IOT INDUSTRY PROTECTION USING ARDUINO

A PROJECT REPORT

Submitted by

DORNADULA LOHITH (211517106028)
KAIPU VINOD KUMAR REDDY (211517106056)
INDURU BHARATH KUMAR REDDY (211517106043)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

PANIMALAR INSTITUTE OF TECHNOLOGY, CHENNAI-600 123

ANNA UNIVERSITY : CHENNAI 600 025

APRIL 2021

ANNA UNIVERSITY : CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"IOT INDUSTRY PROTECTION USING ARDUINO "** is bonafide work of **"DORNADULA LOHITH, KAIPU VINOD KUMAR REDDY, and INDURU BHARATH KUMAR REDDY"** who carried out the project work under my supervision.

SIGNATURE

Dr.M.P.CHITHRA, M.E., Ph.D.
HEAD OF THE DEPARTMENT,
Electronics and Communication Engineering,
Panimalar Insitute of Technology
Poonamalle,
Chennai-600 123.

SIGNATURE

Mr.L. ASHOK KUMAR, M.Tech., (Ph.D)
ASSOCIATE PROFESSOR,
Electronics and Communication Engineering,
Panimalar Institute of Technology
Poonamalle,
Chennai-600 123

Certified that candidates were examined by us in the project work viva voce examination held on_____at Panimalar Institute of Technology, Chennai – 600 123

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We have taken efforts to complete this project. However, it would not have been possible without the kind support and help of many individuals and our institution. We are using this opportunity to express our gratitude to everyone who supported us throughout the course of this final yearproject.

Our sincere thanks to **Our Honorable Founder and Chairman**, **Dr.JEPPIAR M.A., Ph.D**., for educating us in his premier institution.

We would like to express special gratitude and thanks to **Our Beloved Secretary and Correspondent**, **Dr.P.CHINNADURAI, M.A., Ph.D.,** for providing us such a platform to complete our project.

We express warm thanks to **Our Directors Mrs.C.VIJAYA RAJESWARI** and **Mr.C.SAKTHI KUMAR, M.E, M.Phil.,** for their support and valuable guidance.

We are grateful to **Our Principal, Dr.T.JAYANTHY, M.E., Ph.D.,** for sharing her illuminating views on a number of issues related to our project.

We are highly indebted to **Our Head of the Department, Dr. M.P. CHITRA, M.E., Ph.D., Department of Electronics and Communication Engineering,** for her guidance and constant supervision as well as for providing necessary information regarding the project and also for her support in completing the project.

We are thankful to our guide **Mrs. R. PRATHIPA M.E., Departmentof Electronics and Communication Engineering** for her aspiring guidance, invaluably constructive criticism and friendly advice during the project work.

We would also like to thank all **our department staff** who provided us with the facilities being required and conductive conditions for our project. Our thanks and appreciations also go to **our colleagues** in developing the project and **our friends** who have willingly helped us out with their abilities.

From,

Date:19-03-2020

Place : Chennai

Batch Names : Dornadula Lohith

Kaipu Vinod Kumar Reddy

Induru Bharath Kumar Reddy

Batch Number:27

ECE Department

Panimalar Institute of Technology

To

The Head of the Department

ECE

Panimalar Institute of Technology

Sub: Self/In-house Project – Reg.

Respected Madam,

We bring you to the notice that the Project titled **INDUSTRIAL PROTECTION SYSTEM USING IOT** is our final year project work carried out Under the guidance and supervision of **Mrs.V.JEYARAMYA** by ourselves in the college campus as the In-house Project without referring any external environment or centers.

Signature of the

Project Guide Project Coordinator HOD/ECE

ABSTRACT:

Today, smart grid, smart homes, smart water networks, intelligent transportation, are infrastructure systems that connect our world more than we ever thought possible. The common vision of such systems is usually associated with one single concept, the Internet of Things (IoT), where through the use of sensors, the entire physical infrastructure is closely coupled with information and communication technologies; where intelligent monitoring and management can be achieved via the usage of networked embedded devices. These devices will connect to internet to share different types of data. We have proposed an Industrial Monitoring System using WIFI module and sensing based applications for internet of things. By detecting the values of sensors it can easily find out the Temperature, humidity, and gas present in the industrial area. And also it controls the power with abnormalities. So that critical situation can be avoided and preventive measures are successfully implemented. It is the most effective and most economical means of equipment safety monitoring. So it has very good social prospect

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

LDR - LIGHT DEPENDENT RESISTOR
IoT - INTERNET OF THINGS
LCD - LIQUID CRYSTAL DISPLAY

CHAPTER-1

INTRODUCTION

CHAPTER-1

INTRODUCTION

Now- a – days, the industrial monitoring field requires more manual power to monitor and control the industrial parameters such as temperature, humidity, gas etc. this is the most upcoming issues in the industrial sectors. if the parameters are not monitored and control properly, it leads to a harmful situation. Most of the industries are facing those kinds of situation because of some manual mistakes. To overcome manual mistakes we are using industrial automation with internet of things. WIRELESS SENSOR NETWORKS (WSN) has been employed to collect data about physical phenomena in various applications such as habitat monitoring, and ocean monitoring, and surveillance[1]-[3]. As an emerging technology brought about rapid advances in modern wireless telecommunication, Internet of Things (IoT) has attracted a lot of attention and is expected to bring benefits to numerous application areas including industrial WSN systems, and healthcare systems manufacturing[4],[5].WSN systems are well-suited for longterm industrial. There is a growing interest in using IoT technologies in various industries. A number of industrial IoT projects have been conducted in areas such as agriculture, food processing industry, environmental monitoring, security surveillance, and others. There has been much research and various attempts to apply new IoT technologyto industrial areas. IoT provides a promising opportunity to build powerful industrial systems and applications by leveraging the growing ubiquity of radio frequency identification (RFID) and wireless, mobile and sensor devices. A wide range of industrial IoT applications have been developed and deployed in recent years. In an effort to understand the development of IoT in industries.

CHAPTER- 2

LITERATURE SURVEY

CHAPTER-2

LITERATURE SURVEY

The concept of the internet of things was introduced by the members of the radio frequency identification development community in 1999. This concept is very popular because of the growth of mobile devices, embedded and real time communication, cloud computing and data analytics. The internet of things is a network of physical objects embedded with electronics, software and sensors having the ability to collect data from the world around us and share data across the internet.The term internet of things refers to the general idea of things, especially everyday objects that are readable, locatable, recognizable, addressable and controllable through the internet, irrespective of the communication means such as wired or wireless LAN, WAN or any mean. The things or objects of real world can be People, Location (object), Time of information (object) or Condition. These things can easily get integrated in the virtual world enabling anytime, anywhere connectivity. Now, the system architecture also includes different types of elements which are shown as follows:

1. Sensors: The sensors are the devices that are useful for gathering the information at the point of activity. This information is actually captured by appliances, wearable devices, some specific device mounted controls, and so on. Thus these are the elements of IoT that sense any type of information depending upon purpose of the application.

2. Communication: The information sensed by various sensors needs to be transmitted to a cloud based service for subsequent processing. This requires Wi-Fi, WAN, LAN or some internet communication network. Along with this communication networks the support for other capabilities such as Bluetooth, short range communication method or GPS for locating the positions is often required for effective communication. The communication network is typically based on the M2M technique. The M2M stands for machine to machine communication system

in which at one end sensors are attached to sense any desired information and at other end the devices that deliver the information to the actual user are attached.

3. Cloud based capture and consolidation: Gathered data is transmitted to a cloud based service. At this cloud, the useful information is provided for the end user. Some information processing is also done at this level.

4. Delivery of Information: This is actually the last step of delivery. This is the point at which end user, commercial user or industrial user comes into picture. The goal of delivery of information is to provide information in as simple and transparent manner as possible. The delivery of information typically needs the execution of well-designed and optimized user interface across multiple platforms.

CHAPTER- 3

Internet of Things-IOT

CHAPTER-3

Internet of Things-IOT

3.1 INTRODUCTION:

The IOT concept was coined by a member of the Radio Frequency Identification (RFID) development community in 1999, and it has recently become more relevant to the practical world largely because of the growth of mobile devices, embedded and ubiquitous communication, cloud computing and data analytics.[12] Imagine a world where billions of objects can sense, communicate and share information, all interconnected over public or private Internet Protocol (IP) networks. These interconnected objects have data regularly collected, analyzed and used to initiate action, providing a wealth of intelligence for planning, management and decision making. This is the world of the Internet of Things (IOT). [12] Internet of things common definition is defining as: Internet of things (IOT) is a network of physical objects. The internet is not only a network of computers, but it has evolved into a network of device of all type and sizes , vehicles, smart phones, home appliances, toys, cameras, medical instruments and industrial systems, animals, people, buildings, all connected ,all communicating & sharing information based on stipulated protocols in order to achieve smart reorganizations, positioning, tracing, safe & control & even personal real time online monitoring , online upgrade, process control & administration[1,2]. We define IOT into three categories as below: Internet of things is an internet of three things: (1). People to people, (2) People to machine /things, (3) Things /machine to things /machine, Interacting through internet. Internet of Things Vision: Internet of Things (IoT) is a concept and a paradigm that considers pervasive presence in the environment of a variety of things/objects that through wireless and wired connections and unique addressing schemes are able to interact with each other and cooperate with other things/objects to create new applications/services and reach common goals. In this context the research and development challenges to create a smart world are enormous. A world where the real, digital and the virtual are converging

to create smart environments that make energy, transport, cities and many other areas more intelligent. [1, 2]

Internet of Things is refer to the general idea of things, especially everyday objects, that are readable, recognisable, locatable, addressable through information sensing device and/or controllable via the Internet, irrespective of the communication means (whether via RFID, wireless LAN, wide area networks, or other means). Everyday objects include not only the electronic devices we encounter or the products of higher technological development such as vehicles and equipment but things that we do not ordinarily think of as electronic at all - such as food , clothing ,chair, animal, tree, water etc. [1,2] Internet of Things is a new revolution of the Internet. Objects make themselves recognizable and they obtain intelligence by making or enabling context related decisions thanks to the fact that they can communicate information about themselves. They can access information that has been aggregated by other things, or they can be components of complex services. This transformation is concomitant with the emergence of cloud computing capabilities and the transition of the Internet towards IPv6 with an almost unlimited addressing capacity. [1, 2] The goal of the Internet of Things is to enable things to be connected anytime, anyplace, with anything and anyone ideally using any path/network and any service.

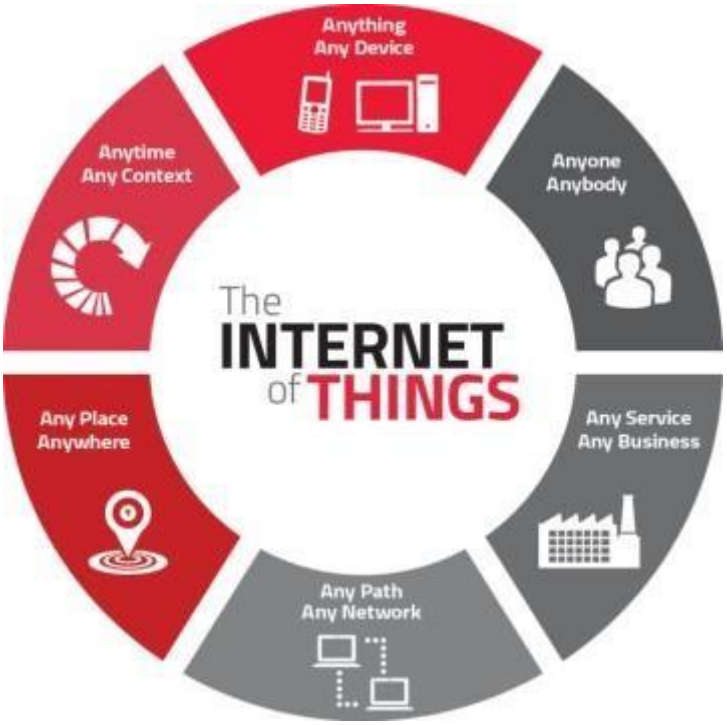


Fig no 3.1.1 Internet of things

3.2 ENABLING TECHNOLOGIES FOR IOT

Internet of things (IoT) is a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies. With the Internet of Things the communication is extended via Internet to all the things that surround us. The Internet of Things is much more than machine to machine communication, wireless sensor networks, sensor networks , 2G/3G/4G,GSM,GPRS,RFID, WI-FI, GPS, microcontroller, microprocessor etc. These are considered as being the enabling technologies that make "Internet of Things" applications possible. Enabling technologies for the Internet of Things are considered in [1] and can be grouped into three categories: (1) technologies that enable "things" to acquire contextual information, (2) technologies that enable "things" to process contextual information, and (3) technologies to improve security and privacy. The first two categories can be jointly understood as functional building blocks required building "intelligence" into "things", which are indeed the features that differentiate the IoT from the usual Internet. The third category is not a functional but rather a de facto

requirement, without which the penetration of the IoT would be severely reduced. [2] The Internet of Things is not a single technology, but it is a mixture of different hardware & software technology. The Internet of Things provides solutions based on the integration of information technology, which refers to hardware and software used to store, retrieve, and process data and communications technology which includes electronic systems used for communication between individuals or groups. There is a heterogeneous mix of communication technologies, which need to be adapted in order to address the needs of IoT applications such as energy efficiency, speed, security, and reliability. In this context, it is possible that the level of diversity will be scaled to a number a manageable connectivity technologies that address the needs of the IoT applications, are adopted by the market, they have already proved to be serviceable, supported by a strong technology alliance. Examples of standards in these categories include wired and wireless technologies like Ethernet, WI-FI, Bluetooth, ZigBee, GSM, and GPRS. [1, 2] .

3.3 CHARACTERISTICS

The fundamental characteristics of the IoT are as follows [2, 6]: Interconnectivity: With regard to the IoT, anything can be interconnected with the global information and communication infrastructure. Things-related services: The IoT is capable of providing thing-related services within the constraints of things, such as privacy protection and semantic consistency between physical things and their associated virtual things. In order to provide thing-related services within the constraints of things, both the technologies in physical world and information world will change. Heterogeneity: The devices in the IoT are heterogeneous as based on different hardware platforms and networks. They can interact with other devices or service platforms through different networks. Dynamic changes: The state of devices change dynamically, e.g., sleeping and waking up, connected and/or disconnected as well as the context of devices including location and speed. Moreover, the number of devices can change dynamically. Enormous scale: The number of devices that need to be managed and that communicate with each other will be at least an order of magnitude larger than the devices connected to the current Internet. Even more critical will be the management of

the data generated and their interpretation for application purposes. This relates to semantics of data, as well as efficient data handling. Safety: As we gain benefits from the IoT, we must not forget about safety. As both the creators and recipients of the IoT, we must design for safety. This includes the safety of our personal data and the safety of our physical well-being. Securing the endpoints, the networks, and the data moving across all of it means creating a security paradigm that will scale. Connectivity: Connectivity enables network accessibility and compatibility. Accessibility is getting on a network while compatibility provides the common ability to consume and produce data.

CHAPTER-4

METHODOLOGY

CHAPTER-4

METHODOLOGY

4.1 Existing system:

1. Implementation of ZigBee-GSM based home security monitoring and remote control systemArbab Waheed Ahmad , Naeem Jan, Saeed Iqbal and Chankil Lee proposed Implementation of ZigBeeGSM based Home Security Monitoring and Remote Control system. Home security and control is one among the essential needs of mankind from youth. But today it's to be updated with the rapidly changing technology to make sure vast coverage, remote, reliability, and real time operation. Deploying wireless technologies for security and control in home automation systems offers attractive benefits along side user friendly interfaceINDUSTRY BASED SECURITY SYSTEM USING GSM AND ARDUINO(Shubham Raut, Avinash Gaikwad , Mudaliyar Raghurajan , Pratiksha Patil)This paper describes the design of a simple low cost GSM based security monitoring system using GSM technique. The security monitoring system using GSM undergoes analog to digital converter and with GSM Modem the message is send to mobile. ADC is used because Arduino works with digital inputs. GSM modem can be used at the transmitter side, the user sends an SMS to the GSM modem using.

4.2 PROPOSED SYSTEM:

The IOT industry protection system using Arduino is a system designed to protect industries from losses due to accidents using Internet of things. Gas leakages may lead to fires leading to huge industrial losses, also instant fire detection is needed in case of furnace blasts or other conditions. Also low lighting in industries may create improper work conditions increasing the probability of accidents. The system makes use of Arduino to achieve this functionality. The system makes use of temperature sensing along with light and gas sensing to detect fire, gas leakage as well as low lighting to avoid any industrial accidents and prevent losses. The system consists of light, gas and temperature sensors interfaced with Arduino and LCD screen. The sensor data is constantly scanned to record values and check for fire, gas leakage or low light and then this data is transmitted online. The Wi-Fi module is used to achieve internet functionality. The IOT server then displays this information online, to achieve the desired output.

