A Major Project Report

on

IOT BASED REMOTE OPERATED BOMB DETECTING AND DIFFUSION ROBOT

Submitted to JNTU Hyderabad

In Partial Fulfilment of the requirements for the Award of Degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Submitted By

ADYASISH BHANJA	**(208R1A1202)**
I. SAHITHI	**(208R1A1222)**
J. ANJALI	**(208R1A1224)**
MOHAMMED MUTALIB	**(208R1A1239)**

Under the guidance of

Mrs. Sushma Kariveda

Assistant professor, Department of IT

Department of Information Technology

CMR ENGINEERING COLLEGE
(UGC AUTONOMOUS)

(Accredited by NBA & NAAC, Approved by AICTE, New Delhi, Affiliated to JNTU Hyderabad)
Kandlakoya, Medchal Road, Medchal Dist. Hyderabad - 501 401.

A.Y. (2023-2024)

CMR ENGINEERING COLLEGE
(UGC AUTONOMOUS)
(Accredited by NBA & NAAC, Approved by AICTE, New Delhi, Affiliated to JNTU Hyderabad)
Kandlakoya, Medchal Road, Medchal Dist. Hyderabad - 501 401.

Department of Information Technology

CERTIFICATE

This is to certify that the project entitled “**IOT Based Remote Operated Bomb Detection And Diffusion Robot**” is a bonafide work carried out by

ADYASISH BHANJA	**(208R1A1202)**
I. SAHITHI	**(208R1A1222)**
J. ANJALI	**(208R1A1224)**
MOHAMMED MUTALIB	**(208R1A1239)**

in a partial fulfilment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY** in **INFORMATION TECHNOLOGY** from CMR Engineering College, affiliated to JNTU Hyderabad, under our guidance and supervision.

The results presented in this project have been verified and are found to be satisfactory. The results embodied in this project have not been submitted to any other university for the award of any degree or diploma.

Internal Guide	Head of the Department
Mrs. Kariveda Sushma	**Dr. MADHAVI PINGILI**
Assistant Professor	Professor & HOD
Department of IT	Department of IT
CMREC, Hyderabad.	CMREC, Hyderabad.

Project External Examiner

DECLARATION

This is to certify that the work reported in the present project entitled **"IOT Based Remote Operated Bomb Detection And Diffusion Robot"** is a record of bonafide work done by us in the Department of Information Technology, CMR Engineering College, UGC Autonomous, Hyderabad. The reports are based on the project work done entirely by us and not copied from any other source. We submit our project for further development by any interested students who share similar interests to improve the project in the future.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of its knowledge and belief.

ADYASISH BHANJA	**(208R1A1202)**
I. SAHITHI	**(208R1A1222)**
J. ANJALI	**(208R1A1224)**
MOHAMMED MUTALIB	**(208R1A1239)**

ACKNOWLEDGEMENT

We are extremely grateful to **Dr. A. Srinivasula Reddy,** Principal **and Dr. Madhavi Pingili**, HOD, Department of IT, CMR Engineering College for their constant support.

We are extremely thankful to **Mrs. Sushma Kariveda**, Assistant Professor, Internal Guide, Department of IT, for her constant guidance, encouragement and moral support through out the project.

We will be failing in duty if we do not acknowledge with grateful thanks to the authors of the references and other literatures referred in this Project.

Finally, We are very much thankful to our parents who guided us for every step.

ADYASISH BHANJA	**(208R1A1202)**
I. SAHITHI	**(208R1A1222)**
J. ANJALI	**(208R1A1224)**
MOHAMMED MUTALIB	**(208R1A1239)**

CONTENTS

ABSTRACT

The increasing threat of explosive devices in today's world necessitates advanced security measures. This project introduces a Bomb Diffusion Robot equipped with a robotic arm and a metal detector, utilizing Arduino technology. The robot's primary purpose is to safely locate, assess, and defuse explosive devices. The robotic arm enhances its capabilities for precise operations, while the metal detector aids in identifying hidden threats. The project offers a cost-effective, customizable, and user friendly solution for bomb disposal units, making it suitable for educational and practical applications in the field of security. The main objective behind this paper is to develop a robot to perform the act of surveillance Robots can be manually controlled using IOT app which is used for detection of metals and bomb detection. The purpose of this robot is to roam around environment to detection of metals and bomb detection and to send that obtained information to the user. In this project, one can control the robot with the help of mobile or laptop through Internet of Things (IoT). This robot will collect data from remote place and able to send those data to a remote IoT cloud database. This robot will be controlled via android mobile phone using WIFI communication. We can control the movement of the robot by sending instructions via IOT app from our android phone. In proposed system we are going to designed a low-cost Microcontroller Based Android controlled Robot. The robot will move forward, backward, left and right direction by following the instructions given from the mobile. This system can be helpful for various purpose of metal detection. The major goal of building this robot is to monitor human activity in combat zones or border regions to limit enemy infiltration. Military personnel faces a significant danger of death when approaching uncharted territory. The robot will be an appropriate machine for the defense sector, reducing human life loss and preventing criminal operations. There will assist all military personnel and armed forces in understanding the state of the land before entering it.

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Introduction and Objectives

Our project aims to provide a robotic vehicle equipped with a wireless camera having night vision capability for remote monitoring/spying purposes. The night vision camera allows for transmitting real time night vision video even in dark environments. Whatever is recorded by the camera can be viewed in PC for reference. This system is to be useful in war, terrorism and sensitive areas. It can also be used to operate in jungles and other environments humans cannot possibly enter during the night. The vehicle can be controlled remotely by an android device for easy operation. It uses android application commands to move in front, back and left right directions. The vehicle consists of receivers interfaced to an 8051 microcontroller. On receiving command from the receiver. The 8051 micro controller now operates the movement motor through a driver IC. The robotic vehicle can be easily operated from any android device. It provides a good user interface for handling the vehicle. The android device can operate the vehicle at a good IOT communication range. The IOT receiver at the vehicle is used to transmit control movement data from app to vehicle. The night vision camera mounted on robot allows for efficient spying even in darkest areas using infrared lighting.

Cloud robotics is an emerging field that is centered on the benefits of converged infrastructure and shared services of a cloud computing environment. In this paper, a system is designed with an autonomous robot to sense environmental data such as temperature, humidity, and air quality, along with GPS coordinates and store them on the cloud. The mobile robot is controlled using an Arduino microcontroller and communicates with the cloud via a ESP-CAM. A private cloud is set up using OpenStack that provides Infrastructure as a Service. The collected data are stored in a cloud server which could be viewed through a web browser and can be used to create awareness about the environmental changes of the location under study. A proof- of-concept prototype has been developed to illustrate the effectiveness of the proposed system. Cloud robotics is an emerging field that merges the concepts of cloud technologies and service robots. It is a disruptive technology based on the advantages of rapid fall in costs of servers, data centers, and broadband access, inexpensive cloud storage, and distributed computing. Internet is used to complement the capabilities of the robots by relieving them from on -board computation-intensive tasks and enable them to provide effective services on demand. Robotics is a technology that deals with the design, construction, operation, and application of robots, as well as computer systems for their control, sensory feedback, and information processing.

Mobile robots with on-board environmental sensors offer several advantages - low cost, ease of automation, wide operational range, and flexibility - in the monitoring of wide geographical areas. Indoor and outdoor environmental monitoring using mobile robots has been considered by several researchers, e.g., . Small mobile robots called Beo bots have been built to capture image that will be processed by a cloud setup using Microsoft windows Azure A standalone low cost device for transmitting data with touch screen display had been built using ESP-CAM and IOT. A robot to recognize voice had been developed using Google voice API and ESP-CAM . Raihan et. al. had developed an economical automated toll system that work by processing images using ESP-CAM. The system was developed as an alternative to the more costly system using RFID. In this paper, a robot is designed to move autonomously in the open space and to monitor the environmental conditions. The sensor data collected by the robot are stored in a cloud server that could be also be displayed in a webpage as well. Since very large amounts of spatio-temporal environmental data are collected in the process, a cloud server is used for economical storage, analysis, and retrieval of the data. The cloud environment is set up using OpenStack in Ubuntu Linux. The ESP-CAM microcontroller is used in the robot for communicating with the cloud server, while an Arduino microcontroller is used for control of the robot project is to deal with the security issues such as combating of the terrorists activities by tracking their locations and launch pads and reducing soldier's efforts and involvement in the mission. This can be achieved by the RF BASED spy robot which consists of a night vision wireless camera. The robot consists of night vision camera which is wireless and it can record real time videos and footages even in dark and these footages are displayed on our mobile screen which is connected through Wi-Fi via MI-app spy. This robot is capable carrying all kinds of military operations under all conditions without much involvement of the soldiers, thus saving the loss of lives and neutralizing any terrorists' activities. This can be also useful in gathering information about the arms and ammunitions of the rivals, destroying them from a sufficient safe distance. This device can easily be connected to the rocket launcher and tanks, acting as a guide machine. Not only in defense sector but also in disaster management can be fruitful in managing the situations like flood, earthquakes etc. This can be easily operated either through IOT or Wi-Fi. But in our project we are more concerned and focused on Wi-Fi, since it has better communication parameters and range.

1.2 Project Objective

The integration of a robotic arm within the Bomb Diffusion Robot significantly extends its capabilities, allowing it to perform intricate and precise operations essential for successful bomb diffusion. Additionally, the inclusion of a metal detector equips the robot with the capability to identify concealed threats, thus providing an added layer of security and ensuring thorough threat assessment.

The significance of this project extends beyond its technical aspects. It addresses the pressing need for enhancing safety and security in environments where explosive threats are a constant concern. The Bomb Diffusion Robot's Arduino-based technology not only empowers it with advanced functionalities but also ensures cost-effectiveness and customizability. This makes the project invaluable for both practical applications in the security field and for educational purposes, where it can serve as a valuable learning tool for aspiring engineers and robotics enthusiasts.

As this project unfolds, it will delve into the intricacies of the Bomb Diffusion Robot's design, construction, and operational aspects, emphasizing the role of Arduino technology in making it a versatile and accessible solution. The ensuing chapters will explore the hardware, software, control interfaces, and real-time feedback systems that contribute to the robot's effectiveness in bomb disposal scenarios. In addition to its practical applications, this project is poised to illustrate the potential of technology to enhance safety and security in our increasingly complex world.

1.3 Purpose of the Project

The primary objective behind creation of this robot keeping an alert watch especially in war field when something fishy is caught at border side due to some suspected act of enemy or any if any unnatural things is felt to happen. This is done so, in order to avoid loss of human life as the military personnel have great danger of losing their life if they are found to spying any suspected area. This robot vehicle will serve as an suitable material not only in aspect of providing border security but moreover can be utilized for different characteristic adversity and this machine for the Defense segment will reduce loss of human life too. It may guide all the military personnel and make them prepared for any misfortune if going to occur within their shelter region. Different Finder can be utilized that can be embedded on mechanical vehicle like metallic finder sensor is utilized to distinguish metallic objects This robot is valuable at places where one cannot reach like mystery spots or little areas. The foremost centre of this sort of model is to supply one extraordinary security degree. The great advancement that we come across in in designing this robot is the use of Wi-Fi.

1.4 Existing system with Disadvantages

In Existing system, the trained commandos are placed to dispose the bomb with the wearing blast suit. And also Commercial EOD (explosive ordnance disposal) robots are using in military field. The EOD have robotic arm use to dispose the bomb. The bomb is identified and disposed using the ROBO which is mainly help the lives of military officers.

Disadvantages Of Existing System

- EOD robot have only camera so it can't find hidden bombs.
- The use of EOB robot is only to dispose the bomb.

1.5 Proposed System With Features

The primary objective behind creation of this robot keeping a alert watch especially in war field when something fishy is caught at border side due to some suspected act of enemy or any if any unnatural things is felt to happen. This is done so, in order to avoid loss of human life as the military personnel have great danger of losing their life if they are found to spying any suspected area. So, to avoid it this robot will be useful to use in such cases. This robot vehicle will serve as an suitable material not only in aspect of providing border security but moreover can be utilized for different characteristic adversity and this machine for the defense segment will reduce loss of human life too. Fire finder is utilized to distinguish correct heading of fire source. This robot is valuable at places where one cannot reach like mystery spots or little areas. The foremost centre of this sort of model is to supply one extraordinary security degree. The great advancement that we come across in in designing this robot is the use of Wi-Fi.We can use here IOT module also instead of Wi-Fi but IOT have a short range of connection to make the robot work more efficiently as compared to Wi-Fi based system. Wi-Fi technique is useful in case if we are very far from the gadget also but our connection and Wi-Fi network is good then it works more significantly. The Node MCU ESP8266 used here acts as a link between the camera and the motor driver module fixed on the robot. It consists of motor driver module acts as a controller to contol the motion of the robot for working of the wheels of the robot fixed in it. The motor module used is named as L293D and a connector is provided between Node MCU module and motor driver module. That connector will be utilized for supplying external power supply. Four wheels are which operates on DC Motor is used for the motion of the robot. The camera used here can rotate whole 360 degree to record each and everything at every side wherever we wish to figure out the situation at the place where it is used for spying purpose.

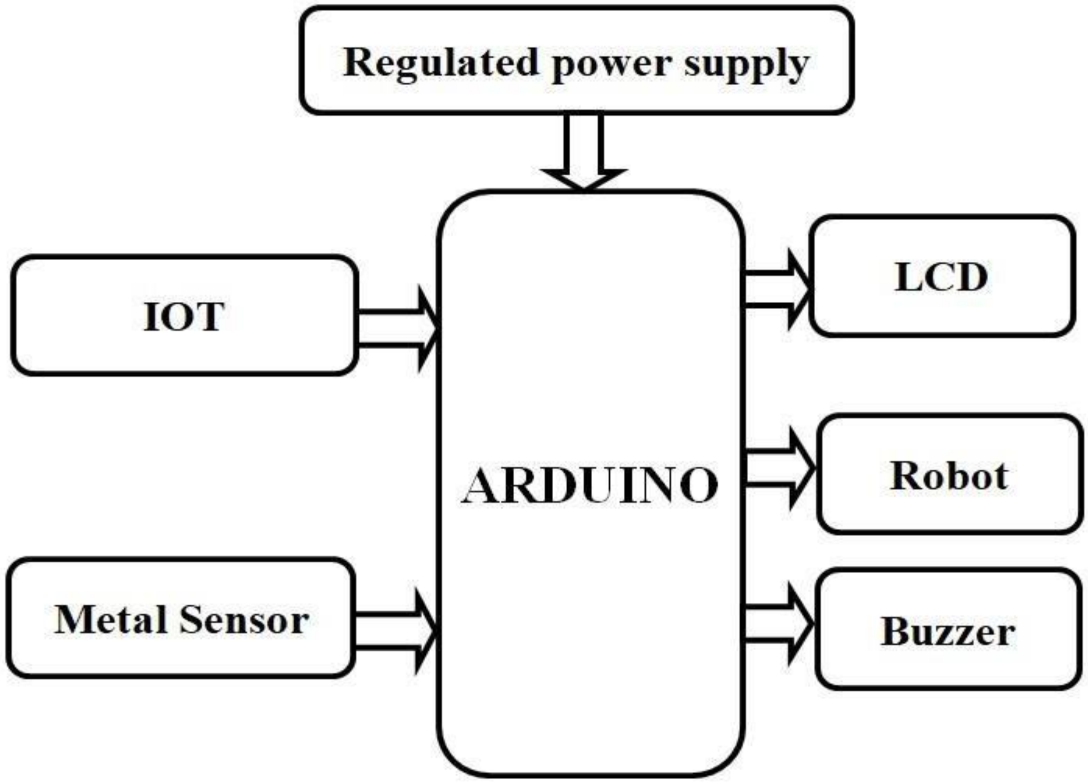


Fig: 1.5.1 Proposed System

Working:

We designed the Robot Unit to be equipped with a robotic arm, utilizing an Arduino and ESP32 microcontroller. This compact microcontroller serves as the brain of the robot, facilitating its navigation and the live, real-time video broadcast to a ground station via the Internet of Things (IoT) and a Wi-Fi network. This capability is invaluable in high-risk activities where human lives can be saved through remote operation. The primary objective of the robot is to patrol and gather real-time data, including video transmission, from a specified location. This data is then made available for human operators to make informed decisions and provide assistance. The robotic arm at the front of the robot is designed for pick-and-place operations and other tasks, making it suitable for various applications, including military and industrial uses.

The block diagram of the IoT-based firefighting robot includes a variety of sensors, an Arduino Uno, DC motors, and the ESP-32 WiFi module. The power supply is a crucial element in providing the necessary electrical power to the components. Typically, this power is delivered to voltage-consuming components, which include various electronics. In this configuration, a 12V DC power source is provided to the electronics. To achieve this, there is a requirement for voltage conversion, which involves a step-down transformer, rectifier, and filtering circuit to ensure a smooth and steady supply of 12V DC power.

Commands and instructions are communicated to the electronic components from a remote location. Conversely, the electronic components transfer stored data to the wireless module. A microcontroller, in this case, the ATMEGA 328 within an Arduino UNO board, processes the IoT web commands.

After validating the received commands, it instructs the robot or device to carry out specific tasks.

This block diagram represents the components of metal detector. In this configuration, the focus shifts from the camera to the metal detector, which is crucial for identifying concealed threats. The components include an ESP-32 for communication, an LCD monitor, a regulated power supply for providing the necessary electrical power, the robot with its wheels for mobility, and a Wi-Fi module for connectivity. The regulated power supply, typically using a 7805 voltage regulator, converts 230V AC to a stable 12V DC supply, with the capability to further convert this to 5 volts if needed.

The metal detector is integrated into the system to enhance its capabilities in identifying hidden metal objects, a critical function in security and safety applications. The robot's mobility and control are achieved through the DC motors connected to its wheels, allowing it to move in various directions with precision. This setup is well-suited for security and safety applications where the detection of concealed metal threats is of utmost importance.

2. LITERATURE SURVEY

After going through various articles and research papers we concluded that some of the papers were beneficial for designing our project and make it a successful one. In Military 2020 Spying Robot by Sarmad Hameed, Muhammad Hamza Khan, Naqi Jafri, the massive tasks is dangerous in war field. In border region it gets difficult for the humans working in the battle field to protect themselves from harm. Both protecting themselves and keeping keen observation on enemy becomes a little bit difficult task so in that situation robot is better option. Consequently robot replaces the trooper. In Spying Robot With Night Vision Camera by Aaruni Jha, Apoorva Singh, Ravinder Turna -The robot sends the flag to the RF collector mounted on the robot through RF transmitter at the base station. Due to this robot records real time footages and videos and can deliver those at our phone screen even in dark also as LED lights are used which even the enemy in the at border region or in suspected area can't even recognize that something is getting recorded. And the work done by PriyankaYadav, Swati Gawhale-She concluded that during the period of battle against enemy this robot can be used to collect all the necessary data that may weaken the opponent's plan if in case plotting something dangerous to attack them. In this way the military men would get prepared themselves for anything that the opponent is plotting against them and retort to their action in a better way that the enemy could not even think of at correct time. Robot navigation problems can be generally classified as global or local, depending upon the environment surrounding the robot. In global navigation, the environment surrounding the robot is known and a path which avoids the obstacles is selected. In one example of the global navigation techniques, graphical maps which contain information about the obstacles are used to determine a desirable path. In local navigation, the environment surrounding the robot is unknown, or only partially known, and sensors have to be used to detect the obstacles and a collision avoidance system must be incorporated into the robot to avoid the obstacles. The artificial potential field approach is one of the well-known techniques which has been developed for this purpose. Krogh, for example, used a generalized potential field approach to obstacle avoidance. Kilm and Khosla used instead harmonic potential functions for obstacle avoidance. On the other hand, Krogh and Fang used the dynamic generation of sub goals using local feedback information.

