Internet of Things (IOT) based Smart Home

Thesis submitted to
Visvesvaraya National Institute of Technology, Nagpur
In partial fulfilment of requirement for the award of
Degree of

Bachelor of Technology
(Electronics and Communication Engineering)

By
Nihar Jain **(BT12ECE053)**
Avinash Paliwal **(BT12ECE056)**
Rishi Mody **(BT12ECE065)**

Guide
Dr. Kishore D. Kulat
Department of Electronics & Communication Engineering
Visvesvaraya National Institute of Technology

Declaration

I, hereby declare that the thesis titled "**Internet of Things (IOT) based Smart Home"** submitted herein has been carried out by us in the Department of Electronics and Communication Engineering of Visvesvaraya National Institute of Technology, Nagpur. The work is original and has not been submitted earlier as a whole or in part for the award of any degree/diploma at this or any other Institution/University.

Nihar Jain **(BT12ECE053)**
Avinash Paliwal **(BT12ECE056)**
Rishi Mody **(BT12ECE065)**

Date:-

Certificate

The thesis titled "**Internet of Things (IOT) based Smart Home**" submitted by

Nihar Jain	(BT12ECE053)
Avinash Paliwal	(BT12ECE056)
Rishi Mody	(BT12ECE065)

For the award of Bachelor of Technology, has been carried out under my supervision at the Department of Electronics and Communication Engineering of Visvesvaraya National Institute of Technology, Nagpur. The work is comprehensive, complete and fit for evaluation.

Dr. K. D. Kulat
Professor,
Department of Electronics and Communication Engineering
VNIT, Nagpur.

Dr. A.S. Gandhi
Head of Department,
Department of Electronics and Communication Engineering
VNIT, Nagpur.

Acknowledgement

We would like to express our deepest appreciation and sincere gratitude to all those who made it possible for us to complete this project. A special gratitude to our final year project guide, Dr. K. D. Kulat, who gave us his valuable suggestions from time to time and encouraged us throughout the course of this project. We would also like to thank the college authorities for providing us the lab equipment and other facilities which proved to be vital for the completion of our project.

Contents

Table of Figures

1. Introduction

1.1 What is Home Automation?

Home automation refers to the automatic and electronic control of household features, activity, and appliances [1]. Automation of appliances is in general seen not only in households, but also in office buildings, bank vaults and places of national and historic importance. In case of an automated home, the central automation system is responsible for controlling the operation of appliances, locks, doors, windows, lighting system, etc.

An automated home provides ease of functionality and reduces the efforts needed to manage all the devices in the household. It also provides increased quality of life for disabled and elderly people. A good home automation system is the one which runs on its own without any need for user intervention, while at the same time allowing the user to access and control the status of all devices and appliances running in the household through a centralized control unit. Another important feature of home automation systems is that they are energy efficient i.e. they are built for conservation and efficient utilization of energy.

Implementation of the home automation depends on the type of controls like wired or wireless [2]. There are mainly three types of home automation systems:

1. Power line Based Home Automation
2. Wired or BUS Cable Home Automation
3. Wireless Home Automation

The concept of the "Internet of Things" has tied in closely with the popularization of home automation systems.

1.2 What is IOT (Internet of things)?

Different sources have defined IoT differently. The Internet of Things is a network of physical objects - devices, vehicles, buildings and other items embedded with electronics, software, sensors and network connectivity that enable these objects to collect and exchange data [3].

The Internet of Things (IoT) refers to the ever-growing network of physical objects that feature an IP address for internet connectivity, and the communication that occurs between these objects and other Internet-enabled devices and systems [4].

The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction [5].

Our implementation of IoT based home automation is based on collecting all the sensor, appliance and device data in a central server and using this collective data to make decisions, thus minimizing human intervention.

To give an example, let us consider a situation where an automatic washing machine has stopped working for some reason. The central processing unit and server knows the current operating mode of washing machine. It also knows the current water level in the water tank. So after encountering the above problem the algorithm looks for dependent resources and concludes that the water supply to the washing machine has stopped due to low water level in water tank. The central unit then automatically switches on the water pump for the water tank.

1.3 Existing home automation systems

With the advent of technology, a huge amount of research has been conducted in home automation systems and as a result, we have some very high end systems which have been developed. The Honeywell home automation system [6] is one such example, which provides a complete set of functionality starting from thermostats and lighting to locks, shades and even home entertainment systems. Savant [7] is another system which provides end to end solutions for home automation using a central host for communicating with devices and remote control and an android app for their control.

These and many other home automation systems are available which run on various technologies. Some run entirely on communication using Wi-Fi modules [8] while some use other technologies such as Zigbee [9], [10] and Bluetooth [11], [12]. A detailed comparison and contrast within the different types of home automation systems with their features, advantages and disadvantages can be found in [13].

These systems are generally controlled using an interface on phone, PC or tablet but can also be controlled using voice recognition through GSM [14]. There are also some mixed type home automation systems which are controlled using a combination of GSM, Bluetooth and Zigbee technologies and the user interface is provided through an android application. The

application is used to issue commands to the home automation system and control the circuitry. The quality of the user interface determines the usability of the system.

Nowadays, security of the household is also a very important issue and hence security and data encryption constitute vital parts of a modern day home automation system [15].

1.4 Drawbacks of the current systems

One of the major drawbacks of the current systems is their high cost. Almost all of the professionally developed and commercially available home automation systems use very high end technologies, which lead to increased cost of production. Also, the cost of implementing these technologies in the system goes on increasing with the increase in area which has to be covered under the home automation system.

Another problem with the current systems if that this advanced technology is not needed for many of the indoor applications. Many of the communications pertaining to controlling devices inside the house can work with technologies which consume low power and have very low data rate.

1.5 System Block Diagram

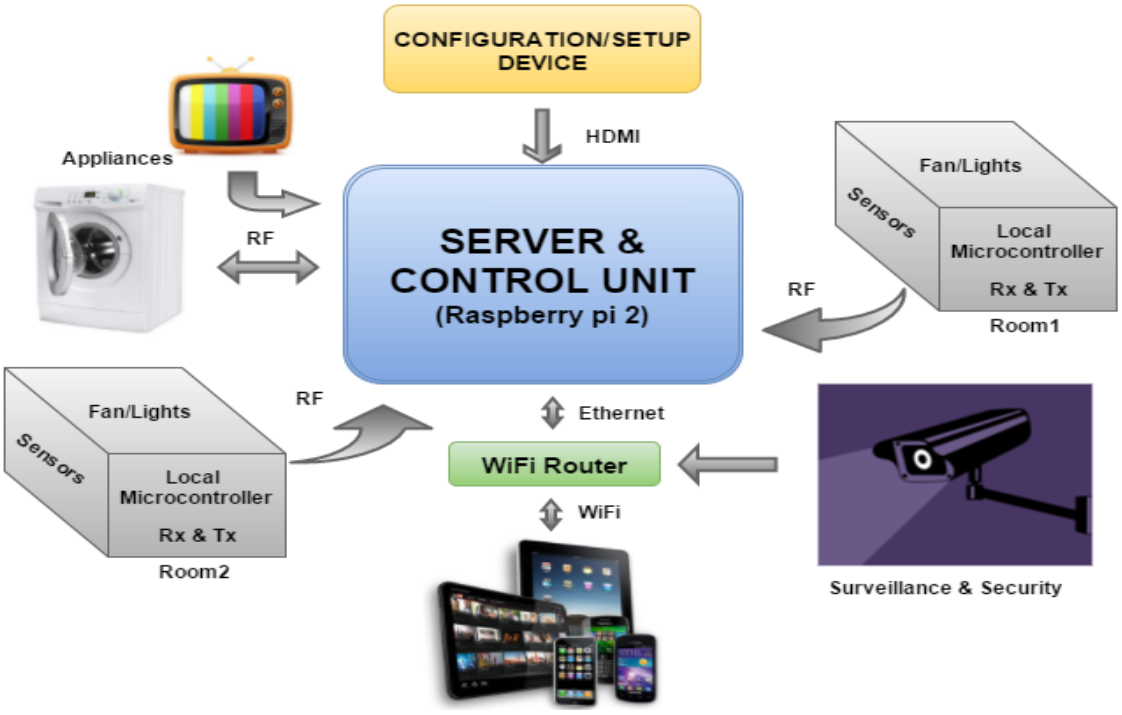


Figure 1.1 System Block Diagram

2. Hardware

In our hardware implementation there is a central processing unit and server (Raspberry Pi 2) as well as a localized control unit (Atmega16) in each room. All the sensors and appliances are monitored and controlled by this localized unit which exchanges this data with the central unit. Different technologies of communication like Ethernet, Wi-Fi, nRF, IR, proprietary RF and Bluetooth modules are used depending on the required application, power consumption, required data rate and signal range. Sensors for application like human presence detection, smoke and LPG leakage detection and ambient data readings are used. Isolated appliances like washing machine are controlled using Wi-Fi. The abovementioned is explained in detail in the following subsections.