BLOCK DIAGRAM:

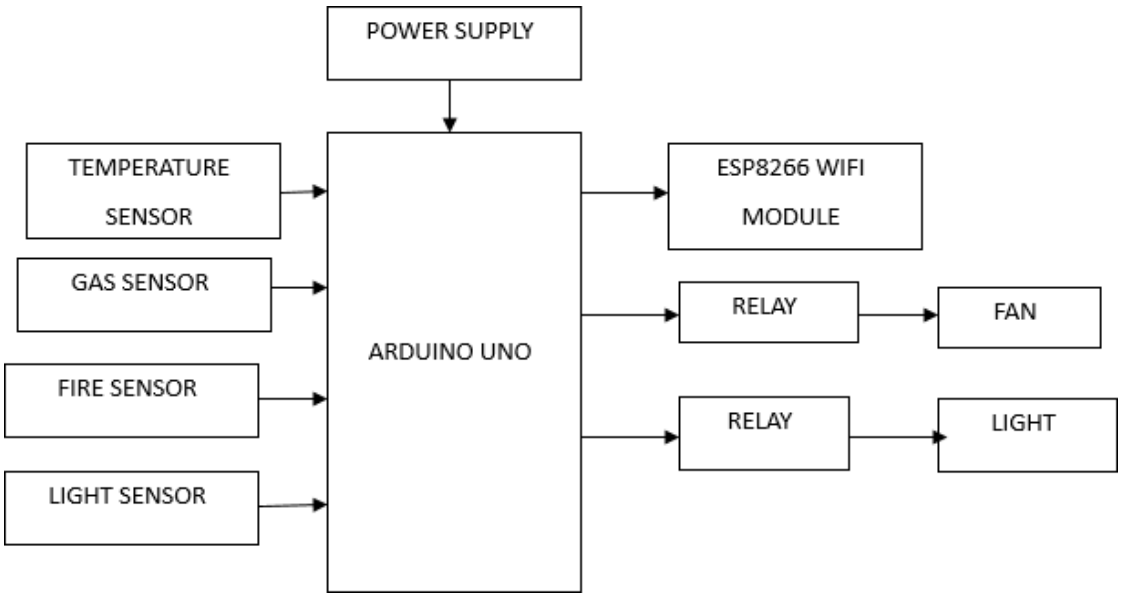


Fig no 4.2.1 BLOCK DIAGRAM

4.3 HARDWARE REQUIREMENTS:

- Arduino Uno
- LCD Display
- Wifi Module
- LDR Sensor
- Gas Sensor
- Temperature Sensor
- RELAY
- POWER SUPPLY

CHAPTER-5

HARDWARE DESCRIPTION

CHAPTER-5

HARDWARE DESCRIPTION

5.1 Arduino Uno Board

Arduino is a single-board microcontroller meant to make the application more accessible which are interactive objects and its surroundings. The hardware features with an open-source hardware board designed around an 8-bit Atmel AVR microcontroller or a 32-bit Atmel ARM. Current models consists a USB interface, 6 analog input pins and 14 digital I/O pins that allows the user to attach various extension boards.

The Arduino Uno board is a microcontroller based on the ATmega328. It has 14 digital input/output pins in which 6 can be used as PWM outputs, a 16 MHz ceramic resonator, an ICSP header, a USB connection, 6 analog inputs, a power jack and a reset button. This contains all the required support needed for microcontroller. In order to get started, they are simply connected to a computer with a USB cable or with a AC-to-DC adapter or battery. Arduino Uno Board varies from all other boards and they will not use the FTDI USB-to-serial driver chip in them. It is featured by the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

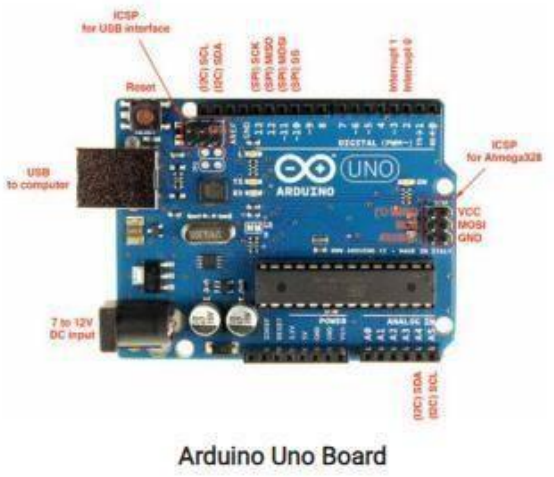


Fig no 5.1.1 ARDUINO UNO BOARD

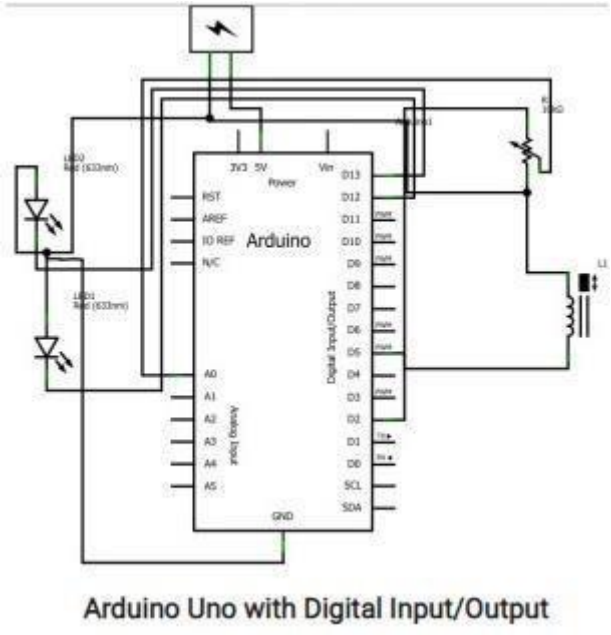


Fig no 5.1.2 Arduino uno with digital input and output

There are various types of Arduino boards in which many of them were third-party compatible versions. The most official versions available are the Arduino Uno R3 and the Arduino Nano V3. Both of these run a 16MHz Atmel ATmega328P 8-bit microcontroller with 32KB of flash RAM 14 digital I/O and six analogue I/O and the 32KB will not sound like as if running Windows. Arduino projects can be stand-alone or they can communicate with software on running on a computer. For e.g. Flash, Processing, Max/MSP). The board is clocked by a 16 MHz ceramic resonator and has a USB connection for power and communication. You can easily add micro SD/SD card storage for bigger tasks.

5.1.1 Features of the Arduino Uno Board:

- It is an easy USB interface. This allows interface with USB as this is like a serial device.
- The chip on the board plugs straight into your USB port and supports on your computer as a virtual serial port. The benefit of this setup is that serial communication is an extremely easy protocol which is time-tested and USB makes connection with modern computers and makes it comfortable.
- It is easy-to-find the microcontroller brain which is the ATmega328 chip. It has more number of hardware features like timers, external and internal interrupts, PWM pins and multiple sleep modes.
- It is an open source design and there is an advantage of being open source is that it has a large community of people using and troubleshooting it. This makes it easy to help in debugging projects.
- It is a 16 MHz clock which is fast enough for most applications and does not speeds up the microcontroller.
- It is very convenient to manage power inside it and it had a feature of built-in voltage regulation. This can also be powered directly off a USB port without any external power. You can connect an external power source of upto 12v and this regulates it to both 5v and 3.3v.
- 13 digital pins and 6 analog pins. This sort of pins allows you to connect hardware to your Arduino Uno board externally. These pins are used as a key for extending the computing capability of the Arduino Uno into the real world. Simply plug your electronic devices and sensors into the sockets that correspond to each of these pins and you are good to go.
- This has an ICSP connector for bypassing the USB port and interfacing the Arduino directly as a serial device. This port is necessary to re-bootload your chip if it corrupts and can no longer used to your computer.
- It has a 32 KB of flash memory for storing your code.
- An on-board LED is attached to digital pin 13 to make fast the debugging of code and to make the debug process easy.

- Finally, it has a button to reset the program on the chip.

Arduino was created in the year 2005 by two Italian engineers David Cuartielles and Massimo Banzi with the goal of keeping in mind about students to make them learn how to program the Arduino uno microcontroller and improve their skills about electronics and use it in the real world.

Arduino uno microcontroller can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing).

Fig no 5.1.1.1 ATmega168-328 ARDUINO PIN MAPPING

5.1.2 Programming:

- The Arduino integrated development environment (IDE) is a cross-platform application written in Java, and is derived from the IDE for the Processing programming language and the Wiring projects
- The Arduino Uno board can be programmed with the Arduino software.
- Select “Arduino Uno from the Tools > Board menu (according to the microcontroller on your board).

- The ATmega328 on the Arduino Uno comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol.
- You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header.
- The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available .

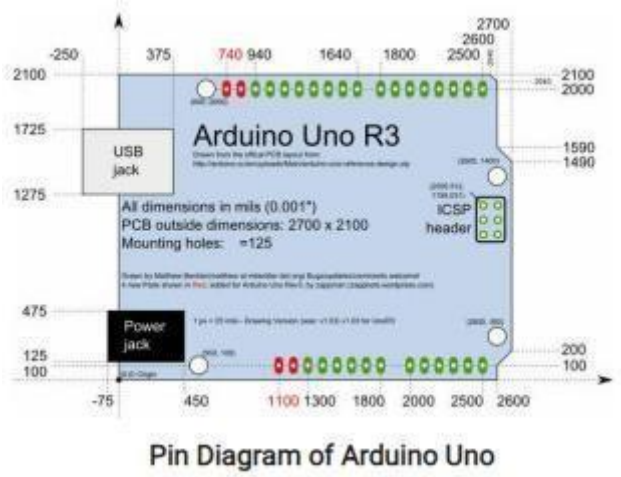


Fig no 5.1.2.1 ARDUINO PIN DIAGRAM

The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWBline to ground, making it easier to put into DFU mode.

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).

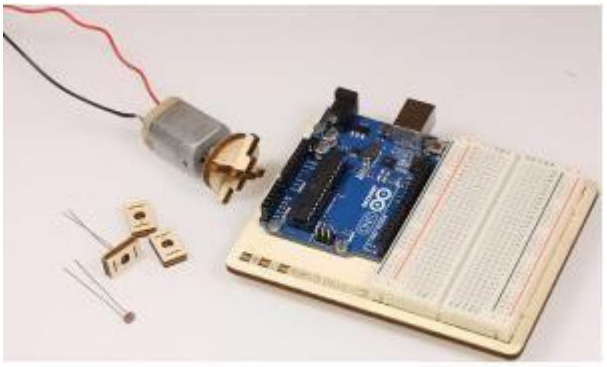


Fig no 5.1.2.2 ARDUINO UNO STARTER KIT

Microcontroller	ATmega28
Operating Voltage	5V
Input Voltage(Recommended)	7-12V
Input Voltage(Limits)	6-20V
Digital I/O Pins	14(of which 6 provide PWM output)
Analog Input Pins	6
DC Current Per I/O pin	40Ma
DC Current For3.3V Pin	50Ma
Flash Memory	32KB(ATmega328)of which 0.5KB used by
SRAM	2KB(ATmega328)
EEPROM	1KB(ATmega328)
Clock Speed	16 MHz

5.2 ESP8266 WiFi

5.2.1 Introduction

ESP8266 is Wi-Fi enabled system on chip (SoC) module developed by Espressif system. It is mostly used for development of IoT (Internet of Things) embedded applications.

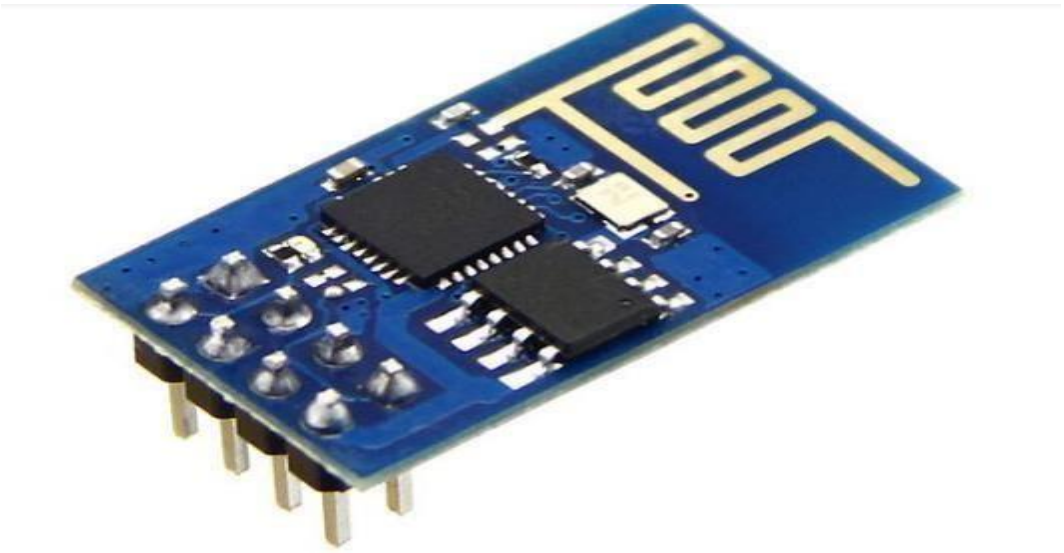


Fig no 5.2.1.1 ESP8266-01 WiFi Module

ESP8266 comes with capabilities of

- 2.4 GHz Wi-Fi (802.11 b/g/n, supporting WPA/WPA2),
- general-purpose input/output (16 GPIO),
- Inter-Integrated Circuit (I^2C) serial communication protocol,
- analog-to-digital conversion (10-bit ADC)
- Serial Peripheral Interface (SPI) serial communication protocol,
- I^2S (Inter-IC Sound) interfaces with DMA(Direct Memory Access) (sharing pins with GPIO),
- UART (on dedicated pins, plus a transmit-only UART can be enabled on GPIO2), and
- pulse-width modulation (PWM).

It employs a 32-bit RISC CPU based on the Tensilica Xtensa L106 running at 80 MHz (or overclocked to 160 MHz). It has a 64 KB boot ROM, 64 KB instruction RAM and 96 KB data RAM. External flash memory can be accessed through SPI.

ESP8266 module is low cost standalone wireless transceiver that can be used for end-point IoT developments.

To communicate with the ESP8266 module, microcontroller needs to use set of AT commands. Microcontroller communicates with ESP8266-01 module using UART having specified Baud rate.

There are many third-party manufacturers that produce different modules based on this chip. So, the module comes with different pin availability options like,

- ESP-01 comes with 8 pins (2 GPIO pins) – PCB trace antenna. (shown inabove figure)
- ESP-02 comes with 8 pins, (3 GPIO pins) – U-FL antenna connector.
- ESP-03 comes with 14 pins, (7 GPIO pins) – Ceramic antenna.
- ESP-04 comes with 14 pins, (7 GPIO pins) – No ant.

For example, below figure shows ESP-01 module pins

5.2.2 ESP8266-01 Module Pin Description

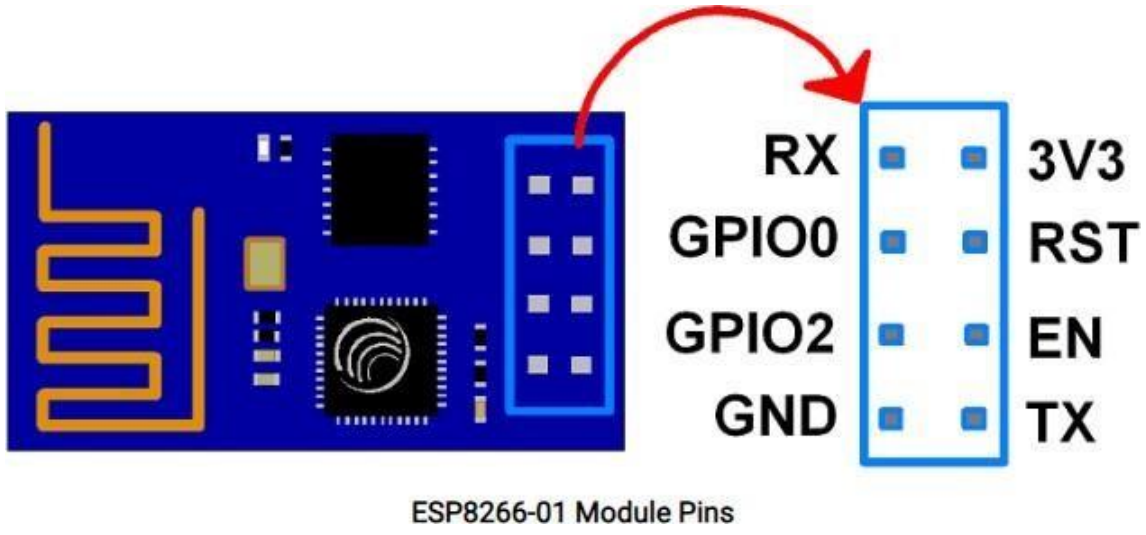


Fig no. 5.2.2.1 ESP8266-01 PIN DESCRIPTION

3V3: - 3.3 V Power Pin.

GND: - Ground Pin.

RST: - Active Low Reset Pin.

EN: - Active High Enable Pin.

TX: - Serial Transmit Pin of UART.

