ELECTRONIC ENERGY METER WITH INSTANT BILLING

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

Bachelor of Technology

In

Electronics and Communication Engineering

By

ANUJ GARG (061026)

TARNISH GOYAL (061133)

TARUN CHAUHAN (061134)

Under the Supervision of

Mr. VIVEK KUMAR SEHGAL

Jaypee University of Information Technology

Waknaghat , Solan - 173 234, Himachal Pradesh

Certificate

This is to certify that the project report entitled "ELECTRONIC ENERGY METER WITH INSTANT BILLING", submitted by Anuj Garg (061026), Tarnish Goyal (061133) and Tarun Chauhan(061134) 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.

Date: **Mr. Vivek Kumar**

Sehgal

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

1. **Anuj Garg**
   Roll No.061026
2. **Tarnish Goyal**
   Roll No.061133
3. **Tarun Chauhan**
   Roll No.061134

Acknowledgement

We would like to express our gratitude to **Prof. Dr. SUNIL BHOOSHAN, Head of Department, Electronics and Communication** for encouraging and inspiring us. Our deepest and most sincere thanks to **Mr. Vivek Kumar Sehgal , Lecturer, Department of Electronics and Communication,** our guide, who in spite of his busy schedule was able to spare time to help and guide us in doing this project by giving his expert guidance, encouragement and valuable suggestion.

. A special thanks to **Mr. MANOJ PANDEY** who provided all possible help. His tireless efforts and willingness to clear all our doubts helped us immensely.

We would like to thank our parents for continuously raising our confidence and for their constant support.

(Anuj Garg) (Tarnish Goyal) (Tarun Chauhan)

Date:

Table of Content

Index of diagram

Abstract

The aim of the project is to automate the billing of energy meter. It is just like postpaid mobile connection. In this project the front end is user friendly and the employees can work on this software with minimum knowledge of computers. Employees can read the meter by sitting in the office. For front end designing c language is used. This project is useful for billing purpose in electricity board authority. We have connected a gsm modem to the energy meter. Each modem will be having its own sim (ususal mobile phone sim). We have a pc connected to other modem, which contains the front end data base. The front end is built in C. The module is designed such that user can have the complete usage details about the energy meter. Just like postpaid mobile connections , we can know our due bill instantly and can even pay for it .We have used a SIM card which is implemented in energy meter and it sends a message to the user about the due bill. We used LCD in the hardware module for the user interface. The LCD displays the current usage and units consumed . After the usage of each unit the amount and total units will be incremented .whenever user wants to know his/her bill she can message a given code to sim card attached to the meter and can know his/her present bill. The user can pay the amount just by knowing the given code which is fed in the meter .

Chapter 1

Introduction

In this project we have shown the concept of postpaid energy meter which will automatically sense the energy used in the home and when it reaches to that value which is initially fed in the hardware it will disconnect the power line.

A user interface is given in the hardware for user interface which will interact with the user with hardware, through user interface user can set a value or we can say a credit limit after which the user wants to be informed about that. GSM Decoder Section is that interface in this project, which will make the user to interact with the hardware so that user is able to initialize the hardware or can initially set the value.

Power consumption measuring circuit will side by side measures the power which is being used in the house. The measured values are then sent to the microcontroller for further process which includes comparison of the measured value with the entered one using the mobile Phone.

1.2 Block Diagram

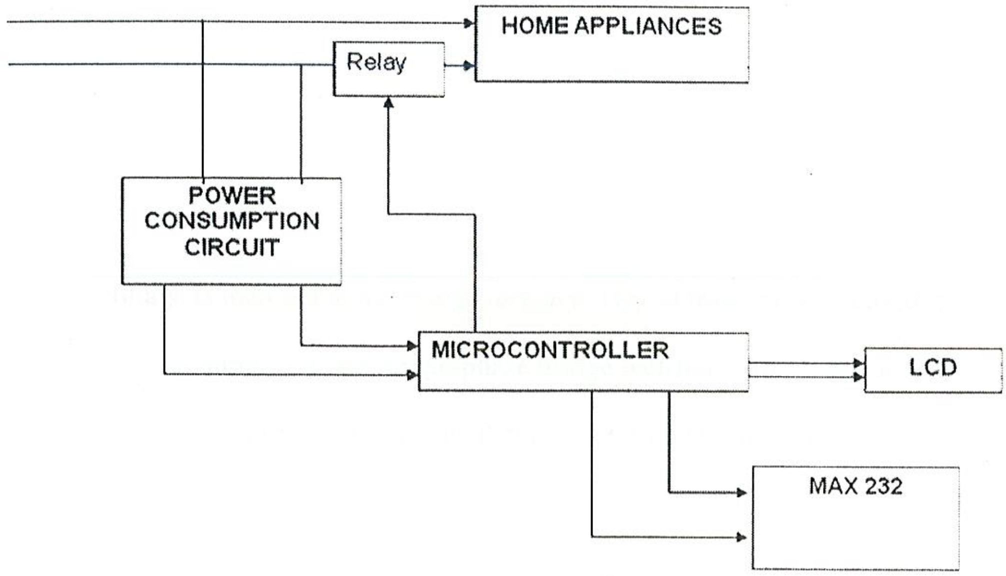

Fig 1.1 Block diagram

1.3 MODULES DESCRIPTION

POWER SUPPLY DESCRIPTION:

The power supply circuit comprises of four basic parts:

The transformer steps down the 220 V a/c. into 12 V a/c. The transformer work on the principle of magnetic induction, where two coils: primary and secondary are wound around an iron core. The two coils are physically insulated from each other in such a way that passing an a/c. current through the primary coil creates a changing voltage in the primary coil and a changing magnetic field in the core. This in turn induces a varying a/c. voltage in the secondary coil.

The a/c. voltage is then fed to the bridge rectifier. The rectifier circuit is used in most electronic power supplies is the single-phase bridge rectifier with capacitor filtering, usually followed by a linear voltage regulator. A rectifier circuit is necessary to convert a signal having zero average value into a non-zero average value. A rectifier transforms alternating current into direct current by limiting or regulating the direction of flow of current. The output resulting from a rectifier is a pulsating D.C. voltage. This voltage is not appropriate for the components that are going to work through it.

<u>THE MICROCONTROLER</u>:

AT89C51 FROM ATMEL CORPORATION

This popular 8051 chip has on-chip ROM in the form of flash memory. This is ideal for fast development since flash memory can be erased in seconds compared to twenty minutes or more needed for the earlier versions of the 8051. To use the AT89C51 to develop a microcontroller-based system requires a ROM burner that supports flash memory: However, a ROM eraser is not needed. Notice that in flash memory you must erase the entire contents of ROM in order to program it again. The PROM burner does this erasing of flash itself and this is why a separate burner is not needed. To eliminate the need for a PROM burner Atmel is working on a version of the AT89C51 that can be programmed by the serial COM port of the PC.

FEATURES OF AT89C51

- 4K on-chip ROM
- 128 bytes internal RAM (8-bit)
- 32 I/O pins
- Two 16-bit timers
- Six Interrupts
- Serial programming facility
- 40 pin Dual-in-line Package

THE ELECTROMAGNETIC RELAY

The electromagnetic relay consists of a multi-turn coil, wound on an iron core, to form an electromagnet. When the coil is energised, by passing current through it, the core becomes temporarily magnetised. The magnetised core attracts the iron armature. The armature is pivoted which causes it to operate one or more sets of contacts.

When the coil is de-energised the armature and contacts are released. The coil can be energised from a low power source such as a transistor while the contacts can switch high powers such as the mains supply. The relay can also be situated remotely from the control source. Relays can generate a very high voltage across the coil when switched off. This can damage other components in the circuit. To prevent this a diode is connected across the coil.

Liquid Crystal Display

Liquid crystal displays (LCD) are widely used in recent years as compares to LEDs. This is due to the declining prices of LCD, the ability to display numbers, characters and graphics, incorporation of a refreshing controller into the LCD, their by relieving the CPU of the task of refreshing the LCD and also the ease of programming for characters and graphics. HD 44780 based LCDs are most commonly used.

CHAPTER 2

CIRCUIT MODELLING

The circuit has a power supply module, a relay circuit module, an electronic valve, LCD microcontroller interfacing, microcontroller GSM module interfacing and microcontroller ADC interfacing. The components of the circuit are as follows

- Circuit Diagram
- Power supply module
- Relay and isolation circuit
- Interfacing with ADC
- Interfacing with GSM
- interfacing DB-9 and MAX 232

Now the explanation of the circuits is as follows

2.1 Circuit Diagram

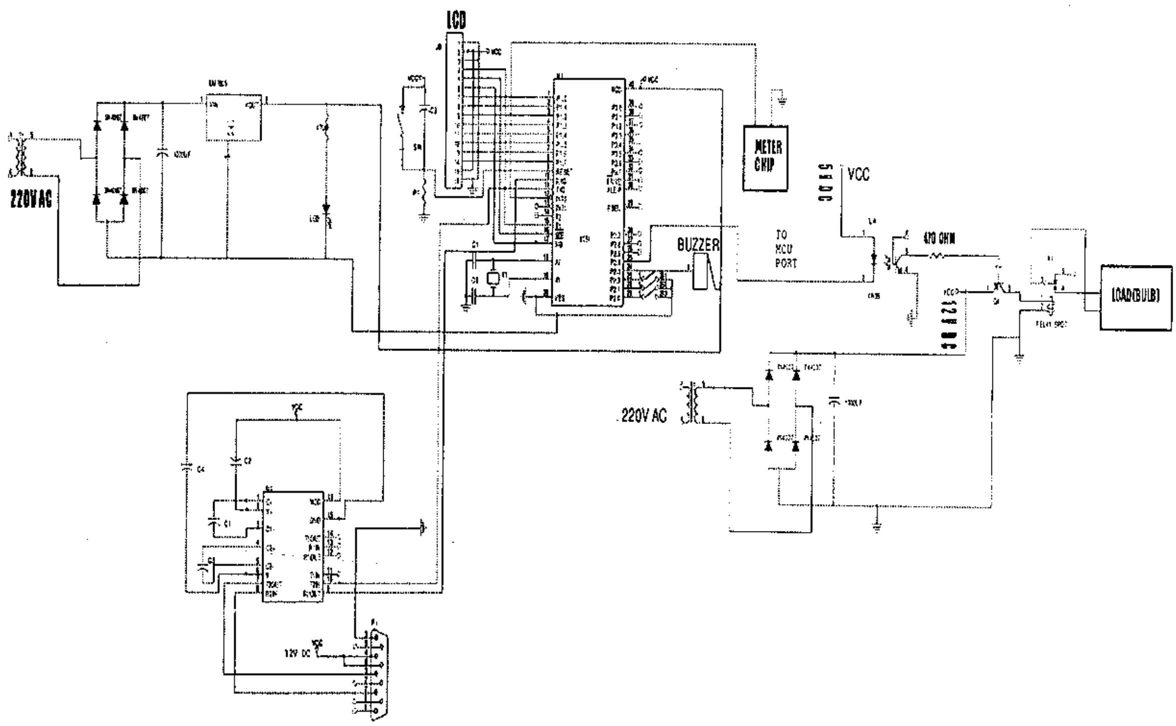

Fig 2.1 Circuit Diagram

2.2 Power Supply

The power supply module is used to supply the power to the circuit. It gives power to both, the relay circuit, as well as the microcontroller. It consists of a transformer (220/12), a 7805 IC, a resistance (), capacitance () and a bridge rectifier circuit consisting of 4 diodes. This power supply is used to give the current to the microcontroller, supply power to the relay circuit and the valve.