2.1. Communication Technologies

2.1.1. Wi-Fi (IEEE 802.11)

Wi-Fi is a networking technology which uses radio waves at 2.4 GHz or 5 GHz to communicate wirelessly. It follows the IEEE 802.11 standard. It is the standard wireless local area network (WLAN) technology for connecting computers and other wireless internet enabled devices [16].

Figure 2.1 Wi-Fi Logo

Wi-Fi Module: ESP8266 ESP-01 Serial Wi-Fi Wireless Transceiver

This module can be used to connect a microcontroller to a Wi-Fi network. It can act as node as well as a wireless access point. It uses serial simplistic UART communication for exchanging messages between microcontroller and module and can be configured with AT commands.

Figure 2.2 Wi-Fi Module

Code (AT Commands):

- Initialization
  - AT+CWMODE=1 //Set mode as station
  - AT+CIOBAUD=9600 //Set baud rate as 9600
  - AT+CWJAP=”SSID”,”Password” //Connect to WiFi network
- Server Setup
  - AT+CIPMUX=1 //Multiple connection mode
  - AT+CIPSERVER=1,8525 //Start server at port 8525
  - AT+CIFSR //Check IP address
- Send/Receive Data
  - AT+CIPSTART=0,”TCP”,”192.168.0.100”,8525 //Establish a TCP connection with host(192.168.0.100) at port 8525
  - AT+CIPSEND=0,LEN //Send data of length LEN
  - >ABCDEFGH //Data of length LEN to be sent
  - +IPD,0,len,data //Received Data of length len
  - AT+CIPCLOSE=0 //Close connection with ID: 0

2.1.2. Bluetooth

Bluetooth is a standardized communication secure protocol for wireless 2.4 GHz link. It has low power consumption and low bitrate. Two Bluetooth devices connect over an ad hoc network known as piconet. It is a network of Bluetooth enabled devices in close vicinity. In a piconet, one device acts as master and others as slaves [17].

Figure 2.3 Bluetooth Logo

Module: HC-05 Bluetooth Module

HC-05 Bluetooth module which uses Bluetooth Serial Port Protocol and is designed for transparent wireless serial connection setup. It can operate in master as well as slave mode. It can supplant wired UART devices on microcontrollers. For a fixed network only initial setup through AT commands is required.

Figure 2.4 HC-05 Module

Code (AT Commands):

- Initialization – Master
  - AT+ROLE=1 //Master mode
  - AT+UART=9600,1,2 //Set baud rate, stop bit and parity
  - AT+CMODE=0 //Connect to a fixed address
  - AT+BIND=SLAVE ADDRESS //Bind to a single slave hc-05
- Initialization – Slave
  - AT+ROLE=0 //Slave mode
  - AT+UART=9600,1,2 //Set baud rate, stop bit and parity
  - AT+CMODE=0 //Connect to a fixed address
  - AT+BIND=MASTER ADDRESS //Bind to a single master hc-05
  - AT+INIT //Initialize SPP profile lib
  - AT+PAIR=MASTER ADDRESS, timeout //Pair with master hc-05

2.1.3. nRF(Nordic RF Module)

nRF24L01+ is a transceiver module manufactured by Nordic Semiconductors. Some key features are:

- It uses 2.4GHz frequency from ISM band.
- Multiple data rates of 250kbps, 1Mbps and 2Mbps is available.
- Power consumption is very low at 1mW and standby mode at 26µA.
- Automatic acknowledgment sent on reception.
- SPI transmission protocol used[18]

.

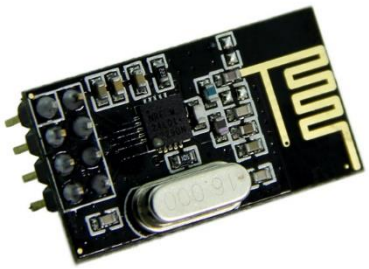


Figure 2.5 nRF24L01+ Module

2.1.4. RF (Radio Frequency) Transmission

RF transmission, as the name suggests, uses radio frequency (RF) to send and receive signals. These signals are transmitted at a particular frequency and a baud rate. A receiver can receive these signals only if it is configured for that frequency.

Why RF Transmission?

Signals through RF can travel through larger distances making it suitable for long range applications. Also, RF signals can travel even when there is an obstruction between transmitter & receiver. RF transmission is more strong and reliable than IR transmission. RF communication uses a specific frequency unlike IR signals which are affected by other IR emitting sources.

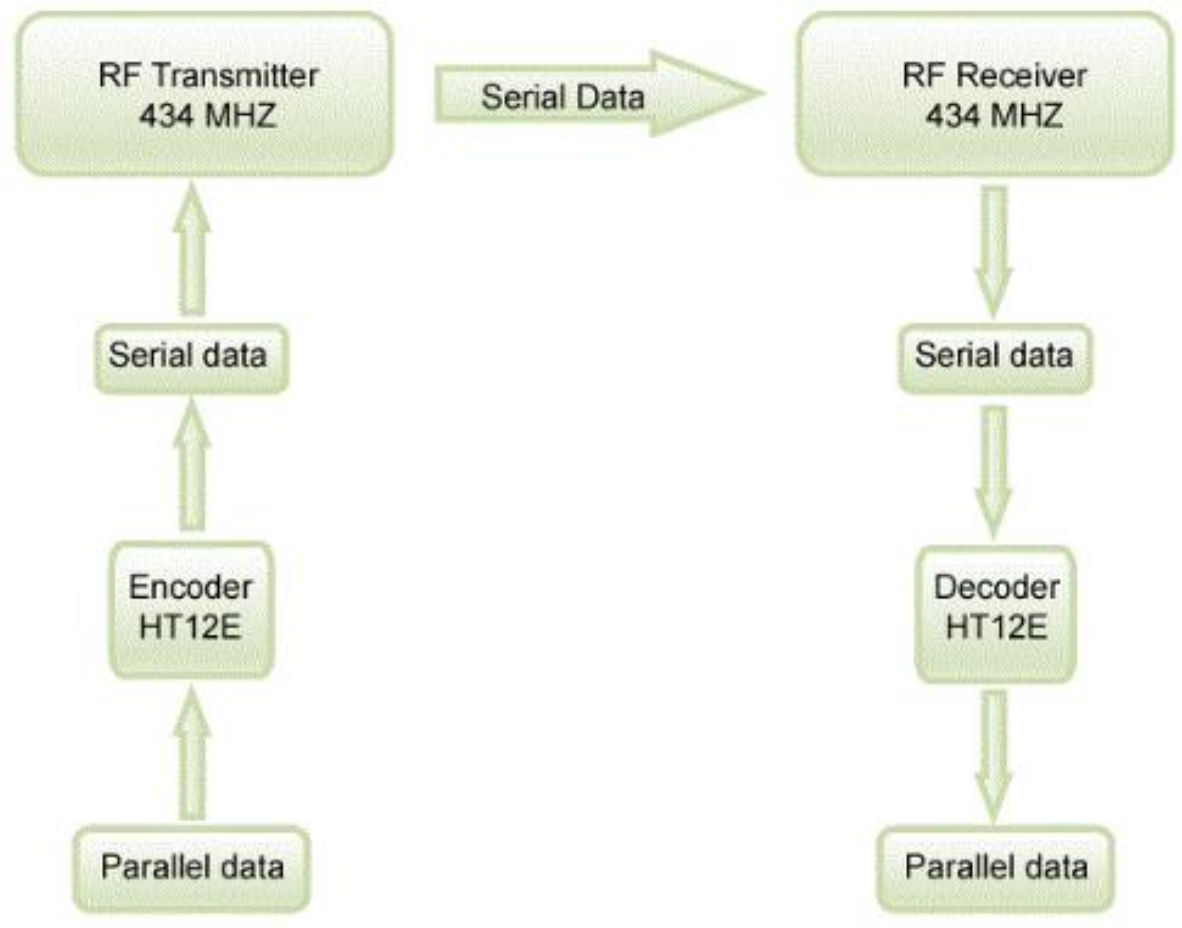


Figure 2.6 RF Block Diagram

2.1.4.1. Components used:

2.1.4.1.1 RF Module:

The RF module comprises of an RF Transmitter and an RF Receiver. The transmitter/receiver (Tx/Rx) pair operates at a frequency of 434 MHz. An RF transmitter receives serial data and transmits it wirelessly through RF through its antenna connected at pin4. The transmission occurs at the rate of 1Kbps - 10Kbps.The transmitted data is received by an RF receiver operating at the same frequency as that of the transmitter.

