new MSC area, the VLR connected to that MSC will request data about the mobile station from the HLR. Later, if the mobile station makes a call, the VLR will have the information needed for call setup without having to interrogate the HLR each time.
- Authentication Center (AUC)—A unit called the AUC provides authentication and encryption parameters that verify the user's identity and ensure the confidentiality of each call. The AUC protects network operators from different types of fraud found in today's cellular world.

- Equipment Identity Register (EIR)—The EIR is a database that contains information about the identity of mobile equipment that prevents calls from stolen, unauthorized, or defective mobile stations. The AUC and EIR are implemented as stand-alone nodes or as a combined AUC/EIR node.

THE BASE STATION SYSTEM (BSS)

All radio-related functions are performed in the BSS, which consists of base station controllers (BSCs) and the base transceiver stations (BTSs).

- **BSC**—The BSC provides all the control functions and physical links between the MSC and BTS. It is a high-capacity switch that provides functions such as handover, cell configuration data, and control of radio frequency (RF) power levels in base transceiver stations. A number of BSCs are served by an MSC.
- **BTS**—The BTS handles the radio interface to the mobile station. The BTS is the radio equipment (transceivers and antennas) needed to service each cell in the network. A group of BTSs are controlled by a BSC.

THE OPERATION AND SUPPORT SYSTEM

The operations and maintenance center (OMC) is connected to all equipment in the switching system and to the BSC. The implementation of OMC is called the operation and support system (OSS). The OSS is the functional entity from which the network operator monitors and controls the system. The purpose of OSS is to offer the customer cost-effective support for centralized, regional, and local operational and maintenance activities that are required for a GSM network. An important function of OSS is to provide a network overview and support the maintenance activities of different operation and maintenance organizations.

4.3 AT COMMANDS

The AT commands are standard control tools based on GSM (07.07) to establish communication with the mobile GSM phone or modem. The commands set consist of strings, which will enable the exchange of serial data, according to certain syntax rules, between the mobile and the microcontroller at the client unit, and the laptop or PC at the server. As an example: "AT+CMGS=140", is a command for sending a SMS message, where "AT" is a prefix used for all commands, "CMGS" is a description to the kind of task to be performed, and "140" is the message length. Similarly, when a message is received by the laptop according to the AT command "AT+CMGR=1", the program will be able to divide the message contents in order to extract the binary 8- bit samples to display the temperature.

AT COMMAND FEATURES

- Multi-Tech Line Settings

A serial link handler is set with the following default values (factory settings): auto baud, 8 bits data, 1 stop bit, no parity, RTS/CTS flow control. The +IPR, +IFC and +ICF commands can be used to change the settings.

- Command Line

Commands always start with AT (which means Attention) and finish with a <CR> character.

- Information Responses And Result Codes

Responses start and end with <CR><LF>, except for the ATV0 DCE response format) and the ATQ1 (result code suppression) commands.

- If command syntax is incorrect, an **ERROR** string is returned.
- If command syntax is correct but with some incorrect parameters, the **+CME ERROR:** <Err> or **+CMS ERROR:** <SmsErr> strings are returned with different error codes.
- If the command line has been performed successfully, an **OK** string is returned.
- In some cases, such as "AT+CPIN?" or (unsolicited) incoming events, the product does not return the **OK** string as a response.

EXAMPLES FOR AT COMMANDS

- **AT+CMGF=1**

  Set TEXT mode

- **AT+CMGS="+919444944941"<CR>**

  Please call me soon, Friend. <ctrl-Z>

  Send a message in text mode

- **AT+CMSS=5, 9003344559**

  Send the message 5 to a different destination numbers.

4.4 SERIAL COMMUNICATION

To transfer to a device located many meters away, the serial data transfer is used. In serial communication, the data is sending one bit at a time. ATMEL 89S51 has serial communication capability built into it, thereby making possible the data transfer using only a few wire.

For serial data communication to work, the byte of data must be converted to serial bits using a parallel-in-serial-out shift register, because the Microcontroller is sending parallel data. Then it can be transmitted over a single data line. It also means that at the receiving end there must be a serial-in parallel-out shift register.

4.4.1 SERIAL DATA TRANSFER

Serial data transfer uses two methods, Asynchronous and synchronous. The synchronous method transfers a block of data (character) at a time while the asynchronous transfers a single byte at a time. It is possible to write software to use either of these methods, but the programs can be tedious and long.

Asynchronous serial data communication is widely used for character oriented transmissions, while block-oriented data transfers use the synchronous methods. In the asynchronous method, each character is placed in between a start bit and a stop bit. The start bit is always a 0 (low) and the stop bit can be one (or) two bits. The start bit is always a 0 (low) and the stop bit is 1 (high).

The rate of data transfer in serial data communication is stated in bps (bits per second). Another widely used terminology for bps is baud rate. However, the baud rate is the modem terminology and is defined as the number of signal changes per second.

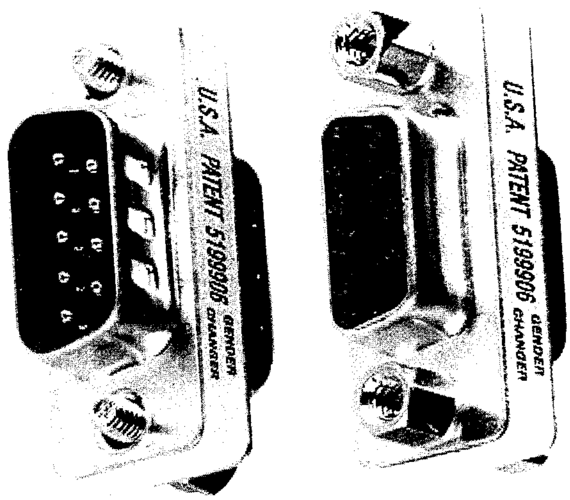
The data transfer rate of a given computer system depends on communication ports incorporated into the system. For e.g. the early IBM PC could transfer data at the rate of 100 to 9600bps. Now the recent computers serial communication the baud rate is as high as 512 Kbps. In asynchronous serial communication the baud rate is generally limited to 100,000 bps.