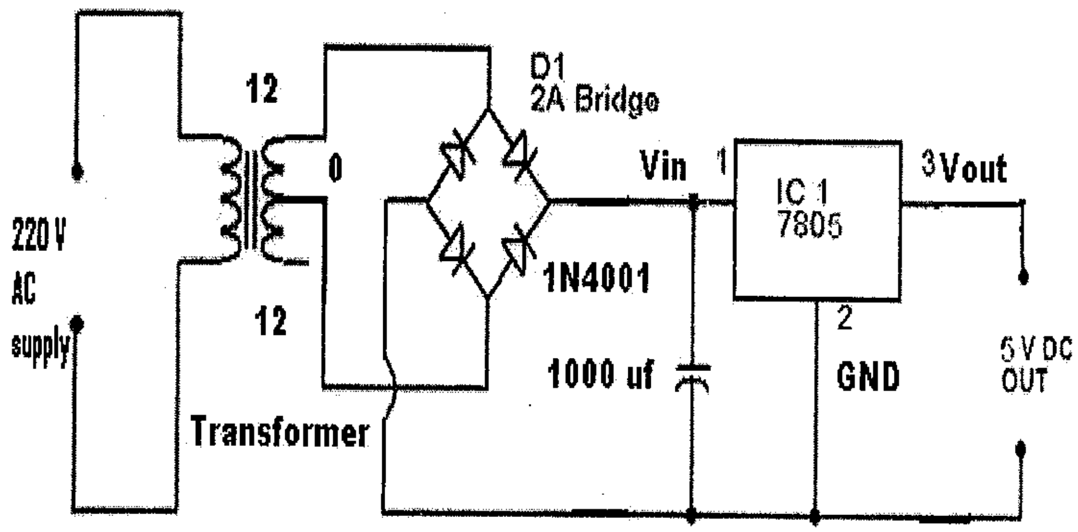

Fig 2.2 Power supply

First a supply of 220V AC is fed to the transformer. The transformer steps it down to 12V AC. Now this resultant output is passed through a bridge rectifier which then converts the AC voltage to the DC voltage. Now to reduce the ripples we pass the output through a shunt capacitor in parallel. The shunt capacitor has a property to allow only AC voltage to pass through. So AC voltage is passed through it and then we have the DC voltage of 12V. This is passed through the 7805 voltage regulator IC which now converts it to the 5V dc. Below is shown the circuit diagram of the power supply.

2.3 Isolation and relay circuit

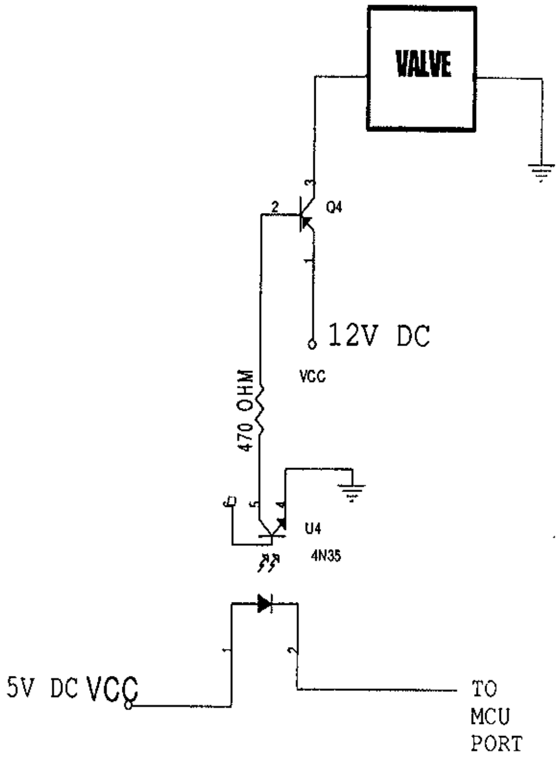

Fig 2.3 Isolation and relay circuit

The microcontroller provides the power of only 5V, whereas the voltage required to operate the valve is 12V. So to solve this problem we use an isolation circuit. The power to operate the valve is derived from the power supply module and then the signal is derived from the microcontroller.

2.4 Interfacing the ADC and microcontroller

Microcontroller is a DC device, i.e. it works on a DC voltage. Now to operate the microcontroller we require an ADC, i.e. analog to digital converter. This device now converts the AC to DC voltage and then the microcontroller is operated. Microcontroller is interfaced with the ADC in the following manner.

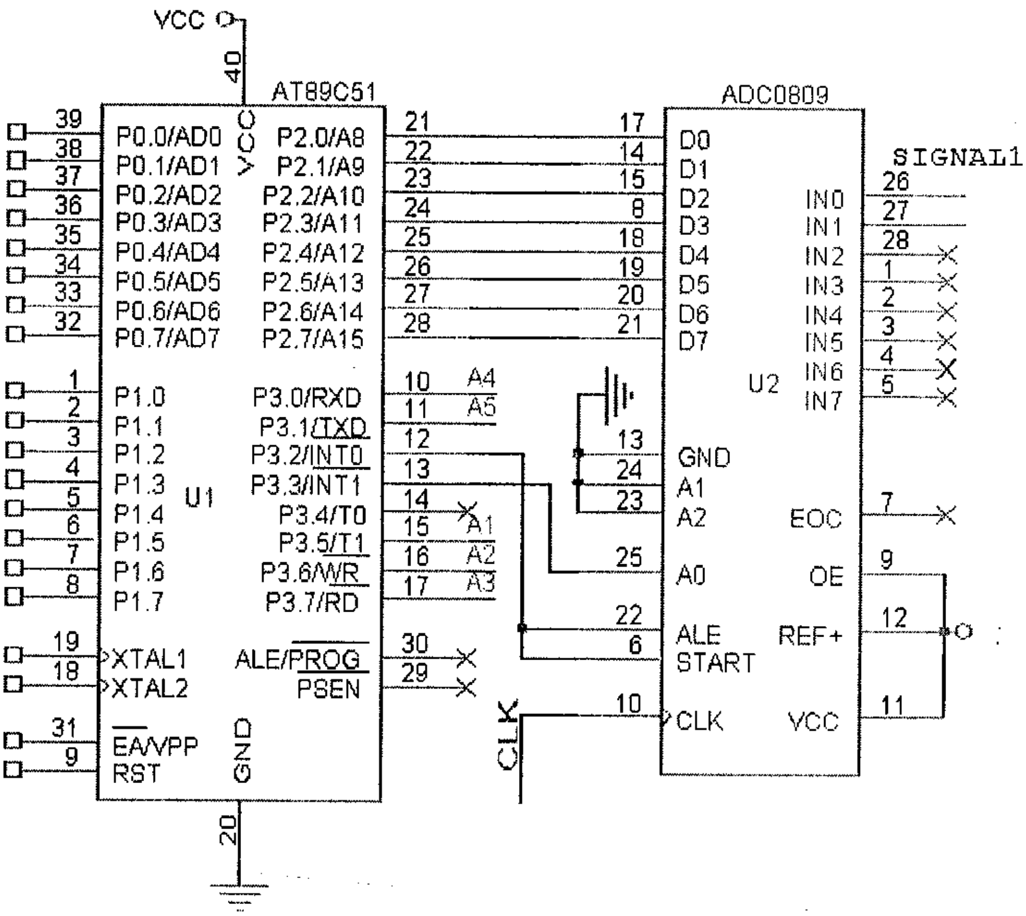

Fig 2.4 Interfacing microcontroller with the ADC

2.5 Interfacing with the GSM module

The microcontroller output is not compatible with the GSM module. To make it compatible we require the DB9 connector and the MAX 232 connector. This will enable the microcontroller to send a message to a predefined phone number. The pin diagram is as follows:

2.6 Display and alarm system

The system also consists of a display system having a LED and an alarm system. When the meter is working LED glows .when the GSM module is sending the . The buzzer will also buzz indicating the message is sent or recieved.

2.7 Interfacing MAX 232 with DB9 connector

When communicating with various micro processors one needs to convert the RS232 levels down to lower levels, typically 3.3 or 5.0 **Volts.** Here is a cheap and simple way to do that. **Serial RS-232** (V.24) communication works with voltages -15V to +15V for high and low. On the other hand, **TTL** logic operates between 0V and +5V. Modern low power consumption logic operates in the range of 0V and +3.3V or even lower.

RS-232	TTL	Logic
-15V … -3V	+2V … +5V	High
+3V … +15V	0V … +0.8V	Low

Thus the RS-232 signal levels are far too high **TTL electronics,** and the negative RS-232 voltage for high can't be handled at all by computer logic. To receive serial data from an RS-232 interface the voltage has to be reduced. Also the low and high voltage level has to be inverted. This level converter uses a Max232 and capacitors.

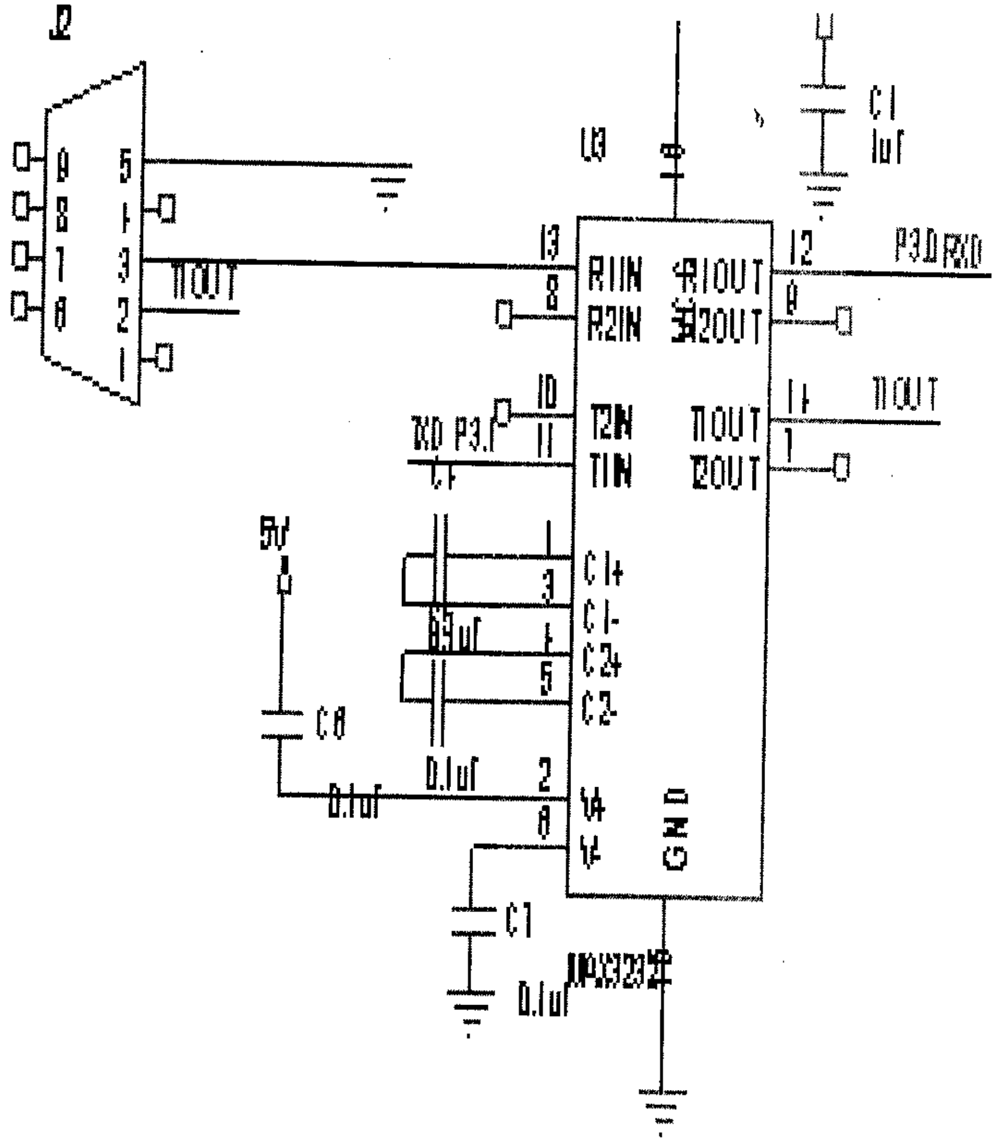

Fig 2.5 Interfacing RS 232-DB9 with MAX 232

2.8 Interfacing LCD with microcontroller

LCD module has 8-bit data interface and control pins.

