Ministry of Higher Education

Modern Academy

For Engineering and Technology in Maadi

Electronics and Communication Engineering Department

Bat Guidance system

Fifth Year Final Project

Prepared By

- Mahmoud Elsayed
- Ahmed Ashraf
- Aya Awwad
- Asala Saad
- Mohamed Gamal
- Mohamed Mohasseb
- Mahmoud Gamal
- Israa Mohamed
- Israa Elsayed
- Ahmed Tarek
- Taha Ibrahim

Supervised By

Dr: Lubna Fekry

Appreciation

Praise be to God Almighty, who helped us in completing the first part of this project, and who inspired us determination praise be to God, many thanks.

We extend our thanks and appreciation to supervisor

DR"Lubna Fekry "for all the guidance and valuable information which provided to us that contributed to enriching our project in its various aspects.

We also extend our thanks to the members of the distinguished discussion committee.

Contents

Chapter 1: Introduction & History

Chapter 2: Major operating system

Chapter 3: Smart Stick

Chapter 4 : Smart cap

Chapter 5 : Design

List of Figures

Chapter 1 :Introduction

Chapter 2 : Major operation system

Chapter 3 : Smart stick

Chapter 4 : Smart cap

Chapter 5: Design

Tables list

Chapter 3

Chapter 4

Abstract

Blind stick is an innovative stick designed for visually people for improved navigation . we here propose an advanced blind stick that allows visually challenged people to navigate with ease using advances technology .we will use some sensors that make blind person more determined his around environment, and we will make a mobile application to attach with stick and output will be on the speaker of mobile like ‘if he asked about the time ?’ the application will help him about this ,and the hat will knew who around him if he known or unknown person .

Chapter 1

Introduction

&

History

1.1 Introduction

Electronic oriented technology like Ultrasonic sensor can be used to assist the visually impaired person. In this technology, energy waves are emitted ahead, and then it is reflected from obstacles in the path of the user and detected by a matching sensor. Thus, the distance to the obstacle is calculated according to the time variance between the two signals. We have used different vibration intensities to indicate the distance of the object using the vibration motor. If the visually impaired person is too close to the obstacle the motor will vibrate at a higher intensity and also a buzzer will be turned on, alerting the visually impaired man to walk in different direction. In addition to this the smart cane will be linked with the smart phone, so that he can make use of maps. GPS system is used by visually impaired persons to determine and verify the correct route and also if the person fell on the ground the GSM-GPS module receives the information from the GPS satellite and transfers the latitude and longitude information as SMS message to a predefined mobile number.

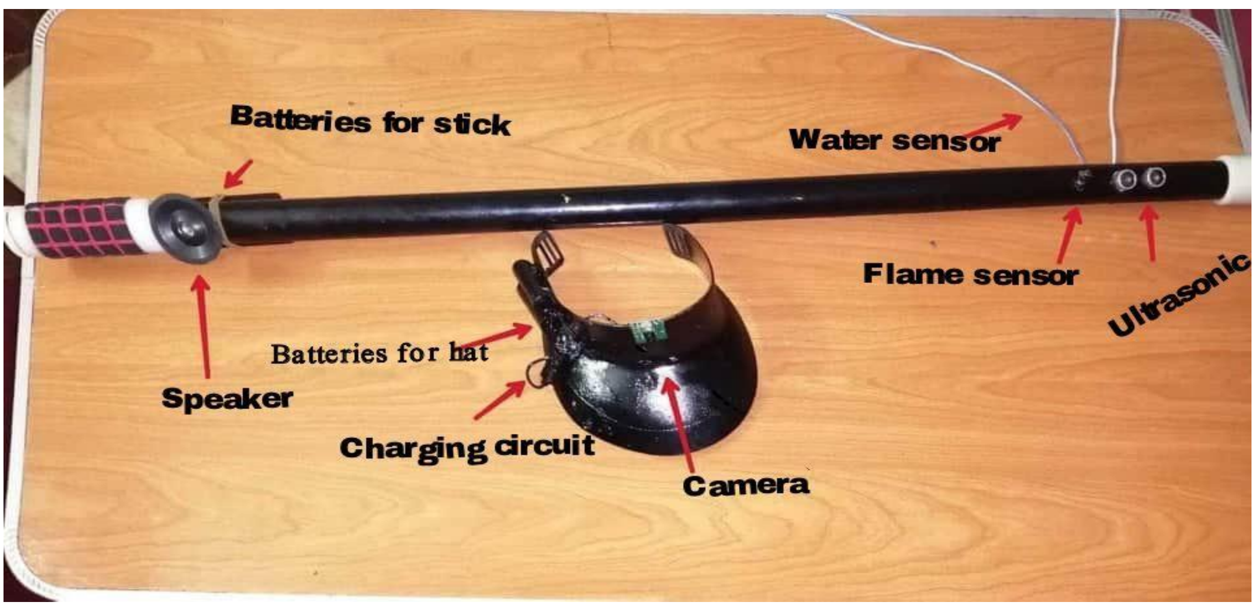


Figure (1.1) the stick and the hat

1.2 History

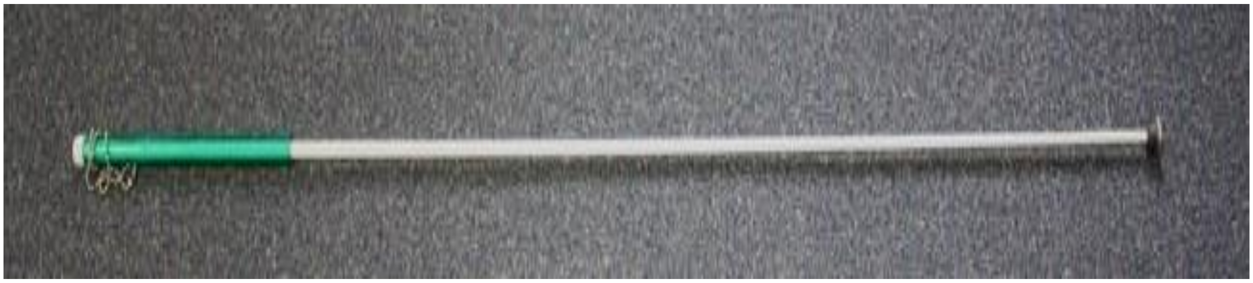


Figure (1.2) white cane in the past

The white cane originated in Europe in 1921 when James Biggs, a photographer who had lost his vision, began to paint his walking cane white to alert others to his presence.

1- When veterans of (World War II) returned to America with vision impairment and blindness they wanted to have the same level of independence as they had before the war. Because of this, the white walking cane was altered into the long cane form that is still prevalent today.

2- At present, 82% of the world’s blind population is at the age of 50 and above. Approximately 90% of the world’s visually impaired live in developing nations due to the lack of healthcare and medical treatments.

Chapter 2

Major

Operating systems

2.1 Embedded systems

2.1.1 History of embedded systems

1961 to reduce the size and weight of the Apollo Guidance Computer, the digital system installed on the Apollo Command Module and Lunar Module. The first computer to use ICs, it helped astronauts collect real-time flight data. Embedded systems date back to the 1960s. Charles Stark Draper developed an integrated circuit (IC)

In 1965, Automatics, now a part of Boeing, developed the D-17B; the computer used in the Minuteman I missile guidance system. It is widely recognized as the first mass-produced embedded system. When the Minuteman II went into production in 1966, the D-17B was replaced with the NS-17 missile guidance system, known for its high-volume use of integrated circuits. In 1968, the first embedded system for a vehicle was released; the Volkswagen 1600 used a microprocessor to control its electronic fuel injection system. Embedded system photo Macro photo of a little embedded system motherboard with attached cables. By the late 1960s and early 1970s, the price of integrated circuits dropped and usage surged. The first microcontroller was developed by Texas Instruments in 1971. The TMS 1000 series, which became commercially available in 1974, contained a 4-bit processor, read-only memory (ROM) and random-access memory (RAM), and cost around $2 apiece in bulk orders. Also in 1971, Intel released what is widely recognized as the first commercially available processor, the 4004. The 4-bit microprocessor was designed for use in calculators and small electronics, though it required eternal memory and support chips. The 8-bit Intel 8008, released in 1972 had 16 KB of memory; the Intel 8080 followed in 1974 with 64 KB of memory. The 8080's successor, x86 series, was released in 1978 and is still largely in use today.

In 1987, the first embedded operating system, the real-time Vx Works, was released by Wind River, followed by Microsoft's Windows Embedded CE in

1996. By the late 1990s, the first embedded Linux products began to appear. Today, Linux is used in almost all embedded devices.

2.1.2 What is embedded system?

In our day-to-day life we frequently use many electrical and electronic circuits and kits which are designed using embedded systems technology. The electrical and electronics engineering students and electronics and communications engineering students are required to design final year electronics projects to gain hands on experience with the real time embedded systems and also to fulfill the engineering graduation criteria. The engineering final year electronics projects are designed using embedded systems and applications. The computers, mobile phones, tablets, laptops, digital electronic systems, and other electrical and electronic gadgets are designed using embedded systems.

Embedded System: is the electronic system which integrates the hardware circuit with the software programming techniques for providing project solutions is called as embedded systems. By using this embedded system technology the complexity of the circuits can be reduced to a great extent which further reduces the cost and size. Embedded system was primarily developed by Charles Stark for reducing the size and weight of the project circuitry.

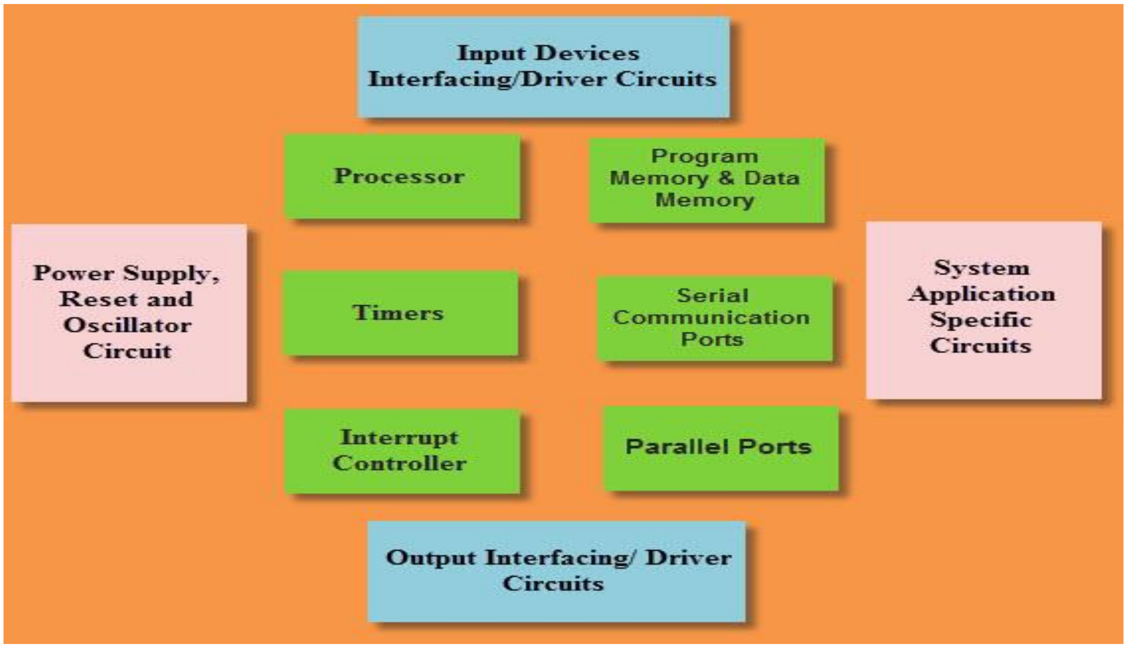


Figure (2.1) Embedded system description

An embedded system is basically an electronic system that can be programmed or non-programmed to operate, organize, and perform single or multiple tasks based on the application. In the real time embedded systems, all the assembled units work together based on the program or set of rules or code embedded into the microcontroller. But, by using this microcontroller programming techniques only a limited range of problems can be solved.

2.1.3 What are the applications of embedded systems?

Figure (2.2) application of embedded system

2.1.4 Applications of Embedded Systems

Embedded systems find numerous applications in various fields such as digital electronics, telecommunications, computing network, smart cards, satellite systems, military defense system equipment, research system equipment, and so on. Let us discuss a few practical applications of embedded systems that are used in designing embedded projects as a part of engineering final year electronics projects.

2.1.5 Embedded Systems Hardware

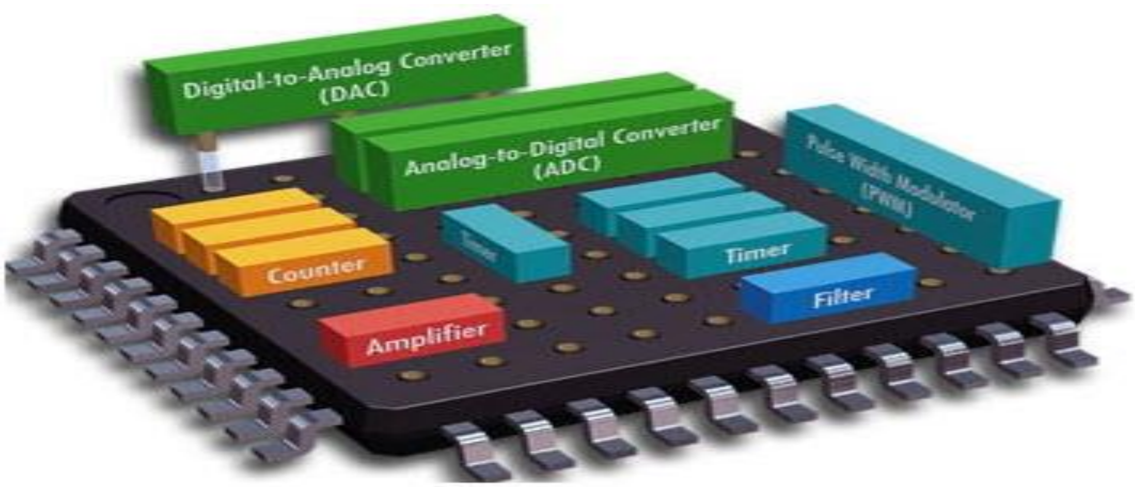


Figure (2.3) Embedded system hardware

Every electronic system consists of hardware circuitry; similarly, embedded system consists of hardware such as power supply kit, central processing unit, memory devices, timers, output circuits, serial communication ports, and system application specific circuit components & circuits.

2.1.6 Embedded Systems Software

Stream	Microprocessor Family	Source	CISC or RISC or Both Features
Stream 1	68HCxx	Motorola	CISC
Stream 2	80x86	Intel	CISC
Stream 3	SPARC	Sun	RISC
Stream 4	ARM	ARM	RISC with CISC functionality

Stream	Microcontroller Family	Source	CISC or RISC or Both Features
Stream 1	68HC11xx,HC12xx,HC16xx	Motorola	CISC
Stream 2	8051,8051MX	Intel, Philips	CISC
Stream 3	PIC 16F84 or 16C76,16F876 and PIC18	Microchip	CISC
Stream 4	Microcontroller Enhancements of CORTEX-M3 ARM9/ARM7 from Philips, Samsung and ST Microelectronics	ARM,Texas,Philips ,Samsung and ST Microelectronics etc	RISC Core with CISC functionality

Figure (2.4) Embedded system software

An embedded system is integration of hardware and software, the software used in the embedded system is set of instructions which are termed as a program. The microprocessors or microcontrollers used in the hardware circuits of embedded systems are programmed to perform specific tasks by following the set of instructions. These programs are primarily written using any programming software like Proteus or Lab-view using any programming languages such as C or C++ or embedded C. Then, the program is dumped into the microprocessors or microcontrollers that are used in the embedded system circuits.

2.2 IOT Technology

2.2.1 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.

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).

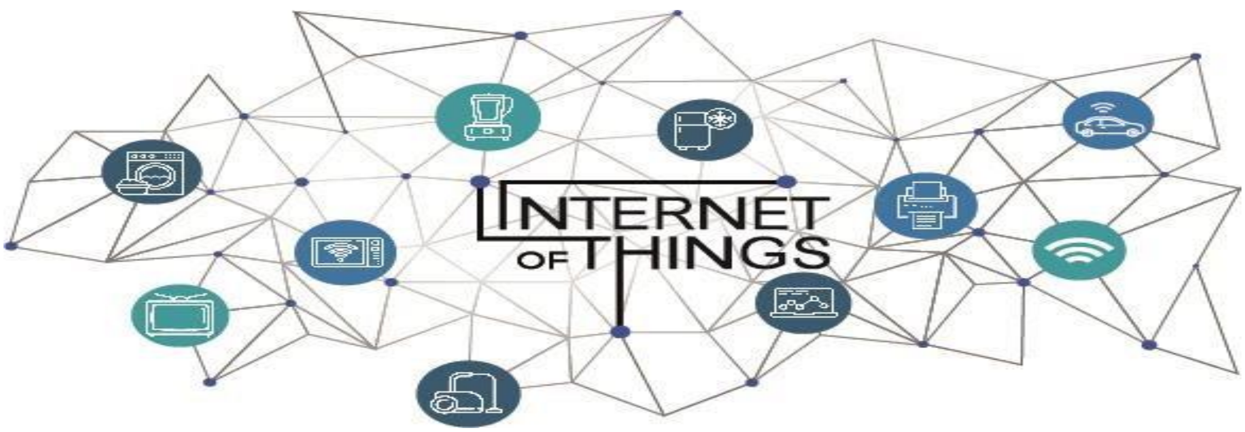


Figure (2.5) IOT Technology

2.2.2 Defination of IOT

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 and sharing information based on stipulated protocols in order to achieve smart reorganizations, positioning, tracing, safe and control and even personal real time online monitoring, online upgrade, process control and administration.

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.

2.2.3 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 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

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.

The Internet of Things is not a single technology, but it is a mixture of different hardware and 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.

Examples of standards in these categories include wired and wireless technologies like Ethernet, WI-FI, Bluetooth, ZigBee, GSM, and GPRS .

2.2.4 Characteristics

1. **Interconnectivity:** With regard to the IOT, anything can be interconnected with the global information and communication infrastructure.

2. **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.

3. **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.

4. **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.

5. **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.

6. **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.

7. **Connectivity:** Enables network accessibility and compatibility. Accessibility is getting on a network while compatibility provides the common ability to consume and produce data.

2.2.5 IOT Architecture

2.2.5.1 smart device / sensor layer

The lowest layer is made up of smart objects integrated with sensors. The sensors enable the interconnection of the physical and digital worlds allowing real-time information to be collected and processed. There are various types of sensors for different purposes. The sensors have the capacity to take measurements such as temperature, air quality, speed, humidity, pressure, flow, a degree of memory, enabling them to record a certain number of measurements.

A sensor can measure the physical property and convert it into signal that can be understood by an instrument. Sensors are grouped according to their unique purpose such as environmental sensors, body sensors, home appliance sensors and vehicle telematics sensors, etc.

