

AURDINO INTERFACING WITH REAL TIME APPLICATIONS

Submitted in partial fulfillment of the requirements for the
Professional Training –I of
Bachelor of Engineering Degree in
Electrical and Electronics Engineering

By

ISWARYA S (39140018)



DEPARTMENT OF ELECTRICAL AND ELECTRONICS
SCHOOL OF ELECTRICAL AND ELECTRONICS

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI – 600 119

ACKNOWLEDGEMENT

The satisfaction and elation that accompany the successful completion of any task would be incomplete without the mention of the people who have made it possible. It is our great privilege to express our gratitude and respect to all those who have guided me during the course of my Professional Training.

First and foremost, we would express our sincere gratitude to our beloved Founder Chancellor Col. Dr. JEPPIAAR, M.A., B.L., Ph.D., and Chancellor Dr. Mariazeena Johnson, B.E., M.B.A., M.Phil., Ph.D., I extend my sincere thanks to our President Dr. Marie Johnson, B.E., M.B.A., M.Phil., Ph.D., and for providing me the necessary facilities for the completion of the professional training. I also acknowledge our Vice Chancellor Dr. T. Sasipraba, M.E., Ph.D, and the Pro Vice Chancellor Dr. E.Logashanmugam, M.E,Ph.D, the Pro Vice Chancellor Dr.Wilson Aruni Ph.D., FAMPV, for their constant support and endorsement.

I would like to express our gratitude to our Registrar Dr. S. S. Rao, MBA,Ph.D. and Controller of Examinations Dr. Igini Sabasti Prabu M.E., Ph.D., for their valuable support offered to complete my professional training successfully.

I like to express our gratitude to Dr. V.Sivachidambaranathan M.E.,Ph.D., Head of the Department of Electrical and Electronics Engineering, Sathyabama Institute of Science and Technology for having been a constant source of support and encouragement for the completion of the professional training.

I would also like to express our sincere thanks to our internal guide who guided me in the preparation of the report. Also, I thank the Almighty and my Parents for supporting me in the completion of the professional

ISWARYA.S



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | Approved by AICTE

School of Electrical and Electronics
Department of Electrical and Electronics Engineering

CERTIFICATE OF PARTICIPATION

This is to certify that

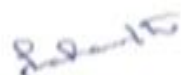
ISWARYA S

from

Sathyabama Institute of Science and Technology

has successfully completed the Value-Added Course on "Hands on Training in Arduino Interface using Tinkercad and Proteus for Real Time Applications" at Sathyabama Institute of Science and Technology, Chennai from 23rd August 2021 to 16th September 2021.


Dr. N. M. Nandhitha
Professor & Dean


Dr. V. Sivachidambaramanathan
Professor & Head/EEE

LIST OF CONTENT

SI.NO	TITLE	PAGE NO
	Abstract	8
1	MICROCONTROLLER	9
1.1	Introduction	9
1.2	Definition of Microcontroller	9
1.2.1	Key features of Microcontroller	10
1.3	8051 Architecture	11
1.4	Standard features of the 8051	11
1.5	Reset circuit and clock circuit	14
1.6	Family members of 8051	14
1.7	RAM Architecture	15
1.8	Central Processing Unit	18
2	ARDUINO	20
2.1	Definition	20
2.2	Why ARDUINO	21
2.3	ARDUINO UNO	22
2.4	ARDUINO Architecture	23
2.5	ARDUINO Technology	23
2.6	Basic functions of ARDUINO Technology	25
2.7	How to Program an ARDUINO	25
2.8	Advantages of ARDUINO Technology	26
3	SENSORS	27
3.1	Sensors Description	27

3.2	Infrared Sensor	27
3.2.1	Working principle of IR Sensor	27
3.2.2	Advantages of IR Sensor	28
3.3	Tilt Sensor	29
3.3.1	Working of Tilt Sensor	29
3.3.2	Applications of Tilt Sensor	30
3.4	Gas Sensor	31
3.4.1	What is 1 ppm is equal to?	31
3.4.2	Working of Gas sensor	32
3.5	Applications of Gas Sensor	33
4	SOFTWARE'S USED	34
4.1	Tinkercad Software	34
4.2	History of Tinkercad	34
4.3	What is Tinkercad?	35
4.4	Uses of Tinkercad	35
4.5	User Friendly Interface	36
4.6	Proteus Software	36
4.7	Introduction Of Proteus	37
4.8	Proteus Window	38
4.9	Uses Of Proteus	39
5	ARDUINO APPLICATIONS	40
5.1	ARDUINO Applications Using Tinkercad	40
5.2	ARDUINO Applications Using Proteus	40
5.3	Gas Leakage Sensor Detection Using Gas Sensor	40
5.4	Importance of Gas Leakage Detection	44
5.5	Traffic Light Control Using Proteus	45
5.6	Importance Of Traffic Light Management Signals	48

LIST OF FIGURES

Sl.NO	FIGURE NAMES	PAGE NO
1.1	8051 microcontroller	9
1.2	Microcontroller device & it's different sub units	10
1.3	8051 Architecture	11
1.4	Pin Configuration of 8051	12
1.5	Reset Circuit of 8051	14
1.6	RAM Architecture	16
1.7	Simplified Central Processing Unit with three Registers	19
2.1	ARDUINO Board	20
2.2	ARDUINO UNO	22
2.3	ARDUINO Architecture	23
2.4	Important Parts of AURDINO Board	24
2.5	ARDUINO Program	26
3.1	Different Types of Light Sensors	27
3.2	Working of IR Sensor	28
3.3	Tilt Sensor	29
3.4	Gas Sensor	31
3.5	Working Of Gas Sensor	32
4.1	Tinkercad Practice Page	34
4.2	Tinkercad logo	35

4.3	Introduction of Proteus	37
4.4	Proteus Window	38
5.1	Components of Gas Leakage Detection Sensor	40
5.2	Connections of Gas Leakage Detection Sensor	41
5.3	Traffic Light Control System	45

ABSTRACT

ARDUINO is an open-source electronics platform based on easy-to-use hardware and software. These days many people try to use the ARDUINO because it makes things easier due to the simplified version of C++ and the already made ARDUINO microcontroller that you can program, erase and reprogram at any given time. In this paper the hardware components used in the ARDUINO board, the software used to program it (ARDUINO board) with the guide on how to write and construct your own projects, and a couple of examples of an ARDUINO interfacing with real time applications are discussed.

Traffic light control systems are widely used to monitor and control the flow of automobiles through the junction of many roads. They aim to realize smooth motion of cars in the transportation routes. However, the synchronization of multiple traffic light systems at adjacent intersections is a complicated problem given the various parameters involved. Conventional systems do not handle variable flows approaching the junctions. In addition, the mutual interference between adjacent traffic light systems, the disparity of cars flow with time, the accidents, the passage of emergency vehicles, and the pedestrian crossing are not implemented in the existing traffic system. This leads to traffic jam and congestion. We propose a system based on ARDUINO that evaluates the traffic density using IR sensors and accomplishes dynamic timing slots with different levels. Moreover, a portable controller device is designed to solve the problem of emergency vehicles stuck in the overcrowded roads.

Gas leakage is a major problem in industries, residential premises and gas powered vehicles. The leakage if not detected may lead to explosion and cause severe damages to life and environment. The conventional leakage detection system uses on-site alarms for warning. The detection system use to detect the leakage and automatically initiate a warning. The experimental results show that the system is able to detect the leakage in less than a minute.

1. MICROCONTROLLER

1.1 Introduction

Circumstances that we find ourselves in today in the field of microcontrollers had their beginnings in the development of technology of integrated circuits. This development has made it possible to store hundreds of thousands of transistors into one chip. That was a prerequisite for production of microprocessors, and the first computers were made by adding external peripherals such as memory, input-output lines, timers and others. Further increasing the volume of the package resulted in creation of integrated circuits. These integrated circuits contained both processor and peripherals. That is how the first chip containing a microcomputer, or what would later be known as a microcontroller came about.

The first microcontroller 8051 was developed by Intel Corporation in the year 1981. It was called a “System on a chip”. Intel refers to it as MCS-51 now.



Fig 1.1: 8051 microcontroller

1.2 Definition of a microcontroller

Microcontroller, as the name suggests, are small controllers. It is referred to as System on a Chip (SOC) microcontroller because it is a Chip circuit/integrated circuit that holds many components of a computer together on a single chip. These Components includes a CPU, memory, input output ports (I/O ports), timers and secondary storage.

For example, the remote control you are using probably has microcontrollers inside that do decoding and other controlling functions. They are also used in automobiles, washing machines, microwave ovens, where automation is needed.

1.2.1 The key features of microcontrollers

- High Integration of Functionality.
- Microcontrollers sometimes are called single-chip computers because they have on-chip memory and I/O circuitry and other circuitries that enable them to function as small standalone computers without other supporting circuitry.
- Field Programmability, Flexibility.
- Microcontrollers often use EEPROM or EPROM as their storage device to allow field programmability so they are flexible to use. Once the program is tested to be correct then large quantities of microcontrollers can be programmed to be used in embedded systems.
- Easy to Use Assembly language is often used in microcontrollers and since they usually follow RISC architecture, the instruction set is small. The development package of microcontrollers often includes an assembler, a simulator, a programmer to "burn" the chip and a demonstration board. Some packages include a high level language compiler such as a C compiler and more sophisticated libraries.

Most microcontrollers will also combine other devices such as

- A Timer module to allow the microcontroller to perform tasks for certain time periods.
- A serial I/O port to allow data to flow between the microcontroller and other devices such as a PC or another microcontroller.
- An ADC to allow the microcontroller to accept analogue input data for processing.