The RF module is often used along with a pair of encoder/decoder. The encoder is used for encoding parallel data for transmission feed while reception is decoded by a decoder.HT12E-HT12D is the encoder/decoder pair IC used in here.

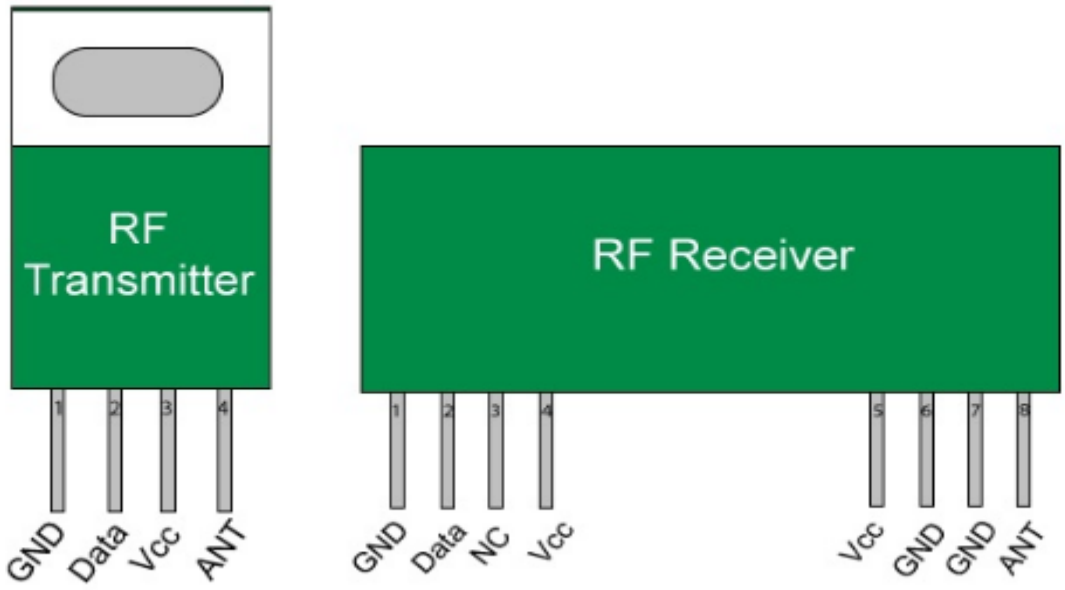


Figure 2.7 Proprietary RF Transmitter and Receiver Modules

2.1.4.1.2. HT12E:

HT12E is an encoder integrated circuit of 2^{12} series of encoders. They are paired with 2^{12} series of decoders for use in remote control system applications. It is mainly used in interfacing RF and infrared circuits.

Simply put, HT12E converts the parallel inputs into serial output. It encodes the 12 bit parallel data into serial for transmission through an RF transmitter. These 12 bits are divided into 8 address bits and 4 data bits.

HT12E has a transmission enable pin which is active low. When a trigger signal is received on TE pin, the programmed addresses/data are transmitted together with the header bits via an RF or an infrared transmission medium. HT12E begins a 4-word transmission cycle upon receipt of a transmission enable. This cycle is repeated as long as TE is kept low. As soon as TE returns to high, the encoder output completes its final cycle and then stops.

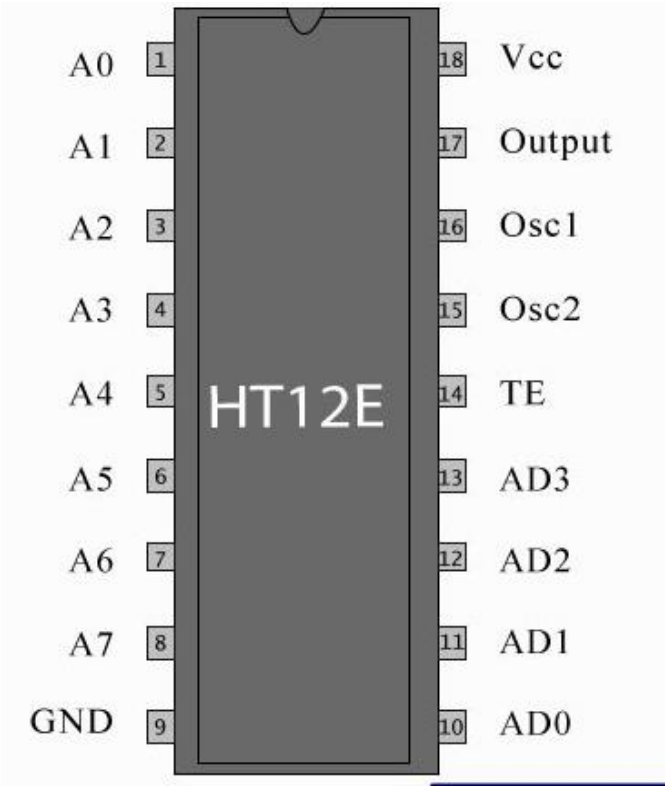


Figure 2.8 Encoder IC

Pin Description:

Pin No	Function	Name
1	8 bit Address pins for input	A0
2		A1
3		A2
4		A3
5		A4
6		A5
7		A6
8		A7
9	Ground (0V)	Ground
10	4 bit Data/Address pins for input	AD0
11		AD1
12		AD2
13		AD3
14	Transmission enable; active low	TE
15	Oscillator input	Osc2
16	Oscillator output	Osc1
17	Serial data output	Output
18	Supply voltage; 5V (2.4V-12V)	Vcc

Figure 2.9 Pin Description

2.1.4.1.3. HT12D:

HT12D is a decoder integrated circuit that belongs to 2^{12} series of decoders. It is mainly provided to interface RF and infrared circuits. They are paired with 2^{12} series of encoders.

In simple terms, HT12D converts the serial input into parallel outputs. It decodes the serial addresses and data received by, say, an RF receiver, into parallel data and sends them to output data pins. The serial input data is compared with the local addresses three times continuously. The input data code is decoded when no error or unmatched codes are found. A valid transmission in indicated by a high signal at VT pin.

HT12D is capable of decoding 12 bits, of which 8 are address bits and 4 are data bits. The data on 4 bit latch type output pins remain unchanged until new is received.

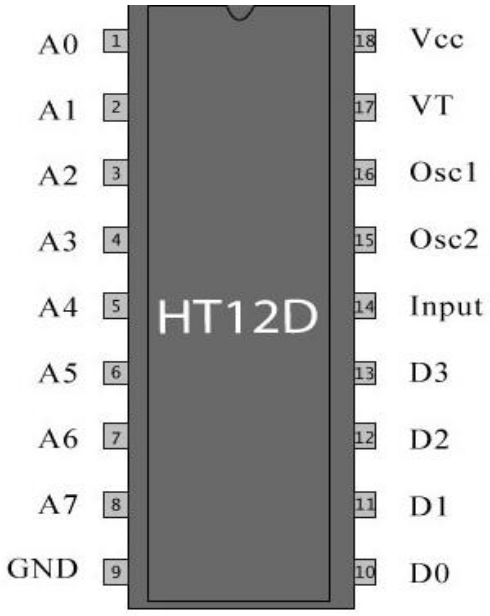


Figure2.10 Decoder IC

Pin Description:

Pin No	Function	Name
1	8 bit Address pins for input	A0
2		A1
3		A2
4		A3
5		A4
6		A5
7		A6
8		A7
9	Ground (0V)	Ground
10	4 bit Data/Address pins for output	D0
11		D1
12		D2
13		D3
14	Serial data input	Input
15	Oscillator output	Osc2
16	Oscillator input	Osc1
17	Valid transmission; active high	VT
18	Supply voltage; 5V (2.4V-12V)	Vcc

Figure 2.11 Pin Description

2.1.5. INFRARED COMMUNICATION

Infrared is an electromagnetic radiation with wavelengths ranging from the nominal red edge of the visible spectrum at 700nm to 1mm, i.e., frequency 430THz to 300GHz.

IR detectors are little microchips with a photocell that are tuned to listen to infrared light. They are almost always used for remote control detection - every TV and DVD player has one of these in the front to listen for the IR signal from the clicker. Inside the remote control is a

matching IR LED, which emits IR pulses to tell the TV to turn on, off or change channels. IR light is not visible to the human eye, which means it takes a little more work to test a setup.

- IR detectors are specially filtered for Infrared light; they are not good at detecting visible light.
- IR detectors have a **demodulator** inside that looks for modulated IR at 38 KHz. Just shining an IR LED won't be detected; it has to be PWM blinking at 38 KHz.
- IR detectors are digital out - either they detect 38KHz IR signal and output low (0V) or they do not detect any and output high (5V).

