Student Minor Research Project

MICROCONTROLLER BASED ULTRASONIC RADAR

Under RUSA 2.0 Scheme

(Through Ch.S.D.St.Theresa's College for Women (Autonomous), Eluru, AP)

Submitted by

Ms R Hema Lalitha , III B.Sc. MECs (Reg.No.11705001)

Ms K T Satyavathi, III B.Sc. MECs (Reg.No.11705006)

Ms D Jessy Vikram, III B.Sc. MECs (Reg.No.11705011)

Under the guidance of

Dr K Venkateswarlu

HOD of Electronics & Project Advisor

Department Of Electronics

SRI Y N COLLEGE

(AUTONOMOUS)

Thrice Accredited by NAAC at 'A' Grade

Recognized by UGC as "College with Potential for Excellence"

Narsapur-534275, AP, India

December-2019

Department Of Electronics

SRI Y N COLLEGE

(AUTONOMOUS)

Thrice Accredited by NAAC at 'A' Grade

Recognized by UGC as "College with Potential for Excellence"

Narsapur-534275, AP, India

Certificate

This is to certify that the project work entitled " **Microcontroller Based Ultrasonic RADAR** *" is bonafied work carried out by Ms R Hema Lalitha (Reg.No: 11705001), Ms K T Satyavathi (Reg.No: 11705006), Ms D Jessy Vikram (Reg.No: 11705011), submitted in Third Year of the degree B.Sc. in Electronics during the year 2019-20 is an authentic work under my supervision and guidance.*

To the best of my knowledge, the matter embodied in the project work has not been submitted to any other College/Institute.

Date: 29-12-2019

Dr K Venkateswarlu
Project Advisor
Department of Electronics

ACKNOWLEDGEMENT

We place on record and warmly acknowledge the continuous encouragement, invaluable supervision, timely suggestions and inspired guidance offered by our Project advisor, ***Dr K Venkateswarlu,*** *Head, Department of Electronics,* ***Sri Y N College (Autonomous), Narsapur*** *in bringing this report to a successful completion.*

We are grateful to ***Mr K Vinaya Phaneendhra****, Lecturer, Department of Electronics for permitting us to make use of the facilities available in the department to carry out the project successfully. Last but not the least we express our sincere thanks to all of our friends who have patiently extended all sorts of help for accomplishing this undertaking.*

Finally we extend our gratefulness to one and all who are directly or indirectly involved in the successful completion of this project work.

Ms. R Hema Lalitha	**Ms. K T Satyavathi**	**Ms. D Jessy Vikram**
III.B.Sc.MECs	III.B.Sc.MECs	III.B.Sc.MECs
Reg. No 11705001	Reg. No.11705006	Reg.No.11705011

DECLARATION

We, the undersigned, declare that the project entitled "**Microcontroller Based Ultrasonic RADAR**", being submitted in Third Year of Bachelor of Science in Electronics, Sri Y N College (Autonomous), is the work carried out by us.

Ms. R Hema Lalitha	**Ms. K T Satyavathi**	**Ms. D Jessy Vikram**
III.B.Sc.MECs	III.B.Sc.MECs	III.B.Sc.MECs
Reg. No 11705001	Reg. No.11705006	Reg.No.11705011

Contents **Page No.**

List of Figures & Tables — Page No.

1. ABSTRACT

We come across situations where we need to keep a watch over prohibited areas to avoid trespassing. Now keeping human labour for this purpose is costly and also not reliable for keeping a watch over an area 24×7. So for this purpose ultrasonic radar project for unauthorized human / animal or object detection system. The system can monitor an area of limited range and alerts authorities with a buzzer as an alarm. For this purpose we use a microcontroller circuit that is connected to an ultrasonic sensor mounted on a servo motor for monitoring.

We also interface a buzzer and LCD screen for monitoring the detection status. The radar keeps monitoring the environment checking the ultrasonic sensor echo. As soon as an object is detected the data of detection is processed and sent to authorities with an alert of where exactly the object was detected. Thus ultrasonic radar proves to be a very useful system for 24×7 monitoring of a particular area/region.

2. INTRODUCTION TO EMBEDDED SYSTEMS

WHAT IS EMBEDDED SYSTEM?

An Embedded System is a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a specific function. An embedded system is a microcontroller-based, software driven, reliable, real-time control system, autonomous, or human or network interactive, operating on diverse physical variables and in diverse environments and sold into a competitive and cost conscious market.

An embedded system is not a computer system that is used primarily for processing, not a software system on PC or UNIX, not a traditional business or scientific application. High-end embedded & lower end embedded systems. High-end embedded system - Generally 32, 64 Bit Controllers used with OS. Examples Personal Digital Assistant and Mobile phones etc .Lower end embedded systems - Generally 8,16 Bit Controllers used with an minimal operating systems and hardware layout designed for the specific purpose.

SYSTEM DESIGN CALLS:

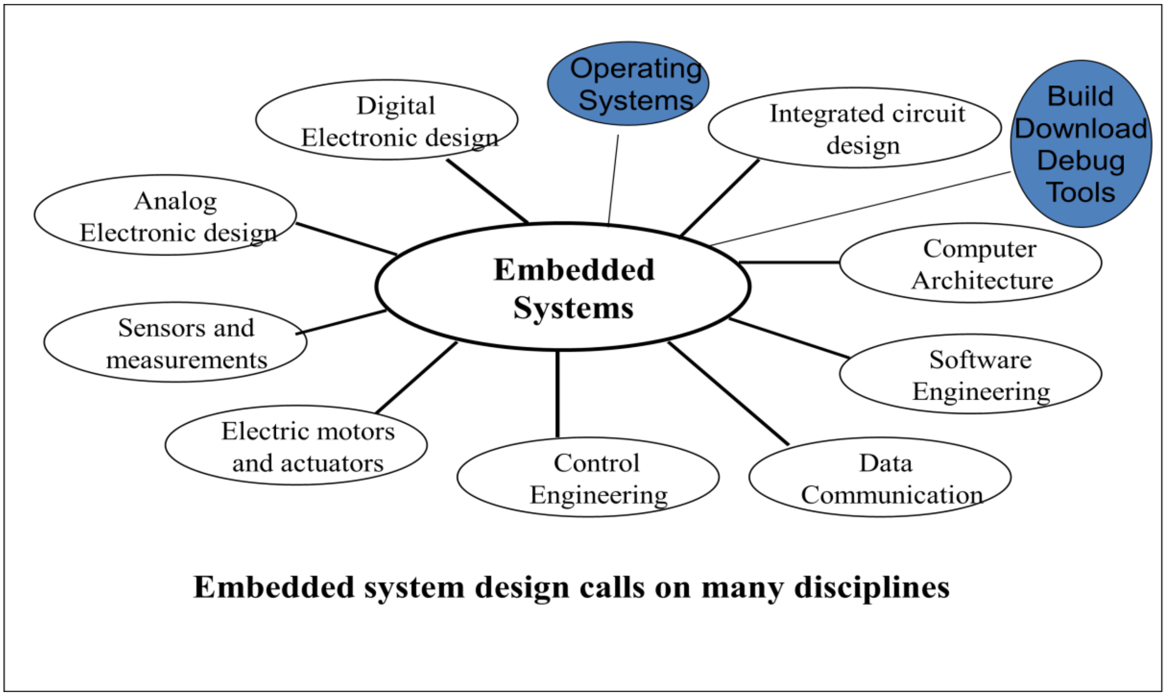


Fig 2(a): Embedded system design calls

EMBEDDED SYSTEM DESIGN CYCLE

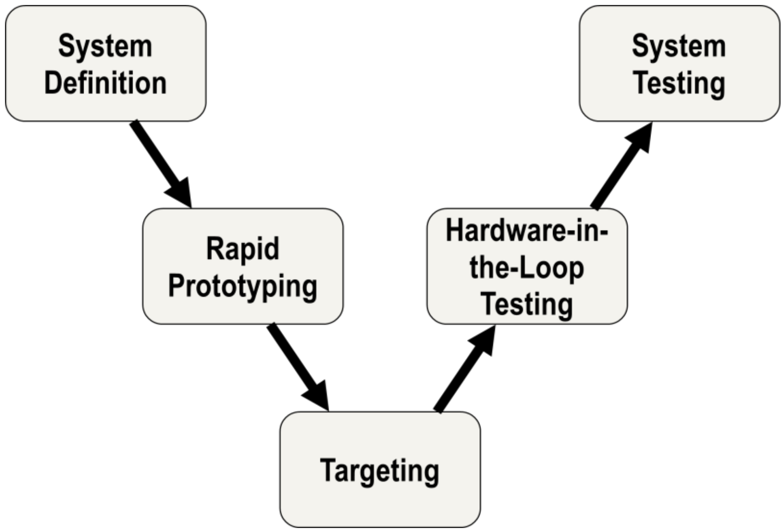


Fig:2(b) "V Diagram"

Characteristics of Embedded System

- An embedded system is any computer system hidden inside a product other than a computer.
- They will encounter a number of difficulties when writing embedded system software in addition to those we encounter when we write applications.
  - Throughput – Our system may need to handle a lot of data in a short period of time.
  - Response–Our system may need to react to events quickly
  - Testability–Setting up equipment to test embedded software can be difficult.
  - Debugability–Without a screen or a keyboard, finding out what the software is doing wrong (other than not working) is a troublesome problem.
  - Reliability – embedded systems must be able to handle any situation without human intervention.
  - Memory space – Memory is limited on embedded systems, and you must make the software and the data fit into whatever memory exists.
  - Program installation – you will need special tools to get your software into embedded systems.

- – Power consumption – Portable systems must run on battery power, and the software in these systems must conserve power.
- – Processor hogs – computing that requires large amounts of CPU time can complicate the response problem.
- – Cost – Reducing the cost of the hardware is a concern in many embedded system projects; software often operates on hardware that is barely adequate for the job.

- Embedded systems have a microprocessor/ microcontroller and a memory. Some have a serial port or a network connection. They usually do not have keyboards, screens or disk drives.

APPLICATIONS

1) Military and aerospace embedded software applications
2) Communication Applications
3) Industrial automation and process control software
4) Mastering the complexity of applications.
5) Reduction of product design time.
6) Real time processing of ever increasing amounts of data.
7) Intelligent, autonomous sensors.

CLASSIFICATION

- Real Time Systems.
- RTS is one which has to respond to events within a specified deadline.
- A right answer after the dead line is a wrong answer.

RTS CLASSIFICATION

- Hard Real Time Systems
- Soft Real Time System

HARD REAL TIME SYSTEM

- "Hard" real-time systems have very narrow response time.
- Example: Nuclear power system, Cardiac pacemaker.

SOFT REAL TIME SYSTEM

- "Soft" real-time systems have reduced constrains on "lateness" but still must operate very quickly and repeatable.
- Example: Railway reservation system – takes a few extra seconds the data remains valid.

3. PROJECT BLOCK DIAGRAM

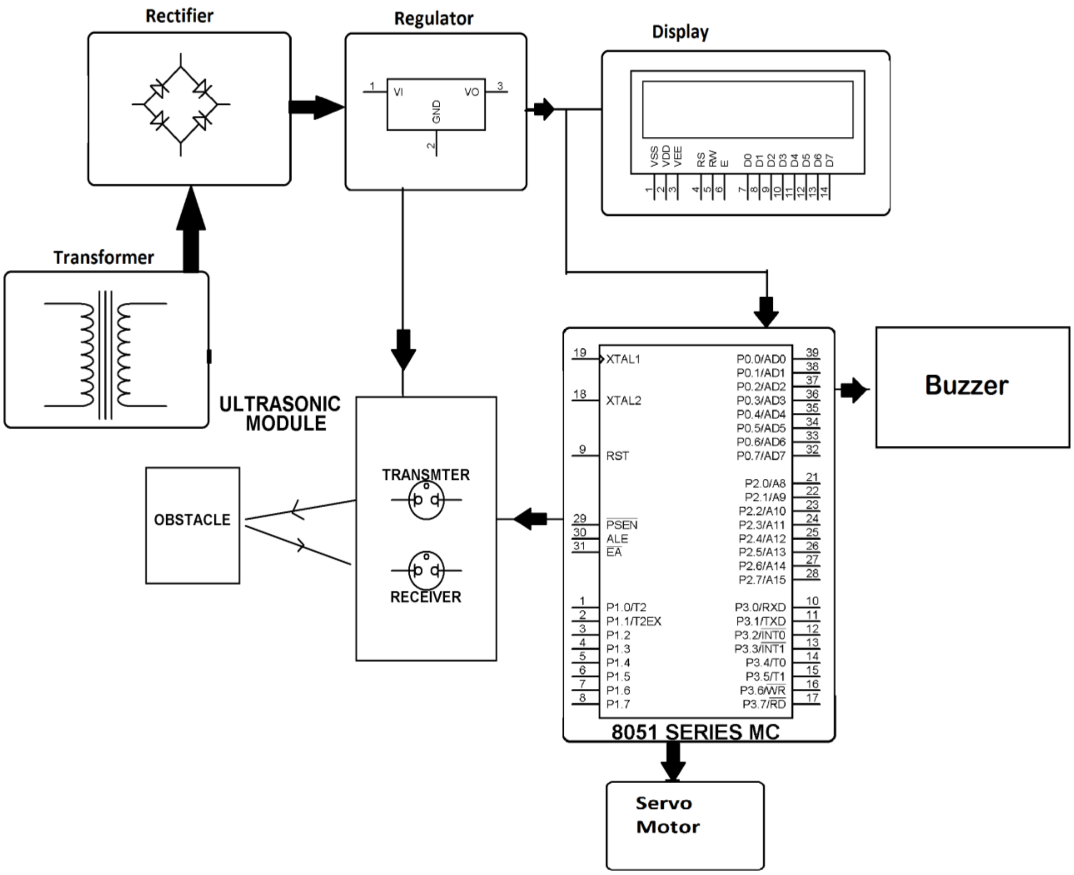


Fig (3) Block diagram of project

4. HARDWARE REQUIREMENTS

HARDWARE COMPONENT:

1. TRANSFORMER
2. BRIDGE RECTIFIER
3. FILTER
4. VOLTAGE REGULATOR (7805)
5. MICROCONTROLLER
6. ULTRASONIC MODULE
7. SERVO MOTOR
8. LCD
9. LED
10. 1N4007
11. RESISTOR
12. CAPACITOR
13. PUSH BUTTON

4.1 BATTERY

An electrical battery is a combination of one or more electrochemical cells, used to convert stored chemical energy into electrical energy. The battery has become a common power source for many household and industrial applications.

Batteries may be used once and discarded, or recharged for years as in standby power applications. Miniature cells are used to power devices such as hearing aids and wristwatches; larger batteries provide standby power for telephone exchanges or computer data centers.

<u>WORKING OF BATTERY:</u>

A battery is a device that converts chemical energy directly to electrical energy. It consists of a number of voltaic cells; each voltaic cell consists of two half cells connected in series by a conductive electrolyte containing anions and cat ions. One half-cell includes electrolyte and the electrode to which anions (negatively-charged ions) migrate, i.e. the anode or negative electrode; the other half-cell includes electrolyte and the electrode to which cat ions (positively-charged ions) migrate, i.e. the cathode or positive electrode. In the red ox reaction that powers the battery, reduction (addition of electrons) occurs to cat ions at the cathode, while oxidation (removal of electrons) occurs to anions at the anode. The electrodes do not touch each other but are electrically connected by the electrolyte. Many cells use two half-cells with different electrolytes. In that case each half-cell is enclosed in a container, and a separator that is porous to ions but not the bulk of the electrolytes prevents mixing.

Each half cell has an electromotive force (or emf), determined by its ability to drive electric current from the interior to the exterior of the cell. The net emf of the cell is the difference between the emfs of its half-cells. Therefore, if the electrodes have emfs and, in other words, the net emf is the difference between the reduction potentials of the half-reactions.

The electrical driving force or across the terminals of a cell is known as the terminal voltage (difference) and is measured in volts. The terminal voltage of a cell that is neither charging nor discharging is called the open-circuit voltage and equals the emf of the cell. Because of internal resistance, the terminal voltage of a cell that is discharging is smaller in magnitude than the open-circuit voltage and the terminal voltage of a cell that is charging exceeds the open-circuit voltage. An ideal cell has negligible internal resistance, so it would maintain a constant terminal voltage of until exhausted, then dropping to zero. If such a cell maintained 1.5 volts and stored a charge of one Coulomb then on complete discharge it would perform 1.5 Joule of work. In actual cells, the internal resistance increases under discharge, and the open circuit voltage also decreases under discharge. If the voltage and resistance are plotted against time, the resulting graphs typically are a curve; the shape of the curve varies according to the chemistry and internal arrangement employed.

An electrical battery is one or more electrochemical cells that convert stored chemical energy into electrical energy. Since the invention of the first battery (or "voltaic pile") in 1800 by Alessandro Volta, batteries have become a common power source for many household and industrial applications. According to a 2005 estimate, the worldwide battery industry generates US$48 billion in sales each year, with 6% annual growth. There are two types of batteries: primary batteries (disposable batteries), which are designed to be used once and discarded, and secondary batteries (rechargeable batteries), which are designed to be recharged and used multiple times. Miniature cells are used to power devices such as hearing aids and wristwatches; larger batteries provide standby power for telephone exchanges or computer data centers.