One can send data as 8-bit or in pair of two 4-bit nibbles. To display any character on LCD micro controller has to send its ASCII value to the data bus of LCD. For e.g. to display 'AB' microcontroller has to send two hex bytes 41h and 42h respectively. LCD display used here is having 16x2 size. It means 2 lines each with 16 characters.

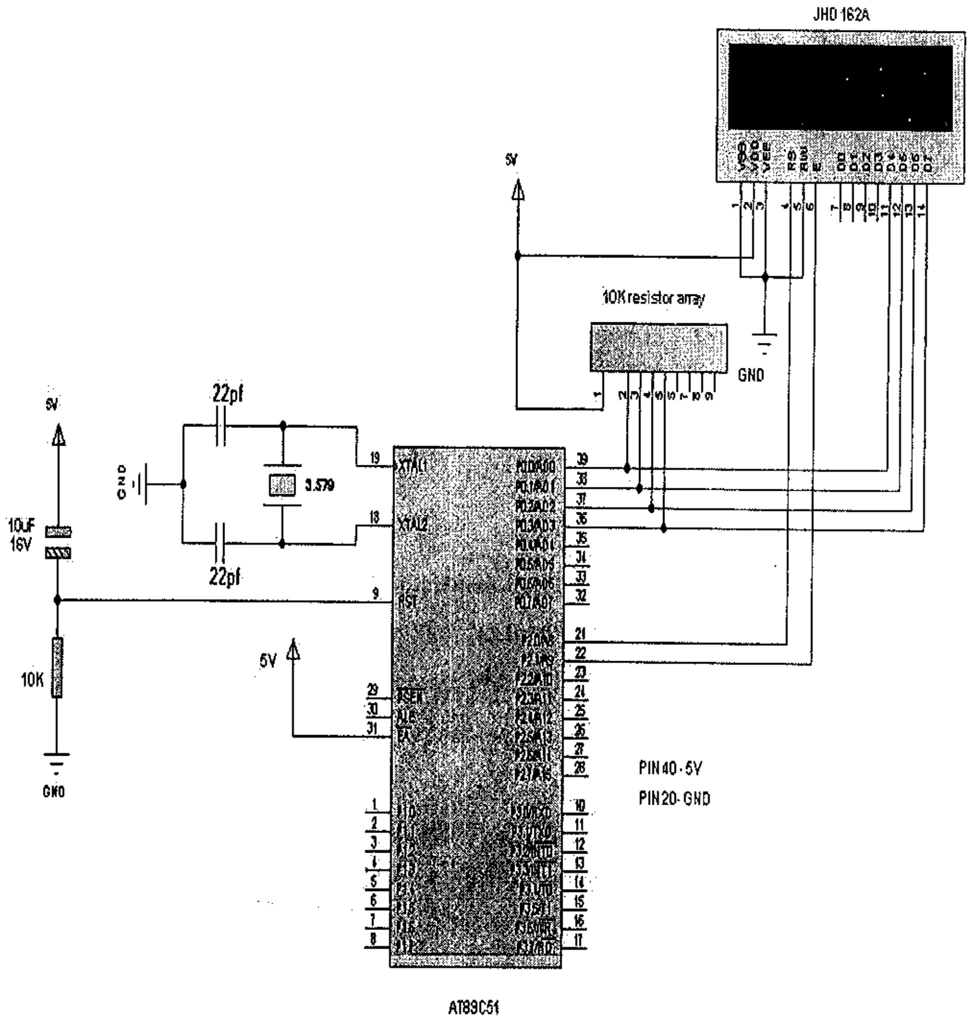

Fig 2.6 LCD interfacing with microcontroller

CHAPTER 3
HARDWARE DESCRIPTION

Each component is explained with its basic theory which helps us get a basic detailed insight of as to how that particular component is working and its use in this particular project. Hence a defined role of each and every component is contained in this chapter. The basic components are as follows.

3.1 Power Consumption Circuit

Power measurement in AC circuits can be quite a bit more complex than with DC circuits for the simple reason that phase shift complicates the matter beyond multiplying voltage by current figures obtained with meters. What is needed is an instrument able to determine the product (multiplication) of instantaneous voltage and current. Fortunately, the common electrodynamometer movement with its stationary and moving coil does a fine job of this.

Three phase power measurement can be accomplished using two dynamometer movements with a common shaft linking the two moving coils together so that a single pointer registers power on a meter movement scale. This, obviously, makes for a rather expensive and complex movement mechanism, but it is a workable solution.

An ingenious method of deriving an electronic power meter (one that generates an electric signal representing power in the system rather than merely move a pointer) is based on the Hall effect. The Hall effect is an unusual effect first noticed by E. H. Hall in 1879, whereby a voltage is generated along the width of a current-carrying conductor exposed to a perpendicular magnetic field:

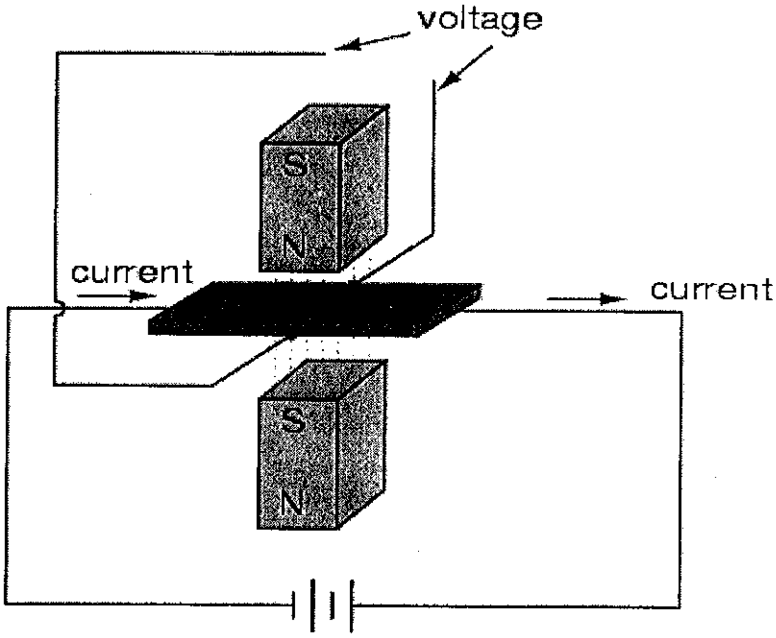

Fig 3.1

The voltage generated across the width of the flat, rectangular conductor is directly proportional to both the magnitude of the current through it and the strength of the magnetic field. Mathematically, it is a product (multiplication) of these two variables. The amount of "Hall Voltage" produced for any given set of conditions also depends on the type of material used for the flat, rectangular conductor. It has been found that specially prepared "semiconductor" materials produce a greater Hall voltage than do metals, and so modern Hall Effect devices are made of these.

It makes sense then that if we were to build a device using a Hall-effect sensor where the current through the conductor was pushed by AC voltage from an external circuit and the magnetic field was set up by a pair or wire coils energized by the current of the AC power circuit, the Hall voltage would be in direct proportion to the multiple of circuit current and voltage. Having no mass to move (unlike an electromechanical movement), this device is able to provide instantaneous power measurement.

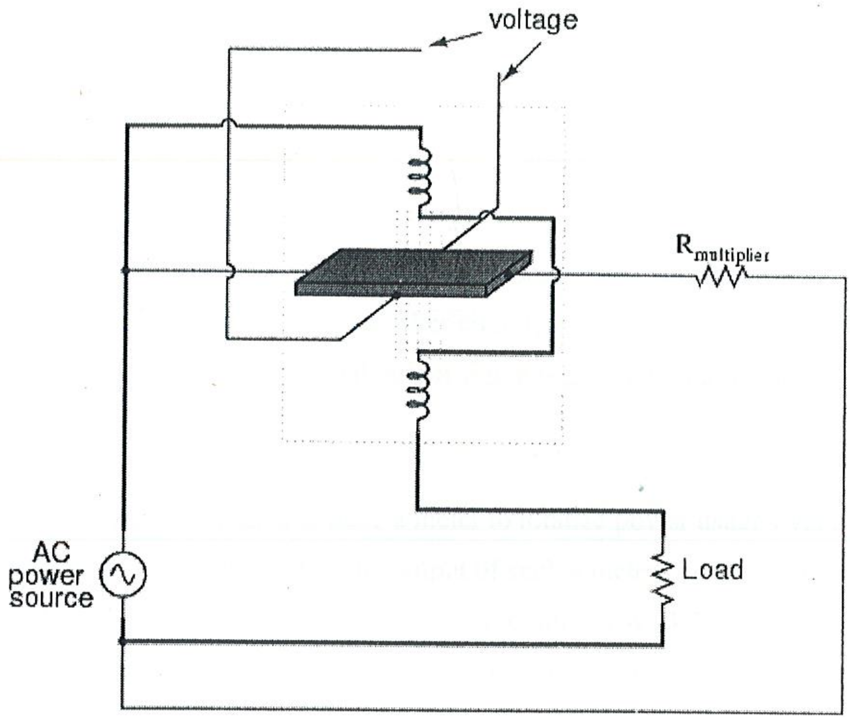

Fig 3.2

Not only will the output voltage of the Hall effect device be the representation of instantaneous power at any point in time, but it will also be a DC signal! This is because the Hall voltage polarity is dependent upon both the polarity of the magnetic field and the direction of current through the conductor. If both current direction and magnetic field polarity reverses -- as it would ever half-cycle of the AC power -- the output voltage polarity will stay the same.

If voltage and current in the power circuit are 90° out of phase (a power factor of zero, meaning no real power delivered to the load), the alternate peaks of Hall device current and magnetic field will never coincide with each other: when one is at its peak, the other will be zero. At those points in time, the Hall output voltage will likewise be zero, being the product (multiplication) of current and magnetic field strength. Between those points in time, the Hall output voltage will fluctuate equally between positive and negative, generating a signal corresponding to the instantaneous absorption and release of power through the reactive load. The net DC output voltage will be zero, indicating zero true power in the circuit.

Any phase shift between voltage and current in the power circuit less than 90^{0} will result in a Hall output voltage that oscillates between positive and negative, but spends more time positive than negative. Consequently there will be a net DC output voltage. Conditioned through a low-pass filter circuit, this net DC voltage can be separated from the AC mixed with it, the final output signal registered on a sensitive DC meter movement.

Often it is useful to have a meter to totalize power usage over a period of time rather than instantaneously. The output of such a meter can be set in units of Joules, or total energy consumed, since power is a measure of work being done per unit time. Or, more commonly, the output of the meter can be set in units of Watt-Hours.

Mechanical means for measuring Watt-Hours are usually centered around the concept of the motor: build an AC motor that spins at a rate of speed proportional to the instantaneous power in a circuit, then have that motor turn an "odometer" style counting mechanism to keep a running total of energy consumed. The "motor" used in these meters has a rotor made of a thin aluminum disk, with the rotating magnetic field established by sets of coils energized by line voltage and load current so that the rotational speed of the disk is dependent on both voltage and current.

3.2 MAX 232

3.2.1 INTRODUCTION

The MAX232 is an electronic circuit that converts signals from a serial port to signals suitable for usage in e.g. microprocessor circuits.

When communicating with various micro processors one needs to convert the RS232 levels down to lower levels, typically 3.3 or 5.0 Volts. Serial RS-232 communication works with voltages -15V to +15V for high and low. On the other hand, TTL logic operates between 0V and +5V. Modern low power consumption logic operates in the range of 0V and +3.3V or even lower.

Thus the RS-232 signal levels are far too high TTL electronics, and the negative RS-232 voltage for high can't be handled at all by computer logic. To receive serial data from an RS-232 interface the voltage has to be reduced. Also the low and high voltage level has to be inverted. The level converter uses a Max232 and five capacitors.

