**ABSTRACT**

Due to an increase in population, there is an increase in the demand for food, daily essentials and basic resources in the household. This, in turn, has significantly increased the amount of waste. Accumulated waste causes air pollution and causes many diseases. Hence, it is very important to avoid accumulation and spillage of the waste from the dustbins installed in public places. IOT and Arduino based Garbage and Waste collection bins overflow indicator is a system that would notify the waste collecting team when a garbage bin is full and needs to be emptied. The smart garbage bins have Ultrasonic Sensors placed on the lid which detects the garbage level in the bins. By this, the garbage bins can be monitored, and the monitoring information can be obtained through the webpage. The microcontroller controls all the sensors and collects, shares the information given by sensors to IOT. This project can be used in the” SMART CITY”. This project is also helpful in the government project of “SWACHH BHARAT ABHIYAN”.

**Chapter 1**

**INTRODUCTION**

Garbage Monitoring System: - Garbage may consist of the unwanted material left over from City, Public area, Society, College, home etc. This project is related to the “Smart City” and based on “Internet of Things” (IOT). So, for a smart lifestyle, cleanliness is needed, and cleanliness begins with Garbage Bin. This project will help to eradicate or minimize the garbage disposal problem. The Internet of Things (IoT) is a recent communication paradigm that envisions near future, in which the objects of everyday life will be equipped with microcontrollers, transceivers for digital communication, and suitable protocol stacks that will make them able to communicate with one another and with the users, becoming an integral part of the Internet.

This project IOT Garbage Monitoring system is a very innovative system which will help to keep the cities clean. This system monitors the garbage bins and informs about the level of garbage collected in the garbage bins via a web page. For this the system uses ultrasonic sensors placed over the bins to detect the garbage level and compare it with the garbage bin's depth. The system makes use of an Arduino family microcontroller, LCD screen, Wi-Fi modem for sending data and a buzzer. The system is powered by a 12V transformer. The LCD screen is used to display the status of the level of garbage collected in the bins.

Whereas a web page is built to show the status to the user monitoring it. The web page gives a graphical view of the garbage bins and highlights the garbage collected in colour in order to show the level of garbage collected. The LCD screen shows the status of the garbage level. The system puts on the buzzer when the level of garbage collected crosses the set limit. Thus, this system helps to keep the city clean by informing about the garbage levels of the bins by providing graphical image of the bins via a web page.

**Chapter 2**

**LITERATURE REVIEW**

**2.1**  **S. Ramesh, A. Usman and B. P. Divakar, “Municipal solid waste management in Bangalore and the concept of mini biogas plant in urban localities,” in the year 2019.**

In this paper they discussed various case studies piloted in some of the urban areas on solid wastes in India. In order to understand the waste management problems being faced by Bangaloreans, a few data sets connected to these issues have been collected. A special survey is also piloted in one of the best engineering campuses, Reva Institute of Technology, Bangalore to measure the waste being produced inside the campus.

**2.2**  **S. Ankitha, K. B. Nayana, S. R. Shravya and L. Jain, “Smart city initiative: Traffic and waste management,” 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, 2017.**

S. Ankitha, K. B. Nayana, S. R. Shravya and et.al addressed issues related to traffic and waste management in smart cities. The proposed work supports traffic management by making use of Infra-red sensors which are installed in the traffic signal spot just ahead of the stop line for the vehicles. Whenever a vehicle crosses the red signal, sensors get activated and the vehicle with radio frequency identification number is recorded. The identification number is then checked in the vehicle database available through which vehicle owner information is retrieved. An android application is used to send fine notice to the owner of the vehicle. Waste management is taken care of through the help of sensor systems which detect the level of waste garbage in the dustbins and sends the status to authorized control room through GSM system and also location of the dustbin is tracked through GPS.

**2.3**  **S. S, R. R, R. R and S. K. A, "Smart Bin For Waste Management System," 2018 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 2018.**

Sreejith S, Ramya R, Roja R, Sanjay Kumar et.al proposed a system for waste monitoring and named it as Smart bin which is automated to detect the level of waste in public dustbins and disposes the same once threshold level is reached. As we all have experienced, the dustbins in public are overfilled causing odors and resulting in various infections and diseases. Thus, the system helps in avoiding overflowing of waste in dustbins.

**2.4**  **A. S Bharadwaj, R. Rego and A. Chowdhury, “IoT based solid waste management system: A conceptual approach with an architectural solution as a smart city application,” 2020 IEEE Annual India Conference (INDICON), Bangalore, 2020,**

SLR method is used o gather information regarding proposed and existing solutions of SSWMS. There are two sensors that are used Proxy sensor, Level sensor. Getting more knowledge about solid waste management using systematic literature review methods, designed for smart cities, Limited sensors are used.

**Chapter 3**

**PROBLEM AND SOLUTION OF THE PROJECT**

**3.1 EXISTING SYSTEM:**

* The city does not have any infrastructure to collect and transport the waste and there is no specified location for the waste disposal.
* Most of the respondents dump their biodegradable waste (68.04%) and nonbiodegradable (67.71%) waste in dustbins which is ultimately thrown in the open fields. A very less proportion of households (1.48%) dispose bio-degradable waste in a safe manner.
* At present, there is no collection and disposal mechanism. The figure below shows the current waste generation and the present methods used for disposal of the waste.
* Safe disposal of bio-degradable waste includes disposing solid waste in an identified place, composting, burying, re-using in the garden and having the GP collect the waste and feed it to the cattle.
* As far as the volume of waste generated is concerned, there is no specific measured data that is available. The national waste generation average can be taken as a basis for designing a management solution for solid waste disposed from the area

**DISADVANTAGES:**

* Time-consuming and less effective.
* Unhygienic environment.
* Bad smell spreads and may cause illness in human beings.
* More traffic and noise**.**

**3.2 PROPOSED SYSTEM:**

* We propose a smart waste collection system for the level of wastes present in the waste bins.
* The data obtained through sensors is transmitted over the Internet to a server for storage and processing mechanisms.
* Sensors: We can determine the waste level by measuring the distance from the top of the trash Bin to the waste by sonar.
* Sensing and data forwarding rates, and wireless technology used have a strong influence on energy consumption. Collection and forwarding of data can be done once or twice a day.
* It is used for monitoring the daily selection of wastebins, based on which the routes to pick several of the wastebins from different locations are decided.
* Predict future state with respect to factors like traffic congestion in an area where the wastebins are placed, cost-efficiency balance, and other factors that is difficult for humans to observe and analyze.
* It can be predicted before the overflow of waste occurs in the wastebins that are placed in a specific location.
* Depending on economic requirements specified at early stages, the optimized selection of wastebins to be collected is expected

**ADVANTAGES:**

* Monitors the garbage bins and informs about the level of garbage collected in the garbage bins.
* To keep our environment clean and green.
* The cost and effort are less in this system.
* Decrease the traffic flow and noise.
* Reduce manpower.

**Chapter 4**

**SYSTEM STUDY**

**FEASIBILITY STUDY:**

The feasibility of the project is analyzed in this phase and the business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**ECONOMICAL FEASIBILITY:**

This study is carried out to check the economic impact that the system will have on the organization. The amount of funds that the company can pour into the research and development of the system is limited. The expenditure must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

**TECHNICAL FEASIBILITY:**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. No system developed must not have a high demand for the available technical resources. This will lead to high demand for the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**SOCIAL FEASIBILITY:**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity.

The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**COST ESTIMATING & SCHEDULING**

In developing cost estimates for a system several cost elements are considered.

1. Personnel Cost

2. Facility Cost

3. Operating Cost

**Personnel Cost:**

Personnel costs include staff salaries and benefits as well as pay for those involved in developing the system.

**Facility cost:**

Facility cost is expenses incurred in the preparation of the physical site where the application or computer will be in operation.

**Operating cost:**

Operating costs include all costs associated with the day-to-day operation of the system. This project is not costly. It includes only hardware cost for purchase of modem, hub and cable

**Chapter 5**

**SYSTEM ANALYSIS**

**SYSTEM REQUIREMENTS:**

**5.1 H/W SYSTEM CONFIGURATION**:

* PIC16F877A / Node MCU.
* Load cell.
* Moisture sensor.
* Load level sensor.
* Relay.
* UART Cable.
* GPRS modem.

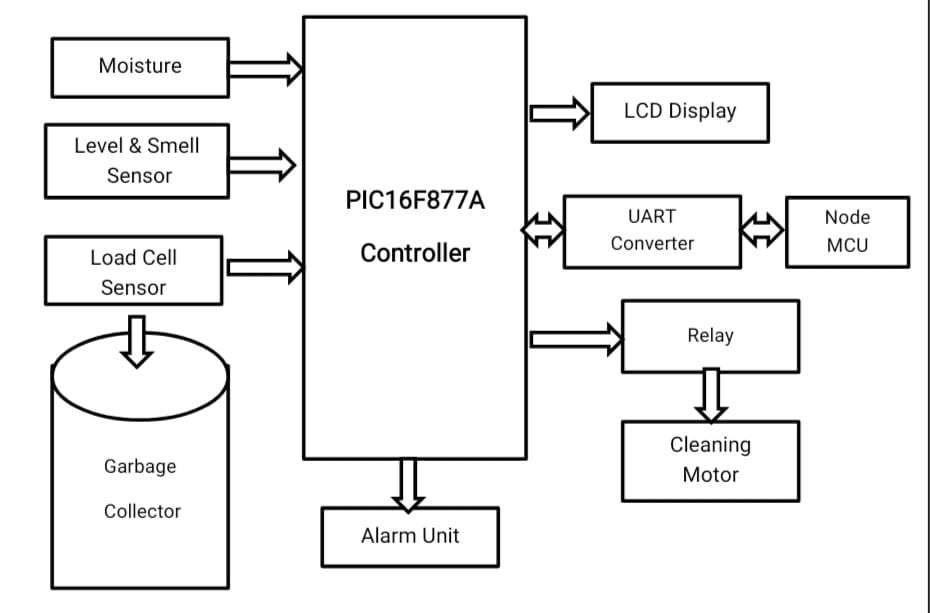
**5.2 S/W SYSTEM CONFIGURATION: -**

* MP LAB IDE.
* Pickit-3.

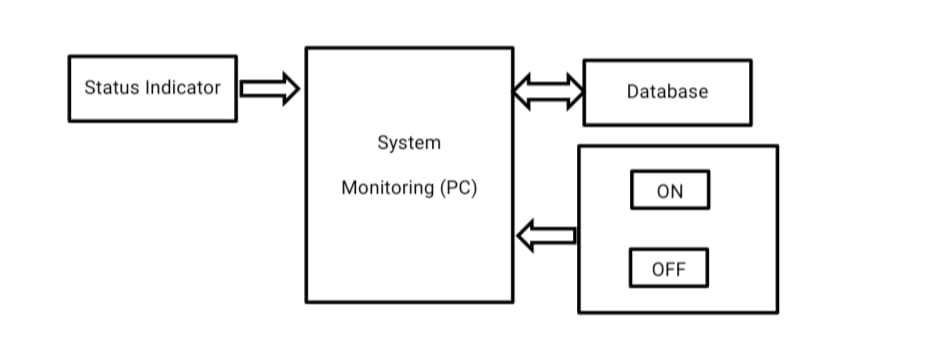
**Chapter 6**

**SYSTEM DESIGN**

**6.1 ARCHITECTURE**:



**IOTsection(pc):**



**6.2 MODULES**:

* **Sensors Architecture:**

Power supply is used to supply the overall power to the sensors and microcontroller. Read the values from the sensors such as Moisture sensor, Humidity sensor, CO sensor, pressure sensor, smoke sensor and using microcontroller. Moisture sensor used to detect the liquid wastages Humidity sensor used to detect the solid wastages CO sensor used to detect Emissions.

* **Garbage Classification:**

This system a dry waste a wet waste separately for that we are using a moisture sensor if that sensor detected then the cap will open for dry waste.

* **Dustbin level detection:**

We propose a smart garbage bin using cloud IOT based controller to identify when the garbage bin is being filled using level sensor, we can get the volume occupied and left in the smart garbage bin. level sensors provide information on an absolute position of target or moving object. For glossy surfaces, transparent objects or in environments with high degree of dust and humidity,

* **Information Sending:**

If the volume is full then the program triggers and alert message through IOT and sends an alert and location of the bin to collect the garbage ‘s. The garbage collector collects the waste and empties the bin for info sending, we will use an IOT Environment.

**6.3 UML DIAGRAMS:**

Uml stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. UML is a very important part of developing objects-oriented software and the software development process. UML uses mostly graphical notations to express the design of software projects.

