1. INTRODUCTION

1.1 OVERVIEW

Water is one of the most precious resources we have water is major part of our lives and is a natural resource which we cannot regenerate in any industry, lab or in any place. Every living thing needs water to survive. Life without water is nothing. We require tonnes of litres of, to manufacture a material which we mostly use it or not. Only 3% of water is available for drinking purpose. So we should use water sensibly. In present day situation every person from cites to villages want a purified water.

The world's population, homes, industry and draught conditions in many regions, are putting ore pressure on water restrictions. Cities and communities are stepping up enforcement of restricted watering times and amounts to ensure the local water supply is protected. As more water usage is mainly in urban areas than in rural areas as the water is supplied through different distribution systems to each corner of the city. Providing sufficient water of appropriate quality and quantity has been one of the most important issues in human history. Most ancient civilizations were initiated near water sources. As populations grew, the challenge to meet user demands also increased. People began to transport water from other locations to their communities.

1.2 AIM OF THE PROJECT

The overall objective of the project is to deliver wholesome water to the consumer at particular area and in sufficient quantity. It achieves continuity and maximum coverage at affordable cost without any loss. It is done by developing an embedded based remote water monitoring and theft prevention system by taking the data of water supply at the consumer/user end.

1.3 METHODOLOGY

Here we are using AT89S52 as our controller and also few sensors are arranged to detect the presence of water in that particular pipeline. As logic level converters are used to detect the water flow. Water should be released as per the instructions by officials, for example alternate days of supply are provided and only during specific period of time but not daily.

All the details will be shown in the web server using IOT module connected to the controller. So that the authorities can take necessary action in case of misuse. This is an advanced, trouble-free, fit and forget system for water board. By using all these malfunctioning can be avoided.

1.4 SIGNIFICANCE OF THE WORK

To help address water waste from aging infrastructure, local governments and regional water authorities can now tap into an internet of things based water monitoring platform. It offers users, governments a cost-effective way to conserve water. In this project we are stopping utility fraud and leakages with wirelessly connected meter monitoring.

Merits

1. Wireless connectivity is revolutionising several areas of smart metering by providing easier maintenance, billing and support beyond monitoring for utility abuse.
2. Water companies can monitor and control pressure, flow rates levels, detect and locate leakages, and reduces of bursts on the network.
3. Highly sensitive.
4. Fit and Forget system.
5. Low cost and reliable circuit.
6. Complete elimination of manpower.
7. Mobilise resources.

Demerits

1. It needs more circuitry, components to implement for larger water distribution systems.
2. Sensors should be maintained properly.

Applications

1. Water board.
2. Water distribution systems in urban areas, villages, organisations, apartments.

1.5 ORGANISATION OF THESIS

The thesis explains the implementation of “public water supply grid monitoring to avoid water theft and leakages using iot”.

Chapter 1 presents introduction to the overall thesis and the overview of the project.

Chapter 2 literature survey and actual problem for this project to be implemented.

Chapter 3 presents the introduction to embedded systems.

Chapter 4 presents hardware description.

Chapter 5 presents wifi module ESP8266 description.

Chapter 6 presents software and code description.

Chapter 7 presents results of the project.

Chapter 8 presents the conclusion project.

Chapter 9 presents the future scope project.

2. PROBLEMS AND LITERATURE SURVEY

2.1 WATER THEFT

While meter under-registration is more of technical problem, water theft is apolitical and social issue. The reason is that illegal connections are nearly always wrongly associated with only the urban poor and informal settlements. However, water theft by households and commercial users, sometimes even large corporations, often accounts for sizable volumes of water lost and even higher losses of revenue. In addition to illegal connections to other forms of water theft include meter tampering and meter bypasses, meter reader corruptions, and illegal hydrant us. Another common problem is ‘inactive accounts’ in cases where a consumer’s contract has been terminated, the physical service connections, or at least the taping point on the main, still exists and is easy to reconnect illegally. A stringent inactive account management and verification program can easily solve this problem.

2.2 WATER LEAKAGE

Figure 2.1 Water leakage in a pipe.

Leakage means that good quality water produced in a plant is lost. It results in less water being available for supply. It can mean that customers are left without water for some of the time. In the long term, the loss of water due to leakage puts added pressure on the water supply system [1]. Leakage causes other problems such as damaged pipes can allow contamination of the mains supply, leaking water can damage infrastructure such as the foundations of buildings. Leakage occurs due to the pipelines being old and corroded, being poorly constructed and maintained, having poor corrosion protection, etc. Preventing leakage in the existing system rather than investing in expansion or new schemes would be economically preferable.

2.3 LITERATURE SURVEY

The concept of National Water Grid for effective management of flood and drought situations in India has been introduced a long back by a number of eminent persons like Arther Cotton DR. K.L. Rao and DR. Abdul Kalam and many others. Water grid is a region wide, long term, water supply scheme that provides a sustainable water infrastructure network. Water grid technology enables better management of the water network, meaning leakages, supply interruptions and uncontrolled discharges and helps in potential water shortage and protects the environment.

The South East Queensland water grid is a region wide, long term, water supply scheme that provides sustainable water infrastructure network for the south east region of the Queensland, Australia. The project was the largest urban response to the drought in Australia, which severely affected water supplies in Brisbane and surrounds, particularly between 2004 and 2007. The basic component of the project was a 535 kilometre network of potable bulk water pipelines that connect areas that have an oversupply of water to those areas lacking water. The project went online in October 2008 and by November 2008 parts of the region were receiving a diversified supply of water for the first time. This was initially managed by a partnership between seqwater, linkwater and seq water grid manager.

Water distribution system is used to deliver water to consumer with appropriate quantity, quality and pressure. It describes collectively the facilities used to supply water from its source to the point of usage. Different layouts of distribution system are Grid iron system is suitable for cities with rectangular layout, where the water mains and branches are laid in rectangles. In Ring system the supply main is laid all along the peripheral roads and sub mains branch out from the mains.

In urban areas the water supply to residence and commercial establishments are provided at a fixed flow rate. There are incidents of excess water drawn by certain customers/users, water will be released unofficially which is considered as water theft. In this project it is proposed to develop an embedded based remote water monitoring and theft prevention system by taking the data of water supply at the consumer/user end.

The overall objective of a distribution system is to deliver wholesome water to the consumer at particular area and in sufficient quantity and achieve continuity and maximum coverage at affordable cost. To attain this objective, the organization has to evolve operating procedures to ensure that the system can be operated satisfactorily, function efficiently and continuously as far as possible at lowest cost.

Here we are using AT89S52 as our controller and also few sensors are arranged to detect the presence of water in that particular pipeline. As logic level converters are used to detect the water flow. Water should be released as per the instructions by officials for example alternate days of supply are provided and only during specific period of time but not daily.

All the details will be shown in the web server using IOT module connected to the controller. So that the authorities can take necessary action in case of misuse. This is an advanced, trouble-free, fit and forgets system for water board. By using all these malfunctioning can be avoided.

This project uses regulated 5V, 500mA power supply. 7805 three terminal voltage regulator is used for voltage regulation. Bridge type full wave rectifier is used to rectify the ac output of secondary of 230/12V step down transformer.

3. INTRODUCTION TO EMBEDDED SYSTEMS

3.1 INTRODUCTION

An embedded system is a system which is going to do a predefined specified task is the embedded system and is even defined as combination of both software and hardware. A general-purpose definition of embedded systems is that they are devices used to control, monitor or assist the operation of equipment, machinery or plant. 'Embedded' reflects the fact that they are an integral part of the system. At the other extreme a general-purpose computer may be used to control the operation of a large complex processing plant, and its presence will be obvious.

All embedded systems are including computers or microprocessors. Some of these computers are however very simple systems as compared with a personal computer. The very simplest embedded systems are capable of performing only a single function or set of functions to meet a single predetermined purpose. In more complex systems an application program that enables the embedded system to be used for a particular purpose in a specific application determines the functioning of the embedded system. The ability to have programs means that the same embedded system can be used for a variety of different purposes. In some cases, a microprocessor may be designed in such a way that application software for a particular purpose can be added to the basic software in a second process.

The simplest devices consist of a single microprocessor (often called a chip), which may itself be packaged with other chips in a hybrid system or Application Specific Integrated Circuit (ASIC). Its input comes from a detector or sensor and its output goes to a switch or activator which may start or stop the operation of a machine or, by operating a valve, may control the flow of fuel to an engine.

As the embedded system is the combination of both software and hardware. Software deals with the languages like ALP, C, and VB etc., and Hardware deals with Processors, Peripherals, and Memory.

Memory: It is used to store data or address.

Peripherals: These are the external devices connected.

Processor: It is an IC which is used to perform some task.

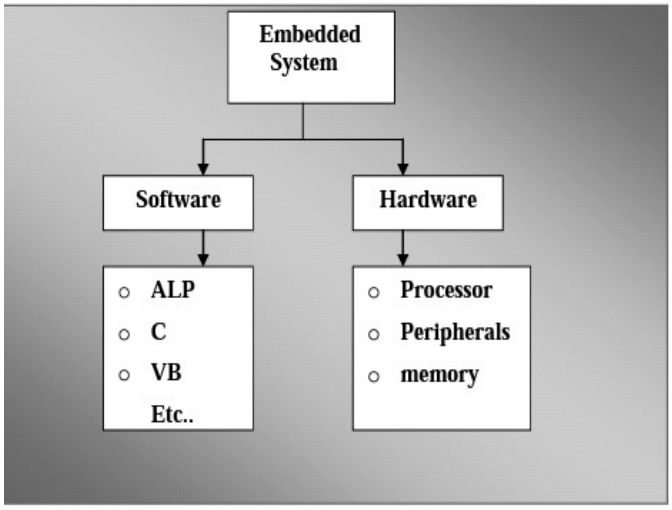


Figure 3.1 Block diagram of Embedded System.

3.2 APPLICATIONS OF EMBEDDED SYSTEMS

1. Manufacturing and process control.
2. Construction industry.
3. Transport.
4. Buildings and premises.
5. Domestic service.
6. Communications.
7. Office systems and mobile equipment.
8. Banking, finance and commercial.
9. Medical diagnostics, monitoring and life support.
10. Testing, monitoring and diagnostic systems.

3.3 TECHNICAL SPECIFICATIONS OF PROJECT

1.	Domain	:	Embedded Systems Design
2.	Software	:	Embedded C, Keil, Proload
3.	Microcontroller	:	AT89S52
4.	Power Supply	:	+5V, 500mA Regulated Power Supply
5.	Display	:	LCD
6.	LCD	:	HD44780 16-character, 2-line (16X2)
7.	Crystal	:	11.0592MHz
8.	Sensor	:	Logic level converter
9.	Communication device	:	ESP8266

4. HARDWARE DESCRIPTION

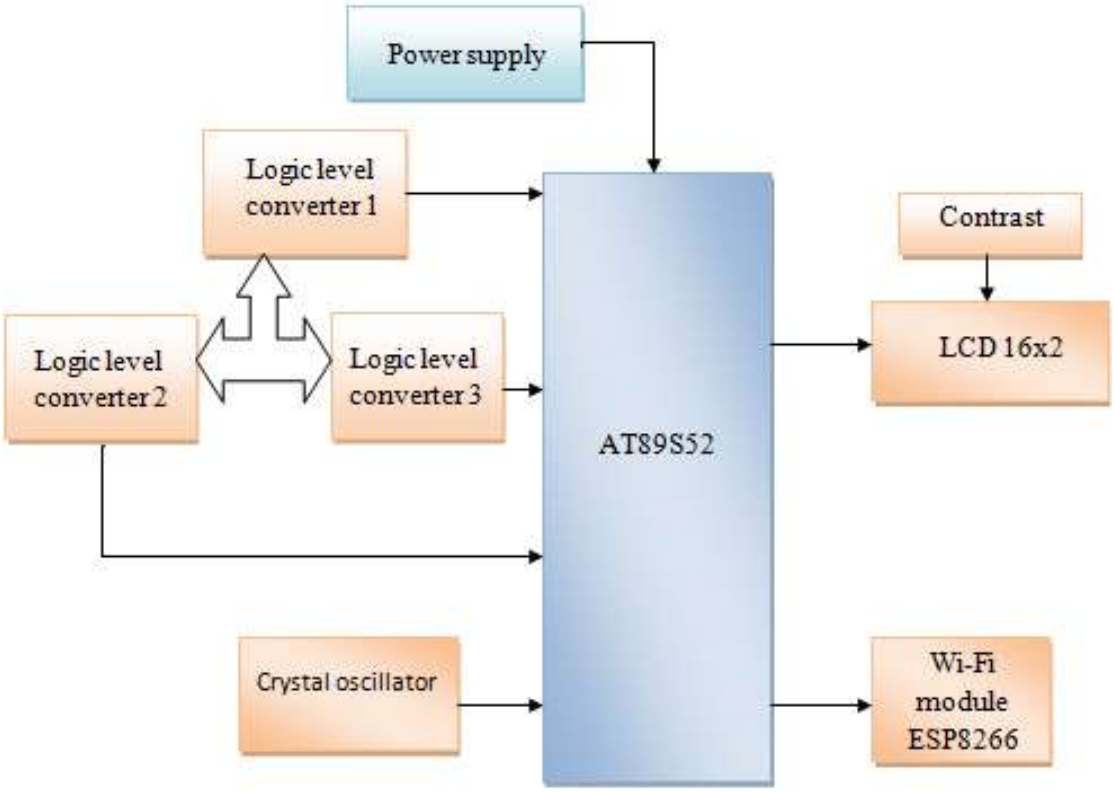
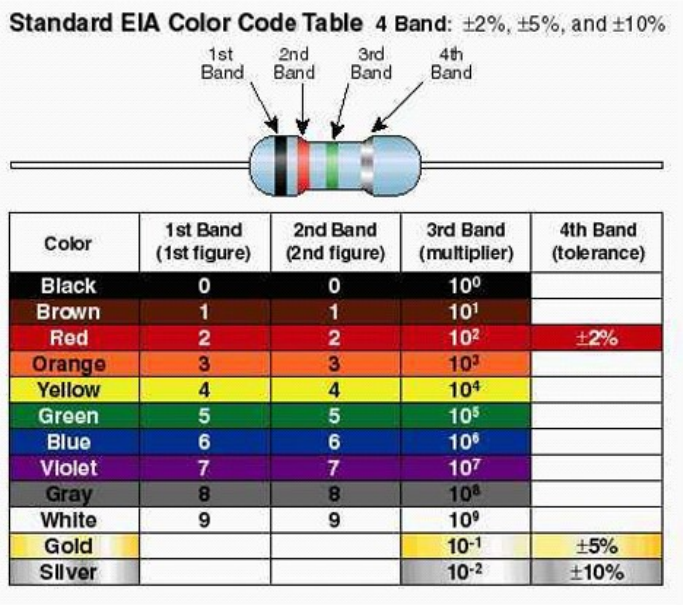


Figure 4.1 Block diagram of Public water grid monitoring to avoid water theft using IOT.

4.1 RESISTOR

Resistors "Resist" the flow of electrical current. The higher the value of resistance (measured in ohms) the lower the current will be. Resistance is the property of a component which restricts the flow of electric current. Energy is used up as the voltage across the component drives the current through it and this energy appears as heat in the component

Resistors are rated according to their maximum power dissipation. Discrete resistors in solid-state electronic systems are typically rated as 1/10, 1/8, or 1/4 watt. They usually absorb much less than a watt of electrical power and require little attention to their power rating.



Color	1st Band (1st figure)	2nd Band (2nd figure)	3rd Band (multiplier)	4th Band (tolerance)
Black	0	0	10^0	
Brown	1	1	10^1	
Red	2	2	10^2	±2%
Orange	3	3	10^3	
Yellow	4	4	10^4	
Green	5	5	10^5	
Blue	6	6	10^6	
Violet	7	7	10^7	
Gray	8	8	10^8	
White	9	9	10^9	
Gold			10^{-1}	±5%
Silver			10^{-2}	±10%

Figure 4.2 Resistor and its color code.