3.2.2 Pin Diagram

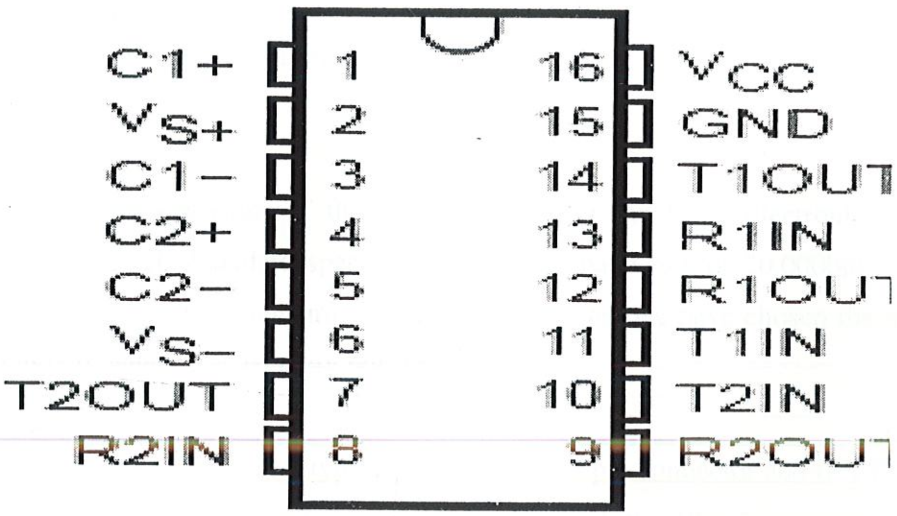

Fig 3.3 MAX 232 pin diagram

3.3 RS-232 standard

3.3.1 Introduction

RS-232 (Recommended Standard - 232) is a telecommunications standard for binary serial communications between devices. It supplies the roadmap for the way devices speak to each other using serial ports. The devices are commonly referred to as a DTE (data terminal equipment) and DCE (data communications equipment); for example, a computer and modem, respectively.

RS232 is the most known serial port used in transmitting the data in communication and interface. Even though serial port is harder to program than the parallel port, this is the most effective method in which the data transmission requires less wires that yields to the less cost. The RS232 is the communication line which enables the data transmission by only using three wire links. The three links provides 'transmit', 'receive' and common ground...

The 'transmit' and 'receive' line on this connecter send and receive data between the computers. As the name indicates, the data is transmitted serially. The two pins are TXD & RXD. There are other lines on this port as RTS, CTS, DSR, DTR, and RTS, RI. The '1' and '0' are the data which defines a voltage level of 3V to 25V and -3V to -25V respectively.

The electrical characteristics of the serial port as per the EIA (Electronics Industry Association) RS232C Standard specifies a maximum baud rate of 20,000bps, which is slow compared to today's standard speed. For this reason, we have chosen the new RS-232D Standard, which was recently released.

The RS-232D has existed in two types. i.e., D-TYPE 25 pin connector and D-TYPE 9 pin connector, which are male connectors on the back of the PC. You need a female connector on your communication from Host to Guest computer. The pin out ofD-9 is shown below.

3.3.2 Rs232 diagram and pin description

D-Type-9 pin no.	D-Type-25 pin no.	Pin outs	Function
3	2	RD	Receive Data (Serial data input)
2	3	TD	Transmit Data (Serial data output)
7	4	RTS	Request to send (acknowledge to modem that UART is re exchange data
8	5	CTS	Clear to send (i.e.; modem is ready to exchange data)
6	6	DSR	Data ready state (UART establishes a link)
5	7	SG	Signal ground
1	8	DCD	Data Carrier detect (This line is active when modem det carrier
4	20	DTR	Data Terminal Ready.
9	22	RI	Ring Indicator (Becomes active when modem detects ringing from PSTN

3.4 LCD (Liquid crystal display)

3.4.1 Introduction

Liquid crystal displays (LCD) are widely used in recent years as compares to LEDs. This is due to the declining prices of LCD, the ability to display numbers, characters and graphics, incorporation of a refreshing controller into the LCD, their by relieving the

CPU of the task of refreshing the LCD and also the ease of programming for characters and graphics. HD 44780 based LCDs are most commonly used.

3.4.2 Pin description

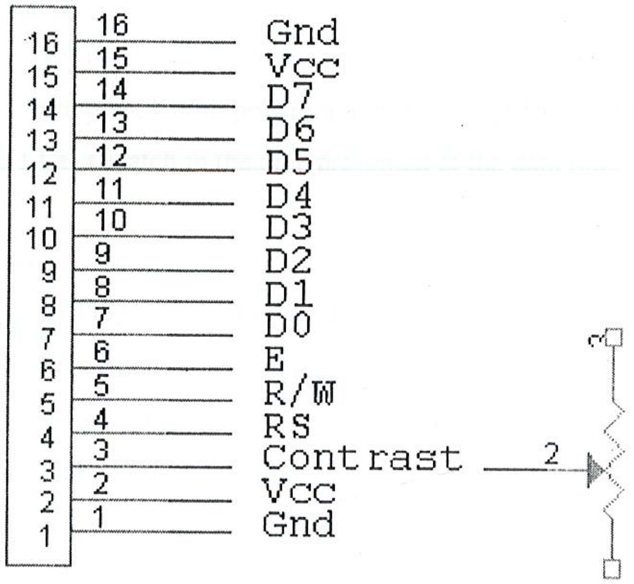

Fig 3.4 Pin diagram of LCD

V_{CC}, V_{SS}, V_{EE}

The voltage V_{CC} and V_{SS} provided by +5V and ground respectively while V_{EE} is used for controlling LCD contrast. Variable voltage between Ground and Vcc is used to specify the contrast (or "darkness") of the characters on the LCD screen.

RS (register select)

There are two important registers inside the LCD. The RS pin is used for their selection as follows. If RS=0, the instruction command code register is selected, then allowing to user to send a command such as clear display, cursor at home etc.. If RS=1, the data register is selected, allowing the user to send data to be displayed on the LCD.

R/W (read/write)

The R/W (read/write) input allowing the user to write information from it. R/W=1, when it read and R/W=0, when it writing.

EN (enable)

The enable pin is used by the LCD to latch information presented to its data pins. When data is supplied to data pins, a high power, a high-to-low pulse must be applied to this pin in order to for the LCD to latch in the data presented at the data pins.

D0-D7 (data lines)

The 8-bit data pins, D0-D7, are used to send information to the LCD or read the contents of the LCD's internal registers. To displays the letters and numbers, we send ASCII codes for the letters A-Z, a-z, and numbers 0-9 to these pins while making RS =1. There are also command codes that can be sent to clear the display or force the cursor to the home position or blink the cursor.

We also use RS =0 to check the busy flag bit to see if the LCD is ready to receive the information. The busy flag is D7 and can be read when R/W =1 and RS =0, as follows: if R/W =1 and RS =0, when D7 =1(busy flag =1), the LCD is busy taking care of internal operations and will not accept any information. When D7 =0, the LCD is ready to receive new information.

3.5 Step down transformer

Step down transformers convert electrical voltage from one level or phase configuration usually down to a lower level. They can include features for electrical isolation, power distribution, and control and instrumentation applications. Step down transformers typically rely on the principle of magnetic induction between coils to convert voltage and/or current levels.

Step down transformers are made from two or more coils of insulated wire wound around a core made of iron. When voltage is applied to one coil (frequently called the primary or input) it magnetizes the iron core, which induces a voltage in the other coil, (frequently called the secondary or output). The turns ratio of the two sets of windings determines the amount of voltage transformation.

An example of this would be: 100 turns on the primary and 50 turns on the secondary, a ratio of 2 to 1.

Step down transformers can be considered nothing more than a voltage ratio device.
With step down transformers the voltage ratio between primary and secondary will mirror the "turn ratio" (except for single phase smaller than 1 KVA which have compensated secondaries). A practical application of this 2 to 1 turns ratio would be a 480 to 240 voltage step down. Note that if the input were 440 volts then the output would be 220 volts. The ratio between input and output voltage will stay constant. Transformers should not be operated at voltages higher than the nameplate rating, but may be operated at lower voltages than rated. Because of this it is possible to do some non-standard applications using standard transformers.

Single phase step down transformers 1 KVA and larger may also be reverse connected to step-down or step-up voltages. (Note: single phase step up or step down transformers sized less than 1 KVA should not be reverse connected because the secondary windings have additional turns to overcome a voltage drop when the load is applied. If reverse connected, the output voltage will be less than desired.)

3.6 7805 voltage regulator

3.6.1 About 7805

These voltage regulator circuits are monolithic integrated circuits designed as fixed voltage regulators for wide variety of applications including local, on card regulation. These regulators employ internal current limiting, thermal shutdown and safe area compensation. With adequate heat sinking these devices can give current output of more than 1.0 A as well. Although designed as fixed voltage source, these can be used with external components to provide variable resistance as well. It gives a constant voltage output of 5V. This is used in the project to provide a supply of 5V to the microcontroller and other devices.

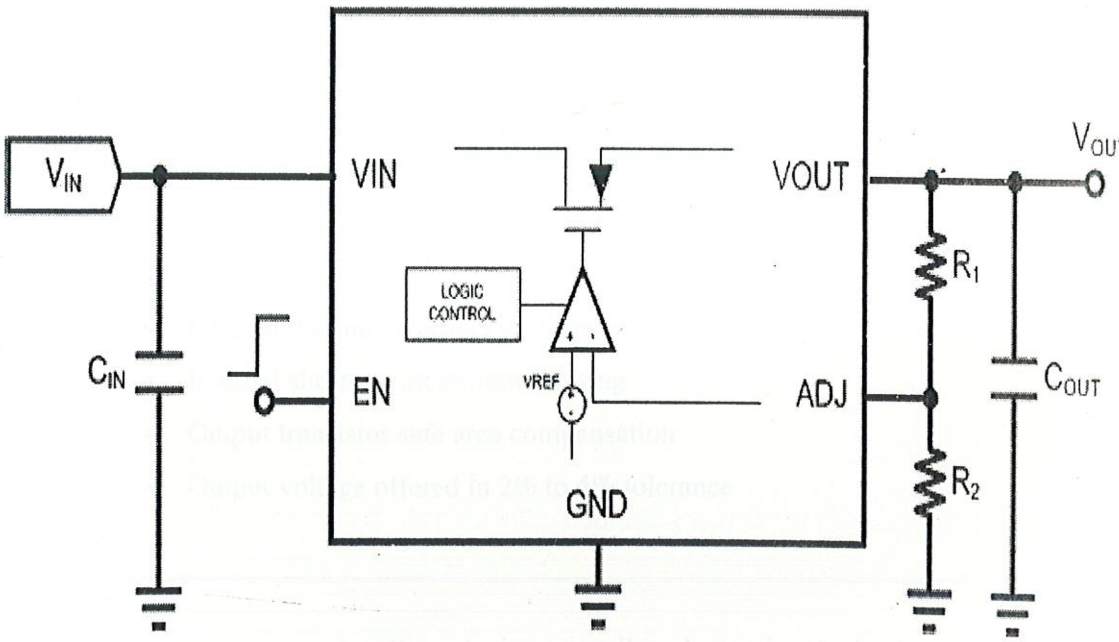

Fig 3.5 Internal Diagram for 7805

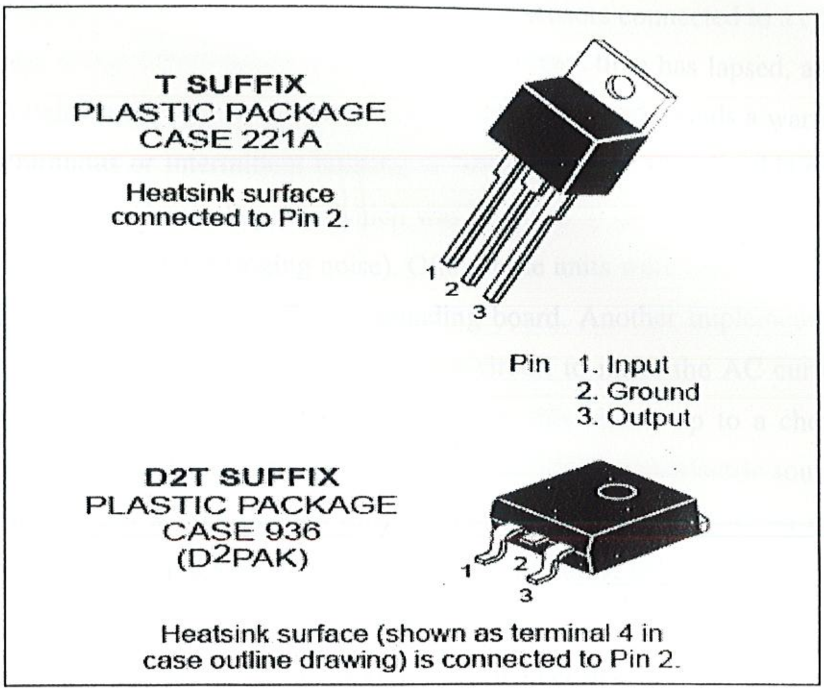

Fig 3.6 Pin Diagram for 7805

3.6.2 Features of 7805

- Output current in excess of 1.0A
- No external components required
- Internal thermal overload protection
- Internal short circuit current limiting
- Output transistor safe area compensation
- Output voltage offered in 2% to 4% tolerance

3.7 Buzzer

A buzzer or beeper is a signaling device, usually electronic, typically used in automobiles, household appliances such as a microwave oven, or game shows.