Principle of operation

A battery is a device that converts chemical energy directly to electrical energy. It consists of a number of voltaic cells; each voltaic cell consists of two half cells connected in series by a conductive electrolyte containing anions and cations. One half-cell includes electrolyte and the electrode to which anions (negatively charged ions) migrate, i.e., the anode or negative electrode; the other half-cell includes electrolyte and the electrode to which cations (positively charged ions) migrate, i.e., the cathode or positive electrode. In the redox reaction that powers the battery, cations are reduced (electrons are added) at the cathode, while anions are oxidized (electrons are removed) at the anode. The electrodes do not touch each other but are electrically connected by the electrolyte. Some cells use two half-cells with different electrolytes. A separator between half cells allows ions to flow, but prevents mixing of the electrolytes.

Each half cell has an electromotive force (or emf), determined by its ability to drive electric current from the interior to the exterior of the cell. The net emf of the cell is the difference between the emfs of its half-cells, as first recognized by Volta. Therefore, if the electrodes have emfs $\mathcal{E}_1$and $\mathcal{E}_2$, then the net emf is $\mathcal{E}_2 - \mathcal{E}_1$; in other words, the net emf is the difference between the reduction potentials of the half-reactions. The electrical driving force or ΔV_{bat} across the terminals of a cell is known as the terminal voltage (difference) and is measured in volts. The terminal voltage of a cell that is neither charging nor discharging is called the open-circuit voltage and equals the emf of the cell. Because of internal resistance, the terminal voltage of a cell that is discharging is smaller in magnitude than the open-circuit voltage and the terminal voltage of a cell that is charging exceeds the open-circuit voltage. An ideal cell has negligible internal resistance, so it would maintain a constant terminal voltage of $\mathcal{E}$until exhausted, then dropping to zero. If such a cell maintained 1.5 volts and stored a charge of one coulomb then on complete discharge it would perform 1.5 joule of work. In actual cells, the internal resistance increases under discharge, and the open circuit voltage also decreases under discharge. If the voltage and resistance are plotted against time, the resulting graphs typically are a curve; the shape of the curve varies according to the chemistry and internal arrangement employed.

As stated above, the voltage developed across a cell's terminals depends on the energy release of the chemical reactions of its electrodes and electrolyte. Alkaline and carbon-zinc cells have different chemistries but approximately the same emf of 1.5 volts; likewise NiCd and NiMH cells have different chemistries, but approximately the same emf of 1.2 volts. On the other hand the high electrochemical potential changes in the reactions of lithium compounds give lithium cells emfs of 3 volts or more.

Lead-acid

Tried, tested, and trusted, lead-acid batteries have been with us since the middle of the 19th century. With an overall rating of 12 volts, they have six separate cells, each producing 2 volts. Crudely reduced to its basic components, each cell has a "spongy" lead metal electrode (negative), a lead dioxide electrode (positive), and a sulfuric acid electrolyte. As the battery discharges, both electrodes become coated with lead sulfate and the sulfuric acid is largely converted into water, while electrons flow out around the external circuit to provide power.

Lead-acid batteries made it possible to start cars without the help of a dangerous and dirty hand crank. Normally, you never have to recharge them—because your car does that automatically. The battery discharges (gives up a little of its energy) to help the car's gasoline engine start up, and recharges (gets energy back again) when the engine begins generating electrical energy through a device called an alternator. As for disadvantages, lead-acid batteries are relatively big, surprisingly heavy (try lifting one!), expensive, and can't be fully charged and discharged too many times. Another problem is their use of toxic lead metal, which can cause environmental problems when they're dumped in landfills.

Nickel-cadmium

Nickel-cadmium (NiCd, pronounced "nicad") are widely used as replacements for disposable 1.5-volt batteries in things like toys, flashlights, and power tools. They're relatively cheap, can be charged and discharged hundreds of times, and, properly treated, will last about a decade.

Although very dependable, it's often said that NiCd batteries need to be discharged fully before you charge them up or the amount of charge they will store (and their effective lifespan) can be greatly reduced. Opinions vary on whether this is true and, if so, why it happens, but as a rule of thumb, regularly discharging batteries completely and then recharging them is a good practice. Another problem with NiCd batteries is the toxic cadmium metal they contain. If they are buried in a landfill, instead of properly recycled, the cadmium can escape into the soil and could potentially pollute watercourses nearby.

Nickel-metal-hydride (NiMH)

Nickel metal hydride batteries work in a similar way, but suffer less from the so-called "memory effect." They became a popular alternative to NiCd batteries in the 1990s, partly because of environmental concerns about cadmium. NiMH batteries work more effectively in gadgets like cellphones, which are often "topped-up" with a quick recharge instead of a complete discharge and recharge (which is more typical with something like power tools).

Lithium-ion

Lithium-ion batteries are the fastest-growing type of rechargeables; there are probably lithium-ion batteries in your cellphone, MP3 player, and laptop computer. What's so good about lithium? It's a lightweight metal that easily forms ions, so it's excellent for making batteries. The latest lithium-ion batteries can store about twice as much energy as traditional NiCd rechargeables, work at higher voltages, and are more environmentally friendly, but don't last as long. Even so, they can be charged and discharged hundreds of times and typically last several years, so they're great for everyday use in electronic gadgets that aren't meant to last that long.

How do they work? When you plug a cellphone or laptop into the power supply, the lithium-ion battery inside starts buzzing with chemical activity. The battery's job is to store as much electricity as possible, as fast as possible. It does this through a chemical reaction that shunts lithium ions (lithium atoms that have lost an electron to become positively charged) from one part of the battery to another. When you unplug the power and use your laptop or phone, the battery switches into reverse: the ions move the opposite way and the battery gradually loses its charge. Read more in our main article on how lithium-ion batteries work.

4.2 VOLTAGE REGULATOR LM7805

Features

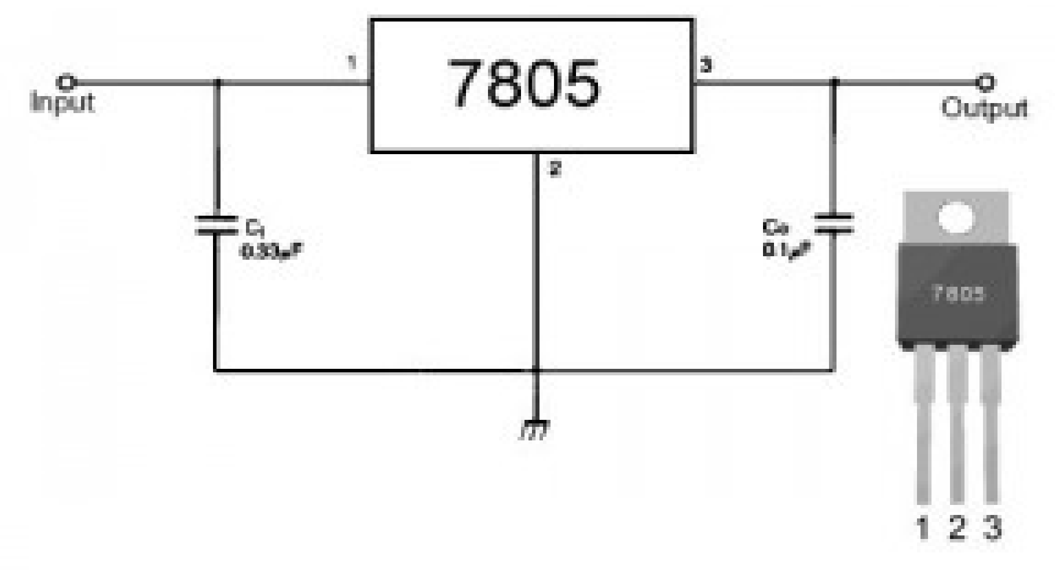
• Output Current up to 1A.

• Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V.

• Thermal Overload Protection.

• Short Circuit Protection.

• Output Transistor Safe Operating Area Protection.



Description

The LM78XX/LM78XXA series of three-terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a Wide range of applications. Each type employs internal current limiting, thermal shutdown and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output Current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.

Internal Block Diagram

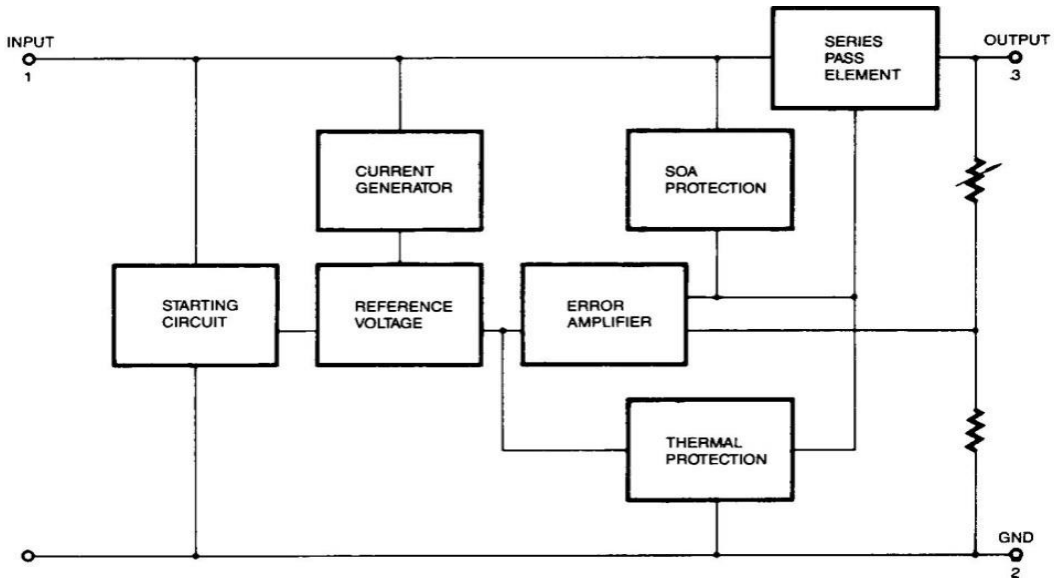


Fig 4.2(a): Block Diagram of Voltage Regulator

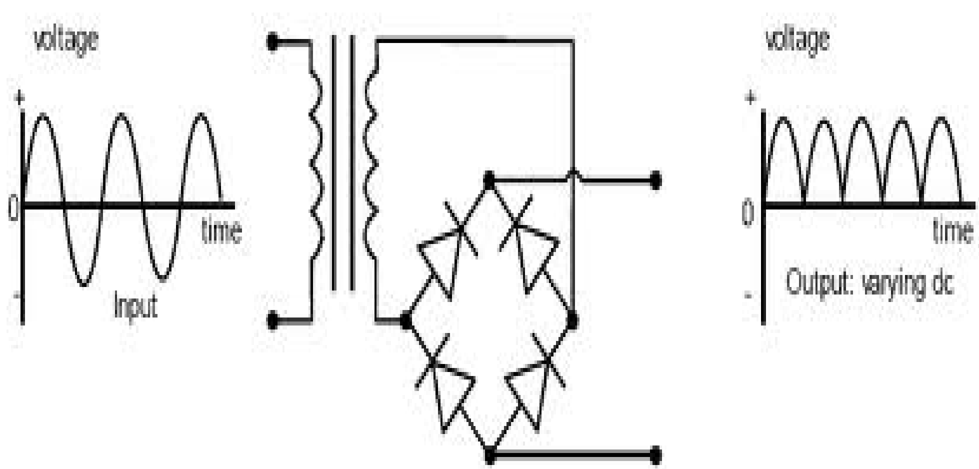
Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for V_O = 5V to 18V) (for V_O = 24V)	V_I V_I	35 40	V V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	°C/W
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	°C/W
Operating Temperature Range (KA78XX/A/R)	T_{OPR}	0 ~ +125	°C
Storage Temperature Range	T_{STG}	-65 ~ +150	°C

Table 4.2(b): Ratings of the Voltage Regulator

4.3 RECTIFIER

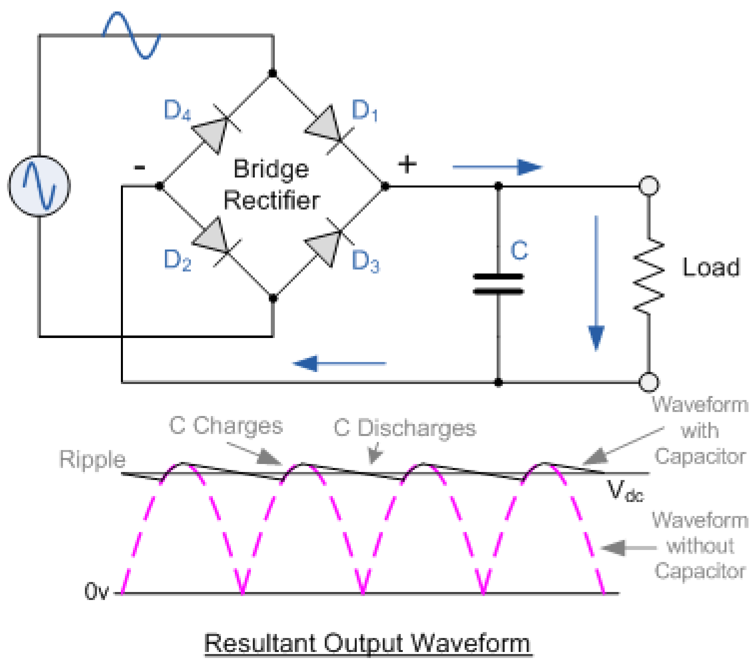
A rectifier is an electrical device that converts alternating current (AC), which periodically reverses direction, to direct current (DC), current that flows in only one direction, a process known as rectification. Rectifiers have many uses including as components of power supplies and as detectors of radio signals. Rectifiers may be made of solid statediodes, vacuum tube diodes, mercury arc valves, and other components. The output from the transformer is fed to the rectifier. It converts A.C. into pulsating D.C. The rectifier may be a half wave or a full wave rectifier. In this project, a bridge rectifier is used because of its merits like good stability and full wave rectification. In positive half cycleonly two diodes (1 set of parallel diodes) will conduct, in negative half cycle remaining two diodes will conduct and they will conduct only in forward bias only.



4.4 FILTER

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothens the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore a regulator is applied at the output stage.

The simple capacitor filter is the most basic type of power supply filter. The use of this filter is very limited. It is sometimes used on extremely high-voltage, low-current power supplies for cathode-ray and similar electron tubes that require very little load current from the supply. This filter is also used in circuits where the power-supply ripple frequency is not critical and can be relatively high. Below figure can show how the capacitor charges and discharges.



4.5 MICROCONTROLLER AT89S52

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of in-system programmable Flash memory.The device is manufactured using Atmel's high-density non volatile memory technology and is compatible with the industry standard 80C51 instruction set and pin out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional non volatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications. The AT89S52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset.