4.4.2 RS232 STANDARDS

To allow compatibility among data communication equipment made by various manufactures, the interface standards called RS232 are said by electronics industries association (EIA) in 1960. However, since the standards were set longer before the advent of the TTL logic family, its input and output voltage levels are not compatible. For this reason to connect any RS232 to a micro controller system we must use voltage converters such as MAX 232 to convert the TTL level to the RS232 voltage level and vice versa. MAX 232 IC chip are commonly referred to as line drivers.

FEATURES OF RS232

There is a standardized pin out for RS-232 on a DB 9 connector, as shown in figure 4.2.



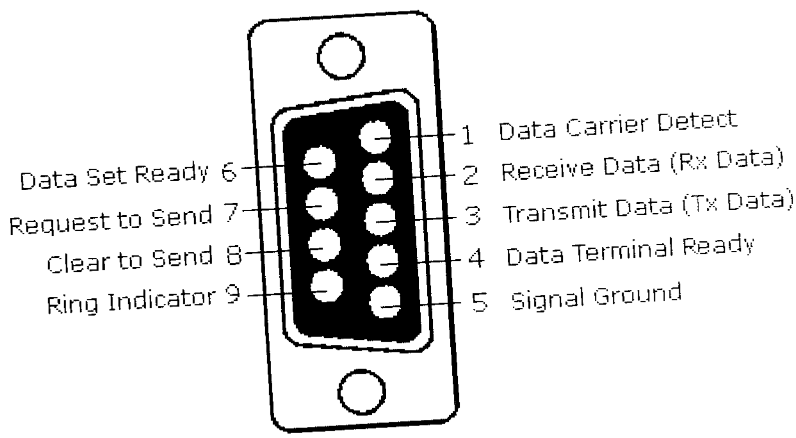


FIGURE 4.2 : DB 9 CONNECTOR

The essential feature of RS-232 is that the signals are carried as single voltages referred to a common ground on pin 5. Data is transmitted and received on pins 3 and 2 respectively. Data set ready (DSR) is an indication from the Dataset (i.e., the modem or DSU/CSU) that it is on. Similarly, DTR indicates to the Dataset that the DTE is on. Data Carrier Detect (DCD) indicates that carrier for the transmit data is on.

Pins 7 and 8 carry the RTS and CTS signals. In most situations, RTS and CTS are constantly on throughout the communication session. However where the DTE is connected to a multipoint line, RTS is used to turn carrier on the modem on and off. On a multipoint line, it is imperative that only one station is transmitting at a time. When a station wants to transmit, it raises RTS. The modem turns on carrier, typically waits a few milliseconds for carrier to stabilize, and raises CTS. The DTE transmits when it sees CTS up. When the station has finished its transmission, it drops RTS and the modem drops CTS and carrier together. This is explained further in our tutorial on the SDLC protocol, which uses multipoint lines extensively.

THE TRUTH TABLE FOR RS232

Signal > +3V=0

Signal < -3V=1

The output signal level usually swings between +12V and -12V. The "dead area" between +3V and -3V is designed to absorb line noise.

4.4.3 IC MAX 232

MAX 232 chip is commonly referred to as line drivers. Since the RS232 is not compatible with today's micro processors and micro controllers, we need a line driver (voltage converter) to convert the RS232's signals to TTL voltage level that will be acceptable to the 89S51's TxD and RxD pins. The MAX 232 convert from RS232 voltage levels to TTL voltage and viz. One advantage of the MAX 232 chip is that it uses a +5V power source, which is the same as the source voltage for the 89S51, with no need for the dual power supplies that is common in many holder system. The pin diagram of IC MAX 232 is shown in figure 4.3.