Most sensors require connectivity to the sensor gateways. This can be in the form of a Local Area Network (LAN) such as Ethernet and Wi-Fi connections or Personal Area Network (PAN) such as ZigBee, Bluetooth and Ultra - Wideband (UWB). For sensors that do not require connectivity to sensor aggregators, their connectivity to backend servers/applications can be provided using Wide Area Network (WAN) such as GSM, GPRS and LTE. Sensors that use low power and low data rate connectivity, they typically form networks commonly (PAN) such as ZigBee, Bluetooth and Ultra-Wideband (UWB).

For sensors that do not require connectivity to sensor aggregators, their connectivity to backend servers/ applications can be provided using Wide Area Network (WAN) such as GSM, GPRS and LTE. Sensors that use low power and low data rate connectivity, they typically form networks commonly known as wireless sensor networks (WSNs). WSNs are gaining popularity as they can accommodate far more sensor nodes while retaining adequate battery life and covering large areas.

2.2.5.2 Gateways and Networks

Massive volume of data will be produced by these tiny sensors and this requires a robust and high performance wired or wireless network infrastructure as a transport medium. Current networks, often tied with very different protocols, have been used to support machine-to-machine networks and their applications.

With demand needed to serve a wider range of IOT services and applications such as high speed transactional services, context-aware applications, etc, multiple networks with various technologies and access protocols are needed to work with each other in a heterogeneous configuration. These networks can be in the form of a private, public or hybrid models and are built to support the communication requirements for latency, bandwidth or security. Various gateways (microcontroller, microprocessor...) and gateway networks (WI-FI, GSM, GPRS…)

2.2.5.3 Data management

Data management is the ability to manage data information flow. With data management in the management service layer, information can be accessed, integrated and controlled. Higher layer applications can be shielded from the need to process unnecessary data and reduce the risk of privacy disclosure of the data source.

Data filtering techniques such as data anonymisation, data integration and data synchronization, are used to hide the details of the information while providing only essential information that is usable for the relevant applications. With the use of data abstraction, information can be extracted to provide a common business view of data to gain greater agility and reuse across domains.

Security must be enforced across the whole dimension of the IOT architecture right from the smart object layer all the way to the application layer. Security of the system prevents system hacking and compromises by unauthorized personnel, thus reducing the possibility of risks.

2.2.5.4 Application Layer

The IOT application covers smart environments/spaces in domains such as: Transportation, Building, City, Lifestyle, Retail, Agriculture, Factory, Supply chain, Emergency, Healthcare, User interaction, Culture and tourism, Environment and Energy

Applications:

1. Industrial systems use sensors to monitor. An increasing number of electric motors, for example, include sensors that collect data used to predict impending motor failures.

2. Smart buildings use sensors to identify the locations of people . That data can be used to control heating/ventilation/air conditioning systems and lighting systems to reduce operating costs.

Figure (2.6) IOT control to system in smart building

3. Smart cities use sensors to monitor pedestrian and vehicular traffic.

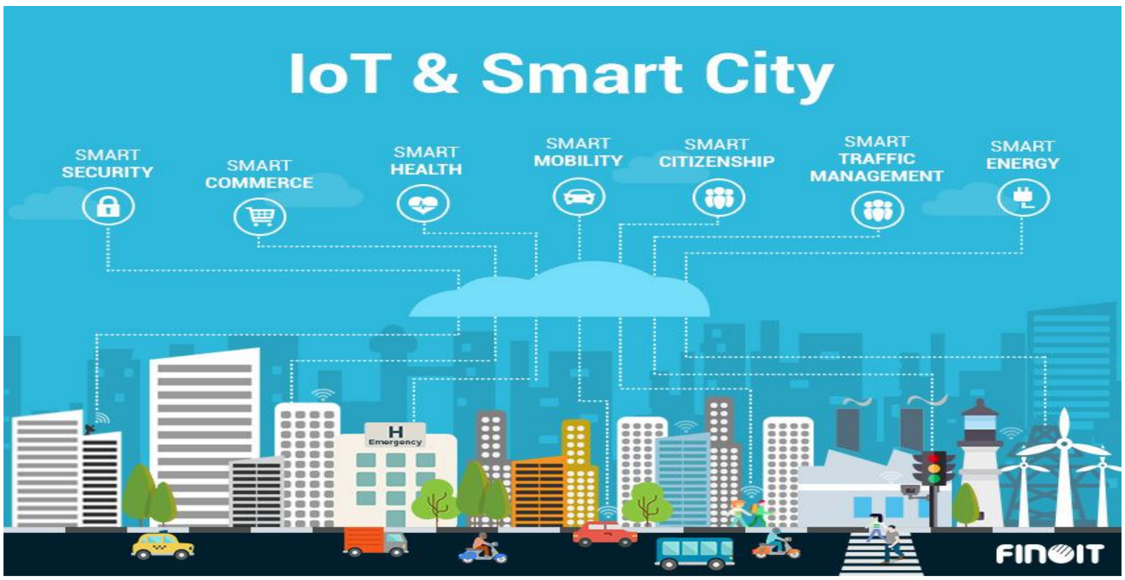


Figure (2.7) IOT control to vehicular traffic in smart city

4. Vehicles use networked sensors to monitor and provide improved dynamics, reduced fuel consumption.

Figure (.28) IOT reduce fuel in vehicles

5. Medical systems connect a wide range of patient monitoring sensors.

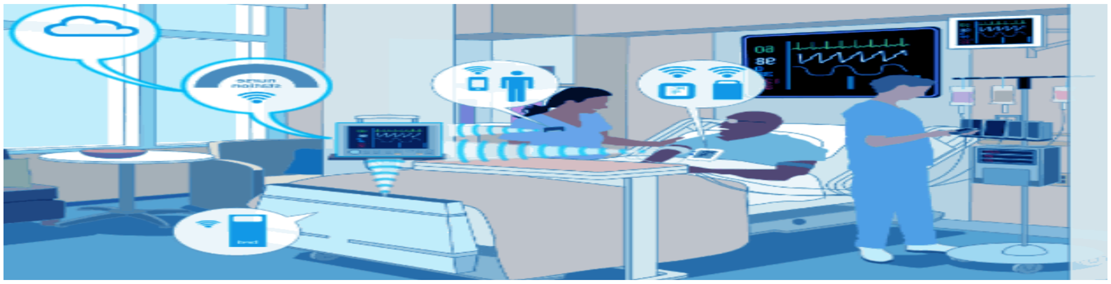


Figure (2.9) IOT in medical system

Chapter 3

Smart stick

3.1 Introduction

Visually impaired persons have difficulty to interact and feel their environment. They have little contact with surroundings. Physical movement is a challenge for visually impaired persons, because it can become tricky to distinguish obstacles appearing in front of them, and they are not able to move from one place to another. They depend on their families for mobility and financial support. Their mobility opposes them from interacting with people and social activities.

In the past, different systems are designed with limitations without a solid understanding of the no visual perception.

Researchers have spent the decades to develop an intelligent and smart stick to assist and alert visually impaired persons from obstacles and give information about their location. Over the last decades, research has been conducted for new devices to design a good and reliable system for visually impaired persons to detect obstacles and warn them at danger places.

Smart walking stick is specially designed to detect obstacles which may help the blind to navigate care-free. The audio messages will keep the user alert and considerably reduce accidents. A voice enabled automatic switching is also incorporated to help them in private space as well. This

system presents a concept to provide a smart electronic aid for blind people, both in public and private space the proposed system contains the ultrasonic sensor, water sensor, voice play back board, raspberry pi and speaker. The proposed system detects the obstacle images which are present in outdoor and indoor with the help of a camera. The Stick measures the distance between the objects and smart walking stick by using

an ultrasonic sensor. When any objects or obstacles come in range of an ultrasonic sensor then the head phone tell the name of obstacle which is in front of the stick. The smart walking stick is a simple and purely mechanical device to detect

the obstacles on the ground. This device is light in weight and portable. But its range is limited due to its own size. It provides the best travel aid for the person. The blind person can move from one place to another independently without the others help. The main aim of the system is to provide a efficient navigation aid for the blind persons which gives a sense of vision by providing the information about their surroundings and objects around them.

3.2 Stick requirements

3.2.1 Arduino

3.2.1.1 Introduction

An Arduino is a great way to create physical interfaces. It was designed to be an easy-to-use electronics platform that allows you to attach electronic components that can send and receive data. The respected Arduino community can be very helpful and creative if you need assistance. The electronic components can be inputs or outputs. I have seen Arduino used to create music, light shows, ovens, robots, art, and so much more. There is such a wide range of components including buttons, motors, potentiometers, sensors, and buzzers that can be attached to an Arduino. Arduino can send and receive data from a web server. This means you can control elements on a web page with physical components and use physical components to display information from your web page or online data.

Arduino allows you to create your own electronics projects. It is a collection of open source hardware and software that allows you to attach and control other components to create an electrical circuit. Projects such as an automated plant watering system, a pizza oven, or a remote controlled toy car can be made with an Arduino.

When you use an Arduino for a project you need to do the following:

- Connect components to it.
- Write a program to control the components.
- Verify that the program is written correctly.

- Upload the program to the Arduino.

The Arduino needs to be connected to a computer via a USB port to upload a program to it. Programs for Arduino are called sketches. Once the sketch is uploaded, it is stored on the microcontroller and will stay there until another sketch is uploaded. Once a new sketch is uploaded the old sketch is no longer available. Once the sketch is uploaded you can disconnect the Arduino from the computer, and if it is connected to another power source the program will still run.

3.2.1.2 Arduino Hardware

An Arduino board is made up of a number of components, including a microcontroller, digital and analog pins, power pins, resistors, a diode, a capacitor, and an LED. A Microcontroller has a central processing unit (CPU); it stores the uploaded sketch and processes and directs the commands. The digital and analog pins are used for sending and receiving digital and analog data.

The Arduino also has a serial interface that allows the Arduino to send data to a computer via the serial port; this is the way we will be sending data to and from a computer Electricity with an Arduino you create an electronic circuit that powers the components attached to it.

3.2.1.3 Digital and Analog

On an Arduino can use digital input, digital output, analog input, and analog output. A digital input or output can have one of two states, on or off (high or low). The analog input or output can be between 0 and 1023 when 5V is being used. Analog Output Analog output and input produce a range of numbers that go up and down in sequence. On an Arduino some of the digital pins have a “~” symbol next to them. These pins are used for analog output and use PWM (pulse width modulation).

Pulse Width Modulation PWM is used to simulate an analog output with digital pins. A digital signal can be on or off, and it sends a pulse for on. PWM simulates an analog system using the digital signal by changing the length of the pulse; it’s “on” time to simulate pulses between 5V and 0V.

Digital Input a good circuit to show a digital input is a switch button. The switch button is either up or down, and it is in one of two states, pressed or not pressed. It brings in another concept called Input Pull up. There is a problem for an Arduino with a switch. When a switch is open, it does not complete a circuit, and there is no voltage so the Arduino doesn’t know what the input is; it could be 0 or it could be 1.

As it doesn’t know you can get strange results, it creates noise as the input value is unknown and it tries to put something in. This problem is solved with pull up resistors; it sets a voltage when the switch is open. Analog Input Analog input is used with components such as photo resistors and potentiometers, components that give varying values.

An Arduino Uno can register values between 0 and 5 volts; with this you can get an analog input value between 0 and 1023. An analog input sends a signal voltage. When the signal voltage is received it is checked against an internal reference

3.2.1.4 Advantages

1. Ready to use

As Arduino comes in a complete package form which includes the 5V regulator, a burner, an oscillator, a micro-controller, serial communications interface LED and headers for the connections. You don’t have to think about programmer connections for programming or any other interface. Just plug it into USB port of your computer and that's it. Your revolutionary idea is going to change the world after just few words of coding.

2. Example of Code

Another big advantage of Arduino is its library of examples present inside the software of Arduino. I'll explain this advantage using an example of voltage measurement. For example if you want to measure voltage using ATmega8 micro-controller and want to display the output on computer screen then you have to go through the whole process. The process will start from learning the ADC's of micro-controller for measurement, went through the learning of serial communication for display and will end at USB – Serial converters.

3. Effortless function

During coding of Arduino, you will notice some functions which make the life so easy. Another advantage of Arduino is its automatic unit conversion capability. You can say that during debugging you don't have to worry about the units conversions. Just use your all force on the main parts of your projects. You don't have to worry about side problems.

4. Large Community

During coding of Arduino, you will notice some functions which make the life so easy. Another advantage of Arduino is its automatic unit conversion capability. You can say that during debugging you don't have to worry about the units conversions. Just use your all force on the main parts of your projects.

3.2.1.5 Disadvantages

1. Structure

During building a project you have to make its size as small as possible. But with the big structures of Arduino we have to stick with big sized PCB's. If you are working on a small micro-controller like ATmega8 you can easily make your PCB as small as possible.

2. Cost

The most important factor which you cannot deny is cost. This is the problem which every hobbyist, Engineer or Professional has to face. Now, we must consider that the Arduino is cost effective or not

3.2.2 Arduino Nano

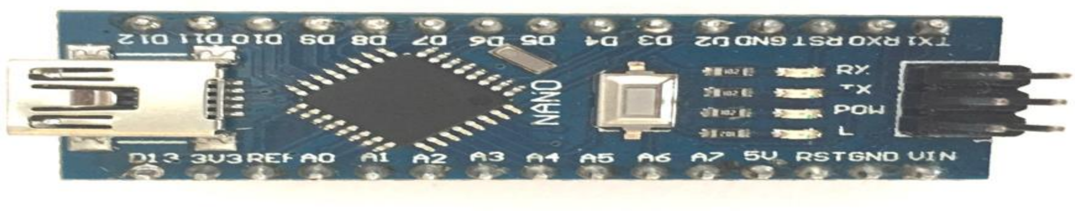


Figure (3.1) Arduino Nano

3.2.2.1 Arduino Nano Pin Configuration

Pin Category	Pin Name	Details
Power	**Vin,** **3.3V,** **5V,** **GND**	**Vin:** Input voltage to Arduino when using an external power source (6-12V). **5V:** Regulated power supply used to power microcontroller and other components on the board. **3.3V:** 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50mA. **GND:** Ground pins.
Reset	**Reset**	Resets the microcontroller.
Analog Pins	**A0 – A7**	Used to measure analog voltage in the range of 0-5V

Input/ Output Pins	**Digital Pins D0 - D13**	Can be used as input or output pins. 0V and 5V
Serial	**Rx, Tx**	Used to receive and transmit TTL serial data.
External Interrupts	**2, 3**	To trigger an interrupt.
PWM	**3, 5, 6, 9, 11**	Provides 8-bit PWM output.
SPI	**10 (SS), 11 (MOSI),12 (MISO) and 13 (SCK)**	Used for SPI communication.
Inbuilt LED	**13**	To turn on the inbuilt LED.
IIC	**A4 (SDA), A5 (SCA)**	Used for TWI communication.
AREF	**AREF**	To provide reference voltage for input voltage.

Table (3.1) Arduino Nano Pin Configuration

3.2.2.2 Arduino Nano Technical Specifications

Microcontroller	ATmega328P – 8 bit AVR family microcontroller
Operating Voltage	5V
Recommended Input Voltage for Vin pin	7-12V
Analog Input Pins	6 (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3V Pin	50 mA
Flash Memory	32 KB (2 KB is used for Bootloader)
SRAM	2 KB
EEPROM	1 KB
Frequency (Clock Speed)	16 MHz
Communication	I2C, SPI, USART

Table (3.2) Arduino Nano Technical Specifications

3.2.2.3 Understanding Arduino Nano

The Arduino board is designed in such a way that it is very easy for beginners to get started with microcontrollers. This board especially is breadboard friendly is very easy to handle the connections. Let's start with powering the Board.

Powering you Arduino Nano:

There are totally three ways by which you can power your Nano.

USB Jack: Connect the mini USB jack to a phone charger or computer through a cable and it will draw power required for the board to function

Vin Pin: The Vin pin can be supplied with a unregulated 6-12V to power the board. The on-board voltage regulator regulates it to +5V

+5V Pin: If you have a regulated +5V supply then you can directly provide this o the +5V pin of the Arduino.

Input/output:

There are totally 14 digital Pins and 8 Analog pins on your Nano board. The digital pins can be used to interface sensors by using them as input pins or drive loads by using them as output pins. A simple function like pinMode() and digitalWrite() can be used to control their operation. The operating voltage is 0V and 5V for digital pins. The analog pins can measure analog voltage from 0V to 5V using any of the 8 Analog pins using a simple function liken analogRead()

These pins apart from serving their purpose can also be used for special purposes which are discussed below:

- **Serial Pins 0 (Rx) and 1 (Tx):** Rx and Tx pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB to TTL serial chip.
- **External Interrupt Pins 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- **PWM Pins 3, 5, 6, 9 and 11:** These pins provide an 8-bit PWM output by using analogWrite() function.
- **SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK):** These pins are used for SPI communication.

- **In-built LED Pin 13:** This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.
- **I2C A4 (SDA) and A5 (SCA):** Used for IIC communication using Wire library.
- **AREF:** Used to provide reference voltage for analog inputs with analogReference() function.

- **Reset Pin:** Making this pin LOW, resets the microcontroller.

3.2.2.4 Applications

- Prototyping of Electronics Products and Systems
- Multiple DIY Projects.
- Easy to use for beginner level DIYers and makers.
- Projects requiring Multiple I/O interfaces and communications.

3.2.3 Survey between Arduino types

Name	Processor	Operating Voltage/Input Voltage	CPU Speed	Analog In/Out	Digital IO/PWM	EEPROM [KB]	SRAM [KB]	Flash [KB]	USB
Uno	ATmega328	5 V/7-12 V	16 Mhz	6/0	14/6	1	2	32	Regular
Due	AT91SAM3X8E	3.3 V/7-12 V	84 Mhz	12/2	54/12	-	96	512	2 Micro
Leonardo	ATmega32u4	5 V/7-12 V	16 Mhz	12/0	20/7	1	2.5	32	Micro
Mega 2560	ATmega2560	5 V/7-12 V	16 Mhz	16/0	54/15	4	8	256	Regular
Mega ADK	ATmega2560	5 V/7-12 V	16 Mhz	16/0	54/15	4	8	256	Regular
Micro	ATmega32u4	5 V/7-12 V	16 Mhz	12/0	20/7	1	2.5	32	Micro
Mini	ATmega328	5 V/7-9 V	16 Mhz	8/0	14/6	1	2	32	-
Nano	ATmega168 ATmega328	5 V/7-9 V	16 Mhz	8/0	14/6	0.512 1	1 2	16 32	Mini-B
Ethernet	ATmega328	5 V/7-12 V	16 Mhz	6/0	14/4	1	2	32	Regular
Esplora	ATmega32u4	5 V/7-12 V	16 Mhz	-	-	1	2.5	32	Micro
ArduinoBT	ATmega328	5 V/2.5-12 V	16 Mhz	6/0	14/6	1	2	32	-
Fio	ATmega328P	3.3 V/3.7-7 V	8 Mhz	8/0	14/6	1	2	32	Mini
Pro (168)	ATmega168	3.3 V/3.35-12 V	8 Mhz	6/0	14/6	0.512	1	16	-
Pro (328)	ATmega328	5 V/5-12 V	16 Mhz	6/0	14/6	1	2	32	-
Pro Mini	ATmega168	3.3 V/3.35-12 V 5 V/5-12 V	8 Mhz 16Mhz	6/0	14/6	0.512	1	16	-
LilyPad	ATmega168V ATmega328V	2.7-5.5 V/2.7-5.5 V	8 Mhz	6/0	14/6	0.512	1	16	-
LilyPad USB	ATmega32u4	3.3 V/3.8-5V	8 Mhz	4/0	9/4	1	2.5	32	Micro
LilyPad Simple	ATmega328	2.7-5.5 V/2.7-5.5 V	8 Mhz	4/0	9/4	1	2	32	-
LilyPad SimpleSnap	ATmega328	2.7-5.5 V/2.7-5.5 V	8 Mhz	4/0	9/4	1	2	32	-

Figure (3.2) difference between types of Arduino

There are some Arduino not popular so that we speak about the most popular types in electronic field.