Features:

• Compatible with MCS®-51 Products

• 8K Bytes of In-System Programmable (ISP) Flash Memory

– Endurance: 10,000 Write/Erase Cycles

• 4.0V to 5.5V Operating Range

• Fully Static Operation: 0 Hz to 33 MHz

• Three-level Program Memory Lock

• 256 x 8-bit Internal RAM

• 32 Programmable I/O Lines

• Three 16-bit Timer/Counters

• Eight 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

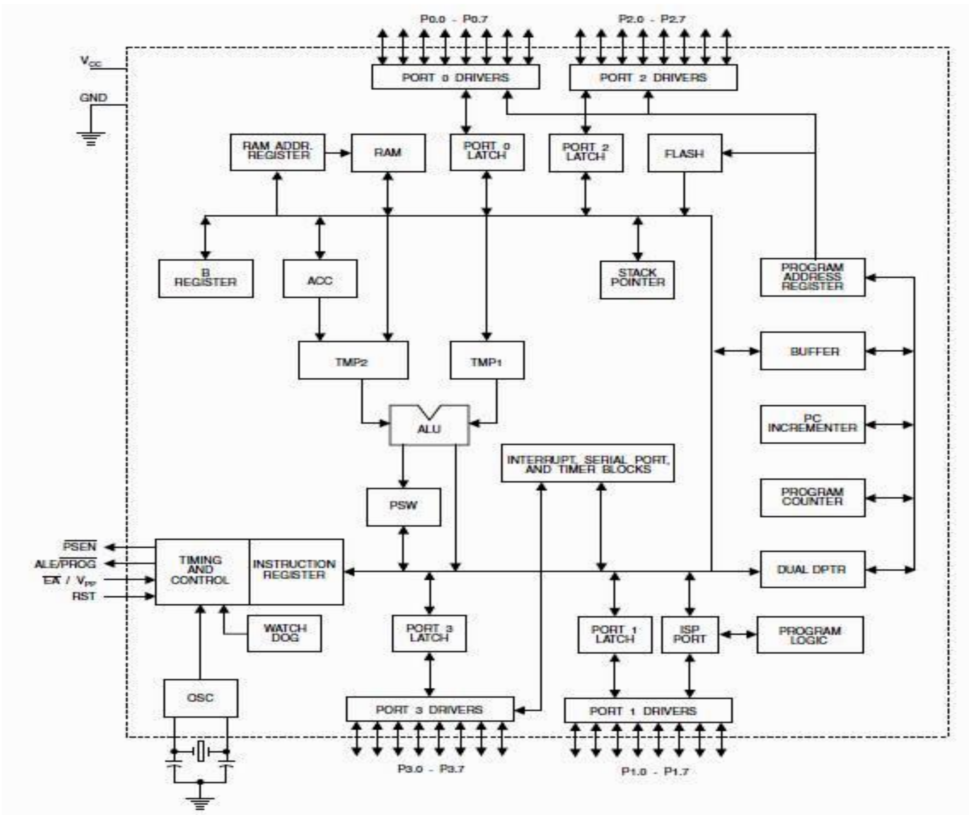
• Power-off Flag

• Fast Programming Time

• Flexible ISP Programming (Byte and Page Mode)

• Green (Pb/Halide-free) Packaging Option

Block Diagram of AT89S52:



4.5(a): Block Diagram of AT89S52

Pin Configurations of AT89S52

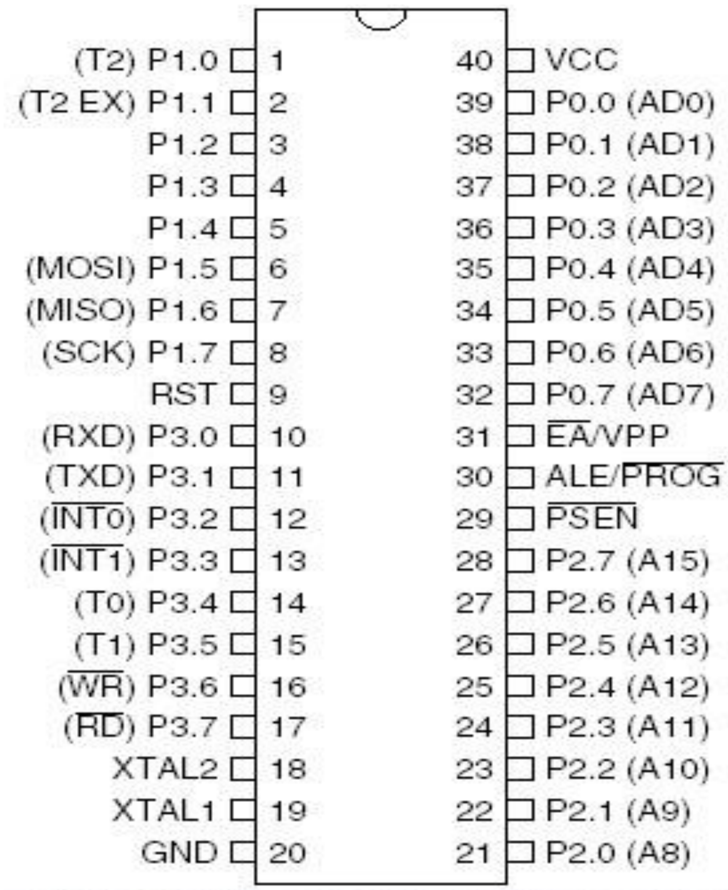


Figure taken from a datasheet provided by ATMEL™

Fig 4.5(b): Pin Diagram of AT89S52

Pin Description:

VCC:

Supply voltage.

GND:

Ground

Port 0:

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

Port 1:

Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (IIL) because of the internal pull-ups. In addition, P1.0 and P1.1 can be configured to be the timer/counter 2 external count input (P1.0/T2) and the timer/counter 2 trigger input (P1.1/T2EX).

Port 2:

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

Port 3:

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

RST:

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives high for 98 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.

ALE/PROG:

Address Latch Enable (ALE) is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

PSEN:

Program Store Enable (PSEN) is the read strobe to external program memory. When the AT89S52 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

EA/VPP:

External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, EA will be internally latched on reset. EA should be strapped to VCC for internal program executions. This pin also receives the 12-volt programming enable voltage (VPP) during Flash programming.

XTAL1:

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2:

Output from the inverting oscillator amplifier.

Oscillator Characteristics:

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 6.2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

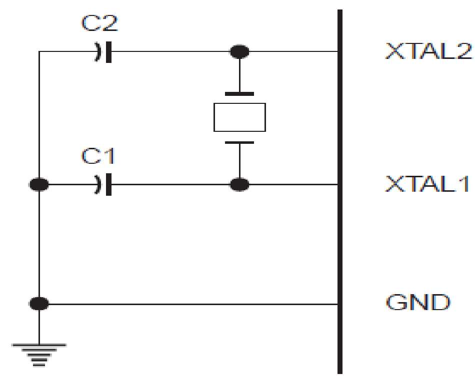


Fig 4.5(c): Oscillator Connections

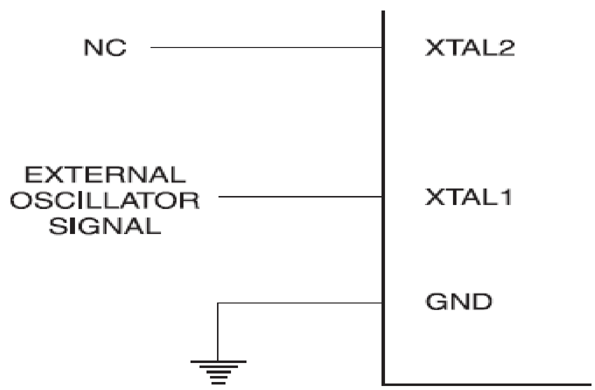


FIG 4.5(d): External Clock Drive Configuration

Idle Mode

In idle mode, the CPU puts itself to sleep while all the on chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

Power down Mode

In the power down mode the oscillator is stopped, and the instruction that invokes power down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power down mode is terminated. The only exit from power down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before VCC is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

4.6 ULTRASONIC SENSOR MODULE

Here is a more easy use serial ultrasonic module. It will auto output the distance information via serial port after power on, you don't need to do any trigger and calculated, just need to read the serial pin and get the distance information.

Ultrasonic sensor provides a very low-cost and easy method of distance measurement. This sensor is perfect for any number of applications that require you to perform measurements between moving or stationary objects. Naturally, robotics applications are very popular but you'll also find this product to be useful in security systems or as an infrared replacement if so desired. You will definitely appreciate the activity status LED and the economic use of just one I/O pin.

The ultrasonic sensor measures distance using sonar; an ultrasonic (well above human hearing) pulse is transmitted from the unit and distance-to-target is determined by measuring the time required for the echo return. Output from the ultrasonic sensor is a variable-width pulse that corresponds to the distance to the target.

Features:

- Provides precise, non-contact distance measurements within a 2 cm to 3 m range
- Simple pulse in/pulse out communication
- Burst indicator LED shows measurement in progress

- 20 mA power consumption
- Narrow acceptance angle
- 3-pin header makes it easy to connect using a servo extension cable, no soldering required

Key Specifications:

- power supply :5V DC
- quiescent current : <15mA
- effectual angle: <15°
- ranging distance : 2cm – 350 cm
- resolution : 0.3 cm
- Output cycle : 50ms
- Baud Rate : 9600

Output frame format （4Bytes）

- 1st Byte : 0xFF (Start bit, Fixed value)
- 2nd Byte: H_Data (High 8 bit of the distance)
- 3rd Byte: L_Date (Low 8 bit of the distance)
- 4th Byte: Check Sum (0XFF+H_DATA+L_DATA=SUM, Check SUM is the low 8 bit of the SUM)

Uses:

Ultrasonic sensors are used to detect the presence of targets and to measure the distance to targets in many automated factories and process plants. Sensors with an on or off digital output are available for detecting the presence of objects, and sensors with an analog output which varies proportionally to the sensor to target separation distance are commercially available.

Because ultrasonic sensors use sound rather than light for detection, they work in applications where photoelectric sensors may not. Ultrasonic are a great solution for clear object detection and for liquid level measurement, applications that photo electrics struggle with because of target translucence. Target colour and/or reflectivity don't affect ultrasonic sensors which can operate reliably in high-glare environments.

Other types of transducers are used in commercially available ultrasonic cleaning devices. An ultrasonic transducer is affixed to a stainless steel pan which is filled with a solvent (frequently water or isopropanol) and a square wave is applied to it, imparting vibrational energy on the liquid.

4.7 LIQUID CRYSTAL DISPLAY (LCD)

Description:

This is the example for the Parallel Port. This example doesn't use the Bi-directional feature found on newer ports, thus it should work with most, if not all Parallel Ports. It however doesn't show the use of the Status Port as an input for a 16 Character x 2 Line LCD Module to the Parallel Port. These LCD Modules are very common these days, and are quite simple to work with, as all the logic required running them is on board.

Pros:

- Very compact and light
- Low power consumption
- No geometric distortion
- Little or no flicker depending on backlight technology
- Not affected by screen burn-in
- No high voltage or other hazards present during repair/service
- Can be made in almost any size or shape
- No theoretical resolution limit

LCD Background:

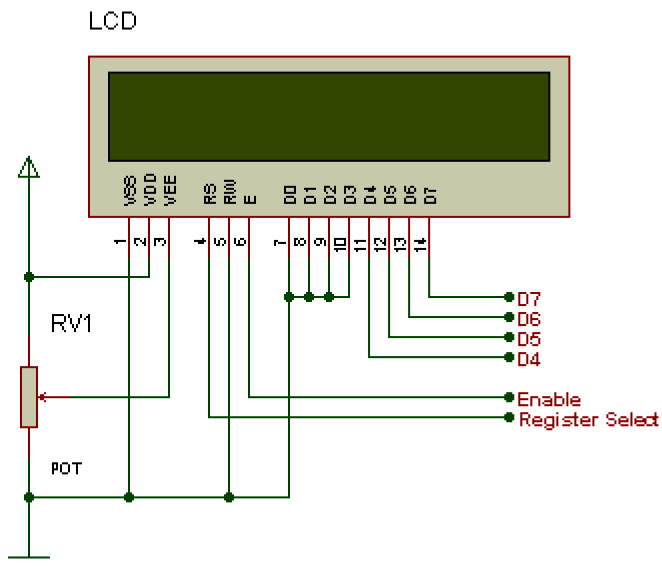
Frequently, an 8051 program must interact with the outside world using input and output devices that communicate directly with a human being. One of the most common devices attached to an 8051 is an LCD display. Some of the most common LCDs connected to the 8051 are 16x2 and 20x2 displays. This means 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

Fortunately, a very popular standard exists which allows us to communicate with the vast majority of LCDs regardless of their manufacturer. The standard is referred to as HD44780U, which refers to the controller chip which receives data from an external source (in this case, the 8051) and communicates directly with the LCD.

Fig 4.7: LCD

44780 LCD BACKGROUND

The 44780 standard requires 3 control lines as well as either 4 or 8 I/O lines for the data bus. The user may select whether the LCD is to operate with a 4-bit data bus or an 8-bit data bus. If a 4-bit data bus is used the LCD will require a total of 7 data lines (3 control lines plus the 4 lines for the data bus). If an 8-bit data bus is used the LCD will require a total of 11 data lines (3 control lines plus the 8 lines for the data bus).



The three control lines are referred to as EN, RS, and RW.

The EN line is called "Enable." This control line is used to tell the LCD that you are sending it data. To send data to the LCD, your program should make sure this line is low (0) and then set the other two control lines and/or put data on the data bus. When the other lines are completely ready, bring EN high (1) and wait for the minimum amount of time required by the LCD datasheet (this varies from LCD to LCD), and end by bringing it low (0) again.

The RS line is the "Register Select" line. When RS is low (0), the data is to be treated as a command or special instruction (such as clear screen, position cursor, etc.). When RS is high (1), the data being sent is text data which should be displayed on the screen. For example, to display the letter "T" on the screen you would set RS high.

The RW line is the "Read/Write" 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) the LCD. Only one instruction ("Get LCD status") is a read command. All others are write commands--so RW will almost always be low .Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, and DB7.

4.8 LED

A Light-Emitting Diode (LED) is a semiconductor light source. LEDs are used as indicator lamps in many devices, and are increasingly used for lighting. When a light-emitting diode is forward biased (switched on), electrons are able to recombine with holes within the device, releasing energy in the form of photons.

This effect is called electroluminescence and the color of the light (corresponding to the energy of the photon) is determined by the energy gap of the semiconductor. An LED is often small in area (less than 1 mm^2), and integrated optical components may be used to shape its radiation pattern. LEDs present many advantages over incandescent light sources including lower energy consumption, longer lifetime, improved robustness, smaller size, faster switching, and greater durability and reliability.

Types of LED'S

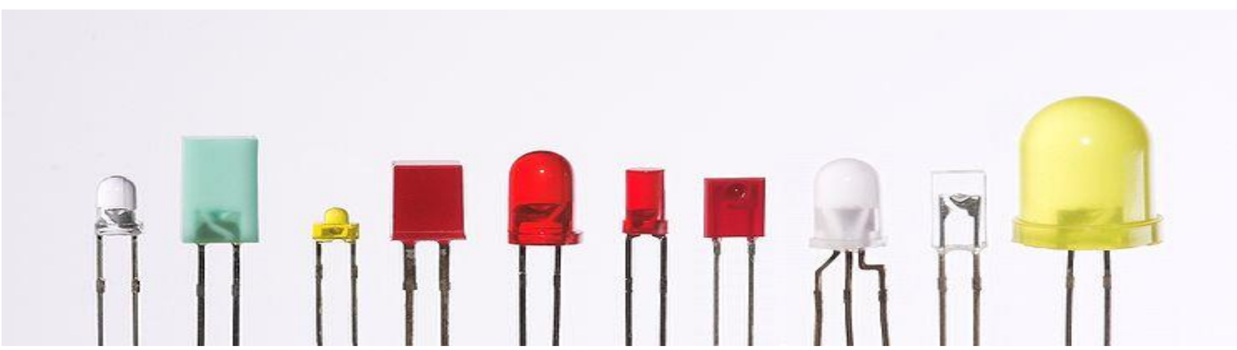


Fig 4.8 (a): Types of LED

Light-emitting diodes are used in applications as diverse as replacements for aviation lighting, automotive lighting as well as in traffic signals. The compact size, the possibility of narrow bandwidth, switching speed, and extreme reliability of LEDs has allowed new text and video displays and sensors to be developed, while their high switching rates are also useful in advanced communications technology.

Electronic Symbol:

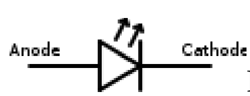


Fig 4.8 (b): symbol of LED

White LED'S

Light Emitting Diodes (LED) have recently become available that are both white and bright, so bright that they seriously compete with incandescent lamps in lighting applications. They are still pretty expensive as compared to a GOW lamp but draw much less current and project a fairly well focused beam.

When run within their ratings, they are more reliable than lamps as well. Red LEDs are now being used in automotive and truck tail lights and in red traffic signal lights. You will be able to detect them because they look like an array of point sources and they go on and off instantly as compared to conventional incandescent lamps.

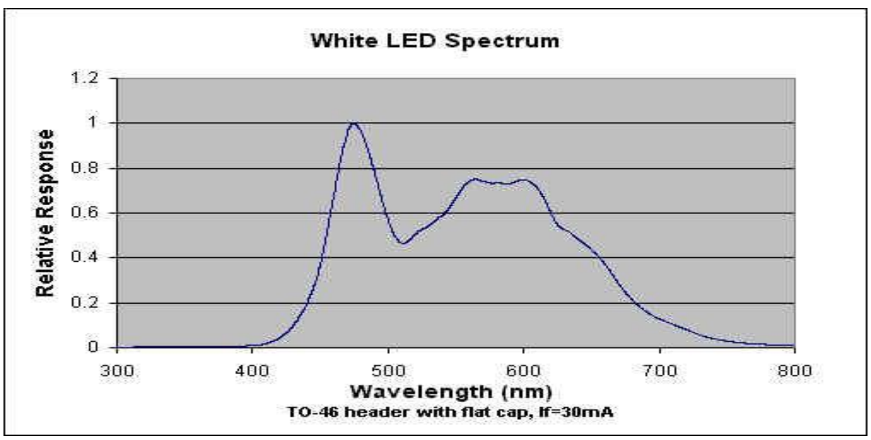


Fig 4.8(c): White LED spectrum

4.9 IN4007

Diodes are used to convert AC into DC these are used as half wave rectifier or full wave rectifier. Three points must he kept in mind while using any type of diode.

1. Maximum forward current capacity
2. Maximum reverse voltage capacity
3. Maximum forward voltage capacity

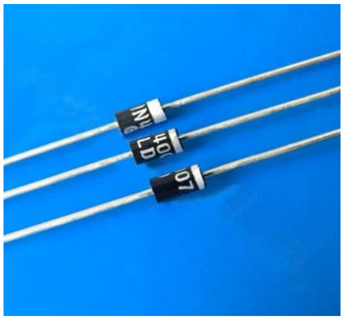


Fig 4.9(a): 1N4007 diodes

The number and voltage capacity of some of the important diodes available in the market are as follows:

- Diodes of number IN4001, IN4002, IN4003, IN4004, IN4005, IN4006 and IN4007 have maximum reverse bias voltage capacity of 50V and maximum forward current capacity of 1 Amp.
- Diode of same capacities can be used in place of one another. Besides this diode of more capacity can be used in place of diode of low capacity but diode of low capacity cannot be used in place of diode of high capacity. For example, in place of IN4002; IN4001 or IN4007 can be used but IN4001 or IN4002 cannot be used in place of IN4007.The diode BY125made by company BEL is equivalent of diode from IN4001 to IN4003. BY 126 is equivalent to diodes IN4004 to 4006 and BY 127 is equivalent to diode IN4007.