**GOALS:**

The Primary goals in the design of the UML are as follows:

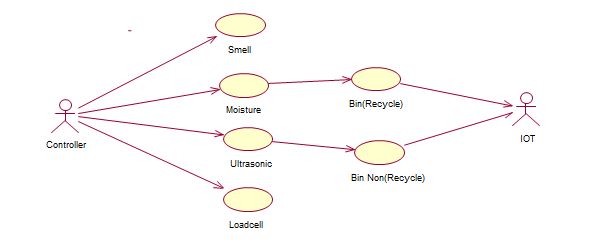
1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.

2. Provide extendibility and specialization mechanisms to extend the core concepts.

3. Be independent of programming languages and development process

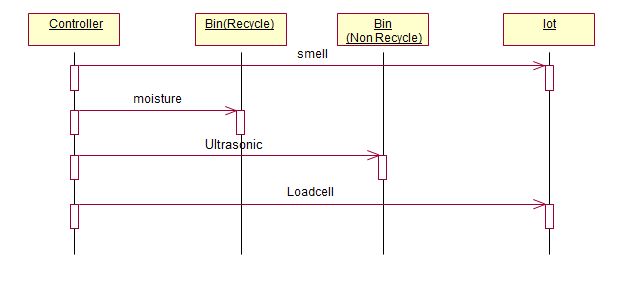
**6.3.1** **USE CASE DIAGRAM**:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



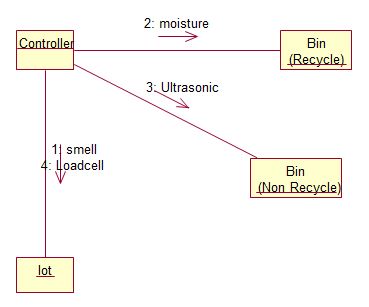
**6.3.2** **SEQUENCE DIAGRAM:**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams



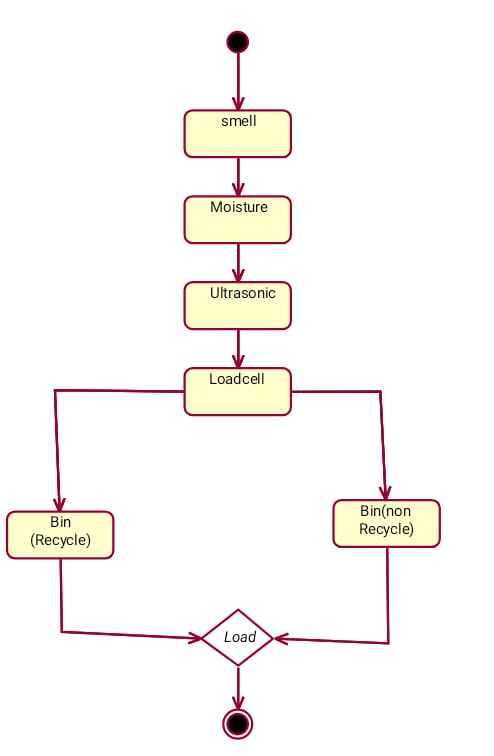
**6.3.3 COLLABORATION DIAGRAM:**

In the collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have used the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization



**6.3.4 ACTIVITY DIAGRAM**:

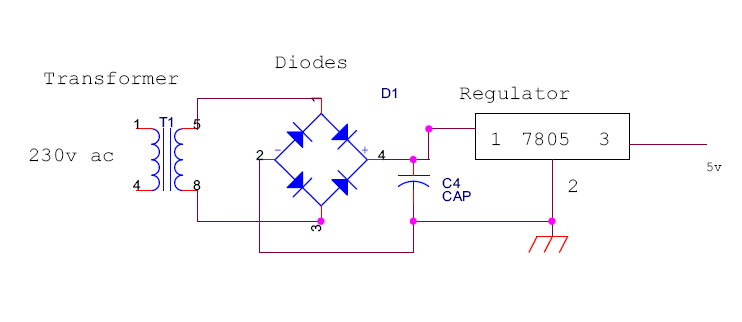
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control



**Chapter 7**

**SOFTWARE ENVIRONMENT**

# 7.1 POWER SUPPLY UNIT



Power supply unit consists of following units**:**

1) Step down transformer

2) Rectifier unit

3) Input filter

4) Regulator unit

5) Output filter

**7.1.1 Stepdown transformer:**

The Step-down Transformer is used to step down the main supply voltage from 230V AC to a lower value. This 230 AC voltage cannot be used directly; thus, it is stepped down. The Transformer consists of primary and secondary coils. To reduce or step down the voltage, the transformer is designed to contain a smaller number of turns in its secondary core. The output from the secondary coil is also AC waveform. Thus, the conversion from AC to DC is essential. This conversion is achieved by using the Rectifier Circuit/Unit.

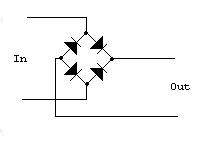
Step down transformers can step down incoming voltage, which enables you to have the correct voltage input for your electrical needs.  For example, if our equipment has been specified for an input voltage of 12 volts, and the main power supply is 230 volts, we will need a *step-down transformer*, which decreases the incoming electrical voltage to be compatible with your 12-volt equipment.

**7.1.2 RECTIFIER UNIT:**

The Rectifier circuit is used to convert the AC voltage into its corresponding DC voltage. The most important and simple device used in the Rectifier circuit is the diode. The simple function of the diode is to conduct when forward biased and not to conduct in reverse bias. Now we are using three types of rectifiers. They are

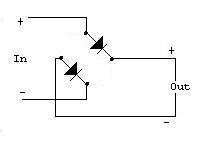
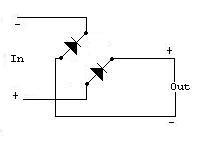
1. Half-wave rectifier
2. Full-wave rectifier
3. Bridge rectifier

* **Half-wave rectifier:** In half wave rectification, either the positive or negative half of the AC wave is passed, while the other half is blocked. Because only one half of the input waveform reaches the output, it is very inefficient if used for power transfer. Half-wave rectification can be achieved with a single diode in a one phase supply, or with three diodes in a three-phase supply.
* **Full-wave rectifier**: A full-wave rectifier converts the whole of the input waveform to one of constant polarity (positive or negative) at its output. Full-wave rectification converts both polarities of the input waveform to DC (direct current), and is more efficient. However, in a circuit with a non-center tapped transformer, four diodes are required instead of the one needed for half-wave rectification. A full-wave rectifier uses a diode bridge, made of four diodes, like this



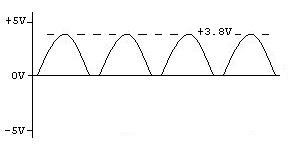
At first, this may look just as confusing as the one-way streets of Boston. The thing to realize is that the diodes work in pairs. As the voltage of the signal flips back and forth, the diodes shepard the current to always flow in the same direction for the output.

Here's what the circuit looks like to the signal as it alternates:

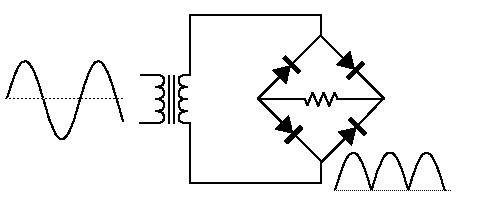
So, if we feed our AC signal into a full wave rectifier, we'll see both halves of the wave above 0 Volts. Since the signal passes through two diodes, the voltage out will be lower by two diode drops, or 1.2 Volts.

AC Wave In:

AC Wave Out (Full-Wave Rectified): 

If we're interested in using the full-wave rectifier as a DC power supply, we'll add a smoothing capacitor to the output of the diode bridge.

* **Bridge rectifier:** A bridge rectifier makes use of four diodes in a bridge arrangement to achieve full-wave rectification. This is a widely used configuration, both with individual diodes wired as shown and with single component bridges where the diode bridge is wired internally.



A **diode bridge** or **bridge rectifier** is an arrangement of four [diodes](http://en.wikipedia.org/wiki/Diode) in a [bridge](http://en.wikipedia.org/wiki/Bridge_circuit) configuration that provides the same [polarity](http://en.wikipedia.org/wiki/Polarity_(physics)) of output [voltage](http://en.wikipedia.org/wiki/Volt) for either polarity of input voltage. When used in its most common application, for conversion of [alternating current](http://en.wikipedia.org/wiki/Alternating_current) (AC) input into [direct current](http://en.wikipedia.org/wiki/Direct_current) (DC) output, it is known as a bridge [rectifier](http://en.wikipedia.org/wiki/Rectifier). A bridge rectifier provides [full-wave rectification](http://en.wikipedia.org/wiki/Rectifier) from a two-wire AC input, resulting in lower cost and weight as compared to a [center-tapped](http://en.wikipedia.org/wiki/Center_tap) [transformer](http://en.wikipedia.org/wiki/Transformer) design.

The Forward Bias is achieved by connecting the diode’s positive with positive of the battery and negative with battery’s negative. The efficient circuit used is the Full wave Bridge rectifier circuit. The output voltage of the rectifier is in rippled form, the ripples from the obtained DC voltage are removed using other circuits available. The circuit used for removing the ripples is called Filter circuit.

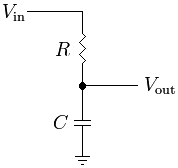
**7.1.3 Input Filter:**

Capacitors are used as filter. The ripples from the DC voltage are removed and pure DC voltage is obtained. And also these capacitors are used to reduce the harmonics of the input voltage. The primary action performed by capacitor is charging and discharging. It charges in positive half cycle of the AC voltage and it will discharge in negative half cycle. So it allows only AC voltage and does not allow the DC voltage. This filter is fixed before the regulator. Thus the output is free from ripples.

There are two types of filters. They are

1. Low pass filter
2. High pass filter

* **Low pass filter:**



One simple [electrical circuit](http://en.wikipedia.org/wiki/Electrical_circuit) that will serve as a low-pass filter consists of a [resistor](http://en.wikipedia.org/wiki/Resistor) in series with a [load](http://en.wikipedia.org/wiki/External_electric_load), and a [capacitor](http://en.wikipedia.org/wiki/Capacitor) in parallel with the load. The capacitor exhibits [reactance](http://en.wikipedia.org/wiki/Reactance_(electronics)), and blocks low-frequency signals, causing them to go through the load instead. At higher frequencies the reactance drops, and the capacitor effectively functions as a short circuit. The combination of resistance and capacitance gives you the [time constant](http://en.wikipedia.org/wiki/Time_constant) of the filter τ = *RC* (represented by the Greek letter [tau](http://en.wikipedia.org/wiki/Tau)). The break frequency, also called the turnover frequency or [cutoff frequency](http://en.wikipedia.org/wiki/Cutoff_frequency) (in hertz), is determined by the time constant: or equivalently (in [radians](http://en.wikipedia.org/wiki/Radians) per second):

One way to understand this circuit is to focus on the time the capacitor takes to charge. It takes time to charge or discharge the capacitor through that resistor:

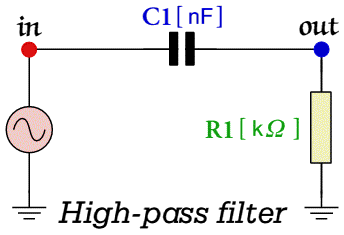
* At low frequencies, there is plenty of time for the capacitor to charge up to practically the same voltage as the input voltage.
* At high frequencies, the capacitor only has time to charge up a small amount before the input switches direction. The output goes up and down only a small fraction of the amount the input goes up and down. At double the frequency, there's only time for it to charge up half the amount.

Another way to understand this circuit is with the idea of [reactance](http://en.wikipedia.org/wiki/Reactance_(electronics)) at a particular frequency:

* Since [DC](http://en.wikipedia.org/wiki/Direct_current) cannot flow through the capacitor, DC input must "flow out" the path marked *V*out (analogous to removing the capacitor).
* Since [AC](http://en.wikipedia.org/wiki/Alternating_current) flows very well through the capacitor — almost as well as it flows through solid wire — AC input "flows out" through the capacitor, effectively [short circuiting](http://en.wikipedia.org/wiki/Short_circuit) to ground (analogous to replacing the capacitor with just a wire).

It should be noted that the capacitor is not an "on/off" object (like the block or pass fluidic explanation above). The capacitor will variably act between these two extremes. It is the [Bode plot](http://en.wikipedia.org/wiki/Bode_plot) and [frequency response](http://en.wikipedia.org/wiki/Frequency_response) that show this variability.

* **High pass filter:**



The above circuit diagram illustrates a simple *'RC'* high-pass filter. we should find that the circuit passes 'high' frequencies fairly well, but attenuates 'low' frequencies. Hence it is useful as a filter to block any unwanted low frequency components of a complex signal whilst passing higher frequencies. Circuits like this are used quite a lot in electronics as a 'D.C. Block' - i.e. to pass a.c. signals but prevent any D.C. voltages from getting through.  
  