  When IR is detected using an oscilloscope, the pulse is seen as following,

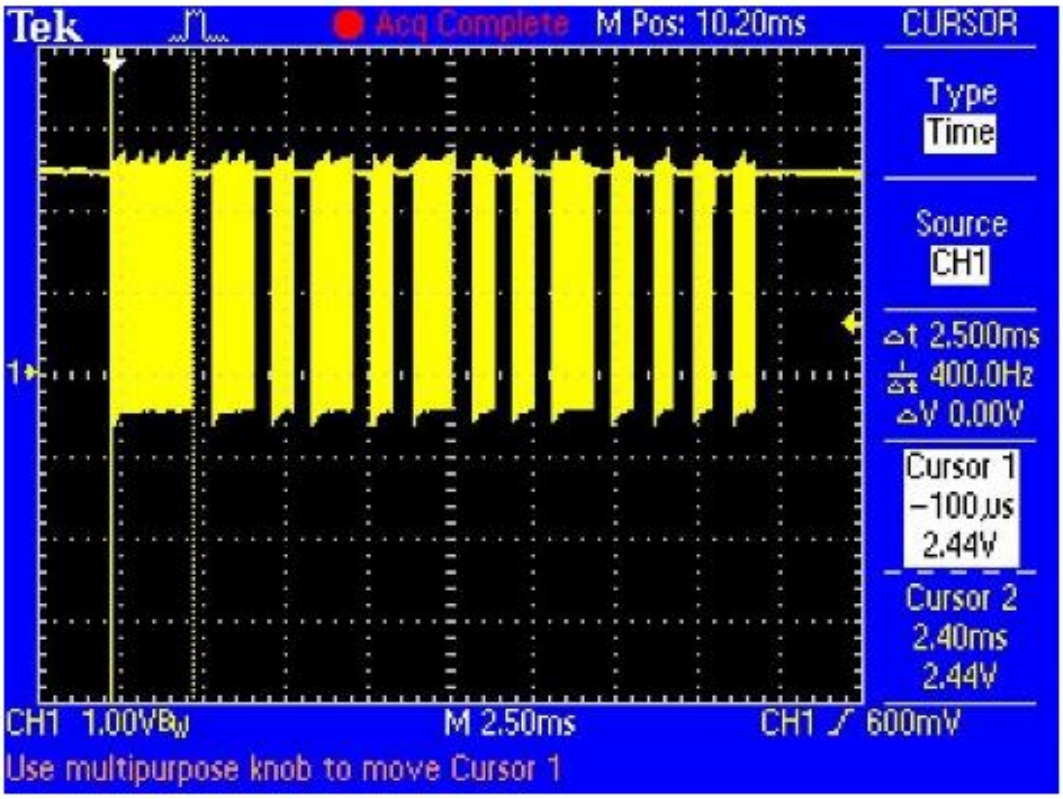


Figure 2.12 IR signal on DSO

We see pulses and IR signal. The yellow blocks show that the signal is transmitting and the plane line shows that the signal is not transmitting.

When we zoom in further, we see,

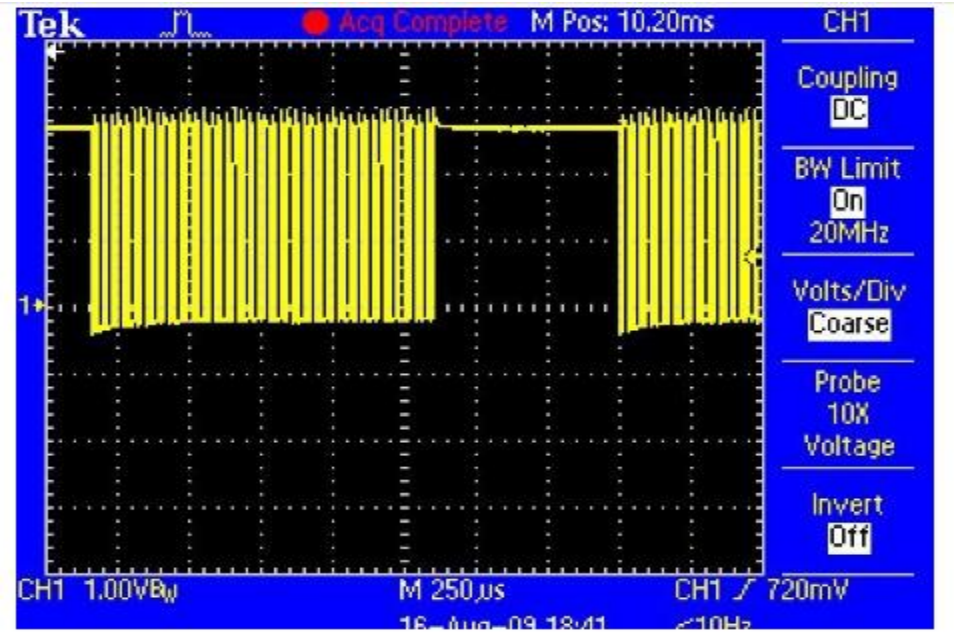


Figure 2.13 IR Signal on DSO (zoomed in)

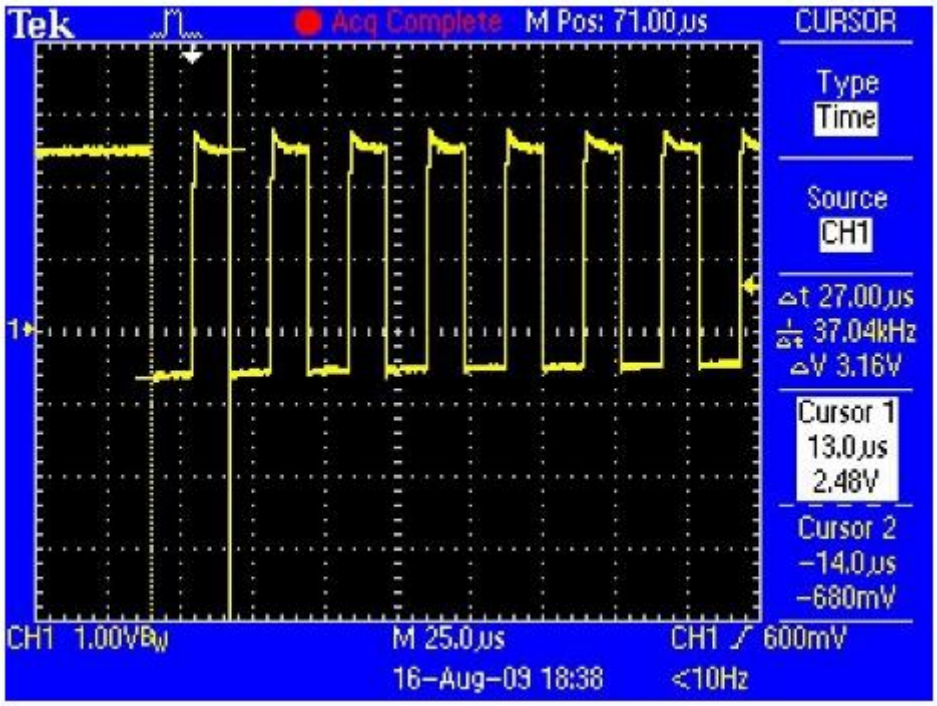


Figure 2.14 IR signal on DSO (zoomed in)

They are not really blocks but very fast pulses. The frequency of these pulses can be measured and for this case, it was about 37.04 KHz.

These IR pulses are sent using an IR transmitter Led which transmits at 38 KHz and is pulses using PWM-pulse width modulation.

Components Used:

A. TSOP1730

It is a miniature receiver used in infrared remote control systems. The output of this receiver can directly be decoded by the microcontroller and it supports all major transmission codes.