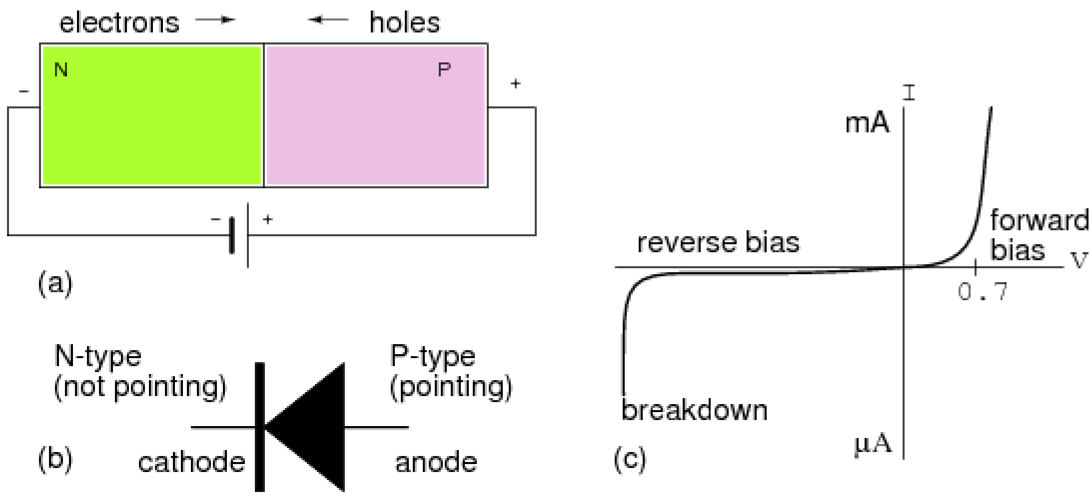


Fig 4.9(b) :PN Junction diode

4.10 RESISTORS

A resistor is a two-terminal electronic component designed to oppose an electric current by producing a voltage drop between its terminals in proportion to the current, that is, in accordance with Ohm's law:

$$V = IR$$

Resistors are used as part of electrical networks and electronic circuits. They are extremely commonplace in most electronic equipment. Practical resistors can be made of various compounds and films, as well as resistance wire (wire made of a high-resistivity alloy, such as nickel/chrome).

The primary characteristics of resistors are their resistance and the power they can dissipate. Other characteristics include temperature coefficient, noise, and inductance. Less well-known is critical resistance, the value below which power dissipation limits the maximum permitted current flow, and above which the limit is applied voltage. Critical resistance depends upon the materials constituting the resistor as well as its physical dimensions; it's determined by design.

Resistors can be integrated into hybrid and printed circuits, as well as integrated circuits. Size, and position of leads (or terminals) are relevant to equipment designers; resistors must be physically large enough not to overheat when dissipating their power.

A resistor is a two-terminalpassiveelectronic component which implements electrical resistance as a circuit element. When a voltage V is applied across the terminals of a resistor, a current I will flow through the resistor in direct proportion to that voltage. The reciprocal of the constant of proportionality is known as the resistance R, since, with a given voltage V, a larger value of R further "resists" the flow of current I as given by Ohm's law:

$$I = \frac{V}{R}$$

Resistors are common elements of electrical networks and electronic circuits and are ubiquitous in most electronic equipment. Practical resistors can be made of various compounds and films, as well as resistance wire (wire made of a high-resistivity alloy, such as nickel-chrome). Resistors are also implemented within integrated circuits, particularly analog devices, and can also be integrated into hybrid and printed circuits.

The electrical functionality of a resistor is specified by its resistance: common commercial resistors are manufactured over a range of more than 9 orders of magnitude. When specifying that resistance in an electronic design, the required precision of the resistance may require attention to the manufacturing tolerance of the chosen resistor, according to its specific application. The temperature coefficient of the resistance may also be of concern in some precision applications. Practical resistors are also specified as having a maximum power rating which must exceed the anticipated power dissipation of that resistor in a particular circuit: this is mainly of concern in power electronics applications. Resistors with higher power ratings are physically larger and may require heat sinking. In a high voltage circuit, attention must sometimes be paid to the rated maximum working voltage of the resistor.

The series inductance of a practical resistor causes its behaviour to depart from ohms law; this specification can be important in some high-frequency applications for smaller values of resistance. In a low-noise amplifier or pre-amp the noise characteristics of a resistor may be an issue. The unwanted inductance, excess noise, and temperature coefficient are mainly dependent on the technology used in manufacturing the resistor. They are not normally specified individually for a particular family of resistors manufactured using a particular technology. A family of discrete resistors is also characterized according to its form factor, that is, the size of the device and position of its leads (or terminals) which is relevant in the practical manufacturing of circuits using them.

Units

The ohm (symbol: Ω) is the SI unit of electrical resistance, named after Georg Simon Ohm. An ohm is equivalent to a volt per ampere. Since resistors are specified and manufactured over a very large range of values, the derived units of milliohm (1 mΩ = 10^{-3} Ω), kilohm (1 kΩ = 10^{3} Ω), and megohm (1 MΩ = 10^{6} Ω) are also in common usage.

The reciprocal of resistance R is called conductance G = 1/R and is measured in Siemens (SI unit), sometimes referred to as a mho. Thus a Siemens is the reciprocal of an ohm: $S = \Omega^{-1}$. Although the concept of conductance is often used in circuit analysis, practical resistors are always specified in terms of their resistance (ohms) rather than conductance.

Theory of operation

Ohm's law

The behaviour of an ideal resistor is dictated by the relationship specified in Ohm's law:

$$V = I \cdot R$$

Ohm's law states that the voltage (V) across a resistor is proportional to the current (I) passing through it, where the constant of proportionality is the resistance (R).

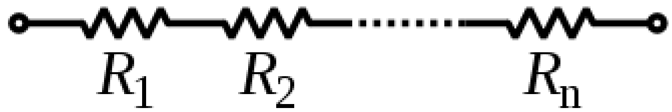
Equivalently, Ohm's law can be stated:

$$I = \frac{V}{R}$$

This formulation of Ohm's law states that, when a voltage (V) is present across a resistance (R), a current (I) will flow through the resistance. This is directly used in practical computations. For example, if a 300 ohm resistor is attached across the terminals of a 12 volt battery, then a current of 12 / 300 = 0.04 amperes (or 40 mill amperes) will flow through that resistor.

Series and parallel resistors

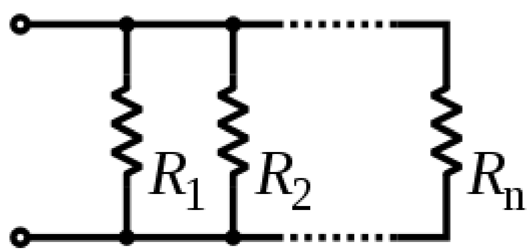
In a series configuration, the current through all of the resistors is the same, but the voltage across each resistor will be in proportion to its resistance. The potential difference (voltage) seen across the network is the sum of those voltages, thus the total resistance can be found as the sum of those resistances:



$$R_{eq} = R_1 + R_2 + \cdots + R_n$$

As a special case, the resistance of N resistors connected in series, each of the same resistance R, is given by NR.

Resistors in a parallel configuration are each subject to the same potential difference (voltage), however the currents through them add. The conductance of the resistors then add to determine the conductance of the network. Thus the equivalent resistance (R_{eq}) of the network can be computed:



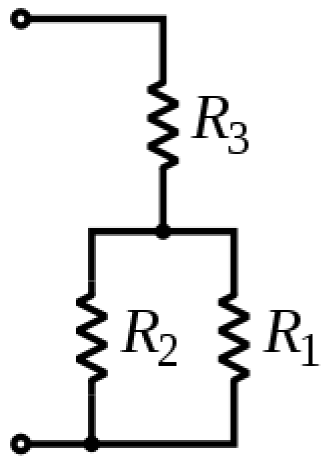
$$\frac{1}{R_{eq}} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_n}$$

The parallel equivalent resistance can be represented in equations by two vertical lines "||" (as in geometry) as a simplified notation. For the case of two resistors in parallel, this can be calculated using:

$$R_{eq} = R_1 \| R_2 = \frac{R_1 R_2}{R_1 + R_2}$$

As a special case, the resistance of N resistors connected in parallel, each of the same resistance R, is given by R/N.

A resistor network that is a combination of parallel and series connections can be broken up into smaller parts that are either one or the other. For instance,

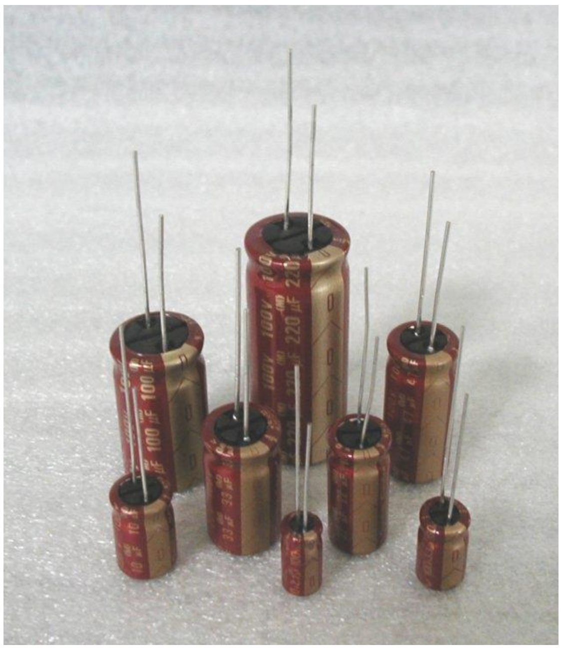


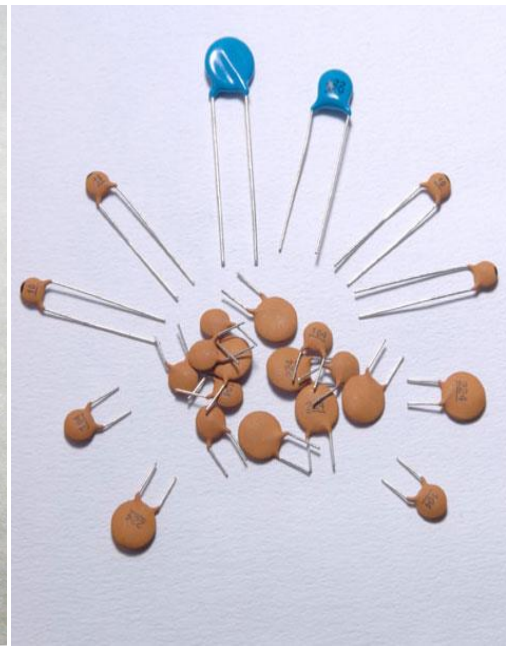
$$R_{\mathrm{eq}} = (R_1 \| R_2) + R_3 = \frac{R_1 R_2}{R_1 + R_2} + R_3$$

However, some complex networks of resistors cannot be resolved in this manner, requiring more sophisticated circuit analysis. For instance, consider a cube, each edge of which has been replaced by a resistor. What then is the resistance that would be measured between two opposite vertices? In the case of 12 equivalent resistors, it can be shown that the corner-to-corner resistance is ⁵⁄₆ of the individual resistance. More generally, the Y-Δ transform, or matrix methods can be used to solve such a problem. One practical application of these relationships is that a non-standard value of resistance can generally be synthesized by connecting a number of standard values in series and/or parallel. This can also be used to obtain a resistance with a higher power rating than that of the individual resistors used. In the special case of N identical resistors all connected in series or all connected in parallel, the power rating of the individual resistors is thereby multiplied by N.

4.11 CAPACITORS

A capacitor or condenser is a passive electronic component consisting of a pair of conductors separated by a dielectric. When a voltage potential difference exists between the conductors, an electric field is present in the dielectric. This field stores energy and produces a mechanical force between the plates. The effect is greatest between wide, flat, parallel, narrowly separated conductors.





An ideal capacitor is characterized by a single constant value, capacitance, which is measured in farads. This is the ratio of the electric charge on each conductor to the potential difference between them. In practice, the dielectric between the plates passes a small amount of leakage current. The conductors and leads introduce an equivalent series resistance and the dielectric has an electric field strength limit resulting in a breakdown voltage.

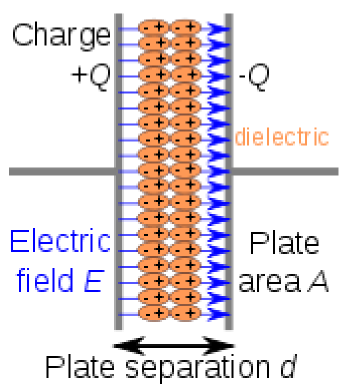
The properties of capacitors in a circuit may determine the resonant frequency and quality factor of a resonant circuit, power dissipation and operating frequency in a digital logic circuit, energy capacity in a high-power system, and many other important aspects.

A capacitor (formerly known as condenser) is a device for storing electric charge. The forms of practical capacitors vary widely, but all contain at least two conductors separated by a non-conductor. Capacitors used as parts of electrical systems, for example, consist of metal foils separated by a layer of insulating film.

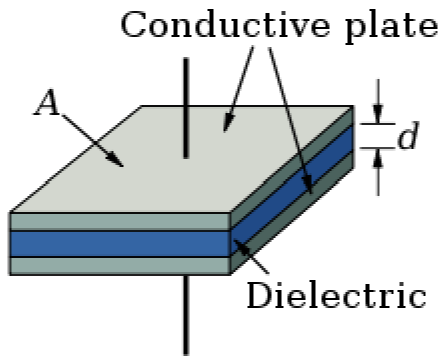
Capacitors are widely used in electronic circuits for blocking direct current while allowing alternating current to pass, in filter networks, for smoothing the output of power supplies, in the resonant circuits that tune radios to particular frequencies and for many other purposes.

A capacitor is a passiveelectronic component consisting of a pair of conductors separated by a dielectric (insulator). When there is a potential difference (voltage) across the conductors, a static electric field develops in the dielectric that stores energy and produces a mechanical force between the conductors. An ideal capacitor is characterized by a single constant value, capacitance, measured in farads. This is the ratio of the electric charge on each conductor to the potential difference between them.

The capacitance is greatest when there is a narrow separation between large areas of conductor, hence capacitor conductors are often called "plates", referring to an early means of construction. In practice the dielectric between the plates passes a small amount of leakage current and also has an electric field strength limit, resulting in a breakdown voltage, while the conductors and leads introduce an undesired inductance and resistance.



Parallel plate model



Dielectric is placed between two conducting plates, each of area A and with a separation of d.

The simplest capacitor consists of two parallel conductive plates separated by a dielectric with permittivityε (such as air). The model may also be used to make qualitative predictions for other device geometries. The plates are considered to extend uniformly over an area A and a charge density $\pm\rho = \pm Q/A$ exists on their surface. Assuming that the width of the plates is much greater than their separation d, the electric field near the centre of the device will be uniform with the magnitude $E = \rho/\varepsilon$. The voltage is defined as the line integral of the electric field between the plates

$$V = \int_0^d E\mathrm{d}z = \int_0^d \frac{\rho}{\varepsilon}\mathrm{d}z = \frac{\rho d}{\varepsilon} = \frac{Qd}{\varepsilon A}.$$

Solving this for $C = Q/V$ reveals that capacitance increases with area and decreases with separation

$$C = \frac{\varepsilon A}{d}.$$

The capacitance is therefore greatest in devices made from materials with a high permittivity.

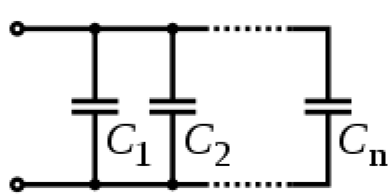
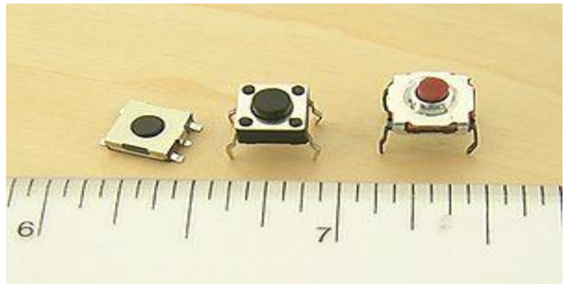


Fig 4.11 Several capacitors in parallel

4.12 PUSH BUTTONS



A push-button (also spelled pushbutton) or simply button is a simple switch mechanism for controlling some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal. The surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed. Buttons are most often biased switches, though even many un-biased buttons (due to their physical nature) require a spring to return to their un-pushed state. Different people use different terms for the "pushing" of the button, such as press, depress, mash, and punch.