The basic quantities which describe this circuit are similar to those used for the [Low Pass Filter](http://www.st-andrews.ac.uk/~www_pa/Scots_Guide/experiment/lowpass/lpf.html). In effect, this circuit is just a simple low-pass filter with the components swapped over.

http://www.st-andrews.ac.uk/~www_pa/Scots_Guide/experiment/highpass/tau.gif

The action of the circuit can also be described in terms of a related quantity, the *Turn over Frequency*, *f0*, which has a value

http://www.st-andrews.ac.uk/~www_pa/Scots_Guide/experiment/highpass/f0.gif

As with the low-pass filter, the circuit's behavior we can be understood as arising due to the time taken to change the capacitor's charge when we alter the applied input voltage. It always takes a finite (i.e. non-zero) time to change the amount of charge stored by the capacitor. Hence it takes time to change the potential difference across the capacitor. As a result, any sudden change in the input voltage produces a similar sudden change on the other side of the capacitor. This produces a voltage across the resistor and causes a current to flow thorough it, charging the capacitor until all the voltage falls across it instead of the resistor. The result is that steady (or slowly varying) voltages appear mostly across the capacitor and quick changes appear mostly across the resistor. Since we're using the voltage across the resistor as out output the main properties of the circuit are

Therefore  
  
The *Voltage Gain*:

http://www.st-andrews.ac.uk/~www_pa/Scots_Guide/experiment/highpass/av.gif

The *Phase Delay*:

http://www.st-andrews.ac.uk/~www_pa/Scots_Guide/experiment/highpass/phi.gif

Try using the above experimental system to collect results and plot a graph of how the voltage gain, *Av*, (and the phase change) depend upon the input frequency and if we check result agrees with the above formulae. Compare this with a low-pass filter that uses the same component values, and you should see that they give 'opposite' results.

**7.1.4 REGULATOR UNIT**

****

7805 Regulator

Regulator regulates the output voltage to be always constant. The output voltage is maintained irrespective of the fluctuations in the input AC voltage. As and then the AC voltage changes, the DC voltage also changes. Thus, to avoid this Regulators are used. Also, when the internal resistance of the power supply is greater than 30 ohms, the output gets affected. Thus, this can be successfully reduced here. The regulators are mainly classified for low voltage and for high voltage. Further they can also be classified as:

i) Positive regulator

1---> input pin

2---> ground pin

3---> output pin

It regulates the positive voltage.

ii) Negative regulator

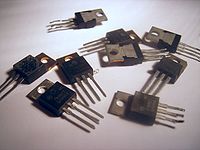
1---> ground pin

2---> input pin

3---> output pin

It regulates the negative voltage.

**Fixed regulators:**

[](http://en.wikipedia.org/wiki/File:7800_IC_regulators.jpg)

An assortment of [78xx](http://en.wikipedia.org/wiki/78xx) series ICs

"Fixed" three-terminal linear regulators are commonly available to generate fixed voltages of plus 3 V, and plus or minus 5 V, 9 V, 12 V, or 15 V when the load is less than about 7 [amperes](http://en.wikipedia.org/wiki/Ampere).

## 7805 VOLTAGE REGULATOR:

## The 7805 provides circuit designers with an easy way to regulate DC voltages to 5v. Encapsulated in a single chip/package (IC), the 7805 is a positive voltage DC regulator that has only 3 terminals. They are Input voltage, Ground, Output Voltage.

**General 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

**7812 12V Integrated Circuit3-Terminal Positive Voltage Regulator:**

* The 7812 fixed voltage regulator is a monolithic integrated circuit in a TO220 type package designed for use in a wide variety of applications including local, onboard regulation. This regulator employs internal current limiting, thermal shutdown, and safe area compensation.
* With adequate heatsinking it can deliver output currents in excess of 1.0 ampere. Although designed primarily as a fixed voltage regulator, this device can be used with external components to obtain adjustable voltages and currents.

**7.1.5 Output Filter:**

The Filter circuit is often fixed after the Regulator circuit. Capacitor is most often used as filter. The principle of the capacitor is to charge and discharge. It charges during the positive half cycle of the AC voltage and discharges during the negative half cycle. So, it allows only AC voltage and does not allow DC voltage. This filter is fixed after the Regulator circuit filters any of the possibly found ripples in the output received finally. Here we used 0.1µF capacitor. The output at this stage is 5V and is given to the Microcontroller. The output voltage overshoots when the load is removed or a short clear. When the load is removedfrom a switching mode power supply with a LC low-pass output filter, the only thing the control loop can do is stop the switching action so no more energy is taken from the source. The energy that is stored in the output filter inductor is dumped into the output capacitor causing a voltage overshot.

The magnitude of the overshoot is the vector sum of two orthogonal voltages, the output voltage before the load is removed and the current through the inductor times the characteristic impedance of the output filter, Zo = (L/C) ^1/2. This can be derived from conservation of energy considerations.

The initial energy, Ei, is:

Ei = 1/2\*(L\*Ii^2 + C\*Vi^2)

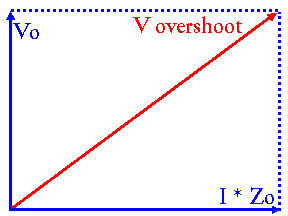
The final energy, Ef, is:

Ef = 1/2\*(L\*If^2 = C\*Vf^2)

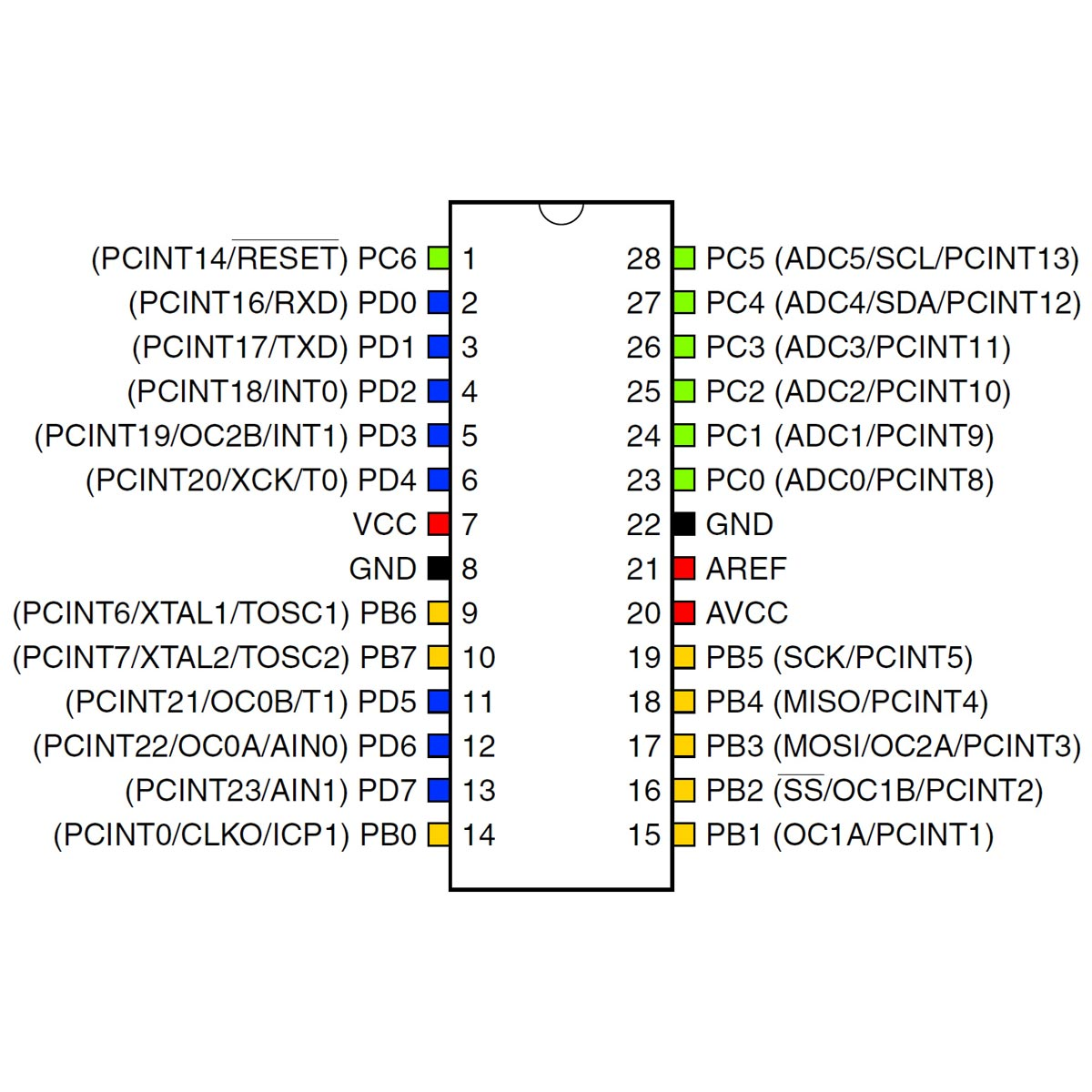
The two energies are equal when the load is removed, since the load is no longer taking energy from the system. Equating the two energies, substituting zero current for the final inductor current, then the solution for the final voltage Vf is:

Vf = (Vi^2 + (Ii\*Zo)^2)^1/2

This is the orthogonal vector sum of the output voltage and the load current times the characteristic impedance and is illustrated in Figure 1.

  
Figure 1: Overshoot Voltage as Vector Sum

The problem becomes worse if the current in the inductor is established by a short circuit on the output and the short circuit clears. In this case, the initial voltage is zero (short circuit) and the overshoot is I\*Zo, where I can be very large, resulting in a ruinous overshooot

**PIN DIAGRAM** 

ATMEGA 328 microcontroller, which acts as a processor for the arduino board. Nearly it consists of 28 pins. From these 28 pins, the inputs can be controlled by transmitting and receiving the inputs to the external device. It also consists of pulse width modulation (PWM). These PWM are used to transmit the entire signal in a pulse modulation. Input power supply such as Vcc and Gnd are used. These IC mainly consists of analog and digital inputs. These analog and digital inputs are used for the process of certain applications.

The power pins are as follows:

* VIN: The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
* 5V: The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
* 3V3: A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
* GND: Ground pins.
* AREF. Reference voltage for the analog inputs. Used with analogReference().
* Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

**Analog Input**:

Arduino atmega-328 microcontroller board consist of 6 analog inputs pins. These analog inputs can be named from A0 to A5. From these 6 analog inputs pins, we can do the process by using analog inputs. Analog inputs can be used in the operating range of 0 to 5V. Analog signal is considered as the continuous time signal, from which these analog signal can be used for certain applications. These are also called as non-discrete time signal. Inputs such as voltage, current etc.., are considered to beeither analog signal or digital signal only by analysing thetime signal properties. Various applications of arduinomicrocontroller can use only an analog input instead of digital inputs. For these applications, analog input ports or pins can be used.

**Digital Input:**

Digital inputs can be defined as the non-continuous time signal with discrete input pulses. It can be represented as 0’s and 1’s. These digital inputs can be either on state or in off state. Arduino atmega328 microcontroller also consists of 12 digital input pins. It can be stated as D0 to D11. Nearly 12 inputs can be used for digital input/output applications. The working of the digital input ports is where the discrete input pulses can be triggered and supplied to the ports. These ports receive the input and therefore the port can be used for both input and output process. These digital pins can access only the digital inputs.

**Power Jack Cable / USB PORT:**

This Arduino atmega-328 microcontroller can be interfaced with the other electronic devices such as computer by using USB port or power jack cable from these power jack cable, we can upload the program of Arduino for their applications. At first, the program can be initialised or can be edited by using Arduino software tools. Then these programs can be transferred through arduino microcontroller board by using usb cable or power jack cable.

**Power Supply:**

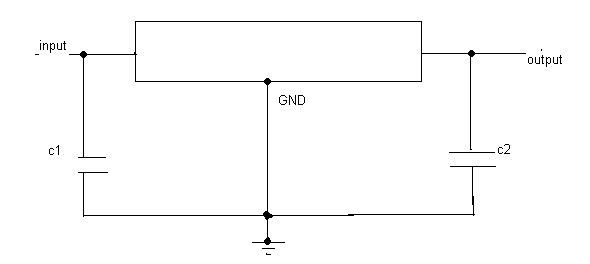
There is an additional power supply source present in Arduino microcontroller. Power supply port is present at the corner of the arduino microcontroller. Either we can use this power supply port by connecting with external power supply.(ie, ac power supply), or by connecting an dc power supply through input pins. These power supplies produce an active form to the arduino microcontroller. These arduino microcontrollers can accept a range of power supply. When the power supply voltage rangeexceeds, the microcontroller gets damaged. Hence, onlythe particular range of power supply should be given to thearduino microcontroller.