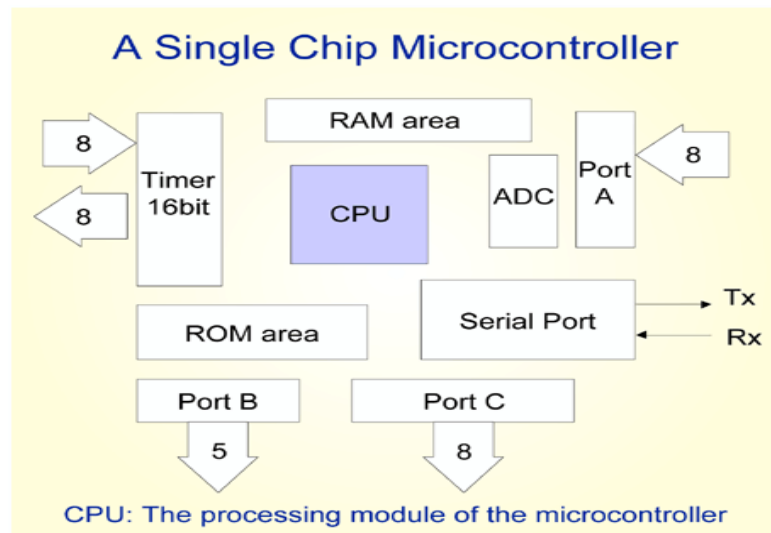


Fig 1.2: Microcontroller device and its different subunits

1.3 8051 Architecture

The 8051 is a high performance single chip computer intended for use in sophisticated real time applications such as instrumentation, industrial control and computer peripherals. It provides extra features like interrupts, bit address ability and an enhanced set of instructions, which makes the chip very powerful and cost effective.

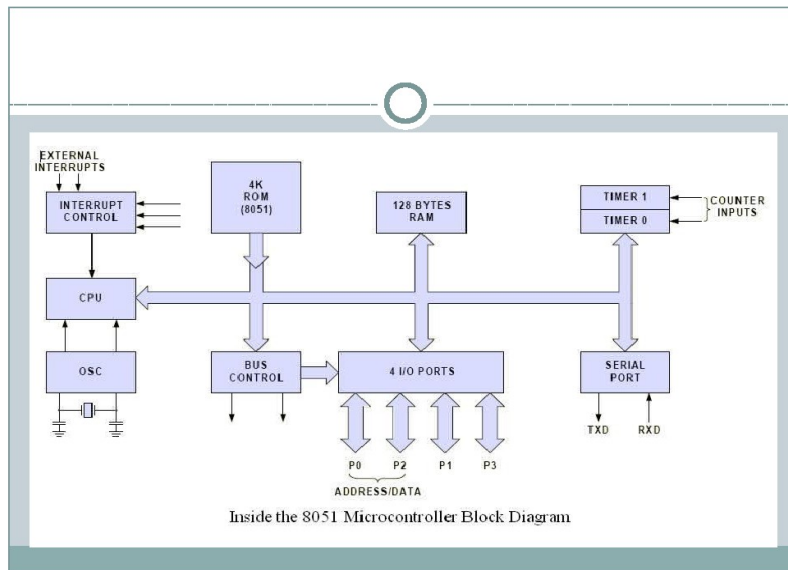


Fig 1.3: 8051 Architecture

1.4 Standard features of the 8051

- 8 bit data path and ALU
- On chip flash memory
- 4K X 8 ROM – Program Memory
- 128 X 8 RAM – Data Memory
- Multiple 16 – bit Timer/Counter
- Full duplex UART (serial port)
- On chip clock oscillator
- 32 I/O pins

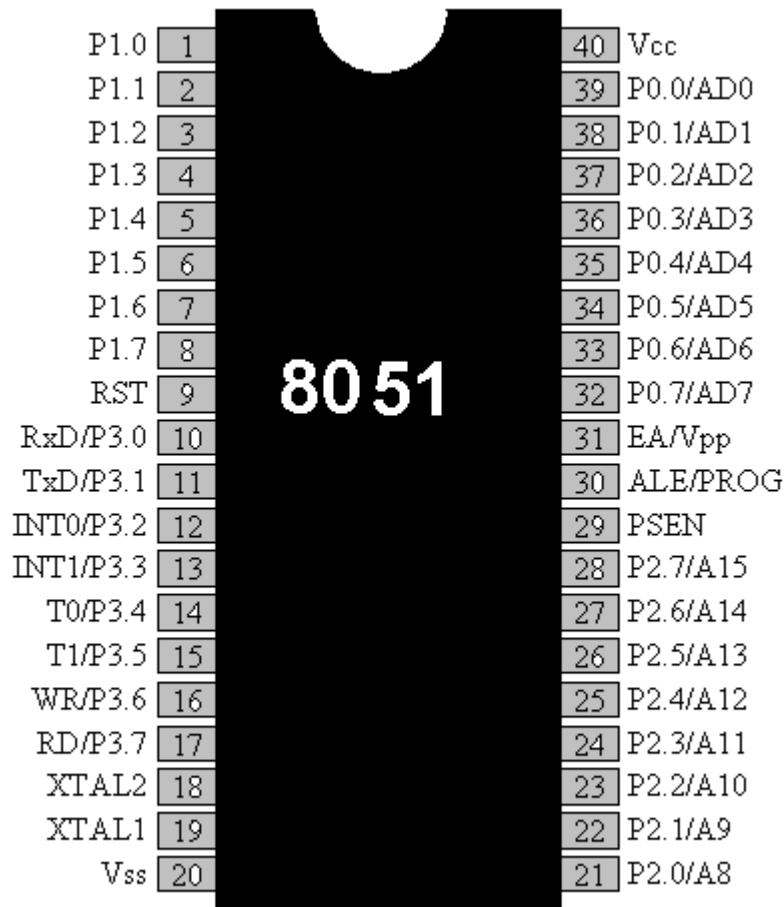


Fig 1.4: Pin configuration of 8051

Port 0 (pin 32-39)

- Dual purpose I/O port.
- In min. component design, it is used as a general purpose I/O port. In
- Larger designs with external memory, it becomes a multiplexed data bus.
- Low byte of address bus, strobe by ALE.
- 8-bit instruction bus, stored by PSEN.
- 8-bit data bus, stored by WR and RD.

Port 1 (pin 1-8)

- As an I/O port: Standard bi-directional port for interfacing to external.
- Devices as required for I/O.
- Alternate functions: Only on some derivative

Port 2 (pin 21-28)

- Dual purpose I/O port.
- As an I/O port: Standard bi-directional general purpose I/O port.
- Alternate functions: High byte of address bus for external program and
- Data memory accesses

Port 3 (pin 10-17)

- Is an 8-bit bi-directional I/O port with internal Pull-ups/ It also serves the functions of various special features of the 80C51

Alternate uses of Port- 3 pins:

PINS	ALTERNATE USE	SFR
P3.0 RXD	Serial data input	SBUF
P3.1 TXD	Serial data output	SBUF
P3.2 INT0	External Interrupt 0	TCON 1
P3.3 INT1	External Interrupt 1	TCON.3
P3.4 T0	External Timer 0 I/P	TMOD
P3.5 T1	External Timer 1 I/P	TMOD
P3.6 WR	External Memory Write pulse	-
P3.7 RD	External Memory read pulse	-

Reset: It resets total 8051 micro controller.

XTAL1 & XTAL2: To connect the crystal oscillator for the 8051 crystal oscillator of 11.0592 MHZ is connected between these pins.

AL: Address latch enable which is used to access the address locations from external memory.

PSEN: Program store enable- which is used for storing programming code into the external memory.

EA: External Access- 64 KB of ROM is the limit for external memory. Supply and Ground pins: Pin 40 is for +5V and pin 20 is for GND.

1.5 Reset circuit and clock circuit

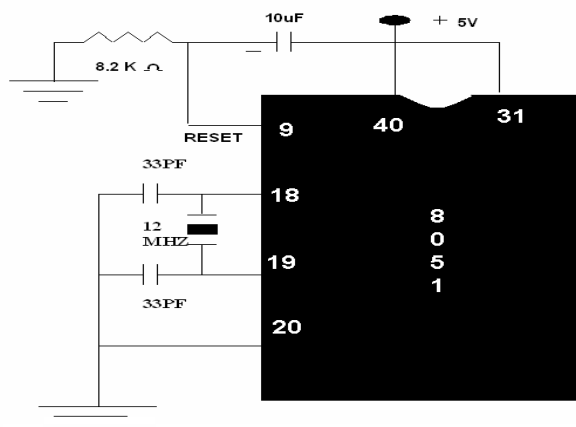


Fig 1.5: Reset circuit of 8051

1.6 Family members of 8051

Table 1.1: family members of 8051

Features	8051	8031	8052	8032
ROM	4K	0K	8K	0K
RAM(bytes)	128	128	256	256

Timers	2	2	3	3
I/O Pins	32	32	32	32
Serial Port	1	1	1	1
Interrupt Sources	6	6	7	7
ROM	4K	0K	8K	0K
RAM(bytes)	128	128	256	256
Timers	2	2	3	3
I/O Pins	32	32	32	32
Serial Port	1	1	1	1
Interrupt Sources	6	6	7	7

1.7 RAM Architecture

The 8051 has a bank of 128 bytes of Internal RAM. This Internal RAM is found on-chip on the 8051 so it is the fastest RAM available, and it is also the most flexible in terms of reading, writing, and modifying its contents. Internal RAM is volatile, so when the 8051 is reset this memory is cleared. The 128 bytes of internal ram is subdivided as shown on the memory map. The first 8 bytes (00h - 07h) are "register bank 0". These alternative register banks are located in internal RAM in addresses 08h through 1Fh. Bit memory actually resides in internal RAM, from addresses 20h through 2Fh. The 80 bytes remaining of Internal RAM, from addresses 30h through 7Fh, may be used by user variables that need to be accessed frequently or at high-speed. This area is also utilized by the microcontroller as a storage area for the operating stack.