Title: "CGI Script And MJPG Video Streamer Based Robot".

Author: Muhammad Hamza, Subhan Bin Khalid.

This paper proposes a system that uses various sensors, including PIR sensors, vibration sensors, and temperature sensors, to detect suspicious activity around the ATM. The system also uses facial recognition to identify authorized users. If suspicious activity is detected, the system will sound an alarm and send an alert to the authorities.

Title: 'Low-Cost Land Wheeled Autonomous Robot For In-Door Surveillance' Author: Juan G. Parada-Salado, Luis E. Ortega-García.

This paper discusses the use of Zigbee technology for ATM security. Zigbee is a low-power, low-cost wireless communication protocol that can be used to create a network of sensors around the ATM. These sensors can detect things like tampering, forced entry, and unauthorized access.

Title: "Automated Hybrid Surveillance Robot".

Author: S M Ashish, Madhurya Manjunath, Ravindra L

This paper proposes an IoT-based ATM security system that uses edge computing and blockchain technology. Edge computing allows for real-time data processing at the edge of the network, while blockchain provides a secure and tamper-proof way to store data.

Title: "Surveillance Robot For ESP-CAM IOT"

Author: T. Saravankumar, D. Keerthana

This paper explores an IoT-enabled ATM security system with features like biometric access control, real-time monitoring, and emergency response protocols. An IoT-enabled ATM security system with these features has the potential to significantly improve the safety and security of cash withdrawals. However, it's important to carefully consider the potential challenges and implement safeguards to address privacy concerns and ensure system reliability.

Title: "IOT Based Remote Operated Robot for Bomb Detection and Diffusion".

Author: N. V. D. P. Murthy et al.

This paper proposes an IoT-based WSN system for ATM security that utilizes vibration sensors, temperature sensors, and cameras to detect suspicious activity and trigger alarms. An IoT-based WSN system for ATM security using vibration, temperature, and camera sensors holds significant promise. By addressing the potential considerations and ensuring proper implementation, this approach has the potential to significantly enhance ATM security and user safety.

Title: "An Intelligent Robot for Security System using WSN and IoT". Author: M. Rajesh et al.

This research explores a WSN-IoT system for security with real-time monitoring, intrusion detection, and emergency alert features. Combining these technologies holds significant potential for enhancing safety and deterring crime. Your focus on real-time monitoring, intrusion detection, and emergency alerts are crucial aspects of any effective security system.

Title: "WSN and IoT Based Robot Security System with Intelligent Video Surveillance". Author: S. Chauhan et al.

This paper proposes a WSN and IoT-based security system with intelligent video surveillance for real- time monitoring and anomaly detection in ATM environments. An automatic surveillance and control system for ATM is proposed which can monitor the ATM system for any attack either on ATM machine or on user.

Title: "Machine Learning-driven Fog Computing for Secure and Efficient Robot Crime Prevention with WSN and IoT".

Author: P. Sharma et al.

This paper explores machine learning algorithms deployed on fog computing platforms for real-time threat detection and analysis in WSN and IoT-based ATM security systems.

Title: "IoT-enabled Intelligent Robot For Security System using WSN". Author: S. Kumari et al.

This paper proposes an IoT-based WSN system for ATMs using vibration, temperature, and camera sensors to detect suspicious activity and alert authorities. Utilizing vibration, temperature, and camera sensors within an IoT-based WSN system holds great promise for enhancing ATM security.

Title: "Bomb Detection and Tamper Detection System for Security with WSN and IoT". Author: M. Khan et al.

This paper proposes a WSN system with multimodal biometric authentication and tamper detection for enhanced security. Combines biometric security with environmental sensors for multi-layered protection. Multimodal biometric authentication with environmental sensor-based tamper detection offers a promising multi-layered protection system.

Title: "WSN-based Robot for Security System with Intelligent Video Surveillance". Author: A. Joshi et al.

This paper proposes a WSN system with intelligent video analytics to detect suspicious behaviour and trigger alarms. Leverages AI for real-time video analysis, improving accuracy and reducing false positives.

Title: "IoT-based Robot for Security System for Real-time Monitoring and Threat Detection". Author: A. Kumar et al.

Proposes an IoT system with vibration, temperature, and camera sensors for real-time monitoring and threat detection, triggering alarms and sending alerts. Utilizing a combination of IoT sensors for real-time monitoring and threat detection in Army environments has the potential to significantly enhance security and deter criminal activity.

3. EMBEDDED SYSTEMS

3.1 Embedded Systems:

An embedded system is a computer system designed to perform one or a few dedicated functions often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts. By contrast, a general-purpose computer, such as a personal computer (PC), is designed to be flexible and to meet a wide range of end-user needs. Embedded systems control many devices in common use today. Embedded systems are controlled by one or more main processing cores that are typically either microcontrollers or digital signal processors (DSP). The key characteristic, however, is being dedicated to handle a particular task, which may require very powerful processors. For example, air traffic control systems may usefully be viewed as embedded, even though they involve mainframe computers and dedicated regional and national networks between airports and radar sites. (Each radar probably includes one or more embedded systems of its own.) Since the embedded system is dedicated to specific tasks, design engineers can optimize it to reduce the size and cost of the product and increase the reliability and performance. Some embedded systems are mass- produced, benefiting from economies of scale. Physically embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations like traffic lights, factory controllers, or the systems controlling nuclear power plants.

Complexity varies from low, with a single microcontroller chip, to very high with multiple units, peripherals and networks mounted inside a large chassis or enclosure. In general, "embedded system" is not a strictly definable term, as most systems have some element of extensibility or programmability. For example, handheld computers share some elements with embedded systems such as the operating systems and microprocessors which power them, but they allow different applications to be loaded and peripherals to be connected. The hardware for the system is usually chosen to make the device as cheap as possible. Spending an extra dollar a unit in order to make things easier to program can cost millions. Hiring a programmer for an extra month is cheap in comparison.

This means the programmer must make do with slow processors and low memory, while at the same time battling a need for efficiency not seen in most PC applications. Below is a list of issues specific to the embedded field. Moreover, even systems which don't expose programmability as a primary feature generally need to support software updates. On a continuum from "general purpose" to "embedded", large application systems will have subcomponents at most points even if the system as a whole is "designed to perform one or a few dedicated functions", and is thus appropriate to call "embedded".

A modern example of embedded system is shown in fig: 3.1.

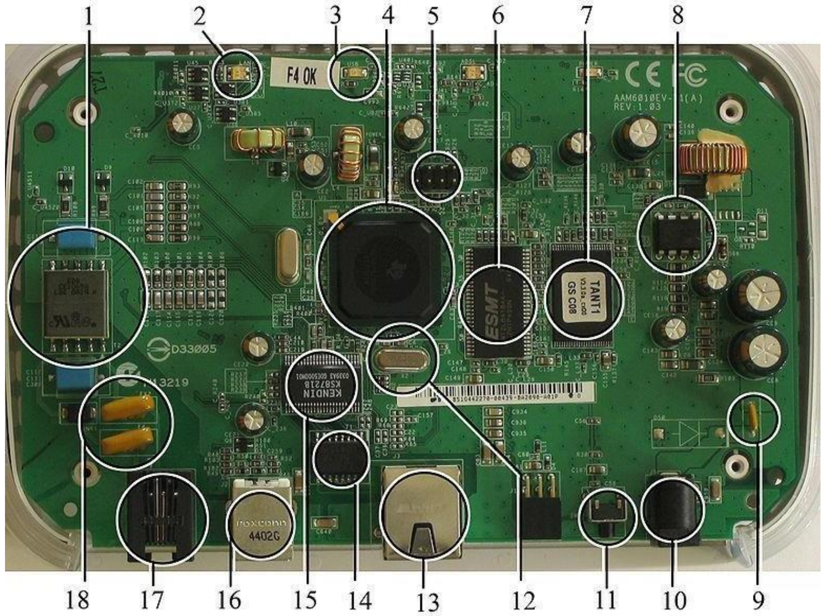


Fig 3.1: A modern example of embedded system

Labeled parts include microprocessor (4), RAM (6), flash memory (7).Embedded systems programming is not like normal PC programming. In many ways, programming for an embedded system is like programming PC 15 years ago. The hardware for the system is usually chosen to make the device as cheap as possible. Spending an extra dollar a unit in order to make things easier to program can cost millions. Hiring a programmer for an extra month is cheap in comparison. This means the programmer must make do with slow processors and low memory, while at the same time battling a need for efficiency not seen in most PC applications. Below is a list of issues specific to the embedded field.

3.1.1 History:

In the earliest years of computers in the 1930–40s, computers were sometimes dedicated to a single task, but were far too large and expensive for most kinds of tasks performed by embedded computers of today. Over time however, the concept of programmable controllers evolved from traditional electromechanical sequencers, via solid state devices, to the use of computer technology. One of the first recognizably modern embedded systems was the Apollo Guidance Computer, developed by Charles Stark Draper at the MIT Instrumentation Laboratory. At the project's inception, the Apollo guidance computer was considered the riskiest item in the Apollo project as it employed the then newly developed monolithic integrated circuits to reduce the size and weight. An early mass-produced embedded system was the Autonetics D-17 guidance computer for the Minuteman missile, released in 1961. It was built from transistor logic and had a hard disk for main memory. When the Minuteman II went into production in 1966, the D-17 was replaced with a new computer that was the first high-volume use of integrated circuits.

3.1.2 Tools:

Embedded development makes up a small fraction of total programming. There's also a large number of embedded architectures, unlike the PC world where 1 instruction set rules, and the Unix world where there's only 3 or 4 major ones. This means that the tools are more expensive. It also means that they're lowering featured, and less developed. On a major embedded project, at some point you will almost always find a compiler bug of some sort.

Debugging tools are another issue. Since you can't always run general programs on your embedded processor, you can't always run a debugger on it. This makes fixing your program difficult. Special hardware such as JTAG ports can overcome this issue in part. However, if you stop on a breakpoint when your system is controlling real world hardware (such as a motor), permanent equipment damage can occur. As a result, people doing embedded programming quickly become masters at using serial IO channels and error message style debugging.

3.1.3 Resources:

To save costs, embedded systems frequently have the cheapest processors that can do the job. This means your programs need to be written as efficiently as possible. When dealing with large data sets, issues like memory cache misses that never matter in PC programming can hurt you. Luckily, this won't happen too often- use reasonably efficient algorithms to start, and optimize only when necessary. Of course, normal profilers won't work well, due to the same reason debuggers don't work well. Memory is also an issue. For the same cost savings reasons, embedded systems usually have the least memory they can get away with. That means their algorithms must be memory efficient (unlike in PC programs, you will frequently sacrifice processor time for memory, rather than the reverse). It also means you can't afford to leak memory. Embedded applications generally use deterministic memory techniques and avoid the default "new" and "malloc" functions, so that leaks can be found and eliminated more easily.

Other resources programmers expect may not even exist. For example, most embedded processors do not have hardware FPUs (Floating-Point Processing Unit). These resources either need to be emulated in software, or avoided altogether.

3.1.4 Real Time Issues:

Embedded systems frequently control hardware, and must be able to respond to them in real time. Failure to do so could cause inaccuracy in measurements, or even damage hardware such as motors. This is made even more difficult by the lack of resources available. Almost all embedded systems need to be able to prioritize some tasks over others, and to be able to put off/skip low priority tasks such as UI in favor of high priority tasks like hardware control.

3.2 Need For Embedded Systems:

The uses of embedded systems are virtually limitless, because every day new products are introduced to the market that utilizes embedded computers in novel ways. In recent years, hardware such as microprocessors, microcontrollers, and FPGA chips have become much cheaper. So when implementing a new form of control, it's wiser to just buy the generic chip and write your own custom software for it. Producing a custom-made chip to handle a particular task or set of tasks costs far more time and money. Many embedded computers even come with extensive libraries, so that "writing your own software" becomes a very trivial task indeed. From an implementation viewpoint, there is a major difference between a computer and an embedded system. Embedded systems are often required to provide Real-Time response. The main elements that make embedded systems unique are its reliability and ease in debugging.

3.2.1 Debugging:

Embedded debugging may be performed at different levels, depending on the facilities available. From simplest to most sophisticate they can be roughly grouped into the following areas:

- Interactive resident debugging, using the simple shell provided by the embedded operating system (e.g. Forth and Basic)
- External debugging using logging or serial port output to trace operation using either a monitor in flash or using a debug server like the Remedy Debugger which even works for heterogeneous multi core systems.

- An in-circuit debugger (ICD), a hardware device that connects to the microprocessor via a JTAG or Nexus interface. This allows the operation of the microprocessor to be controlled externally, but is typically restricted to specific debugging capabilities in the processor.
- An in-circuit emulator replaces the microprocessor with a simulated equivalent, providing full control over all aspects of the microprocessor.
- A complete emulator provides a simulation of all aspects of the hardware, allowing all of it to be controlled and modified and allowing debugging on a normal PC.
- Unless restricted to external debugging, the programmer can typically load and run software through the tools, view the code running in the processor, and start or stop its operation. The view of the code may be as assembly code or source-code.

Because an embedded system is often composed of a wide variety of elements, the debugging strategy may vary. For instance, debugging a software(and microprocessor) centric embedded system is different from debugging an embedded system where most of the processing is performed by peripherals (DSP, FPGA, co-processor). An increasing number of embedded systems today use more than one single processor core.

A common problem with multi-core development is the proper synchronization of software execution. In such a case, the embedded system design may wish to check the data traffic on the busses between the processor cores, which requires very lowlevel debugging, at signal/bus level, with a logic analyzer, for instance.

3.2.2 Reliability:

Embedded systems often reside in machines that are expected to run continuously for years without errors and in some cases recover by themselves if an error occurs. Therefore the software is usually developed and tested more carefully than that for personal computers, and unreliable mechanical moving parts such as disk drives, switches or buttons are avoided.

Specific reliability issues may include:

- The system cannot safely be shut down for repair, or it is too inaccessible to repair. Examples include space systems, undersea cables, navigational beacons, bore-hole systems, and automobiles.
- The system must be kept running for safety reasons. "Limp modes" are less tolerable. Often backups are selected by an operator. Examples include aircraft navigation, reactor control systems, safety-critical chemical factory controls, train signals, engines on single-engine aircraft.

- The system will lose large amounts of money when shut down: Telephone switches, factory
- controls, bridge and elevator controls, funds transfer and market making, automated sales and service. A variety of techniques are used, sometimes in combination, to recover from errors—both software bugs such as memory leaks, and also soft errors in the hardware:
- Watchdog timer that resets the computer unless the software periodically notifies the watchdog
- Subsystems with redundant spares that can be switched over to
- software "limp modes" that provide partial function
- Designing with a Trusted Computing Base (TCB) architecture[6] ensures a highly secure & reliable system environment
- An Embedded Hypervisor is able to provide secure encapsulation for any subsystem component, so that a compromised software component cannot interfere with other subsystems, or privileged-level system software. This encapsulation keeps faults from propagating from one subsystem to another, improving reliability. This may also allow a subsystem to be automatically shut down and restarted on fault detection.
- Immunity Aware Programming

3.3 Explanation of Embedded Systems:

3.3.1 Software Architecture:

There are several different types of software architecture in common use.

- Simple Control Loop: In this design, the software simply has a loop. The loop calls subroutines, each of which manages a part of the hardware or software.
- Interrupt Controlled System: Some embedded systems are predominantly interrupting controlled. This means that tasks performed by the system are triggered by different kinds of events. An interrupt could be generated for example by a timer in a predefined frequency, or by a serial port controller receiving a byte. These kinds of systems are used if event handlers need low latency and the event handlers are short and simple. Usually these kinds of systems run a simple task in a main loop also, but this task is not very sensitive to unexpected delays. Sometimes the interrupt handler will add longer tasks to a queue structure. Later, after the interrupt handler has finished, these tasks are executed by the main loop.

This method brings the system close to a multitasking kernel with discrete processes.

- **Cooperative Multitasking:**

  A non-preemptive multitasking system is very similar to the simple control loop scheme, except that the loop is hidden in an API. The programmer defines a series of tasks, and each task gets its own environment to "run" in. When a task is idle, it calls an idle routine, usually called "pause", "wait", "yield", "nop" (stands for no operation), etc.The advantages and disadvantages are very similar to the control loop, except that adding new software is easier, by simply writing a new task, or adding to the queue-interpreter.

- **Primitive Multitasking:**

  In this type of system, a low-level piece of code switches between tasks or threads based on a timer (connected to an interrupt). This is the level at which the system is generally considered to have an "operating system" kernel. Depending on how much functionality is required, it introduces more or less of the complexities of managing multiple tasks running conceptually in parallel. As any code can potentially damage the data of another task (except in larger systems using an MMU) programs must be carefully designed and tested, and access to shared data must be controlled by some synchronization strategy, such as message queues, semaphores or a non-blocking synchronization scheme.

Because of these complexities, it is common for organizations to buy a real-time operating system, allowing the application programmers to concentrate on device functionality rather than operating system services, at least for large systems; smaller systems often cannot afford the overhead associated with a generic real time system, due to limitations regarding memory size, performance, and/or battery life.

- **Microkernels And Exokernels:**

  Microkernel is a logical step up from a real-time OS. In general, microkernels succeed when the task switching and intertask communication is fast, and fail when they are slow. Exokernels communicate efficiently by normal subroutine calls. The hardware and all the software in the system are available to, and extensible by application programmers. Based on performance, functionality, requirement the embedded systems are divided into three categories.

3.3.2 Stand Alone Embedded System:

These systems takes the input in the form of electrical signals from transducers or commands from human beings such as pressing of a button etc.., process them and produces desired output. This entire process of taking input, processing it and giving output is done in standalone mode. Such embedded systems comes under stand alone embedded systems.

Eg: microwave oven, air conditioner etc..

3.3.3 Real-time embedded systems:

Embedded systems which are used to perform a specific task or operation in a specific time period those systems are called as real-time embedded systems. There are two types of real-time embedded systems.

- **Hard Real-time embedded systems:**

These embedded systems follow an absolute dead line time period i.e.., if the tasking is not done in a particular time period then there is a cause of damage to the entire equipment.

Eg: consider a system in which we have to open a valve within 30 milliseconds. If this valve is not opened in 30 ms this may cause damage to the entire equipment. So in such cases we use embedded systems for doing automatic operations.

- **Soft Real Time embedded systems:**

These embedded systems follow a relative dead line time period i.e.., if the task is not done in a particular time that will not cause damage to the equipment.

Eg: Consider a TV remote control system , if the remote control takes a few milliseconds delay it will not cause damage either to the TV or to the remote control. These systems which will not cause damage when they are not operated at considerable time period those systems comes under soft real-time embedded systems.