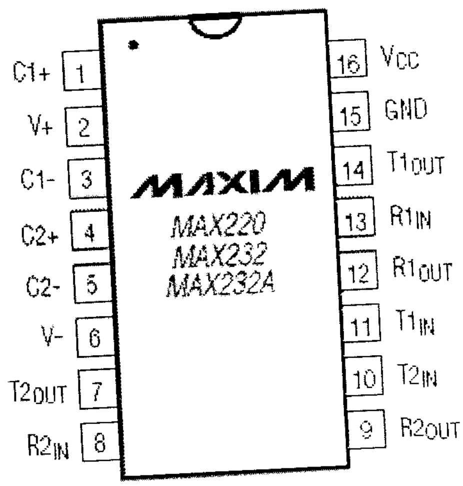


FIGURE 4.3 PIN DIAGRAM OF IC MAX 232

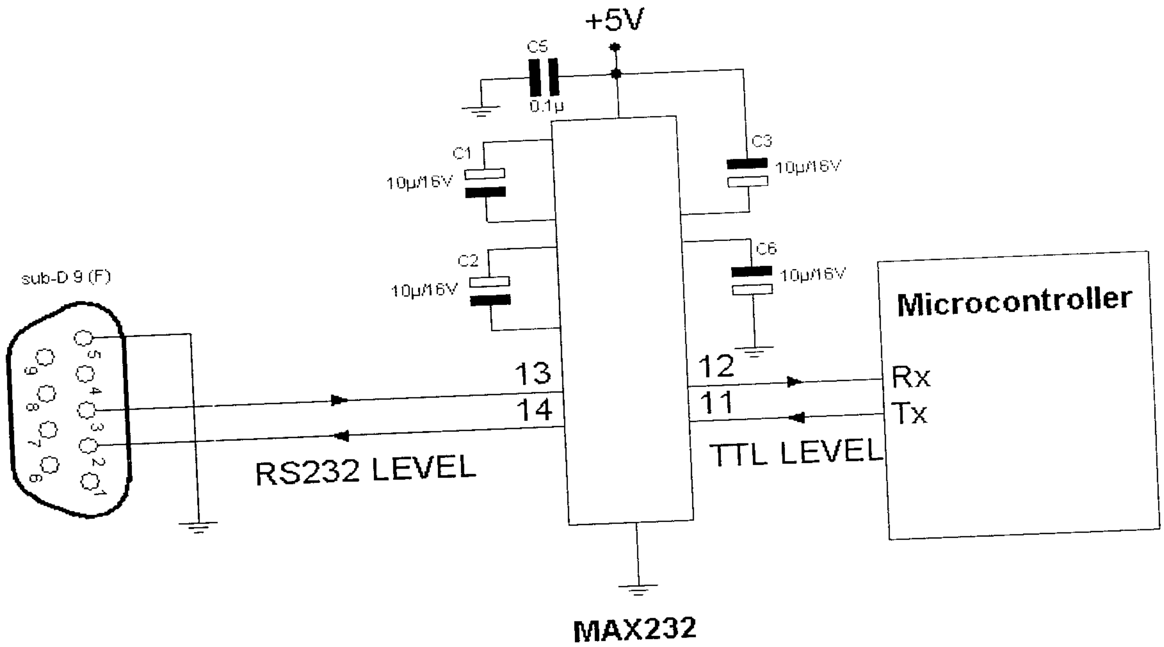


FIGURE 4.4 : INTERFACING OF IC MAX 232

The Max 232 has two sets of line drivers for transferring and receiving data. The line drivers used for TxD are T1 and T2, while the line drivers for RxD designated as R1 and R2 in many application only one of each is used. For example T1 and R1 are used together for TxD and RxD of the 89S51 and the second set is left unused. Notice in MAX 232 that the T1 line driver has a designation of T1 in and T1 out on pin numbers 11 and 14 respectively.

The T1 in pin is the TTL side and is connected to TxD of the Microcontroller, while T1 out is the RS232 side that is connected to the RxD pin of the RS232 DB connector. The R1 line driver has designation of R1 in, R1 out on pin numbers 13 and 12 respectively. Figure 4.4 shows the interfacing of IC MAX 232 with Microcontroller.

CHAPTER 5

LCD DISPLAY:

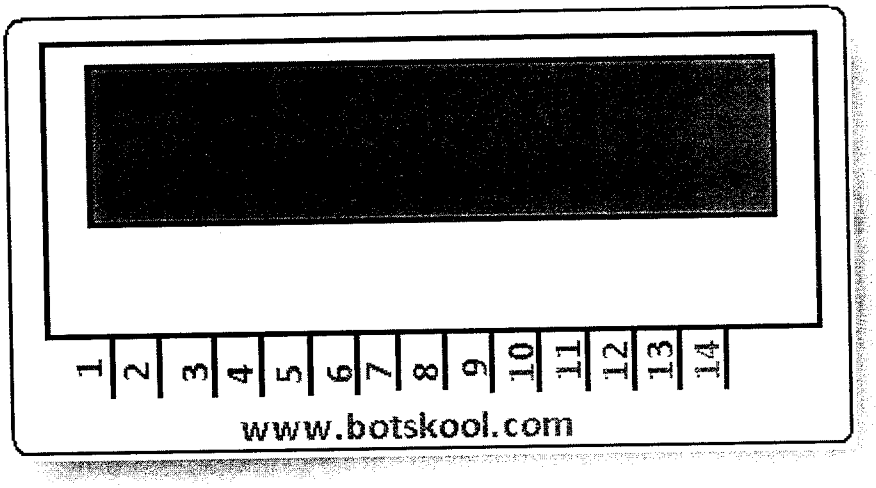


FIGURE 5.1 LCD DISPLAY

An **HD44780 Character LCD** is an industry standard liquid crystal display (LCD) device designed for interfacing with embedded electronics. These screens come in common configurations of 8x1 characters, 16x2, and 20x4 among others. The largest such configuration is 40x4 characters, but these are rare and are actually two separate 20x4 screens seamlessly joined together.

These screens are often found in copiers, fax machines, laser printers, industrial test equipment, networking equipment such as routers and storage devices, etc. These are not the kind of screens one would find in a cell phone, portable television, etc. They are limited to text only, with 8 customizable characters. Character LCDs can come with or without backlights. Backlights can be LED, fluorescent, or electroluminescent. Character LCDs use a standard 14-pin interface. If the screen has a backlight, it will have 16 pins. The pinouts are as follows:

1. Ground
2. VCC (+5V)
3. Contrast adjustment
4. Register Select (R/S)
5. Read/Write (R/W)
6. Clock (Enable)
7. Bit 0
8. Bit 1
9. Bit 2
10. Bit 3
11. Bit 4
12. Bit 5
13. Bit 6
14. Bit 7

Character LCDs can operate in 4-bit or 8-bit mode. In 4 bit mode, pins 7 through 10 are unused and the entire byte is sent to the screen using pins 11 through 14 by sending a nybble at a time.

The most commonly used LCDs found in the market today are 1 Line, 2 Line or 4 Line LCDs which have only one controller and support at most 80 characters, whereas LCDs supporting more than 80 characters make use of 2

HD44780 controllers. Apart from displaying some simple static characters you can create animated text scripts and a lot more!

Most LCDs with 1 controller has 14 Pins and 16 Pins (two extra pins are for back-light LED connections). Pin description is shown in figure 5.2.

Pin No.	Name	Description
1	VSS	GND
2	VCC	+5V
3	VEE	Contrast adjust
4	RS	0 = Command register 1 = Data register
5	R/W	0 = Write to LCD module 1 = Read from LCD module
6	EN	Enable
7	D0	Data bus line 0 (LSB)
8	D1	Data bus line 1
9	D2	Data bus line 2
10	D3	Data bus line 3
11	D4	Data bus line 4
12	D5	Data bus line 5
13	D6	Data bus line 6
14	D7	Data bus line 7 (MSB)

www.botskool.com

FIGURE 5.2 : PIN DESCRIPTION OF LCD DISPLAY

READ/WRITE (RW):

1.) RW= 0, the information is being written on LCD.

2.) RW=1, for reading from LCD. (Only one command that is "Get LCD status" is read commands all others are write command)

Read/Write is a control line. When RW is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively querying (or reading from) the LCD. Only one instruction ("Get LCD status") is a read command. All others are write commands, so RW will be low for majority of the time.