**Regulator unit: -**

The regulator regulates the output voltage to a specific value. The output voltage is maintained irrespective of the fluctuations in the input dc voltage. Whenever there are any ac voltage fluctuations, the dc voltage also changes, and to avoid this regulator are used.

**Regulators can be classified as: -**

1. **Positive regulator, which regulates the positive voltage(7805,7812)**



1**. >**input pin

2. > ground pin

3. > output pin

1. **Negative regulator, which regulates the negative voltage (7912).**

1. > ground pin

2. > input pin

3. > output pin

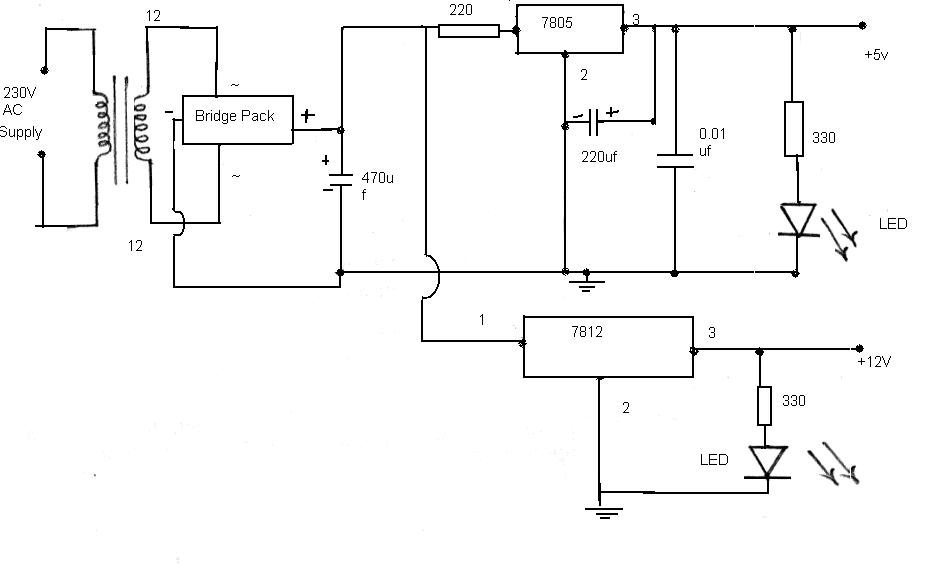
**Regulators used in this application are**: -

7805 which provides 5v dc

7812 which provides 12v dc

7912 which provides -121v dc

**Power Supply Circuit Diagram:**



**7.2 SENSORS:**

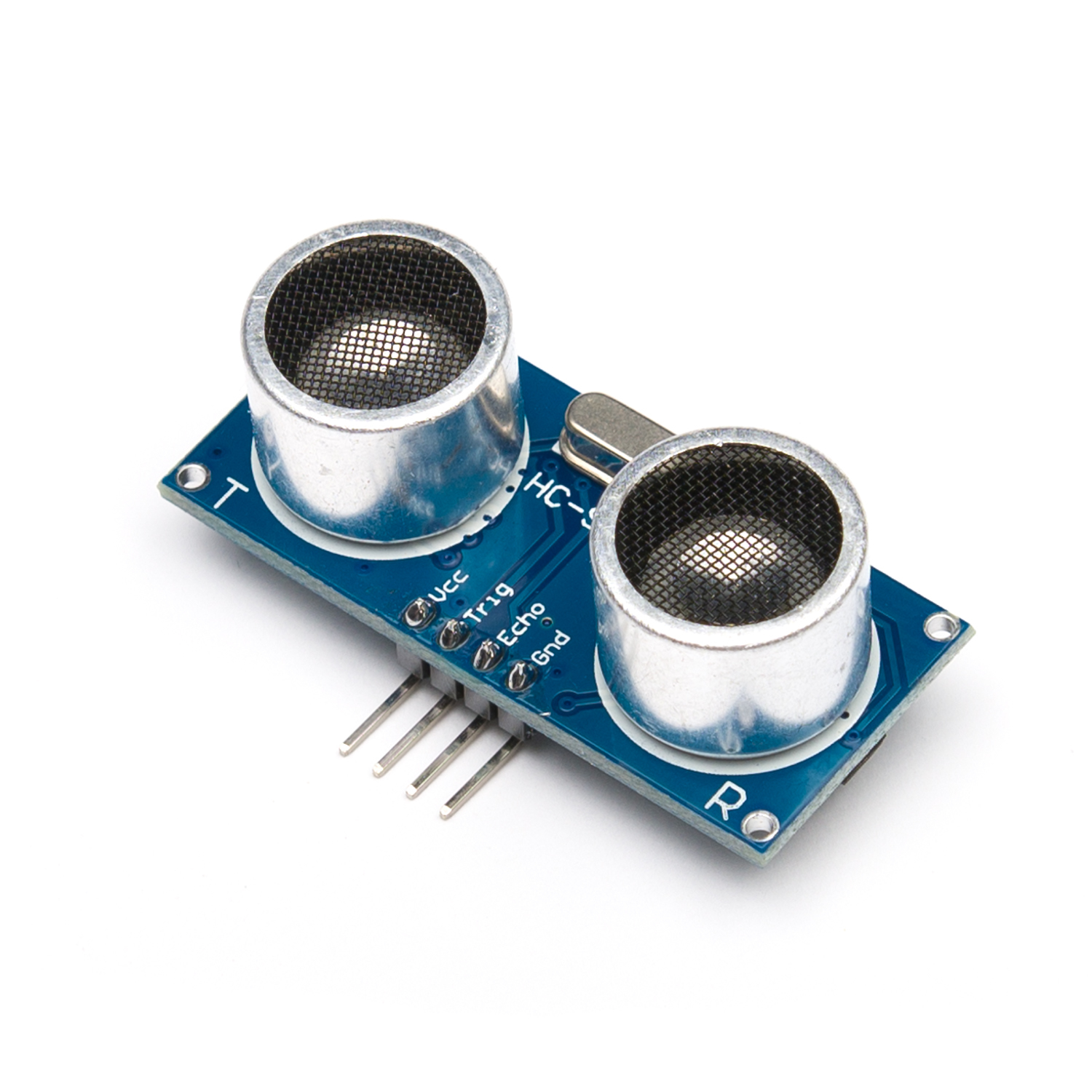
**7.2.1 ULTRASONIC SENSOR :**

* **GENERAL DESCRIPTION :**

Ultrasonic sensors emit ultrasonic pulses, and by measuring the time of ultrasonic pulse reaches the object and back to the transducer. The sonic waves emitted by the transducer are reflected by an object and received back in the transducer. After having emitted the sound waves, the ultrasonic sensor will switch to receive mode. The time elapsed between emitting and receiving is proportional to the distance of the object from the sensor.

* **PRODUCT DESCRIPTION:**

Ultrasonic transmitter emitted an ultrasonic wave in one direction and started timing when it launched. Ultrasonic spread in the air and would return immediately when it encountered obstacles on the way. At last, the ultrasonic receiver would stop timing when it receives the reflected wave. The distance of sensor from the target object is calculated.



**ULTRASONIC SENSOR**

**FEATURES :**

* Working Voltage: 5VDC
* Quiescent Current: <2mA
* Working Current: 15mA
* Detecting Range: 2cm - 4.5m
* Trigger Input Pulse width: 10uS

**APPLICATIONS:**

* Robot navigation
* Obstacle avoidance
* Engineering measurement tools

**7.2.2 WEIGHT RCE SENSOR :**

**GENERAL DESCRIPTION :**

This is a force sensitive resistor with a round, 0.5" diameter, sensing area. This FSR will vary its resistance depending on how much pressure is being applied to the sensing area. The harder the force, the lower the resistance.

**PRODUCT DESCRIPTION:**

This force sensitivity is optimized for use in human touch control of electronic devices such as automotive electronics, medical systems, and in industrial and robotics applications. The standard 400 sensor is a round sensor 7.62mm (about 0.3 in) in diameter.

****



**Loadcell sensor**

**FEATURES**

* Overall length: 2.375"
* Sensing diameter: 0.5"
* Cost effective
* Ultra-thin; 0.35mm
* Robust; up to 10M actuations

**APPLICATIONS**

* Use force for UI feedback
* Enhance tool safety
* Detect presence, position, or motion
* Relative change in force or applied load
* Rate of change in force

**7.2.3 16×2 LCD**

**GENERAL DESCRIPTION:**

LCD stands for liquid crystal display. They come in many sizes 8x1 , 8x2 , 10x2 , 16x1 , 16x2 , 16x4 , 20x2 , 20x4 ,24x2 , 30x2 , 32x2 , 40x2 etc . Many multinational companies like Philips Hitachi Panasonic make their own special kind of LCD'S to be used in their products. All the LCD'S performs the same functions (display characters numbers special characters ASCII characters etc).Their programming is also same and they all have same 14 pins (0-13) or 16 pins (0 to 15).

**PRODUCT DESCRIPTION:**



This is an LCD Display designed for E-blocks. It is a 16 character, 2-line alphanumeric LCD display connected to a single 9-way D-type connector. This allows the device to be connected to most E-Block I/O ports. The LCD display requires data in a serial format, which is detailed in the user guide below. The display also requires a 5V power supply. Please take care not to exceed 5V, as this will cause damage to the device. The 5V is best generated from the E-blocks Multi programmer or a 5V fixed regulated power supply.

**6x2 LCD**

* **FEATURES:**
* Input voltage**:** 5v
* E-blocks compatible
* Low cost
* Compatible with most I/O ports in the E-Block range
* Ease to develop programming code using Flow code icons
* **APPLICATIONS:**
* Monitoring

**7.2.4 BUZZER :**

**GENERAL DESCRIPTION :**

Buzzer is an integrated structure of electronic transducers, DC power supply, widely used in computers, printers, copiers, alarms, electronic toys, automotive electronic equipment, telephones, timers and other electronic products for sound devices. Active buzzer 5V Rated power can be directly connected to a continuous sound, this section dedicated sensor expansion module and the board in combination, can complete a simple circuit design, to "plug and play."

**PRODUCT DESCRIPTION:**

It generates consistent single tone sound just by applying D.C voltage. Using a suitably designed resonant system, this type can be used where large sound volumes are needed. At Future Electronics we stock many of the most common types categorized by Type, Sound Level, Frequency, Rated Voltage, Dimension and Packaging Type.



**Buzzer**

**FEATURES**

* Input supply: 5 VDC
* Current consumption: 9.0 mA max.
* Oscillating frequency: 3.0 ±0.5 KHz
* Sound Pressure Level: 85dB min.

**APPLICATIONS**

* Confirmation of user input (ex: mouse click or keystroke)
* Electronic metronomes
* Sporting events
* Judging Panels
* Annunciator panels

****

**7.3 ESP8266 IOT**

**Internet of Things**

The Internet of Things (IoT) is an important topic in technology industry, policy, and engineering circles and has become headline news in both the specialty press and the popular media. This technology is embodied in a wide spectrum of networked products, systems, and sensors, which take advantage of advancements in computing power, electronics miniaturization, and network interconnections to offer new capabilities not previously possible. An abundance of conferences, reports, and news articles discuss and debate the prospective impact of the “IoT revolution”—from new market opportunities and business models to concerns about security, privacy, and technical interoperability. The large-scale implementation of IoT devices promises to transform many aspects of the way we live. For consumers, new IoT products like Internet-enabled appliances, home automation components, and energy management devices are moving us toward a vision of the “smart home’, offering more security and energy efficiency. Other personal IoT devices like wearable fitness and health monitoring devices and network enabled medical devices are transforming the way healthcare services are delivered. This technology promises to be beneficial for people with disabilities and the elderly, enabling improved levels of independence and quality of life at a reasonable cost.1 IoT systems like networked vehicles, intelligent traffic systems, and sensors embedded in roads and bridges move us closer to the idea of “smart cities’’, which help minimize congestion and energy consumption. A number of companies and research organizations have offered a wide range of projections about the potential impact of IoT on the Internet and the economy during the next five to ten years. Cisco, for example, projects more than 24 billion Internet–connected objects by 2019; 2 Morgan Stanley, however, projects 75 billion networked devices by 2020.3 Looking out further and raising the stakes higher, Huawei forecasts 100 billion IoT connections by 2025.4 McKinsey Global Institute suggests that the financial impact of IoT on the global economy may be as much as $3.9 to $11.1 trillion (about $34,000 per person in the US) by 2025.5 While the variability in predictions makes any specific number questionable, collectively they paint a picture of significant growth and influence. Some observers see the IoT as a revolutionary fully–interconnected “smart” world of progress, efficiency, and opportunity, with the potential for adding billions in value to industry and the global economy.6 others warn that the IoT represents a darker world of surveillance, privacy and security violations, and consumer lock–in.