It most commonly consists of a number of switches or sensors connected to a control unit that determines if and which button was pushed or a preset time has lapsed, and usually illuminates a light on the appropriate button or control panel, and sounds a warning in the form of a continuous or intermittent buzzing or beeping sound. Initially this device was based on an electromechanical system which was identical to an electric bell without the metal gong (which makes the ringing noise). Often these units were anchored to a wall or ceiling and used the ceiling or wall as a sounding board. Another implementation with some AC-connected devices was to implement a circuit to make the AC current into a noise loud enough to drive a loudspeaker and hook this circuit up to a cheap 8-ohm speaker. Nowadays, it is more popular to use a ceramic-based piezoelectric sounder like a Sonalert which makes a high-pitched tone. Usually these were hooked up to "driver" circuits which varied the pitch of the sound or pulsed the sound on and off.

In game shows it is also known as a "lockout system," because when one person signals ("buzzes in"), all others are locked out from signaling. Several game shows have large buzzer buttons which are identified as "plungers".

The word "buzzer" comes from the rasping noise that buzzers made when they were electromechanical devices, operated from stepped-down AC line voltage at 50 or 60 cycles. Other sounds commonly used to indicate that a button has been pressed are a ring or a beep.

3.8 GSM module

It is the device which is interfaced with the microcontroller and is able to send messages to a predefined number. This is used in our project to send a message if the alcohol content is found in the breath of the driver.

3.9 Opto-coupler

An opto-isolator contains an infrared LED and silicon photodiode with an integrated amplifier stage.

In electronics, an opto-isolator (or optical isolator, optocoupler, photocoupler, or photo MOS) is a device that uses a short optical transmission path to transfer a signal between elements of a circuit, typically a transmitter and a receiver, while keeping them electrically isolated — since the signal goes from an electrical signal to an optical signal back to an electrical signal, electrical contact along the path is broken

A common implementation involves a LED and a phototransistor, separated so that light may travel across a barrier but electrical current may not. When an electrical signal is applied to the input of the opto-isolator, its LED lights, its light sensor then activates, and a corresponding electrical signal is generated at the output. Unlike a transformer, the opto-isolator allows for DC coupling and generally provides significant protection from serious overvoltage conditions in one circuit affecting the other.

With a photodiode as the detector, the output current is proportional to the amount of incident light supplied by the emitter. The diode can be used in a photovoltaic mode or a photoconductive mode.

In photovoltaic mode, the diode acts like a current source in parallel with a forward-biased diode. The output current and voltage are dependent on the load impedance and light intensity.

In photoconductive mode, the diode is connected to a supply voltage, and the magnitude of the current conducted is directly proportional to the intensity of light.

An opto-isolator can also be constructed using a small incandescent lamp in place of the LED; such a device, because the lamp has a much slower response time than an LED, will filter out noise or half-wave power in the input signal. In so doing, it will also filter out any audio- or higher-frequency signals in the input. It has the further disadvantage, of course, (an overwhelming disadvantage in most applications) that incandescent lamps have finite life spans. Thus, such an unconventional device is of extremely limited usefulness, suitable only for applications such as science projects.

The optical path may be air or a dielectric waveguide. The transmitting and receiving elements of an optical isolator may be contained within a single compact module, for mounting, for example, on a circuit board; in this case, the module is often called an optoisolator or opto-isolator. The photosensor may be a photocell, phototransistor, or an optically triggered SCR or Triac. Occasionally, this device will in turn operate a power relay or contactor.

Among other applications, opto-isolators can help cut down on ground loops and block voltage spikes.

One of the requirements of the MIDI (Musical Instrument Digital Interface) standard is that input connections be opto-isolated.

They are used to isolate low-current control or signal circuitry from transients generated or transmitted by power supply and high-current control circuits. The latter are used within motor and machine control function blocks.

3.9 Other Components

3.9.1 Resistor

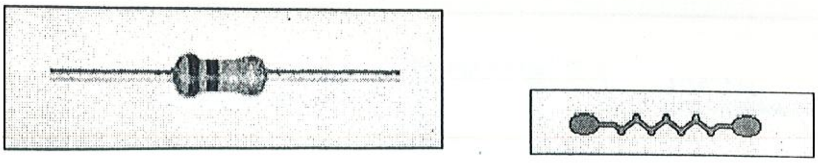

Fig 3.9 Resistor

Resistor is a component of an electric circuit that resists the flow of direct or alternating electric current. Resistors can limit or divide the current, reduce the voltage, protect an electric circuit, or provide large amounts of heat or light.

An electric current is the movement of charged particles called electrons from one region to another. The amount of resistance to the flow of current that a resistor causes depends on the material it is made of as well as its size and shape. Resistors are usually placed in electric circuits, which are devices formed when current moves through an electrical conductor.

When a voltage, or electric potential, is applied to opposite ends of a circuit, it causes current to flow through the circuit. As the current flows, it encounters a certain amount of resistance from the conductor and any resistors in the circuit. Each material has a characteristic resistance. For example, wood is a bad conductor because it offers high resistance to the current; copper is a better conductor because it offers less resistance. In any electric circuit, the current in the entire circuit is equal to the voltage across that circuit divided by the resistance of the circuit. Resistors are often made to have a specific value of resistance so that the characteristics of the circuit can be accurately calculated.

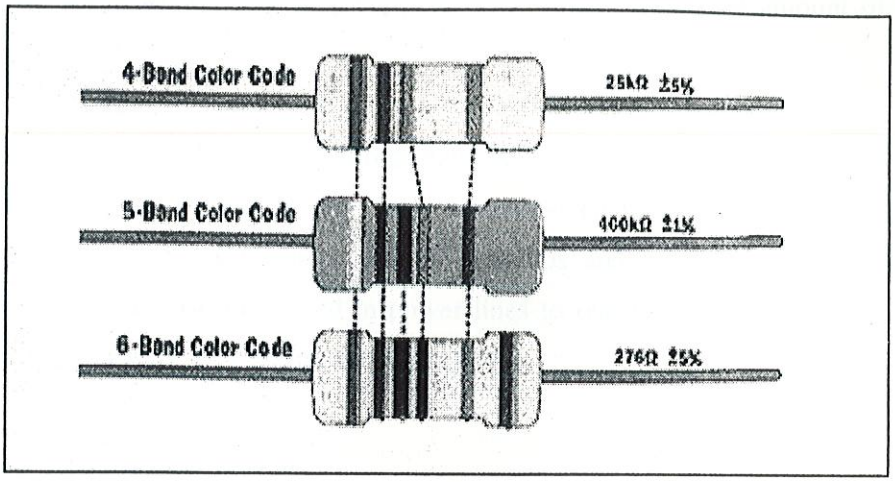

Fig 3.10 Resistor

Resistors are designed to have a specific value of resistance. Most resistors used in electric circuits are cylindrical items a few millimeters long with wires at both ends to connect them to the circuit. Resistors are often color coded by three or four color bands that indicate the specific value of resistance. Some resistors obey Ohm's law, which states that the current density is directly proportional to the electrical field when the temperature is constant. The resistance of a material that follows Ohm's law is constant, or independent of voltage or current, and the relationship between current and voltage is linear.

3.9.2 Capacitor

Capacitor, or electrical condenser, is a device used for storing an electrical charge. In its simplest form a capacitor consists of two metal plates separated by a non-conducting layer called the dielectric. When one plate is charged with electricity from a direct-current or electrostatic source, the other plate will have induced in it a charge of the opposite sign; that is, positive if the original charge is negative and negative if the charge is positive. The electrical size of a capacitor is its capacitance, the amount of electric charge it can hold.

Capacitors are useful when direct current must be prevented from entering some part of an electric circuit. Fixed-capacity and variable-capacity capacitors are used in conjunction with coils as resonant circuits in radios and other electronic equipment. Large capacitors are also employed in power lines to resonate the load on the line and make it possible for the line to transmit more power.

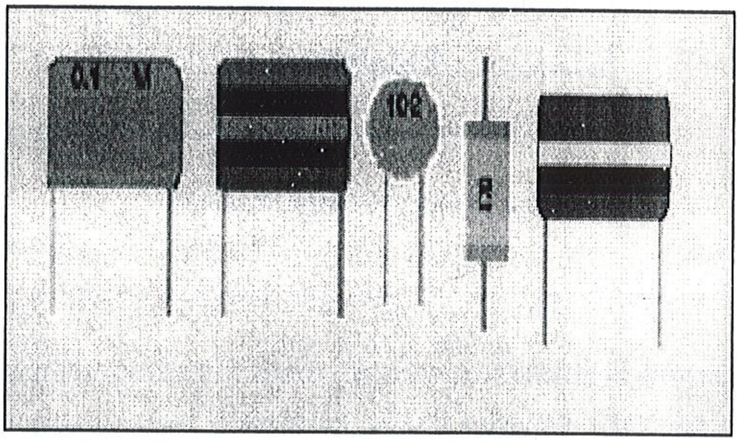

Fig 3.11 Capacitor

Capacitors are produced in a wide variety of forms. Air, mica, ceramics, paper, oil, and vacuums are used as dielectrics, depending on the purpose for which the device is intended.

3.9.3 Semiconductor Diodes

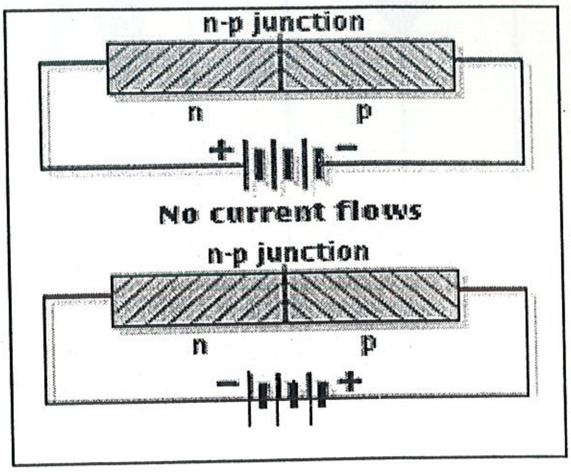

Fig 3.13 Circuit diagram of a diode

Diode is an electronic device that allows the passage of current in only one direction. The diodes, most commonly used in electronic circuits today are semiconductor diodes. The simplest of these, the germanium point-contact diode, dates from the early days of radio, when the received radio signal was detected by means of a germanium crystal and a fine, pointed wire that rested on it. In modern germanium (or silicon) point-contact diodes, the wire and a tiny crystal plate are mounted inside a small glass tube and connected to two wires that are fused into the ends of the tube.