3.2.3 Ultrasonic

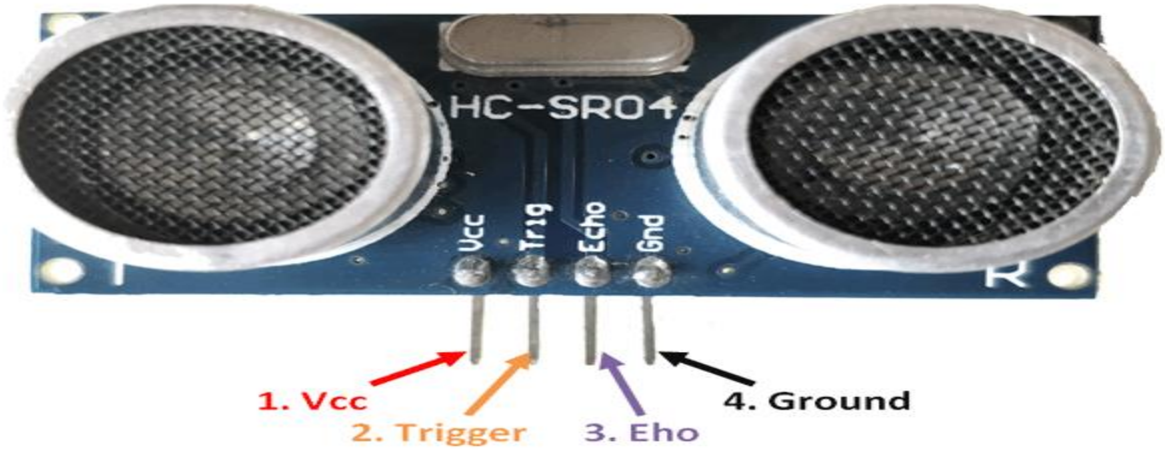


Figure (3.3) Ultrasonic sensor

3.2.3.1 Pin Configuration

Pin Number	Pin Name	Description
1	Vcc	The Vcc pin powers the sensor, typically with +5V
2	Trigger	Trigger pin is an Input pin. This pin has to be kept high for 10us to initialize measurement by sending US wave.
3	Echo	Echo pin is an Output pin. This pin goes high for a period of time which will be equal to the time taken for the US wave to return back to the sensor.
4	Ground	This pin is connected to the Ground of the system.

Table (3.3) pin configuration of ultrasonic

3.2.3.2 HC-SR04 Sensor Features

- Operating voltage: +5V
- Theoretical Measuring Distance: 2cm to 400cm
- Accuracy: 3mm
- Measuring angle covered: <15°
- Operating Current: <15mA
- Operating Frequency: 40KHz

3.2.3.3 HC-SR04 Ultrasonic Sensor Working

Figure (3.4) Ultrasonic working

As shown above the **HC-SR04 Ultrasonic (US) sensor** is a 4 pin module, whose pin names are Vcc, Trigger, Echo and Ground respectively. This sensor is a very popular sensor used in many applications where measuring distance or sensing objects are required. The module has two eyes like projects in the front which forms the Ultrasonic transmitter and Receiver. The sensor works with the simple high school formula that

$$\textbf{Distance} = \textbf{Speed} \times \textbf{Time}$$

The Ultrasonic transmitter transmits an ultrasonic wave, this wave travels in air and when it gets objected by any material it gets reflected back toward the sensor this reflected wave is observed by the Ultrasonic receiver module.

Now, to calculate the distance using the above formulae, we should know the Speed and time. Since we are using the Ultrasonic wave we know the universal speed of US wave at room conditions which is 340m/s. The circuitry inbuilt on the module will calculate the time taken for the US wave to come back and turns on the echo pin high for that same particular amount of time, this way we can also know the time taken. Now simply calculate the distance using a microcontroller or microprocessor.

3.2.3.4 How to use the HC-SR04 Ultrasonic Sensor

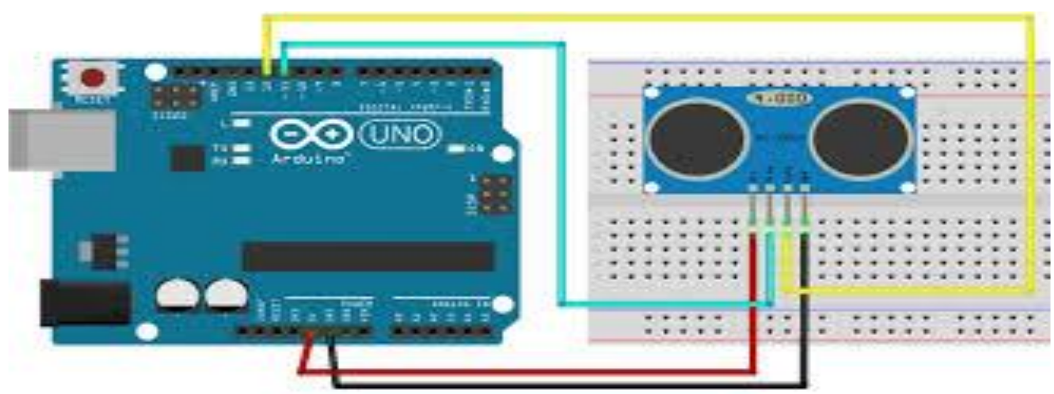


Figure (3.5) ultrasonic with Arduino

HC-SR04 distance sensor is commonly used with both microcontroller and microprocessor platforms like Arduino, ARM, PIC, Raspberry Pie etc. The following guide is universally since it has to be followed irrespective of the type of computational device used.

Power the Sensor using a regulated +5V through the Vcc ad Ground pins of the sensor. The current consumed by the sensor is less than 15mA and hence can be directly powered by the on board 5V pins (If available). The Trigger and the Echo pins are both I/O pins and hence they can be connected to I/O pins of the microcontroller. To start the measurement, the trigger pin has to be made high for 10uS and then turned off. This action will trigger an ultrasonic wave at frequency of 40KHz from the transmitter and the receiver will wait for the wave to return. Once the wave is returned after it getting reflected by any object the Echo pin goes high for a particular amount of time which will be equal to the time taken for the wave to return back to the sensor.

The amount of time during which the Echo pin stays high is measured by the MCU/MPU as it gives the information about the time taken for the wave to return back to the Sensor. Using this information the distance is measured as explained in the above heading.

3.2.3.5 Applications

- Used to avoid and detect obstacles with robots like biped robot, obstacle avoider robot, path finding robot etc.
- Used to measure the distance within a wide range of 2cm to 400cm.
- Can be used to map the objects surrounding the sensor by rotating it.
- Depth of certain places like wells, pits etc can be measured since the waves can penetrate through water.

3.2.4 IR Proximity Sensor

Figure (3.6) Proximity sensor E18-D80NK

3.2.4.1 Specification of IR proximity module

- Extremely easy to setup and use
- Fully adjustable detection range from 3 to 80cm
- Simple logic and LED output indicates when object is detected
- Built-in 45″ cable
- 5V Operation
- Optional mounting bracket available

This is an extremely easy to use sensor that consists of an IR transmitter and receiver pair in one module. The IR transmitter emits IR light and the IR receiver detects the IR light that is reflected off objects.

The sensor was originally designed to automate the counting of objects moving along a production line, but it works equally well as an obstacle avoidance sensor on a robotic vehicle or any other applications where detection of objects or obstacles within its range is desired such as for a security system or monitoring a pet door.

Detection range is approximately 3-80cm. The range is adjustable using a multi-turn screw on the back of the unit. Turning the screw CW increases the range at which objects are detected. Turning the screw CCW decreases the range. If the adjustment is turned CCW too far, the module will no longer detect an object until it is turned CW again. Similarly if it is turned CW all the way, it will always show a detection until it is turned CCW again.

The module can detect that presence of the object within the range it has been set for, but it cannot determine the distance to the object.

The signal output is normally held HIGH due to a built-in 10K pull-up resistor. When an object is detected, the output drops to LOW as long as it is detected. A red LED on the back of the module is lit when an object is detected. The output is an open collector that can sink a max of 100mA.

The module operates at 5V. Current draw is typically only 5mA though max current draw is rated as 25mA.

The body of the module is threaded and 2 knurled nuts are included which allow the sensor to be mounted in a 19mm hole. The amount of protrusion through the hole is adjustable using the nuts.

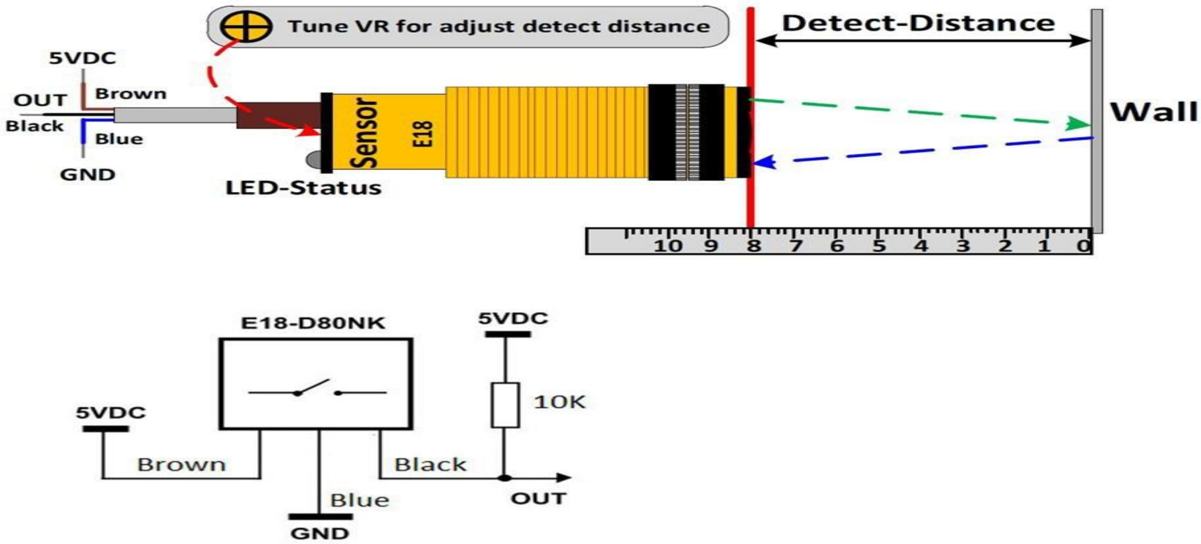
We also have an acrylic bracket available below that can be used to mount the sensor which is handy for some applications such as robotic vehicles.

There is a 3-wire cable attached to the assembly. The leads are tinned. This cable can be cut to length for the application the sensor will be used in.

3.2.4.2 Cable Pin-out

- **Brown Wire** – Connect to 5V.
- **Blue Wire** – Connects to Ground
- **Black Wire** – Logic Output, active LOW

3.2.4.3 The internal circuit of IR sensor



Figure(3.7) The internal circuit of IR proximate sensor

3.2.4.4 Interface IR proximity module with Arduino

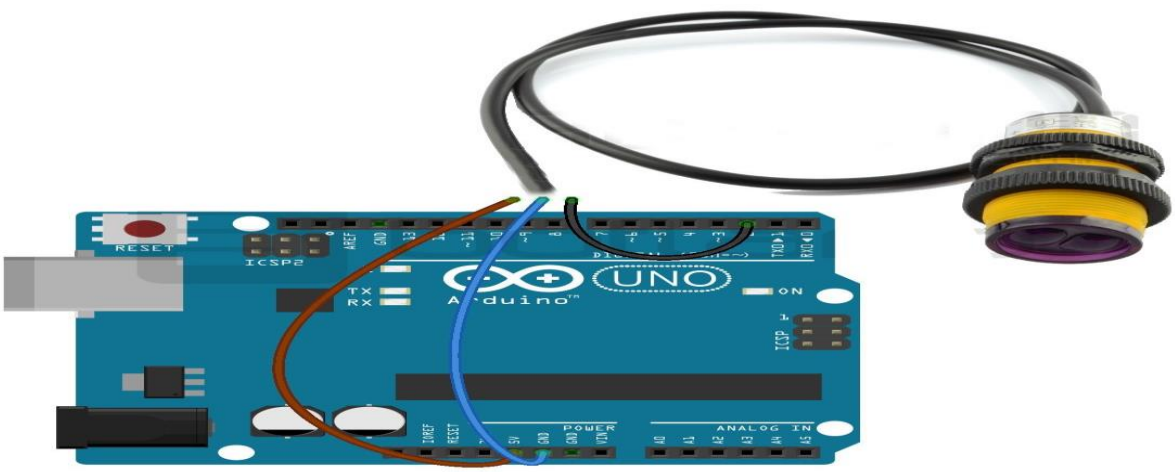


Figure (3.8) IR Proximity sensor with Arduino

3.2.4.5 Applications

- Multi function reminder
- Maze robot
- Kitchen automation system
- Security anti-theft system

3.2.4.6 Comparison between IR proximity &Ultrasonic

Parameters	IR proximity	Ultrasonic
Range	3cm-to-80cm	2cm-to-400cm
Beam-width	15 Deg	30 Deg
Beam pattern	Line (narrow)	Canonical
Current	25mA	15mA
Voltage	5V (DC)	5V (DC)
Dimension	Length 45mm Diameter 17mm	Length 45mm Width 20mm

I) **Table (3.4)** comparison between IR proximity &Ultrasonic

3.2.5 Flame sensor

A flame detector is a sensor designed to detect and respond to the presence of a flame or fire. Responses to a detected flame depend on the installation, but can include sounding an alarm, deactivating a fuel line (such as a propane or a natural gas line), and activating a fire suppression system

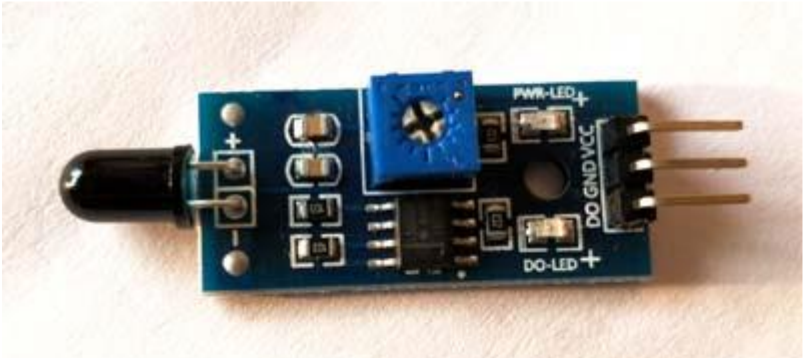


Figure (3.9) Flame sensor

3.2.5.1 Pin Description

Vcc: 3.3-5V power supply

GND: Ground

Dout: Digital output

There are different types of flame detection methods. Some of them are: Ultraviolet detector, near IR array detector, infrared (IR) detector, Infrared thermal cameras, UV/IR detector etc.

When fire burns it emits a small amount of Infra-red light, this light will be received by the Photodiode (IR receiver) on the sensor module. Then we use an Op-Amp to check for change in voltage across the IR Receiver, so that if a fire is detected the output pin (DO) will give 0V(LOW) and if the is no fire the output pin will be 5V(HIGH).

It can detect infrared light with a wavelength ranging from 760nm to 1100nm and its detection angle is about 60°. Flame sensor module consists of a photodiode (IR receiver), resistor, and capacitor, potentiometer, and LM393 comparator in an integrated circuit. The sensitivity can be adjusted by varying the on board potentiometer. Working voltage is between 3.3v and 5v DC, with a digital output. Logic high on the output indicates presence of flame or fire. Logic low on output indicates absence of flame or fire.

3.2.5.2 Applications of flame sensors

- Hydrogen stations
- Combustion monitors for burners
- Oil and gas pipelines
- Automotive manufacturing facilities
- Nuclear facilities
- Aircraft hangars
- Turbine enclosures

3.2.5.3 Interface flame module with Arduino

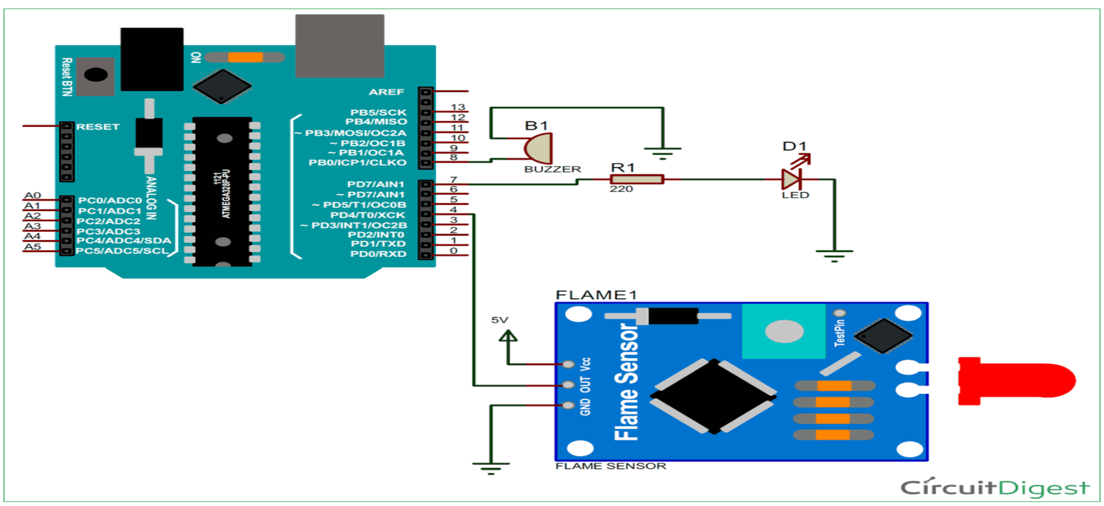


Figure (3.10) Flame sensor with Arduino

The flame sensor detects the presence of fire or flame based on the Infrared (IR) wavelength emitted by the flame. It gives logic 1 as output if flame is detected; otherwise it gives logic 0 as output. Arduino Uno checks the logic level on the output pin of the sensor and performs further tasks such as activating the buzzer and LED, sending an alert message.