RX: - Serial Receive Pin of UART.

GPIO0 & GPIO2: - General Purpose I/O Pins. These pins decide what mode (boot or normal) the module starts up in. It also decides whether the TX/RX pins are used for Programming the module or for serial I/O purpose.

To program the module using UART, Connect GPIO0 to ground and GPIO2 to VCC or leave it open. To use UART for normal Serial I/O leave both the pins open (neither VCC nor Ground).

5.3 LDR – Light Dependent Resistors Circuit and Working Principle

5.3.1 INTRODUCTION

The controlling of lights and home appliances is generally operated and maintained manually on several occasions. But the process of appliances controlling may cause wastage of power due to the carelessness of human beings or unusual circumstances. To overcome this problem we can use the light-dependent resistor circuit for controlling the loads based on the intensity of light. An LDR or a photoresistor is a device that is made up of high resistance semiconductor material. This article gives an overview of what is LDR, light-dependent resistor circuit and its working.

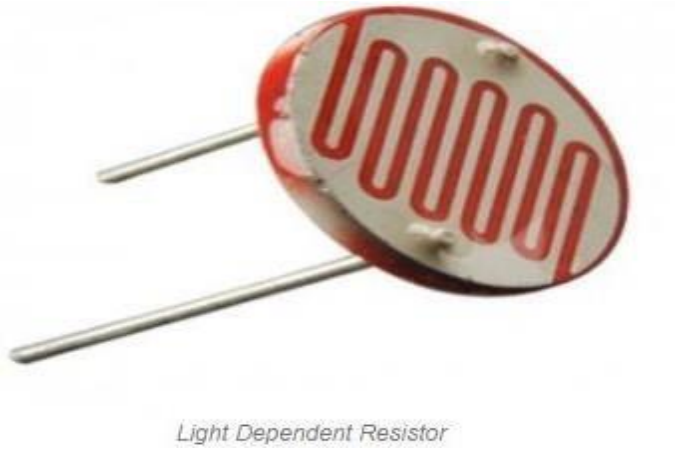


Fig no 5.3.1.1 LDR

5.3.2 Construction of an LDR

The construction of an LDR includes a light-sensitive material that is placed on an insulating substrate like as ceramic. The material is placed in a zigzag shape in order to get the required power rating and resistance. The area of zigzag separates the metal placed areas into two regions.

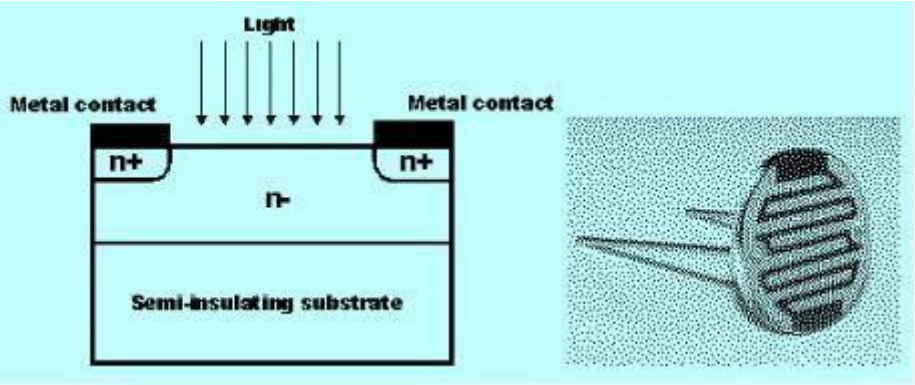


Fig no 5.3.2.1 CONSTRUCTION OF LDR

Where the Ohmic contacts are made either on the sides of the area. The resistances of the contacts must be as less as possible to make sure that the resistance, mainly varies due to the light effect only. The use of lead & cadmium materials is avoided as they are injurious to the environment.

5.3.3 Working Principle of Light Dependent Resistor

The working principle of an LDR is photoconductivity, that is nothing but an optical phenomenon. When the light is absorbed by the material then the conductivity of the material reduces. When the light falls on the LDR, then the electrons in the valence band of the material are eager to the conduction band. But, the photons in the incident light must have energy superior than the bandgap of the material to make the electrons jump from one band to another band (valance to conduction).

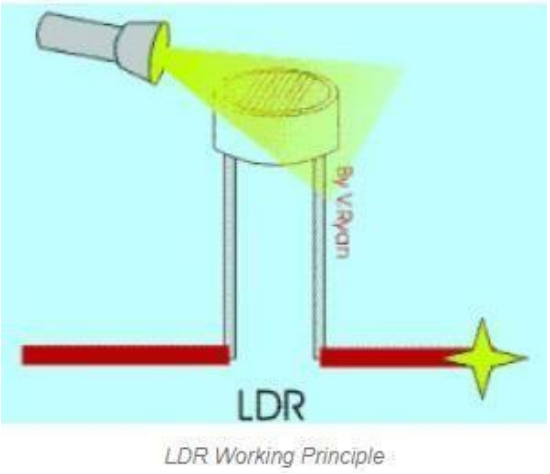
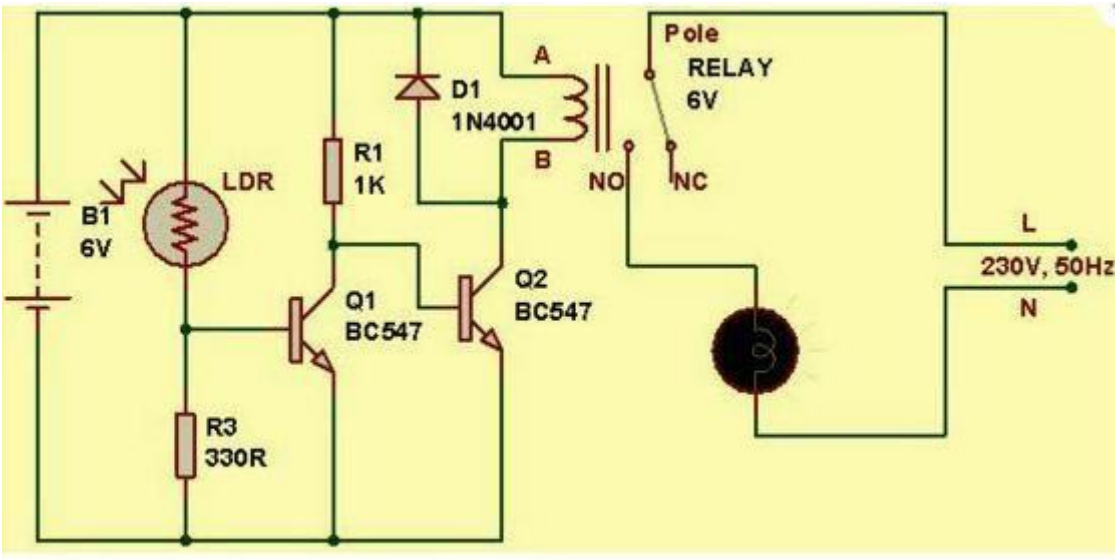


Fig no 5.3.3.1 working of LDR

Hence, when light having ample energy, more electrons are excited to the conduction band which grades in a large number of charge carriers. When the effect of this process and the flow of the current starts flowing more, the resistance of the device decreases.

5.3.4 Light Dependent Resistor Circuit

The circuit of LDR is an electronic circuit built with LDR, relay, Darlington pair, diode, & resistors shown in the below circuit diagram. A voltage supply is given to the load The required DC voltage of the LDR circuit is supplied from a bridge rectifier circuit or a battery. This circuit changes the AC supply into a DC. The bridge rectifier circuit uses a step-down transformer to step-down the voltage from 230v into 12v. The diodes are connected in the form of a bridge used to alter the AC voltage into DC. The voltage regulator is used to change the 12v DC-6v DC, and then, this DC voltage is supplied to the entire circuit. A 230v AC supply for both the bridge rectifier and the load is to be kept continuously for continuous operation of the light sensor circuit.



Light Dependent Resistor Circuit Diagram

Fig no 5.3.4.1 LDR circuit

In the morning time, this sensor has a low resistance around 100Ω. Thus, the power supply flows through the LDR & ground through the variable resistor and resistor as shown in the above light sensor circuit. This is due to the resistance offered by the light-dependent resistor in the daytime or when the light falls on the LDR, then it is less compared to the resistance of the remaining part of the sensor circuit. We are alert of

the principle of current, that the flow of current always flows in the path of low resistance.

Therefore, the relay coil does not get sufficient supplies to get strengthened. Hence, the light is switched off in the daylight. In the same way, during the night time, the resistance of the LDR increases to a high value (20MΩ). Thus, due to the high resistance of the resistor, the flow of current is low or almost zero. Now, the flow of current through the low-resistance lane such that it increases the base voltage of the Darlington pair to reach more than 1.4v. As the Darlington pair transistor is triggered, the relay coil acquires enough supply to get energized, and hence, the light switches in the night time.

5.3.5 Applications of LDR

Light-dependent resistors are simple and low-cost devices. These devices are used where there is a need to sense the presence and absence of light is necessary. These resistors are used as light sensors and the applications of LDR mainly include alarm clocks, street lights, light intensity meters, burglar alarm circuits. For a better understanding of this concept, here we have explained one project namely; power conserving of intensity controlled street lights using LDR.

5.3.6 Power Conserving of Intensity Controlled Street Lights using LDR

Nowadays, lightening-up of highways is done through HID lamps but, the energy consumption of these lights is high as well as there is no particular mechanism to turn on/off the lights from sunrise to sunset. To overcome this problem, here is an alternative method using LEDs i.e. power conserving of intensity controlled street lights using LDR.

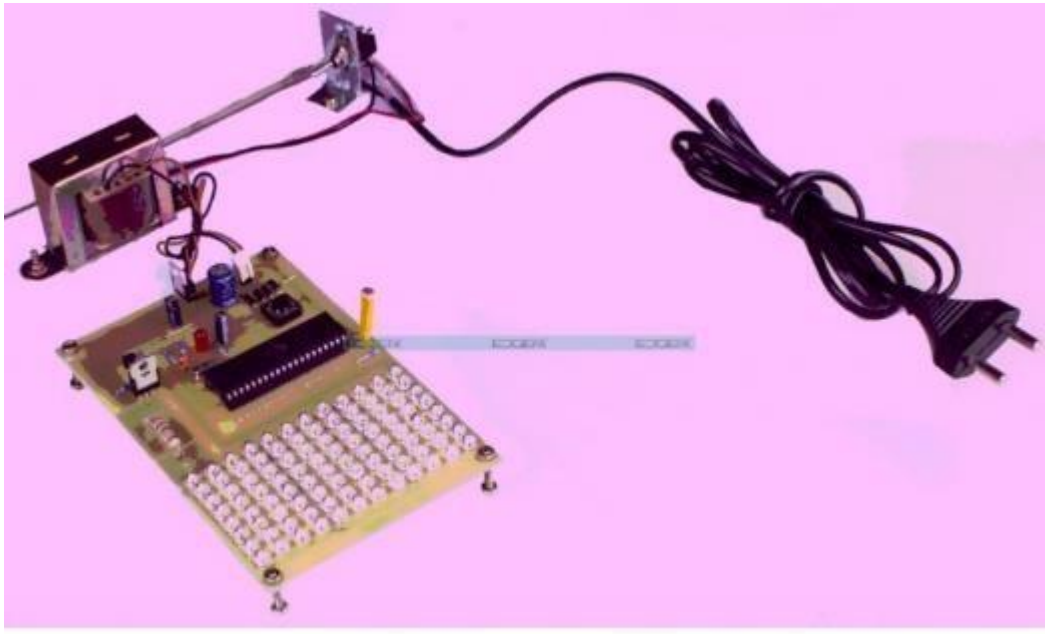


Fig no 5.3.6.1 Intensity Controlled Street Light Using LDR

The proposed system is built to overcome the drawback of the present day of HID lamps. The proposed system determines the usage of light-emitting diodes as a light source and its adjustable intensity control according to the requirement. These lights consume less power, as well as the life span of these lights, which is more compared to conventional HID lamps.

The most important feature of this project is, the light intensity can be controlled according to the necessity during night time, which is not possible in HID lamps. An LDR is used to detect the light and the resistance of the LDR drastically reduces according to the light in the day time, which forms as an i/p signal to the controller.

A bunch of LEDs is used to make a street light and the microcontroller using in the project is preprogrammed with instructions that control the light intensity based on the Pulse width modulation signals generated. The light intensity is kept high during the night time, and as the traffic on the roads tends to fall in peak hours; the intensity also falls gradually till morning. Finally, the LED lights get turned off in the morning at 6 am and continues again at 6 pm in the evening. This process will continue. Furthermore, this project can be developed by mixing it with a solar panel, which changes the

intensity of the solar to the equivalent voltage and solar energy is used to feed the lights on highways.

Thus, this is all about the working principle of the light-dependent resistor and LDR circuit diagram and its applications. We hope that you have got a better concept through this article. Furthermore, any queries regarding this article or electrical and electronics projects

5.4 GAS SENSORS:

A Typical human nose has 400 types of scent receptors enabling us to smell about 1 trillion different odours. But still many of us do not have the capacity to identify the type or concentration of gas present in our atmosphere. This is where Sensors comes in, there are many types of sensors to measure different parameters and a **Gas sensor** is one which comes handy in applications where we have to detect the variation in the concentration of toxic gases in order to maintain the system safe and avoid/cautionany unexpected threats. There are various gas sensors to detect gases like oxygen, Carbon Dioxide, Nitrogen, methane etc. They can also be commonly found in devices that are used to detect the leakage of the harmful gases, monitor the air quality in industries and offices etc.

In this article, we will learn more about **gas sensors**, their construction, types, working and how they can be used to measure the required type and concentration of Gas inour atmosphere. There are many types of Gas sensors but the **MQ type gas sensors** are commonly used and widely popular so will focus more on these types of sensors for this article.

5.4.1 Introduction to Gas Sensor

A **gas sensor** is a device which detects the presence or concentration of gases in the atmosphere. Based on the concentration of the gas the sensor produces a corresponding potential difference by changing the resistance of the material inside the sensor, which can be measured as output voltage. Based on this voltage value the type and concentration of the gas can be estimated.

Fig no 5.4.1.1 gas sensor

The type of gas the sensor could detect depends on the **sensing material** presentinside the sensor. Normally these sensors are available as modules with comparators as shown above. These comparators can be set for a particular threshold value of gas concentration. When the concentration of the gas exceeds this threshold the digital pin goes high. The analog pin can be used to measure the concentration of the gas.

5.4.2 Different Types of Gas sensors

Gas sensors are typically classified into various types based on the type of the sensing element it is built with. Below is the classification of the various types of gas sensors based on the sensing element that are generally used in various applications:

- Metal Oxide based gas Sensor.
- Optical gas Sensor.
- Electrochemical gas Sensor.
- Capacitance-based gas Sensor.
- Calorimetric gas Sensor.
- Acoustic based gas Sensor.

5.4.3 Gas Sensor Construction

Of all the above-listed types, the most commonly used gas sensor is the Metal oxide semiconductor based gas sensor. All Gas sensors will consist of a sensing element which comprises of the following parts.

1. Gas sensing layer
2. Heater Coil
3. Electrode line
4. Tubular ceramic
5. Electrode

The below image illustrates the parts present in a metal oxide gas sensor

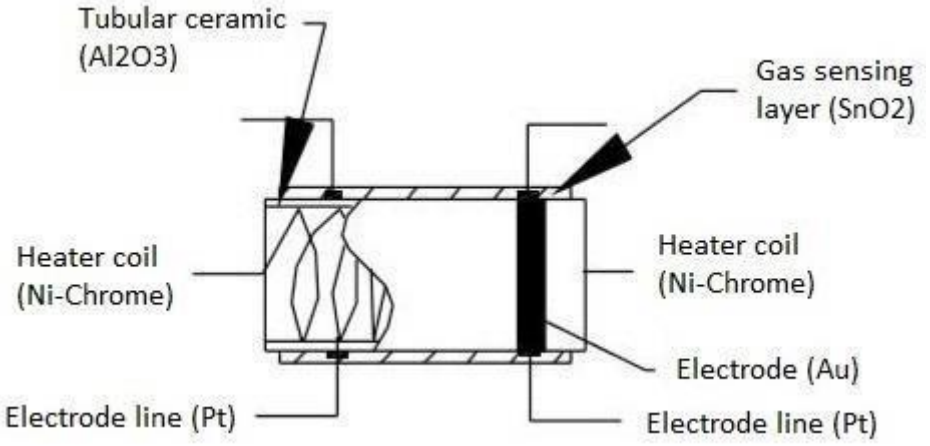


Fig no 5.4.3.1 Metal Oxide Gas Sensor

The purpose of each of these elements is as below:

Gas sensing layer: It is the main component in the sensor which can be used to sense the variation in the concentration of the gases and generate the change in electrical resistance. The gas sensing layer is basically a chemiresistor which changes its resistance value based on theThe concentration of particular gas in the environment. Here the sensing element is made up of a Tin Dioxide (SnO2) which is, in general, has excess electrons (donor element). So whenever toxic gases are being detected the resistance of the element changes and the current flown through it varies which represents the change in concentration of the gases.