REGISTERS:

There are two very important registers in the LCD. The RS pin is used for their selection.

1.) RS= 0; The Instruction command code register, allows the user to send command such as clear display, cursor at home, etc.

2.) RS=1; the data register, allow user to send data to be displayed at LCD.

ENABLE (EN) PIN:

It is used to tell the LCD that we are sending it data. To enable LCD, a high to low pulse (of minimum length 450ns) should be given before sending any command/data to LCD.

STEPS TO PROGRAM:

1) Initialize the LCD.

2) Select the command or instruction register.

3) Set RW low (to write).

4) Send a high to low pulse on Enable pin.

5) Check if the LCD is busy.

6) Move to instruction or command function.

7) Repeat steps 4-7.

CHAPTER 6

POWER SUPPLY:

BLOCK DIAGRAM

The ac voltage, typically 220V rms, is connected to a transformer, which steps that ac voltage down to the level of the desired dc output. A diode rectifier then provides a full-wave rectified voltage that is initially filtered by a simple capacitor filter to produce a dc voltage. This resulting dc voltage usually has some ripple or ac voltage variation. A regulator circuit removes the ripples and also maintains the same dc value even if the input dc voltage varies, or the load connected to the output dc voltage changes. This voltage regulation is usually obtained using one of the popular voltage regulator IC units.

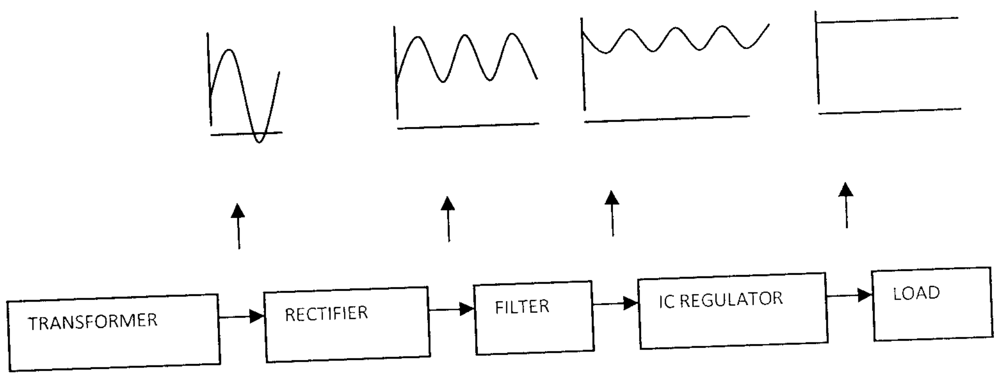


FIGURE 6.1 : BLOCK DIAGRAM OF POWER SUPPLY

WORKING PRINCIPLE

TRANSFORMER

The potential transformer will step down the power supply voltage (0-230V) to (0-6V) level. Then the secondary of the potential transformer will be connected to the Bridge rectifier.

BRIDGE RECTIFIER

When four diodes are connected as shown in figure 6.2, the circuit is called as bridge rectifier. The input to the circuit is applied to the diagonally opposite corners of the network and the output is taken from the remaining two corners.

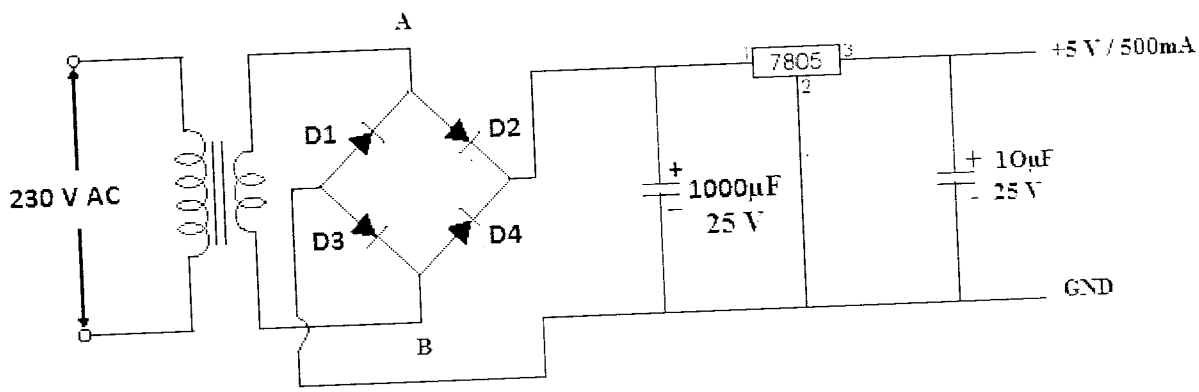


FIGURE 6.2 : CIRCUIT DIAGRAM OF POWER SUPPLY

During the positive half cycle, the diodes D2 & D3 will be forward biased and the diodes D1 & D4 will be reverse biased. Now current flow will be through D2 & D3 diodes. During the negative half cycle, the diodes D1 & D4 will be forward biased and the diodes D2 & D3 will be reverse biased. Now current flow will be through D1 & D4 diodes.

IC VOLTAGE REGULATORS

Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device and overload protection all in a single IC. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustably set voltage. The regulators can be selected for operation with load currents from hundreds of milli amperes to tens of amperes, corresponding to power ratings from milli watts to tens of watts.

An unregulated dc input voltage (Vi) is applied to input terminal of the fixed three-terminal voltage regulator. A regulated dc output voltage (Vo) will be obtained from the second terminal, with the third terminal connected to ground.

CHAPTER 7

CONCLUSION & FUTURE SCOPE

A low cost mobile patient monitoring system that utilizes Short Messaging Service (SMS) was designed, developed, and tested. This new system has a significantly reduced size and weight, which improves its versatility and mobility. Besides, SMS can be the most suitable, if not the only, method of data transmission in emergency situations in remote area where broadband data communications (like GPRS, EDGE … etc.) are not available.

This project can be extended in future by including more health parameters like blood pressure, blood glucose, respiratory rate etc. Transmission of ECG waveform through GSM network can also be done by sampling and encoding the ECG signals.