4.2 CAPACITOR

Capacitors store electric charge. They are used with resistors in timing circuits because it takes time for a capacitor to fill with charge. They are used to smooth varying DC supplies by acting as a reservoir of charge. They are also used in filter circuits because capacitors easily pass AC (changing) signals but they block DC (constant) signals. Electrolytic capacitors are polarized and they must be connected the correct way round, at least one of their leads will be marked + or -.

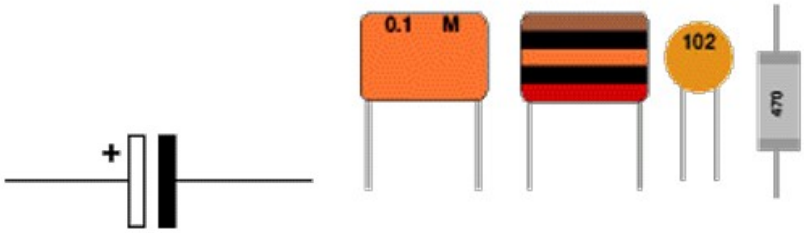


Figure 4.3 Capacitor symbol and types of capacitors.

4.3 DIODES

Diodes allow electricity to flow in only one direction. The arrow of the circuit symbol shows the direction in which the current can flow. Diodes are the electrical version of a valve and early diodes were actually called valves.

Figure 4.4 Diode symbol and types of diode.

Diodes must be connected the correct way round, the diagram may be labeled **a** or **+** for anode and **k** or **-** for cathode (yes, it really is k, not c, for cathode!). The cathode is marked by a line painted on the body. Diodes are labeled with their code in small print; you may need a magnifying glass to read this on small signal diodes.

4.3.1 LIGHT-EMITTING DIODE (LED)

The longer lead is the anode (+) and the shorter lead is the cathode (&minus). In the schematic symbol for an LED (bottom), the anode is on the left and the cathode is on the right. Light emitting diodes are elements for light signalization in electronics.

Figure 4.5 LED symbol.

They are manufactured in different shapes, colors and sizes. For their low price, low consumption and simple use, they have almost completely pushed aside other light sources- bulbs at first place.

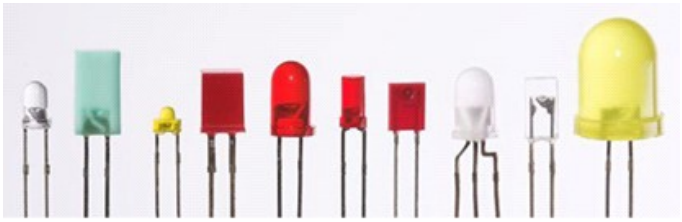


Figure 4.6 Types of LED.

It is important to know that each diode will be immediately destroyed unless its current is limited. This means that a conductor must be connected in parallel to a diode. In order to correctly determine value of this conductor, it is necessary to know diode's voltage drop in forward direction, which depends on what material a diode is made of and what colors it is. Values typical for the most frequently used diodes are

shown in table below: As seen, there are three main types of LEDs. Standard ones get full brightness at current of 20mA. Low Current diodes get full brightness at ten time's lower current while Super Bright diodes produce more intensive light than Standard ones.

4.4 SWITCHES AND PUSHBUTTONS

A push button switch is used to either close or open an electrical circuit depending on the application. Push button switches are used in various applications such as industrial equipment control handles, outdoor controls, mobile communication terminals, and medical equipment, and etc. Push button switches generally include a push button disposed within a housing. The push button may be depressed to cause movement of the push button relative to the housing for directly or indirectly changing the state of an electrical contact to open or close the contact. Also included in a pushbutton switch may be an actuator, driver, or plunger of some type that is situated within a switch housing having at least two contacts in communication with an electrical circuit within which the switch is incorporated.

Figure 4.7 Push button switch.

Typical actuators used for contact switches include spring loaded force cap actuators that reciprocate within a sleeve disposed within the canister. The actuator is typically coupled to the movement of the cap assembly, such that the actuator translates in a direction that is parallel with the cap. A push button switch for a data input unit for a mobile communication device such as a cellular phone, a key board for a personal_computer or the like is generally constructed by mounting a cover member directly on a circuit board. Printed circuit board (PCB) mounted pushbutton switches are an inexpensive means of providing an operator interface on industrial control products. In such push button switches, a substrate which includes a plurality of movable sections is formed of a rubber elastomeric. The key top is formed on a top surface thereof with a figure, a character or the like by printing, to thereby provide a cover member. Push button switches incorporating lighted displays have been used in

a variety of applications. Such switches are typically comprised of a pushbutton, an opaque legend plate, and a back light to illuminate the legend plate.

4.5 TRANSISTOR

A transistor is a semiconductor device used to amplify and switch electronic signals and electrical power. It is composed of semiconductor material with at least three terminals for connection to an external circuit. A voltage or current applied to one pair of the transistor's terminals changes the current flowing through another pair of terminals. Because the controlled (output) power can be higher than the controlling (input) power, a transistor can amplify a signal. Today, some transistors are packaged individually, but many more are found embedded in integrated circuits.

The essential usefulness of a transistor comes from its ability to use a small signal applied between one pair of its terminals to control a much larger signal at another pair of terminals. This property is called gain. A transistor can control its output in proportion to the input signal; that is, it can act as an amplifier. Alternatively, the transistor can be used to turn current on or off in a circuit as an electrically controlled switch, where the amount of current is determined by other circuit elements.

There are two types of transistors, which have slight differences in how they are used in a circuit. A bipolar transistor has terminals labeled base, collector, and emitter. A small current at the base terminal can control or switch a much larger current between the collector and emitter terminals. For a field-effect transistor, the terminals are labeled gate, source, and drain, and a voltage at the gate can control a current between source and drain.

BJT used as an electronic switch, in grounded-emitter configuration. Transistors are commonly used as electronic switches, both for high-power applications such as switched-mode power supplies and for low-power applications such as logic gates.

In any switching circuit, values of input voltage would be chosen such that the output is either completely off, or completely on. The transistor is acting as a switch,

and this type of operation is common in digital circuits where only ‘on’and ‘off" values are relevant.

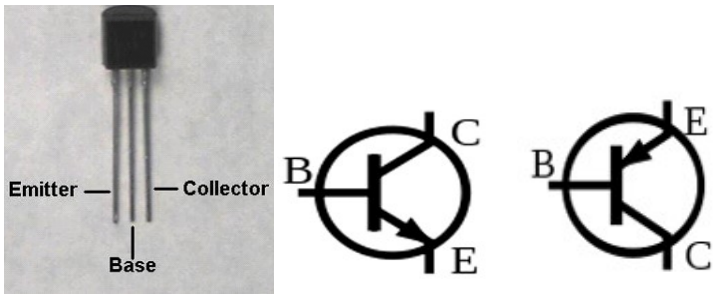


Figure 4.8: Transistor, NPN and PNP symbols.

NPN Transistor

The symbol of NPN is ‘not pointing in’. NPN is one of the two types of bipolar transistors, consisting of a layer of P-doped semiconductor between two N-doped layers. A small current entering the base is amplified to produce a large collector and emitter current. That is, when there is a positive potential difference measured from the emitter of an NPN transistor to its base as well as positive potential difference measured from the base to the collector, the transistor becomes active. In this ‘on’ state, current flows between the collector and emitter of the transistor. Most of the current is carried by electrons moving from emitter to collector as minority carriers in the P-type base region. To allow for greater current and faster operation, most bipolar transistors used today are NPN because electron mobility is higher than hole mobility.

PNP Transistor

The symbol of a PNP is ‘points in proudly’. The other type of BJT is the PNP, consisting of a layer of N-doped semiconductor between two layers of P-doped material. A small current leaving the base is amplified in the collector output. That is, a PNP transistor is ‘on’ when its base is pulled low relative to the emitter.

The arrows in the NPN and PNP transistor symbols are on the emitter legs and point in the direction of the conventional current flow when the device is in forward active mode.

A bipolar junction transistor is a type of transistor that relies on the contact of two types of semiconductor for its operation. BJTs can be used as amplifiers,

switches, or in oscillators. BJTs can be found either as individual discrete components, or in large numbers as parts of integrated circuits.

Bipolar transistors are so named because their operation involves both electrons and holes. These two kinds of charge carriers are characteristic of the two kinds of doped semiconductor material. In contrast, unipolar transistors such as the field-effect transistors have only one kind of charge carrier.

Charge flow in a BJT is due to bidirectional diffusion of charge carriers across a junction between two regions of different charge concentrations. The regions of a BJT are called emitter, collector, and base. A discrete transistor has three leads for connection to these regions. By design, most of the BJT collector current is due to the flow of charges injected from a high-concentration emitter into the base where there are minority carriers that diffuse toward the collector, and so BJTs are classified as minority-carrier devices.

4.6 REGULATED POWER SUPPLY

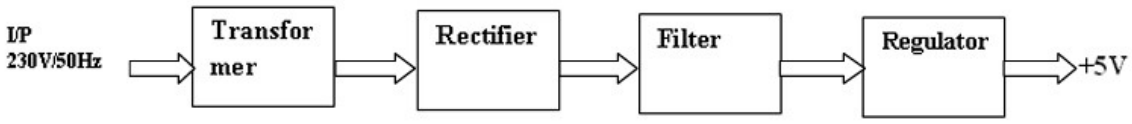


Figure 4.9 Block Diagram of Regulated Power Supply (RPS).

4.6.1 TRANSFORMER

A transformer is a device that transfers electrical energy from one circuit to another through inductively coupled conductors—the transformer's coils. A varying current in the first or primary winding creates a varying magnetic flux in the transformer's core, and thus a varying magnetic field through the secondary winding. This varying magnetic field induces a varying electromotive force (EMF) or voltage in the secondary winding. This effect is called mutual induction.Transformer is a device that converts the one form energy to another form of energy like a transducer.

Basic Principle

A transformer makes use of Faraday's law and the ferromagnetic properties of an iron core to efficiently raise or lower AC voltages. It of course cannot increase

power so that if the voltage is raised, the current is proportionally lowered and vice versa.

Transformer Working

A transformer consists of two coils often called 'windings' linked by an iron core, as shown in figure below. There is no electrical connection between the coils; instead they are linked by a magnetic field created in the core. Transformers are used to convert electricity from one voltage to another with minimal loss of power. They only work with AC because they require a changing magnetic field to be created in their core. Transformers can increase voltage (step-up) as well as reduce voltage (step-down).

Alternating current flowing in the primary coil creates a continually changing magnetic field in the iron core. This field also passes through the secondary coil and the changing strength of the magnetic field induces an alternating voltage in the secondary coil. If the secondary coil is connected to a load the induced voltage will make an induced current flow. The correct term for the induced voltage is 'induced electromotive force' which is usually abbreviated to induced e.m.f.

The iron core is laminated to prevent 'eddy currents' flowing in the core. These are currents produced by the alternating magnetic field inducing a small voltage in the core, just like that induced in the secondary coil. Eddy currents waste power by needlessly heating up the core but they are reduced to a negligible amount by laminating the iron because this increases the electrical resistance of the core without affecting its magnetic properties.

Classification of Transformer

1. Step-Up Transformer
2. Step-Down Transformer

Step-Down Transformer

Step down transformers are designed to reduce electrical voltage. Their primary voltage is greater than their secondary voltage. This kind of transformer ‘steps down’ the voltage applied to it. For instance, a step down transformer is needed to use a 110v product in a country with a 220v supply. Step down transformers

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

Step down transformers are made from two or more coils of insulated wire wound around a core made of iron. When voltage is applied to one coil it magnetizes the iron core, which induces a voltage in the other coil, The turn's ratio of the two sets of windings determines the amount of voltage transformation.

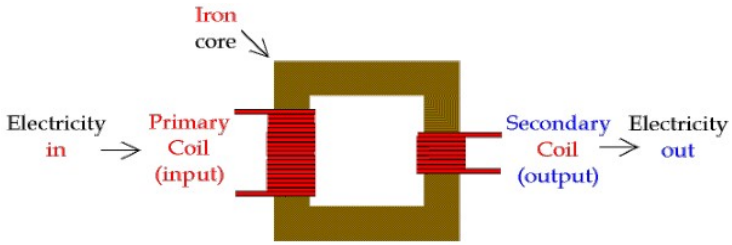


Figure 4.10 Step-Down Transformer.

Step-Up Transformer

A step up transformer has more turns of wire on the secondary coil, which makes a larger induced voltage in the secondary coil. It is called a step up transformer because the voltage output is larger than the voltage input. Step-up transformer 110v 220v design is one whose secondary voltage is greater than its primary voltage. This kind of transformer "steps up" the voltage applied to it. For instance, a step up transformer is needed to use a 220v product in a country with a 110v supply.

The primary components for voltage transformation are the step up transformer core and coil. The insulation is placed between the turns of wire to prevent shorting to one another or to ground. This is typically comprised of Mylar, nomex, Kraft paper, varnish, or other materials. As a transformer has no moving parts, it will typically have a life expectancy between 20 and 25 years.

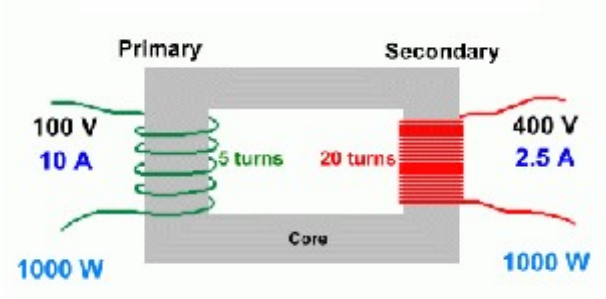


Figure 4.11 Step-Up Transformer.

4.6.2 RECTIFIER

The purpose of a rectifier is to convert an AC waveform into a DC waveform (or) Rectifier converts AC current or voltages into DC current or voltage. There are two different rectification circuits, known as 'half-wave' and 'full-wave' rectifiers. Both use components called diodes to convert AC into DC.

1. The Half-Wave Rectifier

The half-wave rectifier is the simplest type of rectifier since it only uses one diode, as shown in figure.

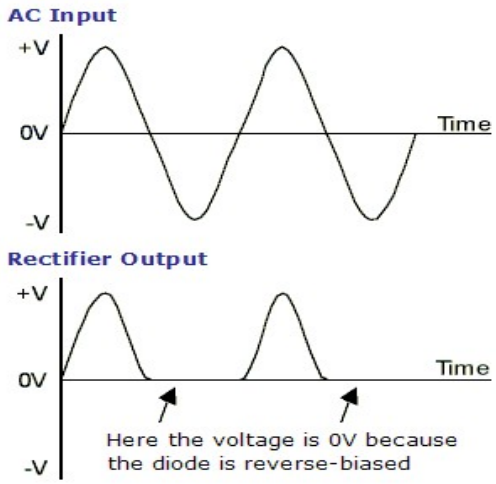


Figure 4.12 Half Wave Rectifier waveforms.

Figure shows the AC input waveform to this circuit and the resulting output. As you can see, when the AC input is positive, the diode is forward-biased and lets the current through. When the AC input is negative, the diode is reverse-biased and the diode does not let any current through, meaning the output is 0V. Because there is a 0.7V voltage loss across the diode, the peak output voltage will be 0.7V less than Vs.

While the output of the half-wave rectifier is DC (it is all positive), it would not be suitable as a power supply for a circuit. Firstly, the output voltage continually varies between 0V and Vs-0.7V, and secondly, for half the time there is no output at all.

2. The Full-wave Rectifier

The circuit in figure addresses the second of these problems since at no time is the output voltage 0V. This time four diodes are arranged so that both the positive

and negative parts of the AC waveform are converted to DC. The resulting waveform is shown in figure.