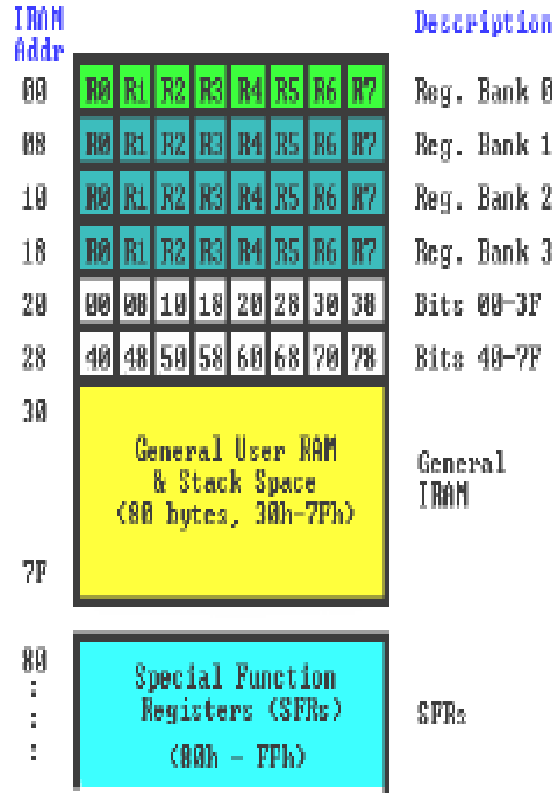


Fig 1.6: RAM architecture

Register Banks

The 8051 uses 8 "R" registers which are used in many of its instructions. These "R" registers are numbered from 0 through 7 (R0, R1, R2, R3, R4, R5, R6, and R7). These registers are generally used to assist in manipulating values and moving data from one memory location to another. The concept of register banks adds a great level of flexibility to the 8051.

Bit Memory

The 8051 being a communication oriented microcontroller gives the user the ability to access a number of bit variables. These variables may be either 1 or 0. There are 128 bit variables available to the user, numbered 00h through 7Fh. The user may make use of these variables with commands such as SETB and CLR. It is important to note that Bit Memory is really a part of Internal RAM. In fact, the 128 bit variables occupy the 16 bytes of Internal RAM from 20h through 2Fh.

Special Function Register (SFR) Memory

Special Function Registers (SFRs) are areas of memory that control specific functionality of the 8051 processor. For example, four SFRs permit access to the 8051's 32 input/output lines. Another SFR allows a program to read or write to the 8051's serial port. SFR is a part of Internal Memory. This is not the case. When using this method of memory access (it's called direct address), any instruction that has an address of 00h through 7Fh refers to an Internal RAM memory address; any instruction with an address of 80h through FFh refers to an SFR control register.

Accumulator

The Accumulator, as its name suggests, is used as a general register to accumulate the results of a large number of instructions. It can hold an 8-bit (1-byte) value and is the most versatile register

"R" registers

The "R" registers are a set of eight registers that are named R0, R1, etc. up to and including R7. These registers are used as auxiliary registers in many operations.

"B" Register

The "B" register is very similar to the Accumulator in the sense that it may hold an 8-bit (1-byte) value. The "B" register is only used by two 8051 instructions: MUL AB and DIV AB.

Data Pointer (DPTR)

The Data Pointer (DPTR) is the 8051's only user-accessible 16-bit (2-byte) register. The Accumulator, "R" registers, and "B" register are all 1-byte values. DPTR, as the name suggests, is used to point to data. It is used by a number of commands which allow the 8051 to access external memory.

Program Counter (PC)

The Program Counter (PC) is a 2-byte address which tells the 8051 where the next instruction to execute is found in memory. When the 8051 is initialized PC always starts at 0000h and is incremented each time an instruction is executed.

Stack Pointer (SP)

The Stack Pointer, like all registers except DPTR and PC, may hold an 8-bit (1-byte) value. The Stack Pointer is used to indicate where the next value to be removed from the stack should be interrupts.

An interrupt is a special feature which allows the 8051 to provide the illusion of "multitasking," although in reality the 8051 is only doing one thing at a time.

Timers

Timers are one of the categories of hardware time delays. Time delays are used to keep a system into halting System or sleepy mode. We have two timers - timer0, timer1. Hardware time delays are used to generate exact time delays.

1.8 Central processing unit

Let add 3 more memory locations to a specific block that will have a built in capability to multiply, divide, subtract, and move its contents from one memory location onto another. The part we just added in is called "central processing unit" (CPU). Its memory locations are called registers. Registers are therefore memory locations whose role is to help with performing various mathematical operations or any other operations with data wherever data can be found. Look at the current situation. We have two independent entities (memory and CPU) which are interconnected, and thus any exchange of data is hindered, as well as its functionality. If, for example, we wish to add the contents of two memory locations and return the result again back to memory, we would need a connection between memory and CPU. Simply stated, we must have some "way" through data goes from one block to another.

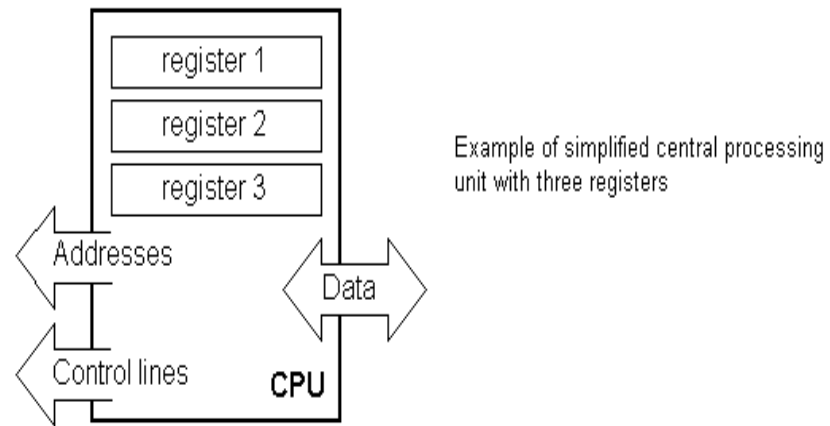


Fig 1.7: Simplified central processing unit with three register

2. ARDUINO

2.1 Definition

ARDUINO is an open-source electronics platform based on easy-to-use hardware and software. ARDUINO boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the ARDUINO programming language (based on Wiring), and the ARDUINO Software (IDE), based on Processing.

ARDUINO was born at the IVREA Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the ARDUINO board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IOT applications, wearable, 3D printing, and embedded environments. All ARDUINO boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software tool is open-source, and it is growing through the contributions of users worldwide.

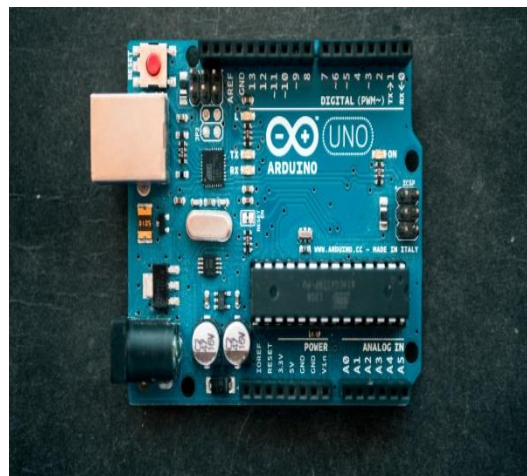


Fig 2.1: ARDUINO Board

2.2 Why ARDUINO?

Thanks to its simple and accessible user experience, ARDUINO has been used in thousands of different projects and applications. The ARDUINO software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Faire, for example. ARDUINO is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step by step instructions of a kit, or sharing ideas online with other members of the ARDUINO community. ARDUINO also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

- Inexpensive - ARDUINO boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the ARDUINO module can be assembled by hand, and even the pre-assembled ARDUINO modules cost less than \$50
- Cross-platform - The ARDUINO Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- Simple, clear programming environment - The ARDUINO Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the ARDUINO IDE works.
- Open source and extensible software - The ARDUINO software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from ARDUINO to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your ARDUINO programs if you want to.
- Open source and extensible hardware - The plans of the ARDUINO boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the

module, extending it and improving it. Even relatively inexperienced users can build the broad board version the module in order to understand how it works and save money.

2.3 ARDUINO UNO

A typical example of the ARDUINO board is ARDUINO UNO. It includes an ATmega328 microcontroller and it has 28-pins.