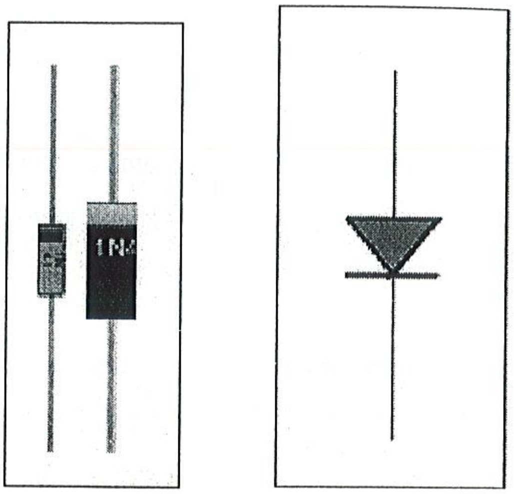

Fig 3.14 Diodes

3.9.4 Transistor

A **bipolar (junction) transistor (BJT)** is a three-terminal electronic device constructed of doped semiconductor material and may be used in amplifying or switching applications. *Bipolar* transistors are so named because their operation involves both electrons and holes. Charge flow in a BJT is due to bidirectional diffusion of charge carriers across a junction between two regions of different charge concentrations. This mode of operation is contrasted with *unipolar transistors,* such as field-effect transistors, in which only one carrier type is involved in charge flow due to drift. By design, most of the BJT collector current is due to the flow of charges injected from a high-concentration emitter into the base where they are minority carriers that diffuse toward the collector, and so BJTs are classified as *minority-carrier* devices.

NPN

NPN is one of the two types of bipolar transistors, in which the letters "N" and "P" refer to the majority charge carriers inside the different regions of the transistor. Most bipolar transistors used today are NPN, because electron mobility is higher than hole mobility in semiconductors, allowing greater currents and faster operation.

NPN transistors consist of a layer of P-doped semiconductor (the "base") between two N-doped layers. A small current entering the base in common-emitter mode is amplified in the collector output. In other terms, an NPN transistor is "on" when its base is pulled **high** relative to the emitter.

The arrow in the NPN transistor symbol is on the emitter leg and points in the direction of the conventional current flow when the device is in forward active mode.

PNP

The other type of BJT is the PNP with the letters "P" and "N" referring to the majority charge carriers inside the different regions of the transistor.

PNP transistors consist of a layer of N-doped semiconductor between two layers of P-doped material. A small current leaving the base in common-emitter mode is amplified in the collector output. In other terms, a PNP transistor is "on" when its base is pulled **low** relative to the emitter.

The arrow in the PNP transistor symbol is on the emitter leg and points in the direction of the conventional current flow when the device is in forward active mode.

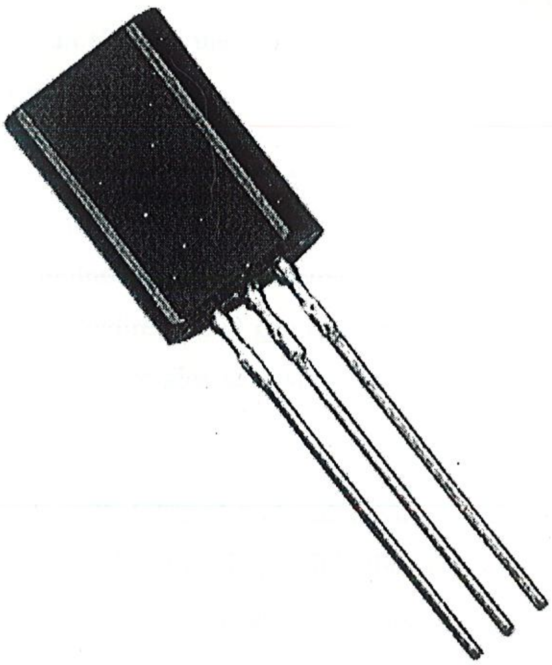

Fig 3.15 BJT

3.9.5 LED

A **light-emitting diode** (**LED**) is a semiconductor diode that emits incoherent narrow-spectrum light when electrically biased in the forward direction of the p-n junction. This effect is a form of electroluminescence.

An LED is usually a small area source, often with extra optics added to the chip that shapes its radiation pattern. The color of the emitted light depends on the composition and condition of the semiconducting material used, and can be infrared, visible, or near-ultraviolet. An LED can be used as a regular household light source.

Like a normal diode, an LED consists of a chip of semiconducting material impregnated, or *doped*, with impurities to create a *p-n junction*. As in other diodes, current flows easily from the p-side, or anode, to the n-side, or cathode, but not in the reverse direction.

Charge-carriers—electrons and holes—flow into the junction from electrodes with different voltages. When an electron meets a hole, it falls into a lower energy level, and releases energy in the form of a photon.

The wavelength of the light emitted, and therefore its color, depends on the band gap energy of the materials forming the *p-n junction*. In silicon or germanium diodes, the electrons and holes recombine by a *non-radiative transition* which produces no optical emission, because these are indirect band gap materials. The materials used for an LED have a direct band gap with energies corresponding to near-infrared, visible or near-ultraviolet light.

LED development began with infrared and red devices made with gallium arsenide. Advances in materials science have made possible the production of devices with ever-shorter wavelengths, producing light in a variety of colors.

LEDs are usually built on an n-type substrate, with an electrode attached to the p-type layer deposited on its surface. P-type substrates, while less common, occur as well. Many commercial LEDs, especially GaN/InGaN, also use sapphire substrate. Substrates that are transparent to the emitted wavelength, and backed by a reflective layer, increase the LED efficiency. The refractive index of the package material should match the index of the semiconductor, otherwise the produced light gets partially reflected back into the semiconductor, where it may be absorbed and turned into additional heat, thus lowering the efficiency. This type of reflection also occurs at the surface of the package if the LED is coupled to a medium with a different refractive index such as a glass fiber or air. The refractive index of most LED semiconductors is quite high, so in almost all cases the LED is coupled into a much lower-index medium. The large index difference makes the reflection quite substantial (per the Fresnel coefficients), and this is usually one of the dominant causes of LED inefficiency. Often more than half of the emitted light is reflected back at the LED-package and package-air interfaces. The reflection is most commonly reduced by using a dome-shaped (half-sphere) package with the diode in the center so that the outgoing light rays strike the surface perpendicularly, at which angle the reflection is minimized. An anti-reflection coating may be added as well. The package may be cheap plastic, which may be colored, but this is only for cosmetic reasons or to improve the contrast ratio; the color of the packaging does not substantially affect the color of the light emitted. Other strategies for reducing the impact of the interface reflections include designing the LED to reabsorb and reemit the reflected light (called *photon recycling*) and manipulating the microscopic structure of the surface to reduce the reflectance, either by introducing random roughness or by creating programmed *moth eye* surface patterns.

Conventional LEDs are made from a variety of inorganic semiconductor materials, producing different colors.

Chapter 4

MICROCONTROLLER

4.1 What is a microcontroller?

A microcontroller is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. Program memory in the form of NOR flash or OTP ROM is also often included on chip, as well as a typically small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications.

Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, and toys. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes. Mixed signal microcontrollers are common, integrating analog components needed to control non-digital electronic systems.

Some microcontrollers may use four-bit words and operate at clock rate frequencies as low as 4 kHz, for low power consumption (milliwatts or microwatts). They will generally have the ability to retain functionality while waiting for an event such as a button press or other interrupt; power consumption while sleeping (CPU clock and most peripherals off) may be just nanowatts, making many of them well suited for long lasting battery applications. Other microcontrollers may serve performance-critical roles, where they may need to act more like a digital signal processor (DSP), with higher clock speeds and power consumption.

Microcontrollers must provide real time (predictable, though not necessarily fast) response to events in the embedded system they are controlling. When certain events occur, an interrupt system can signal the processor to suspend processing the current instruction sequence and to begin an interrupt service routine (ISR, or "interrupt handler"). The ISR will perform any processing required based on the source of the interrupt before returning to the original instruction sequence. Possible interrupt sources are device dependent, and often include events such as an internal timer overflow, completing an analog to digital conversion, a logic level change on an input such as from a button being pressed, and data received on a communication link. Where power consumption is important as in battery operated devices, interrupts may also wake a microcontroller from a low power sleep state where the processor is halted until required to do something by a peripheral event.

4.2 Microcontroller programming

Microcontroller programs must fit in the available on-chip program memory, since it would be costly to provide a system with external, expandable, memory. Compilers and assemblers are used to turn high-level language and assembler language codes into a compact machine code for storage in the microcontroller's memory. Depending on the device, the program memory may be permanent, read-only memory that can only be programmed at the factory, or program memory may be field-alterable flash or erasable read-only memory.

4.3 Other microcontroller features

Microcontrollers usually contain from several to dozens of general purpose input/output pins (GPIO). GPIO pins are software configurable to either an input or an output state. When GPIO pins are configured to an input state, they are often used to read sensors or external signals. Configured to the output state, GPIO pins can drive external devices such as LED's or motors.

Many embedded systems need to read sensors that produce analog signals. This is the purpose of the analog-to-digital converter (ADC). Since processors are built to interpret and process digital data, i.e. 1s and 0s, they won't be able to do anything with the analog signals that may be sent to it by a device. So the analog to digital converter is used to convert the incoming data into a form that the processor can recognize. A less common feature on some microcontrollers is a digital-to-analog converter (DAC) that allows the processor to output analog signals or voltage levels.

In addition to the converters, many embedded microprocessors include a variety of timers as well. One of the most common types of timers is the Programmable Interval Timer (PIT). A PIT just counts down from some value to zero. Once it reaches zero, it sends an interrupt to the processor indicating that it has finished counting. This is useful for devices such as thermostats, which periodically test the temperature around them to see if they need to turn the air conditioner on, the heater on, etc.

Time Processing Unit (TPU) is a sophisticated timer. In addition to counting down, the TPU can detect input events, generate output events, and perform other useful operations. A dedicated Pulse Width Modulation (PWM) block makes it possible for the CPU to control power converters, resistive loads, motors, etc., without using lots of CPU resources in tight timer loops.

Universal Asynchronous Receiver/Transmitter (UART) block makes it possible to receive and transmit data over a serial line with very little load on the CPU. Dedicated on-chip hardware also often includes capabilities to communicate with other devices (chips) in digital formats such as I2C and Serial Peripheral Interface (SPI).

4.4 Difference between a microcontroller and a microprocessor

The microprocessors (such as 8086,80286,68000 etc.) contain no RAM, no ROM and no I/O ports on the chip itself. For this reason they are referred as general- purpose microprocessors. A system designer using general- purpose microprocessor must add external RAM, ROM, I/O ports and timers to make them functional. Although the addition of external RAM, ROM, and I/O ports make the system bulkier and much more expensive, they have the advantage of versatility such that the designer can decide on the amount of RAM, ROM and I/o ports needed to fit the task at hand. This is the not the case with microcontrollers. A microcontroller has a CPU (a microprocessor) in addition to the fixed amount of RAM, ROM, I/O ports, and timer are all embedded together on the chip: therefore, the designer cannot add any external memory, I/O, or timer to it. The fixed amount of on chip RAM, ROM, and number of I/O ports in microcontrollers make them ideal for many applications in which cost and space are critical. In many applications, for example a TV remote control, there is no need for the computing power of a 486 or even a 8086 microprocessor. In many applications, the space it takes, the power it consumes, and the price per unit are much more critical considerations than the computing power. These applications most often require some I/O operations to read signals and turn on and off certain bits. It is interesting to know that some microcontrollers manufactures have gone as far as integrating an ADC and other peripherals into the microcontrollers.

A microcontroller is a specialized form of microprocessor that is designed to be self-sufficient and cost-effective, where a microprocessor is typically designed to be general purpose (the kind used in a PC). Microcontrollers are frequently found in many different appliances.

The microcontroller is the integration of a number of useful functions into a single IC package. These functions are:

1. The ability to execute a stored set of instructions to carry out user defined tasks.
2. The ability to be able to access external memory chips to both read and write data from and to the memory.

Basically, a microcontroller is a device which integrates a number of the components of a microprocessor system onto a single microchip.

The difference is that microcontroller incorporates features of microprocessor (CPU, ALU, Registers) along with the presence of added features like presence of RAM, ROM, I\O ports, counter etc. Here microcontroller control the operation of machine using fixed program stored in Rom that doesn't change with lifetime.