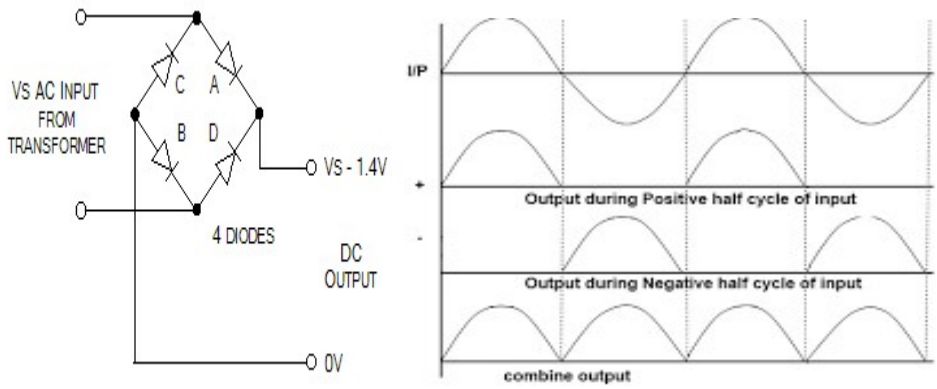


Figure 4.13 Full-Wave Rectifier and waveforms.

When the AC input is positive, diodes A and B are forward-biased, while diodes C and D are reverse-biased. When the AC input is negative, the opposite is true - diodes C and D are forward-biased, while diodes A and B are reverse-biased.

4.6.3 CAPACITOR FILTER

The capacitor-input filter**,** also called "Pi" filter due to its shape that looks like the Greek letter pi, is a type of electronic filter. Filter circuits are used to remove unwanted or undesired frequencies from a signal.

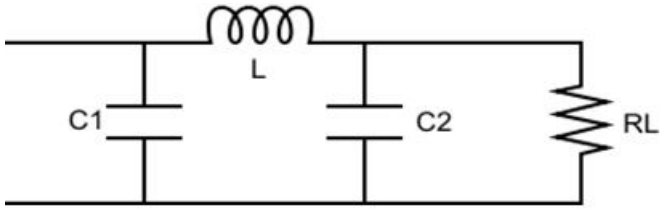


Figure 4.14 Capacitor Filter.

A typical capacitor input filter consists of a filter capacitor C1, connected across the rectifier output, an inductor L, in series and another filter capacitor connected across the load.

1. The capacitor C1 offers low reactance to the AC component of the rectifier output while it offers infinite reactance to the DC component. As a result the capacitor shunts an appreciable amount of the AC component while the DC component continues its journey to the inductor L.
2. The inductor L offers high reactance to the AC component but it offers almost zero reactance to the DC component. As a result, the DC component flows through the inductor while the AC component is blocked.

3. The capacitor C2 bypasses the AC component which the inductor had failed to block. As a result, only the DC component appears across the load RL.

4.6.4 VOLTAGE REGULATOR

A voltage regulator is an electrical regulator designed to automatically maintain a constant voltage level. It may use an electromechanical mechanism, or passive or active electronic components. Depending on the design, it may be used to regulate one or more AC or DC voltages. There are two types of regulator are they.

1. Positive Voltage Series (78xx) and
2. Negative Voltage Series (79xx).

78xx series voltage regulator

'78' indicate the positive series and 'xx' indicates the voltage rating. Suppose 7805 produces the maximum 5V.'05'indicates the regulator output is 5V.

79xx series voltage regulator

'78' indicate the negative series and 'xx' indicates the voltage rating. Suppose 7905 produces the maximum -5V.'05'indicates the regulator output is -5V.

These regulators consist the three pins there are

Pin1: It is used for input pin.

Pin2: This is ground pin for regulator.

Pin3: It is used for output pin. Through this pin we get the output.

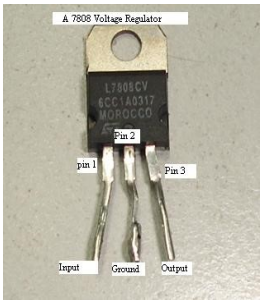


Figure 4.15 Voltage Regulator.

4.7 AT89S52 MICROCONTROLLER

4.7.1 INTRODUCTION TO MICROCONTROLLERS

Microprocessors and microcontrollers are widely used in embedded systems products. Microcontroller is a programmable device. A microcontroller has a CPU in

addition to a fixed amount of RAM, ROM, I/O ports and a timer embedded all on a single chip. The fixed amount of on-chip ROM, RAM and number of I/O ports in microcontrollers makes them ideal for many applications in which cost and space are critical.

The Intel 8052 is Harvard architecture, single chip microcontroller (μC) which was developed by Intel in 1980 for use in embedded systems. It was popular in the 1980s and early 1990s, but today it has largely been superseded by a vast range of enhanced devices with 8052-compatible processor cores that are manufactured by more than 20 independent manufacturers including Atmel, Infineon Technologies and Maxim Integrated Products.

The present project is implemented on Keil uVision. In order to program the device, proload tool has been used to burn the program onto the microcontroller.

4.7.2 FEATURES

1. Compatible with MCS-51 Products.
2. 8K Bytes of In-System Programmable (ISP) Flash Memory - Endurance 1000 Write/Erase Cycles.
3. 4.0V to 5.5V Operating Range.
4. Fully Static Operation: 0 Hz to 33 MHz.
5. Three-level Program Memory Lock.
6. 256 x 8-bit Internal RAM.
7. 32 Programmable I/O Lines.
8. Three 16-bit Timer/Counters.
9. Eight Interrupt Sources.
10. Full Duplex UART Serial Channel.
11. Low-power Idle and Power-down Modes.
12. Interrupt Recovery from Power-down Mode.
13. Watchdog Timer.
14. Dual Data Pointer.
15. Power-off Flag.

4.7.3 DESCRIPTION

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 nonvolatile memory technology and is compatible with the industry- standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer[5]. 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.The oldest models of the 8052 microcontroller family did not have any internal program memory. It was added from outside as a separate chip. These models are recognizable by their label beginning with 803 (for ex. 8031 or 8032). All later models have a few Kbytes ROM embedded, even though it is enough for writing most of the programs, there are situations when additional memory is necessary. A typical example of it is the use of so called lookup tables. They are used in cases when something is too complicated or when there is no time for solving equations describing some process.

4.7.4 PIN CONFIGURATIONS

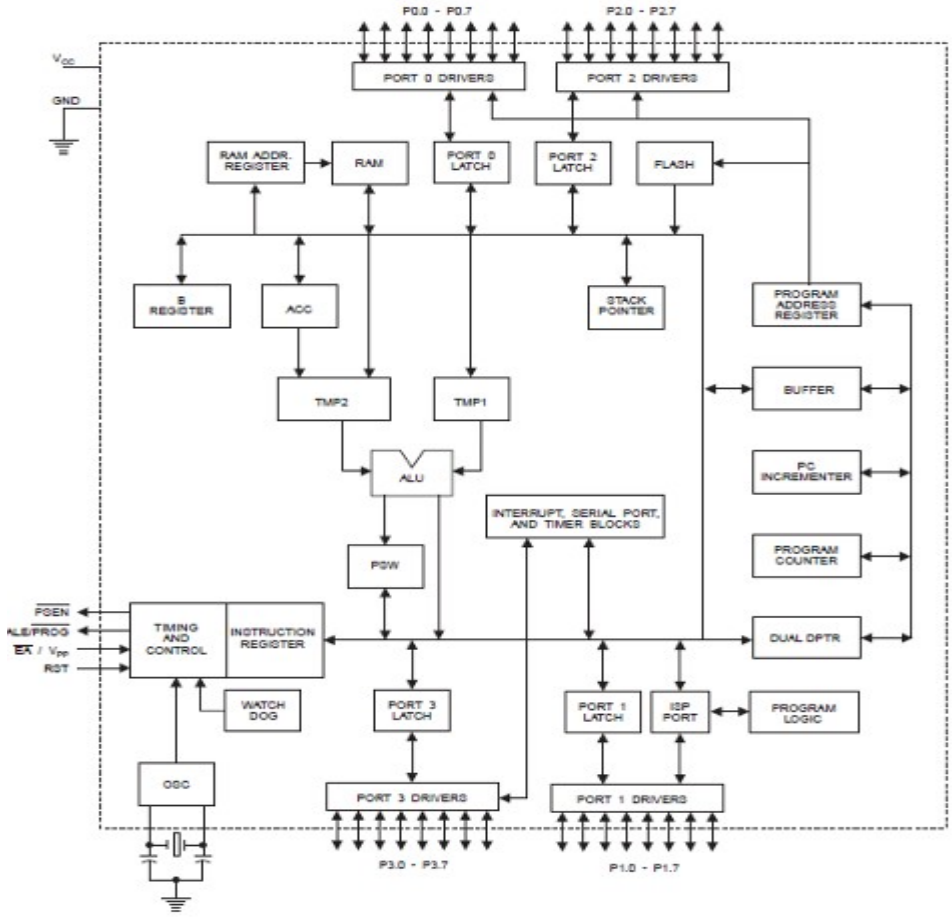


Figure 4.16 AT89S52 architecture.

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), respectively, as shown in the following table. Port 1 also receives the low-order address bytes during Flash programming and verification.

Table 4.1 Port 1 pin functions.

Port Pin	Alternate Functions
P1.0	T2(external count input to Timer/Counter 2), clock-out
P1.1	T2EX(Timer/Counter 2 capture/reload trigger and direction control)
P1.5	MOSI(used for In-System Programming)
P1.6	MISO(used for In-System Programming)
P1.7	SCK(used for In-System Programming)

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 use 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 use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

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. Port 3 also serves the functions of various special features of the AT89S52, as shown in the following table. Port 3 also receives some control signals for Flash programming and verification.

Table 4.2 Port 3 pin functions.

Port Pin	Alternate Functions
P3.0	RXD(serial input port)
P3.1	TXD(serial output port)
P3.2	INT0(external interrupt 0)
P3.3	INT1(external interrupt 1)
P3.4	T0(timer 0 external input)
P3.5	T1(timer 1 external input)
P3.6	WR(external data memory write strobe)
P3.7	RD(external data memory read strobe)

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 96 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. If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

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.

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure. 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 the below figure. 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.

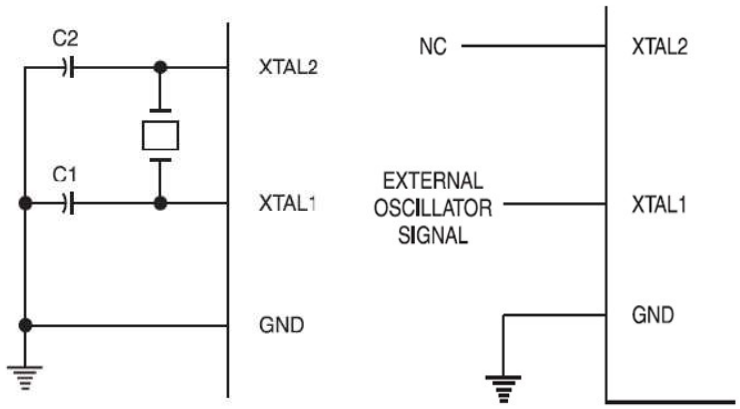


Figure 4.17 Oscillator Connections and External Clock Drive Configuration.

C1, C2 = 30 pF ± 10 pF for Crystals = 40 pF ± 10 pF for Ceramic Resonators

4.7.5 MICROCONTROLLER MEMORY ORGANISATION

The microcontroller memory is divided into Program Memory and Data Memory. Program Memory (ROM) is used for permanent saving program being executed, while Data Memory (RAM) is used for temporarily storing and keeping intermediate results and variables. Depending on the model in use at most a few Kb of ROM and 128 or 256 bytes of RAM can be used. All microcontrollers have 16-bit addressing bus and can address 64 kb memory.

Program Memory

The oldest models of the 8052 microcontroller family did not have any internal program memory. It was added from outside as a separate chip. These models

are recognizable by their label beginning with 803. All later models have a few Kbytes ROM embedded, even though it is enough for writing most of the programs, there are situations when additional memory is necessary. A typical example of it is the use of so called lookup tables. They are used in cases when something is too complicated or when there is no time for solving equations describing some process. The example of it can be totally exotic or totally common . In those cases, all needed estimates and approximates are executed in advance and the final results are put in the tables

EA=0: In this case, internal program memory is completely ignored, only a program stored in external memory is to be executed.

EA=1: In this case, a program from built-in ROM is to be executed first (to the last location). Afterwards, the execution is continued by reading additional memory.
in both cases, P0 and P2 are not available to the user because they are used for data and address transmission. Besides, the pins ALE and PSEN are used too.

Data Memory

As already mentioned, Data Memory is used for temporarily storing and keeping data and intermediate results created and used during microcontroller's operating. Besides, this microcontroller family includes many other registers such as hardware counters and timers, input/output ports, serial data buffers etc. The previous versions have the total memory size of 256 locations, while for later models this number is incremented by additional 128 available registers. In both cases, these first 256 memory locations (addresses 0-FFh) are the base of the memory. Common to all types of the 8052 microcontrollers. Locations available to the user occupy memory space with addresses from 0 to 7Fh. First 128 registers and this part of RAM is divided in several blocks.

The first block consists of 4 banks each including 8 registers designated as R0 to R7. Prior to access them, a bank containing that register must be selected. Next memory block is bit- addressable, which means that each bit being there has its own address from 0 to 7Fh. Since there are 16 such registers, this block contains in total of 128 bits with separate addresses The third groups of registers occupy addresses 2Fh-7Fh (in total of 80 locations) and does not have any special purpose or feature.

Additional Memory Block of Data Memory

In order to satisfy the programmers' permanent hunger for Data Memory, producers have embedded an additional memory block of 128 locations into the latest versions of the 8052 microcontrollers. Naturally, it's not so simple…The problem is that electronics performing addressing has 1 byte (8 bits) on disposal and due to that it can reach only the first 256 locations. In order to keep already existing 8-bit architecture and compatibility with other existing models a little trick has been used.

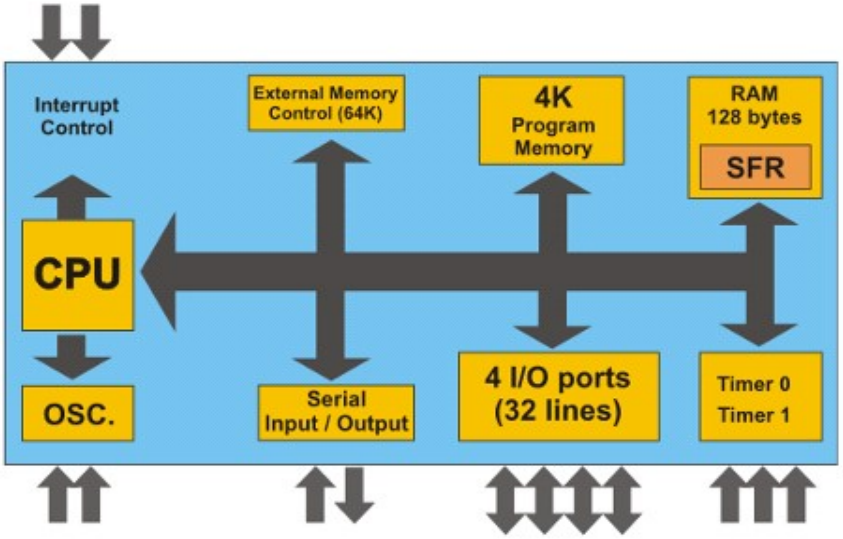


Figure 4.18 Microcontroller internal structure.

How to extend memory?

In case on-chip memory is not enough, it is possible to add two external memory chips with capacity of 64Kb each. I/O ports P2 and P3 are used for their addressing and data transmission. Even though the additional memory is rarely used with the latest versions of the microcontrollers, it will be described here in short what happens when memory chips are connected according to the previous scheme. It is important to know that the whole process is performed automatically, i.e. with no intervention in the program.

1. When the program during execution encounters the instruction which resides in external memory (ROM), the microcontroller will activate its control output ALE and set the first 8 bits of address (A0-A7) on P0. In this way, IC circuit 74HCT573 which "lets in" the first 8 bits to memory address pins is activated.
2. A signal on the pin ALE closes the IC circuit 74HCT573 and immediately afterwards 8 higher bits of address (A8-A15) appear on the port. In this way, a

desired location in additional program memory is completely addressed. The only thing left over is to read its content.

Pins on P0 are configured as inputs, the pin PSEN is activated and the microcontroller reads content from memory chip. The same connections are used both for data and lower address byte.

Similar occurs when it is a needed to read some location from external Data Memory. Now, addressing is performed in the same way, while reading or writing is performed via signals which appear on the control outputs RD or WR.