APPENDIX

AT COMMANDS FOR SHORT MESSAGE SERVICE (SMS)

1. SELECT MESSAGE SERVICE +CSMS

Description:

The supported services are originated short message (SMS-MO), terminated short message (SMS-MT) & Cell Broadcast Message (SMS-CB) services.

Syntax:

Command syntax: AT+CSMS=<service>

Command	Possible responses
AT+CSMS=0 *Note: SMS AT command Phase 2 version 4.7.0*	+CSMS: 1,1,1 OK *Note: SMS-MO, SMS-MT and SMS-CB supported*
AT+CSMS=1 *Note: SMS AT command Phase 2 +*	+CSMS: 1,1,1 *Note: SMS-MO, SMS-MT and SMS-CB supported*
AT+CSMS? *Note: Current values ?*	+CSMS: 0,1,1,1 OK *Note: GSM 03.40 and 03.41 (SMS AT command Phase 2 version 4.7.0*
AT+CSMS=? *Note: Possible services*	+CSMS: (0,1) OK

Defined values:

<service>

0: SMS AT commands are compatible with GSM 07.05 Phase 2 version 4.7.0.

1: SMS AT commands are compatible with GSM 07.05 Phase 2 + version.

2. NEW MESSAGE ACKNOWLEDGEMENT +CNMA

Description:

This command allows reception of a new message routed directly to the TE to be acknowledged.

In TEXT mode, only positive acknowledgement to the network (RP-ACK) is possible. In PDU mode, either positive (RP-ACK) or negative (RP-ERROR) acknowledgement to the network is possible. Acknowledge with +CNMA is possible only if the +CSMS parameter is set to 1 (+CSMS=1) when a +CMT or +CDS indication is shown (see +CNMI command). If no acknowledgement occurs within the network timeout, an RP-ERROR is sent to the network. The <mt> and <ds> parameters of the +CNMI command are then reset to zero (do not show new message indication).

Syntax:

Command syntax in text mode:

AT+CNMA

Command syntax in PDU mode:

AT+CNMA [= <n> [, <length> [<CR>

PDU is entered <ctrl-Z / ESC>]]]

Example of acknowledgement of a new message in TEXT mode

Command	Possible responses
AT+CMGF=1 *Note: Set TEXT mode*	OK *Note: TEXT mode valid*
AT+CNMI=2,2,0,0,0 *Note: <mt>=2*	OK
	+CMT: "123456","98/10/01,12:30 00+00",129,4 ,32,240, "15379",129,5<CR><LF> Received message *Note: message received*
AT+CNMA *Note: acknowledge the message received*	OK *Note: send positive acknowledgement to the network*
AT+CNMA *Note: try to acknowledge again*	+CMS ERROR: 340 *Note: no +CNMA acknowledgment expected*

Example of acknowledgement of a new message in PDU mode:

Command	*Possible responses*
AT+CMGF=0 *Note: Set PDU mode*	*OK* *Note: PDU mode valid*
	+CMT: ,29 *07913366003000F1240B913366920547F30000003003419404800B506215D42ECFE7E17319* *Note: message received*
AT+CNMA=2,<length > <CR> *... Pdu message ...* *<Ctrl-Z/ESC>* *Note: negative acknowledgement for the message.*	*OK* *Note: send a negative acknowledgement to the network (RP-ERROR) with PDU message (<ackpdu> format).*

Defined values:

<n>: Type of acknowledgement in PDU mode

0: send RP-ACK without PDU (same as TEXT mode)

1: send RP-ACK with optional PDU message

2: send RP-ERROR with optional PDU message

<length>: Length of the PDU message

3. PREFERRED MESSAGE STORAGE +CPMS

Description:

This command allows the message storage area to be selected (for reading, writing, etc).

Syntax:

Command syntax: AT+CPMS=<mem1>,[<mem2>]

Command	Possible responses
AT+CPMS=? *Note: Possible message storages*	+CPMS: (("SM","BM","SR"),("SM")) OK *Note:* *Read, list, delete: SMS, CBM or SMS Status Report* *Write, send: SMS*
AT+CPMS? *Note: Read*	+CPMS: "SM",3, 10,"SM",3,10 OK *Note: Read, write…SMS from/to SIM* *3 SMS are stored in SIM. 10 is the total memory available in SIM*
AT+CPMS="AM" *Note: Select false message storage*	+CMS ERROR: 302
AT+CPMS="BM" *Note: Select CBM message storage*	+CPMS: 2,20,3,10 OK *Note: Read, list, delete CBM from RAM 2 CBM are stored in RAM*
AT+CPMS? *Note: Read*	+CPMS: "BM",2,20,"SM",3,10 OK *Note:* *Read list, delete CBM from RAM* *Write SMS to SIM*

Defined values:

<mem1>: Memory used to list, read and delete messages. It can be:

-**"SM"**: SMS message storage in SIM (default)

-**"BM"**: CBM message storage (in volatile memory).

-**"SR":** Status Report message storage (in SIM if the EF-SMR file exists, otherwise in the ME non volatile memory)

<mem2>: Memory used to write and send messages

- **"SM"** : SMS message storage in SIM (default).

If the command is correct, the following message indication is sent:

+CPMS: <used1>,<total1>,<used2>,<total2>

When <mem1> is selected, all following +CMGL, +CMGR and +CMGD commands are related to the type of SMS stored in this memory.

4. PREFERRED MESSAGE FORMAT +CMGF

Description:

The message formats supported are text mode and PDU mode. In PDU mode, a complete SMS Message including all header information is given as a binary string (in hexadecimal format). Therefore, only the following set of characters is allowed: {'0','1','2','3','4','5','6','7','8','9', 'A', 'B','C','D','E','F'}. Each pair or characters are converted to a byte (e.g.: '41' is converted to the ASCII character 'A', whose ASCII code is 0x41 or 65). In Text mode, all commands and responses are in ASCII characters. The format selected is stored in EEPROM by the +CSAS command.