Uses:

In industrial and commercial applications push buttons can be linked together by a mechanical linkage so that the act of pushing one button causes the other button to be released. In this way, a stop button can "force" a start button to be released. This method of linkage is used in simple manual operations in which the machine or process have no electrical circuits for control.

Pushbuttons are often color-coded to associate them with their function so that the operator will not push the wrong button in error. Commonly used colors are red for stopping the machine or process and green for starting the machine or process.

Red pushbuttons can also have large heads (mushroom shaped) for easy operation and to facilitate the stopping of a machine. These pushbuttons are called emergency stop buttons and are mandated by the electrical code in many jurisdictions for increased safety. This large mushroom shape can also be found in buttons for use with operators who need to wear gloves for their work and could not actuate a regular flush-mounted push button. As an aid for operators and users in industrial or commercial applications, a pilot light is commonly added to draw the attention of the user and to provide feedback if the button is pushed. Typically this light is included into the center of the pushbutton and a lens replaces the pushbutton hard center disk.

The source of the energy to illuminate the light is not directly tied to the contacts on the back of the pushbutton but to the action the pushbutton controls. In this way a start button when pushed will cause the process or machine operation to be started and a secondary contact designed into the operation or process will close to turn on the pilot light and signify the action of pushing the button caused the resultant process or action to start.

In popular culture, the phrase "the button" refers to a (usually fictional) button that a military or government leader could press to launch nuclear weapons.

Push to ON button:

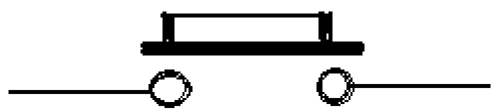


Fig 4.12: push ON button

Initially the two contacts of the button are open. When the button is pressed they become connected. This makes the switching operation using the push button.

4.13 SERVO MOTOR

Servo motors are specially designed motors to be used in control applications and robotics. They are used for precise position and speed control at high torques. It consists of a suitable motor, position sensor and a sophisticated controller. Servo motors can be characterized according the motor controlled by servomechanism, i.e. if DC motor is controlled using servomechanism, it is called as DC Servo motor. Thus major **types of Servo motor** may be - (i) DC Servo motor, (ii) AC Servo motor.

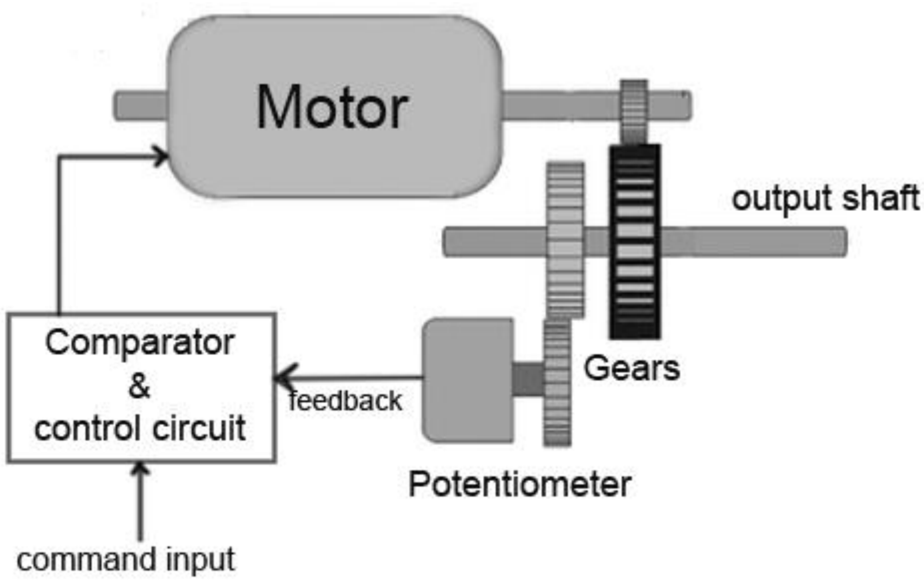
Fig 4.13 Servo Motor

Servo motors are available in power ratings from fraction of watt upto few 100 watts. They are having high torque capabilities. The **rotor of servo motor** is made smaller in diameter and longer in length, so that it has low inertia.

What Is Servomechanism?

Servomechanism is basically a closed-loop system, consisting of a controlled device, controller, output sensor and feedback system. The term servomechanism most probably applies to the systems where position and speed is to be controlled.

How Does A Servo Motor Work?



Servo motors are used to control position and speed very precisely, but in a simple case, only position may be controlled. Mechanical position of the shaft can be sensed by using a potentiometer, which is coupled with the motor shaft through gears. The current position of the shaft is converted into electrical signal by the potentiometer, and the compared with the command input signal. In modern servo motors, electronic encoders or sensors are used to sense the position of the shaft.

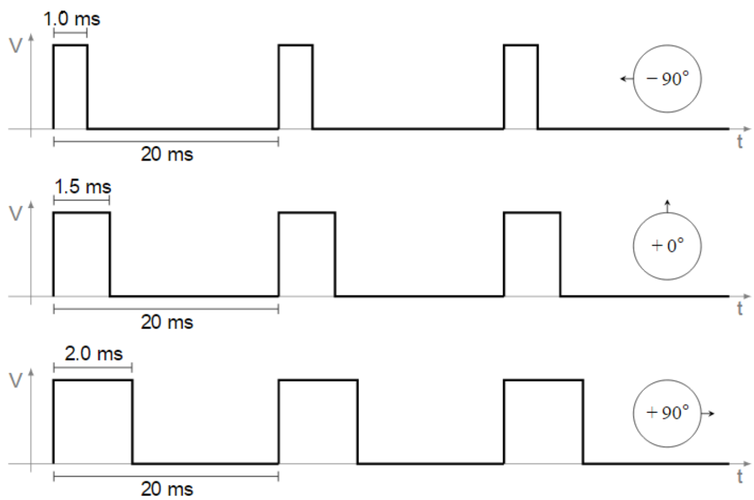
Command input is given according to the required position of the shaft. If the feedback signal differs from the given input, an error signal is generated. This error signal is then amplified and applied as the input to the motor, which causes the motor to rotate. And when the shaft reaches to the required position, error signal becomes zero, and hence the motor stays standstill holding the position.

The command input is given in the form of electrical pulses. As the actual input applied to the motor is the difference between feedback signal (current position) and applied signal (required position), speed of the motor is proportional to the difference between the current position and the required position. The amount of power required by the motor is proportional to the distance it needs to travel.

How Is The Servo Motor Controlled?

Usually a servomotor turns 90° in either direction, i.e. maximum movement can be 180°. A normal servo motor cannot rotate any further due to a built-in mechanical stop.

Three wires are taken out of a servo: positive, ground and control wire. A servo motor is controlled by sending a Pulse Width Modulated (PWM) signal through the control wire. A pulse is sent every 20 milliseconds. Width of the pulses determine the position of the shaft. For example, a pulse of 1ms will move the shaft anticlockwise at -90°, a pulse of 1.5ms will move the shaft at the neutral position that 0° and a pulse of 2ms will move the shaft clockwise at +90°.



When servo motor is commanded to move by applying pulses of appropriate width, the shaft moves to and holds the required position. If an external force is trying to change the position of the shaft, the motor resists to change. Pulses need to be repeated for the motor to hold the position.

Applications Of Servo Motor

Servo motors are popularly used in robotics, computers, CD/DVD players, toys etc. Servos are extensively used in those application where a specific task is to be done repeatedly in a very precise manner.

5. SOFTWARE REQUIREMENTS

5.1 INTRODUCTION TO KEIL MICRO VISION (IDE)

Keil an ARM Company makes C compilers, macro assemblers, real-time kernels, debuggers, simulators, integrated environments, evaluation boards, and emulators for ARM7/ARM9/Cortex-M3, XC16x/C16x/ST10, 251, and 8051 MCU families.

Keil development tools for the 8051 Microcontroller Architecture support every level of software developer from the professional applications engineer to the student just learning about embedded software development. When starting a new project, simply select the microcontroller you use from the Device Database and the µVision IDE sets all compiler, assembler, linker, and memory options for you.

Keil is a cross compiler. So first we have to understand the concept of compilers and cross compilers. After then we shall learn how to work with keil.

5.2 CONCEPT OF COMPILER

Compilers are programs used to convert a High Level Language to object code. Desktop compilers produce an output object code for the underlying microprocessor, but not for other microprocessors. I.E the programs written in one of the HLL like 'C' will compile the code to run on the system for a particular processor like x86 (underlying microprocessor in the computer). For example compilers for Dos platform is different from the Compilers for Unix platform So if one wants to define a compiler then compiler is a program that translates source code into object code.

The compiler derives its name from the way it works, looking at the entire piece of source code and collecting and reorganizing the instruction. See there is a bit little difference between compiler and an interpreter. Interpreter just interprets whole program at a time while compiler analyses and execute each line of source code in succession, without looking at the entire program.

The advantage of interpreters is that they can execute a program immediately. Secondly programs produced by compilers run much faster than the same programs executed by an interpreter. However compilers require some time before an executable program emerges. Now as compilers translate source code into object code, which is unique for each type of computer, many compilers are available for the same language.

5.3 CONCEPT OF CROSS COMPILER

A cross compiler is similar to the compilers but we write a program for the target processor (like 8051 and its derivatives) on the host processors (like computer of x86). It means being in one environment you are writing a code for another environment is called cross development. And the compiler used for cross development is called cross compiler. So the definition of cross compiler is a compiler that runs on one computer but produces object code for a different type of computer.

5.4 KEIL C CROSS COMPILER

Keil is a German based Software development company. It provides several development tools like

- IDE (Integrated Development environment)
- Project Manager
- Simulator
- Debugger
- C Cross Compiler, Cross Assembler, Locator/Linker

The Keil ARM tool kit includes three main tools, assembler, compiler and linker. An assembler is used to assemble the ARM assembly program. A compiler is used to compile the C source code into an object file. A linker is used to create an absolute object module suitable for our in-circuit emulator.

5.5 Building an Application in µVision2

To build (compile, assemble, and link) an application in µVision2, you must:

1. Select Project -(forexample,166\EXAMPLES\HELLO\HELLO.UV2).
2. Select Project - Rebuild all target files or Build target.µVision2 compiles, assembles, and links the files in your project.

5.6 CREATING YOUR OWN APPLICATION IN µVISION2

To create a new project in µVision2, you must:

1. Select Project - New Project.
2. Select a directory and enter the name of the project file.
3. Select Project - Select Device and select an 8051, 251, or C16x/ST10 device from the Device Database™.
4. Create source files to add to the project.
5. Select Project - Targets, Groups, Files. Add/Files, select Source Group1, and add the source files to the project.
6. Select Project - Options and set the tool options. Note when you select the target device from the Device Database™ all special options are set automatically. You typically only need to configure the memory map of your target hardware. Default memory model settings are optimal for most applications.
7. Select Project - Rebuild all target files or Build target.

5.7 DEBUGGING AN APPLICATION IN µVISION2

To debug an application created using µVision2, you must:

1. Select Debug - Start/Stop Debug Session.
2. Use the Step toolbar buttons to single-step through your program. You may enter G, main in the Output Window to execute to the main C function.
3. Open the Serial Window using the Serial #1 button on the toolbar.

Debug your program using standard options like Step, Go, Break, and so on.

5.8 STARTING µVISION2 AND CREATING A PROJECT

µVision2 is a standard Windows application and started by clicking on the program icon. To create a new project file select from the µVision2 menu Project – New Project…. This opens a standard Windows dialog that asks you for the new project file name. We suggest that you use a separate folder for each project. You can simply use the icon Create New Folder in this dialog to get a new empty folder. Then select this folder and enter the file name for the new project, i.e. Project1. µVision2 creates a new project file with the name PROJECT1.UV2 which contains a default target and file group name. You can see these names in the Project.

5.9 WINDOW – FILES.

Now use from the menu Project – Select Device for Target and select a CPU for your project. The Select Device dialog box shows the µVision2 device data base. Just select the microcontroller you use. We are using for our examples the Philips 80C51RD+ CPU. This selection sets necessary tool Options for the 80C51RD+ device and simplifies in this way the tool Configuration.

5.10 BUILDING PROJECTS AND CREATING A HEX FILES

Typical, the tool settings under Options – Target are all you need to start a new application. You may translate all source files and line the application with a click on the Build Target toolbar icon. When you build an application with syntax errors, µVision2 will display errors and warning messages in the Output Window – Build page. A double click on a message line opens the source file on the correct location in a µVision2 editor window. Once you have successfully generated your application you can start debugging.

After you have tested your application, it is required to create an Intel HEX file to download the software into an EPROM programmer or simulator. µVision2 creates HEX files with each build process when Create HEX files under Options for Target – Output is enabled. You may start your PROM programming utility after the make process when you specify the program under the option Run User Program #1.

5.11 CPU SIMULATION

µVision2 simulates up to 16 Mbytes of memory from which areas can be mapped for read, write, or code execution access. The µVision2 simulator traps
and reports illegal memory accesses. In addition to memory mapping, the simulator also provides support for the integrated peripherals of the various 8051 derivatives. The on-chip peripherals of the CPU you have selected are configured from the Device.

5.12 DATABASE SELECTION

You have made when you create your project target. Refer to page 58 for more Information about selecting a device. You may select and display the on-chip peripheral components using the Debug menu. You can also change the aspects of each peripheral using the controls in the dialog boxes.

5.13 START DEBUGGING

You start the debug mode of µVision2 with the Debug – Start/Stop Debug Session Command. Depending on the Options for Target – Debug Configuration, µVision2 will load the application program and run the startup code µVision2 saves the editor screen layout and restores the screen layout of the last debug session. If the program execution stops, µVision2 opens an editor window with the source text or shows CPU instructions in the disassembly window. The next executable statement is marked with a yellow arrow. During debugging, most editor features are still available.

For example, you can use the find command or correct program errors. Program source text of your application is shown in the same windows. The µVision2 debug mode differs from the edit mode in the following aspects:

_ The "Debug Menu and Debug Commands" described on page 28 are available. The additional debug windows are discussed in the following.

_ The project structure or tool parameters cannot be modified. All build commands are disabled.

5.14 DISASSEMBLY WINDOW

The Disassembly window shows your target program as mixed source and assembly program or just assembly code. A trace history of previously executed instructions may be displayed with Debug – View Trace Records. To enable the trace history, set Debug – Enable/Disable Trace Recording.

If you select the Disassembly Window as the active window all program step commands work on CPU instruction level rather than program source lines. You can select a text line and set or modify code breakpoints using toolbar buttons or the context menu commands.

You may use the dialog Debug – Inline Assembly… to modify the CPU instructions. That allows you to correct mistakes or to make temporary changes to the target program you are debugging. Numerous example programs are included to help you get started with the most popular embedded 8051 devices.

The Keil µVision Debugger accurately simulates on-chip peripherals (I^2C, CAN, UART, SPI, Interrupts, I/O Ports, A/D Converter, D/A Converter, and PWM Modules) of your 8051 device. Simulation helps you understand hardware configurations and avoids time wasted on setup problems. Additionally, with simulation, you can write and test applications before target hardware is available.

5.15 EMBEDDED C

Use of embedded processors in passenger cars, mobile phones, medical equipment, aerospace systems and defense systems is widespread, and even everyday domestic appliances such as dish washers, televisions, washing machines and video recorders now include at least one such device.

Because most embedded projects have severe cost constraints, they tend to use low-cost processors like the 8051 family of devices considered in this book. These popular chips have very limited resources available most such devices have around 256 bytes (not megabytes!) of RAM, and the available processor power is around 1000 times less than that of a desktop processor. As a result, developing embedded software presents significant new challenges, even for experienced desktop programmers. If you have some programming experience - in C, C++ or Java - then this book and its accompanying CD will help make your move to the embedded world as quick and painless as possible.