Heater coil: The purpose of the heater coil is to burn-in the sensing element so that the sensitivity and efficiency of the sensing element increases. It is made of Nickel-Chromium which has a high melting point so that it can stay heated up without getting melted.

Electrode line: As the sensing element produces a very small current when the gas is detected it is more important to maintain the efficiency of carrying those small currents. So Platinum wires come into play where it helps in moving the electrons efficiently.

Electrode: It is a junction where the output of the sensing layer is connected to the Electrode line. So that the output current can flow to the required terminal. An electrode here is made of Gold (Au –Aurum) which is a very good conductor.

Tubular ceramic: In between the Heater coil and Gas sensing layer, the tubular ceramic exists which is made of Aluminum oxide (Al2O3). As it has high melting point, it helps in maintaining the burn-in (preheating) of the sensing layer which gives the high sensitivity for the sensing layer to get efficient output current.

Mesh over the sensing element: In order to protect the sensing elements and the setup, a metal mesh is used over it, which is also used to avoid/hold the dust particles entering into the mesh and prevent damaging the gas sensing layer from corrosive particles.

5.4.4 Gas Sensor Working

The ability of a Gas sensor to detect gases depends on the **chemiresister** to conduct current. The most commonly used chemiresistor is Tin Dioxide (SnO2) which is an n-type semiconductor that has free electrons (also called as donor). Normally the atmosphere will contain more oxygen than combustible gases. The oxygen particles attract the free electrons present in SnO2 which pushes them to the surface of the SnO2. As there are **no free electrons** available output current will be zero. The below gif shown the oxygen molecules (blue color) attracting the free electrons (black color) inside the SnO2 and preventing it from having free electrons to conduct current.

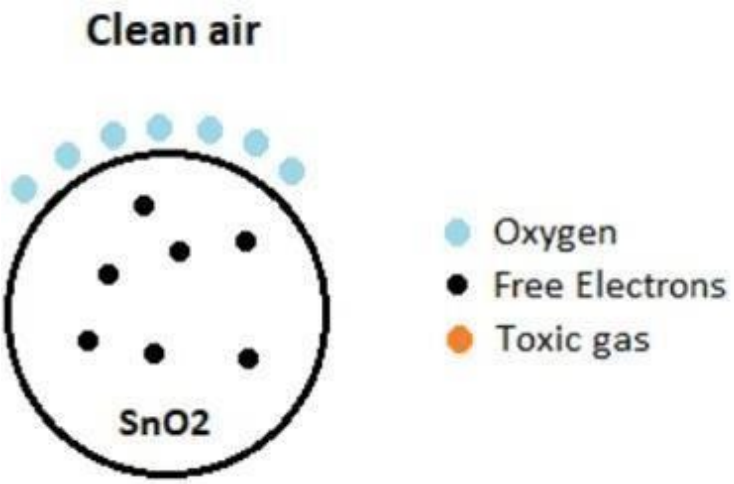


Fig no 5.4.4.1 Working of Gas Sensor

When the sensor is placed in the toxic or combustible gases environment, thisreducing gas (orange color) reacts with the adsorbed oxygen particles and breaks the chemical bond between oxygen and free electrons thus **releasing the free electrons**. As the free electrons are back to its initial position they can now conduct current, this conduction will be proportional the amount of free electrons available in SnO2, if the gas is highly toxic more free electrons will be available.

5.4.5 How to use a Gas sensor?

A basic gas sensor has 6 terminals in which 4 terminals (A, A, B, B) acts input or output and the remaining 2 terminals (H, H) are for heating the coil. Of these 4 terminals, 2 terminals from each side can be used as either input or output (these terminals are reversible as shown in the circuit diagram) and vice versa.

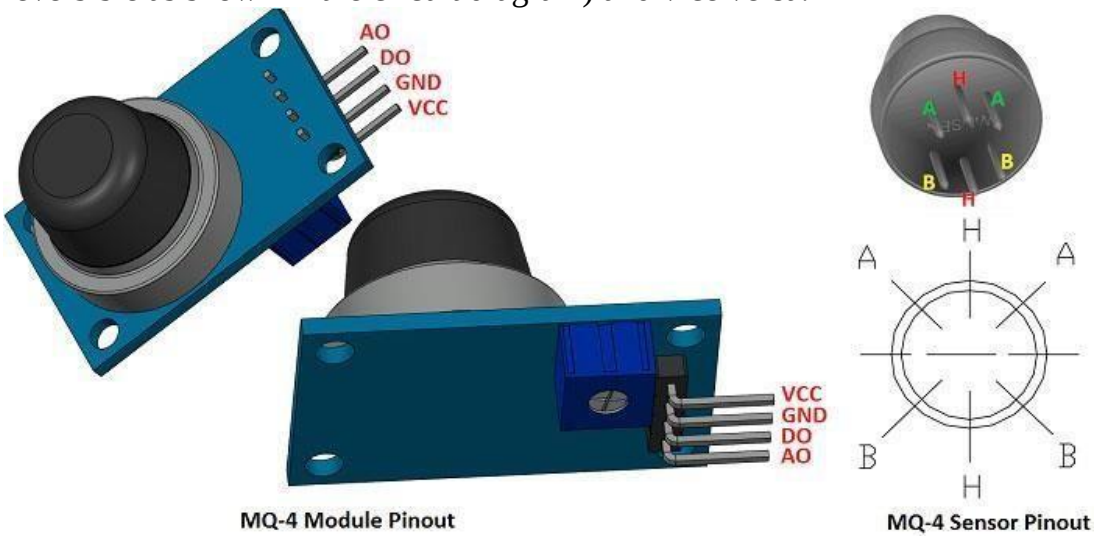


Fig no 5.4.5.1 Module Pinout And Sensor Pinout

These sensors are normally available as modules (shown right), these modules consist of the gas sensor and a comparator IC. Now let's see the pin description of the gas sensor module which we will generally use with an Arduino. The gas sensor module basically consists of 4 terminals

- **Vcc** – Power supply
- **GND** – Power supply
- **Digital output** – This pin gives an output either in logical high or logical low (0 or 1) that means it displays the presence of any toxic or combustible gases near the sensor.
- **Analog output** – This pin gives an output continuous in voltage which varies based on the concentration of gas that is applied to the gas sensor.

5.5 LM35 Temperature Sensor

5.5.1 Introduction

5.5.1.1LM35 Temperature Sensor

- LM35 is a temperature measuring device having an analog output voltage proportional to the temperature.
- It provides output voltage in Centigrade (Celsius). It does not require any external calibration circuitry.
- The sensitivity of LM35 is 10 mV/degree Celsius. As temperature increases, output voltage also increases.

  E.g. 250 mV means 25°C.

- It is a 3-terminal sensor used to measure surrounding temperature ranging from -55 °C to 150 °C.
- LM35 gives temperature output which is more precise than thermistor output.

5.5.2 Pin Description

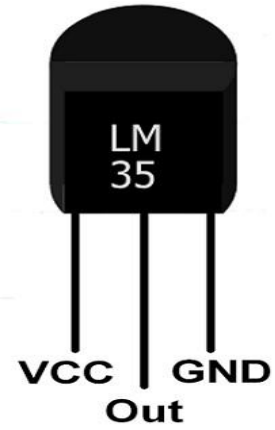


Fig no 5.5.2.1 Pin Description Of Temperature Sensor

VCC: Supply Voltage (4V – 30V)

Out: It gives analog output voltage which is proportional to the temperature (in degree Celsius).

GND: Ground

5.5.3 Application Setup

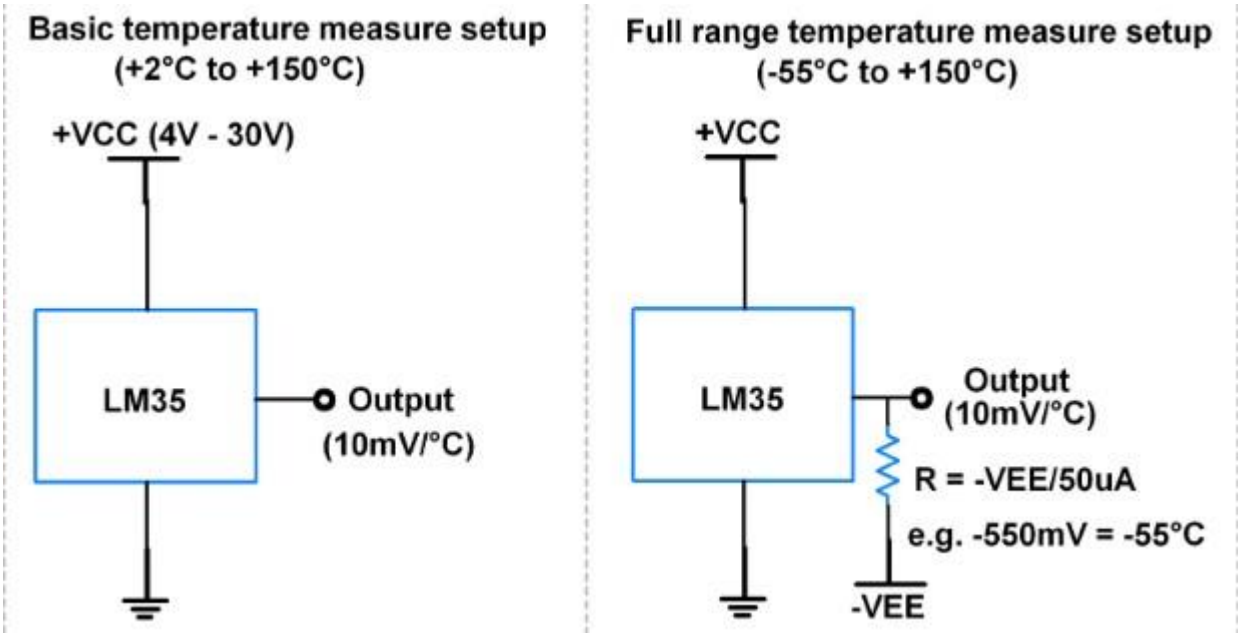


Fig no 5.5.3.1 Application setup of Temperature Sensor

5.5.4 Temperature Sensors – Types, Working & Operation

Temperature is the most often-measured environmental quantity. This might be expected since most physical, electronic, chemical, mechanical, and biological systems are affected by temperature. Certain chemical reactions, biological processes, and even electronic circuits perform best within limited temperature ranges. Temperature is one of the most commonly measured variables and it is therefore not surprising that there are many ways of sensing it. Temperature sensing can be done either through direct contact with the heating source or remotely, without direct contact with the source using radiated energy instead. There are a wide variety of temperature sensors on the market today, including Thermocouples, Resistance Temperature Detectors (RTDs), Thermistors, Infrared, and Semiconductor Sensors.

5.5.4.1 Types of Temperature Sensors

Thermocouple: It is a type of temperature sensor, which is made by joining two dissimilar metals at one end. The joined end is referred to as the HOT JUNCTION. The other end of these dissimilar metals is referred to as the COLD END or COLD JUNCTION. The cold junction is formed at the last point of thermocouple material. If there is a difference in temperature between the hot junction and cold junction, a small voltage is created. This voltage is referred to as an EMF (electro-motive force) and can be measured and in turn used to indicate temperature

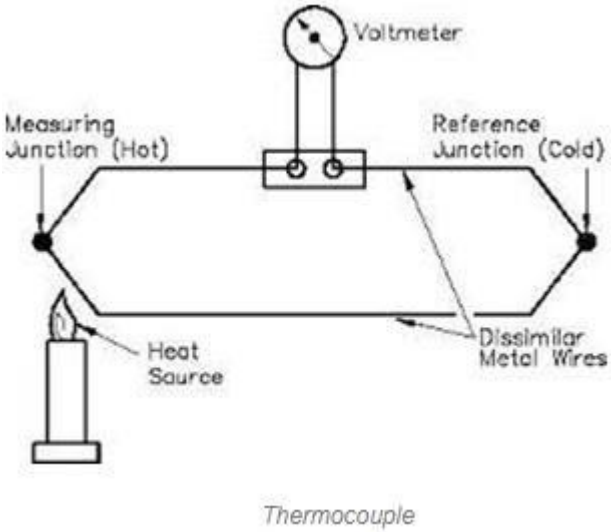


Fig no 5.5.4.1.1 Thermocouple

The RTD is a temperature-sensing device whose resistance changes with temperature. Typically built from platinum, though devices made from nickel or copper are not uncommon,

RTDs can take many different shapes like wire wound, thin film. To measure the resistance across an RTD, apply a constant current, measure the resulting voltage, and determine the RTD resistance. RTDs exhibit fairly linear resistance to temperature curves over their operating regions and any nonlinearity is highly predictable and repeatable. The PT100 RTD evaluation board uses surface mount RTD to measure temperature. An external 2, 3, or 4-wire PT100 can also be associated with measure temperature in remote areas. The RTDs are biased using a constant current source. To reduce self-heat due to power dissipation, the current magnitude is moderately low. The circuit shown in the figure is the constant current source uses a reference voltage, one amplifier, and a PNP transistor.

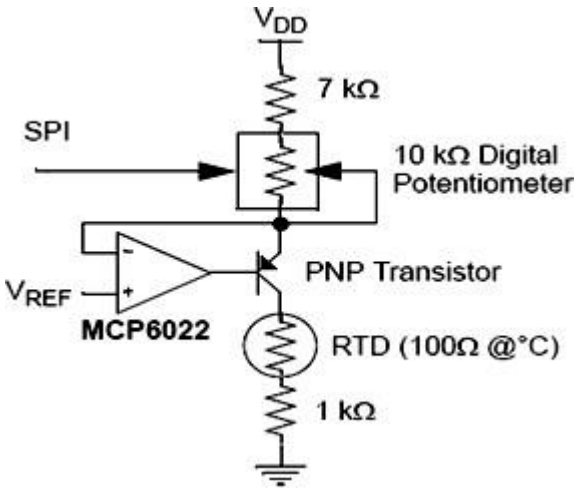


Fig no 5.5.4.1.2 RTD

Thermistors: Similar to the RTD, the thermistor is a temperature-sensing device whose resistance changes with temperature. Thermistors, however, are made from semiconductor materials. Resistance is determined in the same manner as the RTD, but thermistors exhibit a highly nonlinear resistance vs. temperature curve. Thus, in the thermistors operating range, we can see a large resistance change for a very small temperature change. This makes for a highly sensitive device, ideal for set-point applications.

Semiconductor sensors: They are classified into different types like Voltage output, Current output, Digital output, Resistance output silicon, and Diode temperature sensors. Modern semiconductor temperature sensors offer high accuracy and high linearity over an operating range of about 55°C to +150°C. Internal amplifiers can scale the output to convenient values, such as 10mV/°C. They are also useful in cold-junction compensation circuits for wide temperature range

thermocouples. Brief details about this type of temperature sensor are given below.

5.5.4.2 Sensor ICs

There is a wide variety of temperature sensor ICs that are available to simplify the broadest possible range of temperature monitoring challenges. These silicon temperature sensors differ significantly from the above-mentioned types in a couple of important ways. The first is the operating temperature range. A temperature sensor IC can operate over the nominal IC temperature range of -55°C to +150°C. The second major difference is functionality.

A silicon temperature sensor is an integrated circuit, and can, therefore, include extensive signal processing circuitry within the same package as the sensor. There is no need to add compensation circuits for temperature sensor ICS. Some of these are analog circuits with either voltage or current output. Others combine analog-sensing circuits with voltage comparators to provide alert functions. Some other sensor ICs combine analog-sensing circuitry with digital input/output and control registers, making them an ideal solution for microprocessor-based systems.

The digital output sensor usually contains a temperature sensor, analog-to-digital converter (ADC), a two-wire digital interface, and registers for controlling the IC's operation. Temperature is continuously measured and can be read at any time. If desired, the host processor can instruct the sensor to monitor temperature and take an output pin high (or low) if the temperature exceeds a programmed limit. Lower threshold temperature can also be programmed and the host can be notified when the temperature has dropped below this threshold. Thus, the digital output sensor can be used for reliable temperature monitoring in microprocessor-based systems.

Fig no 5.5.4.2.1 Temperature sensor

The above temperature sensor has three terminals and required Maximum of 5.5 V supply. This type of sensor consists of a material that operates according to temperature to vary the resistance. This change of resistance is sensed by the circuit and it calculates the temperature. When the voltage increases then the temperature also rises. We can see this operation by using a diode.

Temperature sensors directly connected to microprocessor input and thus capable of direct and reliable communication with microprocessors. The sensor unit can communicate effectively with low-cost processors without the need for A/D converters.

An example of a temperature sensor is **LM35**. The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius temperature. The LM35 operates at -55˚ to +120˚C.

The basic centigrade temperature sensor (+2˚C to +150˚C) is shown in the figure below.