4.5 MICROCONTROLLERS FOR EMBEDDED SYSTEMS

In the Literature discussing microprocessors, we often see the term Embedded System. Microprocessors and Microcontrollers are widely used in embedded system products. An embedded system product uses a microprocessor (or Microcontroller) to do one task only. A printer is an example of embedded system since the processor inside it performs one task only; namely getting the data and printing it. Contrast this with a Pentium based PC. A PC can be used for any number of applications such as word processor, print-server, bank teller terminal, Video game, network server, or Internet terminal. Software for a variety of applications can be loaded and run. Of course the reason a pc can perform myriad tasks is that it has RAM memory and an operating system that loads the application software into RAM memory and lets the CPU run it.

In an Embedded system, there is only one application software that is typically burned into ROM. An x86 PC contains or is connected to various embedded products such as keyboard, printer, modem, disk controller, sound card, CD-ROM drives, mouse, and so on. Each one of these peripherals has a Microcontroller inside it that performs only one task. For example, inside every mouse there is a Microcontroller to perform the task of finding the mouse position and sending it to the PC.

4.6 AT89C52

4.6.1 Introduction

The AT89C52 is a low-power, high-performance CMOS 8-bit microcomputer with 8K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high density nonvolatile memory technology and is compatible with the industry standard 80C51 and 80C52 instruction set and pin out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C52 is a powerful microcomputer which provides a highly flexible and cost effective solution to many embedded control applications.

4.6.2 Specifications and features

- 4K on-chip ROM
- 256 bytes internal RAM (8-bit)
- 32 I/O pins
- Three 16-bit timers
- Serial programming facility
- 40 pin Dual-in-line Package

4.6.3 Difference between 8051 and 8052

The 8052 microcontroller is the 8051's "big brother." It is a slightly more powerful microcontroller, sporting a number of additional features which the developer may make use of:

- 256 bytes of Internal RAM (compared to 128 in the standard 8051).

- A third 16-bit timer, capable of a number of new operation modes and 16-bit reloads.
- Additional SFRs to support the functionality offered by the third timer.

That's really about all there is to the difference between the 8051 and 8052

4.6.4 Pin Diagram

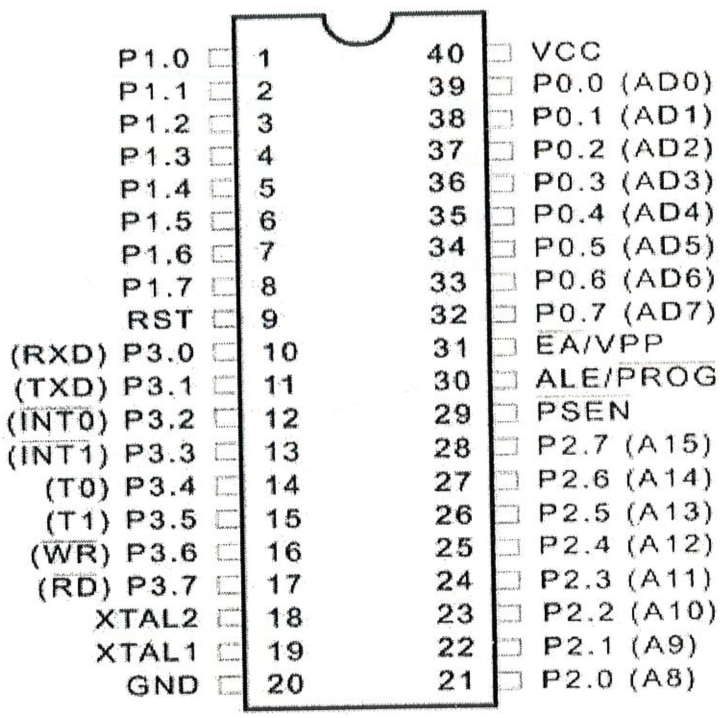

Fig 4.1 Microcontroller pin diagram

ALE/PROG

Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. ALE is emitted at a constant rate of 1/6 of the oscillator frequency, for external timing or clocking purposes, even when there are no accesses to external memory. (However, one ALE pulse is skipped during each access to external Data Memory.) This pin is also the program pulse input (PROG) during EPROM programming.

PSEN

Program Store Enable is the read strobe to external Program Memory. When the device is executing out of external Program Memory, PSEN is activated twice each machine cycle (except that two PSEN activations are skipped during accesses to external Data Memory). PSEN is not activated when the device is executing out of internal Program Memory.

EA/VPP

When EA is held high the CPU executes out of internal Program Memory (unless the Program Counter exceeds 0FFFH in the 80C51). Holding EA low, forces the CPU to execute out of external memory regardless of the Program Counter value. In the 80C31, EA must be externally wired low. In the EPROM devices, this pin also receives the programming supply voltage (VPP) during EPROM programming.

XTAL1

Input to the inverting oscillator amplifier.

XTAL2

Output from the inverting oscillator amplifier.

Port 0

Port 0 is an 8-bit open drain bidirectional port. As an open drain output port, it can sink eight LS TTL loads. Port 0 pins that have 1s written to them float, and in that state will function as high impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external memory. In this application it uses strong internal pull-ups when emitting 1s. Port 0 emits code bytes during program verification. In this application, external pull-ups are required.

Port 1

Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups, and in that state can be used as inputs. As inputs, port 1 pins that are externally being pulled low will source current because of the internal pull-ups.

Port 2

Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 emits the high-order address byte during accesses to external memory that use 16-bit addresses. In this application, it uses the strong internal pull-ups when emitting 1s.

Port 3

Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. It also serves the functions of various special features of the 80C51 Family as follows:

Port Pin Alternate Function

P3.0 - RxD (serial input port)

P3.1 - TxD (serial output port)

P3.2 - INT0 (external interrupt 0)

P3.3 - INT1 (external interrupt 1)

P3.4 -T0 (timer 0 external input)

P3.5 - T1 (timer 1 external input)

P3.6 -WR (external data memory write strobe)

P3.7 -RD (external data memory read strobe)

VCC

Supply voltage

VSS

Circuit ground potential.

All four ports in the 80C51 are bidirectional. Each consists of a latch (Special Function Registers P0 through P3), an output driver, and an input buffer. The output drivers of Ports 0 and 2, and the input buffers of Port 0, are used in accesses to external memory. In this application, Port 0 outputs the low byte of the external memory address, time-multiplexed with the byte being written or read. Port 2 outputs the high byte of the external memory address when the address is 16 bits wide. Otherwise, the Port 2 pins continue to emit the P2 SFR content.

All the Port 3 pins are multifunctional. They are not only port pins, but also serve the functions of various special features as listed below:

Port Pin Alternate Function
P3.0 RxD (serial input port)
P3.1 TxD (serial output port)
P3.2 INT0 (external interrupt)
P3.3 INT1 (external interrupt)
P3.4 T0 (Timer/Counter 0 external input)
P3.5 T1 (Timer/Counter 1 external input)
P3.6 WR (external Data Memory write strobe)
P3.7 RD (external Data Memory read strobe)

Chapter 5

Source code

5.1 Software requirement

The basic software requirements of the project are as follows

1. KEIL: the software platform used for the programming of the microcontroller is KEIL. Using this software, assembly language programming can be actualized.
2. PRELOAD: this is used for IC burning, i.e. writing something on the chip.