**NodeMCU ESP8266**

NodeMCU is an open source IoT stage. It incorporates firmware which keeps running on the ESP8266 Wi-Fi SoC from Expressive Systems, and equipment which depends on the ESP-12 module. The term Nedelcu typically refers to firmware, whereas the board is termed Devkit. NodeMCU Devkit 1.0 consists of associate ESP-12E on a board that facilitates its use. It additionally contains a transformer and a USB interface. The expression "NodeMCU" of course alludes to the firmware as opposed to the improvement units. The firmware utilizes the Lua scripting dialect

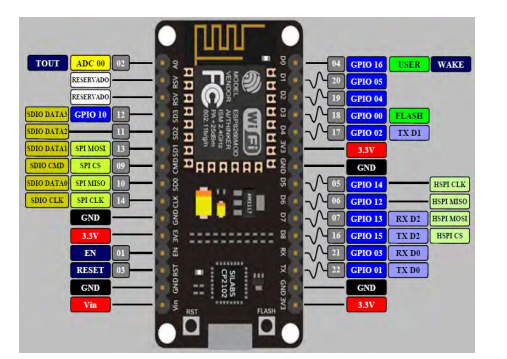
The NodeMCU (Node Micro Controller Unit) is an open-source software and hardware development environment that is built around a very inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266 is designed and manufactured by Express, contains all crucial elements of the modern computer: CPU, RAM, networking (wi-fi), and even a modern operating system and SDK. When purchased at bulk, the ESP8266 chip costs only $2 USD a piece. That makes it an excellent choice for this system design. The NodeMCU aims to simplify ESP8266 development. It has two key components.

1. An open source ESP8266 firmware that is built on top of the chip manufacturer's proprietary SDK. The firmware provides a simple programming environment based on eLua (embedded Lua), which is a very simple and fast scripting language with an established developer community. For newcomers, the Lua scripting language is easy to learn. And to add NodeMCU can be programmed with the Android IDE too.
2. ii. A development kit board that incorporates the ESP8266 chip on a standard circuit board. The board has a built-in USB port that is already wired up with the chip, a hardware reset button, Wi-Fi antenna, LED lights, and standard-sized GPIO (General Purpose Input Output) pins that can plug into a bread board. Figure 2 below shows the NodeMCU development board.



**Fig -: NodeMCU (ESP8266) Development Board**

There are various platforms for IoT system one of them is NodeMCU. It provides lower level control on devices which is known as firmware. These control runs on ESP8266 Wi-Fi SoC, which hardware is based on ESP-12 module.Esp8266 is a Wi-Fi based communication system’s microchip. It uses TCP/IP protocol for communicating with internet through router. It uses 802.11b/g/n slandered technology for Wi-Fi communication. It is a Tensilica L106 32-bit RISC instruction unite microprocessor with 32 KB instruction RAM, 32KB instruction Cache RAM, 80KB Data memory.ESP8266 (Pin) and Arduino Mega’s (Pin) common pins are GND TX (0)-3.3V, RX (0), 3.3V. For running the ESP8266 on Arduino platform we need to install ESP8266 package in Arduino IDE. As a board is using, we need to give the additional board manager.



**Fig: Esp8266 NodeMCU chip**

As shown in Fig it has 12 GPIO pins. One ADC pin. For this project 7 GPIO pins have been used. They are D0 to D6 which deliver digital outputs.

**Chapter 8**

**PROGRAM CODE**

#include<htc.h>

#include<pic.h>

#include<stdio.h>

#include<stdlib.h>

#define \_XTAL\_FREQ 10E6

\_\_CONFIG (WDTDIS & HS & PWRTDIS & BORDIS & LVPDIS);

#define rs RE2

#define rw RE1

#define en RE0

#define lcd PORTD

void adc\_conversion(void);

unsigned char temp\_output[4];

unsigned long value;

double temp;

unsigned char x,y,z;

void lcd\_string(unsigned char name[]);

void serial\_string(unsigned char name[]);

void gprs\_init(void);

void gprs\_init1(void);

void serial\_init(void);

void gsm\_init();

void putch(unsigned char byte);

unsigned char getche(void);

unsigned char getch(void);

void lcd\_init(void);

void lcd\_cmd(unsigned char comm);

void lcd\_data(unsigned char data);

void lcd\_clear(unsigned char row);

void delay(unsigned int time);

void main()

{

ADCON1=0x02;

//TRISB=0b01001110;

TRISB=0xFF;

PORTB=0x00;

TRISC=0x80;

PORTC=0x00;

TRISD=0x00;

PORTD=0x00;

TRISE=0x00;

PORTE=0x00;

RC0=0;

RC1=0;

RB0=0;

RB1=0;

RB2=0;

RB3=0;

lcd\_init();

lcd\_cmd(0X80);

lcd\_string("IOT GARBAGE");

lcd\_cmd(0XC2);

lcd\_string("MANAGEMENT”);

delay(100);

lcd\_cmd(0X01);

serial\_init();

gsm\_init();

PEIE=1;

GIE=1;

RCIE=1;

while(1)

{

adc\_conversion();

lcd\_cmd(0x80);

lcd\_string("MOI:");

lcd\_string(temp\_output);

if (RB0==1)

{

lcd\_cmd(0xC0);

lcd\_string("SMELL DETECTED");

serial\_string("AT+CMGS=");

putch('"');

serial\_string("+919876541230");

putch('"');

printf("\n\r");

printf("HIGH LEVEL SMELL DETECTED ");

delay(10);

printf("DUSTBIN NO: 1545\n\r");

delay(10);

printf("1st Street, Near VEMU college\n\r");

delay(10);

printf("\n\r");

putch(26);

delay(100);

lcd\_cmd(0x01);

}

//RB0=GAS SENSOR, RB1= MOISTURE SENSOR

if(RB0==1&&RB1==1&&((temp\_output[0]=='0'&&temp\_output[1]>='9'&&temp\_output[2]>='0')||temp\_output[0]<='1'&&temp\_output[1]<='2'&&temp\_output[2]<='9'))

{

lcd\_cmd(0X80);

lcd\_string("RECYCLE WASTE");

lcd\_cmd(0XC0);

lcd\_string("DETECTED");

RC0=1;

RC1=0;

delay(100);

delay(100);

lcd\_cmd(0X01);

RC0=0;

RC1=0;

}

if(RB0==1&&RB1==1&&((temp\_output[0]>='1'&&temp\_output[1]>='4'&&temp\_output[2]>='0')||(temp\_output[0]<='1'&&temp\_output[1]<='7'&&temp\_output[2]<='9')))

{

lcd\_cmd(0X80);

lcd\_string("NON-RECYCLE ");

lcd\_cmd(0XC0);

lcd\_string("WASTE DETECTED");

RC0=0;

RC1=1;

delay(100);

delay(100);

lcd\_cmd(0X01);

RC0=0;

RC1=0;

}

if(RB2==1)

{

lcd\_cmd(0X80);

lcd\_string("RECYCLE");

lcd\_cmd(0XC0);

lcd\_string("BIN FULL ");

serial\_string("AT+CMGS=");

putch('"');

serial\_string("+919876541230");

//printf("+918838426600");

putch('"');

printf("\n\r");

printf("RECYCLE BIN FULL ");

delay(10);

printf("DUSTBIN NO: 1545\n\r");

delay(10);

printf("1st Street, Near VEMU college\n\r");

delay(10);

printf("\n\r");

putch(26);

delay(100);

lcd\_cmd(0x01);

delay(100);

delay(100);

gprs\_init();

printf("AT+HTTPPARA=\"URL\",\"http://165.227.185.235:8080/IotWaste/DataUpdat?C1=RECYCLE BIN FULL&C3=HIGH\"\r");

delay(100);

delay(100);

putch(26);

delay(100);

delay(100);

gprs\_init1();

delay(100);

delay(100);

lcd\_cmd(0x01);

}

if(RB3==1)

{

lcd\_cmd(0X80);

lcd\_string("NON-RECYCLE");

lcd\_cmd(0XC0);

lcd\_string("BIN FULL ");

serial\_string("AT+CMGS=");

putch('"');

serial\_string("+919876541230");

putch('"');

printf("\n\r");

printf("NON-RECYCLE BIN FULL ");

delay(10);

printf("DUSTBIN NO: 1545\n\r");

delay(10);

printf("1st Street, Near VEMU college\n\r");

delay(10);

printf("\n\r");

putch(26);

delay(100);

lcd\_cmd(0x01);

delay(100);

delay(100);

gprs\_init();

printf("AT+HTTPPARA=\"URL\",\"http://165.227.185.235:8080/IotWaste/DataUpdat?C2=NON-RECYCLE BIN FULL&C3=LOW\"\r");

delay(100);

gprs\_init1();

delay(100);

delay(100);

lcd\_cmd(0x01);

}

if(RB3==1)

{

lcd\_cmd(0X80);

lcd\_string("NON-RECYCLE");

lcd\_cmd(0XC0);

lcd\_string("BIN FULL ");

serial\_string("AT+CMGS=");

putch('"');

serial\_string("+919876541230");

putch('"');

printf("\n\r");

printf("NON-RECYCLE BIN FULL ");

delay(10);

printf("DUSTBIN NO: 1545\n\r");

delay(10);

printf("1st Street, Near VEMU college\n\r");

delay(10);

printf("\n\r");

putch(26);

delay(100);

lcd\_cmd(0x01);

delay(100);

delay(100);

gprs\_init();

printf("AT+HTTPPARA=\"URL\",\"http://165.227.185.235:8080/IotWaste/DataUpdat?C2=NON-RECYCLE BIN FULL&C3=LOW\"\r");

delay(100);

delay(100);

putch(26);

delay(100);

delay(100);

gprs\_init1();

delay(100);

delay(100);

lcd\_cmd(0x01);

}

}

}

void adc\_conversion(void)

{

ADCON0=0B11000001;//RA0-INPUT

GODONE=1;

while(GODONE==1);

value=ADRESH;

value=value<<8;

value=value+ADRESL;

value=value\*(5.0/1024);

temp=(value-0.361)/0.079;

//temp=value\*0.765;

//value=(int)temp;

value=temp;

x=value/1000;

value=value%1000;

y=value/100;

value=value%100;

z=value/10;

temp\_output[0]=x+0x30;

temp\_output[1]=y+0x30;

temp\_output[2]=z+0x30;

}

void lcd\_cmd(unsigned char comm)

{

lcd=comm;

rs=0;

rw=0;

en=1;

delay(2);

en=0;

}//lcd\_comm

void lcd\_data(unsigned char data)

{

lcd=data;

rs=1;

rw=0;

en=1;

delay(2);

en=0;

}//lcd\_data

void lcd\_clear(unsigned char row)

{

if(row==1)

{

lcd\_cmd(0x80);

for(int i=0;i<16;i++)

lcd\_data(' ');

}//if

else

{

lcd\_cmd(0xc0);

for(int i=0;i<16;i++)

lcd\_data(' ');

}//else

}//lcd\_clear

void lcd\_string(unsigned char name[])

{

unsigned char i=0;

while(name[i]!='\0')

{

lcd\_data(name[i]);

i++;

}

}

void lcd\_init(void)

{

TRISD=0x00;

PORTD=0x00;

ADCON1=0B00000010;

TRISE=0b00000000;

PORTE=0X00;

lcd\_cmd(0x38);

delay(2);

lcd\_cmd(0x01);

delay(2);

lcd\_cmd(0x06);

delay(2);

lcd\_cmd(0x0C);

delay(2);

}//lcd\_initial()

void serial\_init(void)

{

TXSTA=0X20;//TXEN=1

RCSTA=0x90;//SPEN=1,CREN=1

SPBRG=15;

}

void putch(unsigned char byte)

{

TXREG=byte;

while(TXIF==0);

}

void serial\_string(unsigned char name[])

{

unsigned char i=0;

while(name[i]!='\0')

{

while(TXIF==0);

TXREG=name[i];

i++;

delay(2);

}

}

unsigned char getch(void)

{

/\* retrieve one byte \*/

while(RCIF==0); /\* WAIT TILL THAT REGISTER IS FULL\*/

return RCREG;

//return 3;

}

unsigned char getche(void)

{

unsigned char c;

putch(c = getch());

return c;

}

void delay(unsigned int time)

{

unsigned int i,j;

for(i=0;i<time;i++)

for(j=0;j<1275;j++);

}

void gsm\_init()

{

serial\_string("AT\n\r");

delay(100);

serial\_string("AT+CMGF=1\n\r");

delay(100);

}

void gprs\_init(void)

{

lcd\_cmd(0xC0);

lcd\_string(" ");

lcd\_cmd(0x80);

lcd\_string("GPRS INITI.......");

delay(100);

delay(100);

lcd\_cmd(0x80);

lcd\_string("CHECK GPRS ");

printf("AT\r");

delay(100);

delay(100);

lcd\_cmd(0x80);

lcd\_string("CHECK REGISTER.....");

printf("AT+CREG?\r");

delay(100);

delay(100);

lcd\_cmd(0x80);

lcd\_string("CHECK SAPBR ");

printf("AT+SAPBR=2,1\r");

delay(100);

delay(100);

delay(100);

lcd\_cmd(0x80);

lcd\_string("CHECK BEARER CON ");

printf("AT+SAPBR=1,1\r");

delay(100);

delay(100);

delay(100);

lcd\_cmd(0x80);

lcd\_string("HTTP INITI ");

printf("AT+HTTPINIT\r");

delay(100);

delay(100);

}

void gprs\_init1(void)

{

printf("AT+HTTPPARA=\"CID\",1\r");

delay(100);

delay(100);

lcd\_cmd(0x80);

lcd\_string("HTTP ACTION ");

printf("AT+HTTPACTION=1\r");

delay(100);

delay(100);

}

**CHAPTER 9**

**SYSTEM TESTING AND IMPLEMENTATION**

**Testing** :

The concept and functionality of the testing platform was tested on a set of test cases of an on-going project.