6. SCHEMATIC DIAGRAM

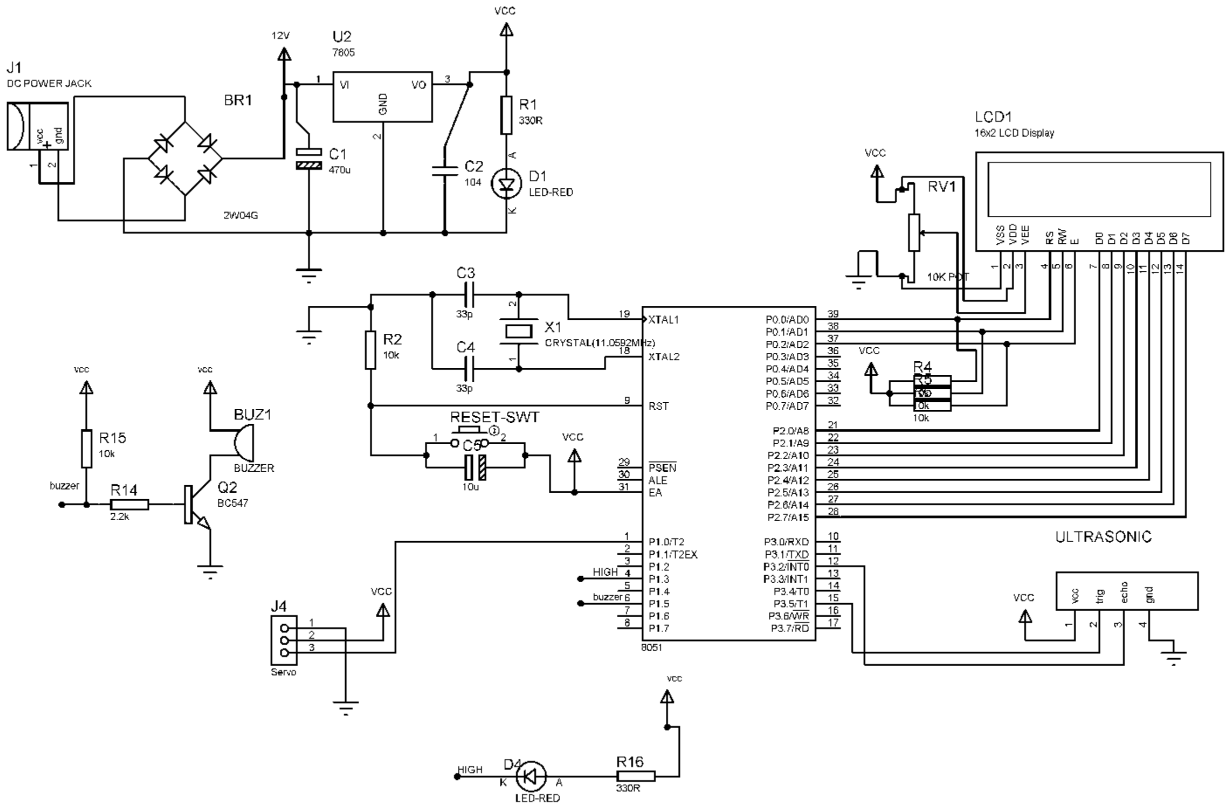


Fig (6) Schematic Diagram

7. PROJECT DESCRIPTION

POWER SUPPLY

The circuit uses standard power supply comprising of a step-down transformer from 230Vto 12V and 4 diodes forming a bridge rectifier that delivers pulsating dc which is then filtered by an electrolytic capacitor of about 470µF to 1000µF. The filtered dc being unregulated, IC LM7805 is used to get 5V DC constant at its pin no 3 irrespective of input DC varying from 7V to 15V. The input dc shall be varying in the event of input ac at 230volts section varies from 160V to 270V in the ratio of the transformer primary voltage V1 to secondary voltage V2 governed by the formula V1/V2=N1/N2. As N1/N2 i.e. no. of turns in the primary to the no. of turns in the secondary remains unchanged V2 is directly proportional to V1.Thus if the transformer delivers 12V at 220V input it will give 8.72V at 160V.Similarly at 270V it will give 14.72V.Thus the dc voltage at the input of the regulator changes from about 8V to 15V because of A.C voltage variation from 160V to 270V the regulator output will remain constant at 5V.

The regulated 5V DC is further filtered by a small electrolytic capacitor of 10µF for any noise so generated by the circuit. One LED is connected of this 5V point in series with a current limiting resistor of 330Ω to the ground i.e., negative voltage to indicate 5V power supply availability. The unregulated 12V point is used for other applications as and when required.

STANDARD CONNECTIONS TO 8051 SERIES MICRO CONTROLLER

ATMEL series of 8051 family of micro controllers need certain standard connections. The actual number of the Microcontroller could be “89C51” , “89C52”, “89S51”, “89S52”, andas regards to 20 pin configuration a number of “89C2051”. The 4 set of I/O ports are used based on the project requirement. Every microcontroller requires a timing reference for its internal program execution therefore an oscillator needs to be functional with a desired frequency to obtain the timing reference as t =1/f.

A crystal ranging from 2 to 20 MHz is required to be used at its pin number 18 and 19 for the internal oscillator. It may be noted here the crystal is not to be understood as crystal oscillator It is just a crystal, while connected to the appropriate pin of the microcontroller it results in oscillator function inside the microcontroller. Typically 11.0592 MHz crystal is used in general for most of the circuits using 8051 series microcontroller. Two small value ceramic capacitors of 33pF each is used as a standard connection for the crystal as shown in the circuit diagram.

RESET

Pin no 9 is provided with an resset arrangement by a combination of an electrolytic capacitor and a register forming RC time constant. At the time of switch on, the capacitor gets charged, and it behaves as a full short circuit from the positive to the pin number 9.After the capacitor gets fully charged the current stops flowing and pin number 9 goes low which is pulled down by a 10k resistor to the ground. This arrangement of reset at pin 9 going high initially and then to logic 0 i.e., low helps the program execution to start from the beginning. In absence of this the program execution could have taken place arbitrarily anywhere from the program cycle.A pushbutton switch is connected across the capacitor so that at any given time as desired it can be pressed such that it discharges the capacitor and while released the capacitor starts charging again and then pin number 9 goes to high and then back to low, to enable the program execution from the beginning. This operation of high to low of the reset pin takes place in fraction of a second as decided by the time constant R and C.

For example: A 10µF capacitor and a 10kΩ resistor would render a 100ms time to pin number 9 from logic high to low, there after the pin number 9 remains low.

External Access (EA):

Pin no 31 of 40 pin 8051 microcontroller termed as $\overline{EA}$ is required to be connected to 5V for accessing the program form the on-chip program memory. If it is connected to ground then the controller accesses the program from external memory. However as we are using the internal memory it is always connected to +5V.

7.1 OPERATION EXPLANATION

CONNECTIONS:

The output of the power supply which is 5v is connected to the 40^{th} Pin of microcontroller and GND is connected to its 20^{th} Pin. Pin 21- 28 of port2 of micro controller is connected to D0-D7 data pins of LCD. Pin10 of Port3.0 of microcontroller is connected to ultrasonic module pin no.2. Pin 15, 16, 17 of Port0 of microcontroller is connected to RS, RW, EN pins of LCD.

WORKING:

The project uses an ultrasonic sensor module comprising of 1 transmitter and 1 receiver. The transmitter can deliver 40 KHz ultrasonic sound while the maximum receiver is designed to accept only 40 KHz sound waves. The receiver ultrasonic sensor that is kept next to the transmitter shall thus be able to receive reflected 40 KHz, once the module faces any obstacle in front. Thus whenever any obstacles come ahead of the ultrasonic module it calculates the time taken from sending the signals to receiving them since time and distance are related for sound waves passing through air medium at 343.2m/sec. Upon receiving the signal MC program while executed displays the data i.e. the distance measured on a 16X2 LCD interfaced to the microcontroller in cms.

7.2. LAYOUT DIAGRAM

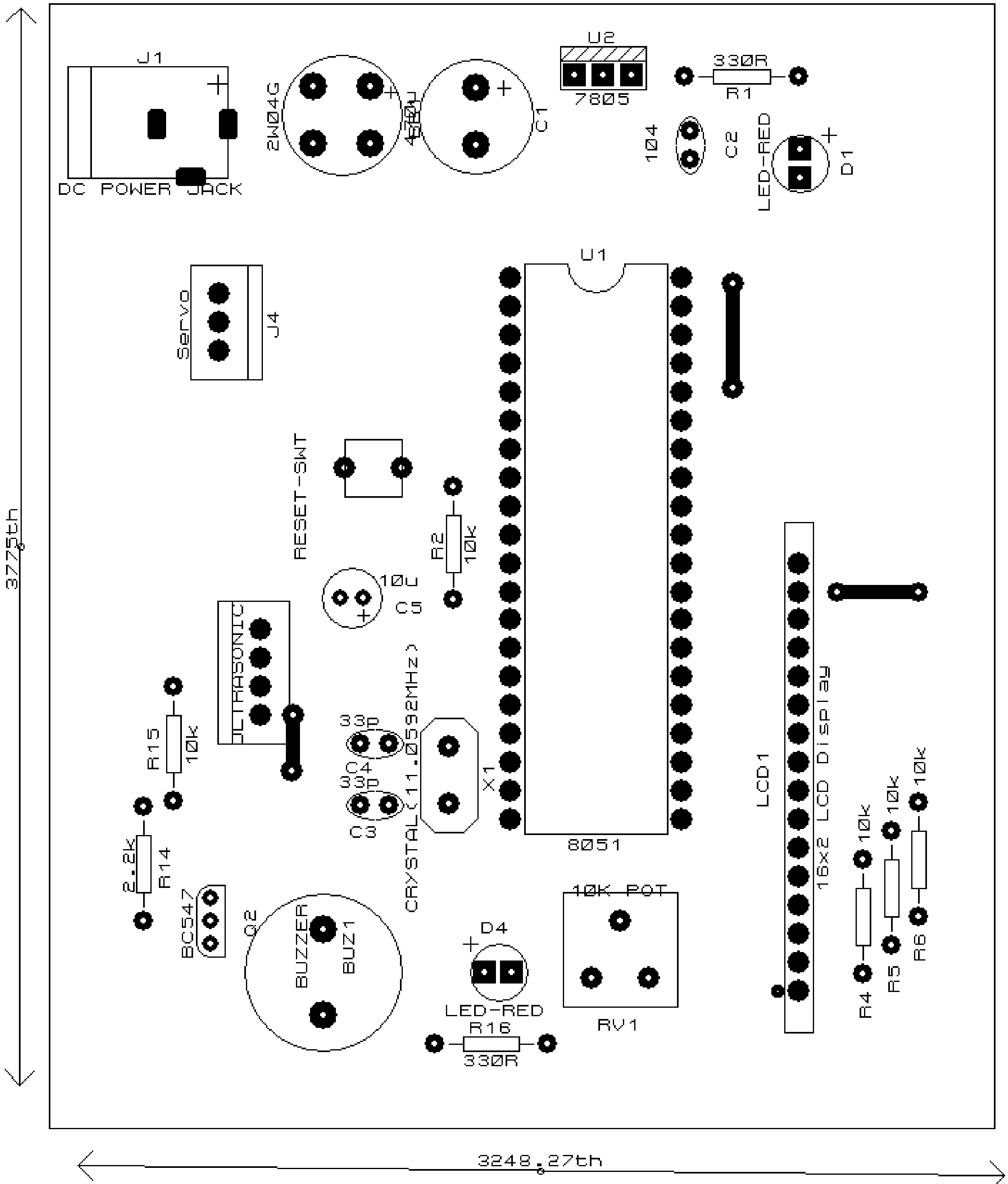


Fig. 7.2 Layout Diagram

8.CODING

```
# include <at89x52.h>
# include <string.h>
#include <intrins.h>
# define LCDPort P2

sbit EN=P3^4;
sbit RW=P3^6;
sbit RS=P3^7;
sfr16 DPTR =0x82;
sbit triger=P3^5;
sbit echo=P3^2;
sbit servo=P1^0;
sbit led=P1^3;
sbit buzzer=P1^5;

unsigned char angle=10,x,dis_angle=0;
unsigned int
ch,high_byte,low_byte,high_byte,distance,dis3='0',dis2='0',dis1='0',dis0='0',
dis4='.',dis5='0';
unsigned int target_range=0,range=0,d=0;
unsigned int s=0;
bit angle_high=0,angle_low=1,sm=0;
void init();
void init_display();
void delay(unsigned int delay_ms);
void lcd_cmd(unsigned char cmd);
void lcd_init();
void lcd_string(char *str);
void lcd_data(unsigned char data1);
void display();
void get_range();
void Delay_servo (unsigned int a);
void display2();
void timer(int msec);
void display_data();

void main()
{
unsigned int i;
//EA = 1;
led=0;
buzzer=0;
      led=1;
buzzer=0;
init();
triger=0;
echo=1;
triger=1;
delay(10);
triger=0;
delay(41);
lcd_cmd(0x38);
  lcd_cmd(0x0c);
  delay(2);

  delay(20);
  TMOD|=0x09;//timer0 in 16 bit mode with gate enable
  TR0=1;//timer run enabled
```

```
  TH0=0x00;TL0=0x00;

  P3=0x04;

servo=0;
do
  {
    //Turn to 0 degree
    servo = 1;
    timer(10);
    servo = 0;
            delay(10);
            i++;
      }while(i<200);

while(1)
{
if(angle_low==1)
{angle++;
dis_angle=dis_angle+2;
}
if(angle_high==1)
{angle--;
dis_angle=dis_angle-2;
}
get_range();
if(range<=20)
{led=0;
buzzer=1;}
else
{led=1;
buzzer=0;}
display();
lcd_cmd(0x80);
lcd_string("Object At");
servo=1;
timer(angle);
servo=0;
timer(360);
lcd_cmd(0x80);
      s=angle;
 x=s%10;
 dis5=x+0x30;
 dis3='.';
 s=s/10;
 x=s%10;
 dis4=x+0x30;
 s=s/10;
 x=s%10;
 dis2=x+0x30;
 s=s/10;
 x=d%10;
 dis1=x+0x30;
 s=s/10;
 x=s%10;
 dis0=x+0x30;
lcd_cmd(0xc0);
lcd_string("Angle");
//lcd_data(dis2);
//lcd_data(dis4);
//lcd_data(dis5);
s=dis_angle;
x=s%10;
```

```
 dis5=x+0x30;
 dis3='.';
 s=s/10;
 x=s%10;
 dis4=x+0x30;
 s=s/10;
 x=s%10;
 dis2=x+0x30;
lcd_cmd(0xcb);
lcd_data(dis2);
lcd_data(dis4);
lcd_data(dis5);

if(angle<46 && angle_low==1)
{angle_low=1;
angle_high=0;}

if(angle==10)
{angle_low=1;
angle_high=0;
angle=10;
dis_angle=0;}



if(angle==46)
{angle_low=0;
angle_high=1;}
delay(1000);
}
}
void init()
{
      lcd_init();
      init_display();
      TMOD|=0X09;
      TR0=1;
      TH0=0X00;
      TL0=0X00;
}


void init_display()
      {
            lcd_cmd(0x83);
            lcd_string("ULTRASONIC");
            //lcd_string("DISTANCE");
            lcd_cmd(0xc5);
            lcd_string("RADAR ");
            delay(4000);
            lcd_cmd(0x01);
            delay(10);
      }


//void init_timer1()
//{
//    TMOD|=0X20;
//    TH1=0XFD;
//    SCON=0X30;
//    TR1=1;
//}
```

```
void delay(unsigned int delay_ms)
{
      unsigned int count,i;

      for(count = 0;count<delay_ms;count++)
      {
            for(i =0;i < 120;i++);
      }
      delay_ms--;
}

void lcd_cmd(unsigned char cmd)
      {
            RS=0;
            RW=0;
            LCDPort=cmd;
            EN=1;
            delay(2);
            EN=0;
      }

void lcd_data(unsigned char data1)
      {
            RS=1;
            RW=0;
            LCDPort=data1;
            EN=1;
            delay(2);
            EN=0;
      }

void lcd_string(char *str)
{
      int i;
      for(i=0;str[i]!='\0';i++)

      lcd_data(str[i]);

}

void lcd_init()
      {
            lcd_cmd(0x01);
            delay(10);
            lcd_cmd(0x06);
            delay(10);
            lcd_cmd(0x0c);
            delay(10);
            lcd_cmd(0x38);
            delay(10);
      }

void display()
{
      d=range;
 x=d%10;
 dis5=x+0x30;
 dis3='.';
 d=d/10;
 x=d%10;
 dis4=x+0x30;
 d=d/10;
```

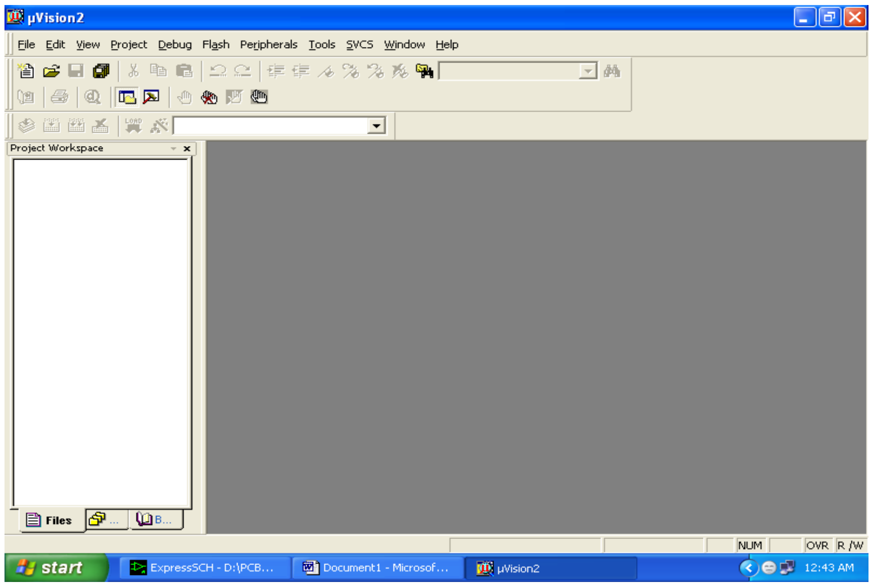
```
 x=d%10;
 dis2=x+0x30;
 d=d/10;
 x=d%10;
 dis1=x+0x30;
 d=d/10;
 x=d%10;
 dis0=x+0x30;
lcd_cmd(0x8a);
lcd_data(dis2);
lcd_data(dis4);
lcd_data(dis5);
lcd_string(" ");
lcd_string("c");
lcd_string("m");
}


void get_range()
{
TH0=0X00;
      TL0=0X00;
      sm=0;
      triger=1;
      lcd_cmd(0x01);
      delay(10);
      triger=0;
      while(!echo);
      while(echo && sm==0)
      {if(TH0==0XFF)
            sm=1;
      }
      //while(echo);
////while(echo || sm==0)
//    {
//    unsigned int count,i,delay_ms=800;
//    lcd_cmd(0xc5);
//          lcd_string("WW2");sm=1;
//    for(count = 0;count<delay_ms;count++)
//    {
//          for(i =0;i < 120;i++)
//          {if(echo==0)
//                {count=800;
//                      i=120;
//                sm=1;}
//          }
//          sm=1;
//    }
//    sm=1;
//    delay_ms--;
//    }
      DPH=TH0;
      DPL=TL0;
      TH0=0XFF;
      TL0=0XFF;
      if(DPTR<35000)
      range=DPTR/59;
      else
      range=0;
}
```

```
void timer(int msec)     // Function for timer
{
int i;
for(i=0;i<msec;i++)
{
TMOD|=0x20;          // Mode2
//TH1=0xFF;
//TL1=0xD2;
TH1= 50;
TR1=1;
while(TF1==0);
TF1=0;
TR1=0;
}
}
```