3.3.4 Network communication embedded systems:

A wide range network interfacing communication is provided by using embedded systems.

Eg: Consider a web camera that is connected to the computer with internet can be used to spread communication like sending pictures, images, videos etc.., to another computer with internet connection throughout anywhere in the world.

Consider a web camera that is connected at the door lock.

Whenever a person comes near the door, it captures the image of a person and sends to the desktop of your computer which is connected to internet. This gives an alerting message with image on to the desktop of your computer, and then you can open the door lock just by clicking the mouse. Figure shows the network communications in embedded systems.

Fig 3.3.4.1: Network communication embedded systems

3.3.5 Different types of processing units:

The central processing unit (c.p.u) can be any one of the following microprocessor, microcontroller, digital signal processing.

- Among these Microcontroller is of low cost processor and one of the main advantage of microcontrollers is, the components such as memory, serial communication interfaces, analog to digital converters etc.., all these are built on a single chip. The numbers of external components that are connected to it are very less according to the application.
- Microprocessors are more powerful than microcontrollers. They are used in major applications with a number of tasking requirements. But the microprocessor requires many external components like memory, serial communication, hard disk, input output ports etc.., so the power consumption is also very high when compared to microcontrollers.

3.4 Applications Of Embedded Systems:

Consumer applications:

At home we use a number of embedded systems which include microwave oven, remote control, vcd players, dvd players, camera etc…

Fig 3.4.1: Automatic coffee makes equipment

Office automation:

We use systems like fax machine, modem, printer etc…

Fig 3.4.2: Fax machine

Fig 3.4.3: Printing machine

Industrial automation:

Today a lot of industries are using embedded systems for process control. In industries we design the embedded systems to perform a specific operation like monitoring temperature, pressure, humidity ,voltage, current etc.., and basing on these monitored levels we do control other devices, we can send information to a centralized monitoring station.

Fig 3.4.4: Robot

In critical industries where human presence is avoided there we can use robot**s** which are programmed to do a specific operation.

Computer networking:

Embedded systems are used as bridges routers etc..

Fig 3.4.6: Computer networking

Tele communications:

Cell phones, web cameras etc.

Fig 3.4.7: Cell Phone

Fig 3.4.8: Web camera

4. SCHEMATIC DIAGRAM AND HARDWARE DESCRIPTION

4.1 System Architecture

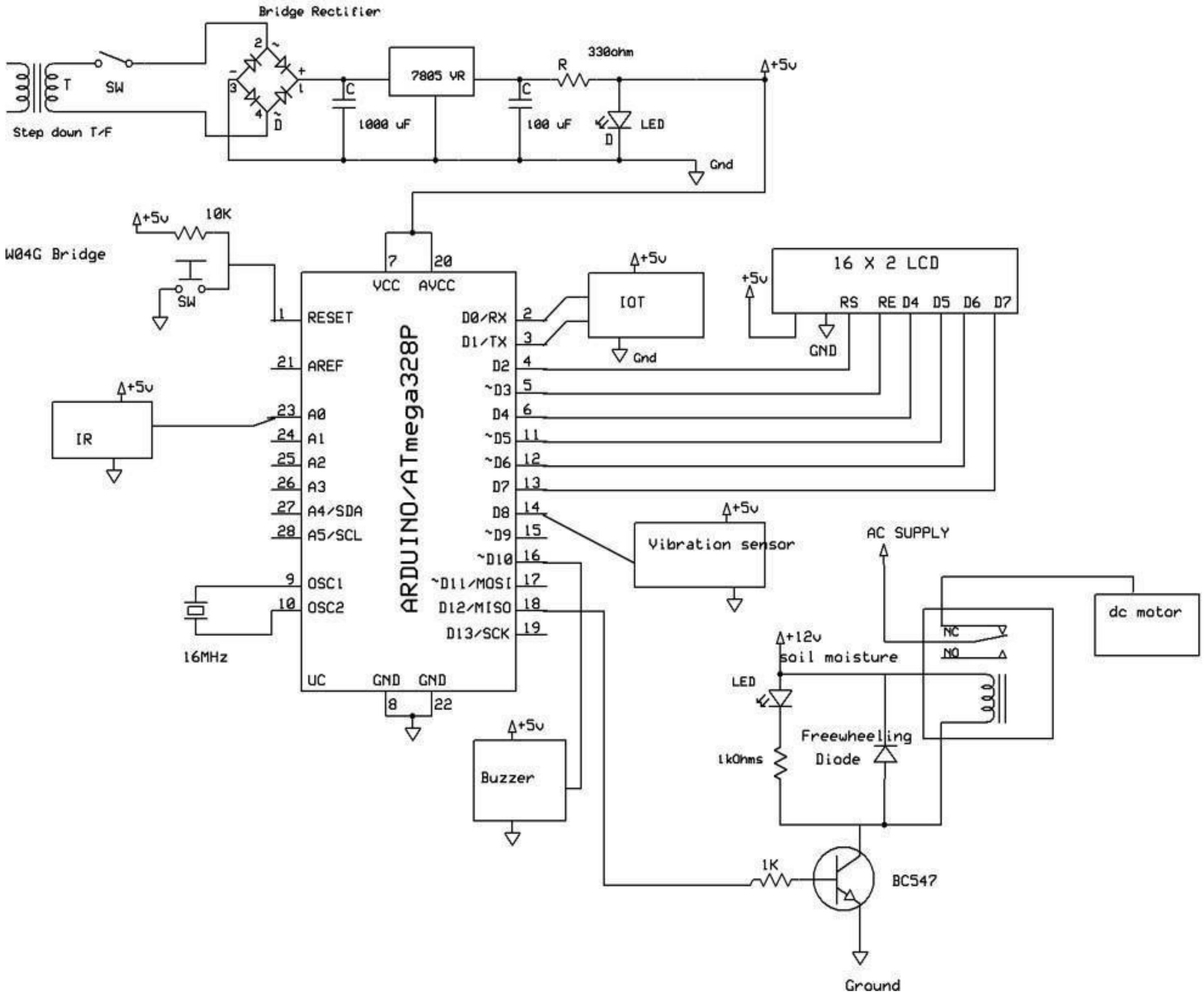


Fig: 4.1.1 System Architecture

Memory: It has 8 Kb of Flash program memory (10,000 Write/Erase cycles durability), 512Bytes of EEPROM (100,000 Write/Erase Cycles). 1Kbyte Internal SRAM I/O Ports: 23 I/ line can be obtained from three ports; namely Port B, Port C and Port D.

Interrupts: Two External Interrupt source, located at port D. 19 different interrupt vectors supporting 19 events generated by internal peripherals. Timer/Counter: Three Internal Timers are available, two 8 bit, one 16 bit, offering various operating modes and supporting internal or external clocking.

SPI (Serial Peripheral interface): ATmega8 holds three communication devices integrated. One of them is Serial Peripheral Interface. Four pins are assigned to Atmega8 to implement this scheme of communication.

USART: One of the most powerful communication solutions is USART and ATmega8 supports both synchronous and asynchronous data transfer schemes. It has three pins assigned for that. In many projects, this module is extensively used for PC-Micro controller communication.

TWI (Two Wire Interface): Another communication device that is present in ATmega8 is Two Wire Interface. It allows designers to set up a commutation between two devices using just two wires along with a common ground connection, As the TWI output is made by means of open collector outputs, thus external pull up resistors are required to make the circuit.

Analog Comparator: A comparator module is integrated in the IC that provides comparison facility between two voltages connected to the two inputs of the Analog comparator via External pins attached to the micro controller.

Analog to Digital Converter: Inbuilt analog to digital converter can convert an analog input signal into digital data of 10bit resolution. For most of the low end application, this much resolution is enough.

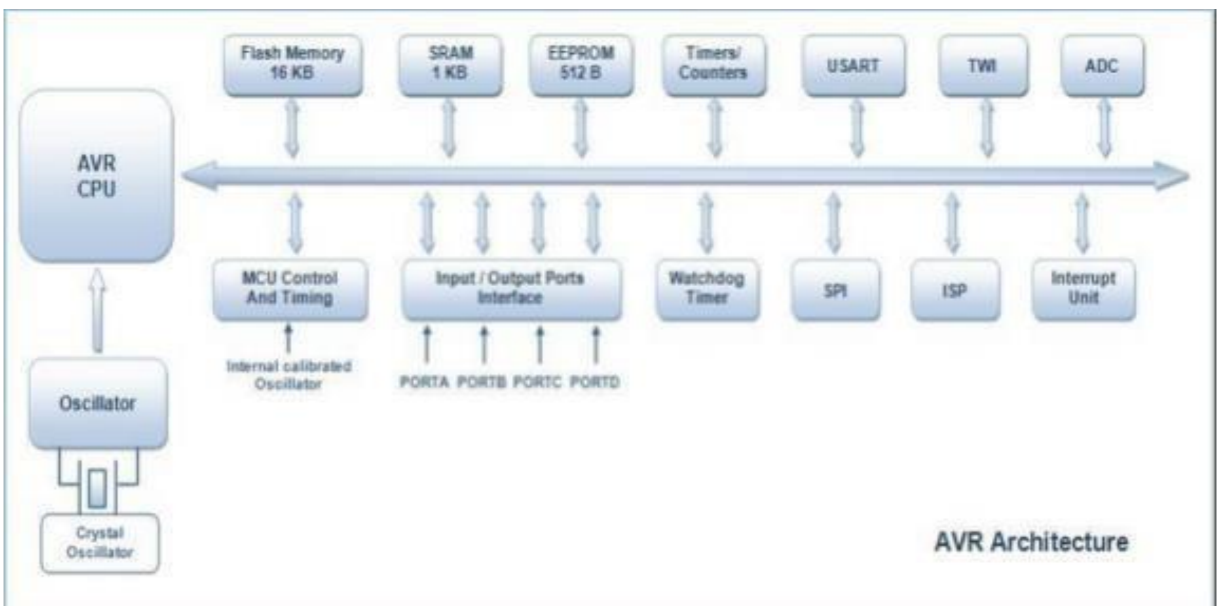


Fig 4.1.2:AVR Architecture

Features:

High-performance, Low-power Atmel®AVR® 8-bit Microcontroller

Advanced RISC Architecture

130 Powerful Instructions – Most Single-clock Cycle Execution

32 × 8 General Purpose Working Registers

Fully Static Operation

Up to 16MIPS Throughput at 16MHz

On-chip 2-cycle Multiplier

• High Endurance Non-volatile Memory segments

8Kbytes of In-System Self-programmable Flash program memory

512Bytes EEPROM

1Kbyte Internal SRAM

Write/Erase Cycles: 10,000 Flash/100,000 EEPROM

Data retention: 20 years at 85°C/100 years at 25°C(1)

Optional Boot Code Section with Independent Lock Bits

In-System Programming by On-chip Boot Program

True Read-While-Write Operation

Programming Lock for Software Security

• Peripheral Features

Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode

One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode

Real Time Counter with Separate Oscillator

Three PWM Channels

8-channel ADC in TQFP and QFN/MLF package

Eight Channels 10-bit Accuracy

6-channel ADC in PDIP package

Six Channels 10-bit Accuracy

Byte-oriented Two-wire Serial Interface

Programmable Serial USART

Master/Slave SPI Serial Interface

Programmable Watchdog Timer with Separate On-chip Oscillator

On-chip Analog Comparator

• Special Microcontroller Features

Power-on Reset and Programmable Brown-out Detection

Internal Calibrated RC Oscillator

External and Internal Interrupt Sources

Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby

• I/O and Packages

23 Programmable I/O Lines

28-lead PDIP, 32-lead TQFP, and 32-pad QFN/MLF

• Operating Voltages

2.7V - 5.5V (ATmega8L)

4.5V - 5.5V (ATmega8)

• Speed Grades

0 - 8MHz (ATmega8L)

0 - 16MHz (ATmega8)

• Power Consumption at 4Mhz, 3V, 25oC

Active: 3.6mA

– Idle Mode: 1.0mA – Power-down Mode: 0.5µA

Power Supply:

A regulated power supply converts unregulated AC (Alternating Current) to a constant DC (Direct Current). A regulated power supply is used to ensure that the output remains constant even if the input changes. A regulated DC power supply is also known as a linear power supply; it is an embedded circuit and consists of various blocks. The regulated power supply will accept an AC input and give a constant DC output. The figure below shows the block diagram of a typical regulated DC power supply.

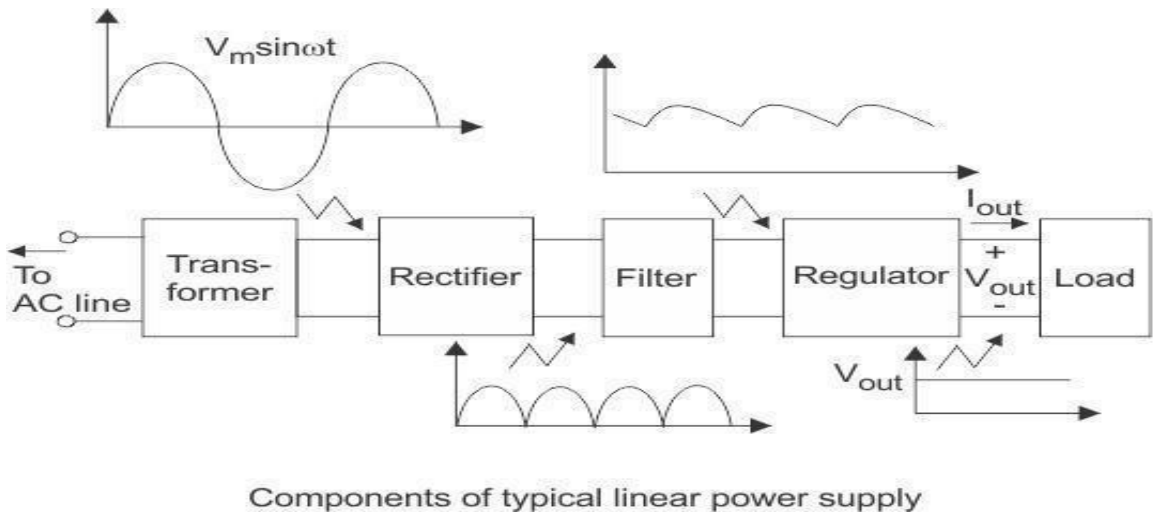


Fig 4.1.3 Block diagram of power supply

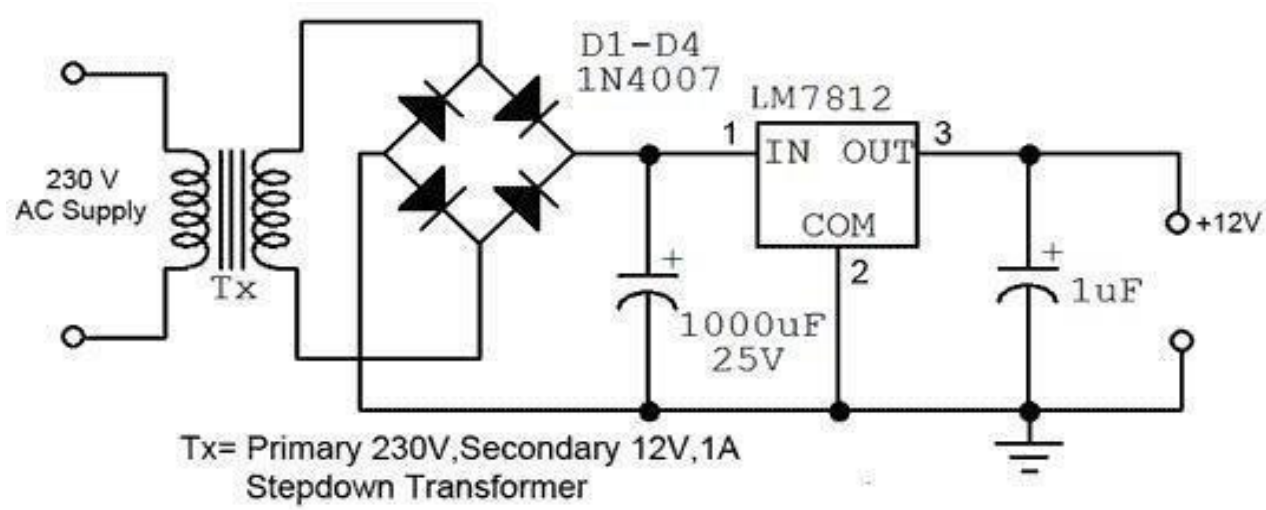
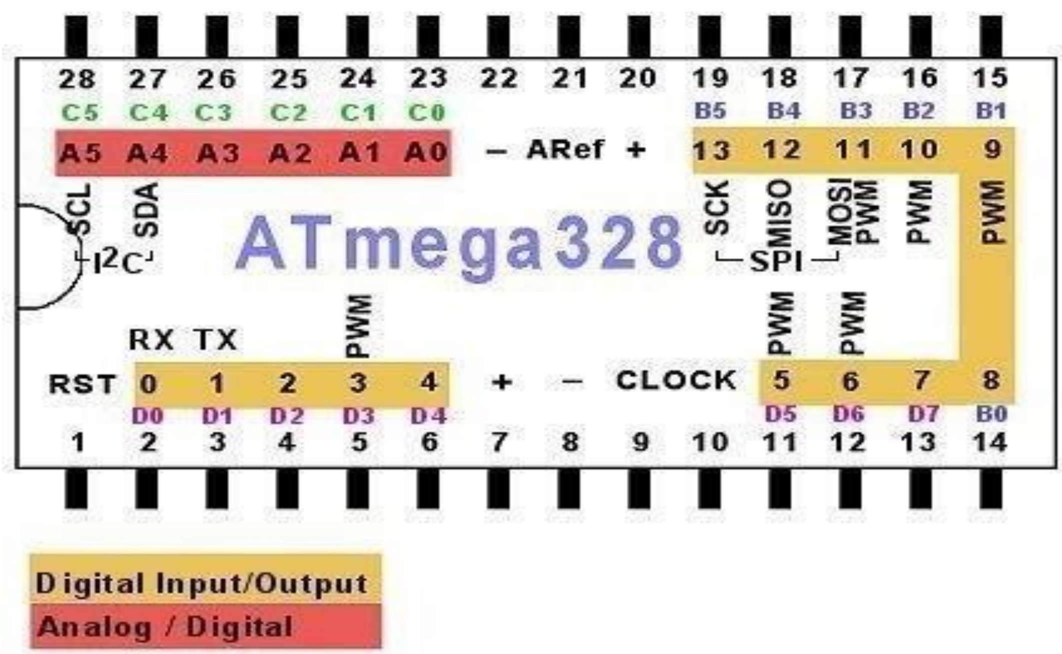


Fig 4.1.4 Power supply circuit diagram

4.2 Arduino:

Fig 4.2.1 ATMEGA 328 Microcontrollers

4.3 Pin Diagram:



Figu 4.3.1: ATMEGA 328 PIN diagram

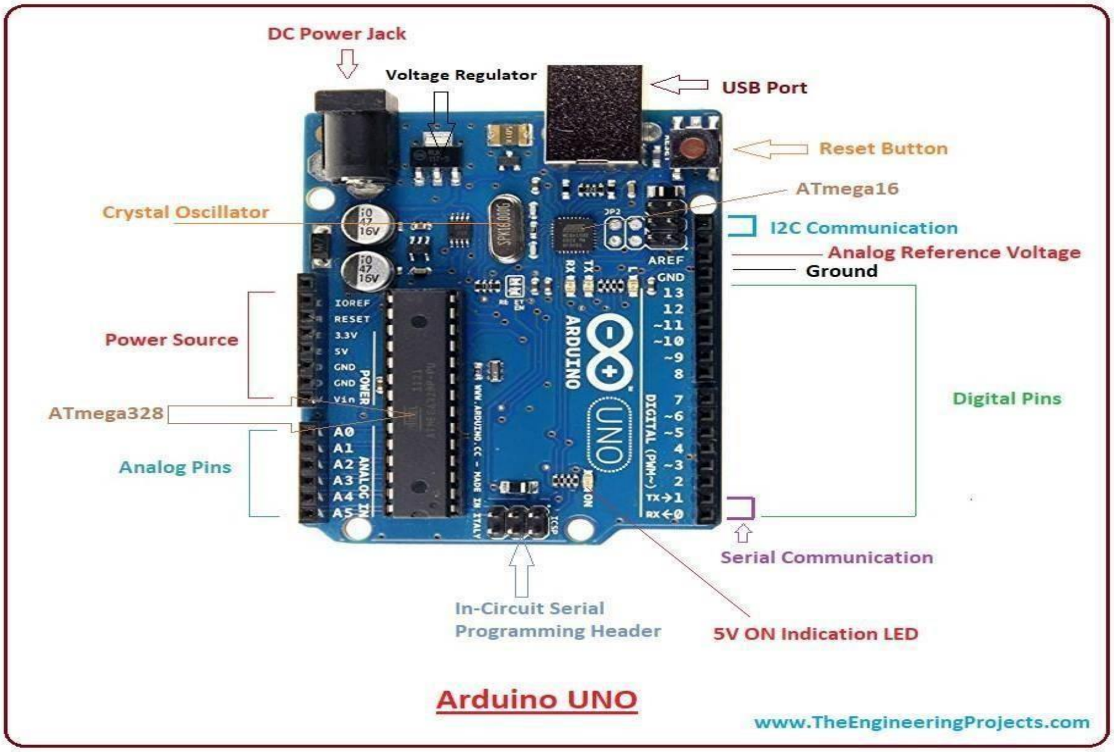


Fig 4.3.2: ARDUINO Development Board

VCC:

Digital supply voltage magnitude of the voltage range between 4.5 to 5.5 V for the ATmega8 and 2.7 to 5.5 V for ATmega8L

GND:

Ground Zero reference digital voltage supply.