Addressing

While operating, processor processes data according to the program instructions. Each instruction consists of two parts. One part describes what should be done and another part indicates what to use to do it. This later part can be data (binary number) or address where the data is stored. All 8052 microcontrollers use two ways of addressing depending on which part of memory should be accessed.

Direct Addressing

On direct addressing, a value is obtained from a memory location while the address of that location is specified in instruction. Only after that, the instruction can process data (how depends on the type of instruction: addition, subtraction, copy…). Obviously, a number being changed during operating a variable can reside at that specified address.

Indirect Addressing

On indirect addressing, a register which contains address of another register is specified in the instruction. A value used in operating process resides in that another register.

SFRs (Special Function Registers)

SFRs are a kind of control table used for running and monitoring microcontroller's operating. Each of these registers, even each bit they include, has its name, address in the scope of RAM and clearly defined purpose (for example: timer control, interrupt, serial connection etc.). Even though there are 128 free memory locations intended for their storage, the basic core, shared by all types of 8052

controllers, has only 21 such registers. Rest of locations are intentionally left free in order to enable the producers to further improved models keeping at the same time compatibility with the previous versions. It also enables the use of programs written a long time ago for the microcontrollers which are out of production now.

A Register (Accumulator)

This is a general-purpose register which serves for storing intermediate results during operating. A number should be added to the accumulator prior to execute an instruction upon it. Once an arithmetical operation is performed by the ALU, the result is placed into the accumulator. If a data should be transferred from one register to another, it must go through accumulator. For such universal purpose, this is the most commonly used register that none microcontroller can be imagined without.

B Register

B register is used during multiply and divide operations which can be performed only upon numbers stored in the A and B registers. All other instructions in the program can use this register as a spare accumulator (A).

R Registers (R0-R7)

This is a common name for the total 8 general purpose registers (R0, R1, R2 ...R7). Even they are not true SFRs, they deserve to be discussed here because of their purpose. The bank is active when the R registers it includes are in use. Similar to the accumulator, they are used for temporary storing variables and intermediate results. Which of the banks will be active depends on two bits included in the PSW Register. These registers are stored in four banks in the scope of RAM.

Machine cycle

The CPU takes a certain number of clock cycles to execute an instruction. these clock cycles are referred to as machine cycles. The length of the machine cycle depends on the frequency of the crystal oscillator. The crystal oscillator, along with on-chip circuitry, provides the clock source for the CPU. The frequency can vary from 4 MHz to 30 MHz, depending upon the chip rating and manufacturer.

But the exact frequency of 11.0592 MHz crystal oscillator is used to make the 8052 based system compatible with the serial port of the IBM PC. In the original

version of 8052, one machine cycle lasts 12 oscillator periods. Therefore, to calculate the machine cycle for the 8052, the calculation is made as 1/12 of the crystal frequency and its inverse is taken.

4.8 LIQUID CRYSTAL DISPLAY

4.8.1 INTRODUCTION TO LCD

LCD stands for **L**iquid **C**rystal **D**isplay. LCD is finding wide spread use replacing LEDs because of the following reasons.

1. The declining prices of LCDs. The ability to display numbers, characters and graphics. This is in contrast to LEDs, which are limited to numbers and a few characters.
2. Incorporation of a refreshing controller into the LCD, thereby relieving the CPU of the task of refreshing the LCD. In contrast, the LED must be refreshed by the CPU to keep displaying the data.
3. Ease of programming for characters and graphics.

It is based on the HD44780 microcontroller and can display messages in two lines with 16 characters each. It displays all the alphabets, Greek letters, punctuation marks, mathematical symbols etc. In addition, it is possible to display symbols that user makes up on its own. Automatic shifting message on display (shift left and right), appearance of the pointer, backlight etc. are considered as useful characteristics.

4.8.2 PIN FUNCTIONS

There are pins along one side of the small printed board used for connection to the microcontroller. There are total of 14 pins marked with numbers. Their function is described in the table below

Table 4.3 LCD pin functions.

Function	Pin Number	Name	Logic State	Description
Ground	1	Vss	-	0V
Power supply	2	Vdd	-	+5V
Contrast	3	Vee	-	0 - Vdd
Control of operating	4	RS	0 1	D0 – D7 are interpreted as commands D0 – D7 are interpreted as data
	5	R/W	0 1	Write data (from controller to LCD) Read data (from LCD to controller)
	6	E	0 1 From 1 to 0	Access to LCD disabled Normal operating Data/commands are transferred to LCD
Data / commands	7	D0	0/1	Bit 0 LSB
	8	D1	0/1	Bit 1
	9	D2	0/1	Bit 2
	10	D3	0/1	Bit 3
	11	D4	0/1	Bit 4
	12	D5	0/1	Bit 5
	13	D6	0/1	Bit 6
	14	D7	0/1	Bit 7 MSB

4.8.3 LCD SCREEN

LCD screen consists of two lines with 16 characters each. Each character consists of 5x7 dot matrix. Contrast on display depends on the power supply voltage and whether messages are displayed in one or two lines. For that reason, variable voltage 0-Vdd is applied on pin marked as Vee. Trimmer potentiometer is usually used for that purpose. Some versions of displays have built in backlight. When used during operating, a resistor for current limitation should be used.

4.8.4 LCD BASIC COMMANDS

All data transferred to LCD through outputs D0-D7 will be interpreted as commands or as data, which depends on logic state on pin RS: RS = 1 - Bits D0 - D7 are addresses of characters that should be displayed. Built in processor addresses built in 'map of characters' and displays corresponding symbols. Displaying position is determined by DDRAM address. This address is either previously defined or the address of previously transferred character is automatically incremented.

Table 4.4 LCD commands.

Command	**RS**	**RW**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	**Execution Time**
Clear display	0	0	0	0	0	0	0	0	0	1	1.64mS
Cursor home	0	0	0	0	0	0	0	0	1	x	1.64mS
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	40uS
Display on/off control	0	0	0	0	0	0	1	D	U	B	40uS
Cursor/Display Shift	0	0	0	0	0	1	D/C	R/L	x	x	40uS
Function set	0	0	0	0	1	DL	N	F	x	x	40uS
Set CGRAM address	0	0	0	1	CGRAM address						40uS
Set DDRAM address	0	0	1	DDRAM address							40uS
Read "BUSY" flag (BF)	0	1	BF	DDRAM address							-
Write to CGRAM or DDRAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	40uS
Read from CGRAM or DDRAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	40uS

I/D 1 = Increment (by 1)
0 = Decrement (by 1)

S 1 = Display shift on
0 = Display shift off

D 1 = Display on
0 = Display off

U 1 = Cursor on
0 = Cursor off

B 1 = Cursor blink on

R/L 1 = Shift right
0 = Shift left

DL 1 = 8-bit interface
0 = 4-bit interface

N 1 = Display in two lines
0 = Display in one line

F 1 = Characterformat5x10 dots
0 = Character format 5x7 dots

D/C 1 = Display shift

0 = Cursor blink off 0 = Cursor shift

4.8.5 LCD INITIALIZATION

Once the power supply is turned on, LCD is automatically cleared. This process lasts for approximately 15mS. After that, display is ready to operate. The mode of operating is set by default.

1. Display is cleared.

2. Mode.

DL = 1 Communication through 8-bit interface.

N = 0 Messages are displayed in one line.

F = 0 Character font 5 x 8 dots.

3. Display/Cursor on/off.

D = 0 Display off.

U = 0 Cursor off.

B = 0 Cursor blink off.

4. Character entry.

ID = 1 Addresses on display are automatically incremented by 1.

S = 0 Display shift off.

4.8.6 CONTRAST CONTROL

To have a clear view of the characters on the LCD, contrast should be adjusted. To adjust the contrast, the voltage should be varied. For this, a preset is used which can behave like a variable voltage device. As the voltage of this preset is varied, the contrast of the LCD can be adjusted.

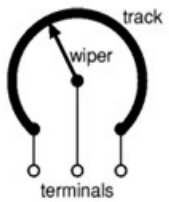


Figure 4.19 Variable resistor.

Potentiometer

Variable resistors used as potentiometers have all three terminals connected. This arrangement is normally used to vary voltage, for example to set the switching point of a circuit with a sensor, or control the volume in an amplifier circuit. If the

terminals at the ends of the track are connected across the power supply, then the wiper terminal will provide a voltage which can be varied from zero up to the maximum of the supply.

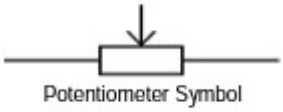


Figure 4.20 Potentiometer.

5. WI-FI MODULE ESP8266EX

5.1 INTRODUCTION

Espressif Systems' Smart Connectivity Platform (ESCP) is a set of high performance, high integration wireless SOCs, designed for space and power constrained mobile platform designers. It provides unsurpassed ability to embed Wi-Fi capabilities within other systems, or to function as a standalone application, with the lowest cost, and minimal space requirement.

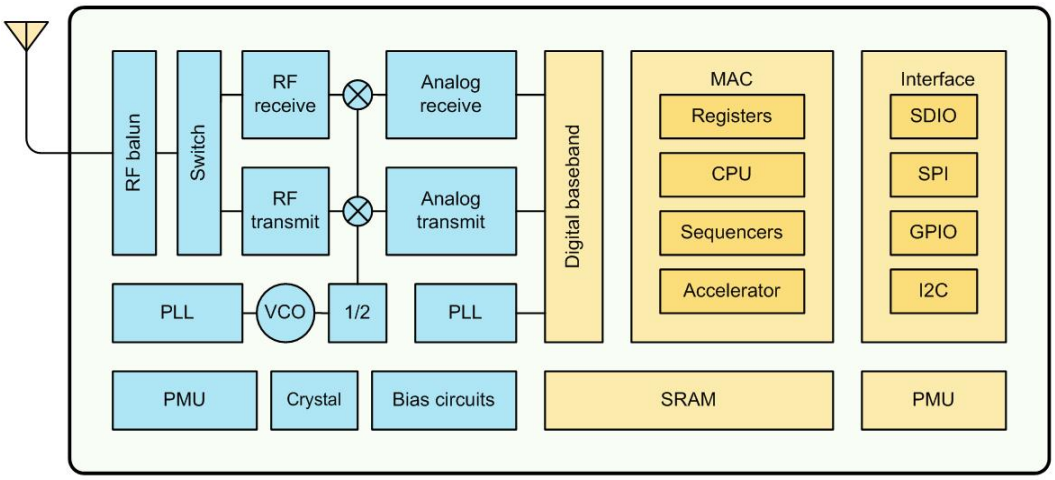


Figure 5.1 ESP8266EX architecture.

ESP8266EX offers a complete and self-contained Wi-Fi networking solution; it can be used to host the application or to offload Wi-Fi networking functions from another application processor. When ESP8266EX hosts the application, it boots up directly from an external flash[7]. In has integrated cache to improve the performance of the system in such applications. Alternately, serving as a Wi-Fi adapter, wireless internet access can be added to any micro controller-based design with simple connectivity (SPI/SDIO or I2C/UART interface). ESP8266EX is among the most integrated Wi-Fi chip in the industry; it integrates the antenna switches, RF balun, power amplifier, low noise receive amplifier, filters, power management modules, it requires minimal external circuitry, and the entire solution, including front-end module, is designed to occupy minimal PCB area. ESP8266EX also integrates an enhanced version of Tensilica's L106 Diamond series 32-bit processor, with on-chip SRAM, besides the Wi-Fi functionalities. ESP8266EX is often integrated with external sensors and other application specific devices through its GPIOs;

5.2 FEATURES

1. 802.11 b/g/n.
2. Integrated low power 32-bit MCU.
3. Integrated 10-bit ADC.
4. Integrated TCP/IP protocol stack.
5. Integrated TR switch, balun, LNA, power amplifier and matching network.
6. Integrated PLL, regulators, and power management units.
7. Supports antenna diversity.
8. Wi-Fi 2.4 GHz, support WPA/WPA2.
9. Support STA/AP/STA+AP operation modes..
10. Support Smart Link Function for both Android and iOS devices.
11. SDIO 2.0, (H) SPI, UART, I2C, I2S, IR Remote Control, PWM, GPIO.
12. STBC, 1x1 MIMO, 2x1 MIMO.
13. A-MPDU & A-MSDU aggregation & 0.4s guard interval.
14. Deep sleep power <10uA, Power down leakage current < 5uA.
15. Wake up and transmit packets in < 2ms.
16. Standby power consumption of < 1.0mW (DTIM3).
17. +20 dBm output power in 802.11b mode.
18. Operating temperature range -40C ~ 125C.
19. FCC, CE, TELEC, Wi-Fi Alliance, and SRRC certified.

ESP8266EX has been designed for mobile, wearable electronics and Internet of Things applications with the aim of achieving the lowest power consumption with a combination of several proprietary techniques. The power saving architecture operates mainly in 3 modes: active mode, sleep mode and deep sleep mode. By using advance power management techniques and logic to power-down functions not required and to control switching between sleep and active modes, ESP8266EX consumes about than 60uA in deep sleep mode and less than 1.0mA (DTIM=3) or less than 0.5mA (DTIM=10) to stay connected to the access point.

When in sleep mode, only the calibrated real-time clock and watchdog remains active. The real time clock can be programmed to wake up the ESP8266EX at any required interval. The ESP8266EX can be programmed to wake up when a specified condition is detected. This minimal wake-up time feature of the

ESP8266EX can be utilized by mobile device SOCs, allowing them to remain in the low-power standby mode until Wi-Fi is needed. In order to satisfy the power demand of mobile and wearable electronics, ESP8266EX can be programmed to reduce the output power of the PA to fit various application profiles[8].

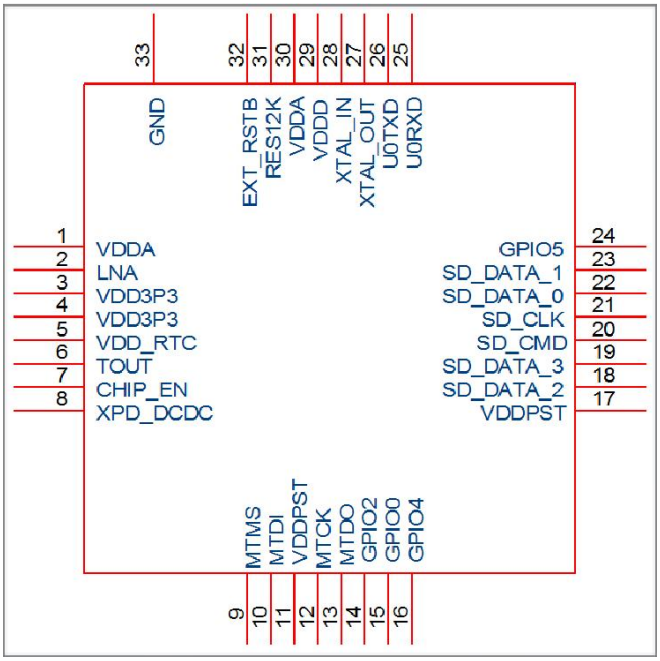


Figure 5.2 pin diagram of ESP8266EX

5.3 MAJOR APPLICATIONS

Major fields of ESP8266EX applications to Internet-of-Things include:

1. Home Appliances
2. Home Automation
3. Smart Plug and lights
4. Mesh Network
5. Industrial Wireless Control
6. Baby Monitors
7. IP Cameras
8. Sensor Networks
9. Wearable Electronics

6. SOFTWARE DESCRIPTION

6.1 KEIL SOFTWARE

Keil compiler is a software used where the machine language code is written and compiled. After compilation, the machine source code is converted into hex code which is to be dumped into the microcontroller for further processing. Keil compiler also supports C language code.