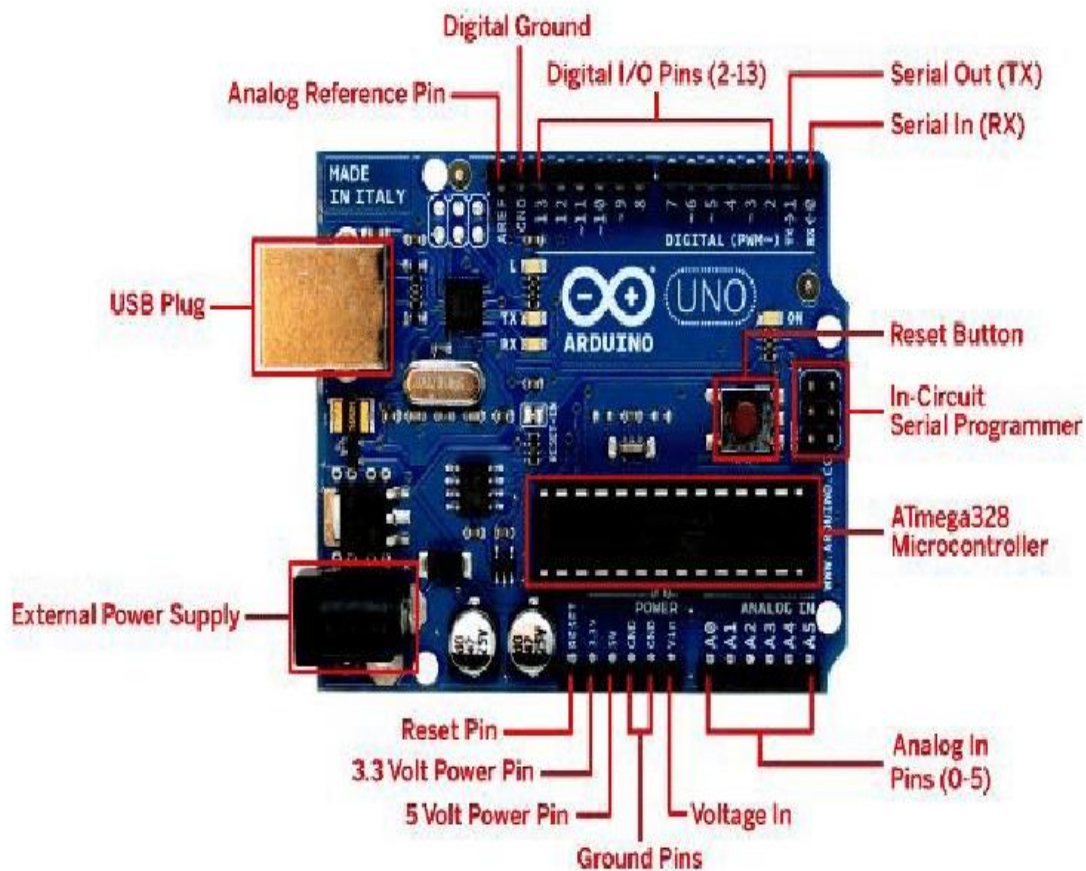


Fig 2.2: ARDUINO UNO

2.4 ARDUINO Architecture

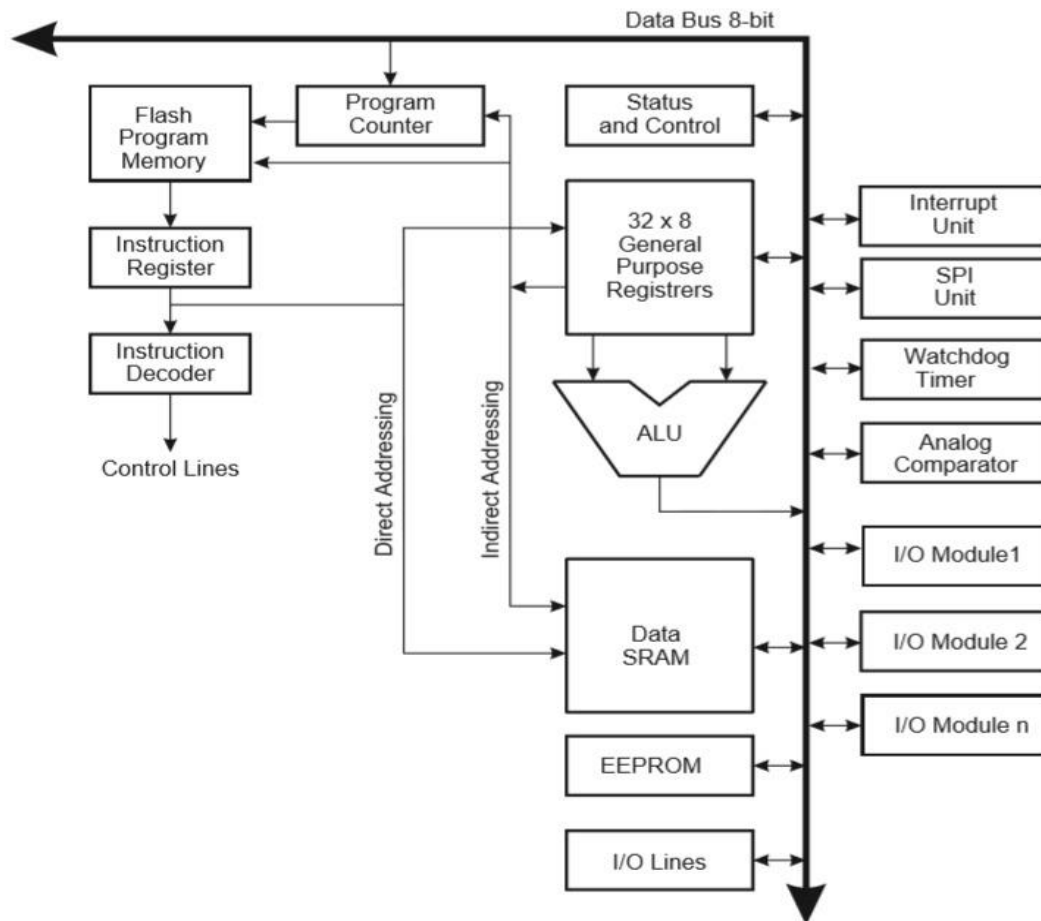


Fig 2.3: ARDUINO architecture

Basically, the processor of the ARDUINO board uses the Harvard architecture where the program code and program data have separate memory. It consists of two memories such as program memory and data memory. Wherein the data is stored in data memory and the code is stored in the flash program memory. The ATmega328P microcontroller has 32kb of flash memory, 2kb of SRAM 1kb of EPROM and operates with a 16MHz clock speed.

2.5 ARDUINO Technology

A typical example of the ARDUINO board is ARDUINO UNO. It includes an ATmega328P microcontroller and it has 28-pins.

The pin configuration of the ARDUINO Uno board is shown in the above. It consists of 14-digital input output pins. Wherein 6 pins are used as pulse width modulation outputs and 6 analog inputs, a USB connection, a power jack, a 16MHz crystal oscillator, a reset button, and an ICSP header. ARDUINO board can be powered either from the personal computer through a USB or external source like a battery or an adaptor. This board can operate with an external supply of 7-12V by giving voltage reference through the IOREf pin or through the pinVin.

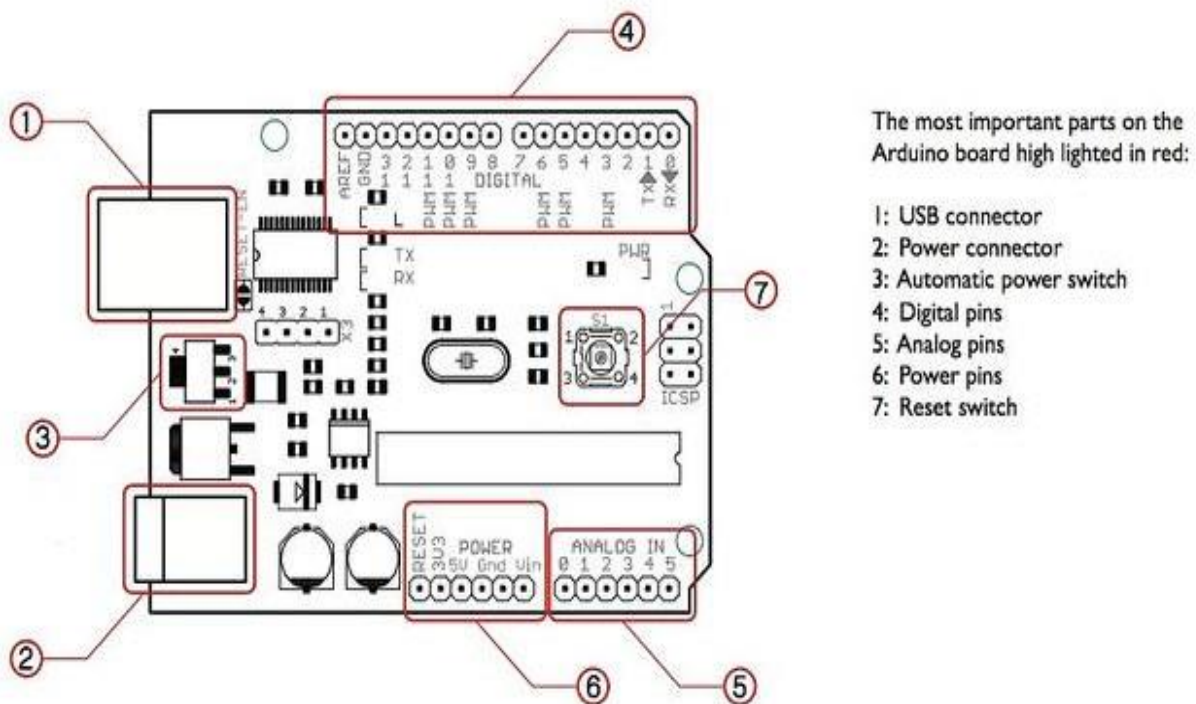


Fig 2.4: Important parts of ARDUINO Board

Digital I/Ps:

It comprises of 14-digital I/O pins, each pin take up and provides 40mA current. Some of the pins have special functions like pins 0 & 1, which acts as a transmitter and receiver respectively. For serial communication, pins-2 & 3 are external interrupts 3,5,6,9,11 pins delivers PWM o/p and pin-13 is used to connect LED.

Analog inputs: It has 6-analog I/O pins, each pin provide a 10 bits resolution.

Aref: This pin gives a reference to the analog i/ps.

Reset: When the pin is low, then it resets the microcontroller.

2.6 Basic Functions of ARDUINO Technology

- Digital read pin reads the digital value of the given pin.
- Digital write pin is used to write the digital value of the given pin.
- Pin mode pin is used to set the pin to I/O mode.
- Analog read pin reads and returns the value.
- Analog write pin writes the value of the pin.
- Serial. Begins pin sets the beginning of serial communication by setting the rate of bit.

2.7 How to program an ARDUINO?

The main advantage of the ARDUINO technology is, you can directly load the programs into the device without the need of a hardware programmer to burn the program. This is done because of the presence of the 0.5KB of boot loader that allows the program to be dumped into the circuit. The ARDUINO tool window contains a toolbar with a various buttons like new, open, verify, upload and serial monitor. And additionally it comprises of a text editor (employed to write the code), a message space (displays the feedback) like showing the errors, the text console, that displays the o/p & a series of menus just like the file, tool menu & edit.

- Programming into the ARDUINO board is called as sketches. Each sketch contains of three parts such as Variables Declaration, Initialization and Control code. Where, Initialization is written in the setup function and Control code is written in the loop function.
- The sketch is saved with .into and any operation like opening a sketch, verifying and saving can be done using the tool menu.
- The sketch must be stored in the sketchbook directory.
- Select the suitable board from the serial port numbers and tools menu.
- Select the tools menu and click on the upload button, then the boot loader uploads the code on the microcontroller.