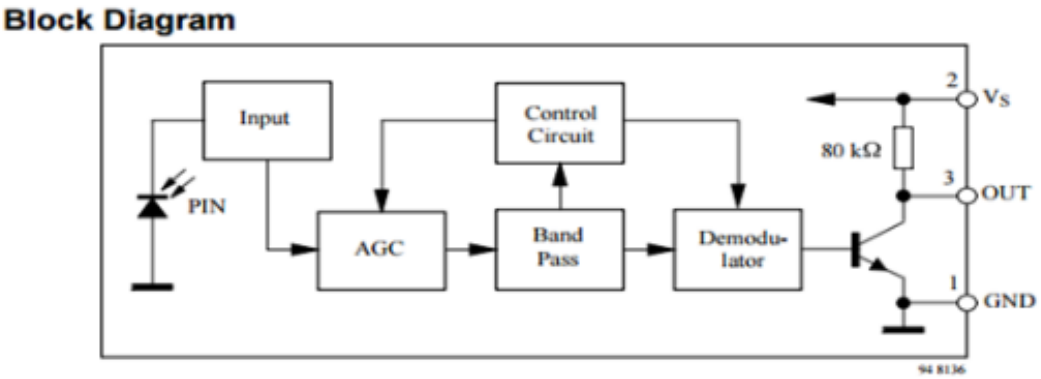


Figure 2.15 Block Diagram

Here we will connect the detector as such:

- Pin 1 is the output so we wire this to a visible LED and resistor
- Pin 2 is ground
- Pin 3 is VCC, connect to 5V

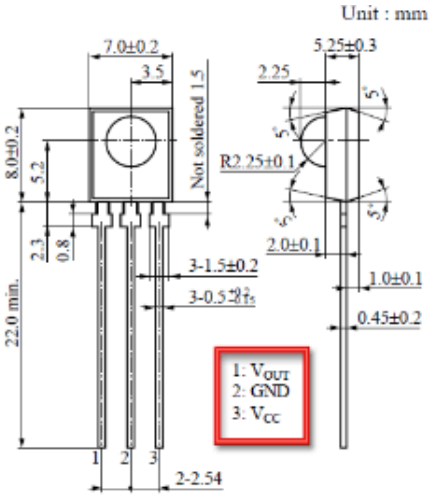


Figure 2.16 IR dimensions

2. IR Transmitter (5mm Infrared LED)

An IR LED, also known as IR transmitter, is a special purpose LED that transmits infrared rays in the range of 760 nm wavelength. Such LEDs are usually made of gallium arsenide or aluminium gallium arsenide.

2.2. Localized Sensor and Control Unit

This is the local microcontroller for monitoring and controlling the appliances and sensors inside the room. The local microcontroller communicates and exchanges local data with the central server using nRF module introduced in SECTION. Wireless sensors are isolated sensors used for sensing smoke, LPG gas leakage and human presence. Local sensors are mounted on the microcontroller unit and are used to measure temperature, illumination. The switches and simple appliances are controlled using proprietary 434 MHz RF modules. More detailed explanation is given in the subsequent subsections.

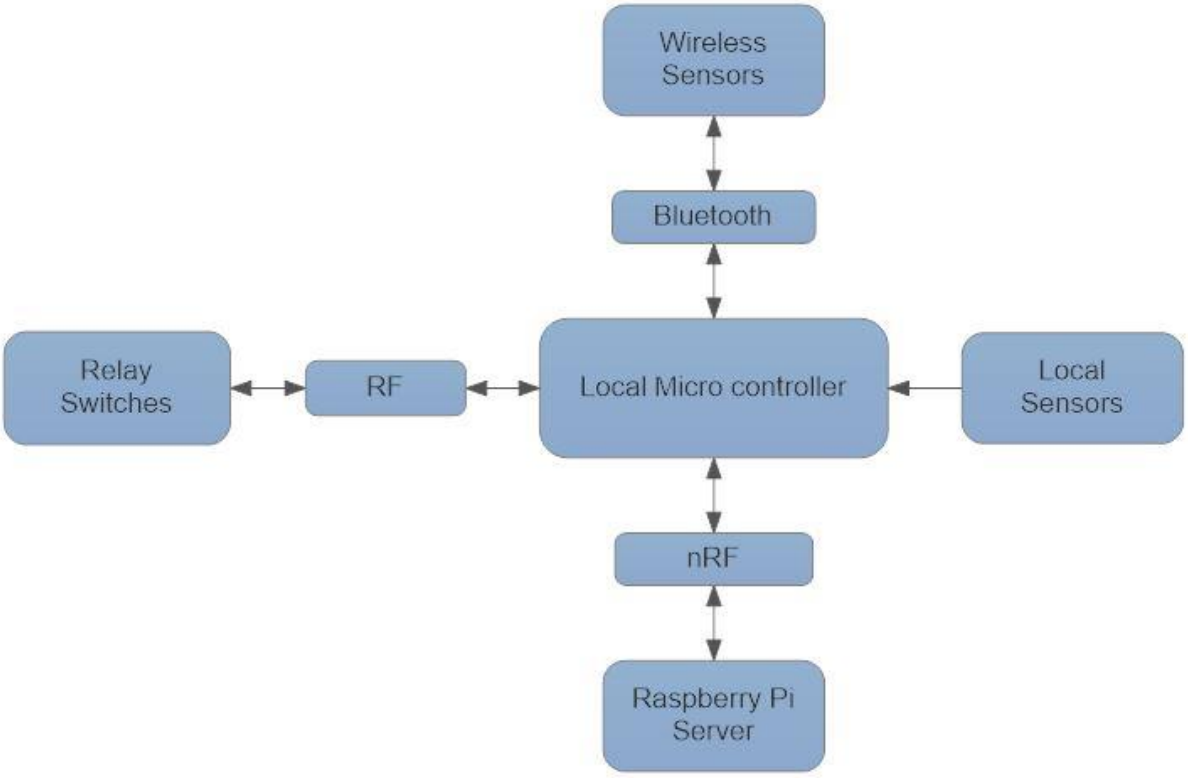


Figure 2.17 Local Micro controller block diagram

2.2.1. Presence Detection

Detecting moving as well as motionless human beings inside a room.

2.2.1.1. Existing methods

Methods	Drawbacks
Camera	Is intrusive and not preferable
Sound	Background noise may cause false triggers.
Passive Infra-Red Sensors	Detects only moving objects
Proximity Sensors at Doors	Unreliable if multiple people pass at the same time.
Infrared Sensors	Unreliable
Pressure Detectors	Costly and not feasible to cover whole surface area.
Ultrasonic Sensors	Narrow beam width.

2.2.1.2. Sensors used

- **HC-SR04 Ultra sonic sensor module**

HC-SR04 is a contact-free range detecting sensor offering range from 2cm to 400cm. It uses SONAR technology i.e. sending pulses and detecting the lag between the received pulses it estimates the distance between the module and the object. It has a narrow beam width with high accuracy.

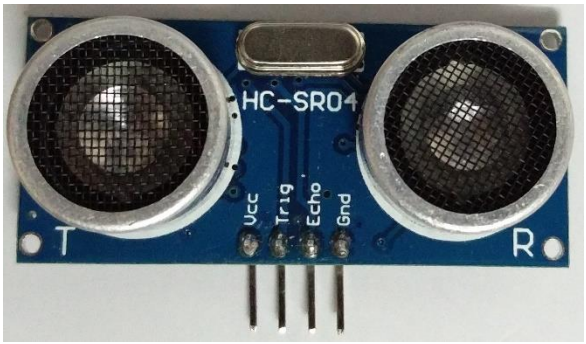


Figure 2.18 Ultrasonic Sensor

- **HC-SR501 Passive Infra-red sensor module**

PIR sensor works by detecting the radiation emitted by human body in the form of heat. Every object above absolute zero emits heat energy in the form of IR radiation. The sensor consists of a bee hive structure cover which is used to maximize the IR radiation received and focus it on a narrow area. The translucent cover also helps in reducing the unwanted radiation from visible spectrum and pass IR. The core part of module consists of the quad sensing material which senses the IR radiation and develops a charge proportional to the IR radiation falling on it. It has sensing range of 3-7 meters and conical angle coverage of about 100°.