Steps to write an assembly language program in keil and how to compile it

1. Install the Keil Software in the PC in any of the drives.
2. After installation, an icon will be created with the name 'Keil uVision3'. Just drag this icon onto the desktop so that it becomes easy whenever you try to write programs in keil.
3. Double click on this icon to start the keil compiler.
4. A page opens with different options in it showing the project workspace at the leftmost corner side, output window in the bottom and an ash colored space for the program to be written.
5. Now to start using the keil, click on the option 'project'.
6. A small window opens showing the options like new project, import project, open project etc. Click on 'New project'.
7. A small window with the title bar 'Create new project'opens. The window asks the user to give the project name with which it should be created and the destination location. The project can be created in any of the drives available. You can create a new folder and then a new file or can create directly a new file.
8. After the file is saved in the given destination location, a window opens where a list of vendors will be displayed and you have to select the device for the target you have created.
9. The most widely used vendor is Atmel. So click on Atmel and now the family of microcontrollers manufactured by Atmel opens. You can select any one of the microcontrollers according to the requirement.
10. When you click on any one of the microcontrollers, the features of that particular microcontroller will be displayed on the right side of the page. The most appropriate microcontroller with which most of the projects can be implemented

is the AT89C51. Click on this microcontroller and have a look at its features. Now click on ‘OK’ to select this microcontroller.

11. A small window opens asking whether to copy the startup code into the file you have created just now. Just click on ‘No’ to proceed further.
12. Now you can see the TARGET and SOURCE GROUP created in the project workspace.
13. Now click on ‘File’ and in that ‘New’. A new page opens and you can start writing program in it.
14. After the program is completed, save it with any name but with the .asm extension. Save the program in the file you have created earlier.
15. You can notice that after you save the program, the predefined keywords will be highlighted in bold letters.
16. Now add this file to the target by giving a right click on the source group. A list of options open and in that select ‘Add files to the source group’. Check for this file where you have saved and add it.
17. Right click on the target and select the first option ‘Options for target’. A window opens with different options like device, target, output etc. First click on ‘target’.
18. Since the set frequency of the microcontroller is 11.0592 MHz to interface with the PC, just enter this frequency value in the XTAL(MHz) text area and put a tick on the Use on-chip ROM. This is because the program what we write here in the keil will later be dumped into the microcontroller and will be stored in the inbuilt ROM in the microcontroller.
19. Now click the option ‘Output’ and give any name to the hex file to be created in the ‘Name of executable’ text area and put a tick to the ‘Create HEX file’ option present in the same window. The hex file can be created in any of the drives. You can change the folder by clicking on ‘Select folder for Objects’.
20. Now to check whether the program you have written is errorless or not, click on the icon exactly below the ‘Open file’ icon which is nothing but Build Target icon. You can even use the shortcut key F7 to compile the program written.
21. To check for the output, there are several windows like serial window, memory window, project window etc. Depending on the program you have written, select the appropriate window to see the output by entering into debug mode.
22. The icon with the letter ‘d’ indicates the debug mode.

23. Click on this icon and now click on the option ‘View’ and select the appropriate window to check for the output.
24. After this is done, click the icon ‘debug’ again to come out of the debug mode.
25. The hex file created as shown earlier will be dumped into the microcontroller with the help of software called Proload.

6.1.1 INTRODUCTION TO KEIL SOFTWARE SETUP

ABOUT KEIL

1. Click on the Keil u Vision Icon on Desktop.
2. The following fig will appear.

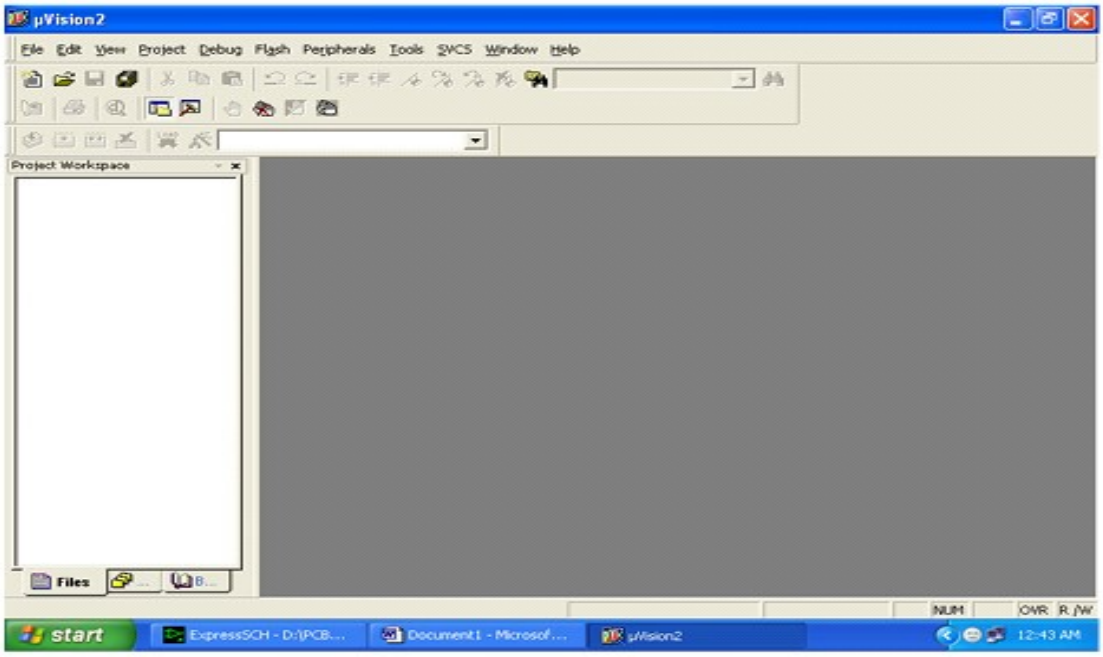


Figure 6.1 Open keil software.

3. Click on the Project menu from the title bar.
4. Then Click on New Project.

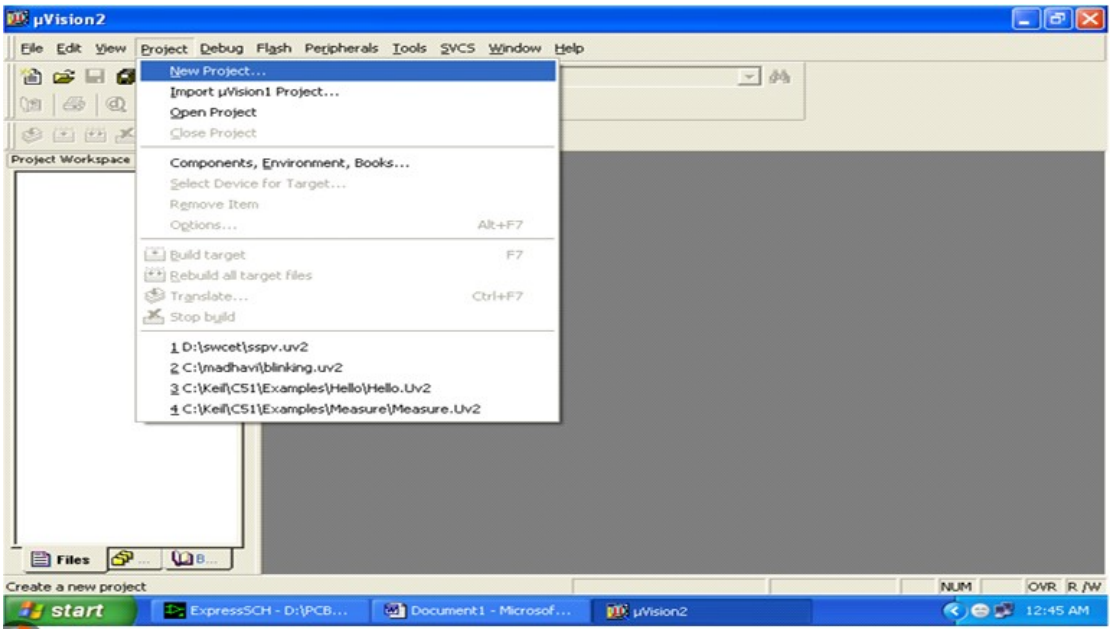


Figure 6.2 Create new project.

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

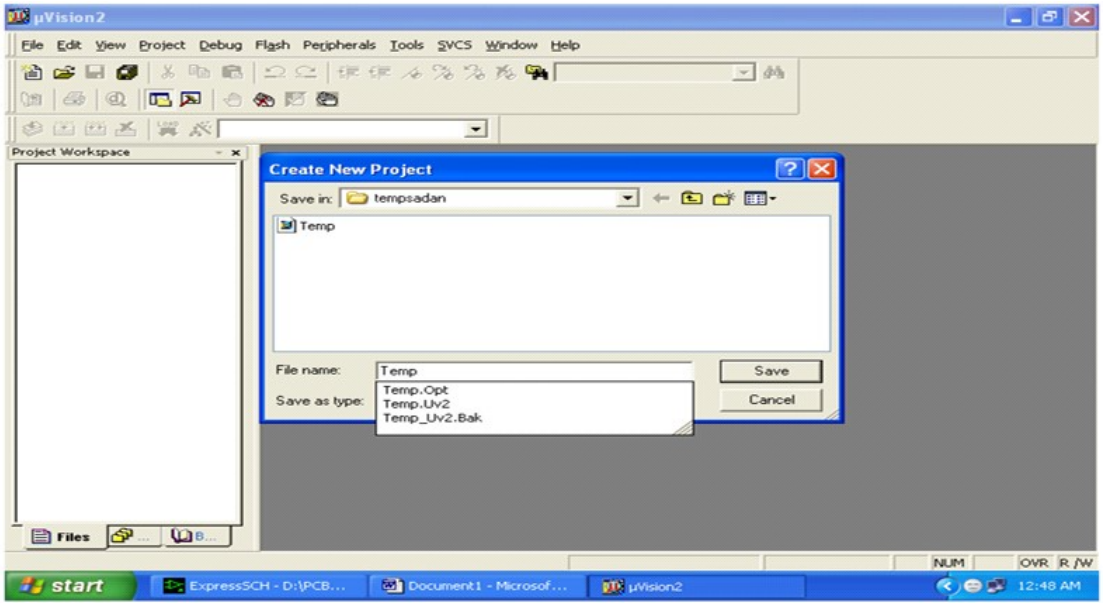


Figure 6.3 Saving project.

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

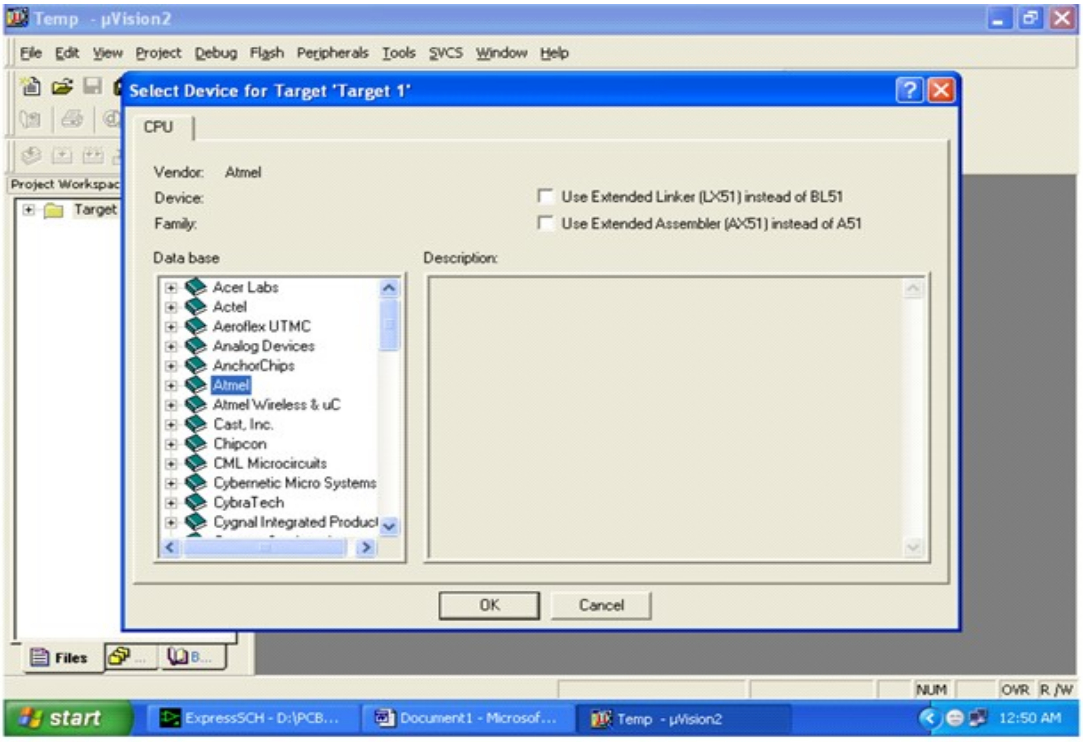


Figure 6.4 Selecting components.

9. Select AT89C51 as shown below.

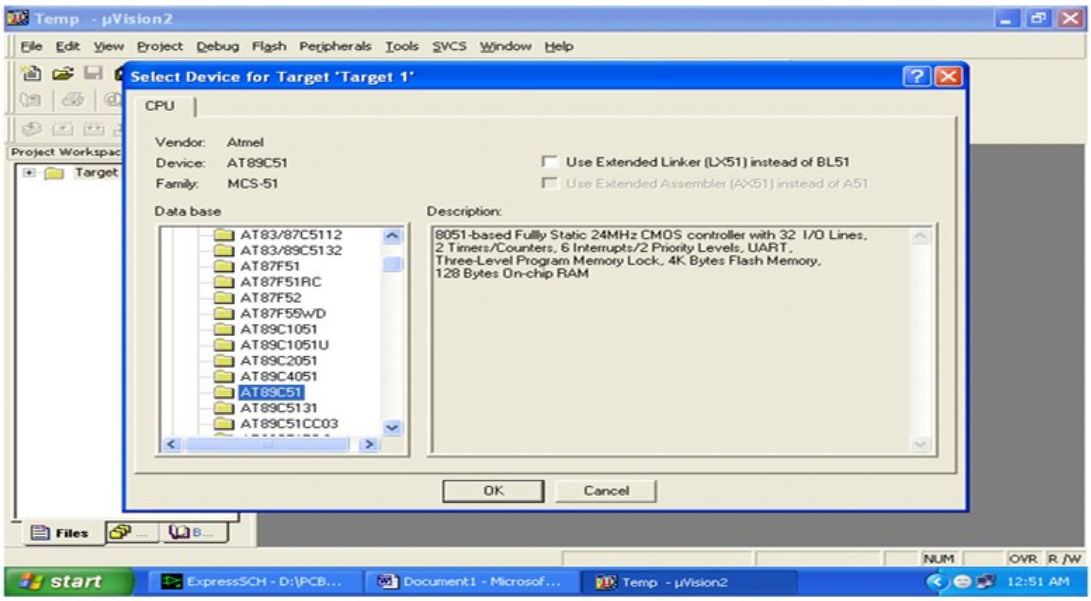


Figure 6.5 Selecting AT89S52.

10. Then Click on “OK”.

11. The Following figure will appear.

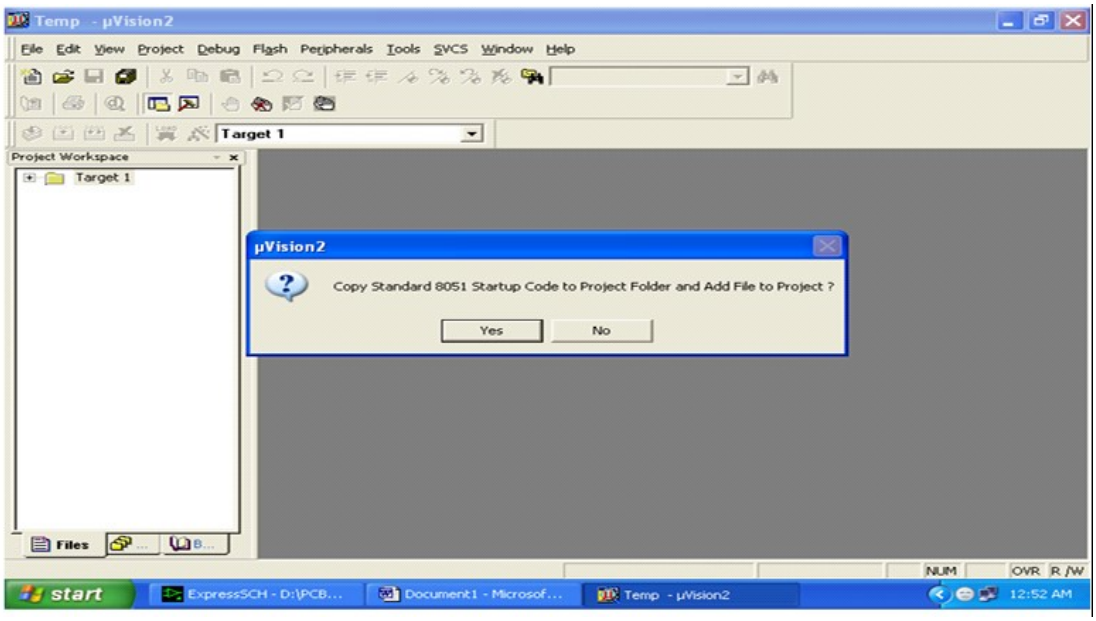


Figure 6.6 Startup code.