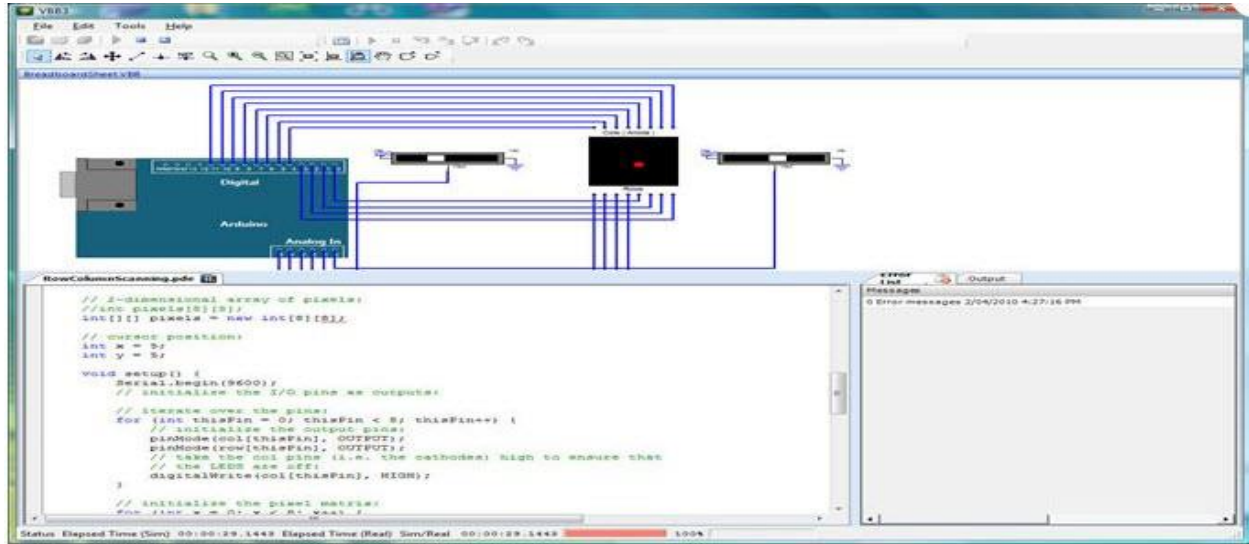


Fig 2.5: ARDUINO Program

2.8 Advantages of ARDUINO Technology

- It is cheap
- It comes with an open supply hardware feature that permits users to develop their own kit
- The software of the ARDUINO is well-suited with all kinds of in operation systems like Linux, Windows, and Macintosh, etc.
- It also comes with open supply software system feature that permits tough software system developers to use the ARDUINO code to merge with the prevailing programming language libraries and may be extended and changed.
- For beginners, it is very simple to use.

3. SENSORS

3.1 Sensors Description

In the broadest definition, a **sensor** is a device, module, machine, or subsystem whose purpose is to detect events or changes in its environment and send the information to other electronics, frequently a computer processor. A sensor is always used with other electronics.

Sensors are used in everyday objects such as touch-sensitive elevator buttons (tactile sensor) and lamps which dim or brighten by touching the base, besides innumerable applications of which most people are never aware.

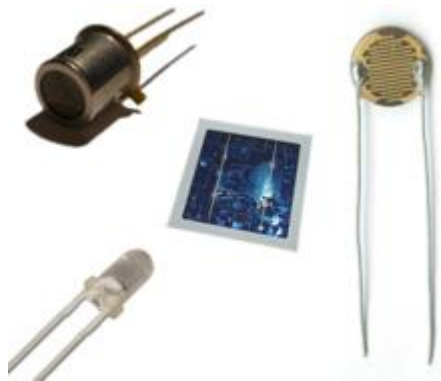


Fig 3.1: Different types of light sensors

3.2 Infrared Sensor

An infrared sensor is an electronic device that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. These types of sensors measures only infrared radiation, rather than emitting it that is called as a passive IR sensor

3.2.1 Working principle IR Sensor

The principle of an IR sensor working as an Object Detection Sensor can be explained using the following figure. An IR sensor consists of an IR LED and an IR Photodiode; together they are called as Photo – Coupler or Opto - Coupler.

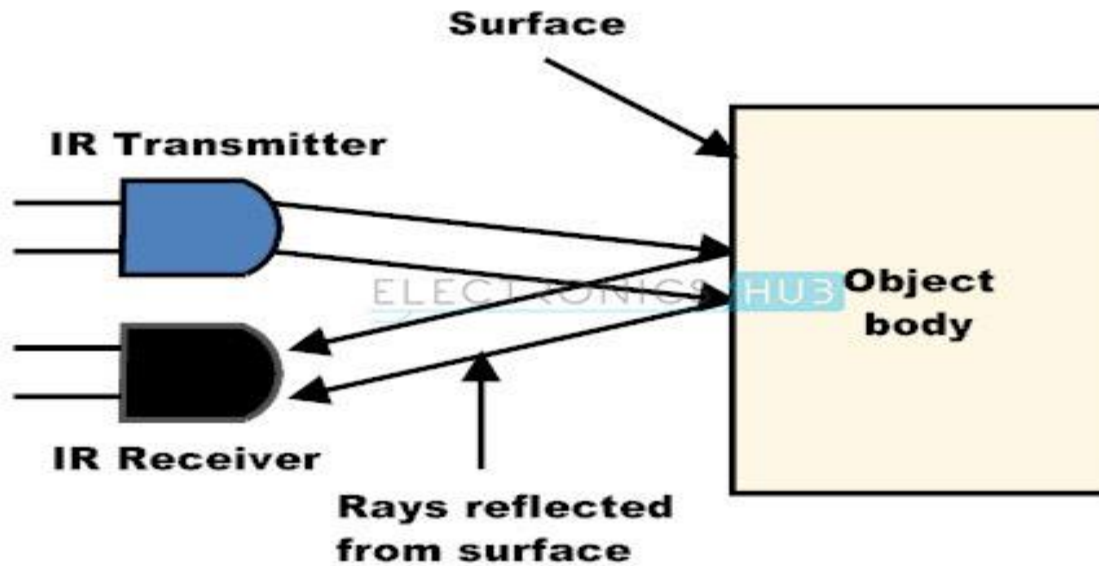


Fig 3.2: Working of IR sensor

When the IR transmitter emits radiation, it reaches the object and some of the radiation reflects back to the IR receiver. Based on the intensity of the reception by the IR receiver, the output of the sensor is defined.

3.2.2 Advantages of IR Sensor

- Low power usage.
- Simple design.
- Convenient features.
- Signals are not noticeable by human eye.
- The IR radiation in the electromagnetic range can be found in the regions of the visible & microwave.

3.3 Tilt sensor

A Tilt Sensor switch is an electronic device that detects the orientation of an object and gives its output High or Low accordingly. Basically, it has a mercury ball inside it which moves and makes the circuit. So tilt sensor can turn on or off the circuit based on the orientation.



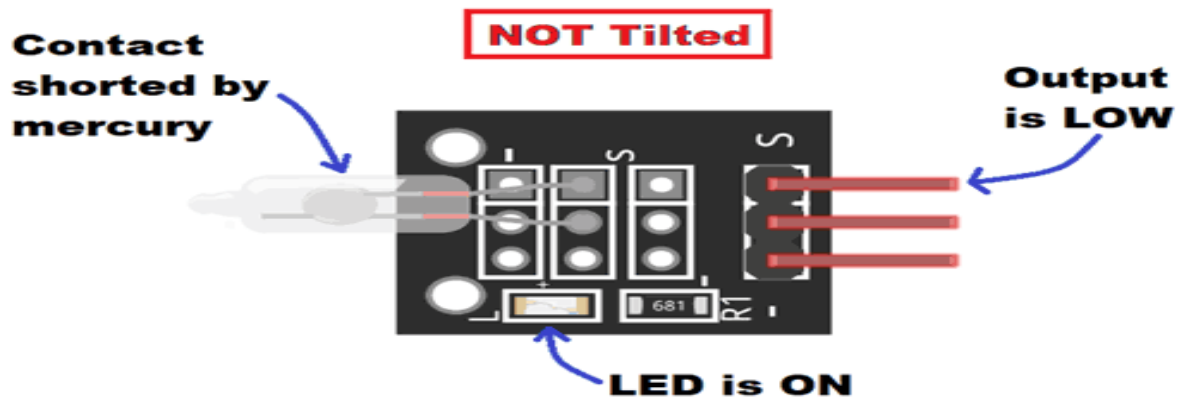
Fig 3.3: Tilt sensor

This is a Mercury switch based tilt sensor module that gives high at its output pin when tilted. It requires a 5V of DC input. It's a three-terminal device consist of input, ground, and output. It has a glass tube consist of two electrode and liquid mercury ball. The liquid mercury ball closes and opens the circuit when inclined in a particular direction.

3.3.1 Working of Tilt Sensor

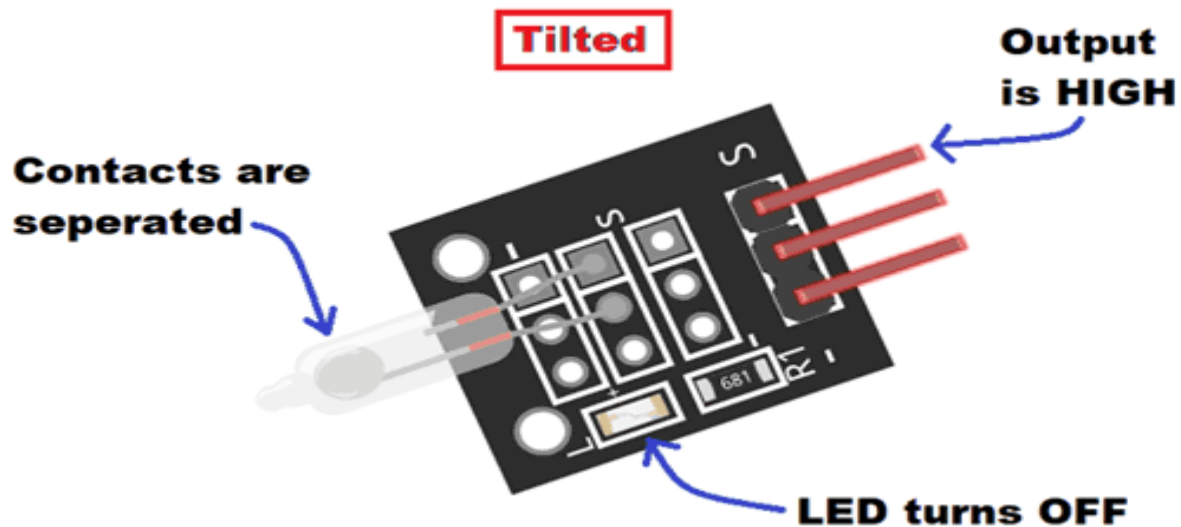
CASE 1: NOT TILTED

Initially, when it is in NOT tilted position as shown in the image below, it gives LOW output because of the liquid mercury complete the circuit by connecting the two electrodes. When the output is LOW on-board LED remain ON.