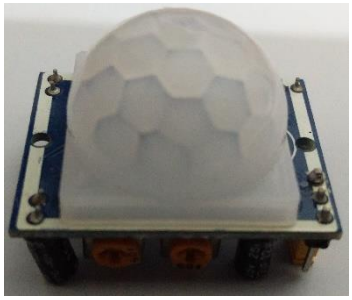


Figure 2.19 PIR Sensor

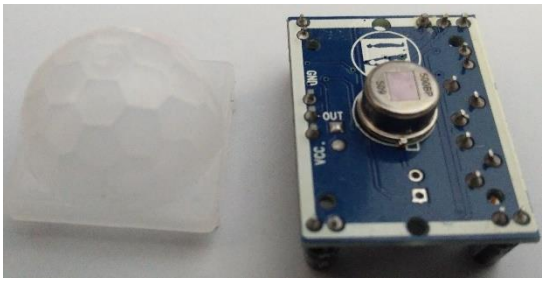


Figure 2.20 PIR sensor - sensing material

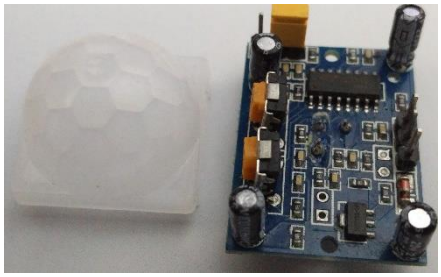


Figure 2.21 PIR sensor - PCB

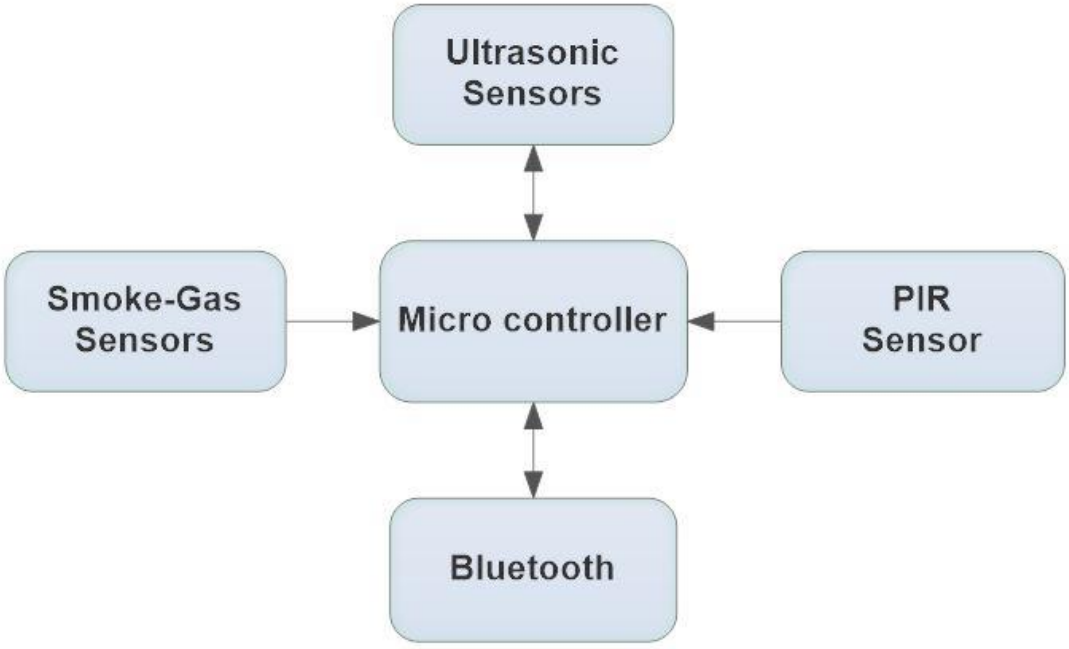


Figure 2.22 Wireless sensor unit block diagram

2.2.1.3. Method used

We are combining two of the abovementioned techniques to increase the reliability of presence detection system. We use two ultrasonic sensors at the entrance of the room. The arrangement is as shown below. Following are the possibilities of persons entering or exiting the room.

- Single person enters/exits the room
  - A person entering a room will trigger US1 first then US2. If he also triggers the PIR sensor we know that someone has entered the room.
  - If US2 and US1 get triggered respectively and PIR remain un-triggered for a predefined time then we can assume that someone has left the room.
- Multiple persons enter/exit the room
  - The sensor to get triggered last will determine whether a group entered or exited.
  - The other conditions are as same as the previous case of single person entering or exiting.

2.2.1.4. Applications

- Energy Saving

  By switching off the appliances like lights and fans when no presence is detected energy can be conserved. Predefined set of appliances can be configured to turn on when a presence is detected. For e.g. TV, fan, light
- Security

  When the house is empty the presence detector system can act as intruder detection system.

2.2.2. Local Sensors

2.2.2.1. DHT11 Temperature and Humidity Sensor module

DHT11 is a digital sensor which measures the humidity and temperature and outputs the data in digital format. It uses a capacitive humidity sensor and thermistor to measure the humidity and a NTC temperature measurement component. The data is transmitted using serial two way communication.

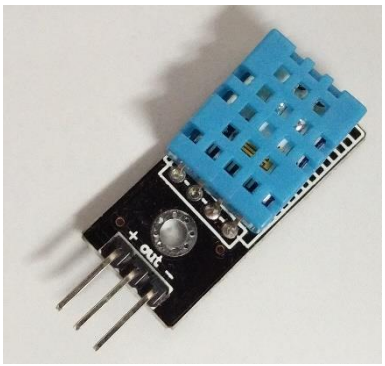


Figure 2.23 DHT11 Sensor

Data received: 8-bit integral Humidity data + 8-bit decimal Humidity data + 8-bit integral Temperature data + 8-bit decimal Temperature data + 8bit check sum. 8-bit checksum are the lowest 8-bits of the sum of all data and is used for error detection.

New data is available every 2 seconds.

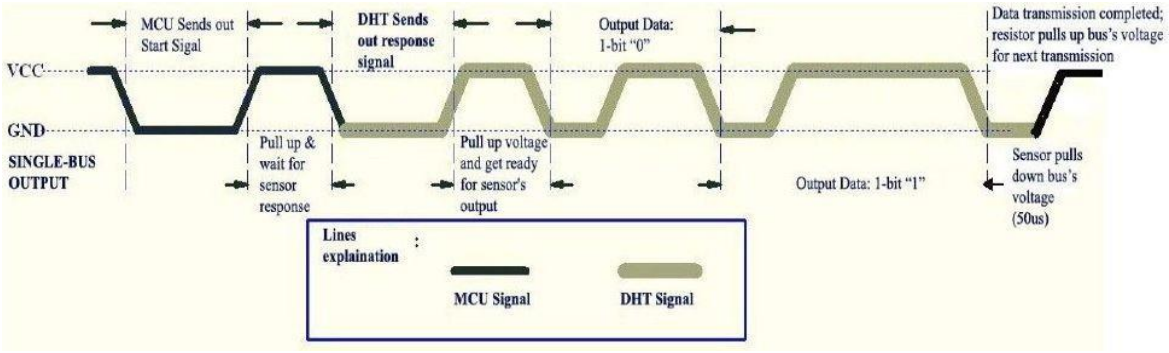


Figure 2.24 DHT11 data acquisition signal waveform

This data is used to monitor the surrounding temperature-humidity and assist in controlling appliances like Air conditioners and fans leading to energy saving.

2.2.2.2. MQ2-MQ6 gas sensor

MQ-2 sensor detects flammable gases like LPG gas, propane, hydrogen, i-butane and smoke. The sensor works using a sensing material with low conductivity in clean air and increased conductivity in the presence of combustible gas vapors.

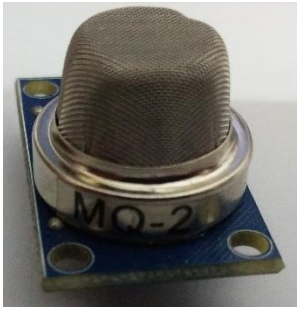


Figure 2.25 MQ-2 Sensor

MQ-6 sensor detects butane, LNG, LPG, propane with high sensitivity to these gases and low sensitivity to alcohol, cooking fumes and smoke due to cigarette.

Combining them both we can determine whether there is a gas leakage or fire break out and take appropriate action depending upon the situation.

Figure 2.26 MQ-6 Sensor

2.2.2.3. Light dependent resistor

A LDR is a photoconductive device whose material conductivity increases with increase in illumination. In the semiconductor material upon absorption of light the electrons get excited and jump to conduction band creating a large number of charge carriers.

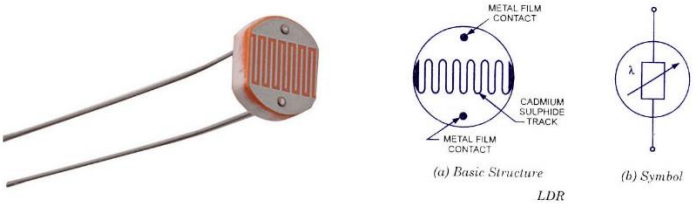


Figure 2.27 LDR Sensor and block diagram

The LDR is used to detect the intensity of light in a room and assist in controlling the lights in the room for energy saving.

2.2.3. Communication with central server

Data is transmitted and received from local microcontroller unit to the central server using nRf24L01+ modules.

Data is transmitted when one of the following event occurs:

- If there is a change in temperature or illumination reading.
- If a human presence is detected entering, leaving the room.
- If a state change is detected in the relay switches or appliances.
- If smoke sensor or LPG gas sensor goes off.

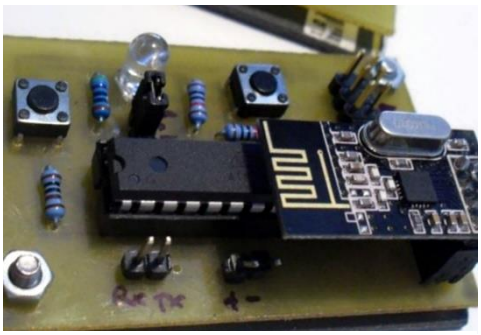


Figure 2.28 nRF module and atmega8 µC

On data reception the following execution takes place:

- The IR remote or RF switch change command received is executed.
- Command for a state change in sensor or micro controller state is executed.