PORTB (PB7.. PB0) :

PORTB is a port I / O two-way (bidirectional) 8-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTB pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

PORTC (PC5.. PC0) :

PORTC is a port I / O two-way (bidirectional) 7-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTC pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

PC6/RESET:

If RSTDISBL Fuse programmed, PC6 then serves as a pin I / O but with different characteristics. PC0 to PC5 If Fuse RSTDISBL not programmed, then serves as input Reset PC6. LOW signal on this pin with a minimum width of 1.5 microseconds will bring the microcontroller into reset condition, although the clock is not running.

PORTD (PD7.. PD0) :

PORTD is a port I / O two-way (bidirectional) 8-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTD pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

Microcontroller: Microcontroller can be termed as a single on chip computer which includes number of peripherals like RAM, EEPROM, Timers etc., required to perform some predefined tasks.

The computer on one hand is designed to perform all the general purpose tasks on a single machine like you can use a computer to run a software to perform calculations or you can use a computer to store some multimedia file or to access internet through the browser, whereas the microcontrollers are meant to perform only the specific tasks, for e.g., switching the AC off automatically when room temperature drops to a certain defined limit and again turning it ON when temperature rises above the defined limit. There are number of popular families of microcontrollers which are used in different applications as per their capability and feasibility to perform the desired task, most common of these are 8051, **AVR** and PIC microcontrollers. In this article we will introduce you with **AVR** family of microcontrollers.

AVR was developed in the year 1996 by Atmel Corporation. The architecture of **AVR** was developed by Alf-Egil Bogen and Vegard Wollan. AVR derives its name from its developers and stands for **Alf-Egil Bogen Vegard Wollan RISC microcontroller**, also known as **A**dvanced **V**irtual **R**ISC. The AT90S8515 was the first microcontroller which was based on **AVR architecture** however the first microcontroller to hit the commercial market was AT90S1200 in the year 1997.

AVR microcontrollers are available in three categories:

1. **TinyAVR** – Less memory, small size, suitable only for simpler applications
2. **MegaAVR** – These are the most popular ones having good amount of memory (upto 256 KB), higher number of inbuilt peripherals and suitable for moderate to complex applications.
3. **XmegaAVR** – Used commercially for complex applications, which require large program memory and high speed.

Brown-out Detector:

If the Brown-out Detector is not needed in the application, this module should be turned off. If the Brown-out Detector is enabled by the BODEN Fuse, it will be enabled in all sleep modes, and hence, always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption. Refer to "Brown-out Detection" on page 38 for details on how to configure the Brown- out Detector.Internal Voltage Reference the Internal Voltage Reference will be enabled when needed by the Brown- out Detector, the Analog Comparator, or the ADC. If these modules are disabled as described in the sections above, the internal voltage reference will be disabled, and it will not be consuming power. Whenturned on again, the user must allow the reference to start up before the output is used. If the reference is kept on in sleep mode, the output can be used immediately. Refer to "Internal Voltage Reference" onpage 40 for details on the start-up time. Watchdog Timer If the Watchdog Timer is not needed in the application, this module should be turned off.

If the Watchdog Timer is enabled, it will be enabled in all sleep modes, and hence, always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption. Refer to "Watchdog Timer" on page 41 for details on how to configure the Watchdog Timer. Port Pins When entering a sleep mode, all port pins should be configured to use minimum power.

The most important thing is then to ensure that no pins drive resistive loads. In sleep modes where the both the I/O clock (clkI/O) and the ADC clock (clkADC) are stopped, the input buffers of the device will be disabled. This ensures that no power is consumed by the input logic when not needed. In some cases, the input logic is needed for detecting wake-up conditions, and it will then be enabled. Refer to the section "Digital Input Enable and Sleep Modes" on page 53 for details on which pins are enabled. Ifthe input buffer is enabled and the input signal is left floating or have an analog signal level close to VCC/2, the input buffer will use excessive power.

Power-on Reset:

A Power-on Reset (POR) pulse is generated by an On-chip detection circuit. The detection level is defined in Table 15. The POR is activated whenever VCC is below the detection level. The POR circuitcan be used to trigger the Start-up Reset, as well as to detect a failure in supply voltage.

A Power-on Reset (POR) circuit ensures that the device is reset from Power-on. Reaching the Power-onReset threshold voltage invokes the delay counter, which determines how long the device is kept in RESET after VCC rise. The RESET signal is activated again, without any delay, when VCC decreases below the detection level.

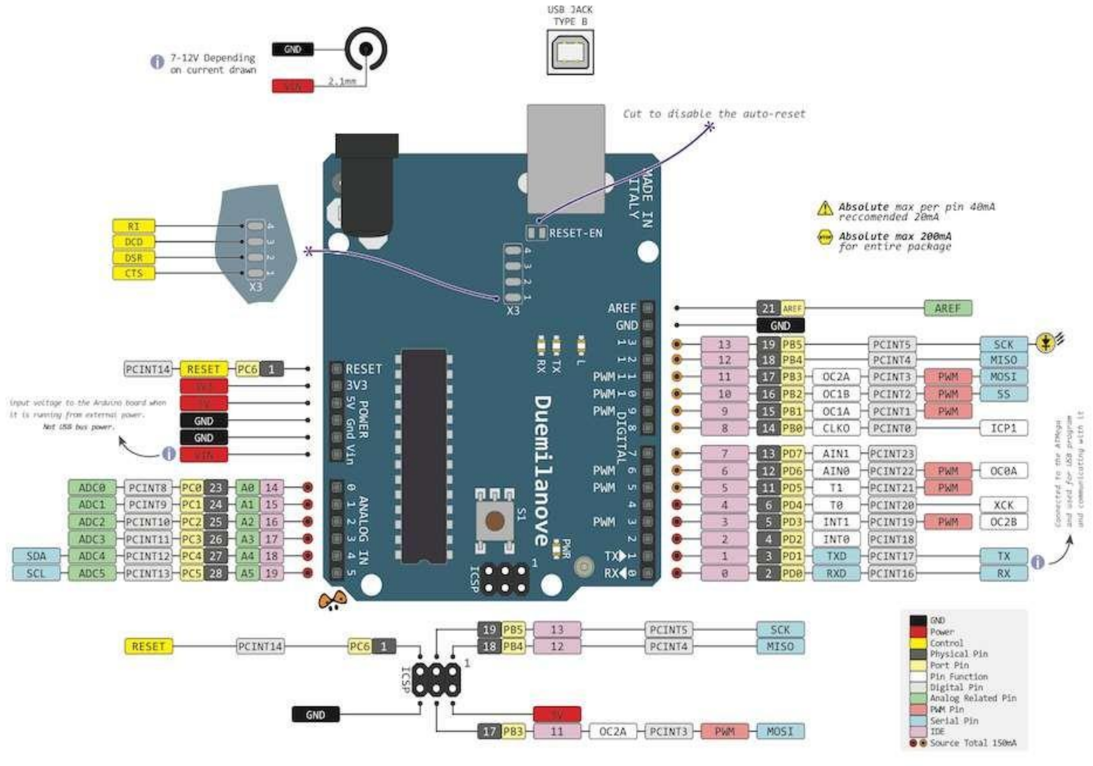


Fig 4.3.3: Pin Diagram of ATMEGA328

VCC

Digital supply voltage magnitude of the voltage range between 4.5 to 5.5 V for the ATmega8 and 2.7 to

5.5 V for ATmega8L

GND

Ground Zero reference digital voltage supply.

PORTB (PB7.. PB0)

PORTB is a port I / O two-way (bidirectional) 8-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTB pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

PORTC (PC5.. PC0)

PORTC is a port I / O two-way (bidirectional) 7-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTC pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

PC6/RESET

If RSTDISBL Fuse programmed, PC6 then serves as a pin I / O but with different characteristics. PC0 to PC5 If Fuse RSTDISBL not programmed, then serves as input Reset PC6. LOW signal on this pin with a minimum width of 1.5 microseconds will bring the microcontroller into reset condition, althoughthe clock is not running.

PORTD (PD7.. PD0)

PORTD is a port I / O two-way (bidirectional) 8-bit with internal pull-up resistor can be selected. This port output buffers have symmetrical characteristics when used as a source or sink. When used as an input, the pull-pin low externally will emit a current if the pull-up resistor is activated it. PORTD pins will be in the condition of the tri-state when RESET is active, although the clock is not running.

RESET

Reset input pin. LOW signal on this pin with a minimum width of 1.5 microseconds will bring the microcontroller into reset condition, although the clock is not running. Signal with a width of less than

1.5 microseconds does not guarantee a Reset condition.

AVCC

AVCC is the supply voltage pin for the ADC, PC3 .. PC0, and ADC7..ADC6. This pin should be connected to VCC, even if the ADC is not used. If the ADC is used, AVCC should be connected to VCCthrough a low-pass filter to reduce noise.

Aref

Analog Reference pin for the ADC.

ADC7 .. ADC6

ADC analog input there is only on ATmega8 with TQFP and QFP packages / MLF.

PORTS

Term "port" refers to a group of pins on a microcontroller which can be accessed simultaneously, or onwhich we can set the desired combination of zeros and ones, or read from them an existing status. Ports represent physical connection of Central Processing Unit with an outside world. Microcontroller uses them.

The Atmega8 has 23 I/O ports which are organized into 3 groups:

- Port B (PB0 to PB7)
- Port C (PC0 to PC6)
- Port D (PD0 to PD7)

We will use mainly 3 registers known as **DDRX, PORTX** & **PINX**. We have total four PORTs on ATmega16. They are **PORTA, PORTB, PORTC** and **PORTD**. They are multifunctional pins. Each ofthe pins in each port (total 32) can be treated as input or output pin.

Applications

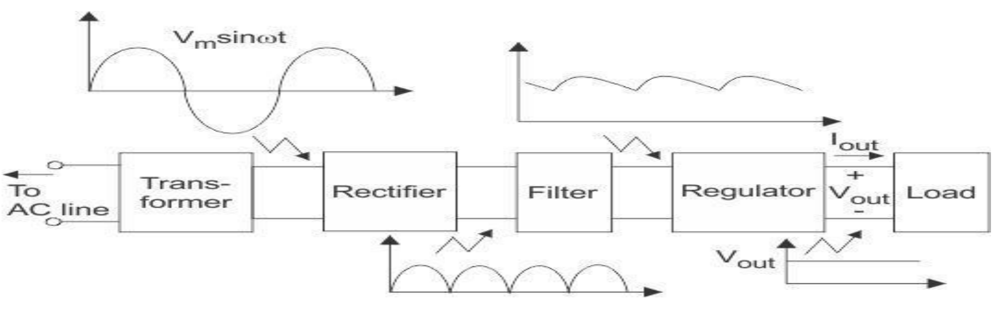
AVR microcontroller perfectly fits many uses, from automotive industries and controlling home appliances to industrial instruments, remote sensors, electrical door locks and safety devices. It is also ideal for smart cards as well as for battery supplied devices because of its low consumption.

EEPROM memory makes it easier to apply microcontrollers to devices where permanent storage of various parameters is needed (codes for transmitters, motor speed, receiver frequencies, etc.). Low cost,low consumption, easy handling, and flexibility make ATmega8 applicable even in areas where microcontrollers had not previously been considered (example: timer functions, interface replacement in larger systems, coprocessor applications, etc.).

In System Programmability of this chip (along with using only two pins in data transfer) makes possiblethe flexibility of a product, after assembling and testing have been completed. This capability can be used to create assembly-line production, to store calibration data available only after final testing, or it can be used to improve programs on finished products.

Regulated Power Supply:

A regulated power supply converts unregulated AC (Alternating Current) to a constant DC (Direct Current). A regulated power supply is used to ensure that the output remains constant even if the input changes. A regulated DC power supply is also known as a linear power supply; it is an embedded circuit and consists of various blocks. The regulated power supply will accept an AC input and give a constant DC output. The figure below shows the block diagram of a typical regulated DC power supply.



Components of typical linear power supply

Fig 4.3.4 Block diagram of power supply

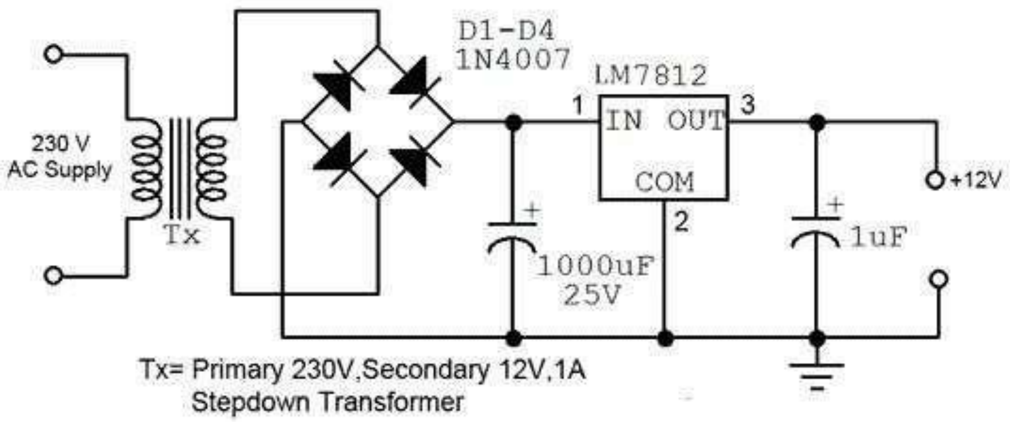


Fig 4.3.5 Power supply circuit diagram

Fig: 4.3.6 Regulated Power Supply

The basic circuit diagram of a regulated power supply (DC O/P) with led connected as load is shownin fig: 4.4.2.

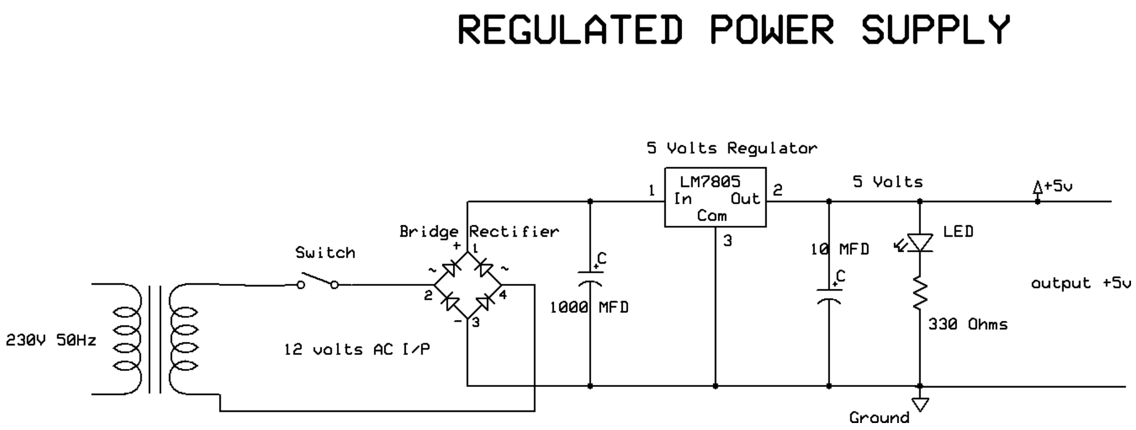


Fig 4.3.7 Circuit diagram of Regulated Power Supply with Led connectionThe

:

- 230V AC MAINS
- TRANSFORMER
- BRIDGE RECTIFIER(DIODES)
- CAPACITOR
- VOLTAGE REGULATOR (IC 7805)
- RESISTOR
- LED (LIGHT EMITTING DIODE)

Bridge full wave rectifier:

The Bridge rectifier circuit is shown in fig:3.8, which converts an ac voltage to dc voltage using both half cycles of the input ac voltage. The Bridge rectifier circuit is shown in the figure. The circuit has fourdiodes connected to form a bridge. The ac input voltage is applied to the diagonally opposite ends of thebridge. The load resistance is connected between the other two ends of the bridge.

For the positive half cycle of the input ac voltage, diodes D1 and D3 conduct, whereas diodes D2 and D4 remain in the OFF state. The conducting diodes will be in series with the load resistance R_L and hence the load current flows through $R_{L.}$

For the negative half cycle of the input ac voltage, diodes D2 and D4 conduct whereas, D1 and D3 remain OFF. The conducting diodes D2 and D4 will be in series with the load resistance R_L and hence the current flows through R_L in the same direction as in the previous half cycle. Thus a bi-directional wave is converted into a unidirectional wave.