3.2.6 Water sensor

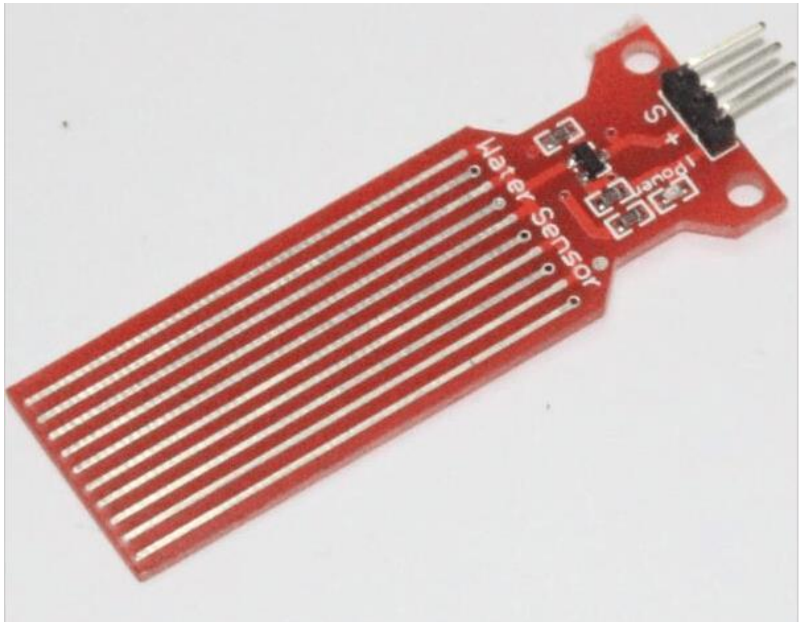


Figure (3.11) water sensor

Continuing with sensors that can be used in aquaponics, hydroponics systems or automatic outdoor plant care, we will explain how to use a water level sensor to control the amount of water we have in a tank. The same sensor can also be used to detect if it is raining.

3.2.6.1 Pin Configuration

- **+ stands for** +5V
- **– stands for** GND
- **S stands for** signal

3.2.6.2 Specifications

This sensor is supplied to 5V or 3.3V on VCC and GND pins. The pin ***S*** will give us an analog value between VCC and GND. So we will use the S pin as analog input connecting Arduino, the value read will be higher depending on the sensor surface is covered with water. This is because the water acts as a conductor, given that the water we use in our deposits not be pure water (H2O), since if water is nonconductive.

3.2.6.3 Application

- **Use as a level in a tank**

  To use it as level detector in a deposit have to install the sensor on the inside of the tank at the level where we want to control the water level. The sensor must be positioned so that parallel lines are perpendicular to the sensor water level. The pin *S* will give us a greater value as the sensor is immersed.

- **Use as rain detector**

  To detect whether it is raining with this sensor have to position it horizontally so that rain falling on the sensor, as raindrops fall on the sensor film of water on the surface will be formed by increasing the value of the pin *S*, this how we can deduce if it is raining

3.2.6.4 Circuit Diagram of water level detector

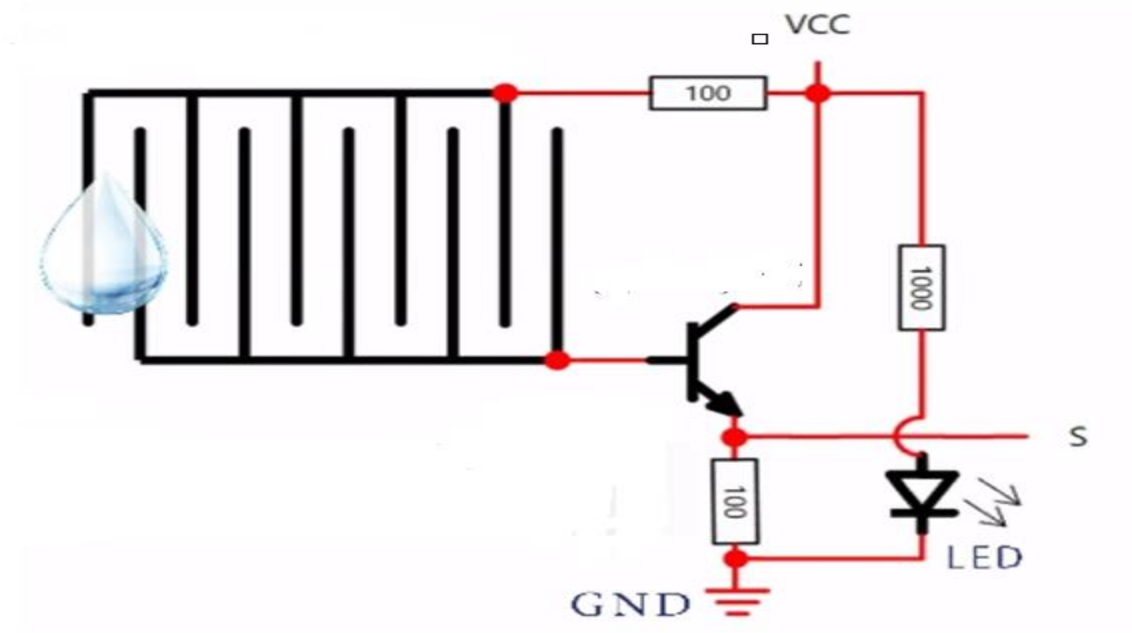


Figure (3.12) circuit diagram of water sensor

3.2.6.5 Interface water sensor with Arduino

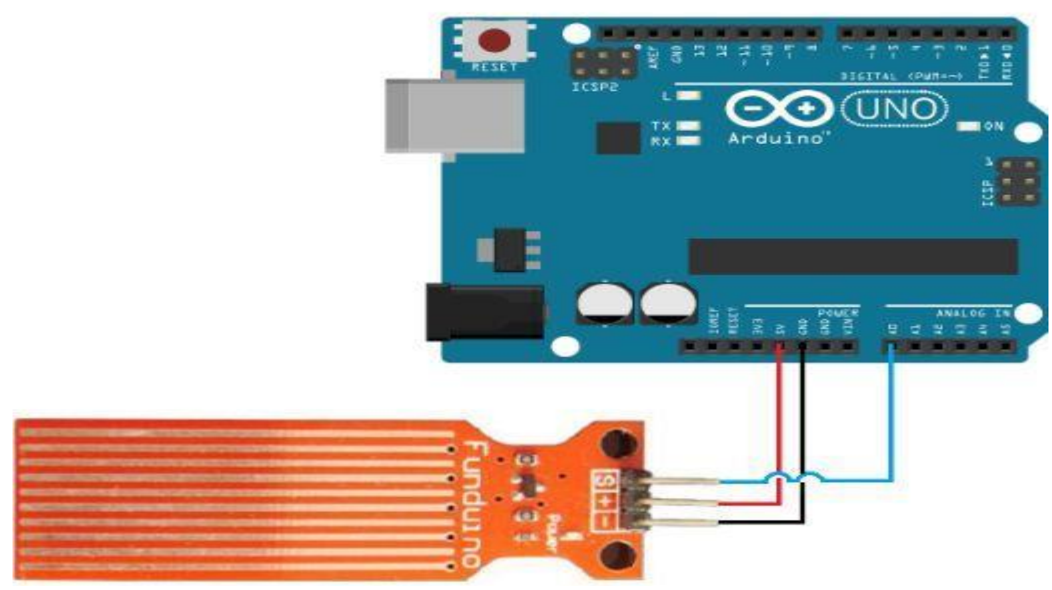


Figure (3.13) Interface water sensor with Arduino

3.3 Circuit of stick

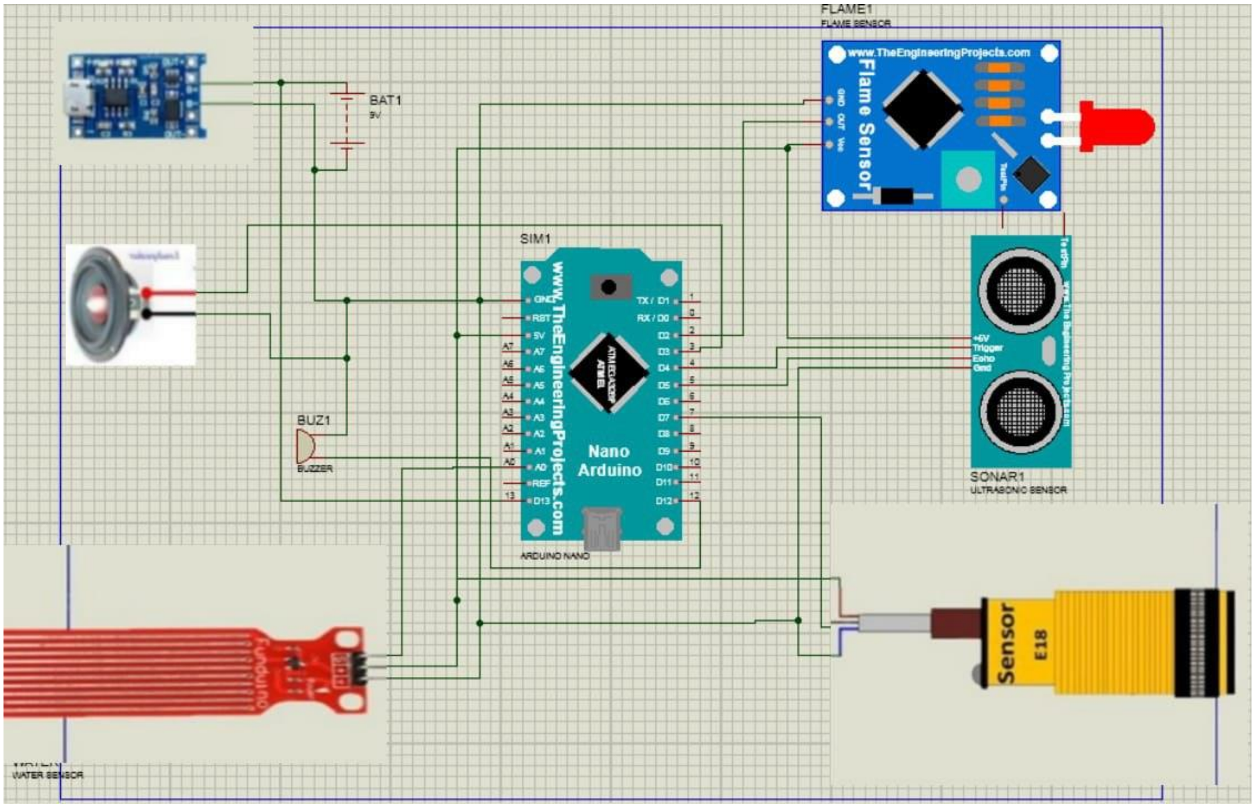


Figure (3.14) Circuit of stick

Chapter 4

Smart hat

4.1 Introduction

Science and technology always try to make human life easier. The term blindness is used for complete or nearly complete vision loss .Visual impairment may cause people difficulties with normal daily activities such as driving, reading, socializing, and walking.

The smart cap narrative about the surrounding environment while establishing communication between the user and a knowledge based system capable of vocalizing narratives.

The aim of the device is to provide the assistive technology to help the visually impaired people and blind people .

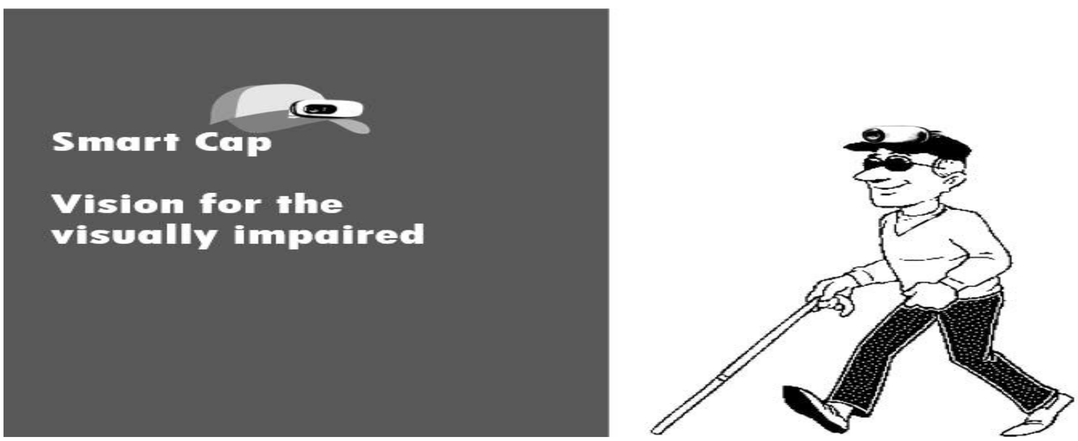


Figure (4.1) smart cap

4.2 Hat requirements

4.2.1 Rusberry-pi

4.2.1.1 Introduction

The Raspberry pi is a single computer board with credit card size, that can be used for many tasks that your computer does, like games, word processing, spreadsheets and also to play HD video. It was established by the Raspberry pi foundation from the UK. It has been ready for public consumption since 2012 with the idea of making a low-cost educational microcomputer for students and children. The main purpose of designing the raspberry pi board is, to encourage learning, experimentation and innovation for school level students. The raspberry pi board is a portable and low cost. Maximum of the raspberry pi computers is used in mobile phones. In the 20th century, the growth of mobile computing technologies is very high, a huge segment of this being driven by the mobile industries. The 98% of the mobile phones were using ARM technology.

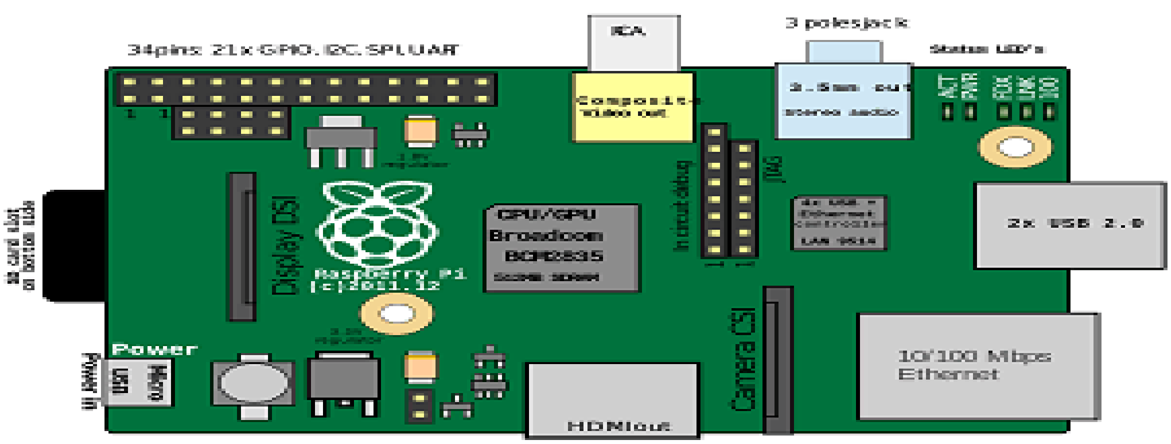


Figure (4.2) Raspberry pi Board

4.2.1.2 Raspberry Pi Technology

The raspberry pi comes in two models, they are model A and model B. The main difference between model A and model B is USB port. Model A board will consume less power and that does not include an Ethernet port. But the model B board includes an Ethernet port and designed in china. The raspberry pi comes with a set of open source technologies, i.e. communication and multimedia web technologies. In the year 2014, the foundation of the raspberry pi board launched the computer module that packages a model B raspberry pi board into module for use as a part of embedded systems, to encourage their use.

4.2.1.3 Raspberry Pi Hardware Specifications

The raspberry pi board comprises a program memory (RAM), processor and graphics chip, CPU, GPU, Ethernet port, GPIO pins, Xbee socket, UART, power source connector. And various interfaces for other external devices. It also requires mass storage, for that we use an SD flash memory card. So that raspberry pi board will boot from this SD card similarly as a PC boots up into windows from its hard disk.

Essential hardware specifications of raspberry pi board mainly include SD card containing Linux OS, US keyboard, monitor, power supply and video cable. Optional hardware specifications include USB mouse, powered USB hub, case, internet connection, the Model A or B: USB WiFi adaptor is used and internet connection to Model B is LAN cable.

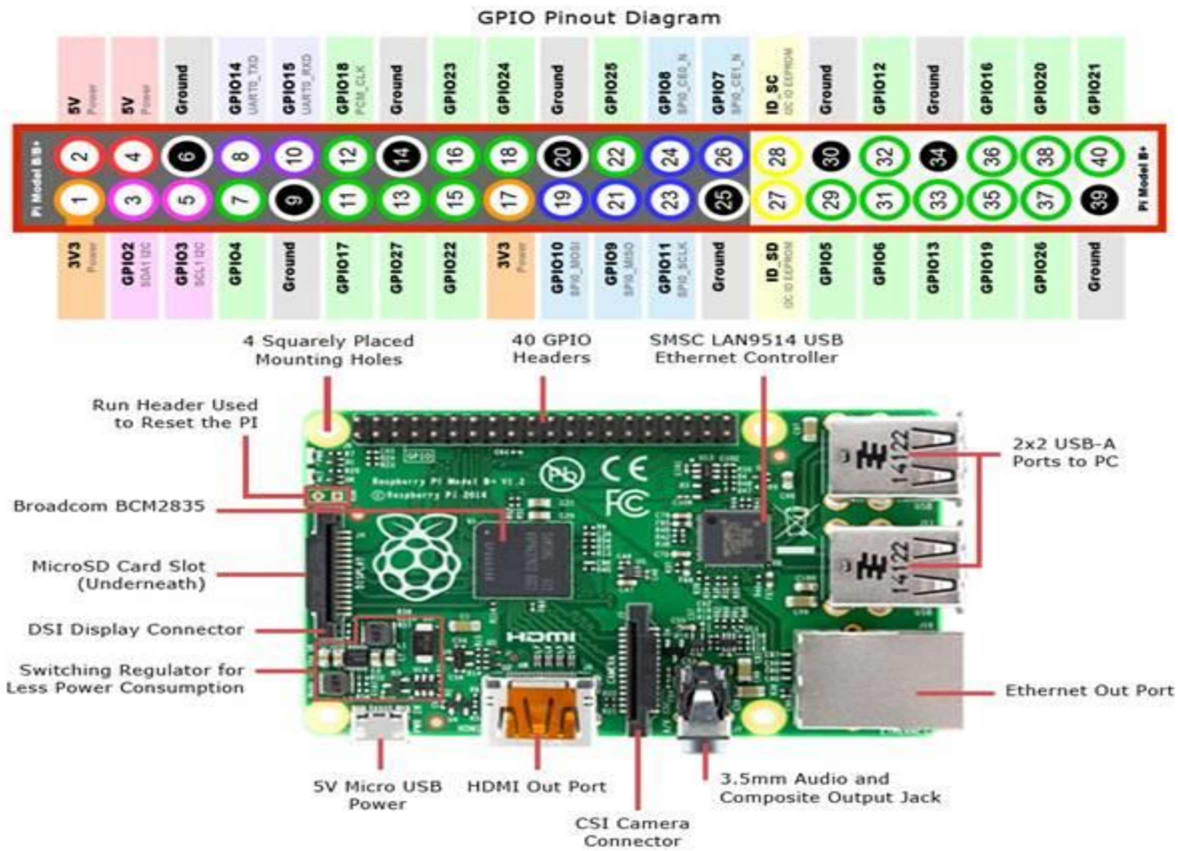


Figure (4.3) Hardware Specifications of Raspberry pi

Memory:

The raspberry pi model A board is designed with 256MB of SDRAM and model B is designed with 51MB. Raspberry pi is a small size PC compare with other PCs. The normal PCs RAM memory is available in gigabytes. But in raspberry pi board, the RAM memory is available more than 256MB or 512MB.