2.2.4. Fan Speed Regulator

Automatic controls play an ever-increasing role in a human way of life. Automatic control is vast technological area whose central aim is to develop control strategies that improve performance when they applied to a system. The distinct characteristic of automatic control is that it reduces the human operator. One such gadget is the fan.

The circuit we made to operate the fan and control the ac voltage using Triac is shown in the following circuit diagram:

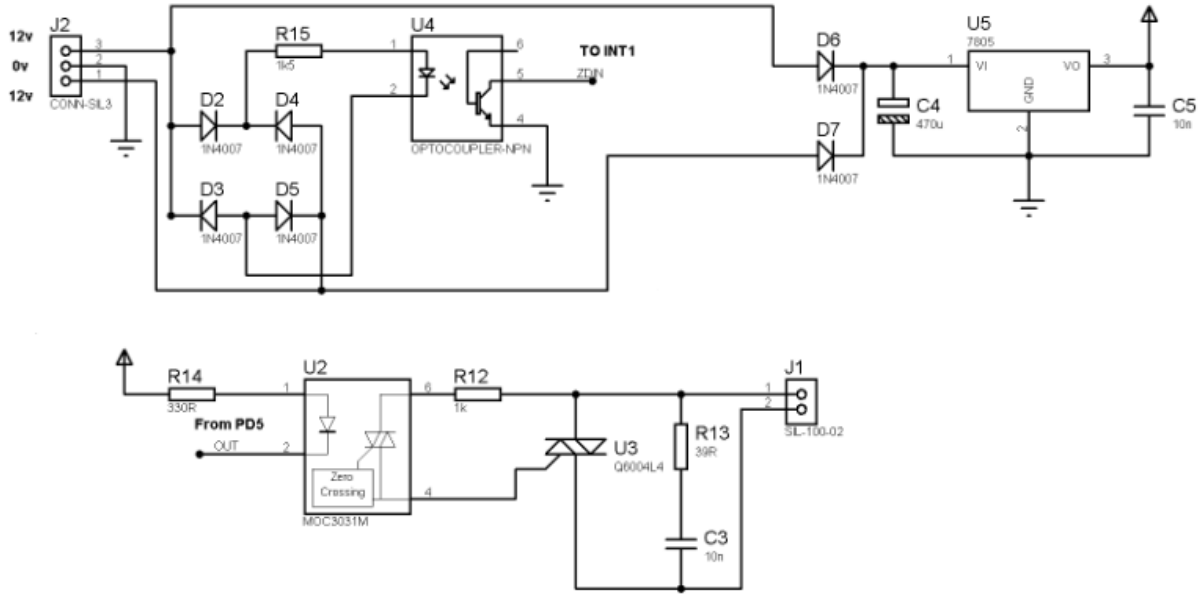


Figure 2.29 Fan Regulator Circuit Diagram

Hardware Tools:

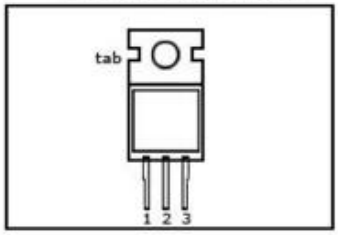
MCU - ATmega8 AVR 8 bit Microcontroller
MOC3021 for driving TRIACs.
BT136 Triac for controlling Fan.
Zero cross detector
RF module
12-0-12v transformer for power supply

BT136 SPECIFICATION

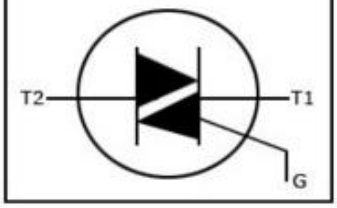
PINNING - TO 220 AB

PIN	DESCRIPTION
1	Main terminal 1
2	Main terminal 2
3	Gate
Tab	Main terminal 2

PIN CONFIGURATION



SYMBOL



LIMITING VALUES

Limiting valuest in accordance with the absolute maximum System(IEC 134)

SYMBOL	PARAMETER	CONDITIONS		MIN.	MAX.		UNIT
V_{DRM}	Repetitive peak of state voltages			-	-600 600[1]	-800 800	V
$I_{T(RMS)}$	RMS on - state current	full sine wave; $T_{mo} \leq 107$ ° C		-	4		A
I_{TSM}	Non - Repetitive peak on - state current	full sine wave; T_J = 25 ° c prior to surge					
		T = 20 ms		-	25		A
		T = 16.7 ms		-	27		A
I^2t	I^2t For fusing Repetitive	T = 10 ms		-	3.1		A^2s
dI_/dt	rate of rise of on - state current after triggering	I_{TM} = 6 A; IG = 0.2A; dI_G/dt = 0.2A/µs	T2+G+	-	50		A/µs
			T2+G-	-	50		A/µs
			T2-G-	-	50		A/µs
			T2-G+	-	10		A/µs
I_{GM}	peak gate current			-	2		A
V_{GM}	Peak gate voltage			-	5		V
P_{GM}	Peak gate power			-	5		W
$P_{G(AV)}$	Average gate power	Over any 20 ms period		-	0.5		W
T_{STG}	Storage temparature			-40	150		°C
T_J	Operating junction temparature			-	125		°C

Figure 2.30 Triac Specification

2.2.4.1. Working

The microcontroller Atmega8 is the heart of the system. Here we can control the speed of the fan from anywhere, anytime using the android application. The circuit is based on the principle of power control using a Triac. The circuit works by varying the firing angle of the Triac. In simple words, how much time should Triac conduct, changes. This directly varies the load power, since load is driven by Triac. The firing pulses are given to the gate of Triac.

Whenever we operate this application, it sends commands to our controlling system through RF module. At controlling system side, we have RF module, micro controller and load controlling circuits, zero cross detector circuit to help the speed controlling method. Whenever this RF module receives commands from android application, then it transfers this command to the micro controller. Micro controller will control the respective loads depends upon the command it received.

2.2.4.2. Advantages:

- No need of separate remote or switchboards for controlling electrical appliances
- Protection from shock while operating and secured
- Easy to use, children, old aged people, physically challenged people can also operate
- Power saver.

2.2.4.3. Applications:

- Fan speed controlling
- Home automation
- Machine's controlling system in industry
- Motor speed controlling systems, Light intensity controlling systems

2.2.5. Switch board

Atmega 16 is used to receive the encoded signal from the raspberry pi via Wi-Fi using UART of the microcontroller. The signal is decoded in the microcontroller and after processing the relevant address is given to the RF module. We have not used many RF modules so we have left three pins of the address and grounded the rest all pins making the address as xxx00000.

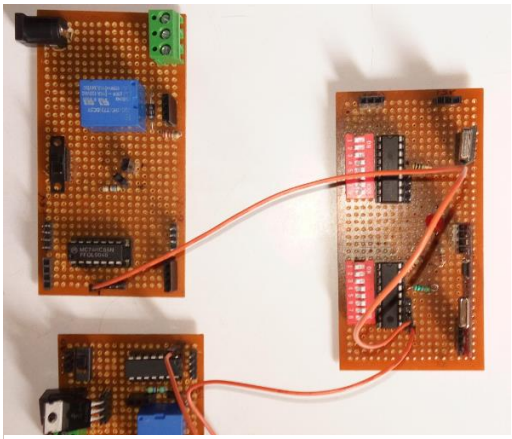


Figure 2.31 Receiving data from RF module to relay circuit

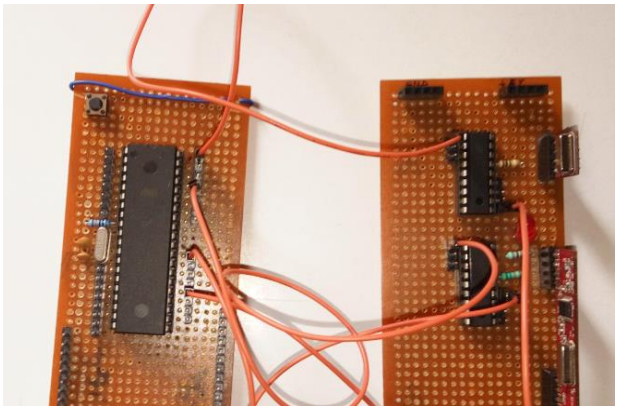


Figure 2.32 Data transmitted from atmega16 using RF module

The connection between the RF modules and the microcontroller is done using jumper wires from RF address lines to the PortA of Atmega16.