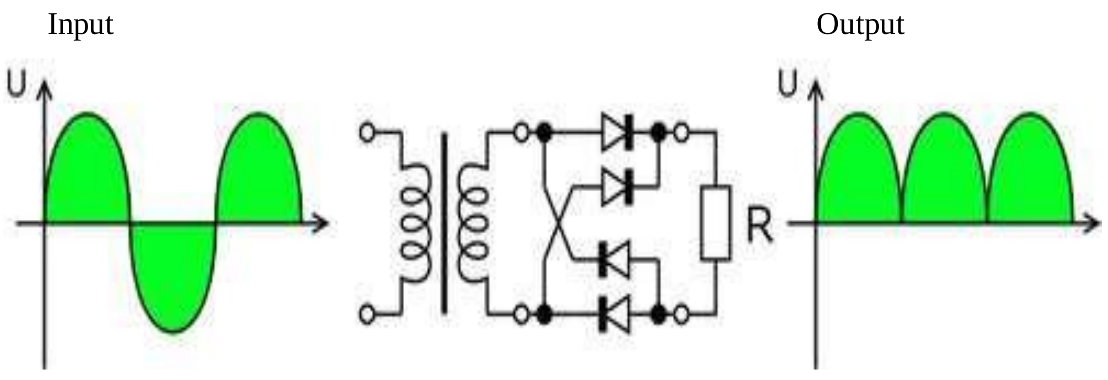


Fig 4.3.8 Bridge rectifier: a full-wave rectifier using 4 diodes.

DB107:

Now -a -days Bridge rectifier is available in IC with a number of DB107. In our project we are using anIC in place of bridge rectifier. The picture of DB 107 is shown in fig: 3.3.8.

Fig 4.3.9 DB107

Capacitor Filtration:

The process of converting a pulsating direct current to a pure direct current using filters is called as filtration._Electronic filters are electronic circuits, which perform signal-processing functions, specifically to remove unwanted frequency components from the signal, to enhance wanted ones. The **Capacitor** or sometimes referred to as a Condenser is a passive device, and one which stores energyin the form of an electrostatic field which produces a potential (static voltage) across its plates. In its basic form a capacitor consists of two parallel conductive plates that are not connected but are electricallyseparated either by air or by an insulating material called the Dielectric. When a voltage is applied to these plates, a current flow charging up the plates with electrons giving one plate a positive charge and the other plate an equal and opposite negative charge this flow of electrons to the plates is known asthe Charging Current and continues to flow until the voltage across the plates (and hence the capacitor)is equal to the applied voltage Vcc. At this point the capacitor is said to be fully charged and this is illustrated below. The construction of capacitor and an electrolytic capacitor are shown in figures 3.3.9 and 3.3.10 respectively.

requiring negative input, the LM79XX series is used. Using a pair of ‘voltage-divider’ resistors can increase the output voltage of a regulator circuit.

It is not possible to obtain a voltage lower than the stated rating. You cannot use a 12V regulator to makea 5V power supply. Voltage regulators are very robust. These can withstand over-current draw due to short circuits and over-heating. In both cases, the regulator will cut off before any damage occurs. The only way to destroy a regulator is to apply reverse voltage to its input. Reverse polarity destroys the regulator almost instantly. Fig: 3.3.11 shows voltage regulator.

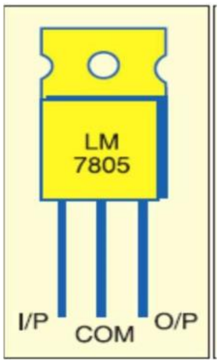


Fig 4.3.10 Voltage Regulator

Resistor:

A resistor is a two-terminal electronic component that produces a voltage across its terminals that is proportional to the electric current passing through it in accordance with Ohm's law:

$$V = IR$$

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

The primary characteristics of a resistor are the resistance, the tolerance, maximum working voltage andthe power rating. Other characteristics include temperature coefficient, noise, and inductance. Less well- known is critical resistance, the value below which power dissipation limits the maximum permitted current flow, and above which the limit is applied voltage. Critical resistance is determined by the design,materials, and dimensions of the resistor.

Dissipated by a resistor (or the equivalent resistance of a resistor network) is calculated using the following:

$$P = I^2 R = IV = \frac{V^2}{R}$$

Fig 4.3.11 Resistor

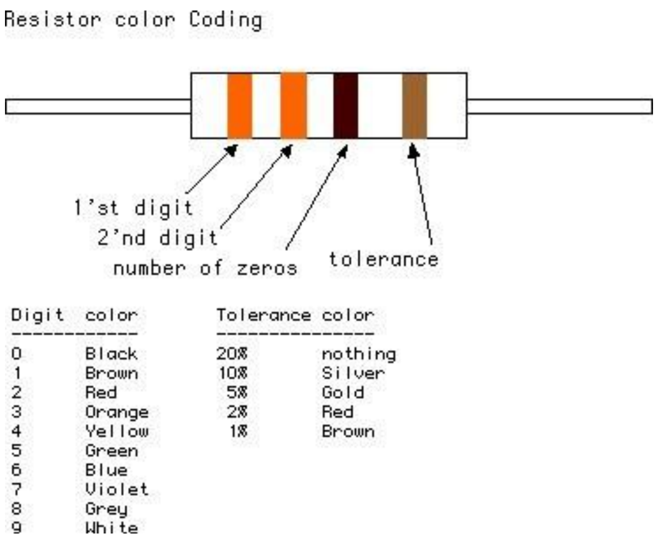


Fig 4.3.12 Color Bands in Resistor

LED:

A light-emitting diode (LED) is a semiconductor light source. LEDs are used as indicator lamps in manydevices and are increasingly used for lighting. Introduced as a practical electronic component in 1962, early LEDs emitted low-intensity red light, but modern versions are available across the visible, ultraviolet, and infrared wavelengths, with very high brightness. The internal structure and parts of a ledare shown in figures 3.4.1 and 3.4.2 respectively.

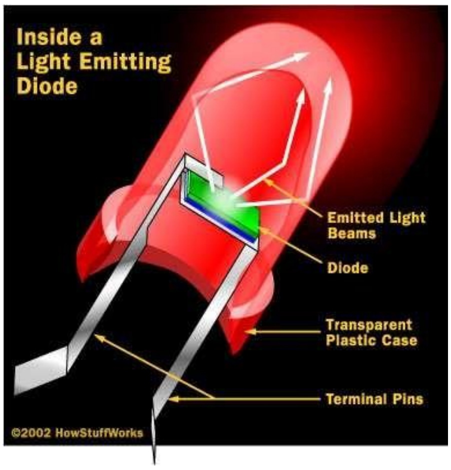


Fig 4.3.16 Inside a LED

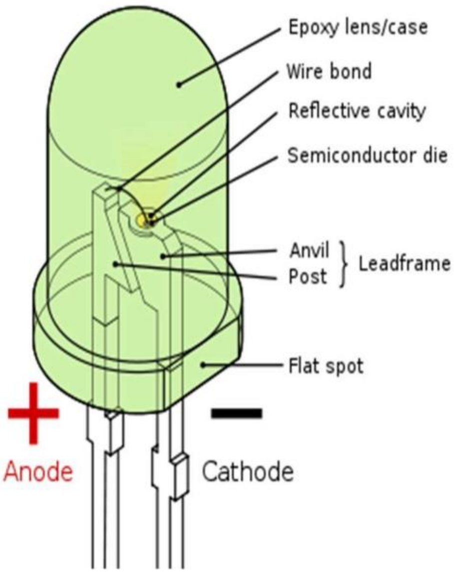


Fig 4.3.17 Parts of a LED

4.4 Buzzer

Basically, the sound source of a piezoelectric sound component is a piezoelectric diaphragm. A piezoelectric diaphragm consists of a piezoelectric ceramic plate which has electrodes on both sides anda metal plate (brass or stainless steel, etc.). A piezoelectric ceramic plate is attached to a metal plate withadhesives. Applying D.C. voltage between electrodes of a piezoelectric diaphragm causes mechanical distortion due to the piezoelectric effect. For a misshaped piezoelectric element, the distortion of the piezoelectric element expands in a radial direction. And the piezoelectric diaphragm bends toward the direction. The metal plate bonded to the piezoelectric element does not expand. Conversely, when the piezoelectric element shrinks, the piezoelectric diaphragm bends in the direction Thus, when AC voltage is applied across electrodes, the bending is repeated, producing sound waves in the air. To interface a buzzer the standard transistor interfacing circuit is used. Note that if a different power supply is used forthe buzzer, the 0V rails of each power supply must be connected to provide a common reference.

If a battery is used as the power supply, it is worth remembering that piezo sounders draw much less current than buzzers. Buzzers also just have one ‘tone’, whereas a piezo sounder can create sounds of many different tones. To switch on buzzer -high 1 To switch off buzzer -low 1.

Fig: 4.4.1 Picture of buzzer

LCD DISPLAY

LCD Background:

One of the most common devices attached to a micro controller is an LCD display. Some of the most common LCDs connected to the many microcontrollers are 16x2 and 20x2 displays. This means 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

Basic 16x 2 Characters LCD

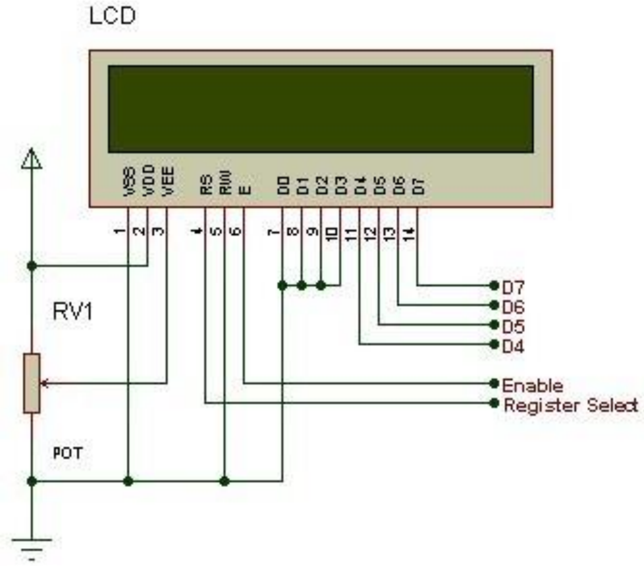


Fig: 4.4.2 LCD Pin diagram

		1 = Read from LCD module
Pin no. 6	**EN**	Enable signal
Pin no. 7	**D0**	Data bus line 0 (LSB)
Pin no. 8	**D1**	Data bus line 1
Pin no. 9	**D2**	Data bus line 2
Pin no. 10	**D3**	Data bus line 3
Pin no. 11	**D4**	Data bus line 4
Pin no. 12	**D5**	Data bus line 5
Pin no. 13	**D6**	Data bus line 6
Pin no. 14	**D7**	Data bus line 7 (MSB)

Table 4.4.3: Character LCD pins with Microcontroller

The LCD requires 3 control lines as well as either 4 or 8 I/O lines for the data bus. The user may selectwhether the LCD is to operate with a 4-bit data bus or an 8-bit data bus. If a 4-bit data bus is used the LCD will require a total of 7 data lines (3 control lines plus the 4 lines for the data bus). If an 8-bit databus is used the LCD will require a total of 11 data lines (3 control lines plus the 8 lines for the data bus).The three control lines are referred to as **EN**, **RS**, and **RW**.

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

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

Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, andDB7.

Schematic:

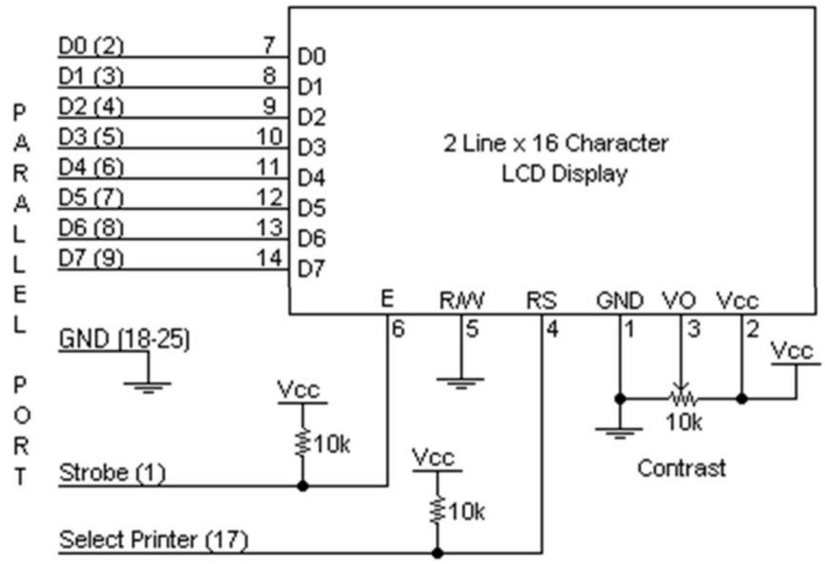


Fig 4.4.4 Schematic diagram of LCD display

Circuit Description:

Above is the quite simple schematic. The LCD panel's Enable and Register Select is connected to the Control Port. The Control Port is an open collector / open drain output. While most Parallel Ports have internal pull-up resistors, there is a few which don't. Therefore, by incorporating the two 10K external pull up resistors, the circuit is more portable for a wider range of computers, some of which may have no internal pull up resistors.

We make no effort to place the Data bus into reverse direction. Therefore, we hard wire the R/W line ofthe LCD panel, into write mode. This will cause no bus conflicts on the data lines. As a result, we cannotread back the LCD's internal Busy Flag which tells us if the LCD has accepted and finished processing the last instruction. This problem is overcome by inserting known delays into our program.

The 10k Potentiometer controls the contrast of the LCD panel. Nothing fancy here. As with all the examples, I've left the power supply out. We can use a bench power supply set to 5v or use an onboard

+5 regulator. Remember a few de-coupling capacitors, especially if we have trouble with the circuit working properly.

SETB RW

Handling the EN control line:

As we mentioned above, the EN line is used to tell the LCD that we are ready for it to execute an instruction that we've prepared on the data bus and on the other control lines. Note that the EN line must be raised/ lowered before/after each instruction sent to the LCD regardless of whether that instruction isread or write text or instruction.

Thus, before we interact in any way with the LCD, we will always bring the **EN** line low with the following instruction:

CLR EN

And once we've finished setting up our instruction with the other control lines and data bus lines, we'll always bring this line high:

SETB EN

The line must be left high for the amount of time required by the LCD as specified in its datasheet. Thisis normally on the order of about 250 nanoseconds but checks the datasheet. In the case of a typical microcontroller running at 12 MHz, However, faster microcontrollers (such as the DS89C420 which executes an instruction in 90 nanoseconds given an 11.0592 MHz crystal)

Checking the busy status of the LCD:

As previously mentioned, it takes a certain amount of time for each instruction to be executed by the LCD. The delay varies depending on the frequency of the crystal attached to the oscillator input of the LCD as well as the instruction which is being executed.

While it is possible to write code that waits for a specific amount of time to allow the LCD to execute instructions, this method of "waiting" is not very flexible. If the crystal frequency is changed, the software will need to be modified. A more robust method of programming is to use the "Get LCD Status" command to determine whether the LCD is still busy executing the last instruction received.

The "Get LCD Status" command will return to us two titbits of information; the information that is useful to us right now is found in DB7. In summary, when we issue the "Get LCD Status" command the LCD will immediately raise DB7 if it's still busy executing a command or lower DB7 to indicate that the LCD is no longer occupied. Thus, our program can query the LCD until DB7 goes low, indicating the LCD is no longer busy. At that point we are free to continue and send the next command.

ROBOT DC MOTOR

A dc motor uses electrical energy to produce mechanical energy, very typically through the interaction of magnetic fields and current-carrying conductors. The reverse process, producing electrical energy from mechanical energy, is accomplished by an alternator, generator or dynamo. Many types of electric motors can be run as generators, and vice versa. The input of a DC motor is current/voltage and its output is torque (speed).

Fig 4.4.5 : DC Motor

The DC motor has two basic parts: the rotating part that is called the armature and the stationary part that includes coils of wire called the field coils. The stationary part is also called the stator. Figure shows a picture of a typical DC motor, Figure shows a picture of a DC armature, and Fig shows a picture of a typical stator. From the picture you can see the armature is made of coils of wire wrapped around the core, and the core has an extended shaft that rotates on bearings. You should also notice that the ends of each coil of wire on the armature are terminated at one end of the armature. The termination points are called the commutator, and this is where the brushes make electrical contact to bring electrical current from the stationary part to the rotating part of the machine.

Operation:

The DC motor you will find in modem industrial applications operates very similarly to the simple DC motor described earlier in this chapter. Figure 12-9 shows an electrical diagram of a simple DC motor. Notice that the DC voltage is applied directly to the field winding and the brushes. The armature and the field are both shown as a coil of wire. In later diagrams, a field resistor will be added in series with the field to control the motor speed.

When voltage is applied to the motor, current begins to flow through the field coil from the negative terminal to the positive terminal. This sets up a strong magnetic field in the field winding.

Current also begins to flow through the brushes into a commutator segment and then through an armature coil. The current continues to flow through the coil back to the brush that is attached to other end of the coil and returns to the DC power source. The current flowing in the armature coil sets up a strong magnetic field in the armature.

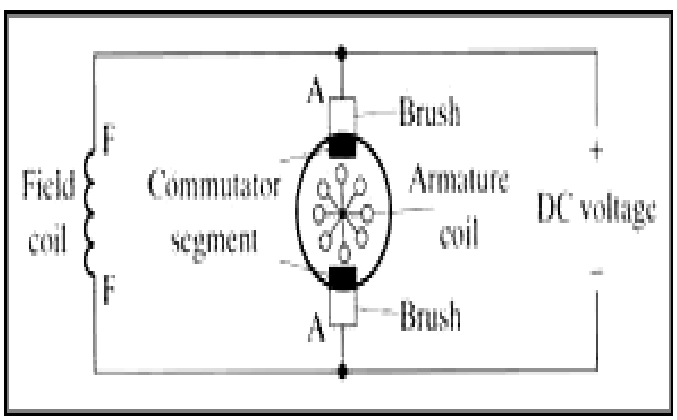


Fig 4.4.6 : Simple electrical diagram of DC motor

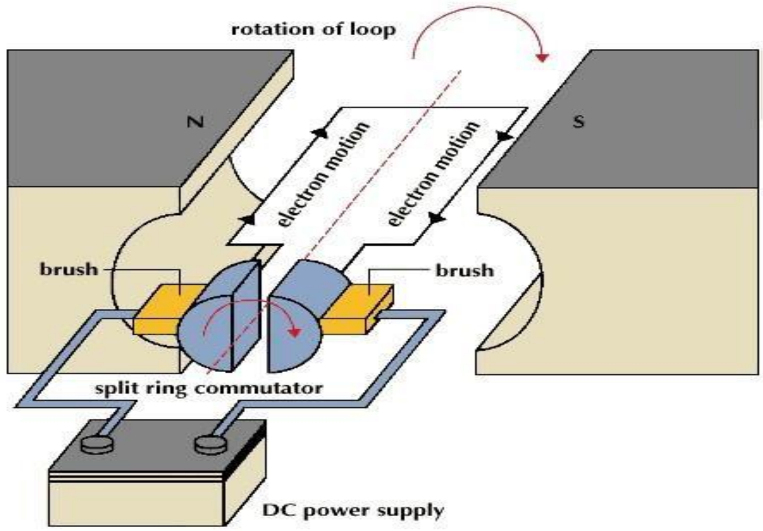


Fig 4.4.7: Operation of a DC Motor

The magnetic field in the armature and field coil causes the armature to begin to rotate. This occurs by the unlike magnetic poles attracting each other and the like magnetic poles repelling each other. As the armature begins to rotate, the commutator segments will also begin to move under the brushes. As an individual commutator segment moves under the brush connected to positive voltage, it will become positive, and when it moves under a brush connected to negative voltage it will become negative. In this way, the commutator segments continually change polarity from positive to negative. Since the commutator segments are connected to the ends of the wires that make up the field winding in the armature, it causes the magnetic field in the armature to change polarity continually from North Pole to South Pole. The commutator segments and brushes are aligned in such a way that the switch in polarity of the armature coincides with the location of the armature's magnetic field and the field winding's magnetic field. The switching action is timed so that the armature will not lock up magnetically with the field. Instead the magnetic fields tend to build on each other and provide additional torque to keep the motor shaft rotating.