5.2 Program to operate the microcontroller

```
#define dev P20
#define buzz P27
#define number2 "9816430508"
#define number1 "9816342668"
#include<lcdrout.h>
#include<atcommands.h>
#include<serial.h>
char count;
void countpulse()interrupt 0
{
count++; ms_delay(200);
}
unsigned char units;
unsigned int bill;

//#include<atcommands.h>
//unsigned                    char                    idata                    msg[210]={
32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32
,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,3
2,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,
32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32
,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,3
2,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32};

unsigned                    char                    idata                    msg[150]={
32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32
,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,3
```

```
2,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,
32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32
,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,3
2,32,32,32,32,32,32};

//unsigned char idata msg[20];
unsigned char index=0;
unsigned char st[12];
//unsigned char *ms="Bill:";// message to send
bit flag=0,save_mem_flag=0;
void v2c(unsigned int val)
{

 unsigned int val2;
 clrscr();
 val2=val%100;
 val=val/100;
 st[0]='B';
 st[1]='i';
 st[2]='l';
 st[3]='l';
 st[4]=':';
 st[5]=(val/100)+48;
 st[6]=((val/10)%10)+48;
 st[7]=(val%10)+48;
 st[8]='.';
 st[9]=(val2/10)+48;
 st[10]=(val2%10)+48;
 st[11]='\0';
 clrscr();
 lcd_cmd1(0x80);
 lcd_puts(st);
 secdelay(3);
 }
 at_send(char *s,unsigned char loc)
 {
 index=0;
  IE=0;
  while(*s != '\0')
   {
    send(*s);
  ms_delay(80);
          s++;
   }
         if(loc==0)
         {
```

```
                send(13);
                goto out1;
        }
        if (loc==240)
                goto out;
        else
        {
            if(loc<10)
                send(loc+48);
                else
                {
                 send(loc/10 +48);
                 send(loc%10 +48);
                }
        send(13);
        }
out1:IE=0x90;
out:ms_delay(100);
}
///*
at_init2()
{
  at_send("AT+CPMS=\"ME\"",0);
  ms_delay(250);ms_delay(250);
  at_send("AT+CNMI=2,1,0,0,0",0)
 ms_delay(250);ms_delay(250);
  at_send("AT",0);     ms_delay(250);
  at_send("AT",0);     ms_delay(250);
  at_send("AT+CMGF=1",0);ms_delay(250);
  at_send("AT+CMGS=",240);ms_delay(250);
}
// */
sms2(unsigned char *msg,unsigned char *num)
{         gsm_check();
          EA=0;
          clrscr();
          lcd_puts("Sending SMS");
          at_send("AT",0);
          at_send("AT",0);
          at_send("AT+CMGF=1",0);  ms_delay(200);ms_delay(200);
          at_send("AT+CMGS=",240);ms_delay(200);
          send("");
          put(num);
          send("");
          send(13);
          ms_delay(200);
```

```
          put(msg);
          send(0x1A);
          secdelay(3);
//        gsm_check();
   clrscr();
          EA=1;
}
gsm_check()   //working Very accurately
{
char l=0,eflag1=0,eflag2=0;
          clrscr();
          IE=0x90;
          lcd_puts("Testing Module");
back1: serial_init(9600);       IE=0x90;
          index=0;
          ms_delay(100);
          l=0;
          clrscr();
          lcd_puts("Test 1");
          while( flag==0)
            {
                  at_send("AT",0);
                  ms_delay(200);
                  }
          ms_delay(200);
   lcd_cmd1(0xc0);
   lcd_data(msg[5]);
   lcd_data(msg[6]);
          ms_delay(200);
   flag=0;
          if((msg[3]=='O'  &&  msg[4]=='K'  )  ||(msg[4]=='O'  &&  msg[5]=='K'  )
||(msg[4]=='O' && msg[5]=='K' ) || (msg[5]=='O' && msg[6]=='K' ) || (msg[6]=='O' &&
msg[7]=='K' )||(msg[7]=='O' && msg[8]=='K' ))
                  {
                  clrscr();
                  lcd_puts("Firtst Test Comp");
                  ms_delay(100);
                  }
                  else
                  {
                  goto back1;
                  }
                  l=0;
          while(l<index)
            {
             msg[l]=32;
```

```
                l++;
        }
back2:
        index=0;
        l=0;
        clrscr();
        lcd_puts("TEST 2");
        ms_delay(200);
        while( flag==0)
        {
                at_send("AT+CMGF=1",0);
                secdelay(1);
                at_send("AT",0);
                secdelay(1);
                at_send("AT+CMGF=1",0);
                secdelay(1);
        }
        lcd_cmd1(0xc0);
   lcd_data(msg[0]);
   lcd_data(msg[1]);
        lcd_data(msg[2]);
   lcd_data(msg[3]);
        lcd_data(msg[4]);
   lcd_data(msg[5]);
        lcd_data(msg[6]);
   lcd_data(msg[7]);
        ms_delay(200);ms_delay(200);ms_delay(200);ms_delay(200);
        flag=0;
        if((msg[4]=='O' && msg[5]=='K' ) || (msg[5]=='O' && msg[6]=='K' ) ||
(msg[6]=='O' && msg[7]=='K' ))
                {
                 clrscr();
                 lcd_puts("IInd Test Comp");
                 ms_delay(200);
                }
                else
                {
                l++;
                        if(l>4)
                        goto back1;
                goto back2;
                }
                l=0;
        while(l<index)
          {
         msg[l] =32;
```

```
        l++;
    }
   flag=0;
IE=0x90;
 }
 delete_all()
   {
   char k=1;
   clrscr();
   EA=0;
   lcd_puts("Deleteing All");
         while(k<11)
         {
                 lcd_cmd1(0xc0);
                 lcd_data(k+48);
                 at_send("AT",0);
                 if(k<10)
                 at_send("AT+CMGD=",k);
                 else
                 {
                 at_send("AT+CMGD=",(k/10));
                 at_send("AT+CMGD=",(k%10));
                 }
                 k++;
          ms_delay(100);
         }
         EA=1;
   }

read_sms(unsigned char loc)
{
unsigned char temp=0;
unsigned char charloc=0x80,sms_data=0,sms_number=0,sms_date=0,sms_time=0;
                 //gsm_check();
                 index=0;
                 flag=0;
                 clrscr();
                 lcd_puts("Reading Message");
                 secdelay(1);
                 save_mem_flag=1;
         at_send("at+cmgr=",loc);
                 secdelay(3);
                 lcd_cmd1(0xc0);
                 displaypval(index);
                 secdelay(1);
                 IE=0;
```

```
            clrscr();
            while(temp<index)
             {
             /****************************************
              For Getting the message's Start Location
              ****************************************/
                    if(msg[temp]==34)
                     if(msg[temp+1]==13)
                           if(msg[temp+2]==10)
                                  sms_data=temp+3;

            /****************************************
              For Getting the Mobile Number's Start Location
              ****************************************/

                    if(msg[temp]==34)//
                           if(msg[temp+1]==44)//
                                  if(msg[temp+2]==34)//
                                         sms_number=temp+3;
            /****************************************
              For Get   }

read_sms(unsigned char loc)
{
unsigned char temp=0;
unsigned char charloc=0x80,sms_data=0,sms_number=0,sms_date=0,sms_time=0;
            //gsm_check();
            index=0;
            flag=0;
            clrscr();
            lcd_puts("Reading Message");
            secdelay(1);
            save_mem_flag=1;
       at_send("at+cmgr=",loc);
            secdelay(3);
            lcd_cmd1(0xc0);
            displaypval(index);
            secdelay(1);
            IE=0;
            clrscr();
            while(temp<index)
             {
             /****************************************
              For Getting the message's Start Location
              ****************************************/
                    if(msg[temp]==34)
```

```
        if(msg[temp+1]==13)
                if(msg[temp+2]==10)
                        sms_data=temp+3;

/****************************************
 For Getting the Mobile Number's Start Location
 ****************************************/

        if(msg[temp]==34)//
                if(msg[temp+1]==44)//
                ting the SMS REC DATE's Start Location
 ****************************************/

        if(msg[temp]==44)//
                if(msg[temp+1]==44)//
                        if(msg[temp+2]==34)//
                                sms_date=temp+3;
  temp++;
 }
 /****************************************
  For Getting the SMS REC TIME's Start Location
  ****************************************/
sms_time=sms_date+9;
/****************************************
 For Displaying SMS Sender Number Or Identity
****************************************/
 temp=sms_number;
 charloc=0xc0;
 clrscr();
// lcd_puts("Sender:");
 while(temp<index)
 {
        if(msg[temp]==34)
        break;
  lcd_cmd1(charloc);
  lcd_data(msg[temp]);
  temp++;
  charloc++;
  if(charloc>0xcf)
   {
   secdelay(2);
        charloc=0xc0;
        lcd_cmd1(charloc);
        lcd_puts("                ");
   }
  }
```

```
// secdelay(3);

/********************************
End Of Showing SMS Senders Identity DATA
********************************/
/****************************************
For Displaying SMS DATA
****************************************/
temp=sms_data;
charloc=0x80;
clrscr();
lcd_puts("SMS DATA:");
lcd_data(msg[temp]);
lcd_data(msg[temp+1]);
secdelay
 if(msg[temp]=='A'&& msg[temp+1]=='1')
        {
        clrscr();
  lcd_puts("BILL REQUEST");
        secdelay(2);
        v2c((unsigned int)bill);
        clrscr();
        lcd_puts(":-");
        lcd_puts(st);
        sms(st,number1);
  sms(st,number2);
  secdelay(3);
        }
        if(msg[temp]=='A'&& msg[temp+1]=='A')
        {
        clrscr();
  lcd_puts("BILL CLEARED");
        secdelay(2);
        bill=0;
        units=0;
        v2c(0);
        clrscr();
        lcd_puts(":-");
        lcd_puts(st);
        sms(st,number1);
  sms(st,number2);
  secdelay(3);
        }
 else
 {
        clrscr();
```

```
  lcd_puts("Not Valid");
  secdelay(1);
 }

while(temp<index)
{
        if(msg[temp]==13)
        break;
 lcd_cmd1(charloc);
 lcd_data(msg[temp]);
 lcd_cmd1(0xc0);
 temp++;
 charloc++;
 if(charloc>0x8f)
  {
  secdelay(2);
        charloc=0x80;
        lcd_cmd1(charloc);
        lcd_puts("                ");
  }
 }
secdelay(3);


/********************************
End Of Showing SMS DATA
********************************/
 /***************************************
For Displaying SMS REC DATE
***************************************/
/*
temp=sms_date;
 charloc=0xc0;
 clrscr();
 lcd_puts("DATE:");
 while(temp<index)
{
   if(msg[temp]==44)
   break;
   lcd_cmd1(charloc);
  lcd_data(msg[temp]);
  temp++;
  charloc++;
  if(charloc>0xcf)
   {
         secdelay(2);
```

```
            charloc=0xc0;
            lcd_cmd1(charloc);
            lcd_puts("                ");
        }
}
    secdelay(3);
 */
    //********************************
    //End Of Showing SMS REC DATE DATA
    //********************************
    ///*****************************************
    //For Displaying SMS RECEPTION TIME
    //*****************************************/
    /*
    temp=sms_time;
    charloc=0xc0;
    clrscr();
    lcd_puts("TIME:");
    while(temp<index)
    {
            if(msg[temp]==43)
            break;
             lcd_cmd1(charloc);
             lcd_data(msg[temp]);
            temp++;
           charloc++;
           if(charloc>0xcf)
     {
          secdelay(2);
            charloc=0xc0;
            lcd_cmd1(charloc);
            lcd_puts("                ");
     }
    }
            secdelay(3);
 */
    //********************************
    // End Of Showing SMS RECEPTION TIME DATA
    //********************************
     //********************************
    //For CLEARING MSG RRAY FOR NEW DATA
    //********************************
    temp=0;
    while(temp<index)
    {
    msg[temp]=32;
```

```
temp++;
}
///*********************************
// End Of CLEARING LOOP
//*******************************
 temp=0;
index=0;
gsm_check();
clrscr();
flag=0;// For preventing in recieved loop
lcd_init();
}
void serial_rec(void)interrupt 4 using 3
{

 if(RI==1)
 {
 RI=0;
 if(save_mem_flag==0)
 {
   msg[index]=SBUF;
 }
   if(index>205)
 {
    index=206;
 }
 if(index>3)
  flag=1;
  if(index==24 && save_mem_flag==1)
  {
   index=0;
   save_mem_flag=0;
   }
 index++;
 }
}
void main()
{
char k=0;
unsigned char loc;
index=0;
lcd_initialize();
lcd_puts("POST PAID ");
lcd_cmd1(0xc0);
lcd_puts("ENERGY METER");
secdelay(1);
```

```
init_serial(9600);
clrscr();
gsm_check();
clrscr();
lcd_puts("Init..");
at_init2();
clrscr();
EA=0;
call_at_number(number2);
lcd_puts("SMS......");
sms2("Hello ",number1);
sms2("Hello ",number2);
clrscr();
flag=0;
index=0;
IE=0x90;
while(1)
        { //      lcd_cmd1(0x80);
          // lcd_puts("Waiting:::::");
        lcd_cmd1(0x80);
        lcd_puts("Bill:");
        bill=units*214;
        displaypval(bill/100);
        lcd_puts(".");
        display2val(bill%100);
        lcd_puts(" Rs.");
    dev=0;
        lcd_cmd1(0xc0);
        lcd_puts("Uts:");
        display4val(units);
    if(count>9)
        {
         count=0;
         units++;
        }
        lcd_puts(" SU:");
        display2val(count);
        IE=0x9
         if(flag==1)
                {
                                buzz=0;
                                ms_delay(200);
                                ms_delay(200);
                                ms_delay(200);
                                ms_delay(200);
                                buzz=1;
```

```
        clrscr();
lcd_puts("RECEIVED...");
secdelay(3);
lcd_cmd1(0xc0);
      /* k=3;
       ms_delay(200);
       while(k<16)
       {
        lcd_cmd1(0xc0+k-3);
        lcd_data(msg[k]);
        k++;
}

lcd_cmd1(0x80);
        lcd_data(msg[14]);
lcd_data(msg[15]);
        lcd_data(32);
        secdelay(1);
        */
        if(msg[15]==13)
      loc=msg[14]-48;
        else
                loc=((msg[14]-48)*10)+(msg[15]-48);
        lcd_cmd1(0xC0);
        lcd_puts("L=");
        displaypval(loc);
        secdelay(2);
        k=0;
        while(k<index)//Clearing Received Message For New Message.
        {
         msg[k]=32;
         k++;
        }
        k=0;
        if(loc<11)
                read_sms(loc);
        else
        {
         clrscr();
         lcd_puts("not_valid_loc");
         secdelay(2);
        }
        if(loc>10 )
        {
          gsm_check();
                delete_all();
```

```
                    gsm_check();
                }
                index=0;
                flag=0;
                clrscr();
        }       }
}
```

Chapter 6

Conclusion and future scope

Importance of the project

The importance of the project can be well understood if we keep in mind the amount of electricity being stolen everyday.With heavy loads on the power house one cannot track each and every household or commercial site. So to track any misuse at any stage of distribution we can use this method of billing. As a user can get his\her at any instant and can even pay it at any instant.So any kind o misuse by any other person an be avoided.
An extensive amount of energy can be saved If we can track the misuse. The whole power problem can be dealt with by using this technique .The can chance of tempering with this is very low. It's low cost reporting of usage data. Validate accuracy of Utility billing charges – consumption, demand and power factor. Real time data display and access to metering equipment and usage data. So that people at power house and the person who owns the equipment both can monitor the usage whenever the want to.

Conclusion And Future scope

The proposed system for energy billing is automatic without human intervention and consumer can directly know the amount he has to pay .So it is both consumer and power stationfriendly.
The software can developed very easily with the present IT technology. Here the security measures are also taken in consideration so that this system overcomes the drawbacks in present system and also the new system does not give any such chances.
In future, this system can be improved by some additional features meeting the consumer requirements like emergency signal from the power station employees; bill payment acknowledgement alarm etc., by just replacing the transmitter in the consumer side with a

transmitter and receiver antenna. The receiver microcontroller can be made designated for some other applications also.

Bibliography

1. The 8051 microcontroller and embedded systems by Muhammad Ali Mazidi, Janice Gillispie Mazidi and Rolin D. McKinlay
2. Electronic devices and circuit theory by Robert L Boylstead and Louis L Nashalskey.
3. www.atmel.com
4. www.ieee.com
5. www.hplindia.com
6. www.amazon.com
7. Damnjanović M., Predrag Petković P., Jovanović B., *Integrated Power Meter* Proc. of Small Systems Simulation Symposium, pp. 59-62, March 2005, Niš
8. 3-Phase Energy Meter IC, TDK Semiconductors, 2006