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.

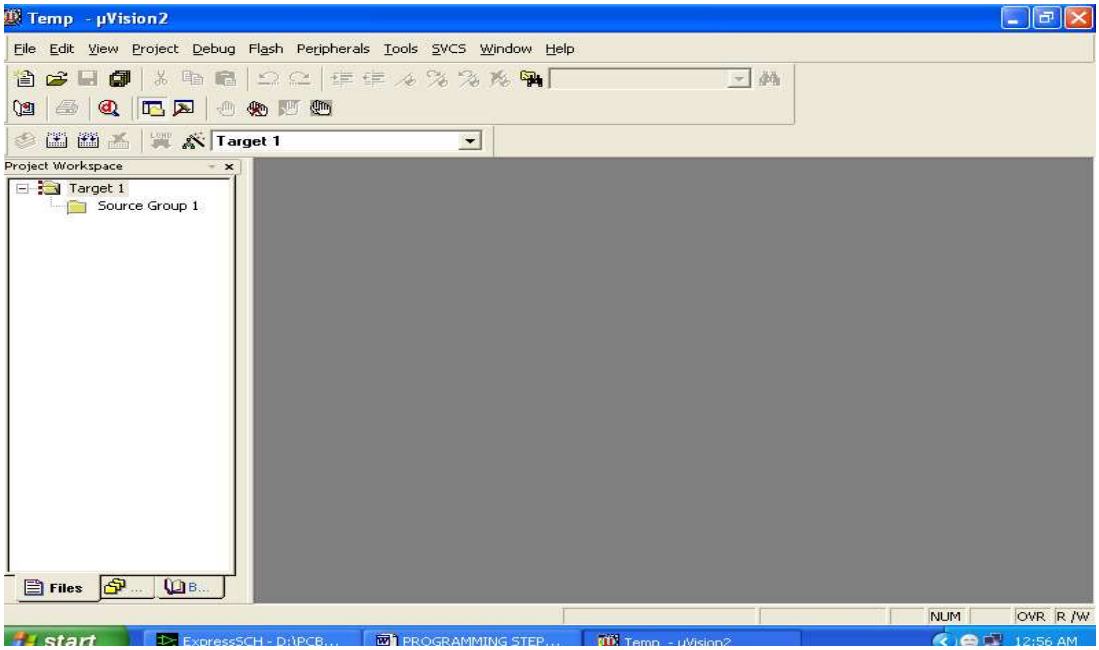


Figure 6.7 Target 1.

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

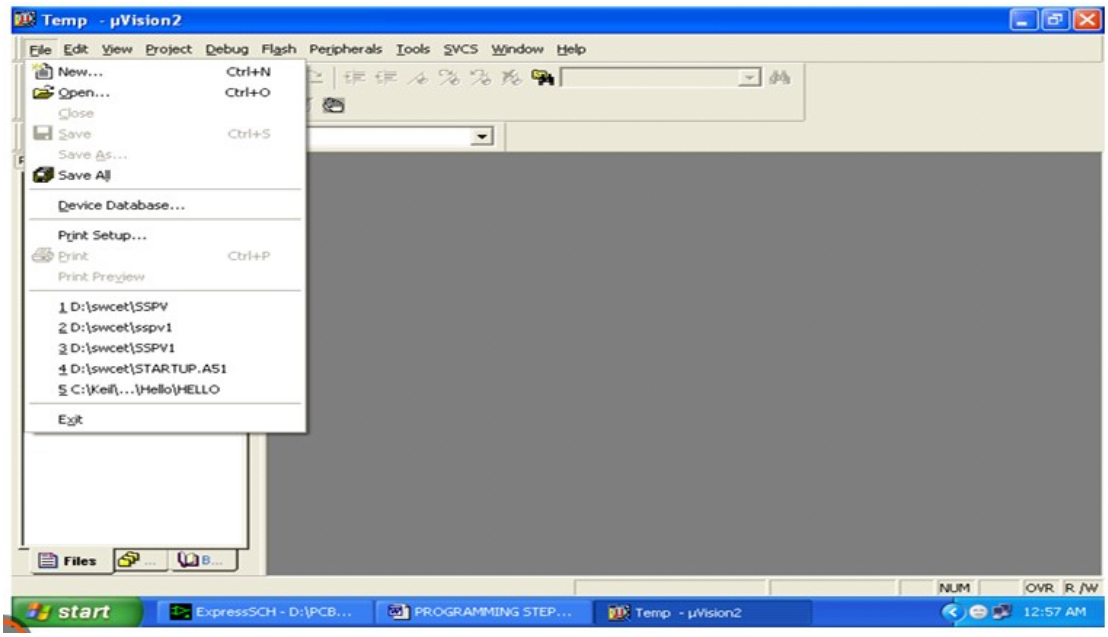


Figure 6.8 New file.

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

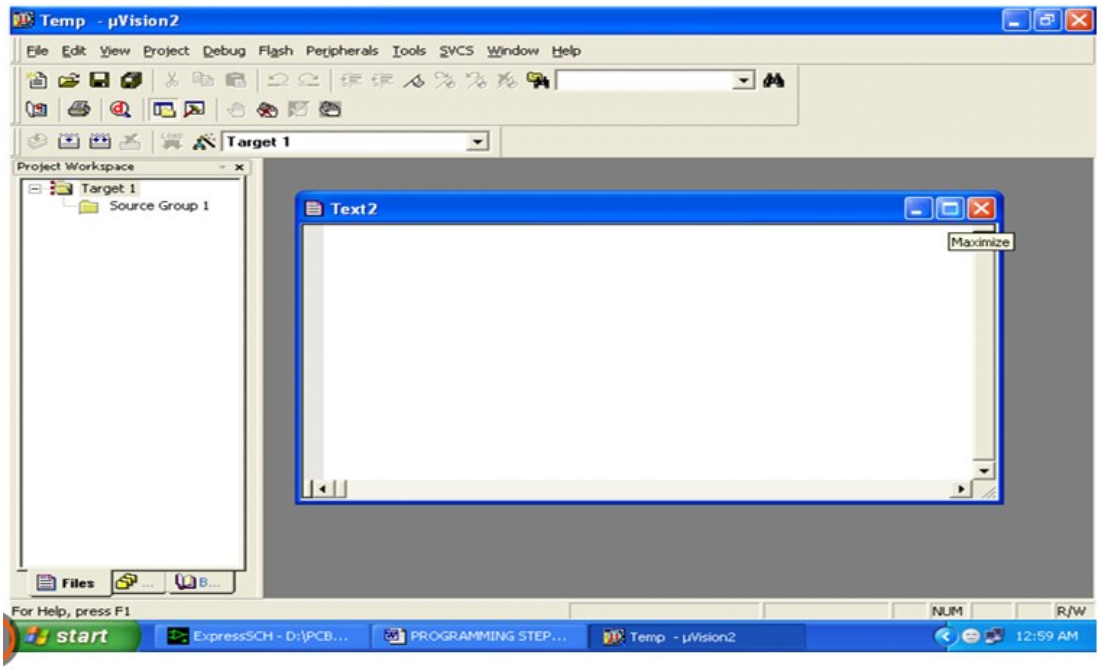


Figure 6.9 Write code.

17. Now start writing program in either in “C” or “ASM”.

18. For a program written in Assembly, then save it with extension “. asm” and for “C” based program saves it with extension “.C”.

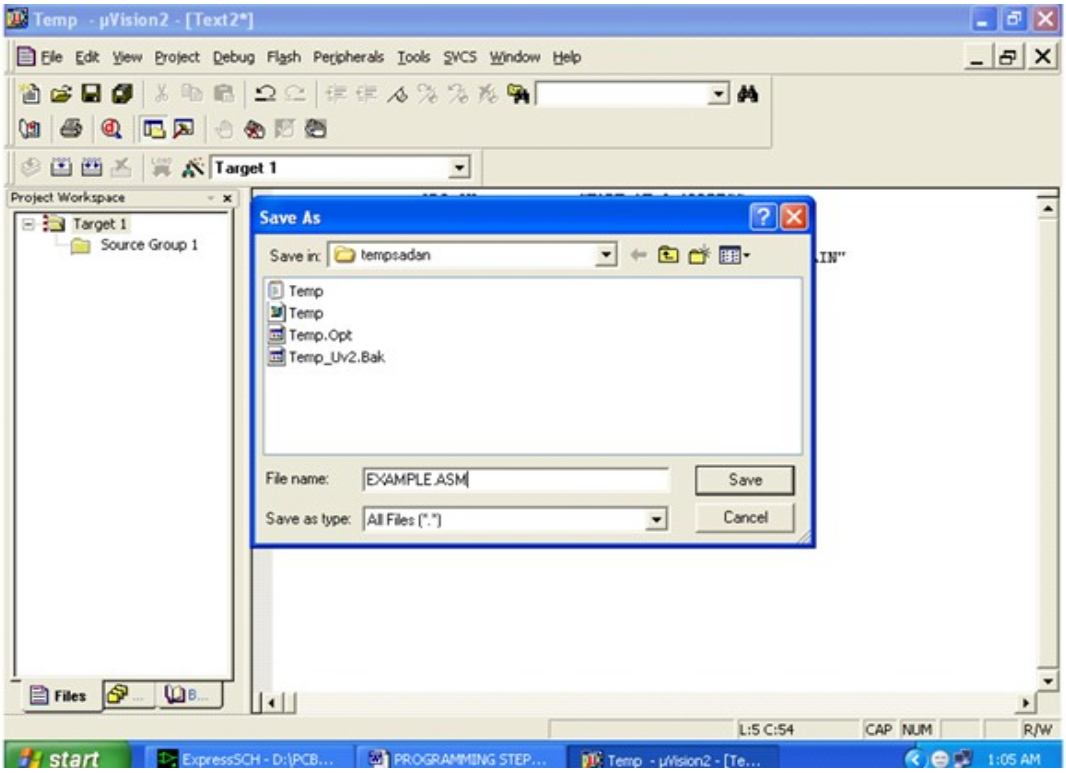


Figure 6.10 Save the file.

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

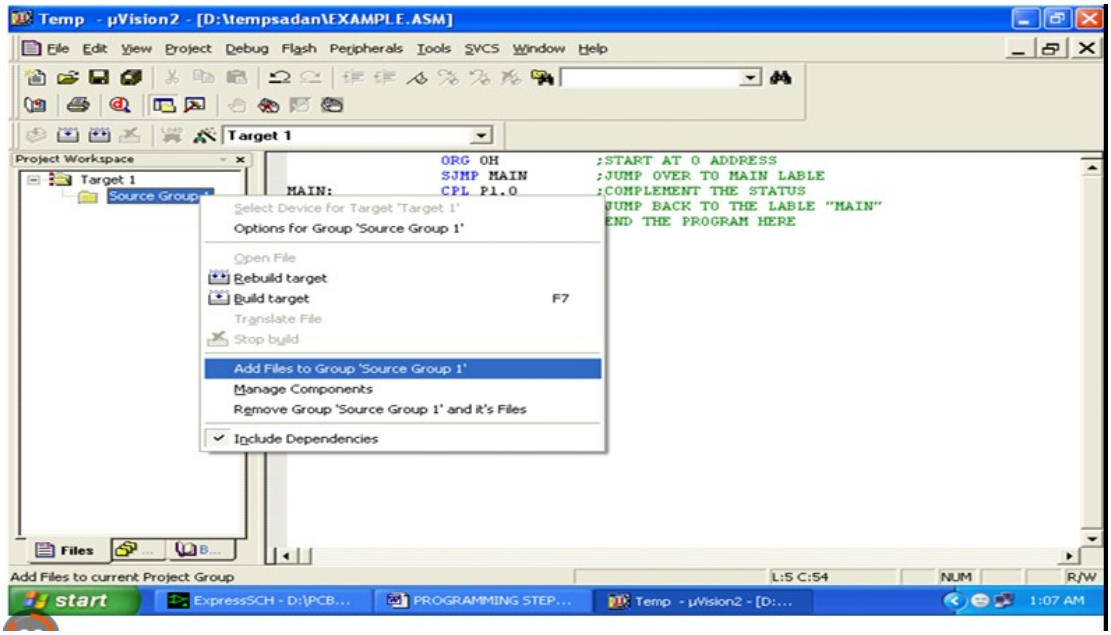


Figure 6.11 Add file to source group.

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

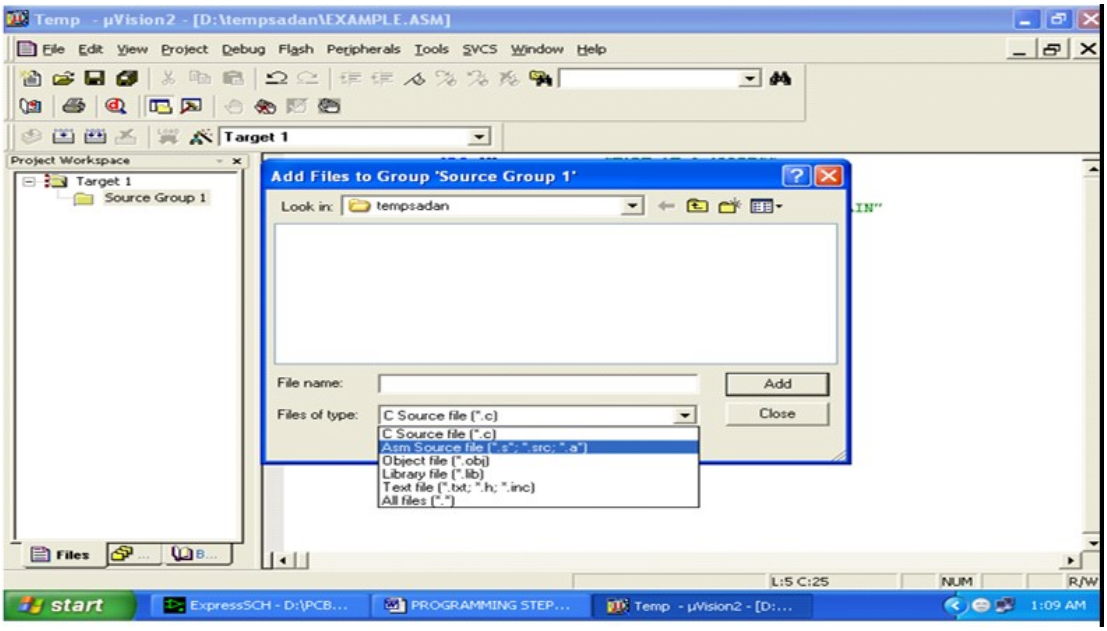


Figure 6.12 Select file.

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.