CPU (Central Processing Unit):

The Central processing unit is the brain of the raspberry pi board and that is responsible for carrying out the instructions of the computer through logical and mathematical operations. The raspberry pi uses ARM11 series processor, which has joined the ranks of the Samsung galaxy phone.

GPU (Graphics Processing Unit):

The GPU is a specialized chip in the raspberry pi board and that is designed to speed up the operation of image calculations. This board designed with a Broadcom video core IV and it supports OpenGL

Ethernet Port:

The Ethernet port of the raspberry pi is the main gateway for communicating with additional devices. The raspberry pi Ethernet port is used to plug your home router to access the internet.

GPIO Pins:

The general purpose input & output pins are used in the raspberry pi to associate with the other electronic boards. These pins can accept input & output commands based on programming raspberry pi. The raspberry pi affords digital GPIO pins. These pins are used to connect other electronic components. For example, you can connect it to the temperature sensor to transmit digital data.

If you're using the Raspberry Pi B+, 2, Zero, 3 or the latest Raspberry Pi 4 Model B, you'll discover that your board have a total of 40 GPIO pins. The older iterations of the RPI such as the Raspberry Pi Model B will only contain a total of 26pins.

- Any of the designated GPIO pins can be used either as a input/output pin, with a wide variety of applications suitable

Raspberry Pi GPIO Pinout	Pin Functionality and Explanation
GPIO	GPIO pins seen above are your standard pins to be used for turning devices on and off. E.g. When connected to an LED source
Power Pins (+)	Two 5V and two 3V3 pins are present on the board to draw power from the Raspberry Pi
I2C Pins	I2C pins are primarily used for connecting and hardware communication purposes for external modules that support such protocol. Such I2C communication typically uses 2 pins
SPI	SPI (Serial Peripheral Interface Bus) pins share similar functionality as I2C, for hardware communication purposes The only difference between both is that it uses a different protocol
UART	UART (Universal asynchronous receiver/trasmitter) pins are used for serial communication with other devices
DNC	DNC pins refer to pins that you shouldn’t connect, as its name (Do Not Connect) suggest

GND	GND (Ground) pins refer to pins to ground your devices. As they are all connected in the same line, it won't matter which GND pin you use to ground.

II) **Table (4.1)** GPIO Pins

XBee Socket:

The XBee socket is used in raspberry pi board for the wireless communication purpose.

Power Source Connector:

The power source cable is a small switch, which is placed on side of the shield. The main purpose of the power source connector is to enable an external power source.

UART:

The Universal Asynchronous Receiver/ Transmitter is a serial input & output port. That can be used to transfer the serial data in the form of text and it is useful for converting the debugging code.

Display:

The connection options of the raspberry pi board are two types such as HDMI and Composite. Many LCD and HD TV monitors can be attached using an HDMI male cable and with a low-cost adaptor. The versions of HDMI are 1.3 and 1.4 are supported and 1.4 version cable is recommended. The O/Ps of the Raspberry Pi audio and video through HMDI, but does not support HDMI I/p. Older TVs can be connected using composite video. When using a composite video connection, audio is available from the 3.5mm jack socket and can be sent to your TV. To send audio to your TV, you need a cable which adjusts from 3.5mm to double RCA connectors.

4.2.1.4 Model A Raspberry Pi Board

The Raspberry Pi board is a Broadcom(BCM2835) SOC(system on chip) board. It comes equipped with an ARM1176JZF-S core CPU, 256 MB of SDRAM and 700 MHz. The raspberry pi USB 2.0 ports use only external data connectivity options. The board draws its power from a micro USB adapter, with min range of 2 Watts (500 MA). The graphics, specialized chip is designed to speed up the operation of image calculations. This is in built with Broadcom video core IV cable that is useful if you want to run a game and video through your raspberry pi.

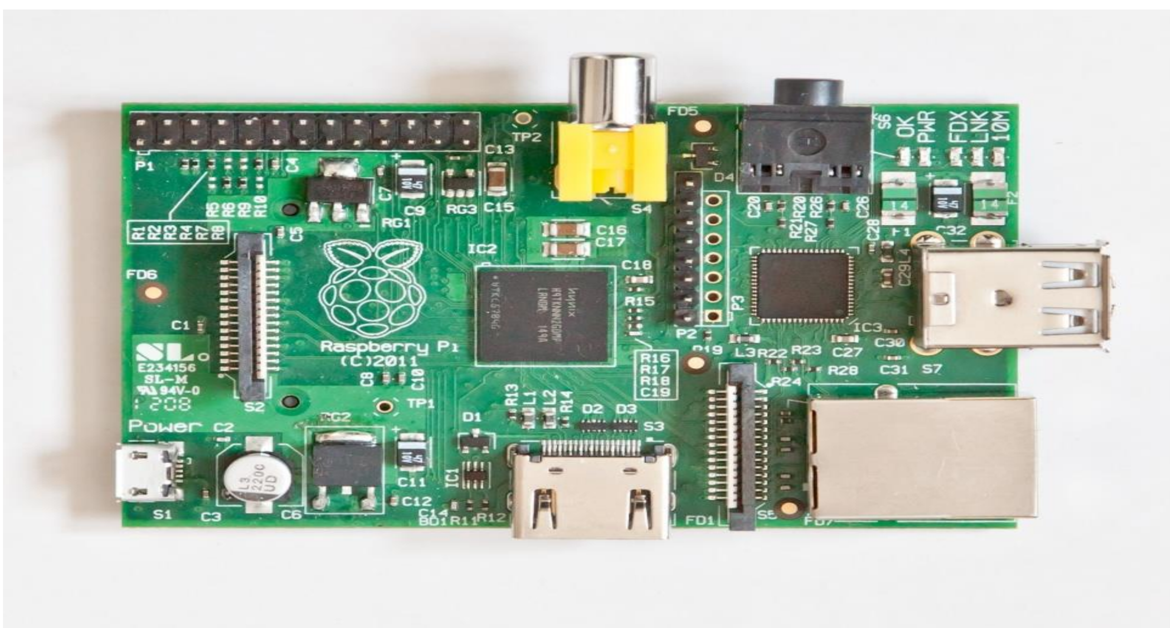


Figure (4.4) Features of Raspberry PI Model A

4.2.1.4.1 The Model A raspberry pi features mainly includes

- 256 MB SDRAM memory
- Single 2.0 USB connector
- Dual Core Video Core IV Multimedia coprocessor
- HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC) Video Out
- 3.5 MM Jack, HDMI, Audio Out
- SD, MMC, SDIO Card slot on board storage
- Linux Operating system
- Broadcom BCM2835 SoC full HD multimedia processor
- 8.6cm*5.4cm*1.5cm dimensions

4.2.1.5 Model B Raspberry pi Board

The Raspberry Pi is a Broadcom BCM2835 SOC (system on chip board). It comes equipped with a 700 MHz, 512 MB of SDRAM and ARM1176JZF-S core CPU. The USB 2.0 port of the raspberry pi boars uses only external data connectivity options. The Ethernet in the raspberry pi is the main gateway to interconnect with other devices and the internet in model B. This draws its power from a micro USB adapter, with a minimum range of 2.5 watts(500 MA). The graphics, specialized chip is designed to speed up the manipulation of image calculations. This is in built with Broadcom video core IV cable that is useful if you want to run a game and video through your raspberry pi.

Figure (4.5) Model B Raspberry pi Board

4.2.1.5.1 Features of Raspberry PI Model B

- 512 MB SDRAM memory
- Broadcom BCM2835 SoC full high definition multimedia processor
- Dual Core Video Core IV Multimedia coprocessor
- Single 2.0 USB connector
- HDMI Dimensions are 8.6cm*5.4cm*1.7cm
- On board 10/100 Ethernet RJ45 jack
- (rev 1.3 and 1.4) Composite RCA (PAL & NTSC) Video Out
- 3.5 MM Jack, HDMI Audio Out
- MMC, SD, SDIO Card slot on board storage
- Linux Operating system

4.2.1.6 Applications of Raspberry Pi

The raspberry pi boards are used in many applications like Media streamer, Arcade machine, Tablet computer, Home automation, Carputer, Internet radio, Controlling robots, Cosmic Computer, Hunting for meteorites, Coffee and also in raspberry pi based projects.

Raspberry Pi based Motor Speed Control

The main intention of this project is to control the speed of a DC Motor using Raspberry Pi.

4.2.1.7 List of Raspberry Pi Based Projects

- Programmable Sequential Switching by Using Raspberry pi
- Raspberry Pi Based Solar Street Light
- Synced Music and Christmas Lights
- Raspberry pi Powered Wearable Computer
- Home Automation Using Raspberry pi

- Touch Screen Tablet
- Raspberry pi Based Industrial Automation Using Zigbee Communication
- Lego Raspberry pi Enclosure
- Raspberry pi as an FM Transmitter
- Autonomous Robot Arm Control Based on Raspberry pi with Bluetooth Control

4.2.1.8 The differences between Raspberry Pi and Arduino

SL	Raspberry Pi	Arduino
1	It is a mini computer with Raspbian OS .It can run multiple programs at a time.	Arduino is a microcontroller, which is a part of the computer. It runs only one program again and again.
2	It is difficult to power using a battery pack.	Arduino can be powered using a battery pack.
3	It requires complex tasks like installing libraries and software for interfacing sensors and other components	It is very simple to interface sensors and other electronic components to Arduino.
4	It is expensive	It is available for low cost.

5	Raspberry Pi can be easily connected to the internet using Ethernet port and USB Wi-Fi dongles.	Arduino requires external hardware to connect to the internet and this hardware is addressed properly using code.
6	Raspberry Pi did not have storage on board. It provides an SD card port.	Arduino can provide onboard storage.
7	Raspberry Pi has 4 USB ports to connect different devices.	Arduino has only one USB port to connect to the computer.
8	The processor used is from ARM family.	Processor used in Arduino is from AVR family Atmega328P
9	This should be properly shutdown otherwise there is a risk of files corruption and software problems.	This is a just plug and play device. If power is connected it starts running the program and if disconnected it simply stops.
10	The Recommended programming language is python but C, C++, Python, ruby are pre-installed.	Arduino uses Arduino, C/C++.

III) **Table (4.2)** The differences between Raspberry Pi and Arduino

4.2.2 Camera

4.2.2.1 Image processing

4.2.2.1.1 Introduction

Signal processing is a discipline in electrical engineering and in mathematics that deals with analysis and processing of analog and digital signals and deals with storing, filtering, and other operations on signals. These signals include transmission signals, sound or voice signals, image signals, and other signals e.t.c.

Out of all these signals, the field that deals with the type of signals for which the input is an image and the output is also an image is done in image processing. As it name suggests it deals with the processing on images.

It can be further divided into analog image processing and digital image processing.

Digital image processing is a technical means used in this research, and it is one of the key factors influencing our research progress. The understanding of digital image processing may be different for different disciplines. Therefore, it is necessary at first to give a brief introduction to digital image processing technology and related concepts.
Digital image processing is to process images by computer. Digital image processing can be defined as subjecting a numerical representation of an object to a series of operations in order to obtain a desired result. Digital image processing consists of the conversion of a physical image into a corresponding digital image and the extraction of significant information from the digital image by applying various algorithms. Digital image processing mainly includes image collection, image processing, and image analysis. At its most basic level, a digital image processing system is comprised of three components, ie, a computer system on which to process images, an image digitizer, and an image display device.

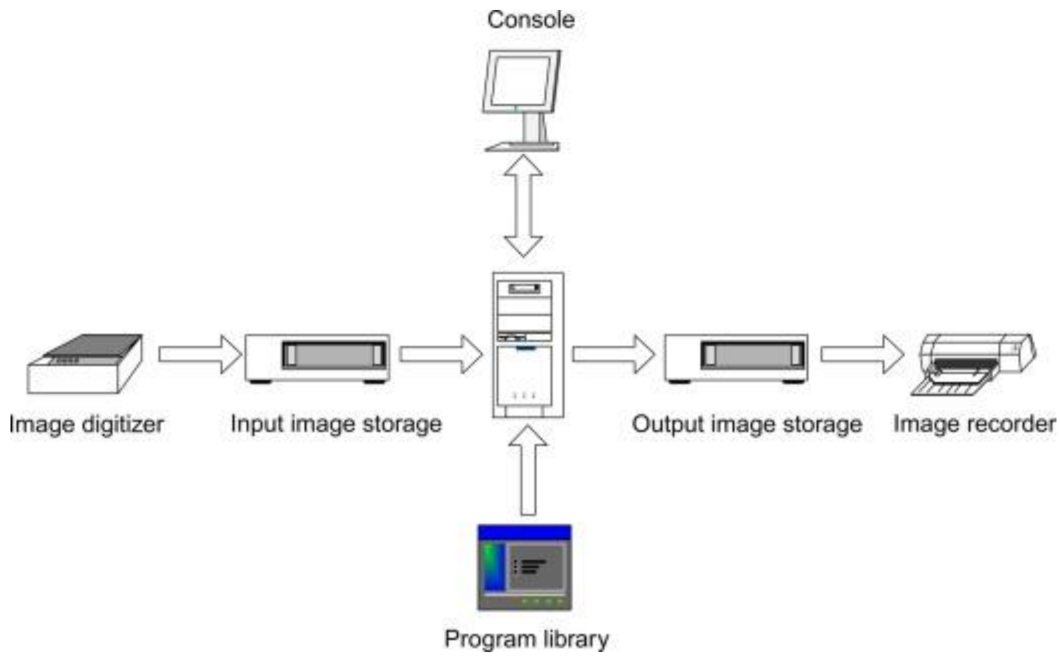


Figure (4.6) complete system for image processing.

Physical images are divided into small areas called pixels. The division plan used often is the rectangular sampling grid method shown in Fig. 13.6, in which an image is segmented into many horizontal lines composed of adjacent pixels, and the value of each pixel position reflects the brightness of corresponding point on the physical image.

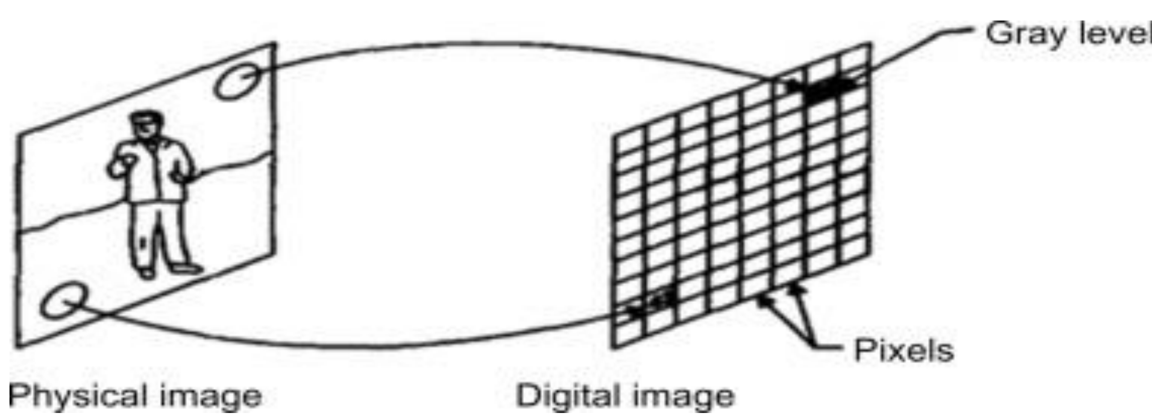


Figure (4.7) Physical image and corresponding digital image.

Physical images cannot be directly analyzed by a computer because the computer can only process digits rather than images, so an image must be converted into a digital form before processed by a computer. The conversion process is called digitization

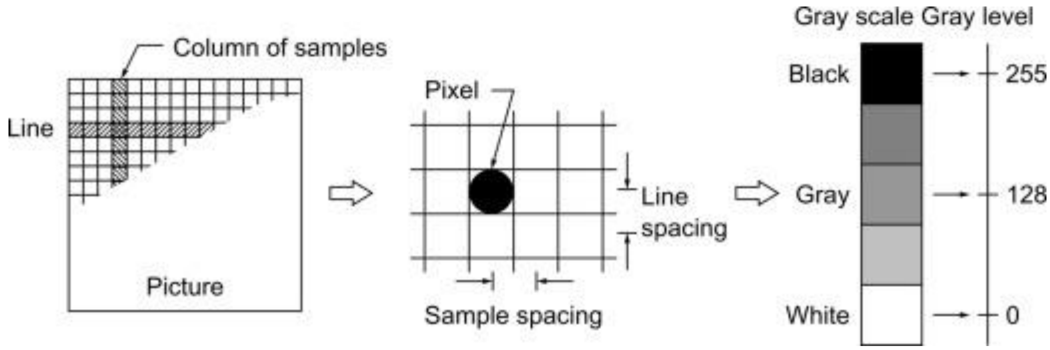


Figure (4.8) Digitizing image.

4.2.2.1.2 Image processing mainly include the following steps

1. Importing the image via image acquisition tools.
2. Analysing and manipulating the image.
3. Output in which result can be altered image or a report which is based on analysing that image.

4.2.2.1.3 What is an image?

An image is defined as a two-dimensional function F(x,y) where x and y are spatial coordinates, and the amplitude of F at any pair of coordinates (x,y) is called the intensity of that image at that point. When x,y, and amplitude values of F are finite, we call it a **digital image**.
In other words, an image can be defined by a two-dimensional array specifically arranged in rows and columns.
Digital Image is composed of a finite number of elements, each of which elements have a particular value at a particular location. These elements are referred to as *picture elements, image elements and pixels*. A *Pixel* is most widely used to denote the elements of a Digital Image.