When the voltage is de-energized to the motor, the magnetic fields in the armature and the field winding will quickly diminish and the armature shaft's speed will begin to drop to zero. If voltage is applied to the motor again, the magnetic fields will strengthen and the armature will begin to rotate again.

Fig 4.4.8: Model of a car

Buzzer

Basically, the sound source of a piezoelectric sound component is a piezoelectric diaphragm. A piezoelectric diaphragm consists of a piezoelectric ceramic plate which has electrodes on both sides and a metal plate (brass or stainless steel, etc.). A piezoelectric ceramic plate is attached to a metal plate with adhesives. Applying D.C.

If a battery is used as the power supply, it is worth remembering that piezo sounders draw much less current than buzzers. Buzzers also just have one ‘tone’, whereas a piezo sounder is able to create sounds of many different tones.

To switch on buzzer -high 1

To switch off buzzer -low 1

Notice (Handling) In Using Self Drive Method

When the piezoelectric buzzer is set to produce intermittent sounds, sound may be heard continuously even when the self drive circuit is turned ON / OFF at the "X" point shown in Fig. 9. This is because of the failure of turning off the feedback voltage.

Build a circuit of the piezoelectric sounder exactly as per the recommended circuit shown in the catalog of the transistor and circuit constants are designed to ensure stable oscillation of the piezoelectric sounder.

Design switching which ensures direct power switching.

The self drive circuit is already contained in the piezoelectric buzzer. So there is no need to prepare another circuit to drive the piezoelectric buzzer.

Rated voltage (3.0 to 20Vdc) must be maintained. Products which can operate with voltage higher than 20Vdc are also available.

is essential to adjust sound pressure, place a capacitor (about 1μF) in parallel with the piezo buzzer.

Do not close the sound emitting hole on the front side of casing.

Carefully install the piezo buzzer so that no obstacle is placed within 15mm from the sound release hole on the front side of the casing.

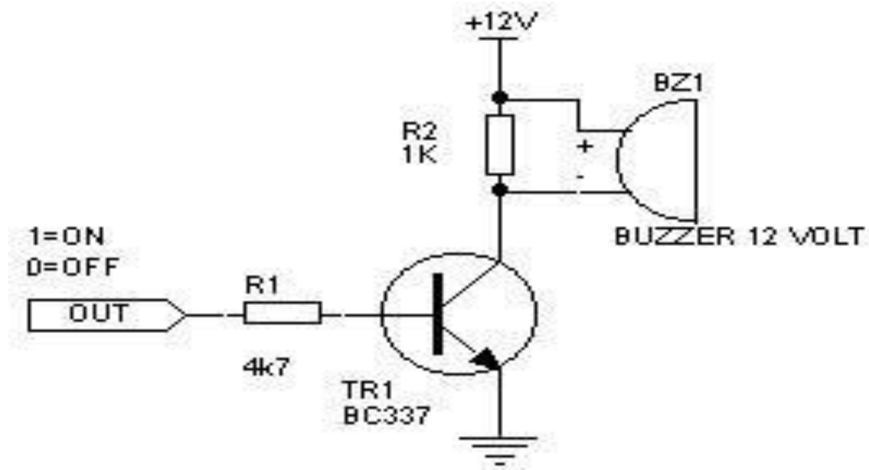


Fig 4.4.9: buzzer

4.5 LCD DISPLAY LCD Background:

One of the most common devices attached to a micro controller is an LCD display. Some of the most common LCD’s connected to the many microcontrollers are 16x2 and 20x2 displays. This means 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

Basic 16x 2 Characters LCD

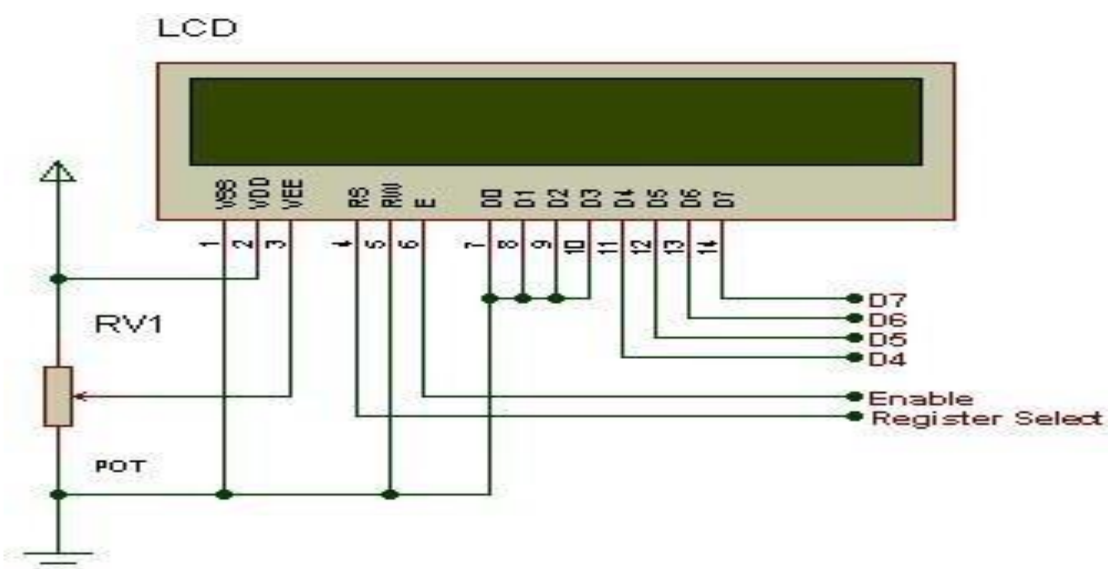


Fig 4.5.1: LCD Pin diagram

Pin description:

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

Table 1: Character LCD pins with Microcontroller

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

If an 8-bit data bus is used the LCD will require a total of 11 data lines (3 control lines plus the 8 lines for the data bus).

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

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

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

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

Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, and DB7.

Schematic:

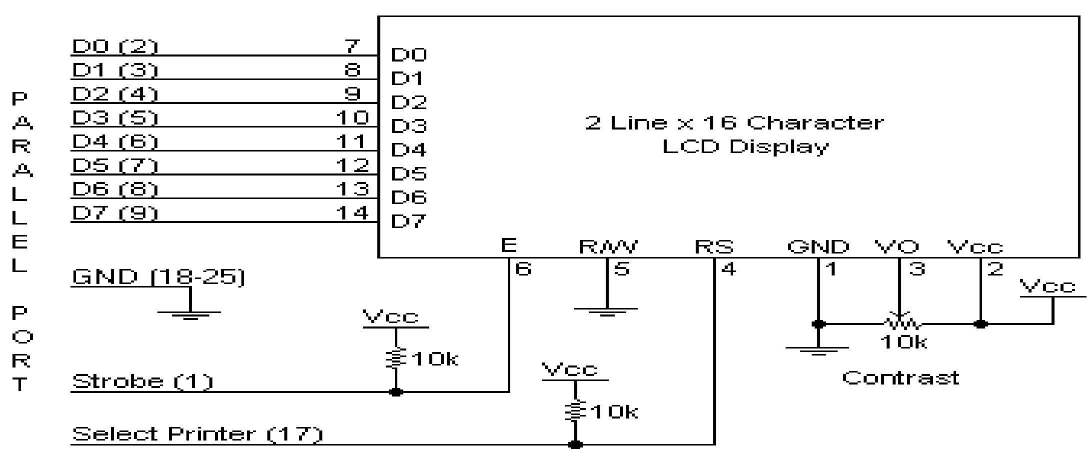


Fig 4.5.2: Schematic Diagram

Circuit Description:

Above is the quite simple schematic. The LCD panel's Enable and Register Select is connected to the Control Port. The Control Port is an open collector / open drain output. While most Parallel Ports have internal pull-up resistors, there is a few which don't. Therefore by incorporating the two 10K external pull up resistors, the circuit is more portable for a wider range of computers, some of which may have no internal pull up resistors. We make no effort to place the Data bus into reverse direction. Therefore we hard wire the R/W line of the LCD panel, into write mode. This will cause no bus conflicts on the data lines. As a result we cannot read back the LCD's internal Busy Flag which tells us if the LCD has accepted and finished processing the last instruction. This problem is overcome by inserting known delays into our program. The 10k Potentiometer controls the contrast of the LCD panel. Nothing fancy here. As with all the examples, I've left the power supply out. We can use a bench power supply set to 5v or use an onboard +5 regulator. Remember a few de-coupling capacitors, especially if we have trouble with the circuit working properly.

4.6 METAL DETECTOR

A **metal detector** is an electronic instrument which detects the presence of metal nearby. Metal detectors are useful for finding metal inclusions hidden within objects, or metal objects buried underground. They often consist of a handheld unit with a sensor probe which can be swept over the ground or other objects. If the sensor comes near a piece of metal this is indicated by a changing tone in earphones, or a needle moving on an indicator. Usually the device gives some indication of distance; the closer the metal is, the higher the tone in the earphone or the higher the needle goes. Another common type are stationary "walk through" metal detectors used for security screening at access points in prisons, courthouses, and airports to detect concealed metal weapons on a person's body.

The simplest form of a metal detector consists of an oscillator producing an alternating current that passes through a coil producing an alternating magnetic field. If a piece of electrically conductive metal is close to the coil, eddy currents will be induced in the metal, and this produces a magnetic field of its own. If another coil is used to measure the magnetic field (acting as a magnetometer), the change in the magnetic field due to the metallic object can be detected.

The first industrial metal detectors were developed in the 1960s and were used extensively for mineral prospecting and other industrial applications. Uses include detecting land mines, the detection of weapons such as knives and guns (especially in airport security), geophysical prospecting, archaeology and treasure hunting.

Metal detectors are also used to detect foreign bodies in food, and in the construction industry to detect steel reinforcing bars in concrete and pipes and wires buried in walls and floors.

Fig 4.6.1: Metal detectors

5. SIMULATION TOOLS

This project is implemented using following software's:

Express PCB – for designing circuit

Arduino IDE compiler - for compilation part • Proteus 7 (Embedded C) – for simulation part

5.1 Express PCB:

Breadboards are great for prototyping equipment as it allows great flexibility to modify a design when needed; however the final product of a project, ideally should have a neat PCB, few cables, and survive a shake test. Not only is a proper PCB neater but it is also more durable as there are no cables which can yank loose.

Express PCB is a software tool to design PCBs specifically for manufacture by the company Express PCB (no other PCB maker accepts Express PCB files). It is very easy to use, but it does have several limitations.

It can be likened to more of a toy then a professional CAD program.

It has a poor part library (which we can work around)

It cannot import or export files in different formats

It cannot be used to make prepare boards for DIY production

Express PCB has been used to design many PCBs (some layered and with surface-mount parts. Print out PCB patterns and use the toner transfer method with an Etch Resistant Pen to make boards. However, Express PCB does not have a nice print layout. Here is the procedure to design in Express PCB and clean up the patterns so they print nicely.

Preparing Express PCB for First Use:

Express PCB comes with a less then exciting list of parts. So before any project is started head over to Audio logical and grab the additional parts by morsel, ppl, and tangent, and extract them into your Express PCB directory. At this point start the program and get ready to setup the workspace to suit your style. Click View -> Options. In this menu, setup the units for “mm” or “in” depending on how you think, and click “see through the top copper layer” at the bottom. The standard color scheme of red and green is generally used but it is not as pleasing as red and blue.

The Interface:

When a project is first started you will be greeted with a yellow outline. This yellow outline is the dimension of the PCB. Typically after positioning of parts and traces, move them to their final position and then crop the PCB to the correct size. However, in designing a board with a certain size constraint, crop the PCB to the correct size before starting.

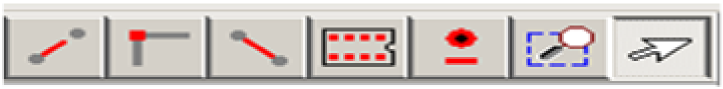


Fig: 5.1.1 show the toolbar in which the each button has the following functions:

Fig 5.1.2: Tool bar necessary for the interface

The select tool: It is fairly obvious what this does. It allows you to move and manipulate parts. When this tool is selected the top toolbar will show buttons to move traces to the top / bottom copper layer, and rotate buttons. The zoom to selection tool: does just that. The place pad: button allows you to place small soldier pads which are useful for board connections or if a part is not in the part library but the part dimensions are available. When this tool is selected the top toolbar will give you a large selection of round holes, square holes and surface mount pads.

The place component: tool allows you to select a component from the top toolbar and then by clicking in the workspace places that component in the orientation chosen using the buttons next to the component list. The components can always be rotated afterwards with the select tool if the orientation is wrong.

The place trace: tool allows you to place a solid trace on the board of varying thicknesses. The top toolbar allows you to select the top or bottom layer to place the trace on.

The Insert Corner in trace: button does exactly what it says. When this tool is selected, clicking on a trace will insert a corner which can be moved to route around components and other traces.

The remove a trace button is not very important since the delete key will achieve the same result.

5.2 Design Considerations:

Before starting a project there are several ways to design a PCB and one must be chosen to suit the project’s needs.

Single sided, or double sided?

When making a PCB you have the option of making a single sided board, or a double sided board. Single sided boards are cheaper to produce and easier to etch, but much harder to design for large projects.

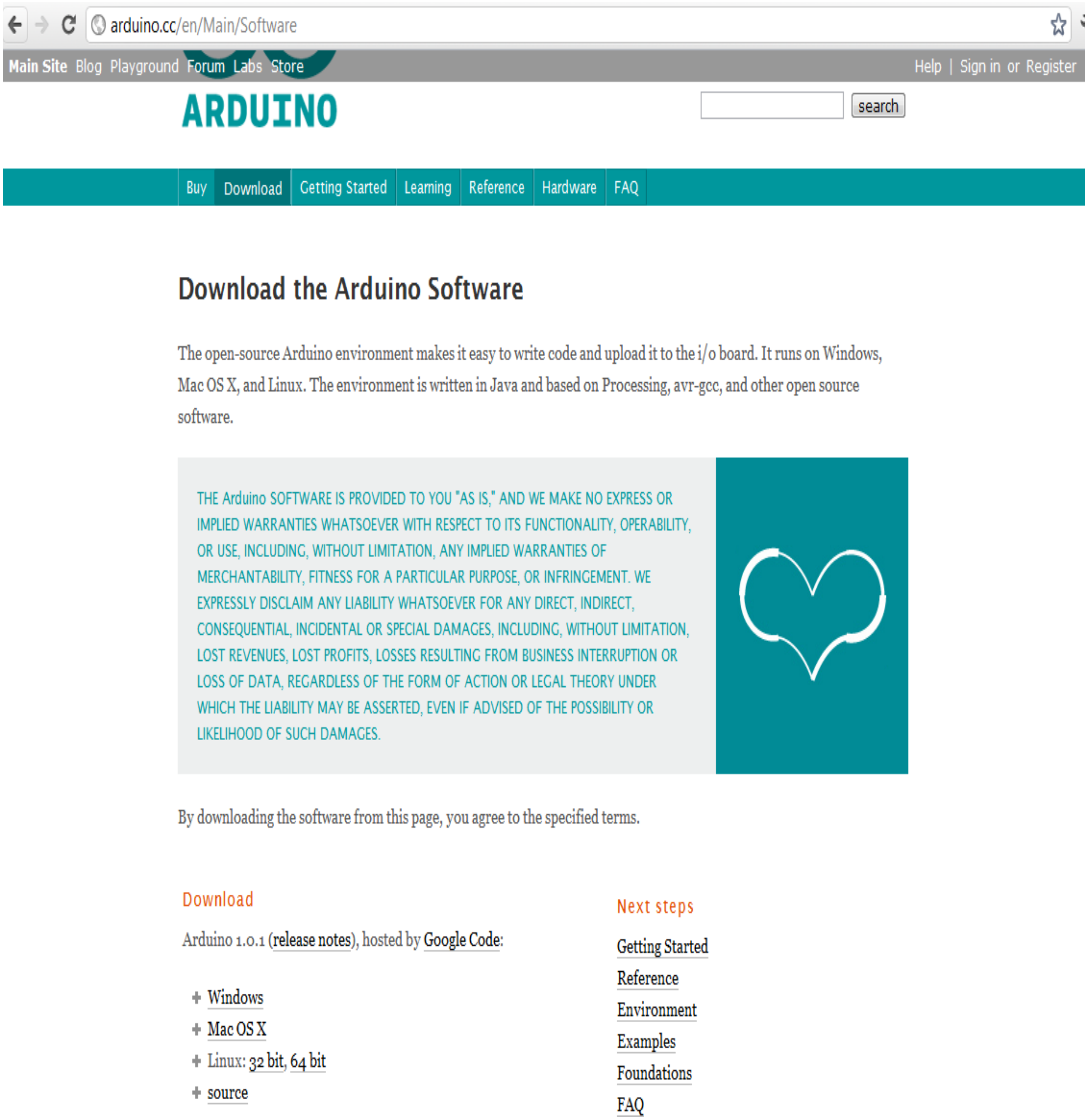
If a lot of parts are being used in a small space it may be difficult to make a single sided board without jumpering over traces with a cable. While there's technically nothing wrong with this, it should be avoided if the signal travelling over the traces is sensitive (e.g. audio signals).

A double sided board is more expensive to produce professionally, more difficult to etch on a DIY board, but makes the layout of components a lot smaller and easier. It should be noted that if a trace is running on the top layer, check with the components to make sure you can get to its pins with a soldering iron. Large capacitors, relays, and similar parts which don't have axial leads can NOT have traces on top unless boards are plated professionally.