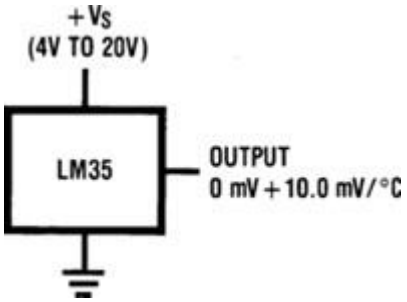


Fig no 5.5.4.2.2 Basic centigrade

5.5.4.3 Features of LM35 Temperature Sensor:

- Calibrated directly in ˚Celsius (centigrade)
- Rated for full 1-55˚ to +150˚C range

- Suitable For Remote Applications
- Low cost due to water-level trimming
- Operates from 4 to 30 volts
- Low Self-heating
- ±1/4˚C of typical nonlinearity

5.5.4.4 Operation of LM35

The LM35 can be connected easily in the same way as other integrated-circuit temperature sensors. It can be stuck or established to a surface and its temperature will be within around the range of 0.01˚C of the surface temperature.

This presumes that the ambient air temperature is just about the same as the surface temperature; if the air temperature were much higher or lower than the surface temperature, the actual temperature of the LM35 die would be at an intermediate temperature between the surface temperature and the air temperature.

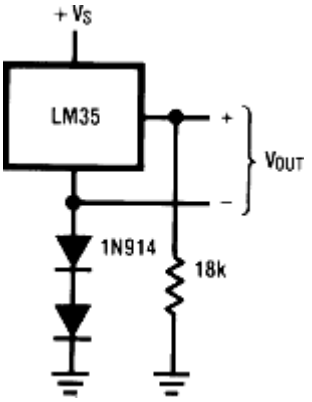


Fig no 5.5.4.4.1 operation of LM35

The temperature sensors have well-known applications in environmental and process control and also in the test, measurement, and communications. A digital temperature is a sensor, which provides 9-bit temperature readings. Digital temperature sensors offer excellent precise accuracy, these are designed to read from 0°C to 70°C and it is possible to achieve ±0.5°C accuracy. These sensors completely aligned with digital temperature readings in degrees Celsius.

5.6 POWER SUPPLY

5.6.1 INTRODUCTION

Many electronic circuits need a direct current (DC) voltage source, but what we commonly find are voltage sources of alternating current (AC). In order to achieve a direct current voltage source, the alternating current input must follow a conversion process like the one shown in the **power supply block diagram** below

The image shows the main components of a basic power supply diagram and the Waveforms at the beginning (AC input), at the end (DC output) and between blocks

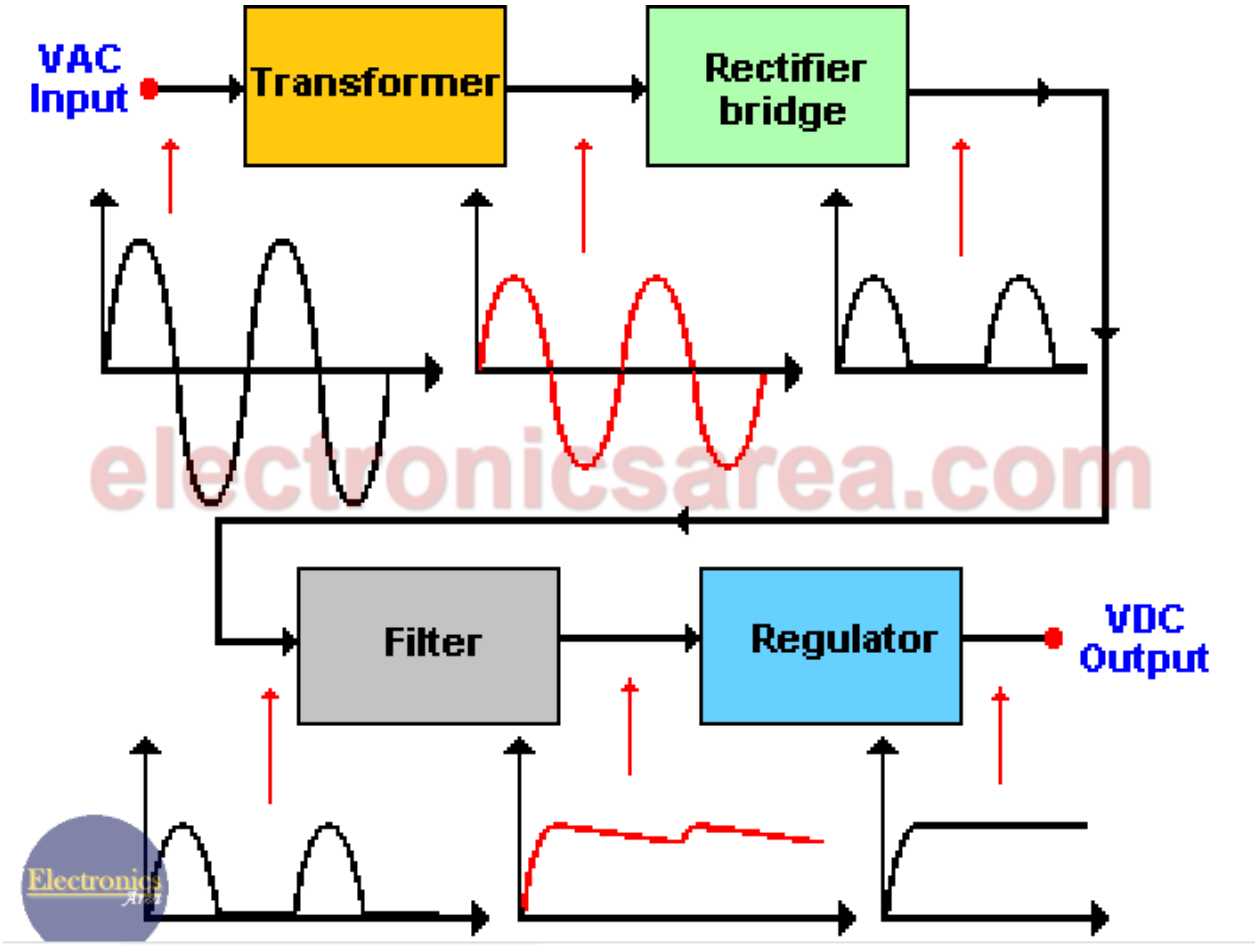


Fig no 5.6.1.1 Power Supply

Input signal which goes to the transformer primary winding is a sine wave and its amplitude depends on the country's electric distribution system (110/220 VAC or other). See the basic units of measurement in electronics.

Power Supply block diagram parts

5.6.2 Electrical Transformer

The electrical transformer receives on the primary winding an AC voltage and delivers on the secondary winding a different AC voltage (a lower one). This AC output voltage must be according to the DC voltage we want to obtain at the end.

For example: If we want a 12 VDC output, the secondary winding of the transformer must have an AC voltage no less of 9 volts.

Electric transformer

Fig no 5.6.2.1 Electric Transformer

The peak value at the transformer secondary winding is Vp = 1.41 x 9 = 12.69 volts. Even thought this value is very close to the one we wanted to get, it is not recommended because we need to take into account the voltage drops at different stages (blocks) of the power supply.

In this case, we can choose a transformer with a 12 volts AC secondary winding. With this AC voltage, we can get a peak voltage of: Vp = 12 x 1.41 = 16.92 volts.

Note: Vpeak = Vrms x 1.41

5.6.3 Rectifier bridge (rectifier diodes)

The rectifier bridge transforms the secondary winding AC voltage into a pulsating DC voltage. (look at the diagram). In our case, we use a ½ wave rectifier, then we eliminate the negative part of the wave.

Rectifier diode

Fig no 5.6.3.1 Rectifier Diode

5.6.4 Filter (capacitors)

Filter are one or more electrolytic capacitors in parallel that flattens or smooths the previous wave eliminating the alternating current (AC) component delivered by the rectifier.

These capacitors are charged to the maximum voltage value that the rectifier can deliver, and they are discharged when the pulsating signal disappears. Look at the picture above.

Electrolytic Capacitor

Fig no 5.6.4.1 Electrolytic Capacitor

5.6.5 Voltage regulator

The voltage regulator receives the signal from the filter and delivers a constant voltage (let's say 12 DC volts) regardless of the variations on the load or the voltage supply.

Voltage regulator can be implemented in several ways. It can be a transistorized voltage regulator or a monolithic voltage regulator.

The image below shows the LM7805 voltage regulator (5VDC output). You can also find the LM7812 voltage regulator (12VDC output).

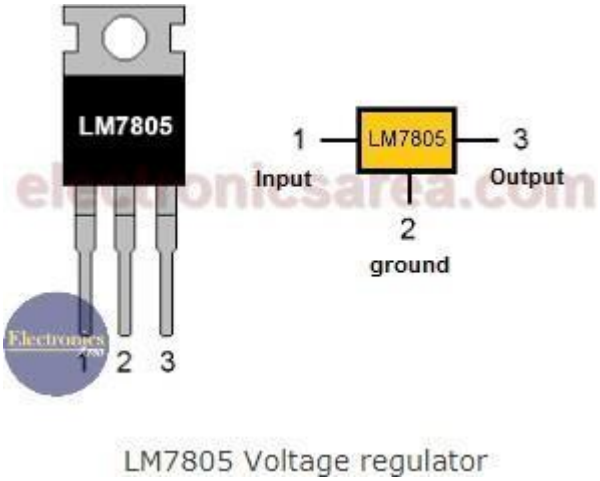


Fig no 5.6.5.1 Voltage Regulator

5.7 RELAY

The relay is the device that open or closes the contacts to cause the operation of the other electric control. It detects the undesirable condition with an assigned area gives the commands to the circuit breaker to disconnect the affected area through ON or OFF. Every electromechanical relay consists of

Electromagnet
Mechanically movable contact
Switching points and
Spring

Fig no 5.7.1 Relay

COM: common pin

NO: Normally open – there is no contact between the common pin and the normally open pin. So, when you trigger the relay, it connects to the COM pin and power is provided to the load.

NC: Normally closed – there is contact between the common pin and the normally closed pin. There is always connection between the COM and NC pins, even when the relay is turned off. When you trigger the relay, the circuit is opened and there is no supply provided to the load.

5.7.1 WORKING PRINCIPLE OF RELAY:

It works on the principle of an electromagnetic attraction. When the circuit of the relay senses the fault current, it energises the electromagnetic field which produces the temporary magnetic field.This magnetic field moves the relay armature for opening or closing the connections. The small power relay has only one contacts, and the high power relay has two contacts for opening the switch.

The inner section of the relay is shown in the figure below. It has an iron core which is wound by a control coil. The power supply is given to the coil through the contacts of the load and the control switch. The current flows through the coil produces the magnetic field around it.

Due to this magnetic field, the upper arm of the magnet attracts the lower arm. Hence close the circuit, which makes the current flow through the load. If the contact is already closed, then it moves oppositely and hence open the contacts.

Types of Relay Based on the principle of operation

1. Electrothermal relay:
2. Electromechanical relay:
3. Solid State relay:
4. Hybrid relay:

5.7.2 APPLICATIONS OF RELAY:

A. They can be used for both ac and dc systems for protection of ac and dc equipment's

B. Electromagnetic relays operating speeds which has the ability to operate in milliseconds are also can be possible

C. They have the properties such as simple, robust, compact and most reliable

D. These relays are almost instantaneous. Though instantaneous the operating time of the relay varies with the current. With extra arrangements like dashpot, copper rings.

E. Electromagnetic relays have fast operation and fast reset

5.7.3 DISADVANTAGES:

a. High burden level instrument transformers are required (CTs and PTs of high b.urden is required for operating the electromagnetic relays compared to static relays) The directional feature is absent in electromagnetic relays

c. Requires periodic maintenance and testing unlike static relays

d. Relay operation can be affected due to ageing of the components and dust, pollution resulting in spurious trips

e. Operation speed for an electromagnetic relays is limited by the mechanical inertia of the component.

5.7.4 APPLICATIONS:

1. Electromagnetic relays are employed for the protection of various ac and dc equipment's
2. The over/under current and voltage protection of various ac and dc equipment's
3. Used as auxiliary relays in the contact systems of protective relay schemes.

5.8 FIRE SENSOR:

5.8.1 INRODUCTION

This tiny Flame sensor infrared receiver module ignition source detection module is Arduino compatible can use to detect flame or wavelength of the light source within 760nm~1100nm also useful for Lighter flame detect at the distance 80cm.

Greater the flame, farther the test distance. It has the Detect angle of 60 and very sensitive to flame spectrum.

It produces the one channel output signal at the D0 terminal for further processing like an alarm system or any switching system. The sensitivity is adjustable with the help of blue potentiometer given on the board.

5.8.2 Features :

1. Indicator light: a green one for the switch, a red one for power.
2. Built in a potentiometer for sensitivity control.
3. Onboard signal output indication, output effective signal is high, at the same time the indicator light up, the output signal can directly connect to microcontroller IO.
4. Can detect fire or wavelength in 760 ~ 1100 nm nano within the scope of the light source.
5. Detection angle about 60 degrees, the flame spectrum especially sensitive.
6. The flame of the most sensitive sensors flame, the regular light is also a response, generally used for fire alarm purposes.

Fig no 5.8.2.1 Fire Sensor

5.9 LCD (Liquid Crystal Display) :

5.9.1 Introduction:

A liquid crystal display (LCD) is a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source or reflector. Each pixel consists of a column of liquid crystal molecules suspended between two transparent electrodes, and two polarizing filters, the axes of polarity of which are perpendicular to each other. Without the liquid crystals between them, light passing through one would be blocked by the other. The liquid crystal twists the polarization of light entering one filter to allow it to pass through the other.

A program must interact with the outside world using input and output devices that communicate directly with a human being. One of the most common devices attached to an controller is an LCD display. Some of the most common LCDs connected to the controllers are 16X1, 16x2 and 20x2 displays. This means 16 characters per line by 1 line 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

5.9.2 Features:

1. Interface with either 4-bit or 8-bit microprocessor.
2. Display data RAM.
3. Character generator ROM.
4. Display data RAM and character generator RAM may be Accessed by the microprocessor.
5. Numerous instructions.
6. Clear Display, Cursor Home, Display ON/OFF, Cursor ON/OFF, Blink Character, Cursor shift, Display Shift.
7. Built-in reset circuit is trigged at power ON
8. Built-in Oscillator

Data can be placed at any location on the LCD. For 16×1 LCD, the address locations are:

POSITION		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
ADDRESS	LINE1	00	01	02	03	04	05	06	07	40	41	42	43	44	45	46	47

Table 5.9.2.1 : Address locations for a 1x16 line LCD

Even limited to character based modules there is still a wide variety of shapes and sizes available. Line lengths of 8,16,20,24,32 and 40 characters are all standard, in one, two and four line versions. Several different LC technologies exists. “supertwist” types, for example, offer Improved contrast and viewing angle over the older “twisted nematic” types. Some modules are available with back lighting, so so that they can be viewed in dimly-lit conditions. The back lighting may be either “electro-luminescent”, requiring a high voltage inverter circuit, or simple LED illumination.

5.9.3 Electrical Block Diagram:

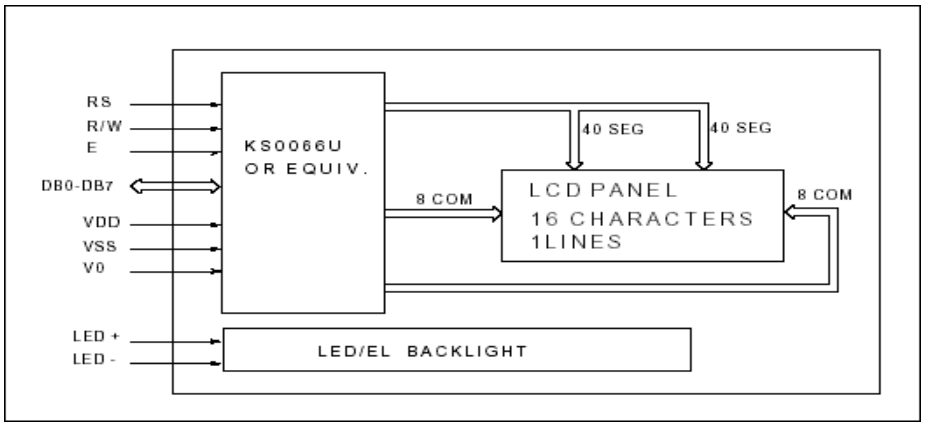


Fig 5.9.3.1 : Electrical Block Diagram

5.9.4 PIN DESCRIPTION:

Most LCDs with 1 controller has 14 Pins and LCDs with 2 controller has 16 Pins (two pins are extra in both for back-light LED connections).