4.2.2.1.4 Applications of Digital Image Processing

Some of the major fields in which digital image processing is widely used are mentioned below

- Image sharpening and restoration
- Medical field
- Remote sensing
- Transmission and encoding
- Machine/Robot vision
- Color processing
- Pattern recognition
- Video processing
- Microscopic Imaging
- Others

4.2.2.2 The Raspberry Pi Camera

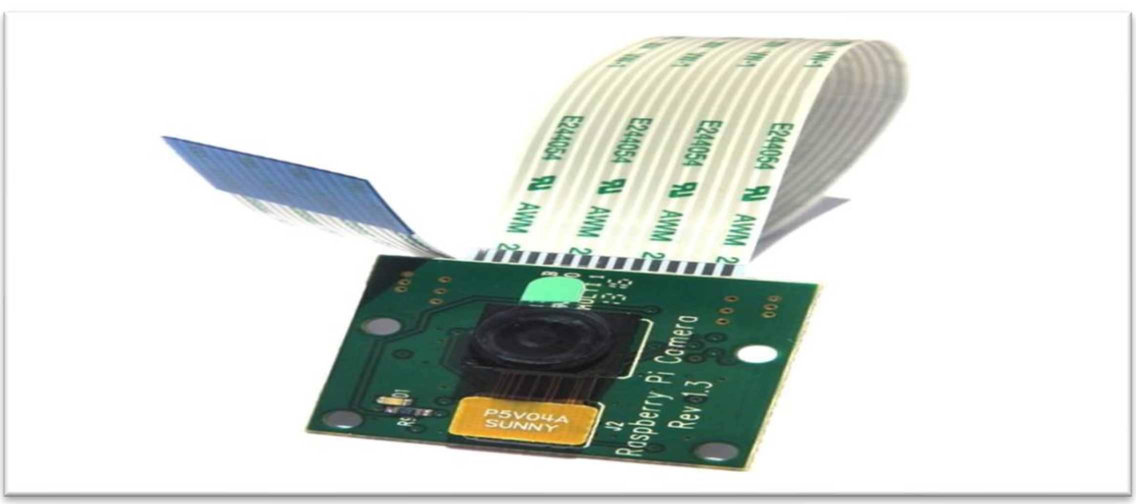


Figure (4.9) The Raspberry Pi Camera

The Raspberry Pi Camera Board plugs directly into the CSI connector on the Raspberry Pi. It's able to deliver a crystal clear 5MP resolution image, or 1080p HD video recording at 30fps! Latest Version 1.3! Custom designed and manufactur ed by the Raspberry Pi Foundation in the UK, the Raspberry Pi Camera Board features a 5MP (2592?1944 pixels) Omnivision 5647 sensor in a fixed focus module. The module attaches to Raspberry Pi, by way of a 15 Pin Ribbon Cable, to the dedicated 15-pin MIPI Camera Serial Interface (CSI), which was designed especially for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to the BCM2835 processor. The board itself is tiny, at around 25mm x 20mm x 9mm, and weighs just over 3g, making it perfect for mobile or other applications where size and weight are important. The sensor itself has a native resolution of 5 megapixel, and has a fixed focus lens onboard. In terms of still images, the camera is capable of 2592 x 1944 pixel static images, and also supports 1080p @ 30fps, 720p @ 60fps and 640x480p 60/90 video recording. The camera is supported in the latest version of Raspbian, the Raspberry Pi's preferred operating system.

4.2.2.3 The Raspberry Pi Camera Board Features:

- Fully Compatible with Both the Model A and Model B Raspberry Pi
- 5MP Omnivision 5647 Camera Module
- Still Picture Resolution: 2592 x 1944
- Video: Supports 1080p @ 30fps, 720p @ 60fps and 640x480p 60/90 Recording
- 15-pin MIPI Camera Serial Interface - Plugs Directly into the Raspberry Pi Board
- Size: 20 x 25 x 9mm
- Weight 3g
- Fully Compatible with many Raspberry Pi cases

4.2.3 IC LM386 &Speaker

4.2.3.1 IC LM386

The IC LM386 is a low-power audio amplifier, and it utilizes low power supply like batteries in electrical and electronic circuits. This IC is available in the package of mini 8-pin DIP. The voltage gain of this amplifier can be adjusted to 20, and the voltage gain will be enhanced to 200 by employing external components like resistors as well as capacitors among the pins 1 & 8. When this amplifier uses a 6V power supply for the operation then the static power drain will be 24 milliwatts to make the amplifier for an ultimate operation of the battery. This amplifier consists of 8-pins where pin-1 and pin-8 are gain control pins of the amplifier, and this IC is most widely used IC that allows a customer to increase volume.

The IC LM386 audio amplifier consists of 8-pins where each pin of this IC is discussed below.

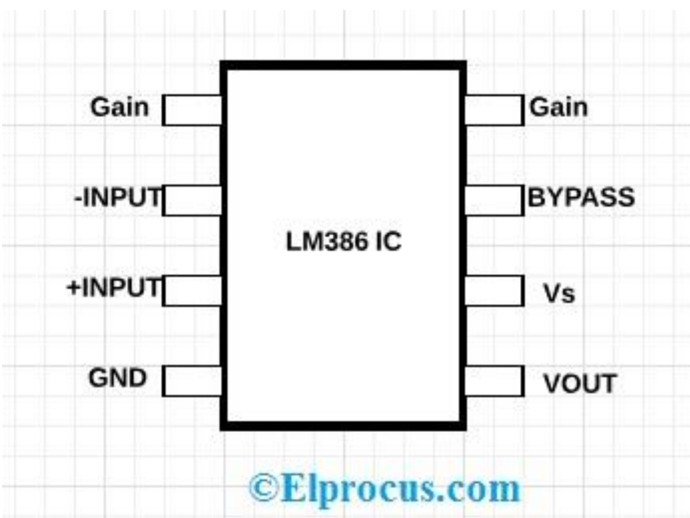


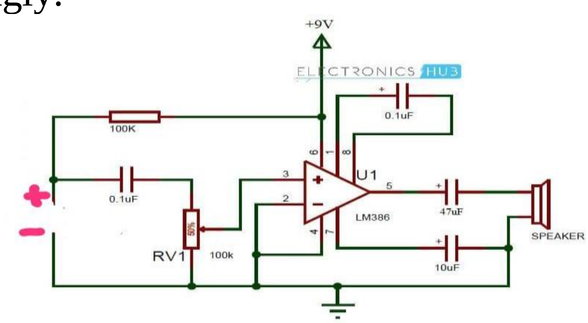
Figure (4.10) IC LM386 Pin Configuration

4.2.3.1.1 IC LM386 Pin Configuration

- Pin1 (Ga+-gain Pin): Pin-1 is gain pin, used adjust the amplifier gain by connecting this IC to an external component capacitor.
- Pin2 (+IN-Non-inverting): Pin-2 is the non-inverting pin, is used to provide the audio signal.
- Pin3 (+IN): Pin-3 is the inverting terminal and it is normally connected to ground.
- Pin4 (GND): Pin-4 is a ground pin connected to the ground terminal of the system
- Pin5 (Vout): Pin-5 is the output pin, used to provide amplified output audio, and allied to the speaker.
- Pin-6 (VCC or VSS): Pin-6 is connected to the power
- Pin-7 (Bypass): Pin-7 bypass pin is used to connect a decoupling capacitor.
- Pin-8 (Gain): Pin-8 is the gain setting pin

4.2.3.1.2 LM386 Audio Amplifier Circuit Diagram and Working

The Audio amplifier can be built with LM386 IC, capacitors like 10 µF, 47 µF, 0.1 µF, Potentiometer – 100 KΩ, resistor-100 KΩ, power supply 9V, speaker-4Ω, breadboard, and connecting wires. Basically, this Audio Amplifier includes 3-functional blocks such as Power as well as Output, Bypass, gain control. Designing of this circuit design is so simple. At first, connect the two power supply pins namely pin4 & pin6 to GND as well as voltage correspondingly.



Receiver Circuit

Figure (4.11) IC LM386 Audio Amplifier Circuit

After that, connect the input from any kind of audio sources such as a mobile phone or a microphone. Here this circuit uses a mobile phone as the audio source with the help of the 3.5mm connector. This connector will be having three connections like the ground right and left audio. This LM386 IC is a simple amplifier and connects the right or left audio to this amplifier using an audio source with the ground terminal. The input level in this circuit can be controlled by connecting a potentiometer to the input. In addition, a capacitor will be connected to the input in series to remove the DC components. This IC gain will be adjusted to 20, and connect a capacitor (0.1 µF) between the two pins 1 & 8 of this IC then the gain will be enhanced to 200

Even though the datasheet of audio amplifier advises the by pass capacitor at the 7th pin is an option, we form that connecting a capacitor (10 µF) was truly helpful because it assists in the noise reduction. For the connection of output, a capacitor (47 µF) will be connected in series among the capacitor (10 µF) as well as a 5th pin of the IC. This forms a Zobel network, a filter including a capacitor and resistor will be utilized for adjusting the input impedance.
The speaker connection can be done with the help of impedance ranges from 4 Ω to 32 Ω, because the IC can drive any type of speaker in this range. The audio amplifier circuit uses a speaker (4 Ω). This speaker can be connected using a capacitor (47 µF) was really useful because it removes the un necessary DC signals.

4.2.3.1.3 Electrical Characteristics of LM386 IC

- The voltage gain of this amplifier can be set from 20 to 200 with a range of voltage supply 4volts to 12volts or 5volts to 18 volts based on the model. There are three amplifier models are available in the market namely LM386N-1, LM386N-3, & LM386N-4

- For LM386N-1: Minimum voltage is 4V, Maximum voltage is 12V, Minimum o/p power is 250 mW and typical o/p power is 325mW.
- For LM386N-3: Minimum voltage is 4V, Maximum voltage is 12V, Minimum o/p power is 500 mW and typical o/p power is 700mW.
- For LM386N-4: Minimum voltage is 5V, Maximum voltage is 18V, Minimum o/p power is 500 mW and typical o/p power is 1000mW.
- The inputs of the amplifier are referenced by ground whereas the output routinely biases toward one half of the voltage supply. The low static current of the amplifier is 4mA and the harmonic distortion will be up to 0.2%

4.2.3.1.4 Features of IC LM386

The main features of LM386 chip include the following.

- IC LM386 is obtainable in the package of 8-pin MSOP
- Exterior components are minimum
- Operation of Battery
- Low static power drain- 4mA
- The range of supply voltage is wide which is ranges from 4Volts to 12Volts or 5Volts to 18 Volts.
- Input is referenced by ground
- Distortion is less 0.2%
- Self-centering o/p static voltage
- The voltage gain range will be from 20 to 200

4.2.3.1.5 LM386 Applications

The IC LM386 is the most important integrated circuit used in the audio section, and it is commonly used in the following applications.

- Wien bridge oscillator
- Power converters
- Ultrasonic drivers
- Small servo drivers
- Intercoms
- Line drivers

- TV sound systems
- Portable tape player amplifiers
- AM to FM radio amplifiers
- Audio boosters
- Used in speakers of laptop & portable
- Used for voice record from microphone, battery operated speakers.

Thus, this is all about IC LM386, and this article gives an overview of how to design an IC LM386 audio amplifier circuit, and the making of this circuit is very simple, small size as well as less cost. So the sound from this amplifier will be very loud. There are various kinds of amplifiers can be built with the help of IC LM386, but the main drawback of this circuit is interference as well as noise. The proposed system can be designed with less noise. This circuit will deliver a 1 Watt of power & can be applied in a wide range of audio devices like speakers in laptops, handy speakers, etc

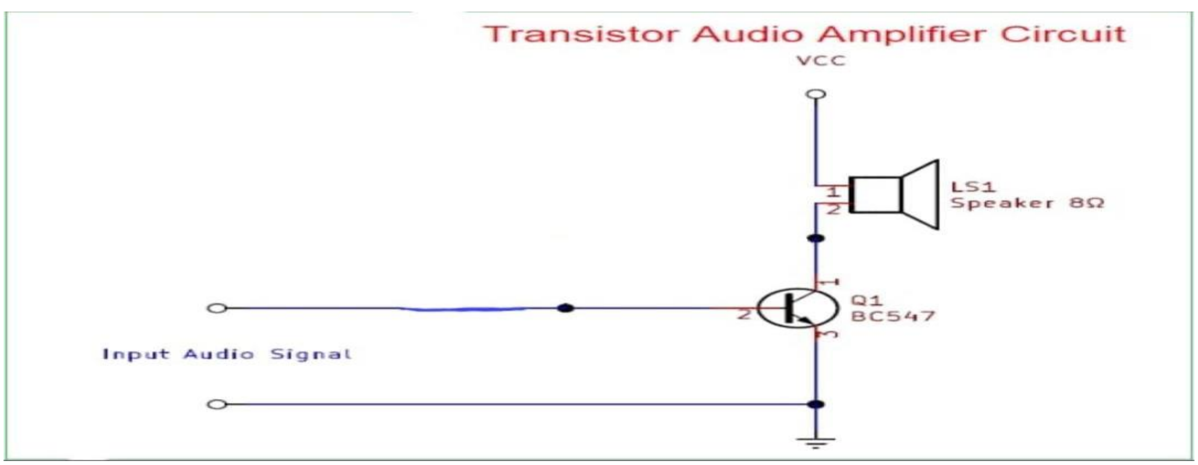


Figure (4.12) Transistor Audio Amplifier Circuit

4.2.3.2 Speaker

Speakers are one of the most common output devices used with computer systems. Some speakers are designed to work specifically with computers, while others can be hooked up to any type of sound system. Regardless of their design, the purpose of speakers is to produce audio output that can be heard by the listener. Speakers are transducers that convert electromagnetic waves into sound waves. The speakers receive audio input from a device such as a computer or an audio receiver.

This input may be either in analog or digital form. Analog speakers simply amplify the analog electromagnetic waves into sound waves. Since sound waves are produced in analog form, digital speakers must first convert the digital input to an analog signal, then generate the sound waves.The sound produced by speakers is defined by frequency and amplitude. The frequency determines how high or low the pitch of the sound is

For example, a soprano singer's voice produces high frequency sound waves, while a bass guitar or kick drum generates sounds in the low frequency range. A speaker system's ability to accurately reproduce sound frequencies is a good indicator of how clear the audio will be. Many speakers include multiple speaker cones for different frequency ranges, which helps produce more accurate sounds for each range. Two-way speakers typically have a tweeter and a mid-range speaker, while three-way speakers have a tweeter, mid-range speaker, and subwoofer. Amplitude, or loudness, is determined by the change in air pressure created by the speakers' sound waves. Therefore, when you crank up your speakers, you are actually increasing the air pressure of the sound waves they produce. Since the signal produced by some audio sources is not very high (like a computer's sound card)

it may need to be amplified by the speakers. Therefore, most external computer speakers are amplified, meaning they use electricity to amplify the signal. Speakers that can amplify the sound input are often called active speakers. You can usually tell if a speaker is active if it has a volume control or can be plugged into an electrical outlet. Speakers that don't have any internal amplification are called passive speakers. Since these speakers don't amplify the audio signal, they require a high level of audio input, which may be produced by an audio amplifier. Speakers typically come in pairs , which allows them to produce stereo sound.

This means the left and right speakers transmit audio on two completely separate channels. By using two speakers, music sounds much more natural since our ears are used to hearing sounds from the left and right at the same time. Surround systems may include four to seven speakers (plus a subwoofer), which creates an even more realistic experience

4.2.4 Mobile Application

4.2.4.1 MIT App Inventor

MIT App Inventor is an intuitive, visual programming environment that allows everyone – even children – to build fully functional apps for smartphones and tablets. Those new to MIT App Inventor can have a simple first app up and running in less than 30 minutes. And what's more, our blocks-based tool facilitates the creation of complex, high-impact apps in significantly less time than traditional programming environments. The MIT App Inventor project seeks to democratize software development by empowering all people, especially young people, to move from technology consumption to technology creation.

A small team of CSAIL staff and students, led by Professor Hal Abelson, forms the nucleus of an international movement of inventors. In addition to leading educational outreach around MIT App Inventor and conducting research on its impacts, this core team maintains the free online app development environment that serves more than 6 million registered users.

Blocks-based coding programs inspire intellectual and creative empowerment. MIT App Inventor goes beyond this to provide real empowerment for kids to make a difference -- a way to achieve social impact of immeasurable value to their communities. In fact, App Inventors in school and outside of traditional educational settings have come together and done just that:

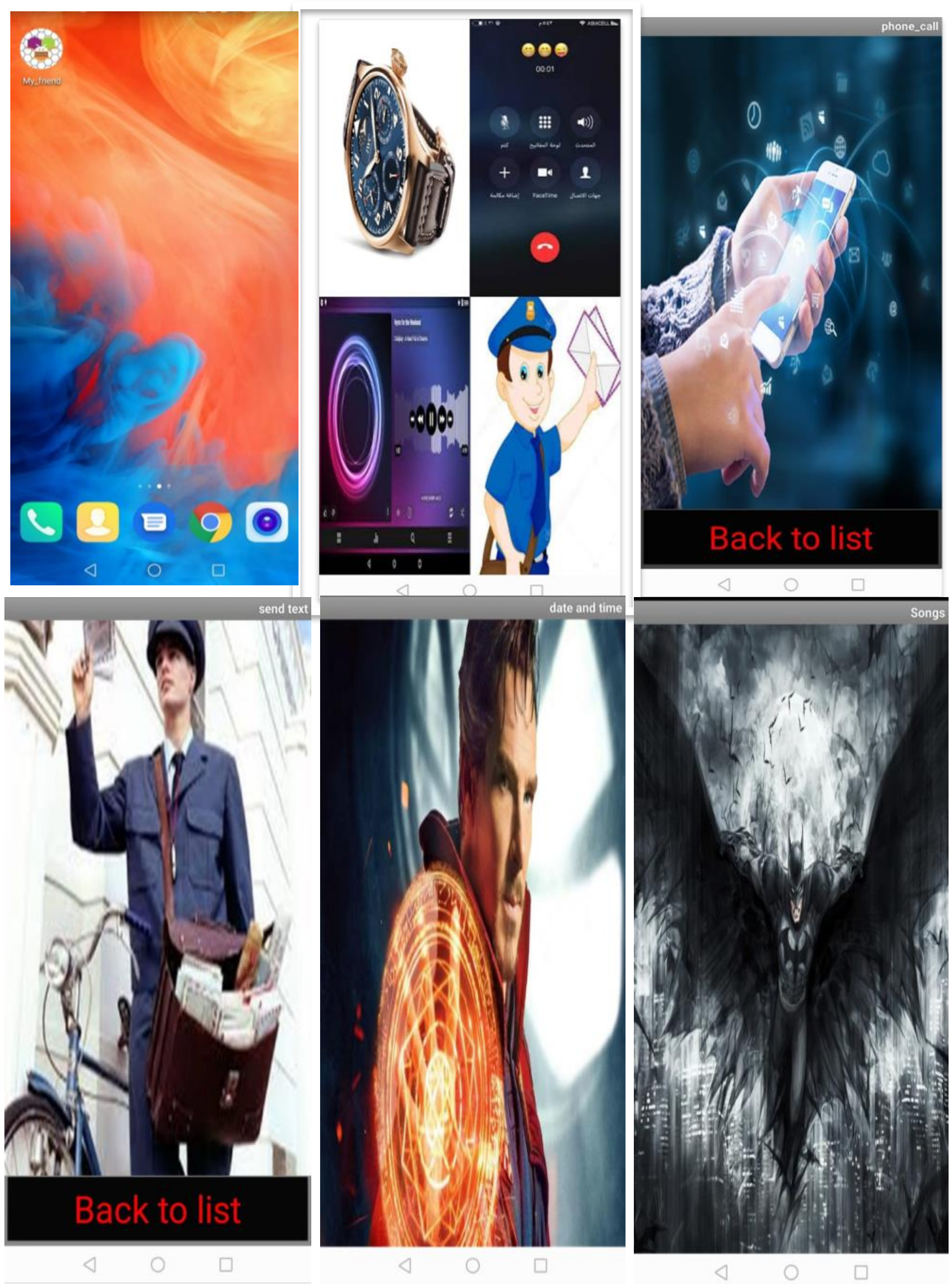
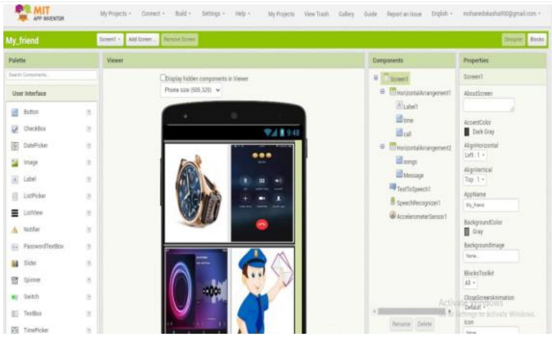
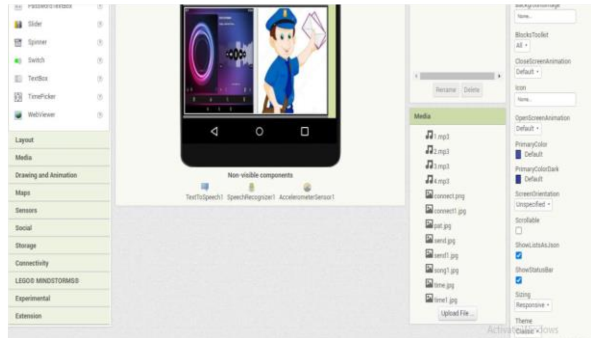
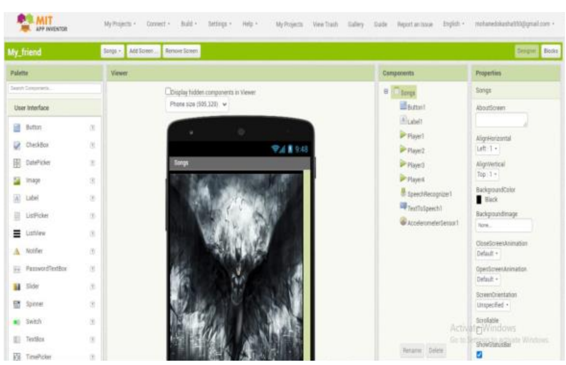
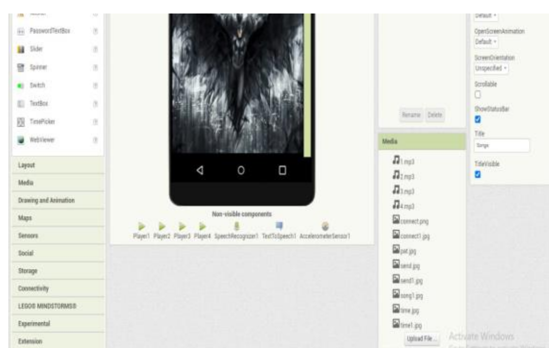