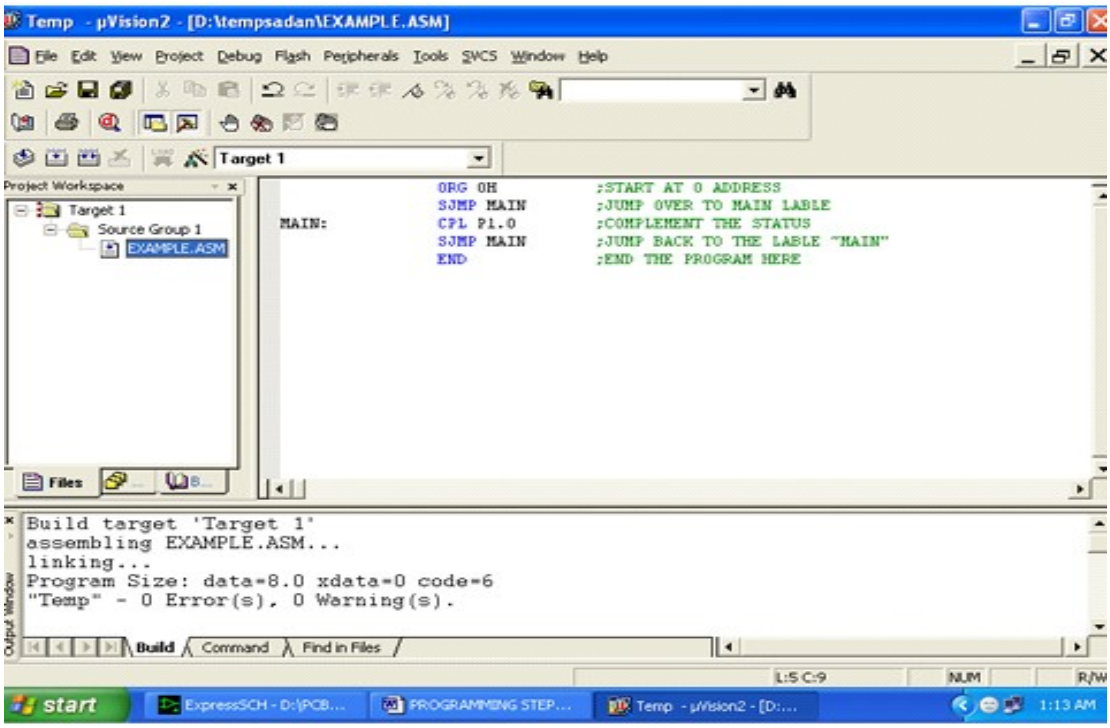


Figure 6.13 Compile the code.

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

25. The new window is as follows.

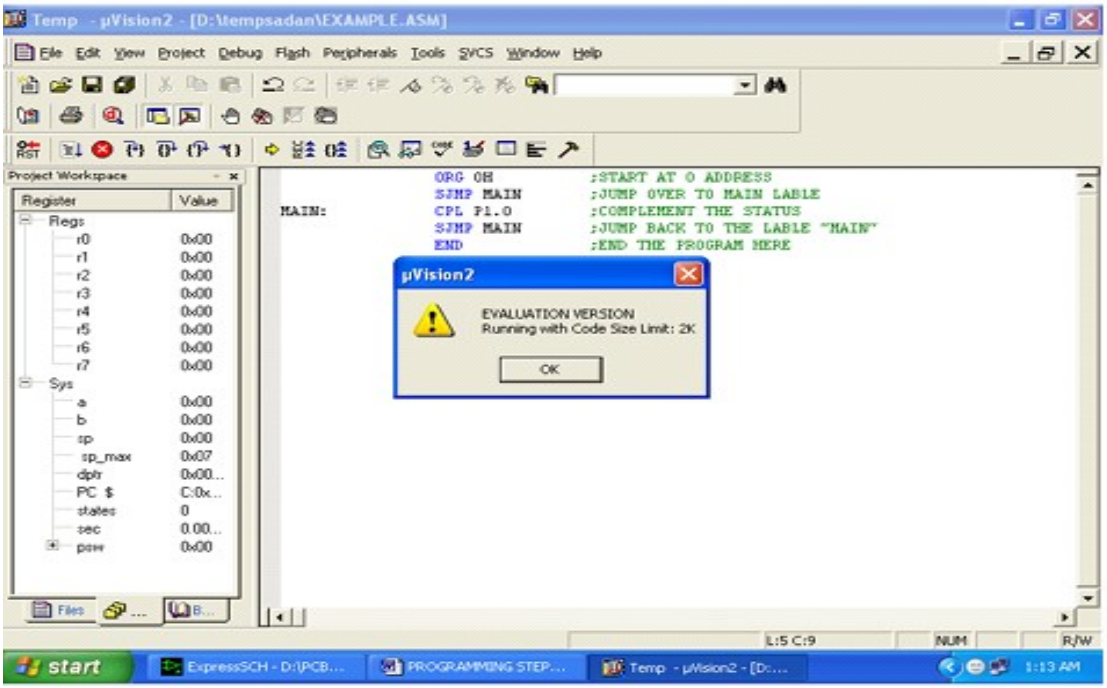


Figure 6.14 Run the code.

26. Then Click “OK”.

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

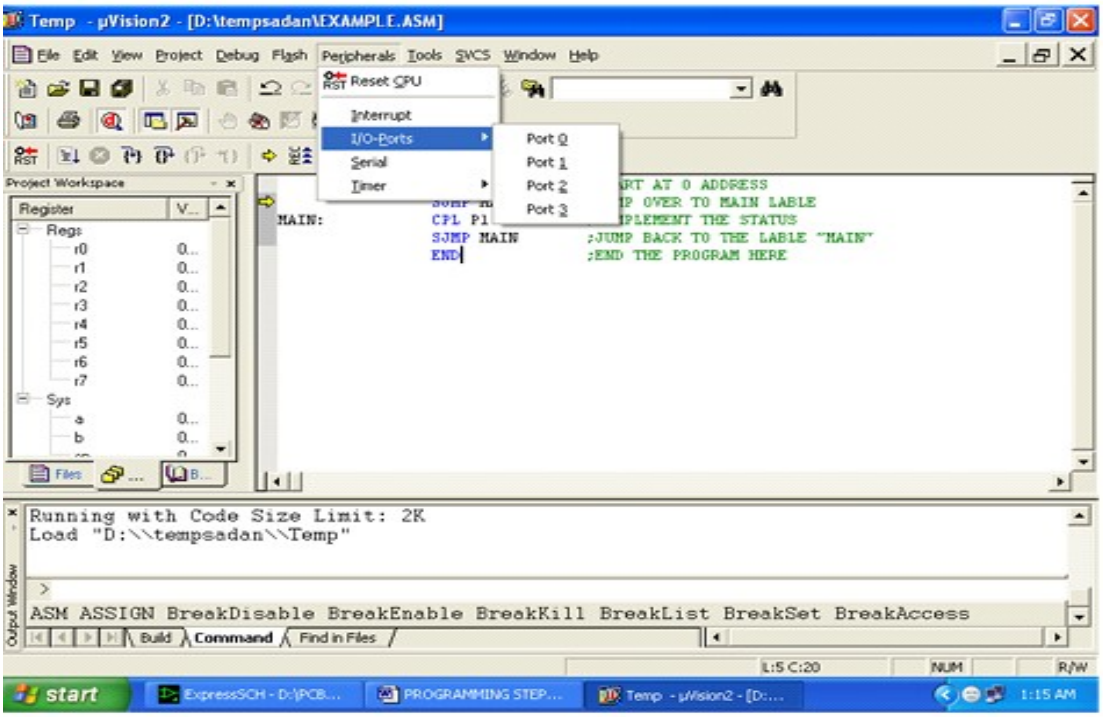


Figure 6.15 Select ports.

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

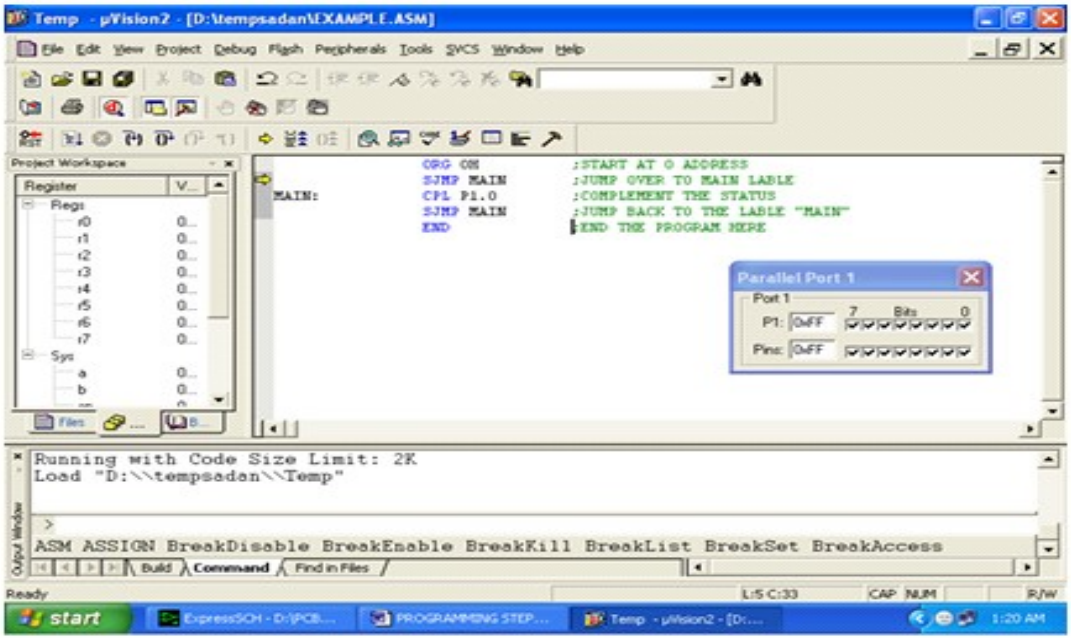


Figure 6.16 Click on program.

29. Now keep Pressing function key “F11” slowly and observe.
30. You are running your program successfully.

6.2 PROLOAD

Proload is a software which accepts only hex files. Once the machine code is converted into hex code, that hex code has to be dumped into the microcontroller placed in the programmer kit and this is done by the Proload. Programmer kit contains a microcontroller on it other than the one which is to be programmed.

This microcontroller has a program in it written in such a way that it accepts the hex file from the keil compiler and dumps this hex file into the microcontroller which is to be programmed. As this programmer kit requires power supply to be operated, this power supply is given from the power supply circuit designed above. It should be noted that this programmer kit contains a power supply section in the board itself but in order to switch on that power supply, a source is required. Thus this is accomplished from the power supply board with an output of 12volts or from an adapter connected to 230 V AC.

1. Install the Proload Software in the PC.
2. Now connect the Programmer kit to the PC (CPU) through serial cable.
3. Power up the programmer kit from the ac supply through adapter.

4. Now place the microcontroller in the GIF socket provided in the programmer kit.
5. Click on the Proload icon in the PC. A window appears providing the information like Hardware model, com port, device type, Flash size etc. Click on browse option to select the hex file to be dumped into the microcontroller and then click on 'Auto program' to program the microcontroller with that particular hex file.
6. The status of the microcontroller can be seen in the small status window in the bottom of the page.
7. After this process is completed, remove the microcontroller from the programmer kit and place it in your system board. Now the system board behaves according to the program written in the microcontroller.

6.3 PROJECT DESCRIPTION

In this section schematic diagram of public water supply grid monitoring to avoid man theft using iot, interfacing of AT89S52 microcontroller with lcd and Wi-Fi module is shown below.

6.3.1 SCHEMATIC DIAGRAM

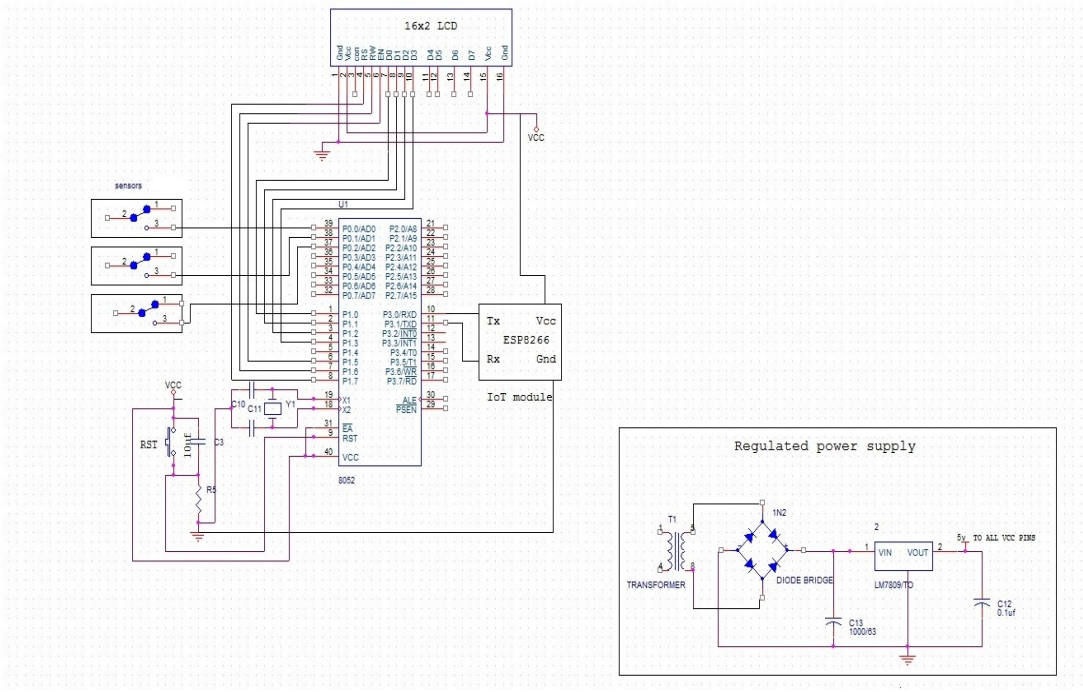


Figure 6.17 Schematic diagram.

6.3.2 LCD INTERFACING WITH THE MICROCONTROLLER

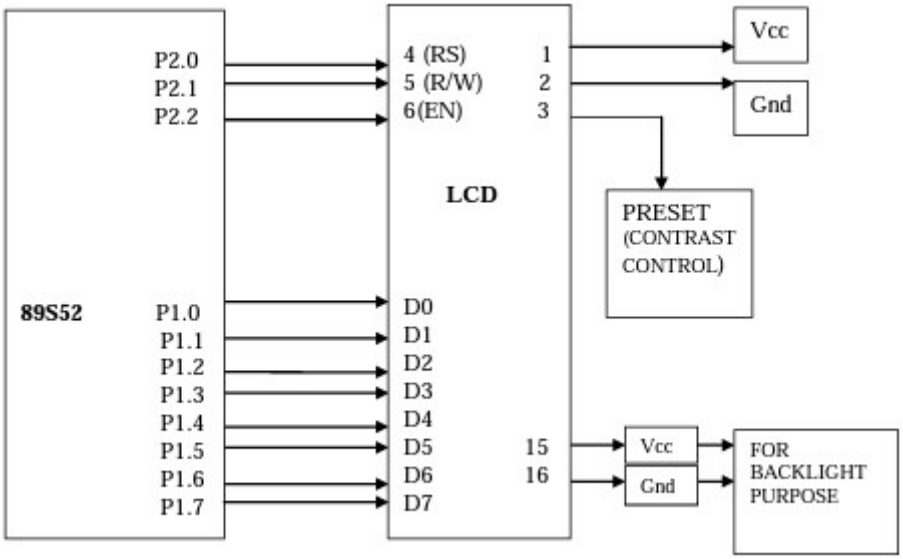


Figure 6.18 LCD Interfacing.