Syntax:

Command syntax: AT+CMGF

Command	Possible responses
AT+CMGF ? *Note: Current message format*	+CMGF: 1 OK *Note: Text mode*
AT+CMGF=? *Note: Possible message format*	+CMGF: (0-1) OK *Note: Text or PDU modes are available*

Example, sending an SMS Message in PDU mode:

Command	Possible responses
AT+CMGF=0 *Note: Set PDU mode*	OK *Note: PDU mode vaild*
AT+CMGS=14<CR> 0001030691214365000004C9E9340B *Note: Send complete MSG in PDU mode, no SC address*	+CMGS: 4 OK *Note: MSG correctly sent, <mr> is returned*

Defined values:

The <pdu> message is composed of the SC address (« 00 means no SC address given, use default SC address read with +CSCA command) and the TPDU message. In this example, the length of **octets** of the TPDU buffer is 14, coded as GSM Technical Specification 03.40. In this case the TPDU is: 0x01 0x03 0x06 0x91 0x21 0x43 0x65 0x00 0x00 0x04 0xC9 0xE9 0x34 0x0B, which means regarding GSM 03.40:

<fo> 0x01 (SMS-SUBMIT, no validity period)

<mr> (TP-MR) 0x03 (Message Reference)

<da> (TP-DA) 0x06 0x91 0x21 0x43 0x65 (destination address+123456)

<pid> (TP-PID) 0x00 (Protocol Identifier)

<dcs> (TP-DCS) 0x00 (Data Coding Scheme: 7 bits alphabet)

<length> (TP-UDL) 0x04 (User Data Length, 4 characters of text)

TP-UD 0xC9 0xE9 0x34 0x0B (User Data: ISSY)

TPDU in hexadecimal format must be converted into two ASCII characters, e.g. octet with hexadecimal value 0x2A is presented to the ME as two characters '2' (ASCII 50) and 'A' (ASCII 65).

5. SAVE SETTINGS +CSAS

Description:

All settings specified by the +CSCA and +CSMP commands are stored in EEPROM if the SIM card is a Phase 1 card or in the SIM card if it is a Phase 2 SIM card.

Syntax:

Command syntax: AT+CSAS

Command	Possible responses
AT+CSAS *Note: Store +CSAS and +CSMP parameters*	OK *Note: Parameters saved*

6. RESTORE SETTINGS +CRES

Description:

All settings specified in the +CSCA and +CSMP commands are restored from EEPROM if the SIM card is Phase 1 or from the SIM card if it is a Phase 2 SIM card.

Syntax:

Command syntax: AT+CRES

Command	Possible responses
AT+CRES *Note: Restore +CSAS and +CSMP parameters*	OK *Note: Parameters restored*

7. SHOW TEXT MODE PARAMETERS +CSDH

Description:

This command gives additional information on text mode result codes. This information is given in brackets in the +CMTI, +CMT, +CDS, +CMGR, +CMGL commands.

Syntax:

Command syntax: AT+CSDH

Command	Possible responses
AT+CSDH? *Note: Current value*	+CSDH: 0 OK *Note: Do not show header values*

8. NEW MESSAGE INDICATION +CNMI

Description:

This command selects the procedure for message reception from the network.

Command	Possible responses
AT+CNMI=2,1,0,0,0 *Note: <mt>=1*	OK
	AT+CMTI: "SM",1 *Note: message received*
AT+CNMI=2,2,0,0,0 *Note: <mt>=2*	OK
	+CMT: "123456","98/10/01,12:30 00+00",129,4 ,32,240, "15379",129,5<CR><LF> message received *Note: message received*
AT+CNMI=2,0,0,1,0 *Note: <ds>=1*	OK
Message to send <ctrl-Z> *Note: Send a message in text mode*	+CMGS: 7 OK *Note: Successful* AT+CMGS="+33146290800"<CR> *transmission*
	+CDS: 2, 116, "+33146290800", 145, "98/10/01,12:30:07+04", "98/10/01 12:30:08+04", 0 *Note: message was correctly delivered*

Syntax:

Command syntax: AT+CNMI=<mode>,<mt>,<bm>,<ds>,<bfr>

Defined values:

<mode>: controls the processing of unsolicited result codes

Only <mode>=2 is supported.

Any other value for <mode> (0,1 or 3) is accepted (return code will be OK), but the processing of unsolicited result codes will be the same as with<mode>=2.

<mode>

0: Buffer unsolicited result codes in the TA. If TA result code buffer is full, indications can be buffered in some other place or the oldest indications may be discarded and replaced with the new received indications

1: Discard indication and reject new received message unsolicited result codes when TA-TE link is reserved. Otherwise forward them directly to the TE

2: Buffer unsolicited result codes in the TA when TA-TE link is reserved and flush them to the TE after reservation. Otherwise forward them directly to the TE

3: Forward unsolicited result codes directly to the TE. TA-TE link specific inband used to embed result codes and data when TA is in on-line data mode

<mt>: sets the result code indication routing for SMS-DELIVERs. Default is 0.

<mt>

0: No SMS-DELIVER indications are routed.

1: SMS-DELIVERs are routed using unsolicited code: +CMTI: “SM”,<index>

2: SMS-DELIVERs (except class 2 messages) are routed using unsolicited code: +CMT:

[<alpha>,] <length> <CR> <LF> <pdu> (PDU mode) or +CMT: <oa>,[<alpha>,] <scts>

[,<tooa>, <fo>, <pid>, <dcs>, <sca>, <tosca>, <length>] <CR><LF><data> (text mode)

3: Class 3 SMS-DELIVERS are routed directly using code in <mt>=2 ; Message of other classes result in indication <mt>=1

<bm>: set the rules for storing received CBMs (Cell Broadcast Message) types depend on its coding scheme, the setting of Select CBM Types (+CSCB command) and <bm>. Default is 0.

<bm>

0: No CBM indications are routed to the TE. The CBMs are stored.

1: The CBM is stored and an indication of the memory location is routed to the customer application using unsolicited result code: +CBMI: “BM”, <index>

2: New CBMs are routed directly to the TE using unsolicited result code. +CBM:

<length><CR><LF><pdu> (PDU mode) or +CBM:<sn>,<mid>,<dcs>,<page>,<pages>(Text mode) <CR><LF> <data>

3: Class 3 CBMs: as <bm>=2. Other classes CBMs: as <bm>=1.