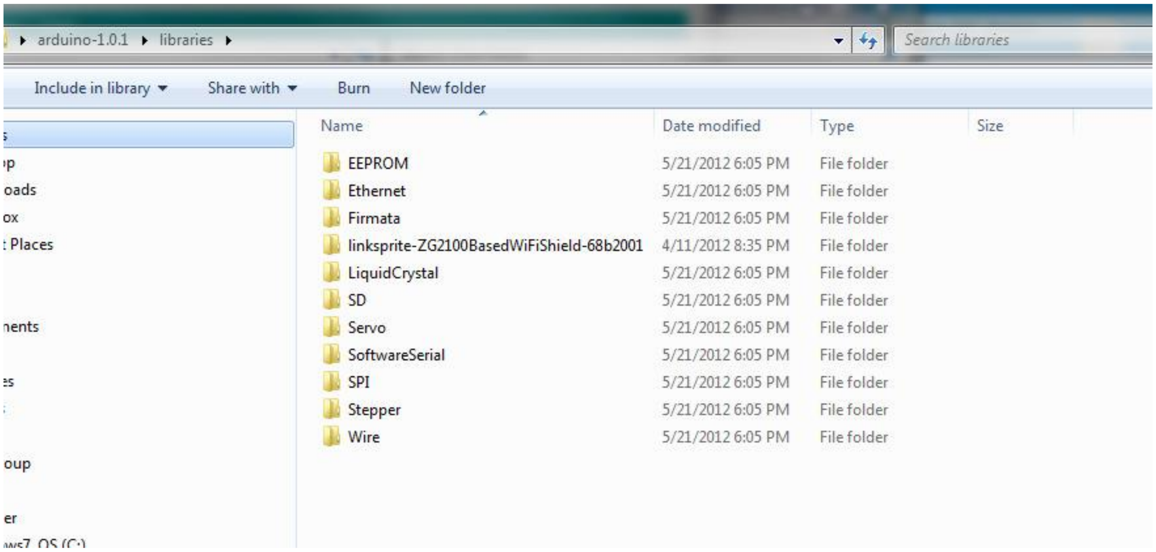
Ground-plane or other special purposes for one side When using a double sided board you must consider which traces should be on what side of the board. Generally, put power traces on the top of the board, jumping only to the bottom if a part cannot be soldiered onto the top plane (like a relay), and vice- versa. Some projects like power supplies or amps can benefit from having a solid plane to use for ground. In power supplies this can reduce noise, and in amps it minimizes the distance between parts and their ground connections, and keeps the ground signal as simple as possible. However, care must be taken with stubborn chips such as the TPA6120 amplifier from TI. The TPA6120 datasheet specifies not to run a ground plane under the pins or signal traces of this chip as the capacitance generated could effect performance negatively.

5.3 AURDINO INSTALLATION AND WORKING

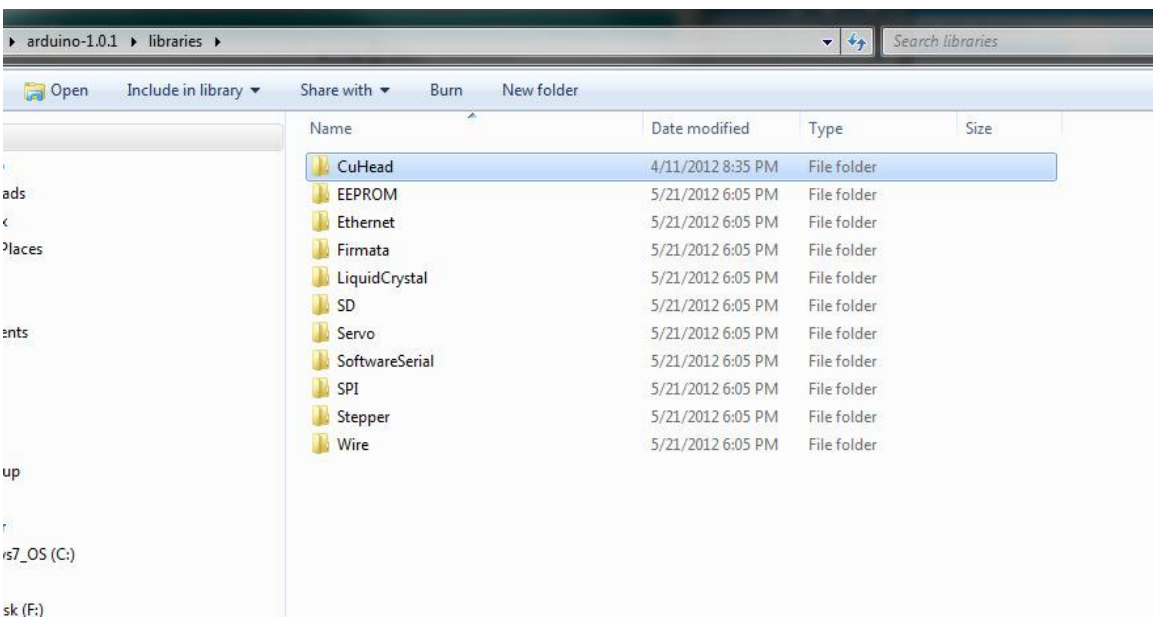
5.3.1Arduino compiling:



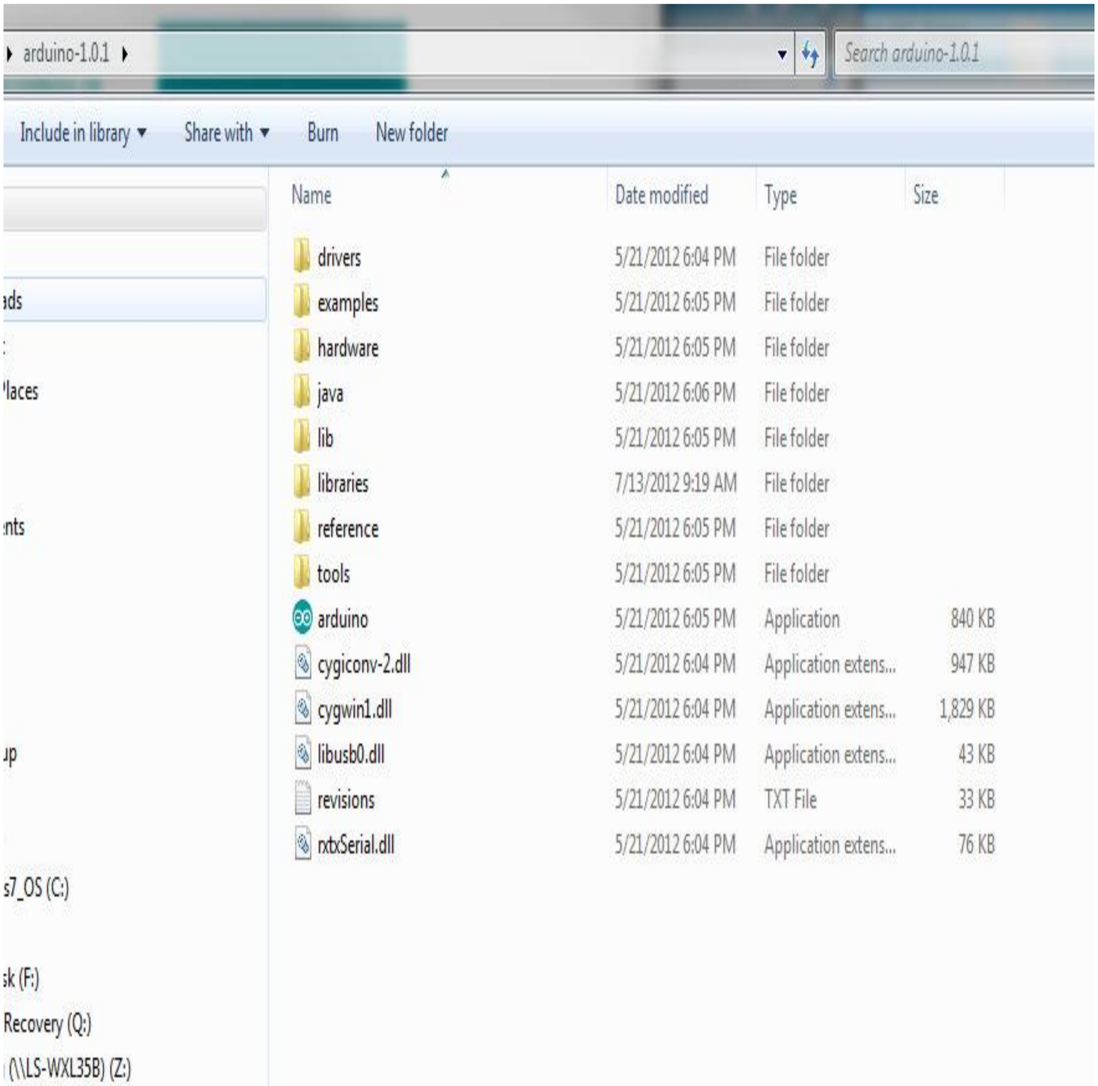
5.3.2 In next step download library



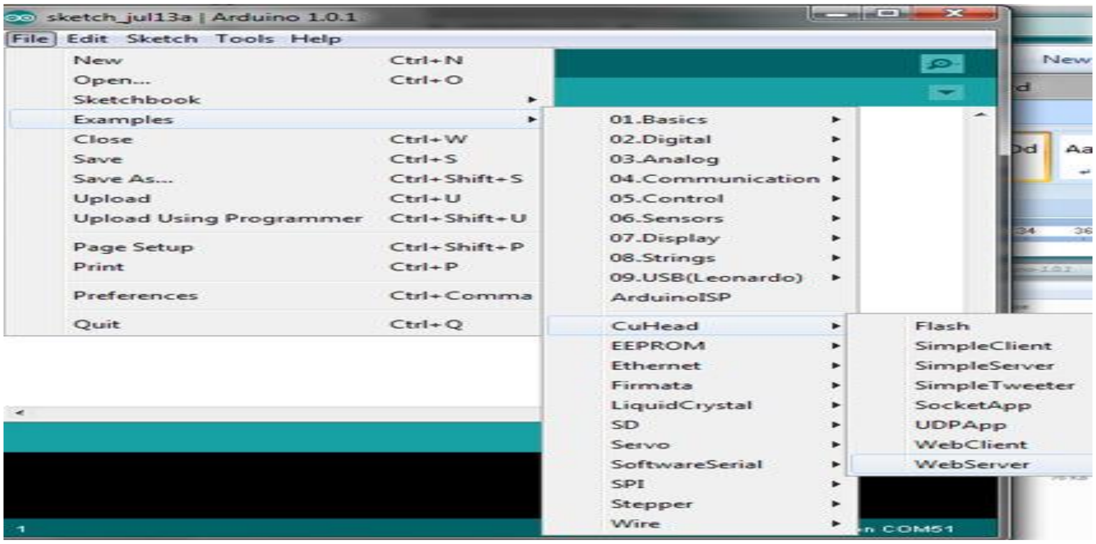
5.3.3 As Arduino doesn't recognize the directory name, please rename it



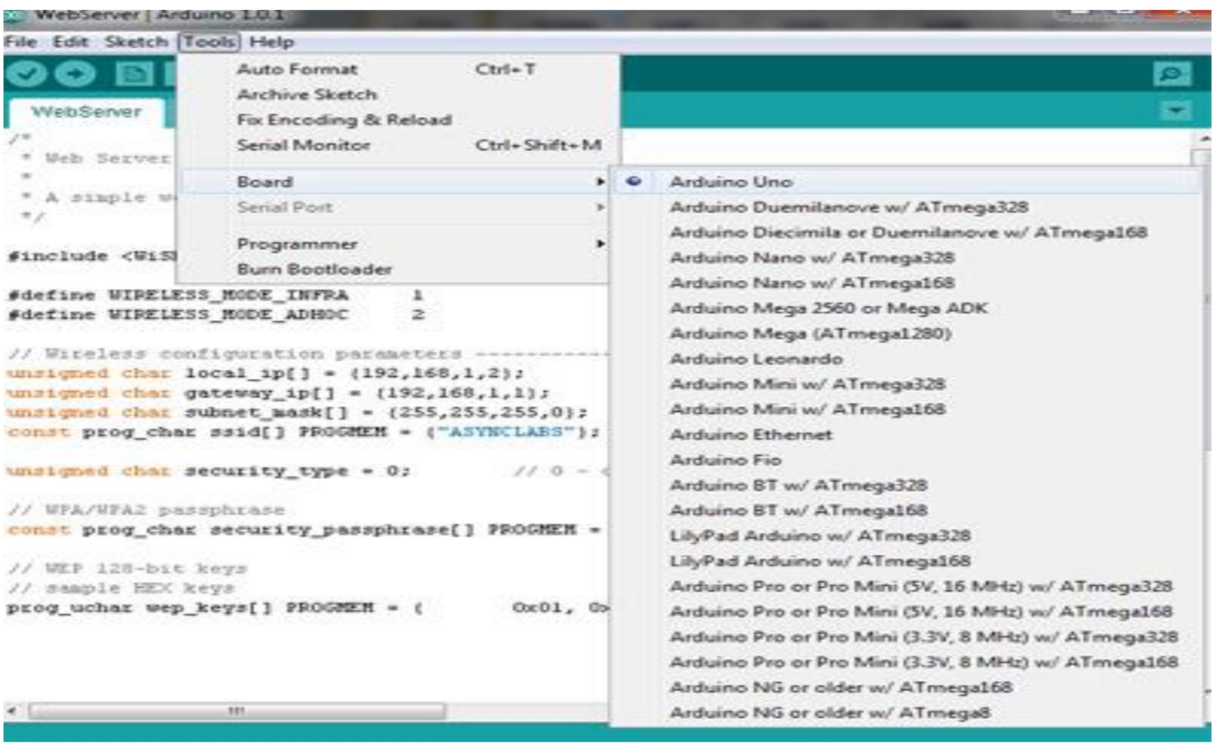
5.3.4 Launch Arduino by double click “arduino” below



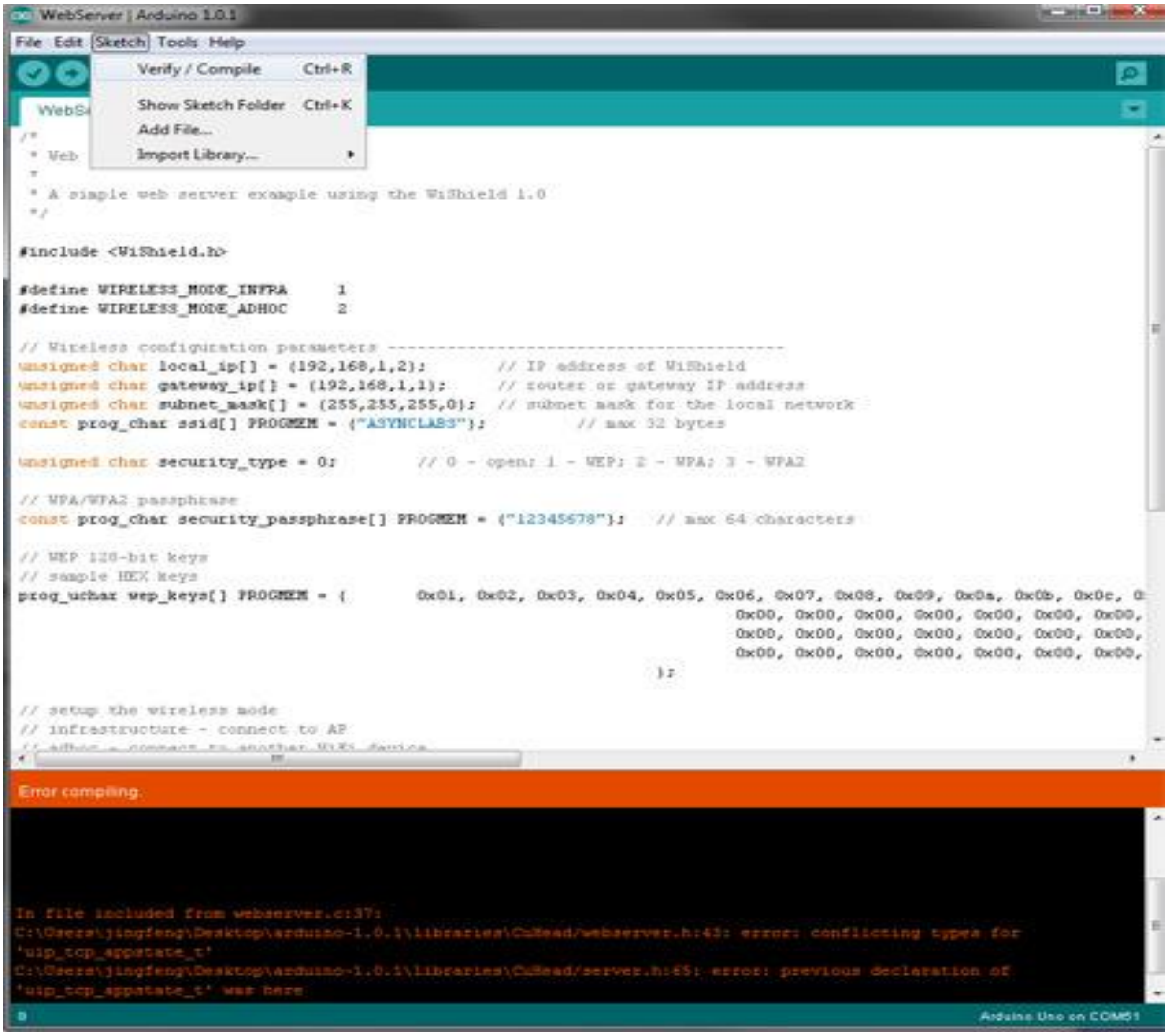
5.3.4.1 Example



Select the target board as "Arduino Uno":



Click Sketch-> Verify/Compile:



6. CODE

6.1 Source code

```
#include <LiquidCrystal.h>
#include <stdio.h>
LiquidCrystal lcd(6, 7, 8, 9, 10, 11);
char gchr='x';
int sti=0;
String inputString = "";        // a string to hold incoming data boolean
stringComplete = false;  // whether the string is complete

int m1a = 12;
int m1b = 13;
int m2a = 11;
int m2b = 10;
int mental = 8;
int buzzer = 9;
void setup()
{
Serial.begin(9600);serialEvent();
// mySerial.begin(9600);
pinMode(m1a, OUTPUT);
pinMode(m1b, OUTPUT);
pinMode(m2a, OUTPUT);
pinMode(m2b, OUTPUT);
pinMode(buzzer,OUTPUT);
pinMode(metal, INPUT);
digitalWrite(m1a, LOW);
digitalWrite(m1b, LOW);
digitalWrite(m2a, LOW);
digitalWrite(m2b, LOW);
digitalWrite(relay1, LOW);
```

```
digitalWrite(relay2, LOW);
digitalWrite(relay3, LOW);
lcd.begin(16, 2);
lcd.print("Bomb detection  Robot");
delay(2000);
serialEvent();

}

void loop()

{


delay(1000);

}

void serialEvent()

{   while (Serial.available())

{

char gchr = (char)Serial.read();
Serial.write(gchr);
if(gchr == 'f')

{gchr='x';lcd.clear();lcd.print("Front  ");
digitalWrite(m1a, HIGH);digitalWrite(m1b, LOW);
digitalWrite(m2a, HIGH);digitalWrite(m2b, LOW);
}  if(gchr == 'b')

{gchr='x';lcd.clear();lcd.print("Back  ");

digitalWrite(m1a, LOW);digitalWrite(m1b, HIGH);
digitalWrite(m2a, LOW);digitalWrite(m2b, HIGH);
}  if(gchr == 'r')

{gchr='x';lcd.clear();lcd.print("Right  ");
```

```
digitalWrite(m2a, HIGH);
digitalWrite(m2b, LOW);
 digitalWrite(m1a, LOW);
digitalWrite(m1b, HIGH);
}  if(gchr == 'l')
{gchr='x';lcd.clear();lcd.print("Left  ");
digitalWrite(m2a, LOW);digitalWrite(m2b, HIGH);
digitalWrite(m1a, HIGH);digitalWrite(m1b, LOW);
}  if(gchr == 's')
{gchr='x';lcd.clear();lcd.print("Stop   ");
digitalWrite(m1a, LOW);digitalWrite(m1b, LOW);
digitalWrite(m2a, LOW);digitalWrite(m2b, LOW);
}
if(digitalRead(metal) == HIGH)  lcd.print(" METAL DETECTED");
digitalWrite(buzzer,HIGH);
}
}
//----------Turn On One-By-One --------- //
else if (IncomingData == "*right")
{
right();
// forward();

}
else if (IncomingData == "*close")
{
ble.println("close"); cutter.write(120); analogWrite(9, LOW);
}
else if (IncomingData == "*open")
{
ble.println("open"); cutter.write(95); analogWrite(9, LOW);
}
IncomingData = "";
}
delay(10);
}
```

7. SYSTEM TEST CASES

7.1 Test Cass on Different Situation and Condition

Test Case ID	Test Case Name	Test Case Description	Expected Result	Pass/Fail	Remarks
A5001	Metal Detection in the Surroundings	Verify that the wireless sensor network detects the Metals nearby and sends the Information to the User.	vibration sensor triggers an alert/notification.	PASS	Metal detection is a critical security measure.
A5002	Bomb Detection in the Surrounding	Ensure that the wireless sensor network triggers an alarm if any Bomb is detected in the Surroundings.	Alarm system activates upon bomb Detection.	PASS	Bomb detection indicate potential criminal Attack.
A5003	Robustnes and Durability Test	It evaluates that it can withstand physical impacts and vibrations without compromising its functionality.	It can be sent to any Difficult Location and find out the Hidden Bombs, Locations of Attackers.	PASS	Finding the Exact co-ordinates of Bombs and Attackers are a crucial part.