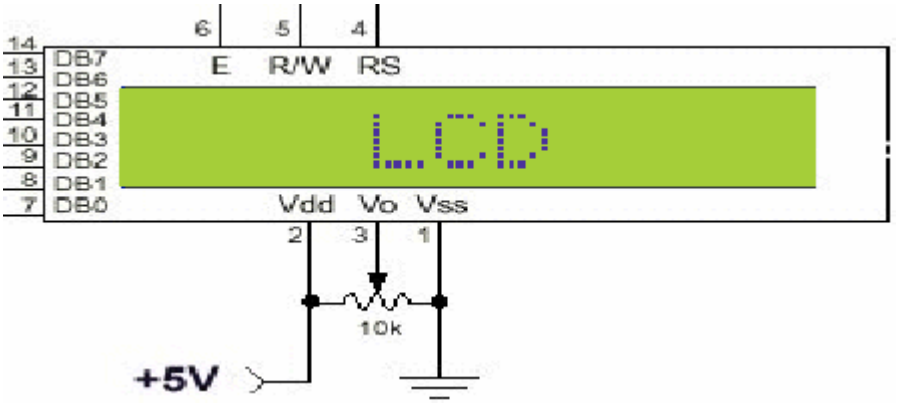


Fig 5.9.4.1 : pin diagram of 1x16 lines lcd

PIN	SYMBOL	FUNCTION
1	Vss	Power Supply(GND)
2	Vdd	Power Supply(+5V)
3	Vo	Contrast Adjust
4	RS	Instruction/Data Register Select
5	R/W	Data Bus Line
6	E	Enable Signal
7-14	DB0-DB7	Data Bus Line
15	A	Power Supply for LED B/L(+)
16	K	Power Supply for LED B/L(-)

Table 5.9.4.1 : Table for LCD

5.9.5 CONTROL LINES:

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

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 sould be displayed on the screen. For example, to display the letter "T" on the screen you would set RS high.

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.

Logic status on control lines:

• E - 0 Access to LCD disabled

- 1 Access to LCD enabled

• R/W - 0 Writing data to LCD

- 1 Reading data from LCD

• RS - 0 Instructions

- 1 Character

Writing data to the LCD:

1. Set R/W bit to low
2. Set RS bit to logic 0 or 1 (instruction or character)
3. Set data to data lines (if it is wrong)
4. Set E line to high
5. Set E line to low

Read data from data lines (if it is reading)on LCD:

1. Set R/W bit to high
2. Set RS bit to logic 0 or 1 (instruction or character)
3. Set data to data lines (if it is writing)
4. Set E line to high
5. Set E line to low

Entering Text:

First, a little tip: it is manually a lot easier to enter characters and commands in hexadecimal rather than binary (although, of course, you will need to translate commands from binary couple of sub-miniature hexadecimal rotary switches is a simple matter, although a little bit into hex so that you know which bits you are setting).

CHAPTER-6

SOFTWARE DESCRIPTION

CHAPTER-6

SOFTWARE DESCRIPTION

6.1 Arduino Software (IDE)

The Arduino Integrated Development Environment - or Arduino Software (IDE)- contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuine hardware to upload programs and communicate with them.

Writing Sketches

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

NB: Versions of the Arduino Software (IDE) prior to 1.0 saved sketches with the extension .pde. It is possible to open these files with version 1.0, you will be prompted to save the sketch with the .ino extension on save.

Verify

Checks your code for errors compiling it.

Upload

Compiles your code and uploads it to the configured board

See uploading below for details.

Note: If you are using an external programmer with your board, you can hold down the "shift" key on your computer when using this icon. The text will change to "Upload using Programmer"

New

Creates a new sketch.

Open

Presents a menu of all the sketches in your sketchbook. Clicking one wil open it within the current window overwriting its content.

Note: due to a bug in Java, this menu doesn't scroll; if you need to open a sketch late in the list, use the File | Sketchbookmenu instead.

Save

Saves your sketch.

SerialMonitor

Opens the serial monitor.

Table no 6.1.1 TYPES OF OPTIONS IN ARDUINO IDE

Additional commands which means only those items relevant to the work currently being carried out are available. are found within the five menus: File, Edit, Sketch, Tools, Help. The menus are context sensitive.

File

- **New**

  Creates a new instance of the editor, with the bare minimum structure of a sketch already in place.
- **Open**

  Allows to load a sketch file browsing through the computer drives and folders.
- **OpenRecent**

  Provides a short list of the most recent sketches, ready to be opened.
- **Sketchbook**

  Shows the current sketches within the sketchbook folder structure; clicking on any name opens the corresponding sketch in a new editor instance.
- **Examples**

  Any example provided by the Arduino Software (IDE) or library shows up in this menu item. All the examples are structured in a tree that allows easy access by topic or library.
- **Close**

  Closes the instance of the Arduino Software from which it is clicked.
- **Save**

  Saves the sketch with the current name. If the file hasn't been named before, a name will be provided in a "Save as.." window.
- **Saveas..**

  Allows to save the current sketch with a different name.
- **PageSetup**

  It shows the Page Setup window for printing.
- **Print**

  Sends the current sketch to the printer according to the settings defined in Page Setup.
- **Preferences**

  Opens the Preferences window where some settings of the IDE may be customized, as the language of the IDE interface.

- **Quit**

  Closes all IDE windows. The same sketches open when Quit was chosen will be automatically reopened the next time you start the IDE.

Edit

- **Undo/Redo**

  Goes back of one or more steps you did while editing; when you go back, you may go forward with Redo.
- **Cut**

  Removes the selected text from the editor and places it into the clipboard.
- **Copy**

  Duplicates the selected text from the editor and places it into the clipboard

- **Copy for forum**

  Copies the code of your sketch to the clipboard in a form suitable for posting to the forum, complete with syntax coloring.

- **Copy as HTML**

  Copies the code of your sketch to the clipboard as HTML, suitable for embedding in web pages.
- **Paste**

  Puts the contents of the clipboard at the cursor position, in the editor.
- **Select All**

  Selects and highlights the whole content of the editor.
- **Comment/Uncomment**

  Puts or removes the // comment marker at the beginning of each selected line.
- **Increase/Decrease Indent**

  Adds or subtracts a space at the beginning of each selected line, moving the text one space on the right or eliminating a space at the beginning.

- **Find**

  Opens the Find and Replace window where you can specify text to search inside the current sketch according to several options.
- **Find Next**

  Highlights the next occurrence - if any - of the string specified as the search item in the Find window, relative to the cursor position.
- **Find Previous**

  Highlights the previous occurrence - if any - of the string specified as the search item in the Find window relative to the cursor position.

Sketch

- **Verify/Compile**

  Checks your sketch for errors compiling it; it will report memory usage for code and variables in the console area.
- **Upload**

  Compiles and loads the binary file onto the configured board through the configured Port.
- **Upload Using Programmer**

  This will overwrite the bootloader on the board; you will need to use Tools > Burn Bootloader to restore it and be able to Upload to USB serial port again. However, it allows you to use the full capacity of the Flash memory for your sketch. Please note that this command will NOT burn the fuses. To do so a Tools -> Burn Bootloader command must be executed.
- **Export Compiled Binary**

  Saves a .hex file that may be kept as archive or sent to the board using other tools.
- **Show Sketch Folder**

  Opens the current sketch folder.
- **Include Library**

  Adds a library to your sketch by inserting #include statements at the start of your code.

For more details, see libraries below. Additionally, from this menu item you can access the Library Manager and import new libraries from .zip files.

- **Add File...**

  Adds a source file to the sketch (it will be copied from its current location). The new file appears in a new tab in the sketch window. Files can be removed from the sketch using the tab menu accessible clicking on the small triangle icon below the serial monitor one on the right side o the toolbar.

Tools

- **Auto Format**

  This formats your code nicely: i.e. indents it so that opening and closing curly braces line up, and that the statements inside curly braces are indented more.
- **Archive Sketch**

  Archives a copy of the current sketch in .zip format. The archive is placed in the same directory as the sketch.
- **Fix Encoding & Reload**

  Fixes possible discrepancies between the editor char map encoding and other operating systems char maps.
- **Serial Monitor**

  Opens the serial monitor window and initiates the exchange of data with any connected board on the currently selected Port. This usually resets the board, if the board supports Reset over serial port opening.
- **Board**

  Select the board that you're using. See below for descriptions of the various boards.
- **Port**

  This menu contains all the serial devices (real or virtual) on your machine. It should automatically refresh every time you open the top-level tools menu.
- **Programmer**

  For selecting a harware programmer when programming a board or chip and not using the onboard USB-serial connection. Normally you won't need this, but if you're burning a bootloader to a new microcontroller, you will use this.

- **Burn Bootloader**

  The items in this menu allow you to burn a bootloader onto the microcontroller on an Arduino board. This is not required for normal use of an Arduino or Genuino board but is useful if you purchase a new ATmega microcontroller (which normally come without a bootloader). Ensure that you've selected the correct board from the Boards menu before burning the bootloader on the target board. This command also set the right fuses.

Help

Here you find easy access to a number of documents that come with the Arduino Software (IDE). You have access to Getting Started, Reference, this guide to the IDE and other documents locally, without an internet connection. The documents are a local copy of the online ones and may link back to our online website.

- **Find in Reference**

  This is the only interactive function of the Help menu: it directly selects the relevant page in the local copy of the Reference for the function or command under the cursor.

Sketchbook

The Arduino Software (IDE) uses the concept of a sketchbook: a standard place to store your programs (or sketches). The sketches in your sketchbook can be opened from the File > Sketchbook menu or from the Open button on the toolbar. The first time you run the Arduino software, it will automatically create a directory for your sketchbook. You can view or change the location of the sketchbook location from with the Preferences dialog.

Beginning with version 1.0, files are saved with a .ino file extension. Previous versions use the .pde extension. You may still open .pde named files in version 1.0 and later, the software will automatically rename the extension to .ino.

Tabs, Multiple Files, and Compilation

Allows you to manage sketches with more than one file (each of which appears in its own tab). These can be normal Arduino code files (no visible extension), C files (.c extension), C++ files (.cpp), or header files (.h).

Uploading

Before uploading your sketch, you need to select the correct items from the Tools > Board and Tools > Port menus. The boards are described below. On the Mac, the serial port is probably something like /dev/tty.usbmodem241 (for an Uno or Mega2560 or Leonardo) or /dev/tty.usbserial-1B1 (for a Duemilanove or earlier USB board), or /dev/tty.USA19QW1b1P1.1 (for a serial board connected with a Keyspan USB-to-Serial adapter). On Windows, it's probably COM1 or COM2 (for a serial board) or COM4, COM5, COM7, or higher (for a USB board) - to find out, you look for USB serial device in the ports section of the Windows Device Manager. On Linux, it should be /dev/ttyACMx , /dev/ttyUSBx or similar. Once you've selected the correct serial port and board, press the upload button in the toolbar or select the Upload item from the Sketch menu. Current Arduino boards will reset automatically and begin the upload. With older boards (pre-Diecimila) that lack auto-reset, you'll need to press the reset button on the board just before starting the upload. On most boards, you'll see the RX and TX LEDs blink as the sketch is uploaded. The Arduino Software (IDE) will display a message when the upload is complete, or show an error.

When you upload a sketch, you're using the Arduino bootloader, a small program that has been loaded on to the microcontroller on your board. It allows you to upload code without using any additional hardware. The bootloader is active for a few seconds when the board resets; then it starts whichever sketch was most recently uploaded to the microcontroller. The bootloader will blink the on-board (pin 13) LED when it starts (i.e. when the board resets).

Libraries

Libraries provide extra functionality for use in sketches, e.g. working with hardware or manipulating data. To use a library in a sketch, select it from the Sketch > Import Library menu. This will insert one or more #include statements at the top of the sketch and compile the library with your sketch. Because libraries are uploaded to the board with your sketch, they increase the amount of space it takes up. If a sketch no longer needs a library, simply delete its #includestatements from the top of your code.

There is a list of libraries in the reference. Some libraries are included with the Arduino software. Others can be downloaded from a variety of sources or through the Library Manager. Starting with version 1.0.5 of the IDE, you do can import a library from a zip file and use it in an open sketch. See these instructions for installing a third-party library.

To write your own library, see this tutorial.

Serial Monitor

Displays serial data being sent from the Arduino or Genuino board (USB or serial board). To send data to the board, enter text and click on the "send" button or press enter. Choose the baud rate from the drop-down that matches the rate passed to Serial.begin in your sketch. Note that on Windows, Mac or Linux, the Arduino or Genuino board will reset (rerun your sketch execution to the beginning) when you connect with the serial monitor.You can also talk to the board from Processing, Flash, MaxMSP, etc (see the interfacing page for details).

Preferences

Some preferences can be set in the preferences dialog (found under the Arduino menu on the Mac, or File on Windows and Linux). The rest can be found in the preferences file, whose location is shown in the preference dialog.

Language Support

Fig no 6.1.1 LANGUAGES SUPPORT FOR ARDUINO IDE

Since version 1.0.1 , the Arduino Software (IDE) has been translated into 30+ different languages. By default, the IDE loads in the language selected by your operating system. (Note: on Windows and possibly Linux, this is determined by the locale setting which controls currency and date formats, not by the language the operating system is displayed in.)

If you would like to change the language manually, start the Arduino Software (IDE) and open the Preferences window. Next to the Editor Language there is a dropdown menu of currently supported languages. Select your preferred language from the menu, and restart the software to use the selected language. If your

operating system language is not supported, the Arduino Software (IDE) will default to English.

You can return the software to its default setting of selecting its language based on your operating system by selecting System Default from the Editor Language drop-down. This setting will take effect when you restart the Arduino Software (IDE). Similarly, after changing your operating system's settings, you must restart the Arduino Software (IDE) to update it to the new default language.

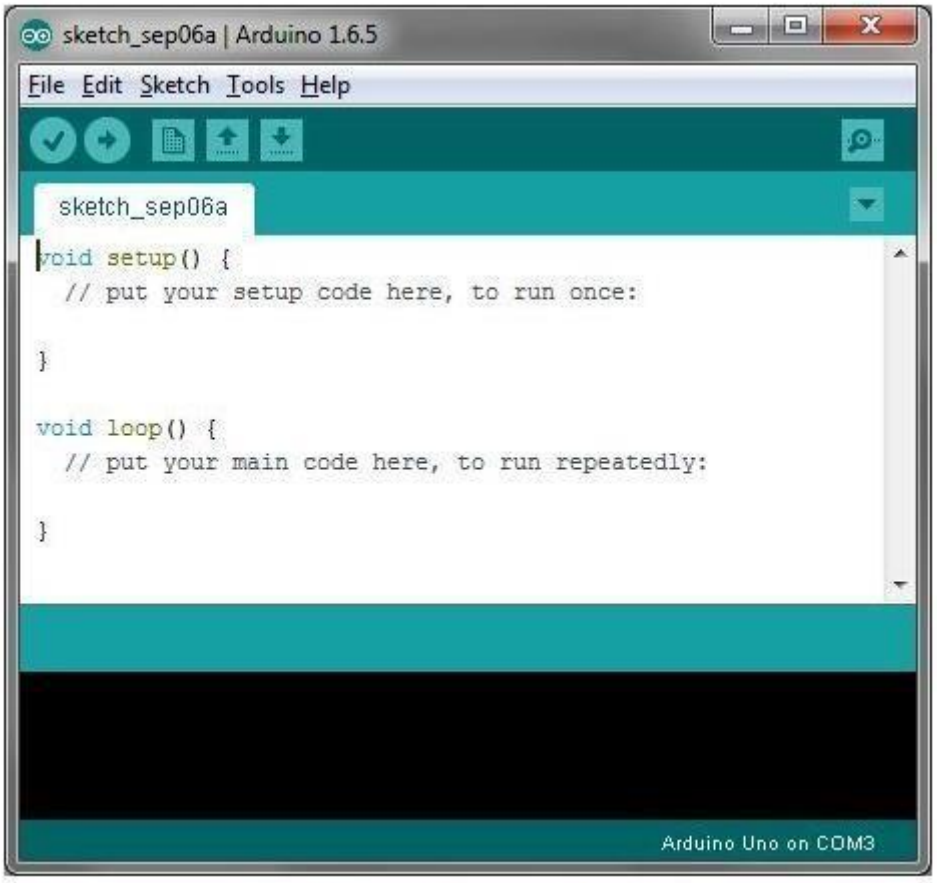
Boards

The board selection has two effects: it sets the parameters (e.g. CPU speed and baud rate) used when compiling and uploading sketches; and sets and the file and fuse settings used by the burn bootloader command. Some of the board definitions differ only in the latter, so even if you've been uploading successfully with a particular selection you'll want to check it before burning the bootloader. You can find a comparison table between the various boards here.

Arduino Software (IDE) includes the built in support for the boards in the following list, all based on the AVR Core. The Boards Manager included in the standard installation allows to add support for the growing number of new boards based on different cores like Arduino Due, Arduino Zero, Edison, Galileo and so on.

6.2 Arduino IDE: Initial Setup

Download Arduino Integrated Design Environment (IDE) here (Most recent version: 1.6.5): https://www.arduino.cc/en/Main/Software This is the Arduino IDE once it’s been opened. It opens into a blank sketch where you can start programming immediately. First, we should configure the board and port settings to allow us to upload code. Connect your Arduino board to the PC via the USB cable.