<ds> for SMS-STATUS-REPORTS. Default is 0.

<ds>

0: No SMS-STATUS-REPORTs are routed.

1: SMS-STATUS-REPORTs are routed using unsolicited code: +CDS: <length> <CR> <LF> <pdu> (PDU mode) or +CDS: <fo>,<mr>, [<ra>] , [<tora>], <scts>,<dt>,<st> (Text mode)

2: SMS-STATUS-REPORTs are stored and routed using the unsolicited result code: +CDSI:

"SR",<index>

<bfr> Default is 0.

<bfr>

0: TA buffer of unsolicited result codes defined within this command is flushed to the TE when <mode> 1…3 is entered (OK response shall be given before flushing the codes)

1: TA buffer of unsolicited result codes defined within this command is cleared when <mode> 1…3 is entered.

9. READ MESSAGE +CMGR

Description:

This command allows the application to read stored messages. The messages are read from the memory selected by **+CPMS** command.

Syntax:

Command syntax: AT+CMGR=<index>

Response syntax for text mode:

+CMGR:<stat>,<oa>,[<alpha>,]<scts>[,<tooa>,<fo>,<pid>,<dcs>,<sca>,<tosca>, <length>] <CR><LF> <data> (for **SMS-DELIVER** only)

+CMGR: <stat>,<da>,[<alpha>,] [,<toda>,<fo>,<pid>,<dcs>, [<vp>], <sca>, <tosca>,<length>]<CR><LF> <data> (for **SMS-SUBMIT** only)

+CMGR: <stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st> (for **SMS-STATUS-REPORT** only)

Response syntax for PDU mode:

+CMGR: <stat>, [<alpha>] ,<length> <CR><LF> <pdu>

A message read with status "REC UNREAD" will be updated in memory with the status "REC READ".

Example:

Command	Possible responses
	AT+CMTI: "SM",1 *Note: New message received*
AT+CMGR=1 *Note: Read the message*	+CMGR: "REC UNREAD","0146290800", "98/10/01,18:22:11+00",<CR><LF> ABCdefGHI OK
AT+CMGR=1 *Note: Read the message again*	+CMGR: "REC UNREAD","0146290800", "98/10/01,18:22:11+00",<CR><LF> ABCdefGHI OK *Note: Message is read now*
AT+CMGR=2 *Note: Read at a wrong index*	+CMS ERROR: 321 *Note: Error: invalid index*
AT+CMGF=0 ;+CMGR=1 *Note: In PDU mode*	+CMGR: 2,,<length> <CR><LF><pdu> OK *Note: Message is stored but unsent, no <alpha>field*

Command	Possible responses
AT+CMGF=1;+CPMS="SR";+CNMI=,,,2 Reset to text mode, set read memory to "SR", and allow storage of further SMS Status Report into "SR" memory	OK
AT+CMSS=3 Send an SMS previously stored	+CMSS: 160 OK
	+CDSI: "SR",1 New SMS Status Report stored in "SR" memory at index 1
AT+CMGR=1 Read the SMS Status Report	+CMGR: "READ",6,160, "+33612345678",129,"01/05/31,15:15:09+00", "01/05/31,15:15:09+00",0 OK

10. LIST MESSAGE +CMGL

Description:

This command allows the application to read stored messages, by indicating the type of the message to read. The messages are read from the memory selected by the **+CPMS** command.

Syntax:

Command syntax: AT+CMGL=<stat>

Response syntax for text mode:

+CMGL: <index>,<stat>,<da/oa>[,<alpha>], [<scts>, <tooa/toda>, <length>] <CR><LF><data>
(for **SMS-DELIVER and SMS-SUBMIT**, may be followed by other <CR><LF>+CMGL:<index>…)

+CMGL: <index>,<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st>
(for **SMS-STATUSREPORT** only, may be followed by other <CR><LF>+CMGL:<index>…)

Response syntax for PDU mode:

+CMGL: <index>,<stat>, [<alpha>], <length> <CR><LF> <pdu>

(for **SMS-DELIVER**, **SMSSUBMIT** and **SMS-STATUS-REPORT**, may be followed by other <CR><LF>+CMGL:<index>…)

Command	Possible responses
AT+CMGL="REC UNREAD" *Note: List unread messages in text mode*	+CMGL: 1,"REC UNREAD","0146290800", <CR><LF> Unread message ! +CMGL: 3,"REC UNREAD", "46290800", <CR><LF> Another message unread! OK *Note: 2 messages are unread, these messages will then have their status changed to "REC READ" (+CSDH:0)*
AT+CMGL="REC READ" *Note: List read messages in text mode*	+CMGL: 2,"REC READ","0146290800", <CR><LF> Keep cool OK
AT+CMGL="STO SENT" *Note: List stored and sent messages in text mode*	OK *Note: No message found*
AT+CMGL=1 *Note: List read messages in PDU mode*	+CMGL: 1,1,,26 <CR><LF> 07913366003000F3040B913366920547F40013001190412530400741AA8E5A9C5201 OK

Defined values:

<stat> possible values (status of messages in memory):

Text mode possible values	PDU mode possible values	Status of messages in memory
"REC UNREAD"	0	received unread messages
"REC READ"	1	received read messages
"STO UNSENT"	2	stored unsent messages
"STO SENT"	3	stored sent messages
"ALL"	4	all messages

11. SEND MESSAGE +CMGS

Description:

The <address> field is the address of the terminal to which the message is sent. To send the message, simply type, <ctrl-Z> character (ASCII 26). The text can contain all existing characters except <ctrl-Z> and <ESC> (ASCII 27). This command can be aborted using the <ESC> character when entering text. In PDU mode, only hexadecimal characters are used ('0'…'9','A'…'F').