**Processor Board (Microcontroller):**

The first step was to build the processor board for the target processor. The board was built on a breadboard, as displayed in the analogue outputs; digital I/O and communication port were wired to separate pin header connectors. The processor board contained also a test pins for PWM output, PWM input and power connections. Addition, the board included RJ11 connector for programmer/debugger. This connector was solely used for programming the blank target MCU with the binary image.

**Test Case (Power Supply & Sensors):**

One of the tests performed concerned a hysteresis of operation modes depending on the current flow through the UUT. The current is an analogue signal from 0V to +5V. The Unit defaults to state A at the power on. The unit should enter state B when

48 the current drops to -0.5A or below and return back to state A when the current is 0.5A or greater. In addition to monitoring current, the unit measures two voltages. To avoid simulating abnormal behavior, the voltages are also scripted so that the voltage potential corresponds with the hysteresis trigger current. The example script can be found in appendix 8. The script is divided into three script groups: Configuration scripts, dynamic scripts and static or irrelevant scripts. This makes it easier to read the script and separate signals that are essential for the current test from the signals that are irrelevant to the result. The first group is a configuration group where the basic configuration is made. The reference voltage for DACs is set to +5V by Vref command. Next, the input and output ports are defined. The Processor board is connected to ports A, C and D. Without data direction definition the port defaults to all outputs, but every port is defined for clarity. The unit also monitors fan speed via pulse capture. This is emulated in the platform with PWM output, which frequency is set to 10Hz. The next group is dynamic script group. This group contains all signals that are mentioned in the test case and/or that change during the test run.

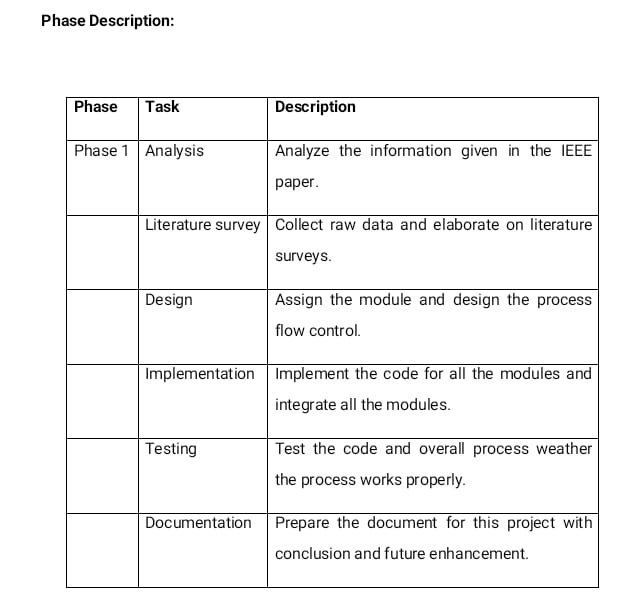
The test contains three main signal inputs: current and two voltages. The actual voltage and current values that the simulated signals represent in the HW environment are commented on in the script to make the script more readable. In addition to the analogue input signals the processor is reset at the beginning of the test to ensure the software is in the right mode for the test. The last group is for static or irrelevant scripts. These are signals that are not mentioned in the test script. These signals can be e.g., external control signals that are used to switch the software in a special mode or static analog signals, like alcohol, Eye blink and Vibration sensor readings.

**Implementation:**

System Implementation is the stage in the project where the theoretical design is turned into a working system. The most critical stage is achieving a successful system and giving confidence in the new system for the user that it will work efficiently and effectively.

The existing system was long time process. The proposed system was developed using Embedded C & Java. The existing system caused long time transmission process, but the system developed now has a very good user-friendly tool, which has a menu-based interface, graphical interface for the end user.

After coding and testing, the project is to be installed on the necessary system. The executable file is to be created and loaded in the system. Again, the code is tested in the installed system. Installing the developed code in system in the form of executable file is implementation.



**Alpha Test:**

In software development, your alpha test will be a test among yourselves (the teams) to confirm that your product works. Originally, the term alpha test meant the first phase of testing in a software development process. The first phase includes unit testing, component testing, and system testing. During this time, you will compress files, edit misspelled words and unclear directions, broken links, and sync audio and video. You will also test your product on the lowest common denominator machines to make sure download times are acceptable and preloaders work.

**Beta Test**

In software development, a beta test is the second phase of software testing in which a sampling of the intended audience tries the product out. Beta testing can be considered "pre-release testing." Beta test versions of software are now distributed to curriculum specialists and teachers to give the program a "real-world" test and partly to provide a preview of the next release.

**SOFTWARE DESCRIPTION:**

**EMBEDDED C**

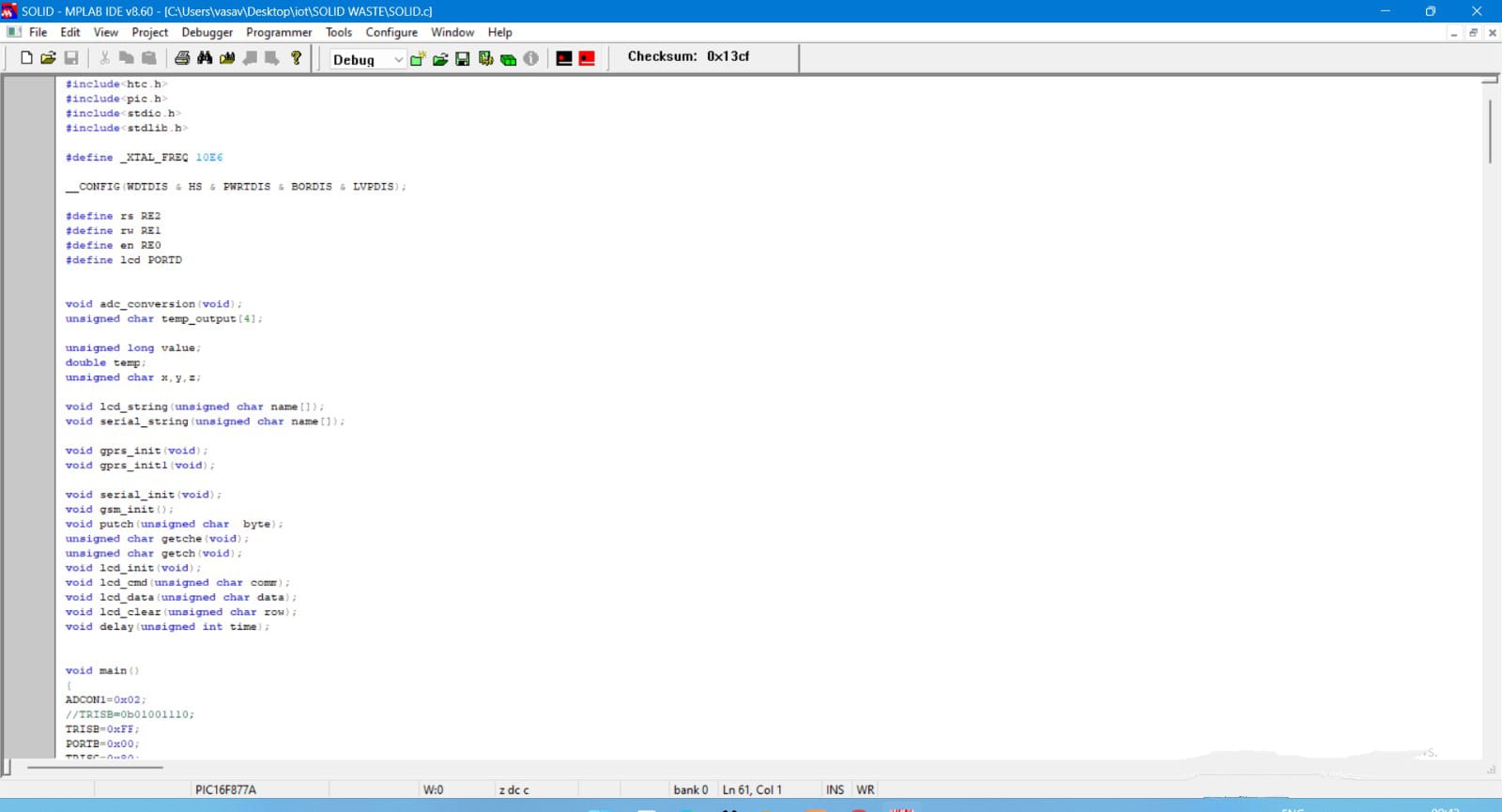
* As it is a high-level language it is shorter and easier to write. It is independent of the Processor. It is used more frequently than assembly language.
* C language has two important advantages, because of which it is much easier for programmers to write quality programs that are easier to read, revise, and port to a different system. Built-in structure. Checking and Abstraction

**MPLAB SOFTWARE:**

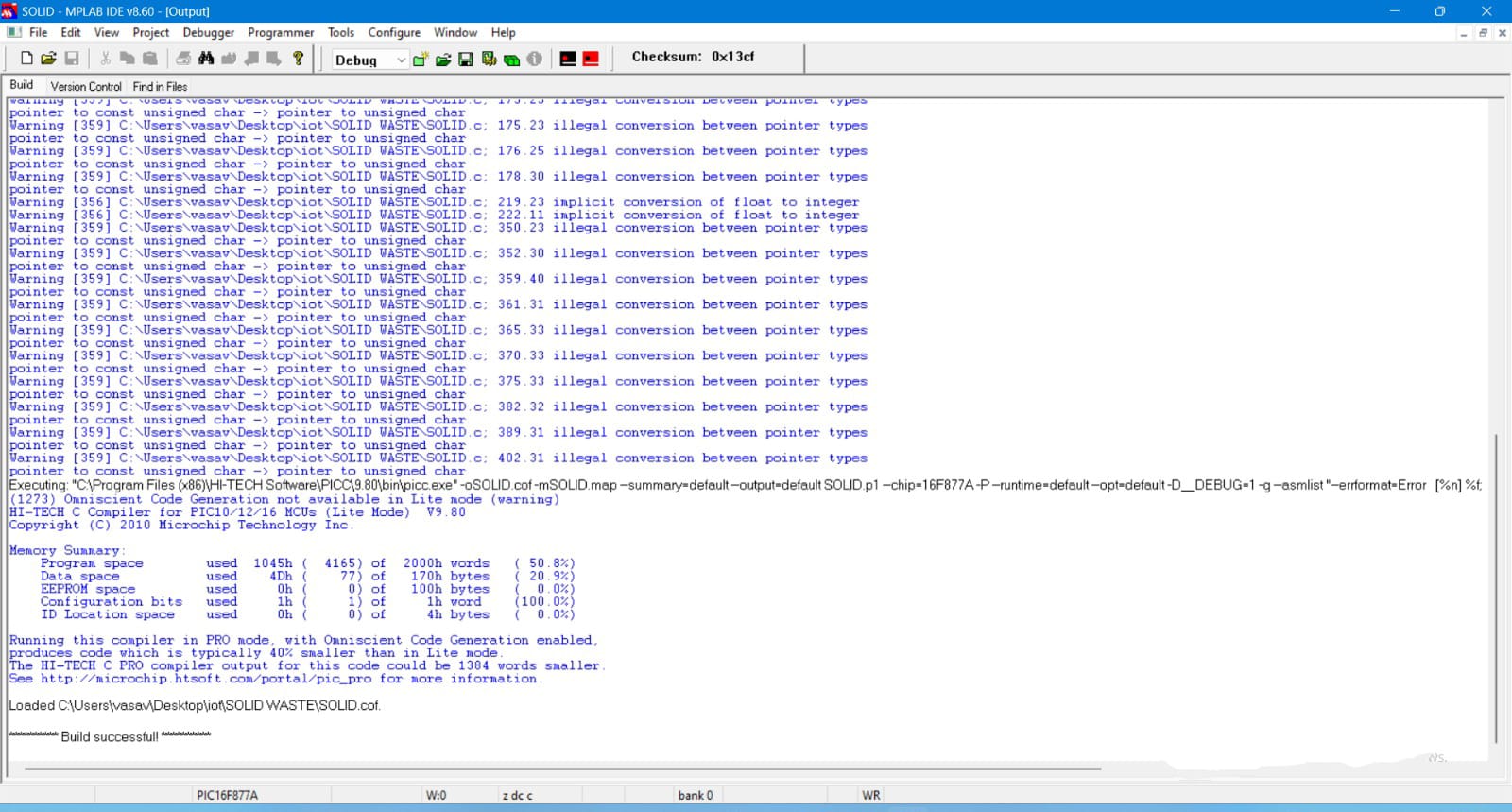
* **TEXT EDITOR:**
* Used for typing the embedded c programs.
* **SIMULATOR:**
* Used for debugging.
* **COMPILER**:
* used for converting high-level language into machine level language (hex code)
* **ANSI C Compiler:**
* Generates fast compact code for the 16F877A and its derivatives
* Advantages of Cover Assembler
* You do not need to know the microcontroller instruction set.
* Register allocation and addressing modes are handled by the compiler.
* Programming time is reduced.
* Code may be ported easily to other microcontrollers
* (Not 100% portable).
* C51 supports a number of C language extensions that have been added to support the 8051-microcontroller architecture e.g.