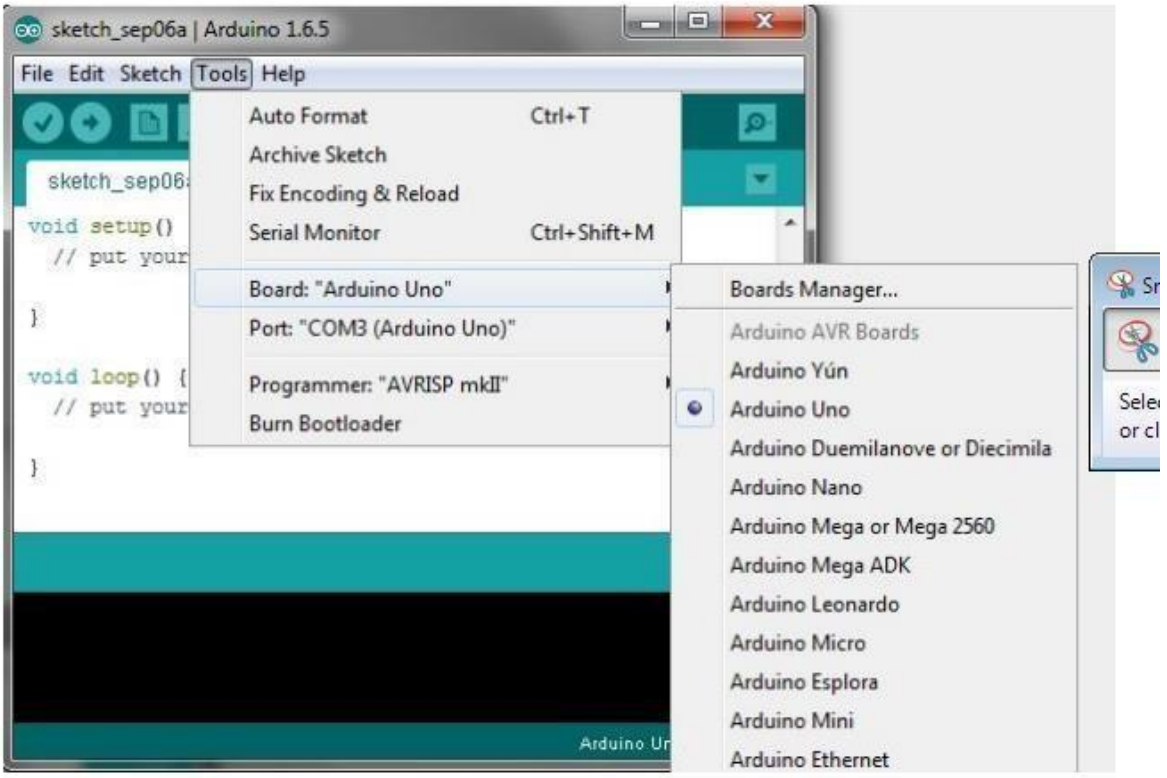


Arduino IDE Default Window

Fig no 6.2.1 ARDUINO IDE DEFAULT WINDOW

IDE: Board Setup

You have to tell the Arduino IDE what board you are uploading to. Select the Toolspulldown menu and go to Board.This list is populated by default with the currently available Arduino Boards that are developed by Arduino. If you are using an Uno or an Uno-Compatible Clone (ex. Funduino, SainSmart, IEIK, etc.), select Arduino Uno. If you are using another board/clone, select that board.

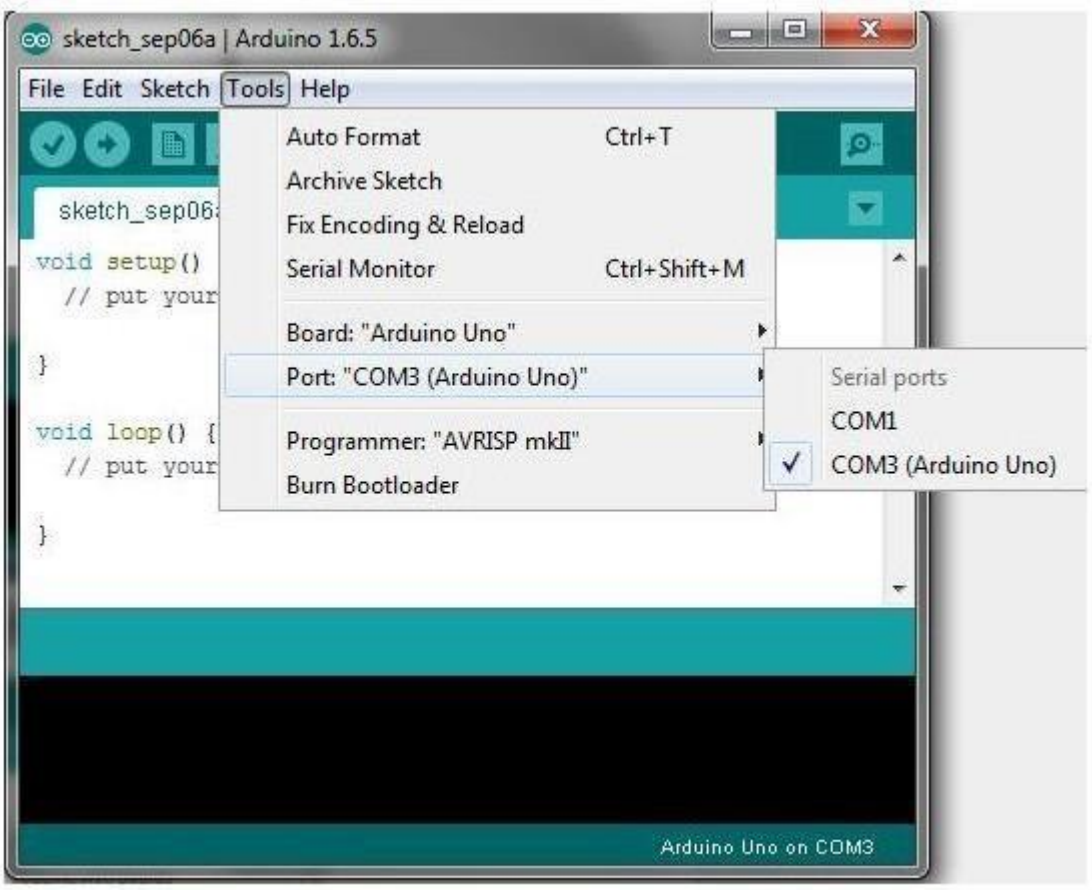


Arduino IDE: Board Setup Procedure

Fig no 6.2.2 ARDUINO IDE: BOARD SETUP PROCEDURE

IDE: COM Port Setup

If you downloaded the Arduino IDE before plugging in your Arduino board, when you plugged in the board, the USB drivers should have installed automatically. The most recent Arduino IDE should recognize connected boards and label them with which COM port they are using. Select the Tools pulldown menu and then Port.Here it should list all open COM ports, and if there is a recognized Arduino Board, it will also give it's name. Select the Arduino board that you have connected to the PC. If the setup was successful, in the bottom right of the Arduino IDE, you should see the board type and COM number of the board you plan to program. Note: the Arduino Uno occupies the next available COM port; it will not always be COM3.



Arduino IDE: COM Port Setup

Fig no 6.2.3 ARDUINO IDE: COM PORT SETUP

At this point, your board should be set up for programming, and you can begin writing and uploading code.

Testing Your Settings: Uploading Blink

One common procedure to test whether the board you are using is properly set up is to upload the “Blink” sketch. This sketch is included with all Arduino IDE releases and can be accessed by the Filepull-down menu and going to Examples, 01.Basics, and then select Blink. Standard Arduino Boards include a surface-mounted LED labeled “L” or “LED” next to the “RX” and “TX” LEDs, that is connected to digital pin 13. This sketch will blink the LED at a regular interval, and is an easy way to confirm if your board is set up properly and you were successful in uploading code. Open the “Blink” sketch and press the “Upload” button in the upper-left corner to upload “Blink” to the board.

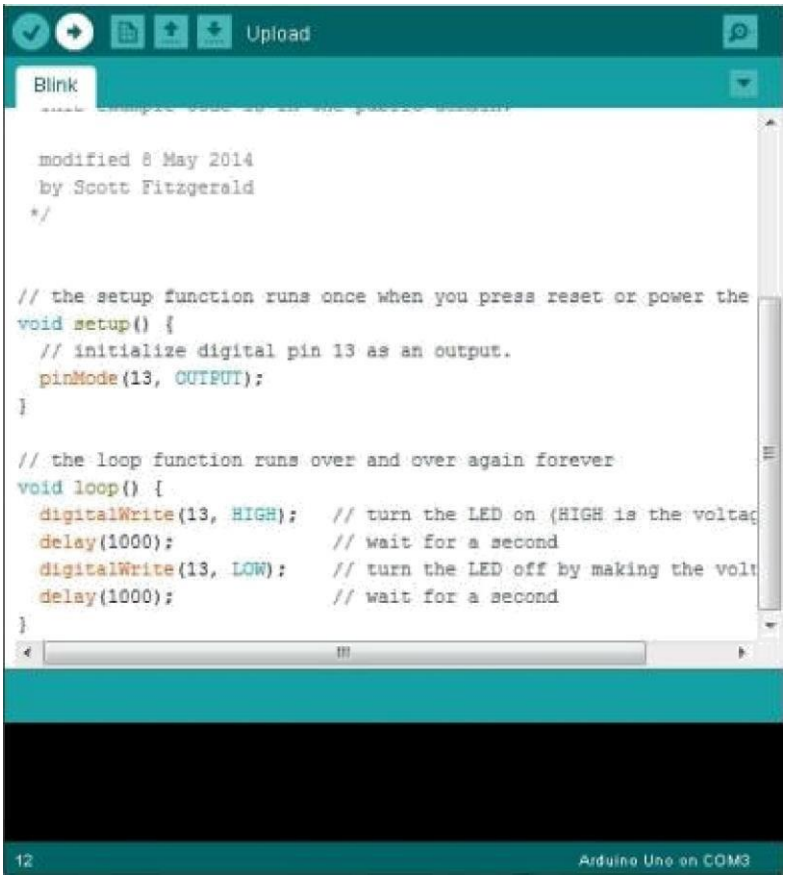


Fig no 6.2.4 ARDUINO IDE: UPLOADING BLINK

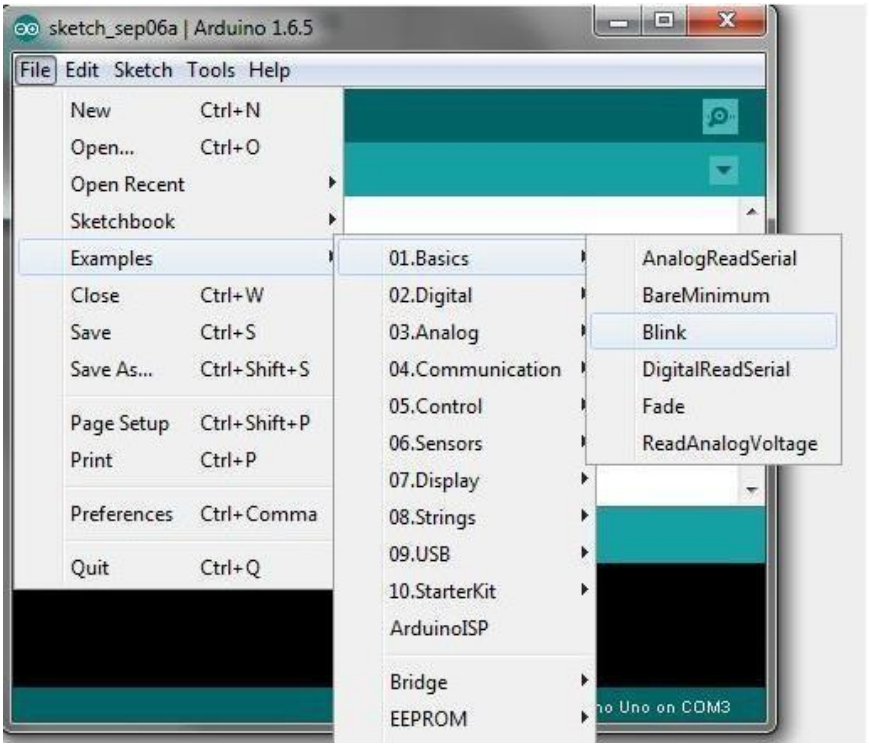


Fig no 6.2.5 ARDUINO IDE: LOADING BLINK SKETCH

Guide Summary:

1. Download and install Arduino IDE (https://www.arduino.cc/en/Main/Software)
2. Plug in your Arduino Board
3. Select the proper board in the IDE (Tools>Boards>Arduino Uno)
4. Select the proper COM port (Tools>Port>COMx (Arduino Uno))
5. Open the “Blink” sketch (File>Examples>Basics>01.Blink)
6. Press the Upload button to upload the program to the board
7. Confirm that your board is working as expected by observing LED

Troubleshooting Uploading Errors:

Arduino has lots of community support and documentation. Your best bet when running into unexpected problems is to search online for help. You should be able to find a forum where someone had the same problem you are having, and someone helped them fix it. If you don’t find results, try modifying your search, or post on the Arduino forums.

● My board isn’t listed under devices and is not recognized by IDE:

○ Most likely, this means that the ATMega328p chip is not programmed with the Arduino firmware. If you have a separate working Uno available, you can program the unprogrammed chip using this guide and a few jumper cables: https://www.arduino.cc/en/Tutorial/ArduinoISP

○ If you don’t have a separate Arduino available, let me know and I can use an Atmel Programmer to upload the firmware.

○ There may be hardware damage if you had the board plugged into USB andexternal power at the same time. You may have to replace the chip if this is the case.

● Error Message: avrdude: stk500_recv(): programmer is not responding

○ Double-check that you are using the correct COM port.

○ Make sure that your Arduino Board is plugged into the computer.

● The IDE says “Uploading…” after pressing the upload button, but nothing is happening.

○ Double-check that you have the correct board selected in the Tools menu.

○ Depending on the size of your program, it may take a few seconds to upload. If you feel like it is taking too long, it may be encountering an error and you can try unplugging and plugging in the Arduino board.

CHAPTER-7

RESULTS

CHAPTER-7

RESULTS

(ADD KIT PHOTOS AND DESCRIPTION)

- This system measures temperature,light frequently
- In this kit we will set normal Temperature value when its reads more than that value fan will rotate
- In this we have fire sensor when it detects fire in industry the buzzer will activate automatically

Fig no 7.1 project kit

Fig no 7.2 PROJECT OUTPUT

CHAPTER-8

CONCLUSION

CHAPTER-8

CONCLUSION

This paper describes a IOT Based Reconfigurable smart WSN unit for industrial safety parameters monitoring. The system can collect sensor data intelligently. It was designed based on application of wireless communication. It is very suitable for real-time and effective requirements of the high-speed data acquisition system in IoT environment. The application of ARDUINO UNO greatly simplifies the design of peripheral circuit, and makes the whole system more flexible and extensible. Different types of sensors can be used as long as they are connected to the system. Main design method of the reconfigurable smart sensor interface device is described in this paper. Finally, by taking industrial safety parameters monitoring in IoT environment as an example, we verified that the system achieved good effects in practical application. Nevertheless, many interesting directions are remaining for further researches in the area of WSN in IoT environment.

REFERENCES

REFERENCES

[1] S. Li, L. Xu, X. Wang, and J. Wang, “Integration of hybrid wireless networks in cloud services oriented enterprise information systems,” Enterp. Inf. Syst., vol. 6, no. 2, pp. 165–187, 2012.

[2] Q. Li, Z. Wang, W. Li, J. Li, C. Wang, and R. Du, “Applications integration in a hybrid cloud computing environment: Modelling and platform,” Enterp. Inf. Syst., vol. 7, no. 3, pp. 237–271, 2013.

[3] L. Wang, L. D. Xu, Z. Bi, and Y. Xu, “Data cleaning for RFID and WSN integration,” IEEE Trans. Ind. In format., vol. 10, no. 1, pp. 408–418, Feb. 2014.

[4] Y. Fan, Y. Yin, L. Xu, Y. Zeng, and F. Wu, “IoT based smart rehabilitation system,” IEEE Trans. Ind. Informat., vol. 10, no. 2, pp. 1568–1577, 2014.

[5] W. He, G. Yan, and L. Xu, “Developing vehicular data cloud services in the IoT environment,” IEEE Trans. Ind. Informat., vol. 10, no. 2, pp. 1587–1595, 2014.

[6] Mihai T. Lazarescu, “Design of a WSN Platform for Long-TermEnvironmental Monitoring for IoT Applications” IEEE Journal on emerging and selected topics in circuits and systems, vol. 3, no. 1, March 2013

[7] Ahmad El Kouche“Towards a Wireless Sensor Network Platform for the Internet of Things”, IEEE ICC 2012 - Ad-hoc and Sensor Networking Symposium

[8] E.J. Duarte-Melo and M. Liu, “Data-Gathering Wireless Sensor Networks: Organization and Capacity,” Computer Networks, vol. 43, pp. 519- 537, 2003

[9] Siyuan Chen, Minsu Huang, Shaojie Tang, Yu Wang, “Capacity of Data Collection in Arbitrary Wireless Sensor Networks”, IEEE transactions on parallel and distributed systems, vol. 23, no. 1, January 2012.