Figure (4.13) the Application on the phone

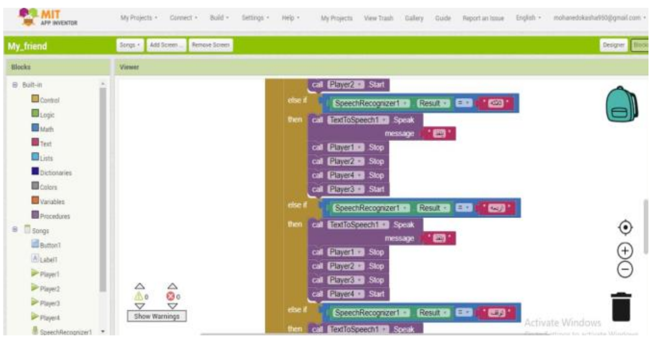
4.2.4.2 Code on MIT App Inventor

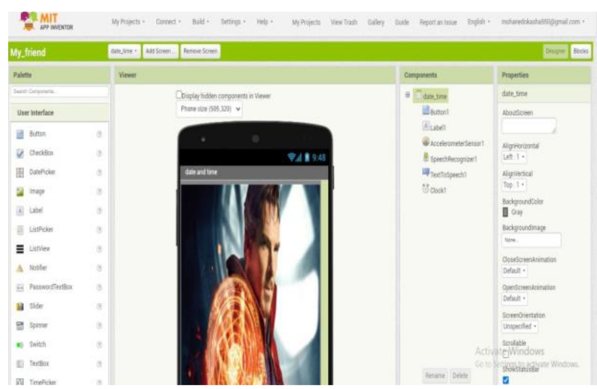
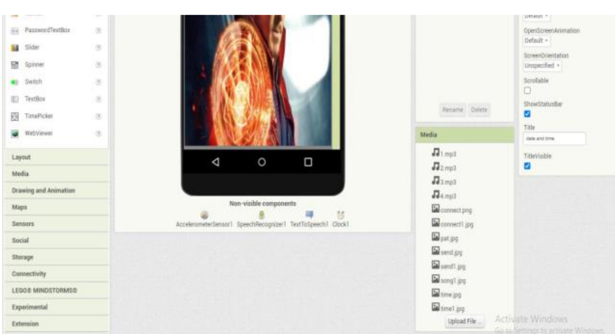







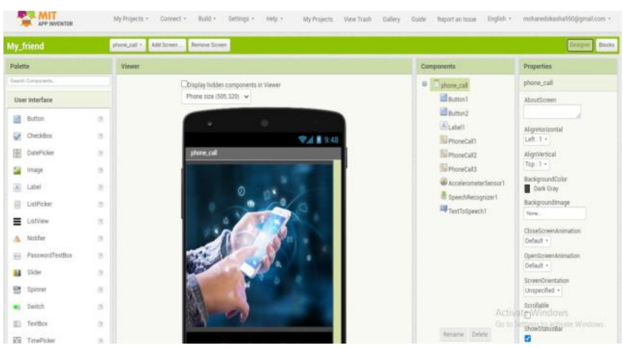


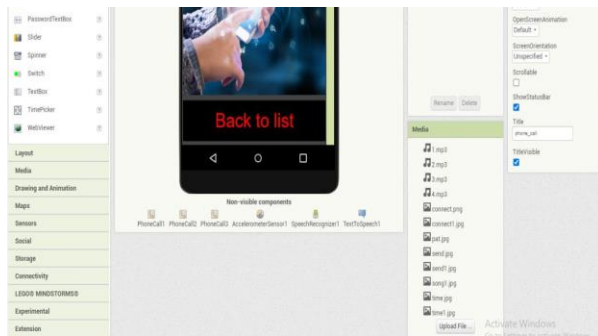






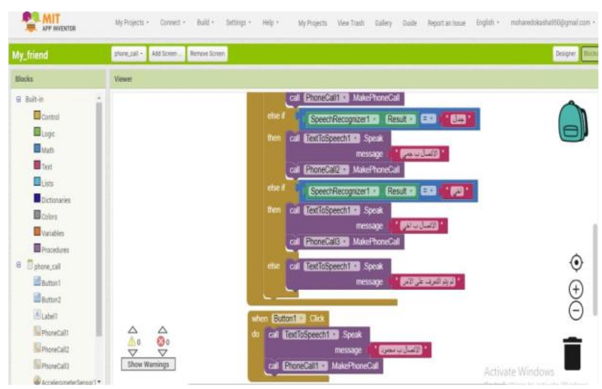
MIT APP INVENTOR
My_friend
Blocks
Viewer
Show Warnings


MIT APP INVENTOR
My_friend
Blocks
Viewer
Show Warnings

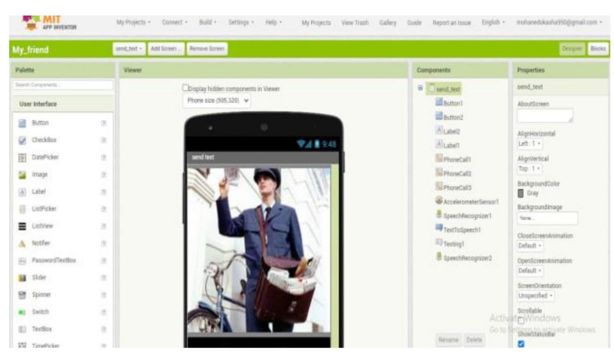

MIT APP INVENTOR
My_friend
Palette
Viewer
Components
Properties

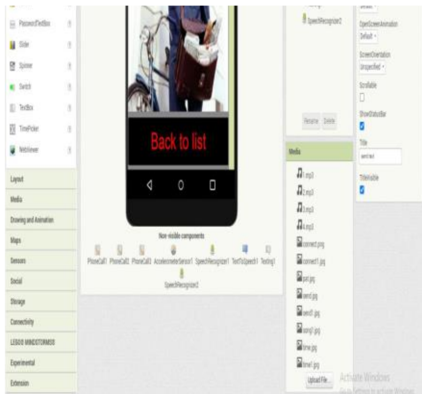
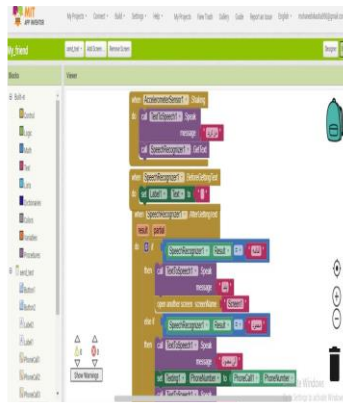
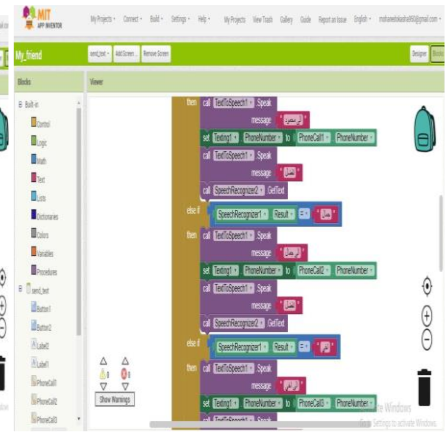

Back to list


MIT APP INVENTOR
My_friend
Blocks
Viewer
Show Warnings


MIT APP INVENTOR
My_friend
Blocks
Viewer
Show Warnings


MIT APP INVENTOR
My_friend
Blocks
Viewer
Show Warnings


MIT APP INVENTOR
My_friend
Palette
Viewer
Components
Properties

4.2.5 Circuit

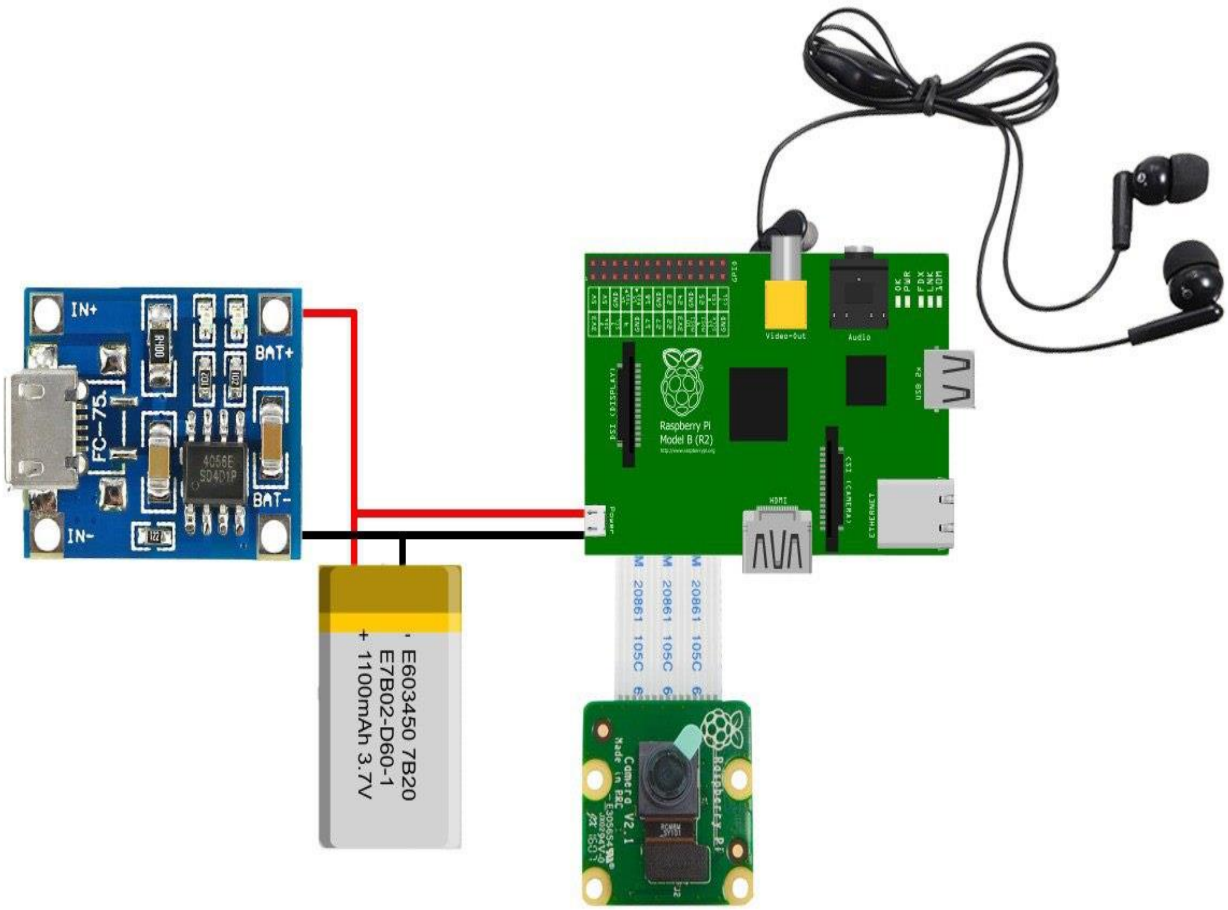


Figure (4.14) Circuit of hat

Chapter 5

Design

5.1 Design of stick

5.1.1 What is the problem facing us to resort to make our design?

Figure (5.1) Shapes of sticks

The shapes of sticks don't work what are required so we resort to design the stick for our project.

5.1.2 Programs that make design

5.1.2.1 ARNOLD

Figure (5.2) Arnold logo

Arnold is based on Monte Carlo Ray Tracing. Its engine is optimized to send billions of spatially incoherent rays throughout a scene. It often uses one level of diffuse inter-reflection so that light can bounce off of a wall or other object and indirectly illuminate a subject.

For complex scenes such as the space station in Elysium, it makes heavy use of instancing.

It uses the Open Shading Language to define the materials and textures.

5.1.2.2 MAYA

Figure (5.3) MAYA logo

Maya was originally an animation product based on code from The Advanced Visualizer by Wave front Technologies, Thomson Digital Image (TDI) Explore, Power Animator by Alias Research, Inc., and Alias Sketch.

The IRIX-based projects were combined and animation features were added, the project codename was Maya.

Walt Disney Feature Animation collaborated closely with Maya's development during its production of Dinosaur.

Disney requested that the user interface of the application be customizable so that a personalized workflow could be created. This was a particular influence in the open architecture of Maya, and partly responsible for it becoming popular in the animation industry.

After Silicon Graphics Inc. acquired both Alias and Wavefront Technologies, Inc., Wave front’s technology (then under development) was merged into Maya.

SGI's acquisition was a response to Microsoft Corporation acquiring Softimage 3D. The new wholly owned subsidiary was named "Alias".

In the early days of development, Maya started with Tcl as the scripting language, in order to leverage its similarity to a UNIX shell language. But after the merger with Wave front, Sophia, the scripting language in Wave front’s Dynamation, was chosen as the basis of MEL (Maya embedded language).

Maya 1.0 was released in February 1998. Following a series of acquisitions, Maya was bought by Autodesk in 2005 under the name of the new parent company, Maya was renamed Autodesk Maya. However, the name "Maya" continues to be the dominant name used for the product.

5.1.2.3 Substance

5.1.2.3.1 Designer

Figure (5.4) Substance Designer

The industry standard for creating custom materials, Substance Designer gives you complete authoring control.

Design tileable textures and patterns, and edit complete texture sets.

Use pre-made resources or create materials from scratch in this non-destructive, node-based environment, and take advantage of Designer’s ever-expanding scripting capabilities.

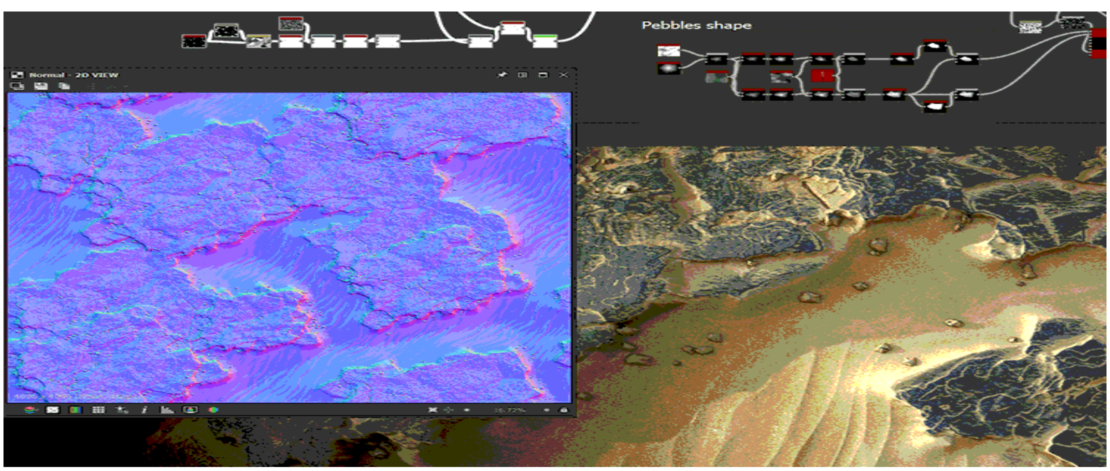


Figure (5.5) Edit materials instantly with the procedural non-linear workflow.

Want to change something or adjust the look of a material after the fact? Make changes at any time; never lose any of your work.

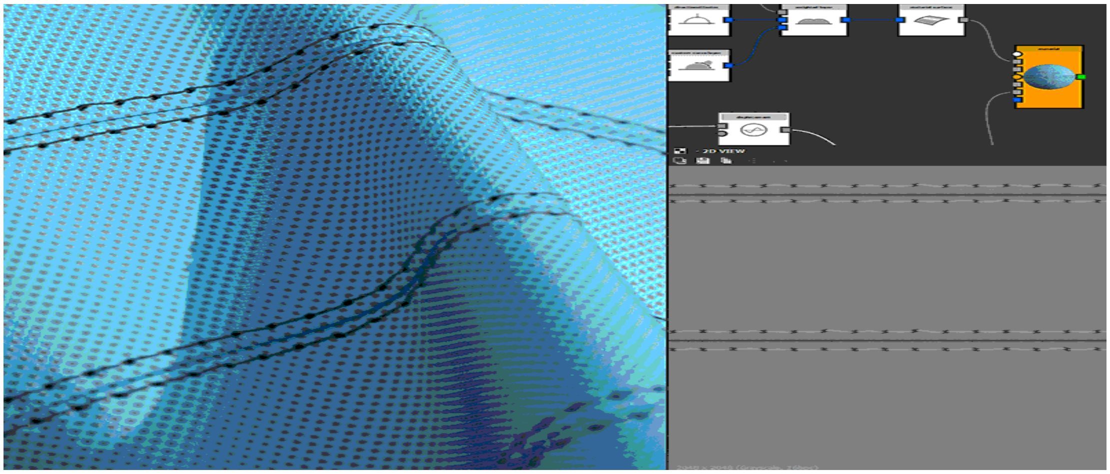


Figure (5.6) Create materials with full control and infinite power of expression

5.1.2.3.2 Painter

Figure (5.7) Substance Painter

The app for texturing your 3D assets.With this powerful app, you can paint 3D textures on your model in real time. You get smart materials that adjust to any object to show realistic wear and tear. And the workflow is nondestructive, so you can experiment all you want. Play with mask presets that adapt to any shape, and paint with brushes you can modify on the fly.