**Chapter 10**

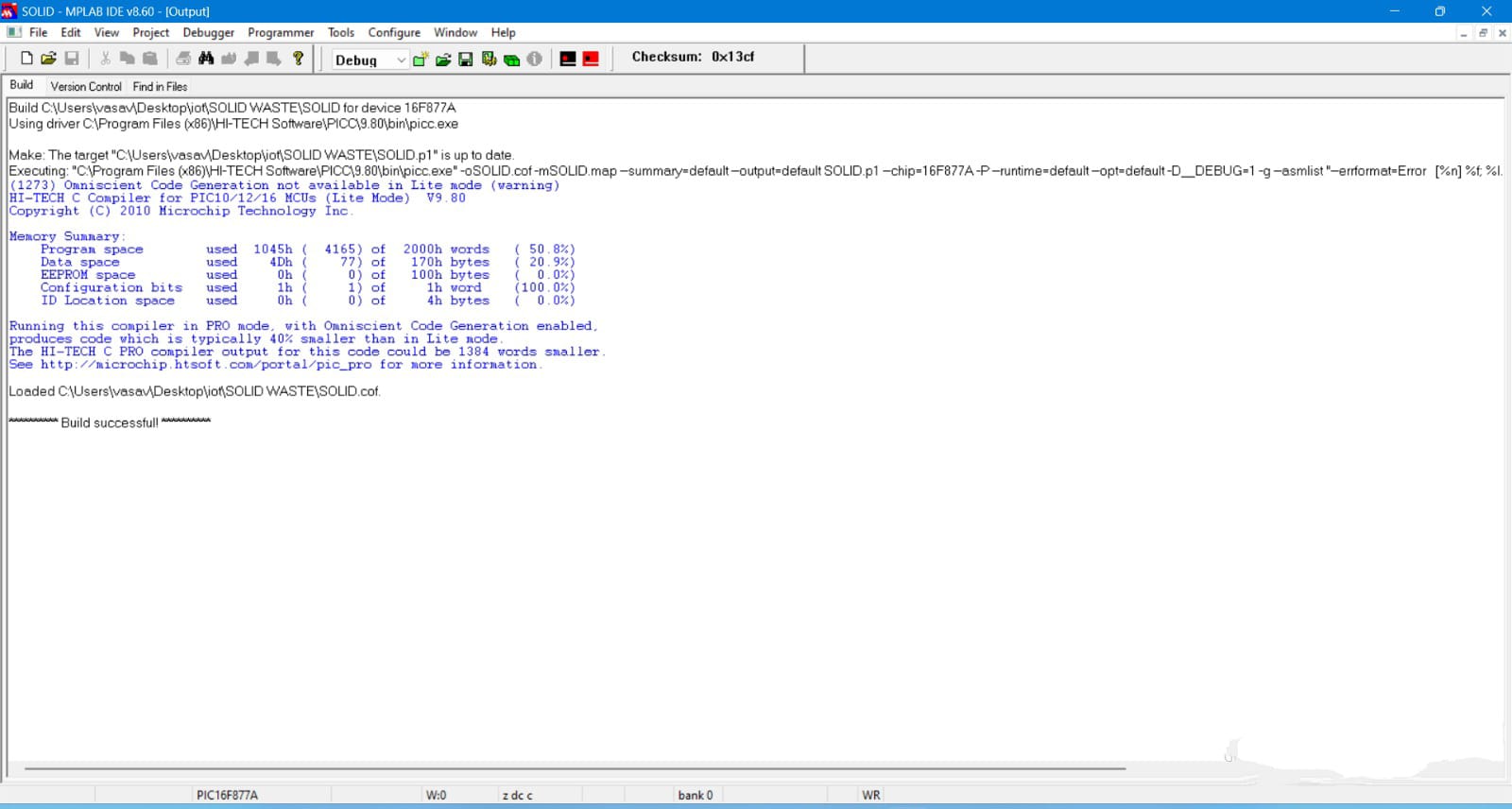
**OUTPUT SCREEN SHOTS**



**Fig 10.1**

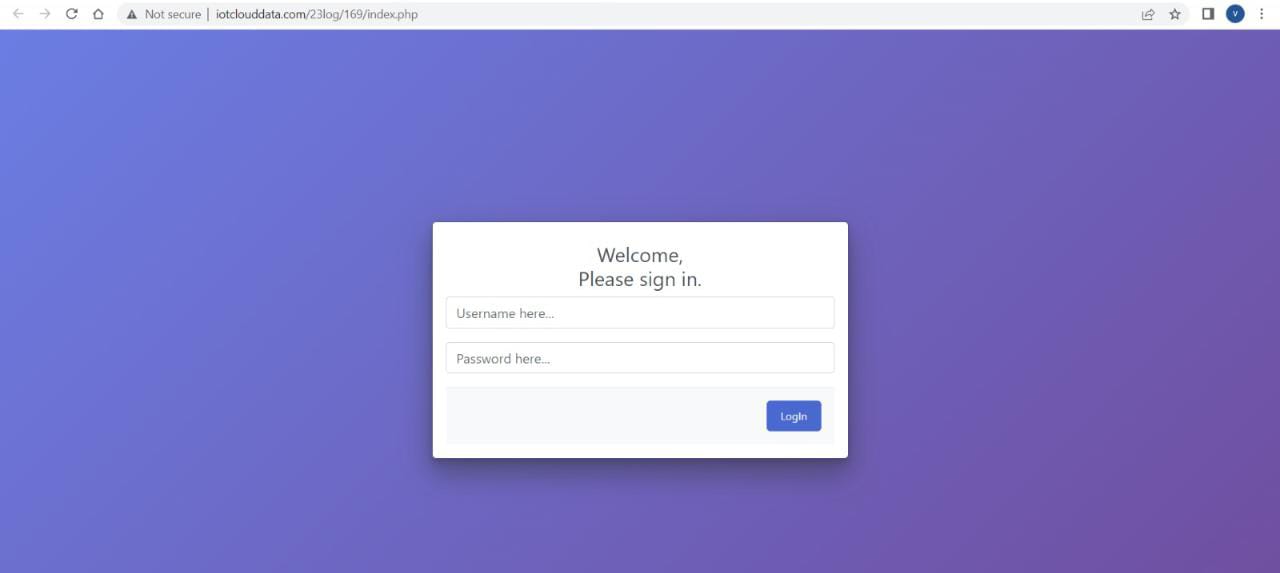


**Fig 10.2**

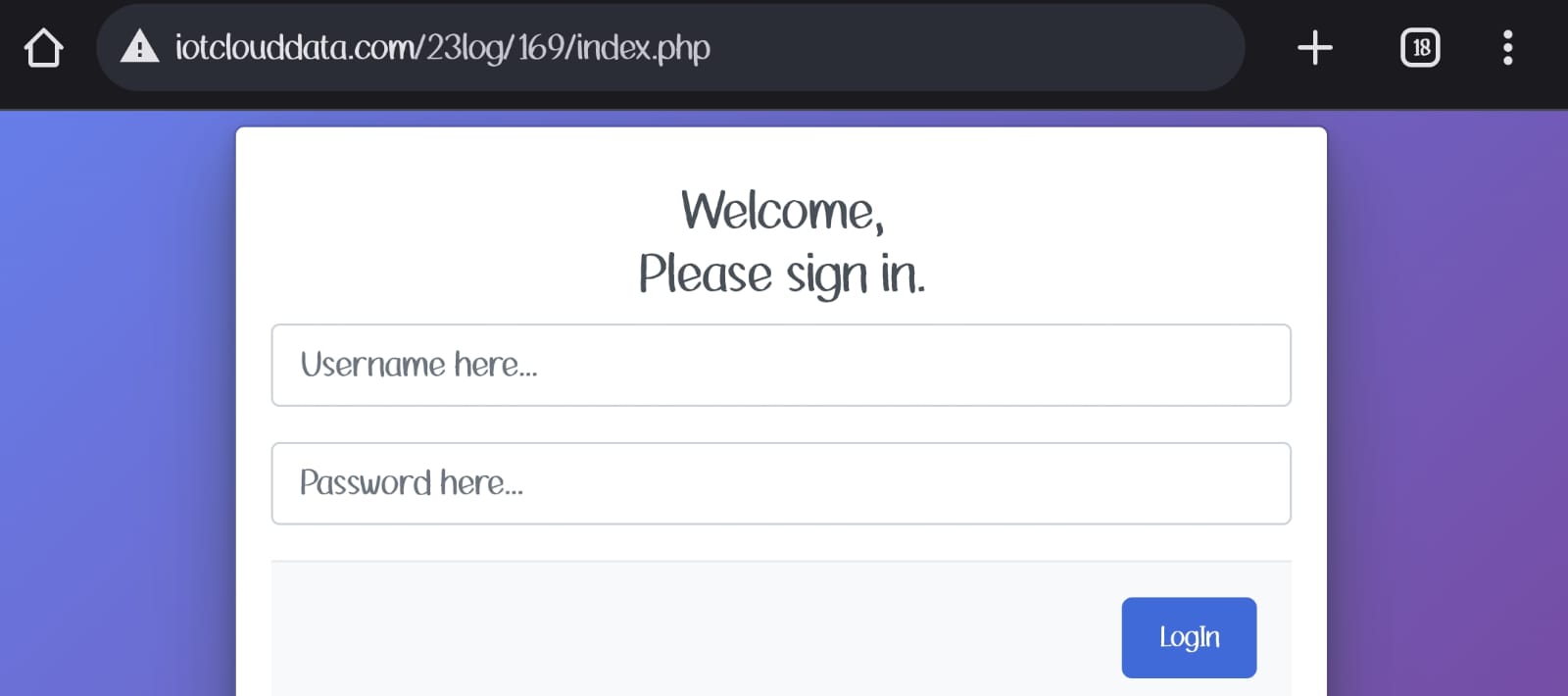


**Fig 10.3**

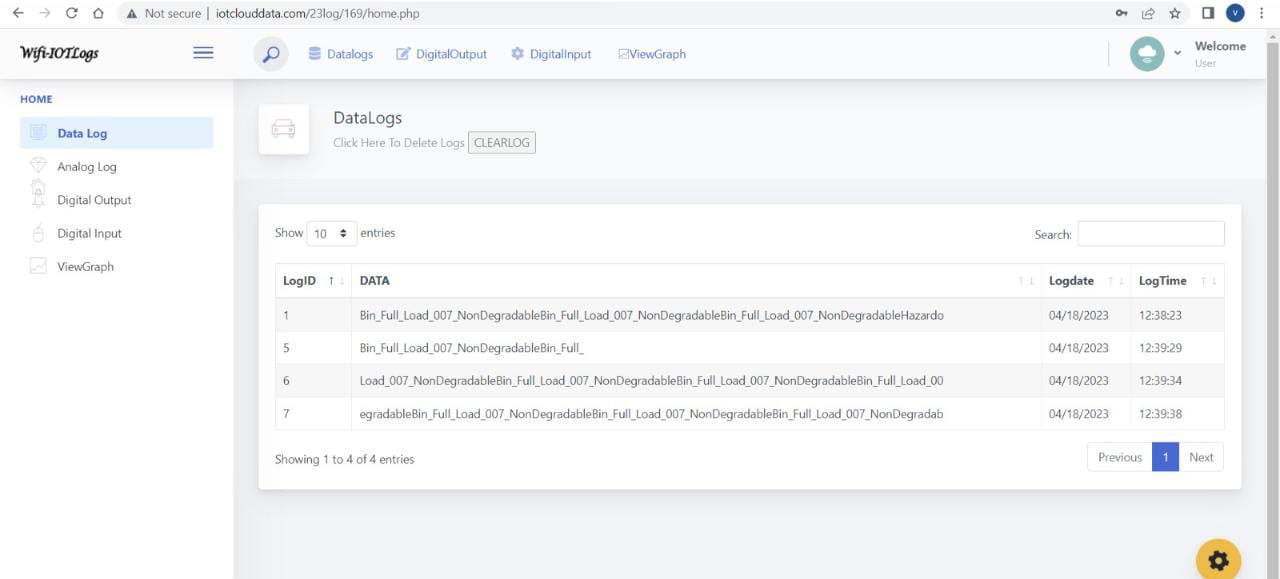
**Cloud web page:**



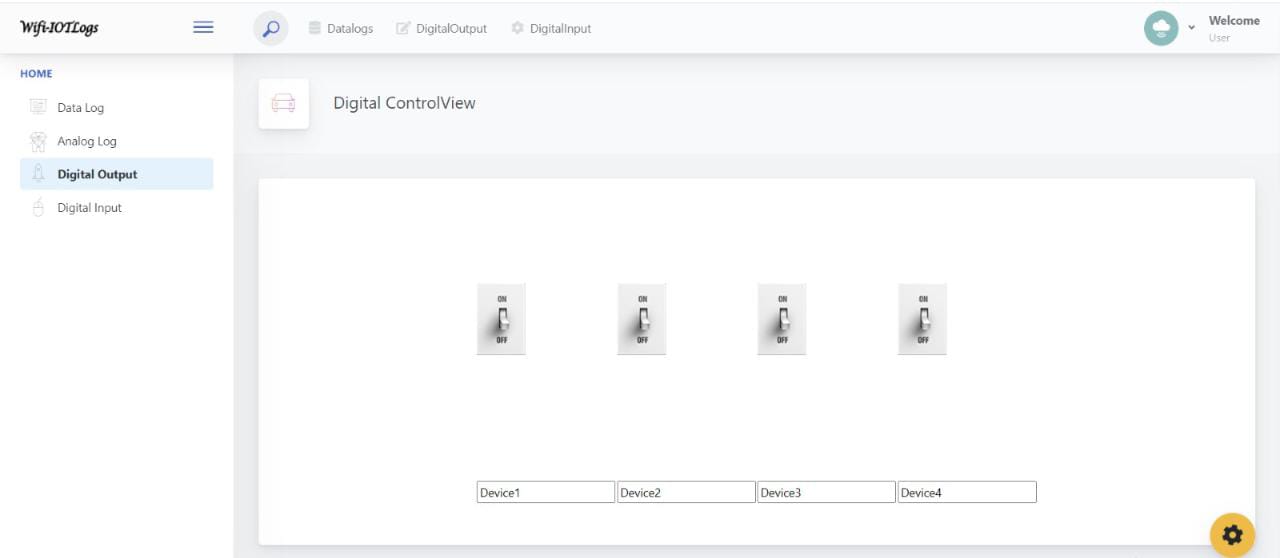
**Fig 10.4**



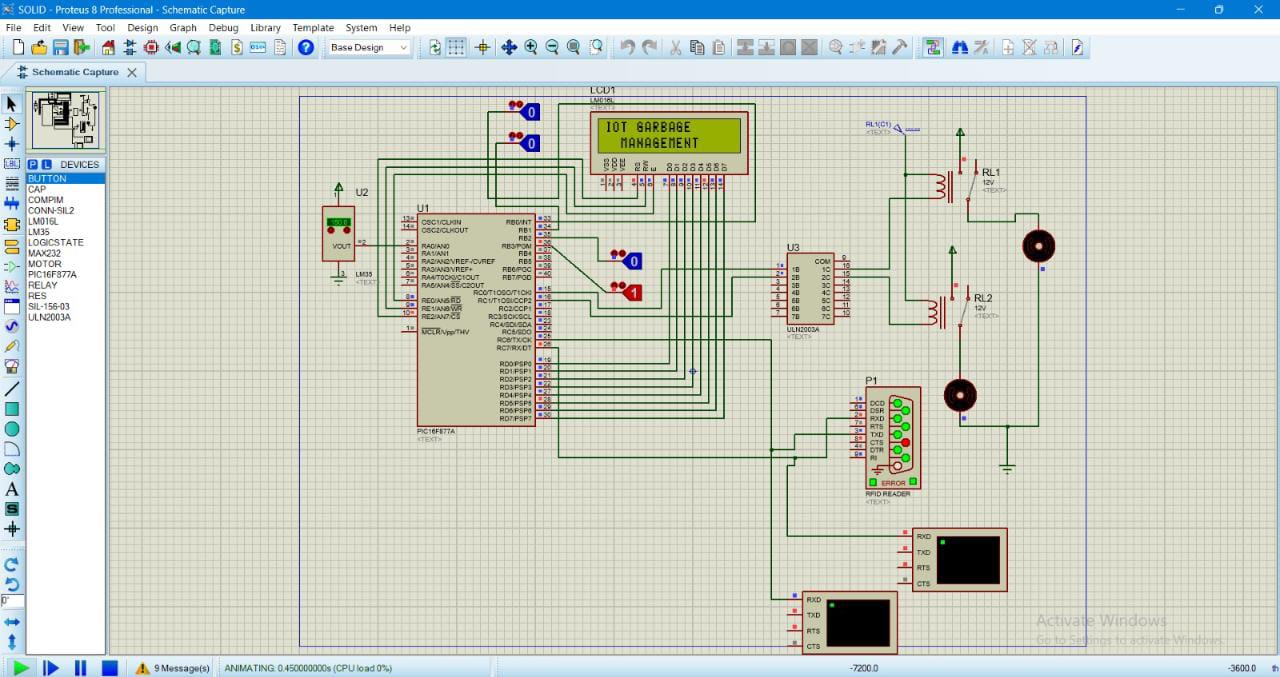
**Fig 10.5**



**Fig 10.6**

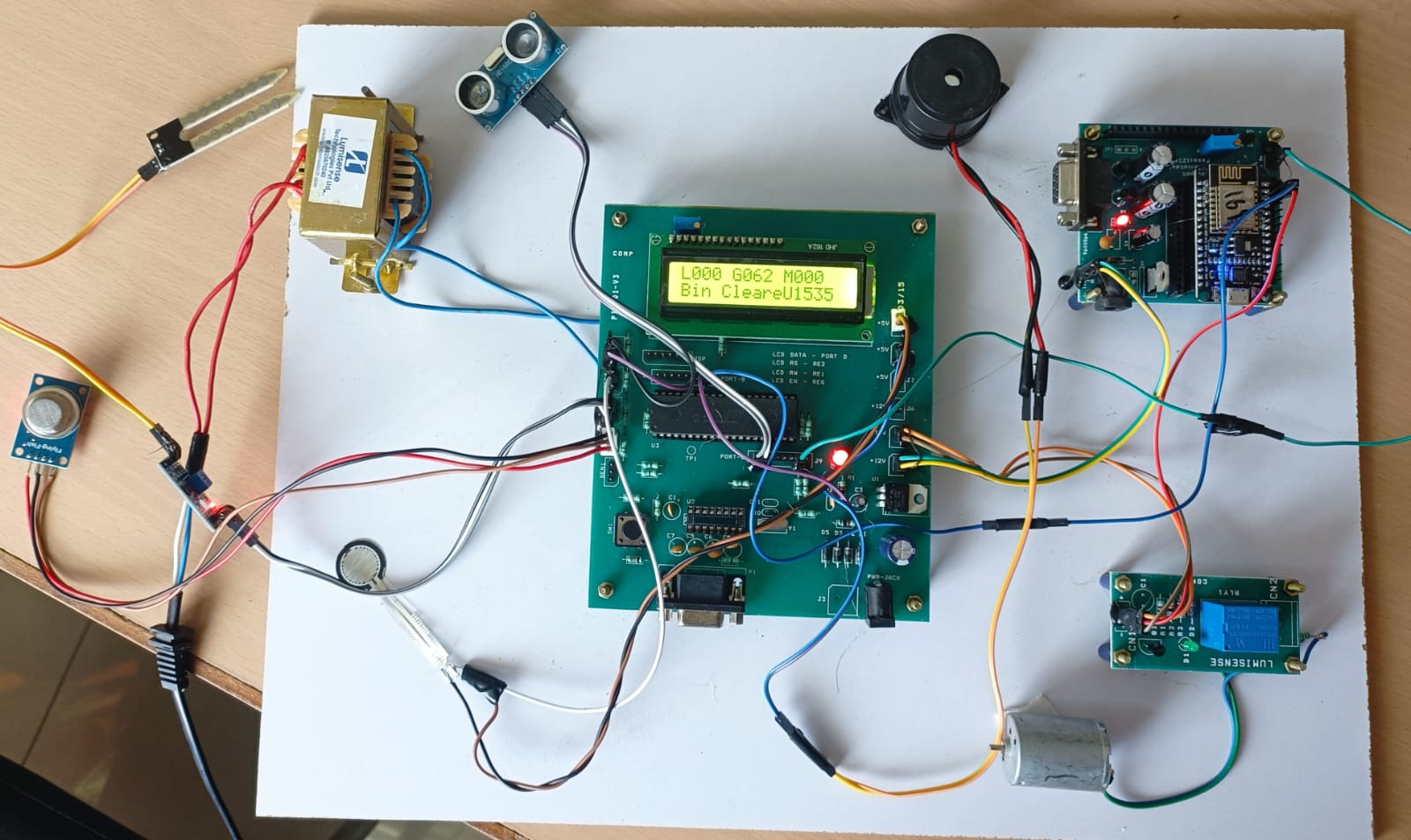


**Fig 10.7**



**Fig 10.8**

**IOT KIT:**



**fig 10.9**



**fig 10.10**

**CHAPTER 11**

**CONCLUSION**

The system provides us with real time information and status of garbage bins located in different areas. With the help of this real-time information, we can monitor the bins and once the bins are full the workers can collect the garbage and set them to empty again. This system is cost effective and can be accessed from anywhere. Traffic can be controlled as the workers collect the garbage only when the bin is full whereas in the traditional way workers collect the garbage daily whether the bin is filled or not. This system has a future scope where this system can be used with time stamp where real-time clock will be made available to the authority stating at what time Garbage bins were full and at what time the garbage is collected from the smart Garbage Bins.

**Chapter 12**

**FUTURE ENHANCEMENT**

The optimization of solution for the waste management is decided upon the results of the correct data obtained from the filling and the level of waste in the bins located at different places of the society. The system provides us with real time information and status of garbage bins located in different areas. With the help of this real-time information, we can monitor the bins and once the bins are full the workers can collect