6.4 CODE

```
#include"reg52.h"
#include"string.h"
#include"stdio.h"
#include"lcd4.h"
sbit L1  = P0^0;
sbit L2  = P0^1;
sbit L3  = P0^2;
void delay(unsigned int);
void send_com(unsigned char *);
void new_line(void );
void longdelay(void);
void send_char(unsigned char );
unsigned int i;
int count=0,prev=-1;
unsigned char buffer[45];
unsigned int loop1;
int a,b,c,d;
```

```
void send_com(unsigned char *arr)
{
unsigned char i;
for(i=0;arr[i]!='0';i++)
{
send_char(arr[i]);
delay(10);
}
}
void main()
{
LCD_init();
TMOD =0X21;
TH1  =0X0FD;
SCON =0X50;
TR1  =1;
delay(10);
LCD_puts(0x80,"Initializing    ");
LCD_puts(0xc0," . ");
send_com("AT\n");
buz1=0;
delay(100);
buz1=1;
delay(200);
LCD_puts(0xc0," .  . ");
buz1=0;
delay(100);
buz1=1;
delay(200);
send_com("AT+CWSAP=\"channels/121585\",\"12345678\",2,3\n");
LCD_puts(0xc0," .  .  . ");
buz1=0;
delay(100);
buz1=1;
```

```
delay(200);
delay(200);
send_com("AT+CWJAP=\"IOT\",\"carvingnotions\"\n");
LCD_puts(0xc0," . . . . ");
buz1=0;
delay(200);
buz1=1;
delay(200);
delay(200);
LCD_command(0x01);
while(1)
{
LCD_puts(0x80,"WaterGrid Status");
if(L1==1)
{
LCD_puts(0xc2,"O");
a=1;
led1=0;
}
else
{
LCD_puts(0xc2,"X");
a=0;
led1=1;
}
if(L2==1)
{
LCD_puts(0xc4,"O");
b=1;
led2=0;
}
else
{
LCD_puts(0xc4,"X");
```

```
b=0;
led2=1;
}
if(L3==1)
{
LCD_puts(0xc6,"O");
c=1;
led3=0;
}
else
{
LCD_puts(0xc6,"X");
c=0;
led3=1;
}
loop1++;
if(loop1>20)
{
loop1=0;
prev=count;
send_com("AT\n");
delay(200);
delay(200);
send_com("AT+CIPSTART=\"TCP\",\"184.106.153.149\"\n");
delay(200);
delay(200);
send_com("AT+CIPSEND=80\n");
delay(200);
delay(200);
send_com("GET /update?api_key=JODGOKBUMTY38HYQ&field1=");
sprintf(buffer,"%d&field2=%d&field3=%d&field4=%d",a,b,c,d);
send_com(buffer);
send_com("\n\n");
delay(200);
```

```
delay(200);
}
}
}
void delay(unsigned int r)
{
unsigned int p,q;
for(p=0;p<r;p++)
for(q=0;q<=200;q++)
{
}
}
void send_char(unsigned char arr_ch)
{
SBUF=arr_ch;
while(TI==0);
TI=0;
}
```

7. RESULT

In this project we are acknowledging the users their usage of water as per the timings their water supply connections are setup from the main grid. It includes the graph showing flow rates of different taps to which the sensors are connected. The graph displays flow rates of any tap on y-axis, time slots on x-axis. If there is no water supply the graph is linear. If there is any leakage or usage of water, the graph varies with respective to the flow rate. The following figures shows various graphs of different taps:

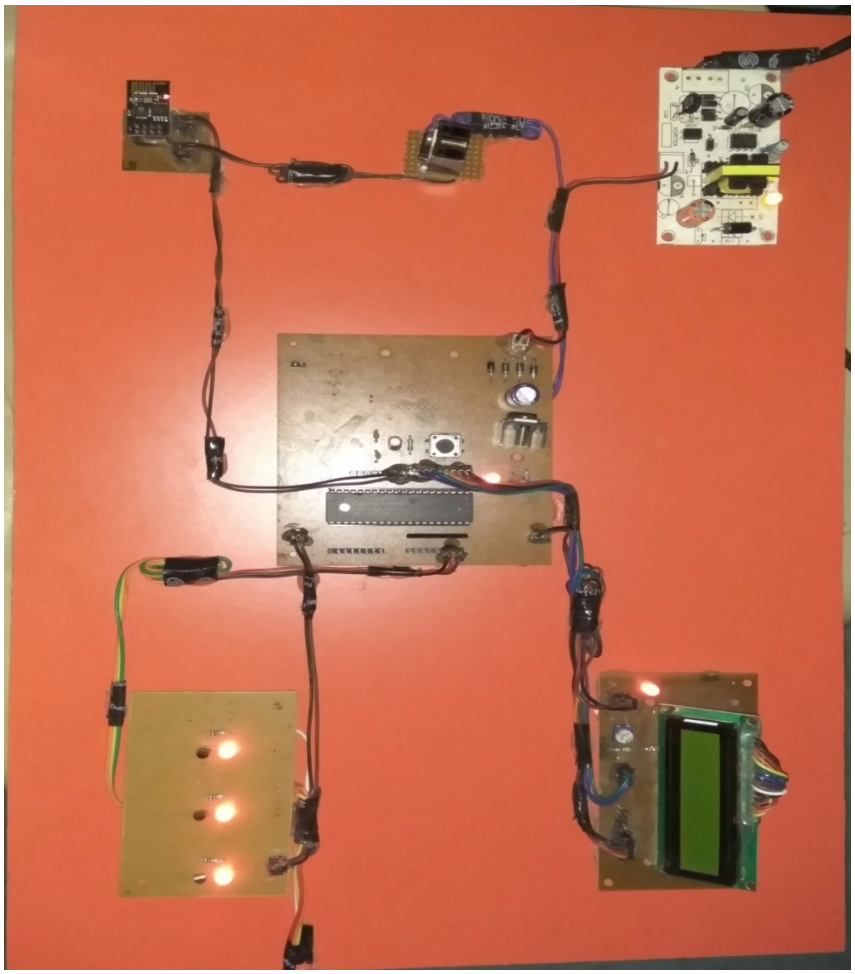


Figure 7.1 complete project kit

Consider each of the taps as pipelines of a grid going in different directions the graphs will show us the daily water flow in each pipe line in terms of graph.

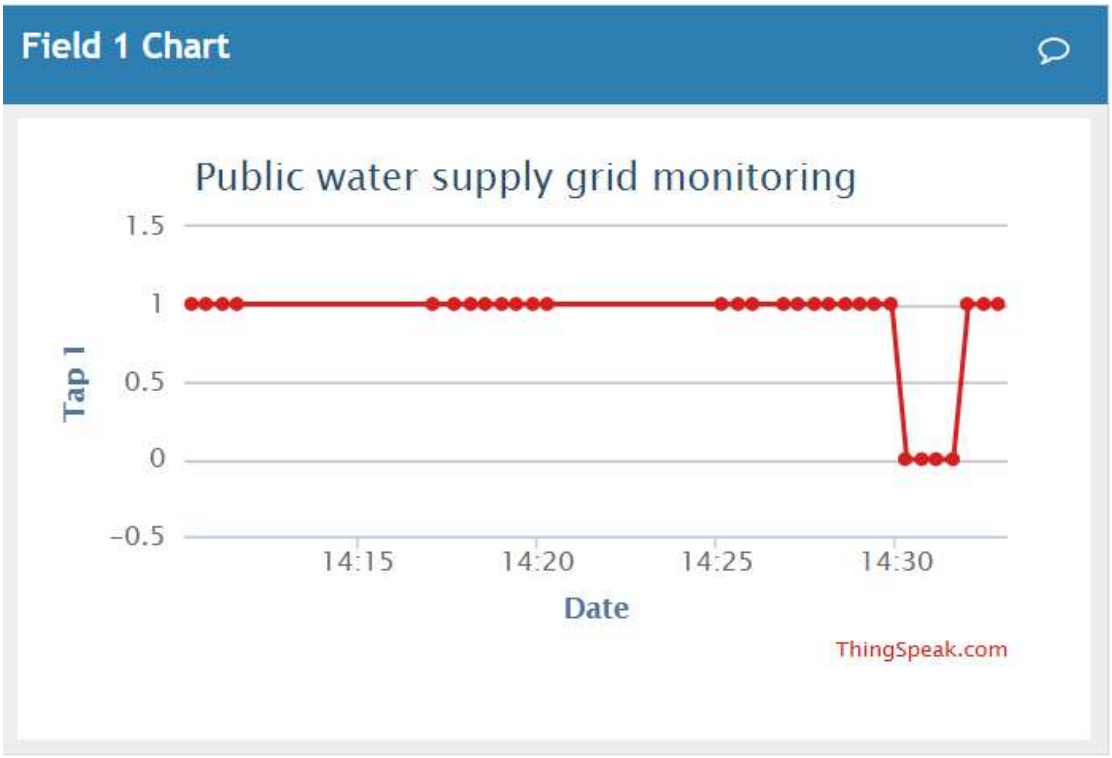


Figure 7.2 Water flow in pipe line 1

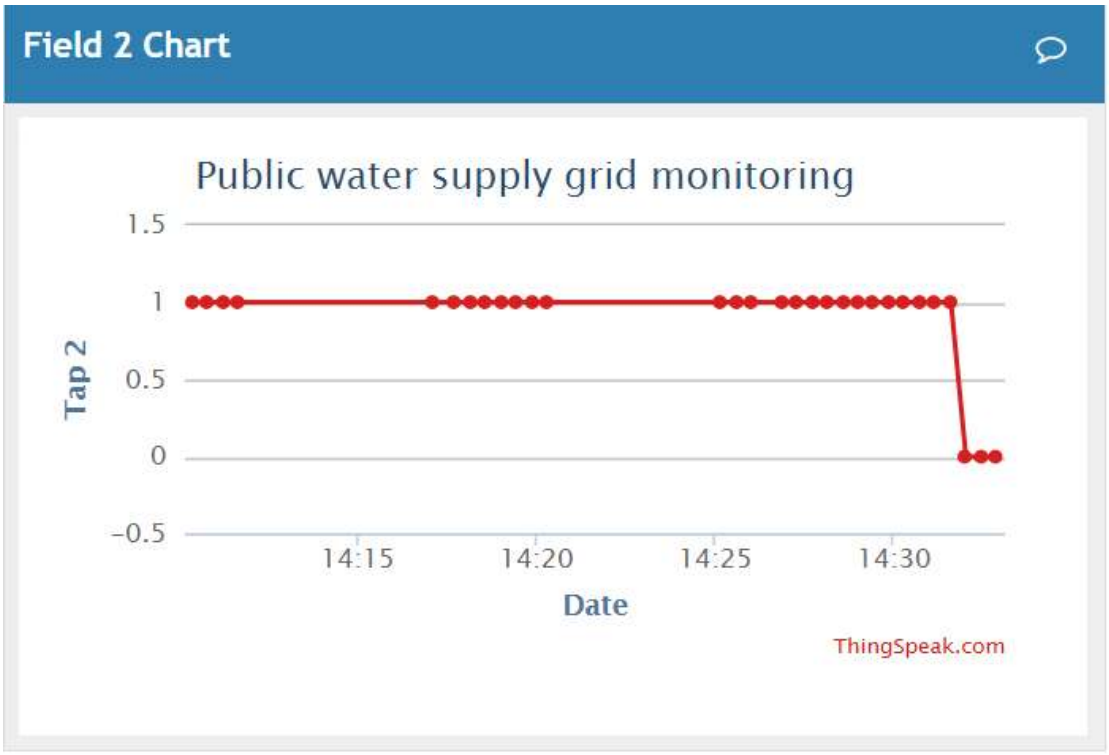


Figure 7.3 Water flow in pipeline 2

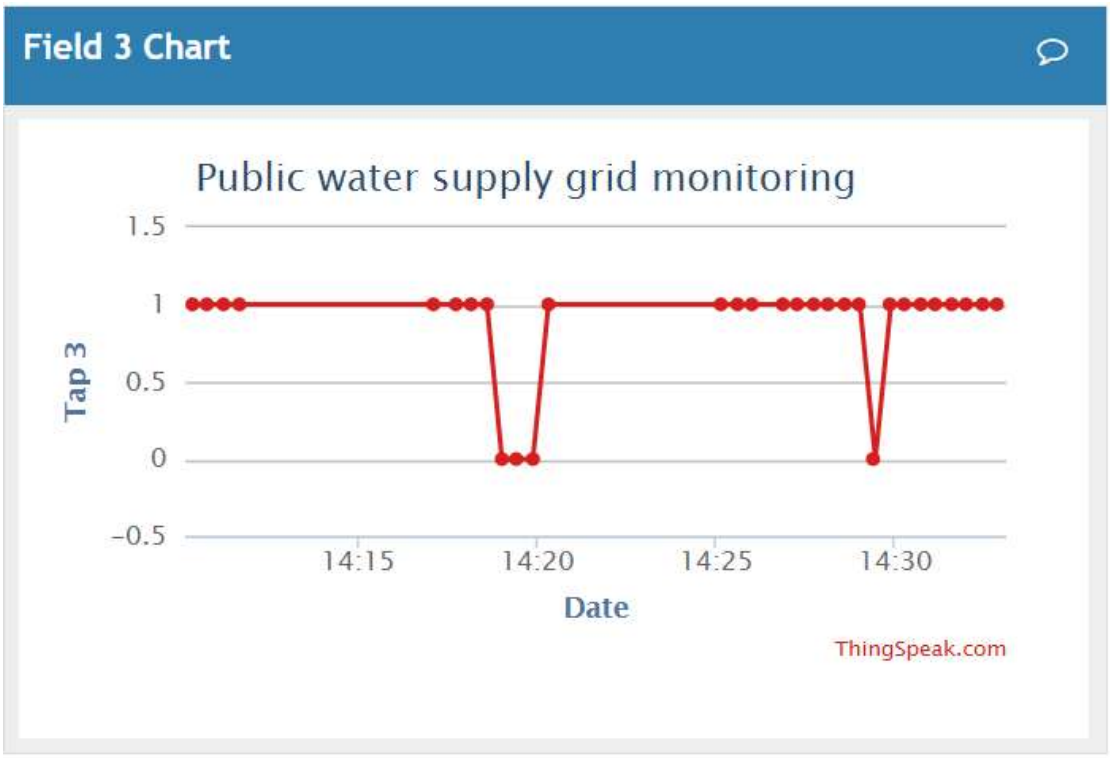


Figure 7.4 Water flow in pipeline 3

8. CONCLUSION

Hence by this project we have designed a system to detect the tampering and leakages in the system. we can monitor the water flow rate through iot module which is updated every 15 minutes. If there is any leakage or water theft, we can easily find as we will get to know about each of the tap connections. By using this project in real time application we can reduce the human effort to the maximum extent. It is easy to install and maintain. Utilities can use this to deploy, integrate and manage the services they choose and pay on a per month, per-meter basis, which is often a more attractive model for smaller cities.

We can assess the damage to the pipe work where the leakage is found. Also identify the resources (human, time and materials) needed to rectify the problem. Customers can know their usage and can demand for billing according to that. By implementing this we use water sensibly.

9. FUTURE SCOPE

Using this system secure and continuous monitoring is possible No need to go on field for monitoring so manual work has reduced it makes system more efficient, reliable, low cost and accurate we can Data monitored from anywhere controlling is possible from a remote server it is Economical in development.

Thus, water supply monitoring and theft detection system was built. Using proposed system, we can make centralized water control and theft detection system. We can ensure fair water supply to all users by preventing water theft and ensuring by taking necessary action. The disadvantage of the existing system that required manpower was eliminated. This real time monitoring implemented in the system avoids wastage of water and reduces time. Due to database, it is possible to monitor the whole system from central office and produce daily, monthly and yearly reports for quantitative analysis of supply water.

REFERENCES

1. Mohammad Rizwan “Leak detection in pipeline system based on flow induced vibration methodology in pipeline International journal of science and research”

2. J. Whittle 1, M. Allen2, A. Preis3 and M. Iqbal3 “Sensor networks for monitoring and control of water distribution system” The 6th International Conference on Structural Health Monitoring of Intelligent Infrastructure Hong Kong 9---11 December 2013 Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, volume: 6, number: 1, pp. 4-23

3. Gouthaman. J, Bharathwajanprabhu. R &Srikanth. A, “Automated urban drinking water supply control and water theft identification system”, Students'

4. Stancel, E, Stoian, I,Kovacs, I, &Gyurka, B.Z, “Urban water supply distributed control system”, IEEE International Conference on Automation,Quality and Testing, Robotics,2008,Vol.3,pp.316-320,2008.

5. MuhammasMazidi, Janice Mazidi, RolinMcKinlay, „The 8051 microcontroller and embedded system using Assembly and C“, PearsonEducation,2nd edition.

6. Technology Symposium,2011 IEEE,14-16 Jan, pp.87-91,2011.

7. DebasisBandyopadhyay · JaydipSen “Internet of Things - Applications and Challenges in Technology and Standardization”

8. http://www.researchgate.net/publication/22257175