[10]Y. Li, M. Hou, H. Liu, and Y. Liu, “Towards a theoretical framework of strategic decision, supporting capability and information sharing under the context of Internet of Things,” Inf. Technol. Manage., vol. 13, no. 4, pp. 205–216, 2012.

[11] Li Da Xu, Wu He, and Shancang Li, “Internet of Things in Industries: A Survey”, IEEE transactions on industrial informatics, vol. 10, no. 4, November 2014.

[12] D.Uckelmann, M. Harrison, and F. Michahelles, “An architectural approach towards the future internet of things,” in Architecting the Internet of Things. D. Uckelmann, M. Harrison, and F. Michahelles, Eds., New York, NY, USA: Springer, 2011, pp 1–24.

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

IOT INDUSTRY PROTECTION USING ARDUINO

DORNADULA LOHITH[1], KAIPU VINOD KUMAR REDDY[2], INDURU BHARATH KUMAR REDDY[3] V.JEYARAMYA[4]

[1]STUDENT, DEPT. OF ECE, PANIMALAR INSTITUTE OF TECHNOLOGY, CHENNAI
[2]STUDENT, DEPT. OF ECE, PANIMALAR INSTITUTE OF TECHNOLOGY, CHENNAI
[3]STUDENT, DEPT. OF ECE, PANIMALAR INSTITUTE OF TECHNOLOGY, CHENNAI
[4]PROFESSOR: V.JEYARAMYA, DEPT. OF ECE, PANIMALAR INSTITUTE OF TECHNOLOGY, CHENNAI

Abstract - *Today, smart grid, smart homes, smart water networks, intelligent transportation, are infrastructure systems that connect our world more than we ever thought possible.The common vision of such systems is usually associated with one single concept, the Internet of Things (IoT), where through the use of sensors, the entire physical infrastructure is closely coupled with information and communication technologies; where intelligent monitoring and management can be achieved via the usage of networked embedded devices. These devices will connect to internet to share different types of data. We have proposed an Industrial Monitoring System using WIFI module and sensing based applications for internet of things. By detecting the values of sensors it can easily find out the Temperature, humidity, and gas present in the industrial area.*

Key Words: IoT, Message Alert system, WIFI-Module,ETC...

1.INTRODUCTION –

Now- a – days, the industrial monitoring field requires more manual power to monitor and control the industrial parameters such as temperature, humidity, gas etc. this is the most upcoming issues in the industrial sectors. if the parameters are not monitored and control properly, it leads to a harmful situation. Most of the industries are facing those kinds of situation because of some manual mistakes. To overcome manual mistakes we are using industrial automation with internet of things. WIRELESS SENSOR NETWORKS (WSN) has been employed to collect data about physical phenomena in various applications such as habitat monitoring, and ocean monitoring, and surveillance

1.1 EXISTING SYSTEM

Implementation of ZigBee-GSM based home security monitoring and remote control systemArbab Waheed Ahmad , Naeem Jan, Saeed Iqbal and Chankil Lee proposed Implementation of ZigBeeGSM based Home Security Monitoring and Remote Control system. Home security and control is one among the essential needs of mankind from youth. But today it's to be updated with the rapidly changing technology to make sure vast coverage, remote, reliability, and real time operation. Deploying wireless technologies for security and control in home automation systems offers attractive benefits along side user friendly interface INDUSTRY BASED SECURITY SYSTEM USING GSM AND ARDUINO(Shubham Raut, Avinash Gaikwad , Mudaliyar Raghurajan , Pratiksha Patil)This paper describes the design of a simple low cost GSM based security monitoring system using GSM technique

1.2 PROPOSED SYSTEM

The IOT industry protection system using Arduino is a system designed to protect industries from losses due to accidents using Internet of things. Gas leakages may lead to fires leading to huge industrial losses, also instant fire detection is needed in case of furnace blasts or other conditions. Also low lighting in industries may create improper work conditions increasing the probability of accidents. The system makes use of Arduino to achieve this functionality. The system makes use of temperature sensing along with light and gas sensing to detect fire, gas leakage as well as low lighting to avoid any industrial accidents and prevent losses. The system consists of light, gas and temperature sensors interfaced with Arduino and LCD screen.

2. HARDWARE REQUIREMENTS

- ARDUINO UNO
- LCD DISPLAY
- WIFI MODULE
- LDR SENSOR
- GAS SENSOR
- TEMPERATURE SENSOR
- RELAY
- POWER SUPPLY

3.SOFTWARE REQUIREMENTS

- ARDUINO IDE
- EMBEDDED C

MODULE DESCRIPTION

The module is divided into three parts

- Power supply
- Hardware connections
- Software interfacing

POWER SUPPLY

The power supply mainly consists of four parts

- Electrical Transformer
- Rectifier Diode
- Electrolytic Capacitor
- LM8705 Voltage Regulator

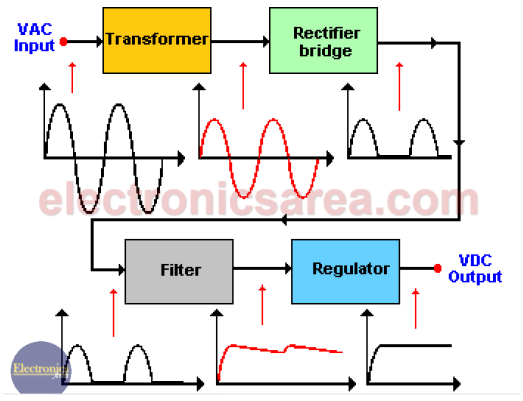


Fig -1: power supply

ELECRTRICAL TRANSFORMER

The electrical transformer receives on the primary winding an AC voltage and delivers on the secondary winding a different AC voltage (a lower one). This AC output voltage must be according to the DC voltage we want to obtain at the end.

RECTIFIER DIODE

The rectifier bridge transforms the secondary winding AC voltage into a pulsating DC voltage. (look at the diagram). In our case, we use a ½ wave rectifier, then we eliminate the negative part of the wave

ELECTROLYTIC CAPACITOR

Filter are one or more electrolytic capacitors in parallel that flattens or smooths theprevious wave eliminating the alternating current (AC) component delivered by the rectifier. These capacitors are charged to the maximum voltage value that the rectifier can deliver, and they are discharged when the pulsating signal disappears

LM8705 VOLTAGE REGULATOR

The voltage regulator receives the signal from the filter and delivers a constant voltage (let's say 12 DC volts) regardless of the variations on the load or the voltage supply. Voltage regulator can be implemented in several ways. It can be a transistorized voltage regulator or a monolithic voltage regulator. The image below shows the LM7805 voltage regulator (5VDC output). You can also find the LM7812 voltage regulator (12VDC output).

HARDWARE CONNECTONS

In this section, The system contains several sensors for measurement of parameters. The sensors used in this project is LDR(Light Dependent Resistor)sensor, GAS sensor, Fire sensor for calculating the values in the industry

THE MAJOR HARDWARE COMPONENTS ARE

- ARDUINO UNO
- LDR SENSOR
- GAS SENSOR
- FIRE SENSOR

ARDUINO UNO

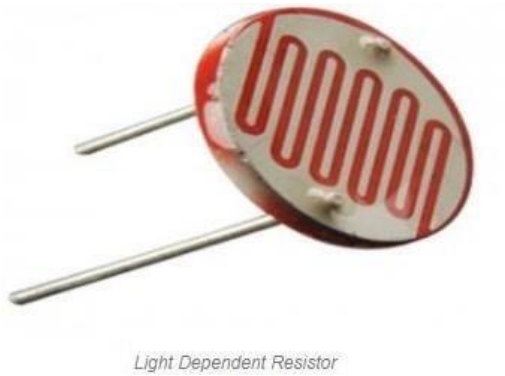
Arduino is a single-board microcontroller meant to make the application more accessible which are interactive objects and its surroundings. The hardware features with an open- source hardware board designed around an 8-bit Atmel AVR microcontroller or a 32-bit Atmel ARM. Current models consists a USB interface, 6

analog input pins and 14 digital I/O pins that allows the user to attach various extension boards

Arduino Uno Board

LDR SENSOR

The controlling of lights and home appliances is generally operated and maintained manually on several occasions. But the process of appliances controlling may cause wastage of power due to the carelessness of human beings or unusual circumstances. To overcome this problem we can use the light-dependent resistor circuit for controlling the loads based on the intensity of light



Light Dependent Resistor

GAS SENSOR

A Typical human nose has 400 types of scent receptors enabling us to smell about 1 trillion different odours. But still many of us do not have the capacity to identify the type or concentration of gas present in our atmosphere. This is where Sensors comes in, there are many types of sensors to measure different parameters and a **Gas sensor** is one which comes handy in applications where we have to detect the variation in the concentration of toxic gases in order to maintain the system safe and avoid/caution any unexpected threats.

FIRE SENOR

This tiny Flame sensor infrared receiver module ignition source detection module is Arduino compatible can use to detect flame or wavelength of the light source within 760nm~1100nm also useful for Lighter flame detect at the distance 80cm Greater the flame, farther the test distance. It has the Detect angle of 60 and very sensitive to flame spectrum

SOFTWARE REQUIREMENTS

- ARDUINO IDE
- EMBEDDED C

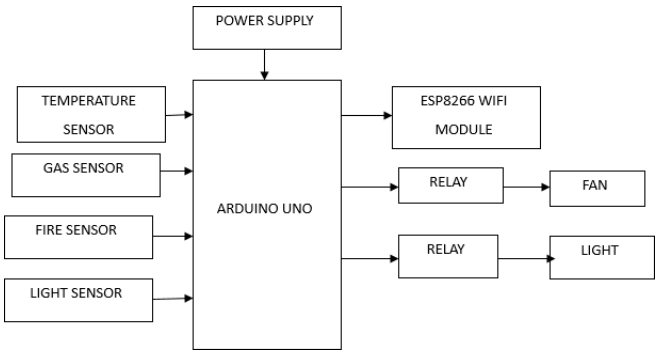
ARDUINO IDE

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuine hardware to upload programs and communicate with them

TOOLS

- ARCHIVE SKETCH
- FIX ENCODING AUTO FORMAT & RELOAD
- SERIAL MONITOR
- BOARD
- PORT
- BURN BOOTLOADER

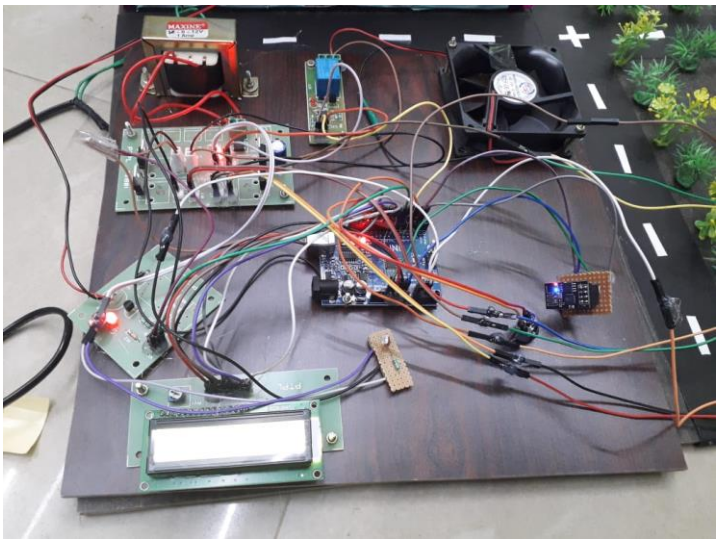
BLOCK DIAGRAM



CONCLISION

This paper describes a IOT Based Reconfigurable smart WSN unit for industrial safety parameters monitoring. The system can collect sensor data intelligently. It was designed based on application of wireless communication. It is very suitable for real-time and effective requirements of the high-speed data acquisition system in IoT environment. The application of ARDUINO UNO greatly simplifies the design of peripheral circuit, and makes the whole system more flexible and extensible. Different types of sensors can be used as long as they are connected to the system. Main design method of the reconfigurable smart sensor interface device is described in this paper. Finally, by taking industrial safety parameters monitoring in IoT environment as an example, we verified that the system achieved good effects in practical application. Nevertheless, many interesting directions are remaining for further researches in the area of WSN in IoT environment.

RESULT



REFERENCES

1 S. Li, L. Xu, X. Wang, and J. Wang, “Integration of hybrid wireless networks in cloud services oriented enterprise information systems,” Enterp. Inf. Syst., vol. 6, no. 2, pp. 165–187, 2012.

2 Q. Li, Z. Wang, W. Li, J. Li, C. Wang, and R. Du, “Applications integration in a hybrid cloud computing environment: Modelling and platform,” Enterp. Inf. Syst., vol. 7, no. 3, pp. 237–271, 2013.

3 L. Wang, L. D. Xu, Z. Bi, and Y. Xu, “Data cleaning for RFID and WSN integration,” IEEE Trans. Ind. In format., vol. 10, no. 1, pp. 408–418, Feb. 2014.

4 Y. Fan, Y. Yin, L. Xu, Y. Zeng, and F. Wu, "IoT based smart rehabilitation system," IEEE Trans. Ind. Informat., vol. 10, no. 2, pp. 1568–1577, 2014.

5 W. He, G. Yan, and L. Xu, "Developing vehicular data cloud services in the IoT environment," IEEE Trans. Ind. Informat., vol. 10, no. 2, pp. 1587–1595, 2014.

6 Mihai T. Lazarescu, "Design of a WSN Platform for Long-Term Environmental Monitoring for IoT Applications" IEEE Journal on emerging and selected topics in circuits and systems, vol. 3, no. 1, March 2013

7 Ahmad El Kouche"Towards a Wireless Sensor Network Platform for the Internet of Things", IEEE ICC 2012 - Ad-hoc and Sensor Networking Symposium

8 E.J. Duarte-Melo and M. Liu, "Data-Gathering Wireless Sensor Networks: Organization and Capacity," Computer Networks, vol. 43, pp. 519-537, 2003

9 Siyuan Chen, Minsu Huang, Shaojie Tang, Yu Wang, "Capacity of Data Collection in Arbitrary Wireless Sensor Networks", IEEE transactions on parallel and distributed systems, vol. 23, no. 1, January 2012.

10 Y. Li, M. Hou, H. Liu, and Y. Liu, "Towards a theoretical framework of strategic decision, supporting capability and information sharing under the context of Internet of Things," Inf. Technol. Manage., vol. 13, no. 4, pp. 205–216, 2012.

PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)

(Jaisakthi Educational Trust)
Approved by AICTE, New Delhi & Affiliated to Anna University
An ISO 9001:2008 Certified Institution
Bangalore Trunk Road ,Poonamallee, Chennai 600123
Accredited by National Board of Accreditation (NBA)

Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology

(ICIETET '21)

Certificate of Participation

This is to certify Prof./Dr./Mr./Ms./Mrs **V.JEYARAMYA** of **PANIMALAR INSTITUTE OF TECHNOLOGY** has presented a paper on **IOT INDUSTRY PROTECTION USING ARDUINO** in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on **12th April, 2021**.

CONFERENCE CHAIR
Dr. T. Jayanthy, Principal

PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)

(Jaisakthi Educational Trust)
Approved by AICTE, New Delhi & Affiliated to Anna University
An ISO 9001:2008 Certified Institution
Bangalore Trunk Road ,Poonamallee, Chennai 600123
Accredited by National Board of Accreditation (NBA)

Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology

(ICIETET '21)

Certificate of Participation

This is to certify Prof./Dr./Mr./Ms./Mrs **DORNADULA LOHITH** of **PANIMALAR INSTITUTE OF TECHNOLOGY** has presented a paper on **IOT INDUSTRY PROTECTION USING ARDUINO** in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on **12th April, 2021**.

CONFERENCE CHAIR
Dr. T. Jayanthy, Principal

PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)

(Jaisakthi Educational Trust)
Approved by AICTE, New Delhi & Affiliated to Anna University
An ISO 9001:2008 Certified Institution
Bangalore Trunk Road ,Poonamallee, Chennai 600123
Accredited by National Board of Accreditation (NBA)

Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology

(ICIETET '21)

Certificate of Participation

This is to certify Prof./Dr./Mr./Ms./Mrs **INDURU BHARATH KUMAR REDDY** of **PANIMALAR INSTITUTE OF TECHNOLOGY** has presented a paper on **IOT INDUSTRY PROTECTION USING ARDUINO** in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on **12th April, 2021**.

CONFERENCE CHAIR
Dr. T. Jayanthy, Principal

PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)

(Jaisakthi Educational Trust)
Approved by AICTE, New Delhi & Affiliated to Anna University
An ISO 9001:2008 Certified Institution
Bangalore Trunk Road ,Poonamallee, Chennai 600123
Accredited by National Board of Accreditation (NBA)

Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology

(ICIETET '21)

Certificate of Participation

This is to certify Prof./Dr./Mr./Ms./Mrs **KAIPU VINOD KUMAR REDDY** of **PANIMALAR INSTITUTE OF TECHNOLOGY** has presented a paper on **IOT INDUSTRY PROTECTION USING ARDUINO** in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on **12th April, 2021**.

CONFERENCE CHAIR
Dr. T. Jayanthy, Principal