the garbage and set them to empty again. This system is cost effective and can be accessed from anywhere. Traffic can be controlled as the workers collect the garbage only when the bin is full whereas in the traditional way workers collect the garbage daily whether the bin is filled or not. This system has a future scope where this system can be used with time stamp where real-time clock will be made available to the authority stating at what time Garbage bins were full and at what time the garbage is collected from the smart Garbage Bins.

**CHAPTER 13**

**REFERENCES**

· Navghane S S, Killedar M S and Rohokale D V 2016 IoT Based Smart Garbage and waste collection, International Journal of Advanced Research in Electronics And Communication.

· Monika K A, Rao N, Prapulla S B and Shobha G 2016 Smart Dustbin-An EfficientGarbage Monitoring System International Journal of Engineering Science andComputing 6 7113-16.

· Medvedev A, Fedchenkov P, Zaslavsky A, Anagnostopoulos T and Khoruzhnikov S,2015 Waste management as an IoT-enabled service in smart cities In Conference on Smart Spaces Springer International Publishing 104-15.

· [www.buildofy.in/smart\_home\_designs](http://www.buildofy.in/smart_home_designs)

· <https://create.arduino.cc/projecthub/Technovation/smart-garbage-monitoring-systemusing-arduino-101-3b813c>

· <https://github.com/sourabhdeshmukh/Smart-Dustbin>

· <http://invent.module143.com/temperature-and-humidity-using-nodemcu-blynk/>

· <http://help.blynk.cc/getting-started-library-auth-token-code-examples/blynk-basics/whatis-virtual-pins>

· <http://help.blynk.cc/getting-started-library-auth-token-code-examples/blynk-basics/howto-display-any-sensor-data-in-blynk-app>

· http://help.blynk.cc/how-to-connect-different-hardware-with-blynk/esp8266/nodemcu