Figure (5.8) sample of what program make

This next-generation app lets you easily create collections of materials. You can mix and tweak existing materials, or make new ones from photographs and high-res scans. It's a go-to for workflows that require real-world references — and for those times when you just want to explore. Let Alchemist help you brainstorm and iterate.

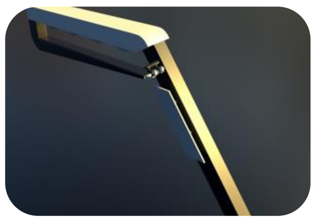
Figure (5.9) Heavy-duty 3D texture authoring tools.

The industry standard for creating custom materials, Substance Designer gives you complete authoring control. Design tile able textures and patterns, and edit complete texture sets. Create from scratch in this nondestructive, node-based environment, or take advantage of Designer's ever-expanding scripting capabilities. Design with precision — exactly how you envision it.

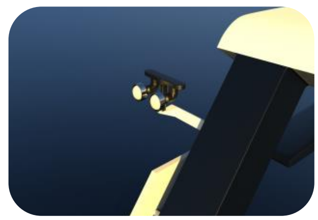
5.1.3 Improvement the designs of stick

5.2.3.1 First design

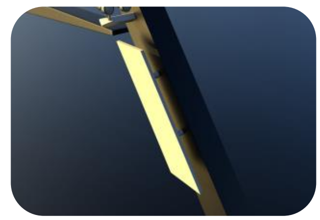
Stick



the hand of stick



Ultrasonic place



the box of circuit

Figure (5.10) first design

In this design all cit. be placed outside the stick. The outward shape has become inappropriate and the companent become object for damage and ruin.

5.2.3.2 Second design

Figure(5.11) second design

We made it hollow stick. Put all companent inside the stick to protect it, but the general shape become classic for the elderly and is somewhat unsuitable(not modern) for youth.

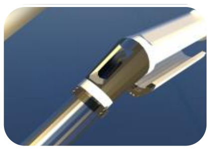
5.2.3.3 Third design

Figure (5.12) third design

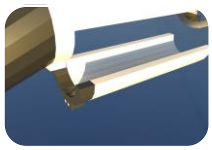
The shape of the stick become circular(improve), but its straight design and sensors need some inclination to work properly.

3.4.3.4 Forth design

cover of upper section of stick



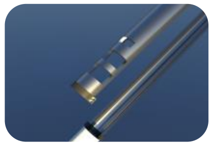
the PIR place



the cover of hand

power socket

stick



IR & Ultrasonic place

Figure(5.13) forth design

The hand is tilted at an angle of 25 from the stick to ensure that sensors work well and the shape is circular and modern (not traditional).

5.2 Design of Hat

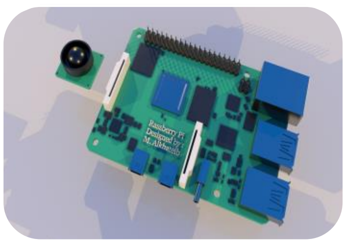


figure (5.14) Raspberry Pi circuit kit

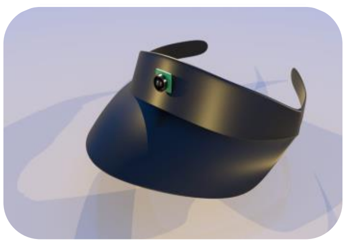


figure (5.15) the hat with camera

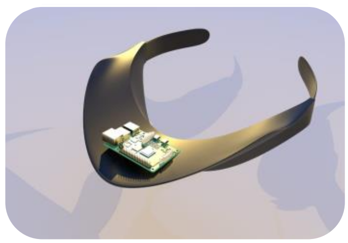


figure (5.16) bottom view for hat for raspberry pi

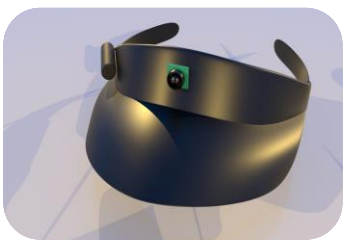


figure (5.17) hat with camera and battery

figure (5.18) side view for hat with battery

figure (5.19) camera at side of hat

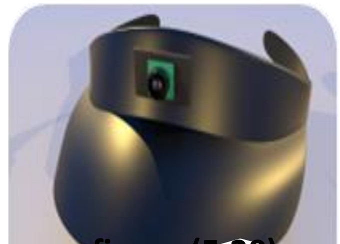


figure (5.20) front side view of hat

5.3 Problems of the project

- We can't find Referances and books easly.
- The code is correct but when we made the project work we found the code didn't work.
- Blind people still find difficulty to enter the bathroom.
- The code of the sensor doesn't make error but in fact this work is wrong for example : the sensor measure from 2 to 40 cm but practicaly the sensor start to work from 15 to 40 cm.

5.4 Future challenges

- The gloves can read books from pin digital.
- The Application can recognize to coins.
- Apply smart speakers can make voice interaction such as music players , audio books and news.
- Smart speakers work by applications on smart phones.

- In hat we can add glass with camera to take pictures it can recognize to face emotions ,text to speech for books and restaurant menu.

Code

```
#include "Talkie.h"

#include "Vocab_US_Large.h"

#include "Vocab_Special.h"

Talkie voice;


//............FLAME SEN PIN...................

const int flamePin = 2;

int Flame = HIGH;

//.............IFRA RED SEN PIN..................

int irPin = 7;  // (IR LED at pin 7)

int sensorOut = HIGH;


//...........ULTASONC SEN PIN....................

int trigger = 4;

int echo = 5;

int dist;

long time_taken;


//.............Buzz PIN..................

const int Buzz = 12;

//...............................

void setup()

{

  Serial.begin(9600);

  pinMode(flamePin, INPUT);

  pinMode(Buzz, OUTPUT);
```

```
  pinMode(irPin, INPUT);
  pinMode(trigger, OUTPUT);
  pinMode(echo, INPUT);
  digitalWrite(Buzz, LOW);
  Serial.println("test");
}
/*###Function to calculate distance###*/
void calculate_distance(int trigger, int echo)
{
digitalWrite(trigger, LOW);
delayMicroseconds(2);
digitalWrite(trigger, HIGH);
delayMicroseconds(10);
digitalWrite(trigger, LOW);
time_taken = pulseIn(echo, HIGH);
dist= time_taken*0.034/2;
}
void loop()
 {
  Flame = digitalRead(flamePin);
  int sensorvalue = analogRead(A0);
  sensorOut = digitalRead(irPin);
  calculate_distance(trigger,echo);

  //..............................................................
 //CODE IR SEN
  if (sensorOut == LOW)
  {
```

```
  digitalWrite(Buzz, LOW);
}
else
{
  digitalWrite(Buzz, HIGH);
  voice.say(sp2_DANGER);
}
delay(200);

//.................................................................
//CODE WATER SEN
if (sensorvalue >200) //this num is from 0 to 1023 for water level
{
  digitalWrite(Buzz, HIGH);
  voice.say(sp2_DANGER);
}
else
{
   digitalWrite(Buzz, LOW);
}
delay(200);

//....................................................................
//CODE FLAME SEN
if (Flame== LOW)
{
  digitalWrite(Buzz, HIGH);
  voice.say(sp2_DANGER);
```

```
  }
  else
  {
    digitalWrite(Buzz, LOW);
  }
  delay(200);

//..................................
//CODE Ultrasonic SEN
  if (dist < 100 && dist >0)
  {
  for (int i=dist; i>0; i--)
        digitalWrite(Buzz, HIGH);
        voice.say(sp2_DANGER);
    else
    {
        digitalWrite(Buzz, LOW);
    }
   delay(200);
}
import cv2
import os
cam = cv2.VideoCapture(0)
cam.set(3, 640) # set video width
cam.set(4, 480) # set video height
face_detector = cv2.CascadeClassifier('HAAR/haarcascade_frontalface_default.xml')
# For each person, enter one numeric face id
face_id = input('\n enter user id end press <return> ==>  ')
```

```
print("\n [INFO] Initializing face capture. Look the camera and wait ...")

# Initialize individual sampling face count

count = 0

while(True):

    ret, img = cam.read()

    #img = cv2.flip(img, -1) # flip video image vertically

    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    faces = face_detector.detectMultiScale(gray, 1.3, 5)

    for (x,y,w,h) in faces:

        cv2.rectangle(img, (x,y), (x+w,y+h), (255,0,0), 2)

        count += 1

        # Save the captured image into the datasets folder

        cv2.imwrite("dataset/User." + str(face_id) + '.' + str(count) + ".jpg", gray[y:y+h,x:x+w])

        cv2.imshow('image', img)

    k = cv2.waitKey(100) & 0xff # Press 'ESC' for exiting video

    if k == 27:

        break

    elif count >= 10: # Take 150 face sample and stop video

         break

# Do a bit of cleanup

print("\n [INFO] Exiting Program and cleanup stuff")

cam.release()

cv2.destroyAllWindows()

import cv2

import numpy as np

from PIL import Image

import os

# Path for face image database
```

```
path = 'dataset'
recognizer = cv2.face.LBPHFaceRecognizer_create()
detector = cv2.CascadeClassifier("HAAR/haarcascade_frontalface_default.xml");
# function to get the images and label data
def getImagesAndLabels(path):
    imagePaths = [os.path.join(path,f) for f in os.listdir(path)]
    faceSamples=[]
    ids = []
    for imagePath in imagePaths:
        PIL_img = Image.open(imagePath).convert('L') # convert it to grayscale
        img_numpy = np.array(PIL_img,'uint8')
        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)
        for (x,y,w,h) in faces:
            faceSamples.append(img_numpy[y:y+h,x:x+w])
            ids.append(id)
    return faceSamples,ids
print ("\n [INFO] Training faces. It will take a few seconds. Wait ...")
faces,ids = getImagesAndLabels(path)
recognizer.train(faces, np.array(ids))
# Save the model into trainer/trainer.yml
recognizer.write('trainer/trainer.yml') # recognizer.save() worked on Mac, but not on Pi
# Print the numer of faces trained and end program
print("\n [INFO] {0} faces trained. Exiting Program".format(len(np.unique(ids))))
import cv2
import numpy
import pyttsx3
engine = pyttsx3.init()  # initializing engine
```

```
rec = cv2.face.LBPHFaceRecognizer_create()
rec.read('trainer/trainer.yml')
cascadePath = "HAAR/haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascadePath)
font = cv2.FONT_HERSHEY_SIMPLEX
cap = cv2.VideoCapture(0)
cap.set(3, 640)  # set video widht
cap.set(4, 480)  # set video height

id = 0
names = ['None', 'Mahmoud', 'Tarek']
while(1):
    status, img = cap.read()
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(gray, 1.3, 5)
    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 2)
        # returns id and confidence level
        id, conf = rec.predict(gray[y:y + h, x:x + w])
        if(conf > 75):  # VERY IMP confidence level is checked
            name = "Unknown"
        else:
            name = names[id]
        engine.say(name)
        engine.runAndWait()

        cv2.putText(img, str(name), (x + 5, y - 5),
                    font, 1, (255, 255, 255), 2)
```

```
        cv2.putText(img, str(conf), (x + 5, y + h - 5),
                    font, 1, (255, 255, 0), 1)
    cv2.imshow('FaceDetect', img)
    k = cv2.waitKey(10) & 0xff  # Press 'ESC' for exiting video
    if k == 27:
        break
f.close()
cap.release()
cv2.destroyAllWindows()
```

Conclusion

We try to help the blind people as we can by that project we modified on the stick of blind to make it more modern and smart to make the blind person control on around him as he was saw we put in the stick some sensors (like Ultrasonic sensor, water level sensor, flame sensor ,,,,,etc) to make him more easy to live a daily life

We also help him to know the people around him through the hat which contain camera and raspberry-pi which operate by imaging process.

And we attach our system with mobile application to make him more smart

References

1. https://components101.com/sensors/ir-sensor-module
2. http://mysmartcane.ca/
3. https://www.google.com/search?client=ms-android-huawei&sxsrf=ACYBGNSy51r1Sz05x80AnmOYmWRFqKNfCw%3A1573419463498&ei=x3nIXYH8HeeugweA4KXwAQ&q=smart+cane+history&oq=smart+cane+history&gs_l=mobile-gws-wiz-serp.3...11627.13351..14151...0.0..0.243.1713.0j1j7......0....1.........0i19j0i22i30j0i22i10i30j33i22i29i30j33i21j33i160.eOzOOPy8PXA#sbfbu=1&pi=smart%20cane%20history
4. https://www.google.com/search?client=ms-android-huawei&sxsrf=ACYBGNTA9AwcxnZstNwvI9kCu6_2r8yE1Q%3A1573420208693&ei=sHzIXfHuKfudjLsPhpOawAg&q=%D8%A7%D8%BA%D9%86%D9%8A%D8%A9+%D8%AA%D8%AD%D9%8A%D8%B1%D9%83+%D8%AF%D9%86%D8%AF%D9%86%D9%87%D8%A7&oq=%D8%A7%D8%BA%D9%86%D9%8A%D8%A9+%D8%AA%D8%AD%D9%8A%D8%B1%D9%83+&gs_l=mobile-gws-wiz-serp.1.1.46i275j0l3j0i10j0l3.6695.45246..48255...2.0..4.690.9459.3-6j12j3......0....1.......8..35i362i39j35i362i39i19j35i39j0i131j46j46i67j0i67j46i131.jO1Y1OEoZx4
5. https://www.google.com/search?q=%D8%AA%D8%B5%D9%85%D9%8A%D9%85%D8%A7%D8%AA+smart+cane+for+blind&tbm=isch&ved=2ahUKEwio0pzmsOzlAhVKYBoKHcyXAu4Q2-cCegQIABAC&oq=%D8%AA%D8%B5%D9%85%D9%8A%D9%85%D8%A7%D8%AA+smart+cane+for+blind&gs_l=mobile-gws-wiz-img.3...15164.18671..19532...0.0..0.269.2150.0j8j3......0....1.........35i39j30i10j33i10.W_I-NNy4EZI&ei=-6_OXejoHcrAacyvivAO&bih=640&biw=360&client=ms-android-huawei&prmd=ivn
6. https://www.hotmcu.com/water-level-sensor-liquid-water-droplet-depth-detection-p-113.html
7. https://www.watelectronics.com/simple-water-level-alarm-circuit/
8. https://www.elprocus.com/flame-sensor-working-and-its-applications/
9. https://www.ornicom.com/products/flame-detectors.html?gclid=CjwKCAiA5o3vBRBUEiwA9PVzaqQC4ErVnlO7W68eBKz6Gg_Oje4cEQ_HjgyQKWiSEkykP8I7Qyg5rRoC4NAQAvD_BwE
10. https://www.google.com/search?q=jumper+wires&oq=jumper+wires&aqs=chrome..69i57j0l3.10558j0j7&client=ms-android-huawei&sourceid=chrome-mobile&ie=UTF-8
11. https://www.electronicshub.org/arduino-flame-sensor-interface/
12. https://www.semanticscholar.org/paper/Infrared-vs.-Ultrasonic-Finger-Detection-on-a-Piano-Pra-Fontana/10e2d18c9b93ce78174e8205ba09c13be6f5d500/figure/1
13. https://www.electronicshub.org/arduino-flame-sensor-interface/

14. https://components101.com/433-mhz-rf-receiver-module
15. https://components101.com/433-mhz-rf-transmitter-module
16. https://www.pololu.com/product/2188
17. https://store.arduino.cc/usa/arduino-micro
18. https://store.arduino.cc/usa/arduino-micro#
19. https://store.arduino.cc/usa/arduino-nano
20. https://store.fut-electronics.com/products/infrared-reflection-switch-sensor-proximity
21. https://components101.com/microcontrollers/arduino-nano
22. https://www.ifuturetech.org/product/e18-d50nk-infrared-proximity-sensor/
23. https://www.arduino.cc/en/products/compare
24. https://roboticx.ps/product/arduino-nano-usb-microcontroller-v3-china/
25. Dr. Ovidiu Vermesan SINTEF, Norway, Dr. Peter FriessEU, Belgium, "Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems", river publishers' series in communications, 2013.
26. Dr. Ovidiu Vermesan SINTEF, Norway, Dr. Peter FriessEU, Belgium, "Internet of Things–From Research and Innovation to Market Deployment", river publishers' series in communications, 2014.
27. O. Vermesan, P. Friess, P. Guillemin, S. Gusmeroli, et al., "Internet of Things Strategic Research Agenda", Chapter 2 in Internet of Things -Global Technological and Societal Trends, River Publishers, 2011.
28. Martin Serrano, Insight Centre for Data Analytics, Ireland ,Omar Elloumi, Alcatel Lucent, France, Paul
29. IoT: https://dzone.com/articles/the-internet-of-thingsgateways-and-next-generation.
30. .[http://www.reloade.com/blog/2013/12/6characteristicswithin internet-things-iot.php].
31. Smart walking stick - an electronic approach to assist visually disabled persons Mohammad Hazzaz Mahmud, Rana Saha, Sayemul Islam
32. Mohd Helmy Abd Wahab, Amirul A. Talib, Herdawatie A. Kadir, Ayob Johari, A. Noraziah, Roslina M. Sidek, Ariffin A. "Smart cane: assistive cane for visually impaired people", IJCSI, Vol.8 Issue 4, July 2011.
33. M. Bousbia-Salah, A. Larbi, and M. Bedda, "An approach for the measurement of distance travelled by blind and visually impaired people," in Proc. 10th IEEE International Conference on Electronics, Circuits and Systems, Sharjah, United Arab Emirates, pp. 1312-1315, 2003
34. Hashino, S.; Ghurchian, R.; A blind guidance system for street crossings based on ultrasonic sensors. Information and Automation (ICIA), 2010 IEEE International Conference on June 2010
35. David Castells, Joao M.F. Rodrigues, J.M. Hans du Buf "Obstacle detection and avoidance on sidewalks" In Proc. Int. Conf. on Computer Vision-Theory and Applications, Vol. 2, pp. 235-240, 201
36. Shruti Dambhare M.E 3rd SEM (ESC) G.H.R.C.E. Nagpur, Prof. A. Sakhare M.Tech (ESC) G.H.R.C.E. Nagpur Smart stick for Blind: Obstacle Detection, Artificial vision and Real-time assistance via GPS.