8. OUTPUT SCREENS & RESULTS

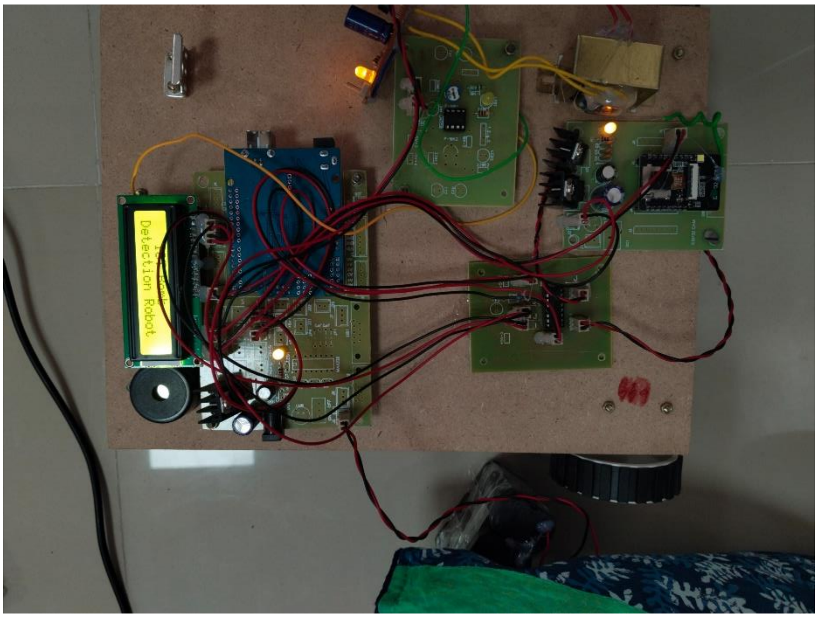


Fig: 8.1 IOT Bomb Detection & Diffusion Kit

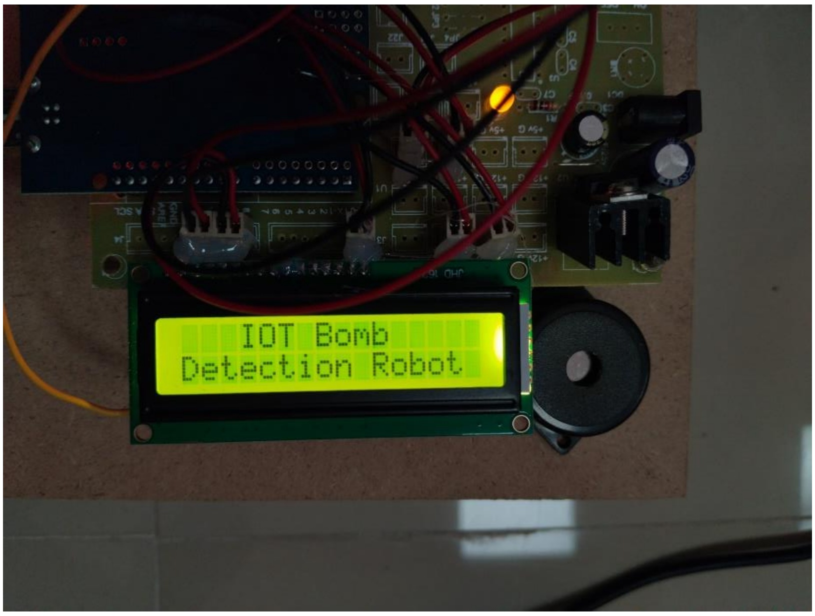


Fig: 8.2 When the system is switched ON, the title gets displayed.

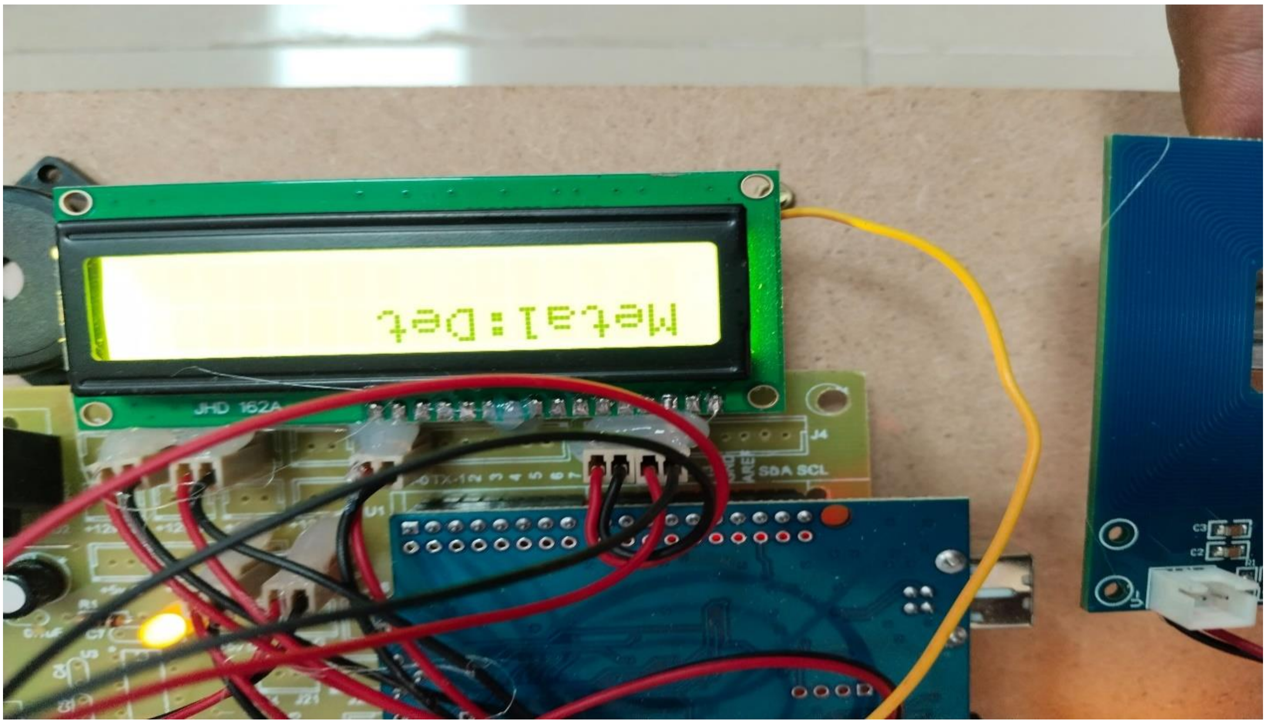


Fig: 8.3 IR sensor in ON condition and the Display condition shows Detected which means Metal detected.

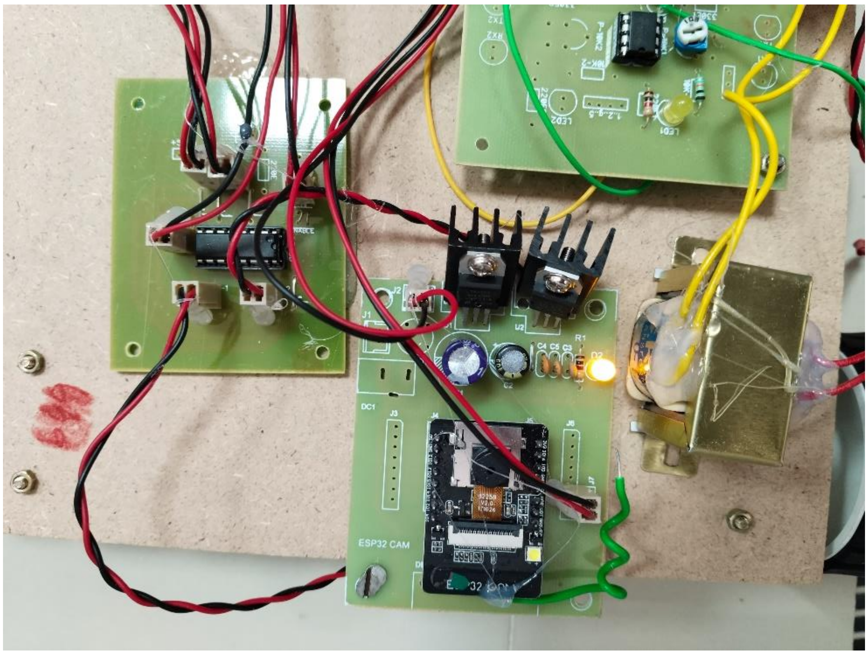


Fig: 8.4 Camera is Connected to the WIFI Module to send us the recordings of the surroundings.

9. ADVANTAGES & APPLICATIONS

Applications:

- The IoT-based remote-operated bomb detection and diffusion robot finds extensive applications across various sectors. In military and defense, it serves as a critical tool for neutralizing explosive threats in conflict zones, protecting soldiers, and minimizing collateral damage.
- Law enforcement agencies utilize it for enhancing security in high-risk environments such as airports, stadiums, and public events, where rapid detection and response to potential threats are paramount. Additionally, it finds applications in industrial settings for inspecting and diffusing hazardous materials or unexploded ordnance, ensuring workplace safety.
- Its deployment in disaster management facilitates timely identification and disposal of explosive hazards in emergency situations, safeguarding civilian lives and infrastructure

Advantages:

- The IoT-based remote-operated bomb detection and diffusion robot offers numerous advantages in enhancing security and safety measures.
- Its remote operation capability reduces the risk to bomb disposal personnel by enabling them to perform tasks from a safe distance, minimizing potential casualties in hazardous environments.
- Real-time data collection and analysis through IoT sensors enhance the robot's detection accuracy, enabling swift identification and assessment of explosive threats.
- Its mobility and maneuverability enable it to navigate challenging terrains and inaccessible areas, ensuring comprehensive surveillance and detection coverage.

Its seamless integration with command centers facilitates

10. CONCLUSION

In this project, the model of robot can be described to build a robot with night vision wireless camera run by android application and the people can learn about developing android application in order to control the robot through wireless application. The robot has reduced the human effort. The robot is designed with high accuracy in movement section. All the objectives of the project were accomplished with high accuracy, camera result was complimentary in that respect is always a way for betterment in any task. More features can be summed in the robot to make it useful. The robot can be made more enhanced by adding features like gas sensors and bomb defuse kit. Future scope of this project is to make the robot move in asymmetrical land, ability to turn off fire as soon as it senses fire, self-balancing to get into small places. we are going to designed a low-cost Microcontroller Based Android controlled Robot. The robot will move forward, backward, left and right direction by following the instructions given from the mobile with audio, video survilleince system using night vision camera. This robot is controlled by Bluetooth module with left, right, forward and backwards postions through android phone. This system can be helpful for various purposes. We have proposed a design of a smart cloud robot to monitor the environmental condition of a remote place. A prototype has been developed and tested in our campus to illustrate the effectiveness of the proposed. Manual control is applied to distantly control the robot from control room. At whatever point signal cautions by distinguishing a metal, a remote camera fixed in robot is used to check whether it is a dangerous object. In the event that so the robotic arm is physically controlled to incapacitate the bomb securely and the attributes of bomb and other information are put away in cloud for future reference. By utilizing this innovation, we can recognize the bomb as right on time as could really be expected and being destroy it effectively with the goal that we can without much of a stretch save the existance. The proposed system of bomb disposing robot will be very useful in the area of security and spying of enemies as well as the areas where human beings cannot reach the robot will do that bomb disposing work .this robot is also remotely operated through internet so there is no harm to human lives.

11. FUTURE ENHANCEMENTS

In future studies this system integrates with GPS get the exact location of fire and gas detection detected. Module it utilizes an interface GPS sensor to transmit area of the leakage over to the IOT login system, here we use IOT to check, get and show the gas leakage caution and location over IOT. Firstly, integrating advanced sensors such as chemical, gas, and thermal imaging sensors can significantly enhance the robot's detection capabilities, enabling it to discern potential threats more accurately. Moreover, the incorporation of AI and machine learning algorithms can elevate detection accuracy by enabling the robot to learn from its environment and distinguish between harmless objects and dangers more effectively over time. Enhancements in autonomous navigation capabilities would empower the robot to navigate complex environments independently, utilizing mapping, obstacle avoidance, and path planning algorithms. Improving the remote control interface with features like live video streaming and interactive maps can enhance user experience and situational awareness for operators. Additionally, robust communication systems, extended battery life, and a modular design for upgradability would ensure the robot's reliability and adaptability in diverse operational scenarios. Integration with other security systems, enhanced manipulation capabilities, and remote monitoring capabilities further augment the robot's effectiveness in bomb detection and diffusion tasks. Compliance with regulations, comprehensive training, and ongoing support are essential aspects to ensure the safe and efficient deployment of the robot in security operations. Through these enhancements, the IoT-based remote-operated robot can become a more formidable tool in enhancing security and safety measures.In the approach, enhancement of the instrument more efficient by exchanging out the IR camera for a thermal night vision camera. User can utilize certain deep learning algorithms to understand humans and issues alerts in place of PIR sensors for person detection. The market is filled with portable robotic arms. It can be changed by the user in order to operate effectively.

12. REFERENCES

[1]. Muhammad Hamza, M Atique-ur-Rehman, Hamza Shafqat, Subhan Bin Khalid, "CGI SCRIPT AND MJPG VIDEO STREAMER BASED SURVEILLANCE ROBOT USING ESP-CAM" in 16th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 8th – 12th January, 2023, IEEE 2023

[2]. Juan G. Parada-Salado, Luis E. Ortega-García, Luis F. Ayala-Ramírez, Francisco J. Pérez Pinal, "A Low-Cost Land Wheeled Autonomous MiniRobot For In-Door Surveillance" In Ieee Latin America Transactions, Vol. 16, No. 5, May 2022

[3]. S M Ashish, Madhurya Manjunath, Ravindra L, Mohammed Nadeem, Neelaja K, "Automated Hybrid Surveillance Robot" In International Journal Of Innovations In Engineering Research And Technology [Ijiert], Issn: 2394-3696; Volume 5, Issue 5, May-2022

[4]. T. Saravanakumar, D. Keerthana, D. Santhiya, J. Sneka, D. Sowmiya, "Surveillance Robot Using ESP-CAM-IoT" in International Journal of Electronics, Electrical and Computational System(IJEECS);ISSN 2348-117X; Volume 7, Issue 3, March 2021.

[5]. Harshitha R, Muhammad Hameem Safwat Hussain, "SURVEILLANCE ROBOT USING ESP-CAM AND IOT" in 2018 IEEE - DOI 10.1109/ICDI3C.2018.00018; International Conference on Design Innovations for 3Cs Compute Communicate Control.

[6]. Gaurav S Bagul, Vikram C Udawant, Kalpana V Kapade and Jayesh M Zope, "IOT BASED SURVEILLANCE ROBOT" in International Journal of Trend in Research and Development, Mar - Apr 2020; Volume 5(2), ISSN: 2394-9333.

[7]. Rutuja. R. Bachhav, Priyanka. S. Bhavsar, Devyani. K. Dambhare, "Spy Robot For Video Surveillance And Metal Detection" In International Journal For Research In Applied Science & Engineering Technology (Ijraset); Issn: 2321-9653; Volume 6 Issue V, May 2019.

[8]. Neha Joisher, Sonaljain, Tejas Malviya, Vaishali Gaikwad (Mohite), "Surveillance System Using Robotic Car" In International Conference On Control, Automation And Systems, Ieee 2018.

[9]. Fatma Salih, Mysoon S.A. Omer, "ESP-CAM As A Video Server" In The International Conference On Computer, Control, Electrical, And Electronics Engineering (Icccee) 2018.

[10]. Sweeta Deshmukh, Priyadarshini, Mamta, Madhura Deshmukh, Dr.Md. Bakhar, "Iot Based Surveillance Robot" In International Journal Of Innovative Research In Computer And Communication Engineering; Vol.5, Special Issue 4, June 2017

[11]. Ghanem Osman Elhaj Abdalla, T. Veeramanikandasamy, "Implementation Of Spy Robot For A Surveillance System Using Internet Protocol Of ESP-CAM" In 2nd Ieee International Conf On Recent Trends In Electronics Information & Communication Technology, May 19-20, 2017.

[12]. Souvik Saha, Arko Singh, Palash Bera, Murshed Nawaz Kamal, Soumya Dutta, Uttam Gorian, Sayak Pramanik, Angshuman Khan, Surajit Sur, "Gps Based Smart Spy Surveillance Robotic System Using ESP-CAM For Security Application And Remote Sensing" In Ieee , 978-1-5386- 33717/17,2016

[13]. Multipurpose Defence Robot G. Jayandhi Asst Professor, ECE,.E.P. Shobana Devi International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 6, 2019.

[14]. Multifunction Robot for Military Application Rohit Botre,Nitin Darade,Pranav Vanve and M.R. Wanjre,International Journal of Trend in Research and Development.

[15]. Wireless multifunctional robot for military applications Proceedings of 2015 RAECS UIET Panjab University Chandigarh 21-22nd December 2015.

[16]. Saurabh Mahawadiwar1, Priti Pai, "Bomb Disposal Robot Using Arduino". Vivekananda Education Society Institute of Technology, Mumbai, India, International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified Vol. 7, Issue 7, July 2018.