CASE 1: TILTED

When it is inclined in a particular direction or angle, the liquid mercury breaks the contact between the metal electrodes and the circuit gets open. Hence, we get HIGH output in this condition and the onboard LED turns off.



3.3.2 Applications of Tilt Sensor

- Cameras.
- Video cameras.
- Aircraft flight controls.

- Construction equipment.
- Robotic Technology.
- Automobile air bags.
- Videos game controller.
- Studying human movement.

3.4 Gas Sensor

MQ2 is one of the commonly used gas sensors in MQ sensor series. It is a Metal Oxide Semiconductor (MOS) type Gas Sensor also known as chemiresistors as the detection is based upon change of resistance of the sensing material when the Gas comes in contact with the material. Using a simple voltage divider network, concentrations of gas can be detected.



Fig 3.4: Gas sensor

MQ2 Gas sensor works on 5V DC and draws around 800mW. It can detect LPG, Smoke, Alcohol, Propane, Hydrogen, Methane and Carbon Monoxide concentrations anywhere from 200 to 10000ppm.

3.4.1 What is 1ppm equal to?

When measuring gases like carbon dioxide, oxygen, or methane, the term concentration is used to describe the amount of gas by volume in the air. The 2 most common units of measurement are parts-per-million, and percent concentration.

Parts-per-million (abbreviated ppm) is the ratio of one gas to another. For example, 1,000ppm of CO means that if you could count a million gas molecules, 1,000 of them would be of carbon monoxide and 999,000 molecules would be some other gases.

3.4.2 How does a gas sensor work?

When tin dioxide (semiconductor particles) is heated in air at high temperature, oxygen is adsorbed on the surface. In clean air, donor electrons in tin dioxide are attracted toward oxygen which is adsorbed on the surface of the sensing material. This prevents electric current flow.

In the presence of reducing gases, the surface density of adsorbed oxygen decreases as it reacts with the reducing gases. Electrons are then released into the tin dioxide, allowing current to flow freely through the sensor.



Fig 3.5: Working of Gas sensor

3.4.3 Applications of Gas sensor

- MQ2 gas sensor can be used to detect the presence of LPG, Propane and Hydrogen, also could be used to detect Methane and other combustible steam.
- It is low cost and suitable for different application.
- Sensor is sensitive to flammable gas and smoke.
- Smoke sensor is given 5 volt to power it.

4. SOFTWARE'S USED

4.1 Tinkercad Software

Tinkercad is a free, easy-to-use app for 3D design, electronics, and coding. It's used by teachers, kids, hobbyists, and designers to imagine, design, and make anything! Users can quickly learn how to use the Tinkercad software through basic tutorial lessons that focus on the basics of the Tinkercad tool.

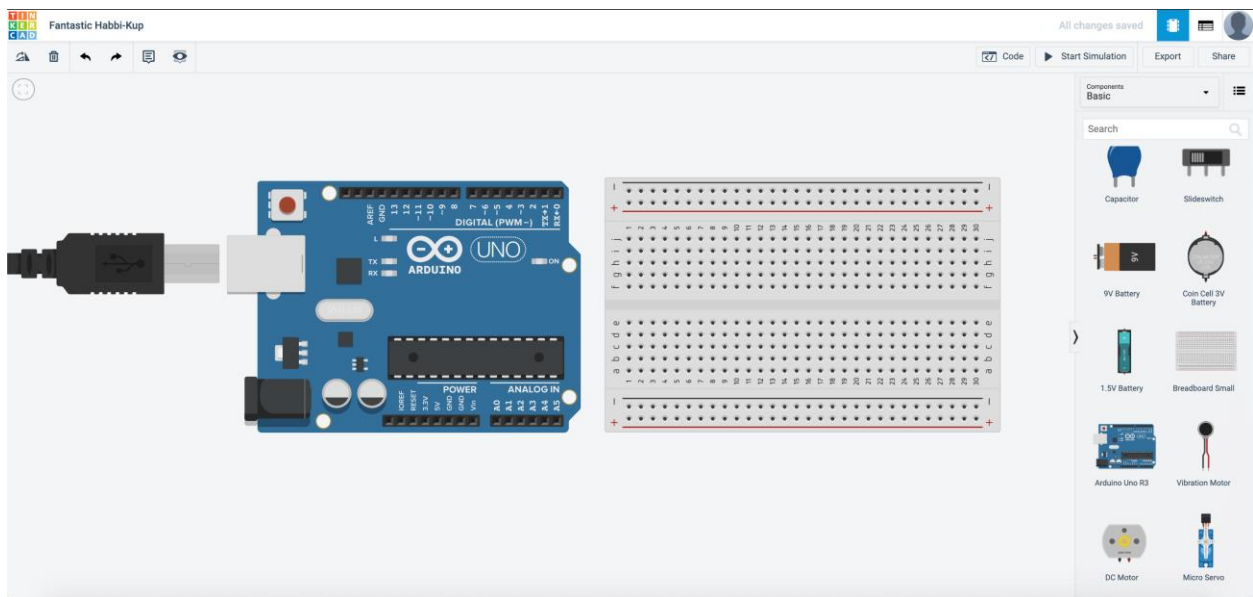


Fig 4.1: Tinkercad practice page

4.2 History

Tinkercad was founded by former Google engineer Kai Backman and his cofounder Mikko Mononen, with a goal to make 3D modeling, especially the design of physical items, accessible to the general public, and allow users to publish their designs under a Creative Commons license. In 2011, the tinkercad.com website was launched as a web-based 3D modeling tool for WebGL-enabled browsers, and in 2012 the company moved its headquarters to San Francisco. By 2012 over 100,000 3D designs had been published by users.

In May 2013, Autodesk announced at a Maker Faire that they would acquire Tinkercad.

In March 2017, Autodesk recommended users of the soon to be retired 123D Sculpt migrate to Tinkercad (or Maya LT). In May, Autodesk discontinued its 123D Circuits (Circuits.io) "Electronics Lab". The program's features were merged into Tinkercad.

4.3 What is Tinkercad?

Tinkercad is a computer - aided design (CAD) software that helps individuals and businesses in the manufacturing and advertising industry. It also caters to healthcare and architectural companies. The freeware allows for file editing, which enables you to recreate existing designs. You and your designers can also upload and alter old 2D and 3D plans. With Tinkercad, you can make new models using the software's modifiable shapes and objects. Tinkercad then gives you the freedom to combine shapes, add movements, and stack blocks. With these actions, you can create a 3D design that is essential in prototypes. You can also develop circuits using Tinkercad lines and customizable shapes. The code blocks then help you in devising marketing campaigns through GIFs. Lastly, Tinkercad offers design presets to help you get started in the software.



Fig 4.2: Tinkercad logo

4.4 Uses of Tinkercad

Tinkercad is an amazingly powerful easy-to-use tool for creating digital designs that are ready to be 3D printed into super-cool physical objects. You will be guided through the 3D design

process via easy hands-on "Lessons" that teach you the basics of Tinkercad before moving on to more complex modeling techniques.

4.5 User-friendly interface

This platform provides a user-friendly interface with options such as 3D design, circuits, and code blocks. Tinkercad also allows for a 2D and 3D input, which permits you to integrate other models into your new plans. With Tinkercad, you can view the design of your prototype from almost any angle. It also enables you to customize existing presets to create a unique design for your product.

4.6 Proteus Software

Proteus is a simulation and design software use for lab center Electronic for Electrical and Electronic design circuits. Proteus is the best simulation software for various with microcontroller. It is mainly common software because of availability of almost all microcontrollers in it. So it is a handy tool test programs and embedded designs for electronic hobbyist. You can see your programming in Proteus Software.

After simulating your circuit in Proteus you can easily make PCB design with it. In short you can say that it could be all in one package. It is a software containing Schematic, Simulation and PCB designing.

➤ **ISIS:**

ISIS abbreviation is **Intelligent Schematic Input System**. ISIS is used to draw schematic and simulate the circuit in real time. The Simulation allows human access during run time, providing real time simulation.

➤ **ARES:**

ARES Stands for **Advance Routing and Editing Software**. ARES is used for PCB designing. It has the feature of viewing 3D view of the designed PCB along with components. The designer can also develop 2D drawings in it.

4.7 Introduction to Proteus

- Proteus is used to simulate, design and drawing of electronic circuits. It was invented by the Lab center electronic.
- By using Proteus you can make two-dimensional circuits designs as well.
- With the use of this engineering software, you can construct and simulate different electrical and electronic circuits on your personal computers or laptops.
- There are numerous benefits to simulate circuits on Proteus before make them practically.
- Designing of circuits on the Proteus takes less time than practical construction of the circuit.
- The possibility of error is less in software simulation such as loose connection that takes a lot of time to find out connections problems in a practical circuit.
- Circuit simulations provide the main feature that some components of circuits are not practical then you can construct your circuit on Proteus.
- There is zero possibility of burning and damaging of any electronic component in Proteus.
- The electronic tools that are very expensive can easily get in Proteus such as an oscilloscope.
- Using Proteus you can find different parents of circuits such as current, a voltage value of any component and resistance at any instant which is very difficult in a practical circuit.