8.1 PROGRAMMING OF THE CODE

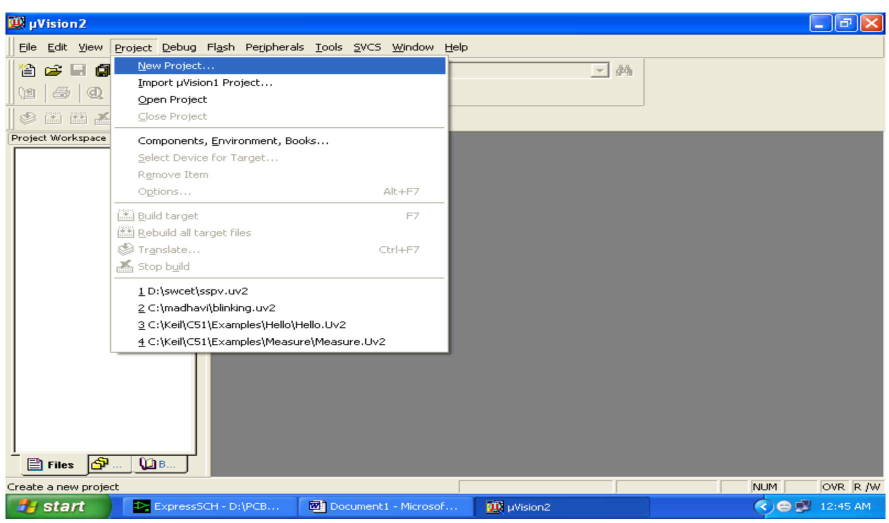
1. Click on the Keil Vision Icon on Desktop

2. The following fig will appear

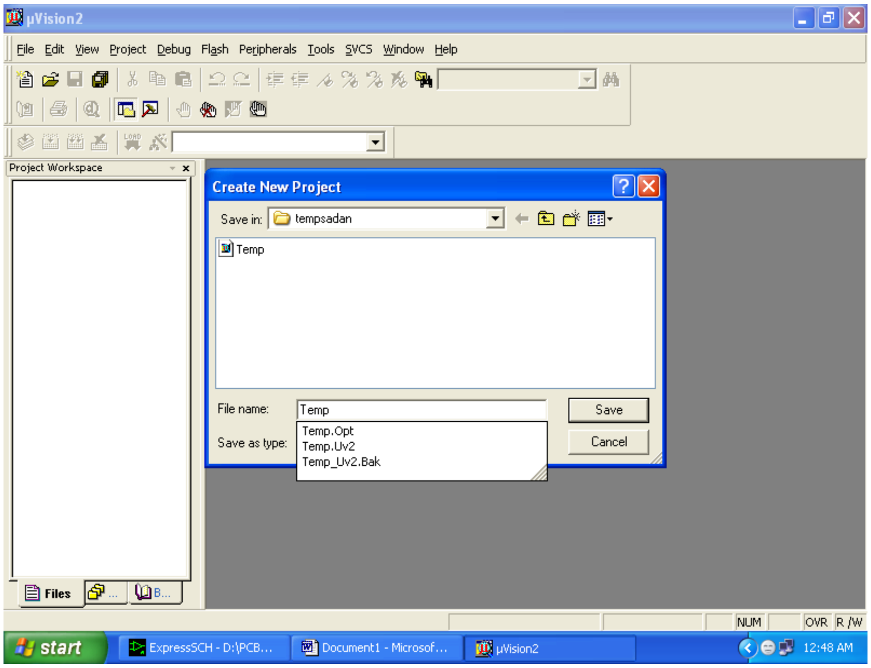


3. Click on the Project menu from the title bar

4. Then Click on New Project



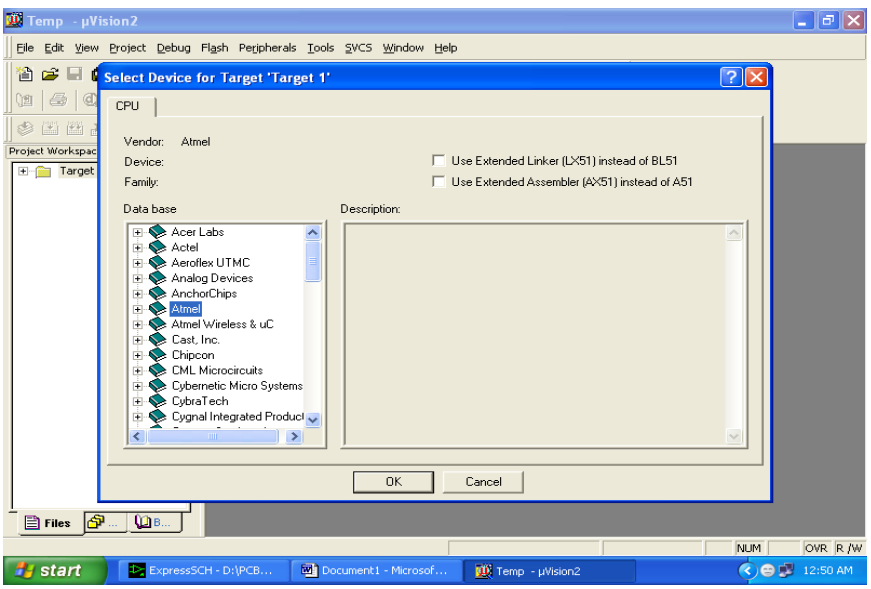
5. Save the Project by typing suitable project name with no extension in u r own folder sited in either C:\ or D:\



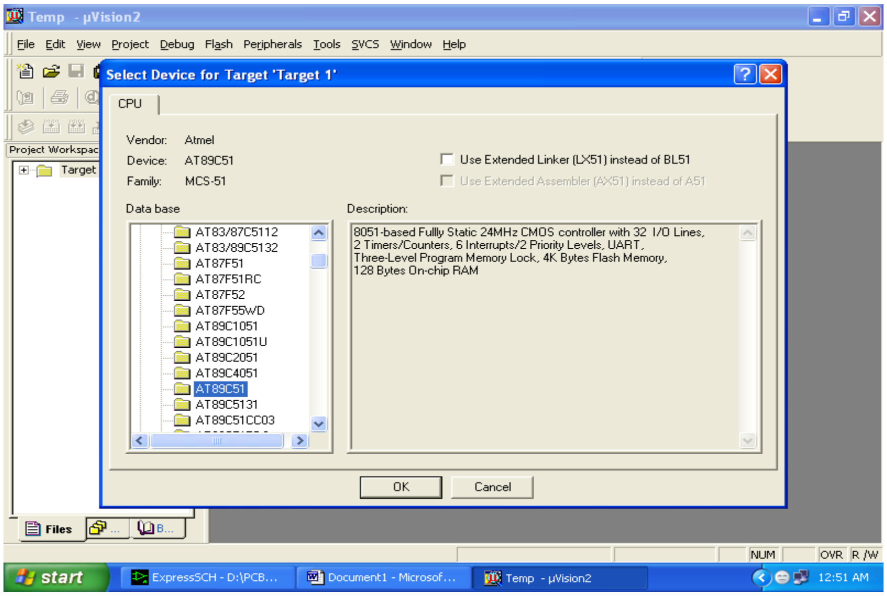
6. Then Click on Save button above.

7. Select the component for u r project. i.e. Atmel……

8. Click on the + Symbol beside of Atmel

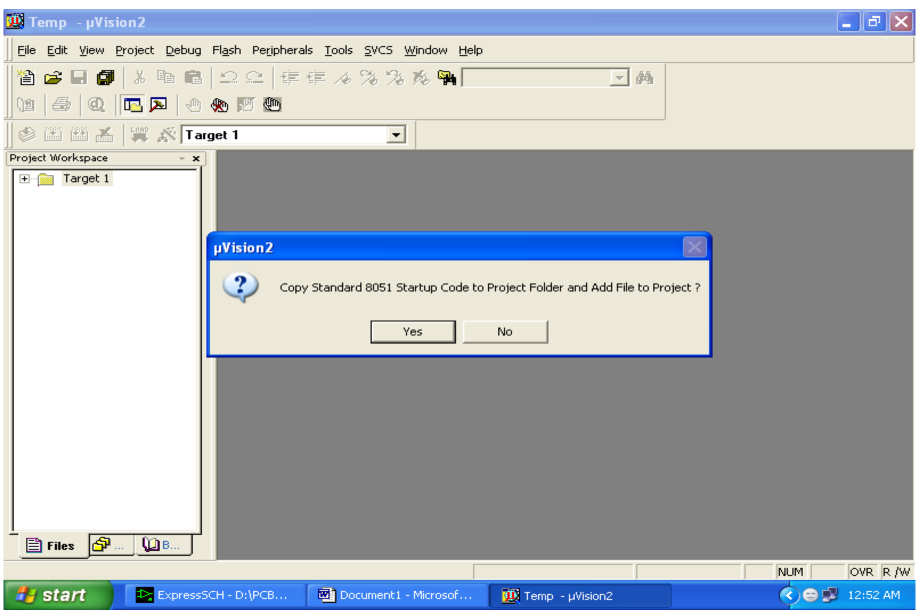


9. Select AT89C51 as shown below



10. Then Click on "OK"

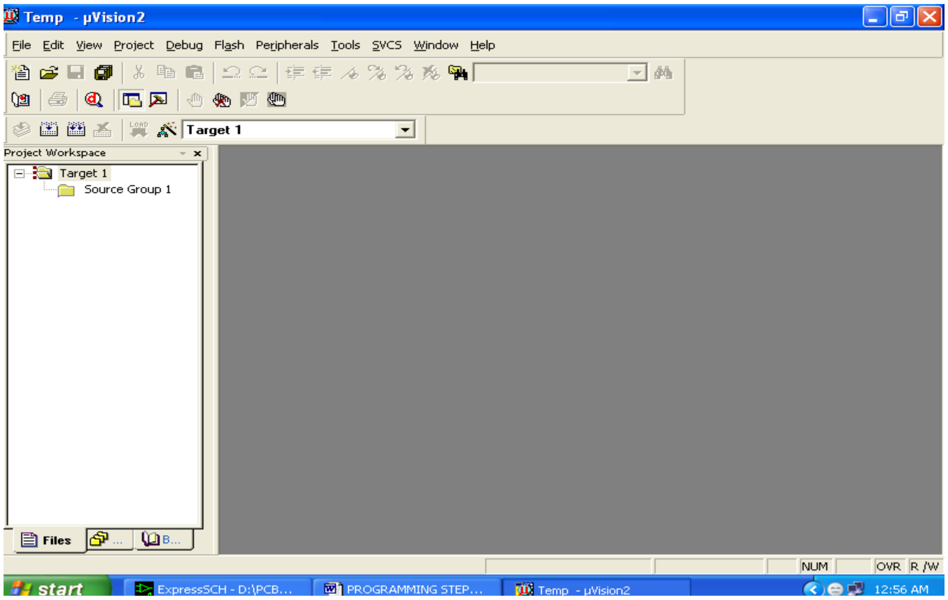
11. The Following fig will appear



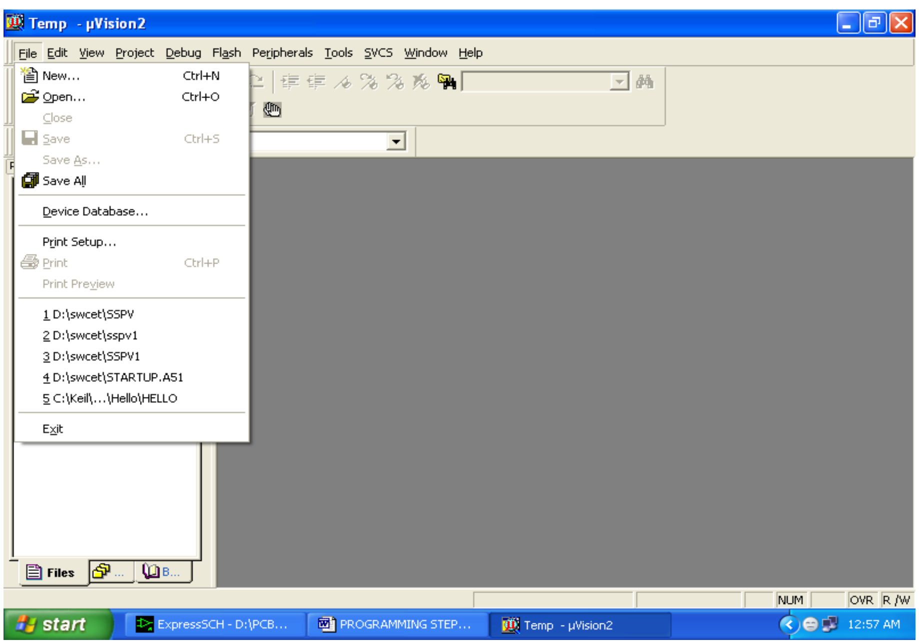
12. Then Click either YES or NO.........mostly "NO".

13. Now your project is ready to USE.

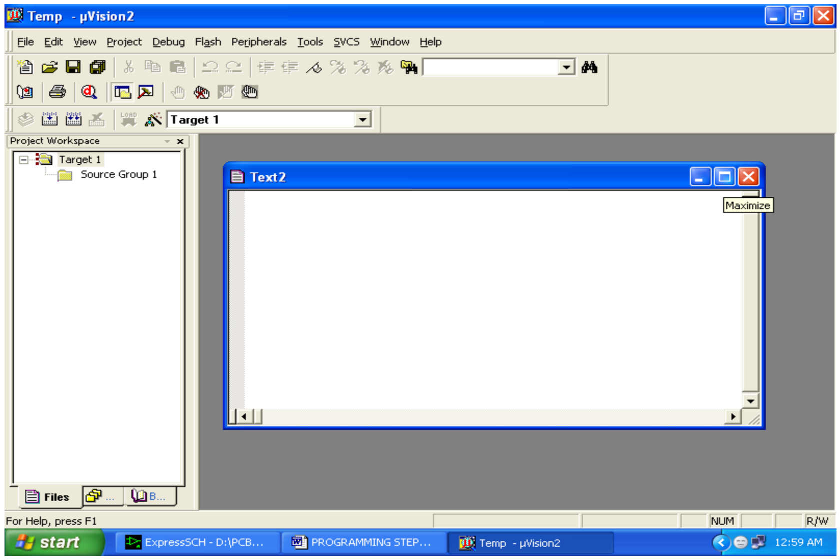
14. Now double click on the Target1, you would get another option "Source group 1" as shown in next page.