Syntax:

Command syntax in text mode:

AT+CMGS= <da> [,<toda>] <CR>

text is entered <ctrl-Z / ESC >

Command syntax in PDU mode:

AT+CMGS= <length> <CR>

PDU is entered <ctrl-Z / ESC >

Command	Possible responses
AT+CMGS="+33146290800"<CR> Please call me soon, Fred. <ctrl-Z> *Note: Send a message in text mode*	+CMGS: <mr> OK *Note: Successful transmission*
AT+CMGS=<length><CR><pdu><ctrl-Z> *Note: Send a message in PDU mode*	+CMGS: <mr> OK *Note: Successful transmission*

The message reference, <mr>, which is returned to the application is allocated by the product. This number begins with 0 and is incremented by one for each outgoing message (successful and failure cases); it is cyclic on one byte (0 follows 255).

12. WRITE MESSAGE TO MEMORY +CMGW

Description:

This command stores a message in memory (either SMS-SUBMIT or SMS-DELIVERS). The memory location <index> is returned (no choice possible as with phonebooks +CPBW). Text or PDU is entered as described for the Send Message +CMGS command.

Syntax:

Command syntax in text mode: (<index> is returned in both cases)

AT+CMGW= <oa/da> [,<tooa/toda> [,<stat>]] <CR>

enter text <ctrl-Z / ESC>

Command syntax in PDU mode:

AT+CMGW= <length> [,<stat>] <CR>

give PDU <ctrl-Z / ESC>

Response syntax:

+CMGW: <index> or +CMS ERROR: <err> if writing fails

Command	Possible responses
AT+CMGW="+33146290800"<CR> Hello how are you ?<ctrl-Z> *Note: Write a message in text mode*	+CMGW: 4 OK *Note: Message stored in index 4*
AT+CMGW=<length><CR><pdu><ctrl-Z> *Note: Write a message in PDU mode*	+CMGW: <index> OK *Note: Message stored in <index>*

Defined values:

Parameter Definition:

<oa/da>: Originating or Destination Address Value in string format.

<tooa/toda>: Type of Originating / Destination Address.

<stat>: Integer type in PDU mode (default 2 for +CMGW), or string type in text mode (default "STO UNSENT" for +CMGW). Indicates the status of message in memory. If <stat> is omitted, the stored message is considered as a message to send.

<stat>

0: "REC UNREAD"

1: "REC READ"

2: "STO UNSENT"

3: "STO SENT"

<length>: Length of the actual data unit in octets

13. SEND MESSAGE FROM STORAGE +CMSS

Description:

This command sends a message stored at location value <index>.

Syntax:

Command syntax: AT+CMSS=<index>[,<da> [,<toda>]]

Response syntax:

+CMSS: <mr> or +CMS ERROR: <err> if sending fails

If a new recipient address <da> is given, it will be used instead of the one stored with the Message.

Command	Possible responses
AT+CMGW=0660123456<CR> Today is my birthday *Note:*	+CMGW: 5 OK *Note:Message stored with index 5*
AT+CMSS=5, 0680654321 *Note: Send the message 5 to a different destination number*	AT+CMSS:<mr> OK *Note: Successful transmission*
AT+CMSS=5, 0680654321 *Note: Send the message 5 to a different destination number*	+CMSS:<mr> OK *Note: Successful transmission*

14. SET TEXT MODE PARAMETERS +CSMP

Description:

This command selects a value for <vp>, <pid>, and <dcs>.

Syntax:

Command syntax: AT+CSMP=<fo>, <vp>, <pid>,<dcs>

Command	Possible responses
AT+CSMP? Note: current values	+CSMP: 0,0,0,0 OK Note: No validity period <dcs>= PCCP437 alphabet (8 bits → 7 bits)
AT+CMPS=17,23,64,244 Note:<vp> = 23 (2 hours, relative format) <dcs> = GSM 8 bits alphabet	OK Note: Command correct

Defined values:

The **<fo>** byte comprises 6 different fields:

B7	B6	B5	b4	b3	b2	b1	b0
RP	*UDHI*	*SRR*	*VPF*		*RD*	*MTI*	

RP: Reply Path, not used in text mode.

UDHI: User Data Header Information, b6=1 if the beginning of the User Data field contains a Header in addition to the short message. This option is not supported in +CSMP command, but can be used in PDU mode (+CMGS).

SRR: Status Report Request, b5=1 if a status report is requested. This mode is supported.

VPF: Validity Period Format

b4=0 & b3=0 -> <vp> field is not present

b4=1 & b3=0 -> <vp> field is present in relative format

Others formats (absolute & enhanced) are not supported.

RD: Reject Duplicates, b2=1 to instruct the SC to reject an SMS-SUBMIT for an SM still held in the SC which has the same <mr> and the same <da> as the previously submitted SM from the same <oa>.

MTI: Message Type Indicator

b1=0 & b0=0 -> SMS-DELIVER (in the direction SC to MS)

b1=0 & b0=1 -> SMS-SUBMIT (in the direction MS to SC)

In text mode <vp> is only coded in "relative" format. The default value is 167 (24 hours). This means that one octet can describe different values:

VP value	Validity period value
0 to 143	(VP + 1) x 5 minutes (up to 12 hours)
144 to 167	12 hours + ((VP – 143) x 30 minutes)
168 to 196	(VP – 166) x 1 day
197 to 255	(VP – 192) x 1 week

<pid> is used to indicate the higher layer protocol being used or indicates interworking with a certain type of telematic device. For example, 0x22 is for group 3 telefax, 0x24 is for voice telephone, 0x25 is for ERMES.

<dcs> is used to determine the way the information is encoded. Compressed text is not supported. Only GSM default alphabet, 8 bit data and UCS2 alphabet are supported.

REFERENCES

1. Ashraf A Tahat (2009) 'Body Temperature and Electrocardiogram Monitoring Using an SMS-Based Telemedicine System', IEEE.

2. D. Bottazi, A. Corradi, and R. Montanari, "Context-aware middleware solutions for anytime and anywhere emergency assistance to elderly people," *IEEE Communications Magazine*, vol.44, pages 82–90, 2006.

3. http://www.gsmworld.com

4. http://www.developershome.com