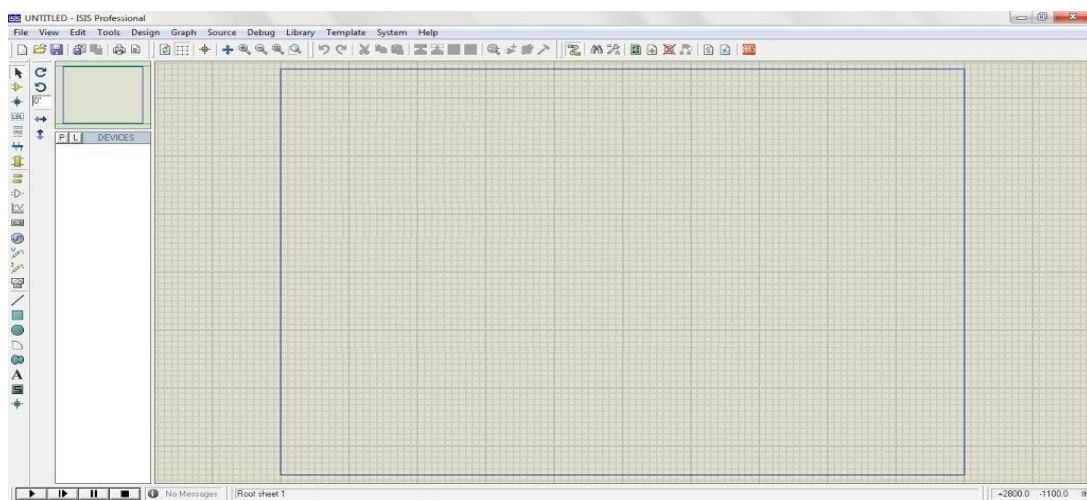


Fig 4.3: Introduction of Proteus

4.8 Proteus Window

Editing Window

In the above figure, you can see that the dotted portion is called an editing window. This is a drawing portion of Proteus where you simulate your engineering circuits and projects.

Overview Window

In the overview window, you see the complete view of your complete design.

Object Selector

This section has 2 buttons P and E. P is used to select different components and shown in this box. The E button is for edit something for example you want to vary any value of components than you can use this edit button.

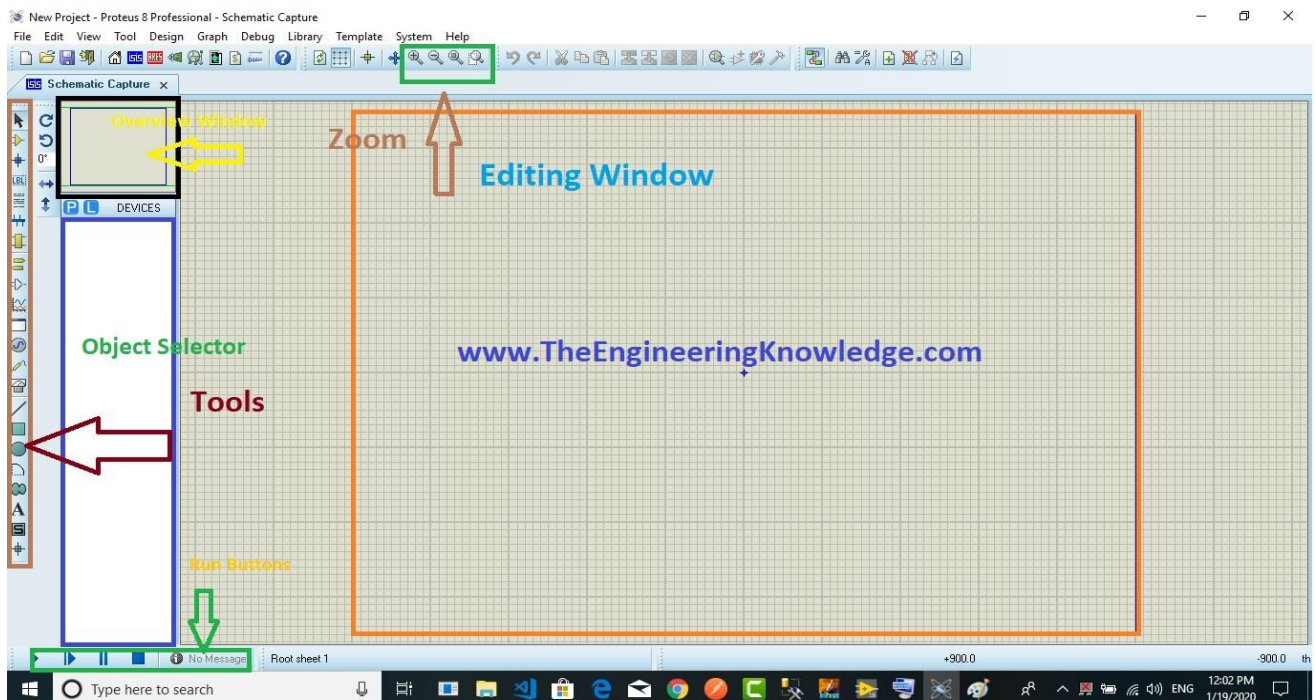


Fig 4.4: Proteus window

Zoom Option

By using this option you can easily zoom in and zoom out your layout and can observe complete simulation very clearly.

Tool Option

By using this option you can select different devices like voltmeter, ammeter, oscilloscope, etc.

Run Buttons

At the left bottom there are 4 buttons Run, stop, pause and stop. These buttons are like the remote control and on and off your circuit.

4.9 Uses of Proteus:

- The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation.
- The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

5. ARDUINO Applications

5.1 ARDUINO applications using Tinkercad

- Gas leakage detection using gas sensor
- Home automation

5.2 ARDUINO applications using Proteus

- Traffic light control

5.3 Gas leakage detection using gas sensor

Components Used:

- Breadboard
- ARDUINO Uno
- LCD Display
- LED's
- Gas Sensor
- Buzzer
- Resistors

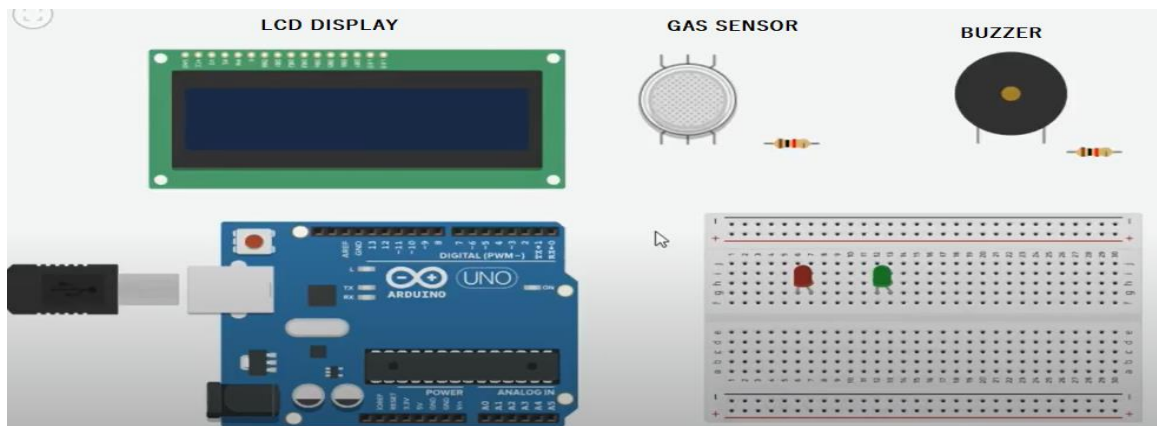


Fig 5.1: Components

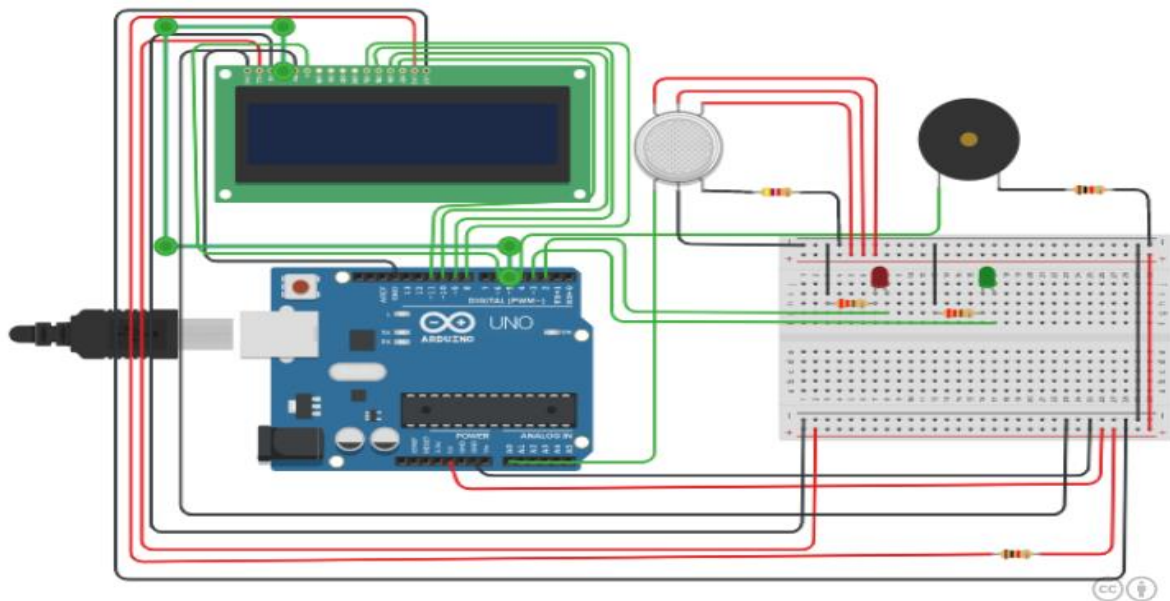


Fig 5.2: Connections

PROGRAM -1

```

int redled = 10;

int greenled = 9;

int buzzer = 7;

int sensor = A0;

int sensorThresh = 400;

void setup()
{
  pinMode(redled, OUTPUT);
  pinMode(greenled,OUTPUT);
  pinMode(buzzer,OUTPUT);
  pinMode(sensor,INPUT);
  Serial.begin(9600);
}