15. Click on the file option from menu bar and select "new".

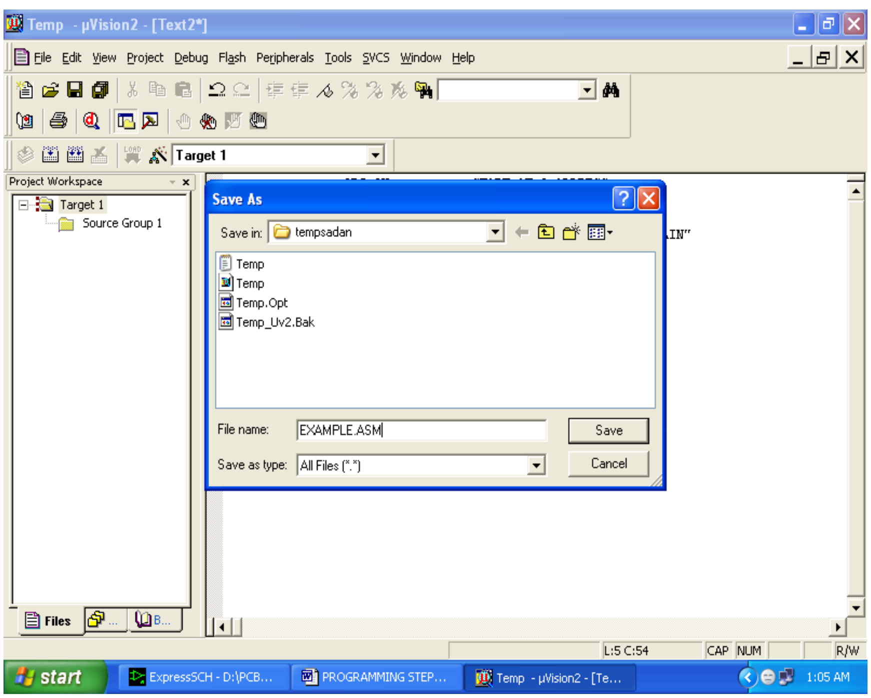


16. The next screen will be as shown in next page, and just maximize it by double clicking on its blue boarder.

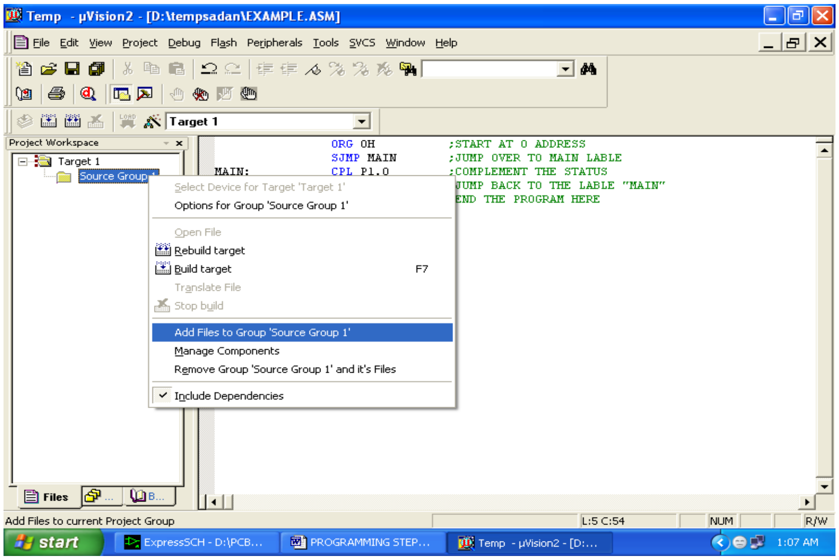


17. Now start writing program in either in "EMBEDDED C" or "ASM".

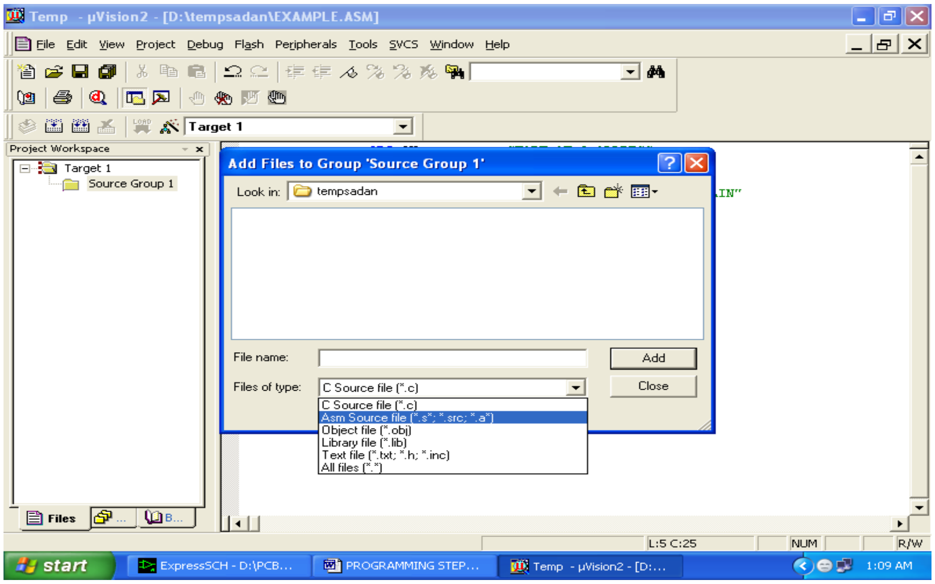
18. For a program written in Assembly, then save it with extension ". asm" and for "EMBEDDED C" based program save it with extension " .C"



19. Now right click on Source group 1 and click on "Add files to Group Source".



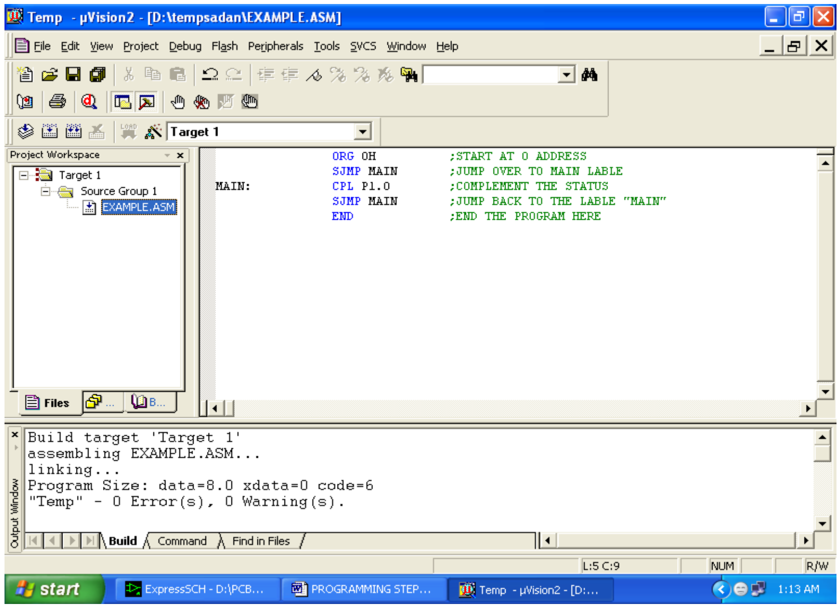
20. Now you will get another window, on which by default "EMBEDDED C" files will appear.



21. Now select as per your file extension given while saving the file

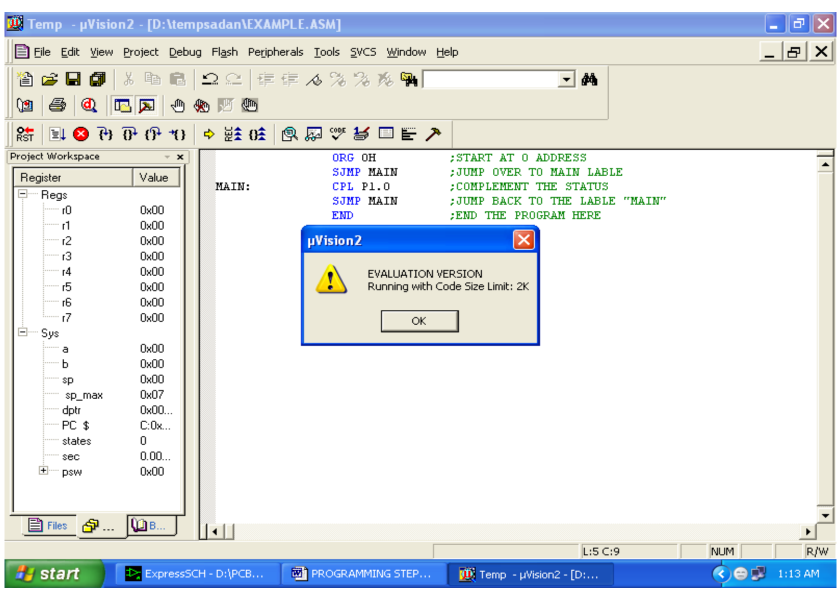
22. Click only one time on option “**ADD**”.

23. Now Press function key F7 to compile. Any error will appear if so happen.



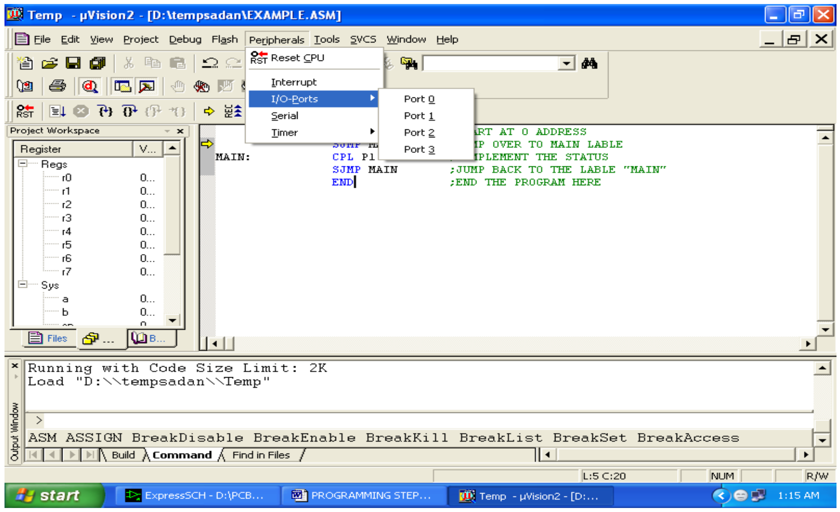
24. If the file contains no error, then press Control+F5 simultaneously.

25. The new window is as follows

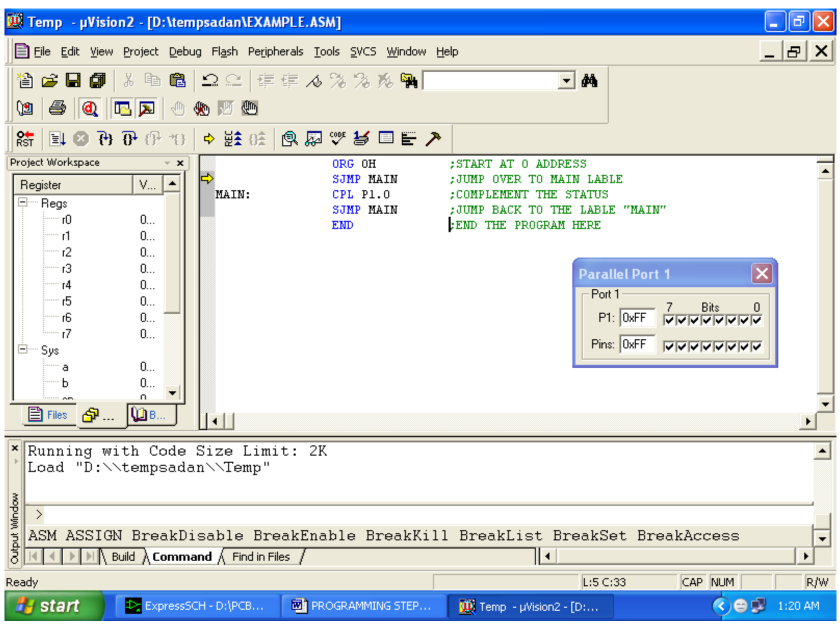


26. Then Click “OK”.

27. Now click on the Peripherals from menu bar, and check your required port as shown in fig below.



28. Drag the port a side and click in the program file.



29. Now keep Pressing function key "F11" slowly and observe.

30. You are running your program successfully.

9.HARDWARE TESTING

9.1 CONTINUITY TEST:

In electronics, a continuity test is the checking of an electric circuit to see if current flows (that it is in fact a complete circuit). A continuity test is performed by placing a small voltage (wired in series with an LED or noise-producing component such as a piezoelectric speaker) across the chosen path. If electron flow is inhibited by broken conductors, damaged components, or excessive resistance, the circuit is "open".

Devices that can be used to perform continuity tests include multi meters which measure current and specialized continuity testers which are cheaper, more basic devices, generally with a simple light bulb that lights up when current flows.

An important application is the continuity test of a bundle of wires so as to find the two ends belonging to a particular one of these wires; there will be a negligible resistance between the "right" ends, and only between the "right" ends.

This test is the performed just after the hardware soldering and configuration has been completed. This test aims at finding any electrical open paths in the circuit after the soldering. Many a times, the electrical continuity in the circuit is lost due to improper soldering, wrong and rough handling of the PCB, improper usage of the soldering iron, component failures and presence of bugs in the circuit diagram. We use a multi meter to perform this test. We keep the multi meter in buzzer mode and connect the ground terminal of the multi meter to the ground. We connect both the terminals across the path that needs to be checked. If there is continuation then you will hear the beep sound.

9.2 POWER ON TEST:

This test is performed to check whether the voltage at different terminals is according to the requirement or not. We take a multi meter and put it in voltage mode. Remember that this test is performed without microcontroller. Firstly, we check the output of the transformer, whether we get the required 12 v AC voltage.

Then we apply this voltage to the power supply circuit. Note that we do this test without microcontroller because if there is any excessive voltage, this may lead to damaging the controller. We check for the input to the voltage regulator i.e., are we getting an input of 12v and an output of 5v. This 5v output is given to the microcontrollers' 40^{th} pin. Hence we check for the voltage level at 40^{th} pin. Similarly, we check for the other terminals for the required voltage. In this way we can assure that the voltage at all the terminals is as per the requirement.

10. RESULT

We have designed short range radar therefore our res was specified and limited. This system can only detect objects from 0 to 180 degrees only because the servo motor that we have used can rotate only to this range. So, our design can be applied to places or areas for obstacle detection on a smaller scale.

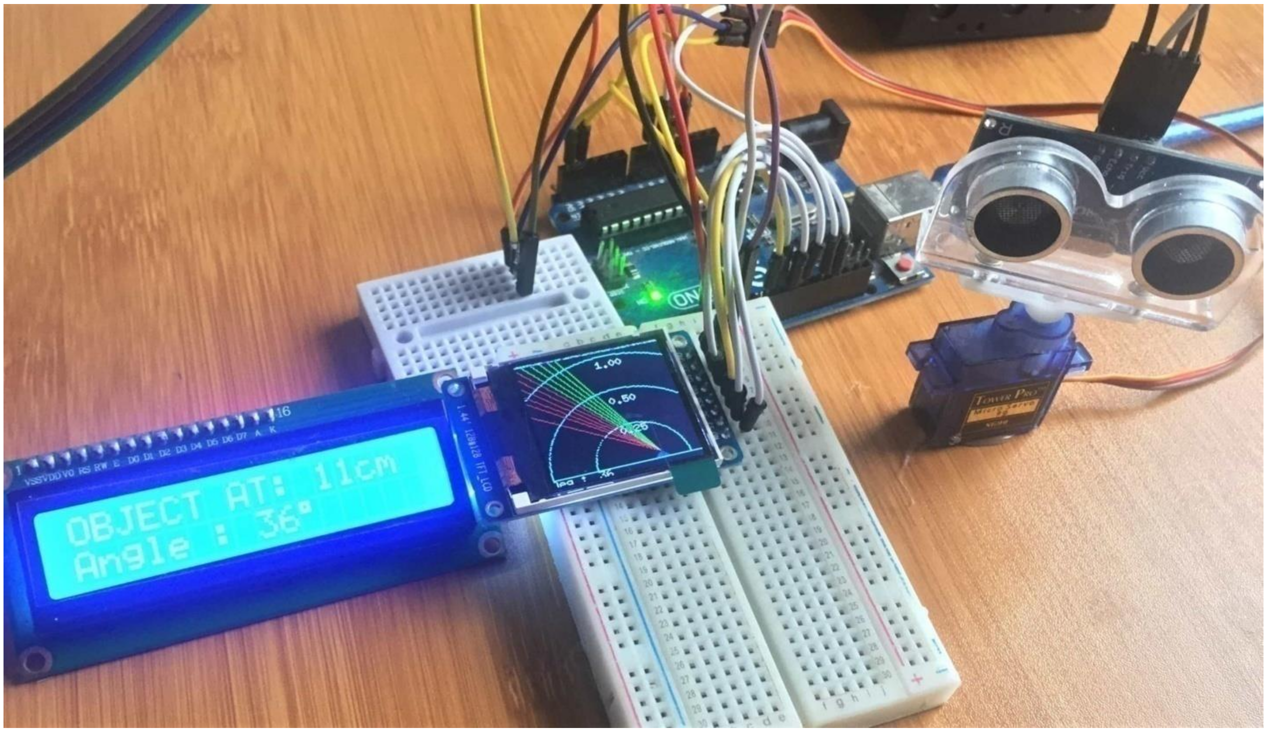


Fig (10) Prototype of the Project

11. CONCLUSION

We have designed a project on Ultrasonic RADAR for security system for human or object interference in a short range. The system has been successfully implemented and the aim is achieved without any deviation. There is a lot of future scope of this project because of its security capacity. It can be used in many applications. This project can also be developed or modified according to the rising needs and demand.

12. BIBLIOGRAPHY

REFERENCES:

1. "The 8051 Microcontroller and Embedded systems" by Muhammad Ali Mazidi and Janice Gillispie Mazidi , Pearson Education.
2. Introduction to Radar Systems by Merrill Skolnik
3. Radar Principles by Peyton Z. Peebles
4. ATMEL 89S52 Data Sheets.

WEBSITES

- www.atmel.com
- www.beyondlogic.org
- www.wikipedia.org
- www.alldatasheets.com