```

```

void loop()
{
  int analogValue = analogRead(sensor);
  Serial.print("Sensor Value: ");
  Serial.println(analogValue);
  if(analogValue>sensorThresh)
  {
    digitalWrite(redled,HIGH);
    digitalWrite(greenled,LOW);
    tone(buzzer,1000,10000);
  }
  else
  {
    digitalWrite(greenled,HIGH);
    digitalWrite(redled,LOW);
    noTone(buzzer);
  }
}

```

PROGRAM 2 (TO DISPLAY THE ALERT MESSAGE IN LCD)

```

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

int redled = 10;
int greenled = 9;
int buzzer = 7;

```

```

int sensor = A0;

int sensorThresh = 400;


void setup()
{
  pinMode(redled, OUTPUT);
  pinMode(greenled,OUTPUT);
  pinMode(buzzer,OUTPUT);
  pinMode(sensor,INPUT);
  Serial.begin(9600);
  lcd.begin(16,2);
}

void loop()
{
  int analogValue = analogRead(sensor);
  Serial.print("Sensor Value: ");
  Serial.println(analogValue);
  if(analogValue>sensorThresh)
  {
    digitalWrite(redled,HIGH);
    digitalWrite(greenled,LOW);
    tone(buzzer,1000,10000);
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("ALERT");
  }
}

```

```

    delay(1000);

    lcd.clear();

    lcd.setCursor(0,1);

    lcd.print("EVACUATE");

    delay(1000);

}

else

{

    digitalWrite(greenled,HIGH);

    digitalWrite(redled,LOW);

    noTone(buzzer);

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("SAFE");

    delay(1000);

    lcd.clear();

    lcd.setCursor(0,1);

    lcd.print("ALL CLEAR");

    delay(1000);

}

}

```

5.4 Importance of gas leakage detection

Gas leak detection services are necessary mainly for health reasons. Combustible gases can start fires or cause explosions in homes and workplaces, injuring family members, friends, or workers. Gas leaks can also cause health issues in the long term from gradual exposure. Some gas leaks are even silent killers.

5.5 Traffic light control using Proteus

ARDUINO is the most popular open-source microcontroller board used for many electronics projects. Traffic light controller by using ARDUINO is a simple electronic project. Nowadays, everyone prefers a personal vehicle. Hence, the number of vehicles on the road is increasing continuously, which results in traffic jams.

Traffic light controller helps to manage the traffic and to maintain proper traffic management. These systems are placed at the intersections of the road or at the crossings to avoid congestions and accidents. The systems indicate to the driver by using different colors of light. Therefore it is simple to avoid congestion at the intersections.

This project is the traffic light controller by using ARDUINO. It is an electronic project by which we will get knowledge about traffic lights and how they work. This project is the simple version of a traffic light system where we have demonstrated it for three sides or ways.

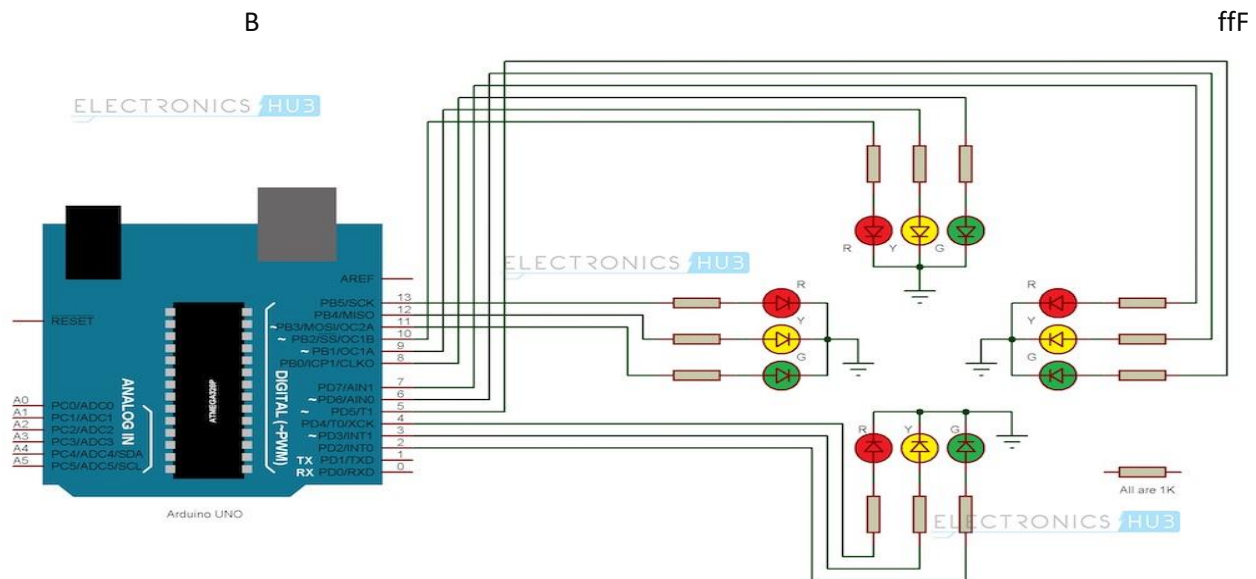


Fig 5.3: Traffic light control System

Components

- ARDUINO UNO
- 1K Ω Resistor X 12
- Red LEDs X 4
- Yellow LEDs X 4
- Green LEDs X 4
- Connecting wires
- Prototyping board
- Power adapter

Program

```
int Lane1[] = {13,12,11}; // Lane 1 Red, Yellow and Green
int Lane2[] = {10,9,8}; // Lane 2 Red, Yellow and Green
int Lane3[] = {7,6,5}; // Lane 3 Red, Yellow and Green
int Lane4[] = {4,3,2}; // Lane 4 Red, Yellow and Green

void setup()
{
  for (int i = 0; i < 3; i++)
  {
    pinMode(Lane1[i], OUTPUT);
    pinMode(Lane2[i], OUTPUT);
    pinMode(Lane3[i], OUTPUT);
    pinMode(Lane4[i], OUTPUT);
  }
  for (int i = 0; i < 3; i++)
  {
    digitalWrite(Lane1[i], LOW);
    digitalWrite(Lane2[i], LOW);
    digitalWrite(Lane3[i], LOW);
    digitalWrite(Lane4[i], LOW);
```

```

}
}
void loop()
{
digitalWrite(Lane1[2], HIGH);
digitalWrite(Lane3[0], HIGH);
digitalWrite(Lane4[0], HIGH);
digitalWrite(Lane2[0], HIGH);
delay(7000);
digitalWrite(Lane1[2], LOW);
digitalWrite(Lane3[0], LOW);
digitalWrite(Lane1[1], HIGH);
digitalWrite(Lane3[1], HIGH);
delay(3000);
digitalWrite(Lane1[1], LOW);
digitalWrite(Lane3[1], LOW);
digitalWrite(Lane1[0], HIGH);
digitalWrite(Lane3[2], HIGH);
delay(7000);
digitalWrite(Lane3[2], LOW);
digitalWrite(Lane4[0], LOW);
digitalWrite(Lane3[1], HIGH);
digitalWrite(Lane4[1], HIGH);
delay(3000);
digitalWrite(Lane3[1], LOW);
digitalWrite(Lane4[1], LOW);
digitalWrite(Lane3[0], HIGH);
digitalWrite(Lane4[2], HIGH);
delay(7000);
digitalWrite(Lane4[2], LOW);
digitalWrite(Lane2[0], LOW);

```

```
digitalWrite(Lane4[1], HIGH);  
digitalWrite(Lane2[1], HIGH);  
delay(3000);  
digitalWrite(Lane4[1], LOW);  
digitalWrite(Lane2[1], LOW);  
digitalWrite(Lane4[0], HIGH);  
digitalWrite(Lane2[2], HIGH);  
delay(7000);  
digitalWrite(Lane1[0], LOW);  
digitalWrite(Lane2[2], LOW);  
digitalWrite(Lane1[1], HIGH);  
digitalWrite(Lane2[1], HIGH);  
delay(3000);  
digitalWrite(Lane2[1], LOW);  
digitalWrite(Lane1[1], LOW);  
}
```

5.6 Importance of Traffic Management Signals:

- It ensures the safety of passengers, drivers, and pedestrians
- It ensures the order of the traffic movement at an intersection
- It helps in the reduction of the frequency and severity of some types of accidents
- It tells people when to go or when to stop, thus making it beneficial in maintaining the order on the road
- It prevents total gridlock on a day-to-day basis
- Reduces the number of accidents as well as fatalities.

CONCLUSION

On the whole, this internship was a useful experience. I have gained new knowledge, skills. I achieved several of my learning goals.

At last I come to conclusion that ARDUINO'S are inseparable part of our life. Whether we are at home or office or on the move, we are always surrounded by microcontroller and ARDUINO systems. These systems are being important role in our day today life like home automation, Traffic light control systems, automated car parking systems and automatic home security system are all examples of ARDUINO interfacing.

ARDUINO is an open-source electronics platform based on easy-to-use hardware and software.

Here Tinkercad and Proteus software's were discussed. Through this we can gain knowledge about this software's. Using these software's and ARDUINO we can do many useful projects.

REFERENCES

<https://www.elprocus.com/pin-diagram-of-8051-microcontroller/>

<http://techknowlearn.blogspot.com/2013/07/internal-ram-structure-of-8051.html>

<https://www.tutorialspoint.com/internal-ram-organization-of-intel-8051>

<https://www.watelectronics.com/arduino-technology-architecture-and-applications/>

<https://robu.in/ir-sensor-working/>

<https://www.elprocus.com/tilt-sensor-types-working-principle-and-its-applications/>

<https://comparecamp.com/tinkercad-review-pricing-pros-cons-features/>

https://en.wikipedia.org/wiki/Proteus_Design_Suite

<https://robu.in/traffic-light-controller-using-arduino/>