The data lines are connected to the PortC of the Atmega16.

PortA is taken as the output port as it is giving addresses to the receiver and the transmitter module. PortC is the input/output port as it is taking input from the receiver module and sending data to the transmitter module.

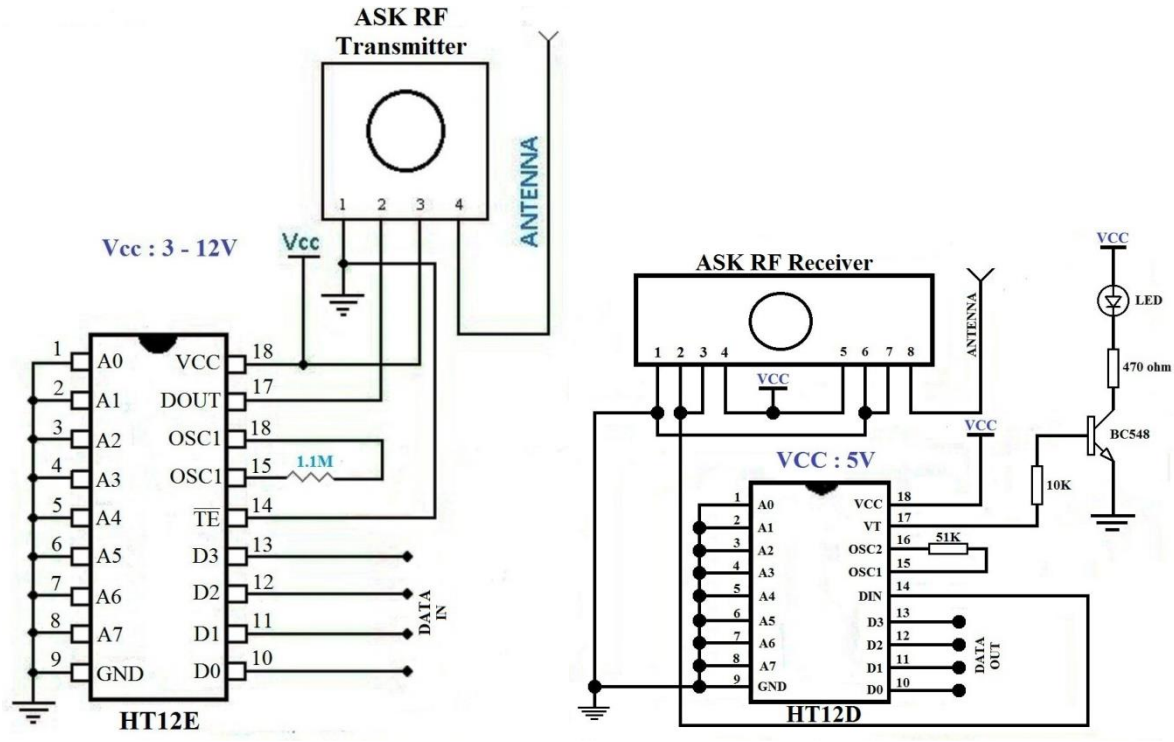


Figure 2.33 RF Transmitter and Receiver circuit

The Other part of the RF module that is the switch board side is connected to the relay circuit which is toggling the switch whenever a high voltage is received.

Two inputs are taken, one from the microcontroller (received through RF) and the other one is the manual switch .Both the inputs are given to the XOR gate which helps us enable toggling between the switches.

- **Relay Switch Circuit**

Relays are electromechanical devices that use an electromagnet to operate a pair of movable contacts from an open position to a closed position. The advantage of relays is that it takes a relatively small amount of power to operate the relay coil, but the relay itself can be used to control motors, heaters, lamps or AC circuits which themselves can draw a lot more electrical power. Relays are used where it is necessary to control a circuit by a low-power signal (with complete electrical isolation between control and controlled circuits), or where several circuits must be controlled by one signal.

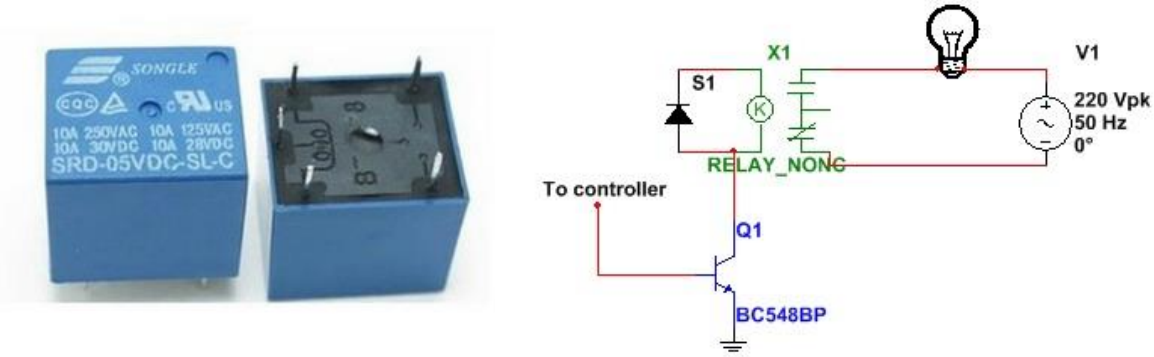


Figure 2.34 Relay and circuit diagram

The design and types of relay switching circuits is huge, but many small electronic projects use transistors and MOSFETs as their main switching device as the transistor can provide fast DC switching (ON-OFF) control of the relay coil from a variety of input sources.

Components used:

Diode 1N4007: It is a popular 1.0 A (ampere) general-purpose silicon rectifier diode.

Transistor BC548: It is a general-purpose NPN bipolar junction transistor.

➢ **Algorithm:**

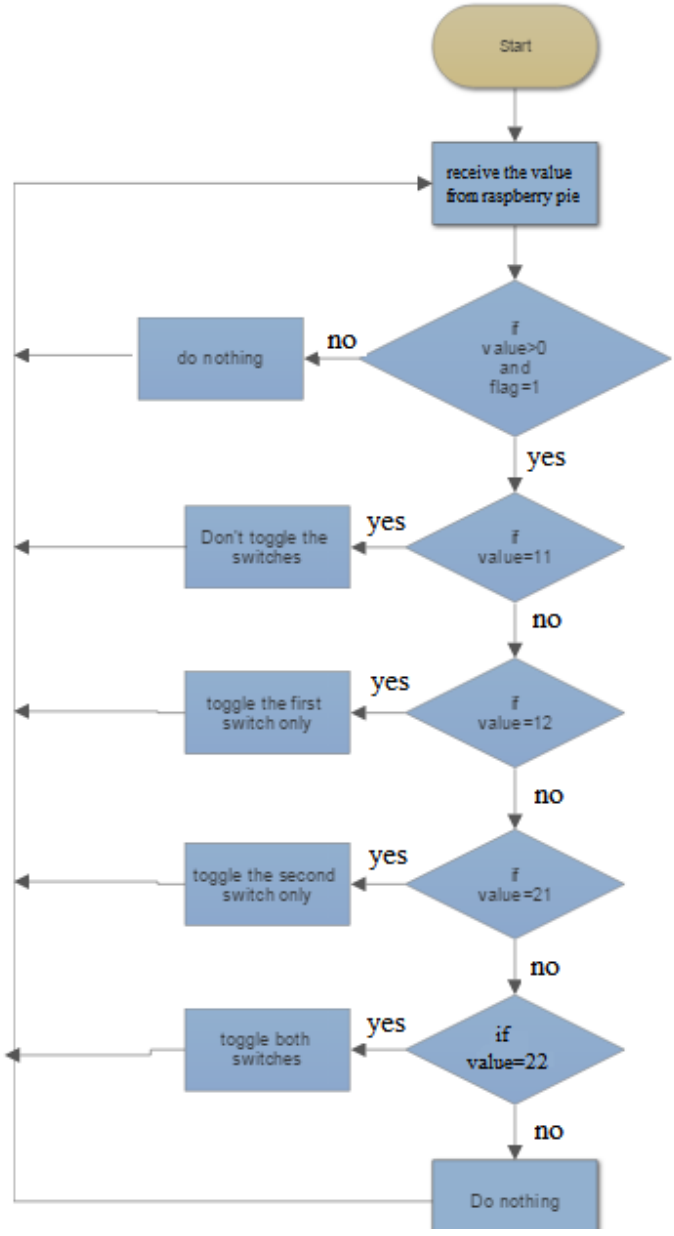


Figure 2.35 Relay Algorithm

- **ATMEGA 16 Microcontroller**

Pin Descriptions:
VCC: Digital supply voltage.

GND: Ground.

Port A (PA7..PA0): Port A serves as the analog inputs to the A/D Converter. Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit).

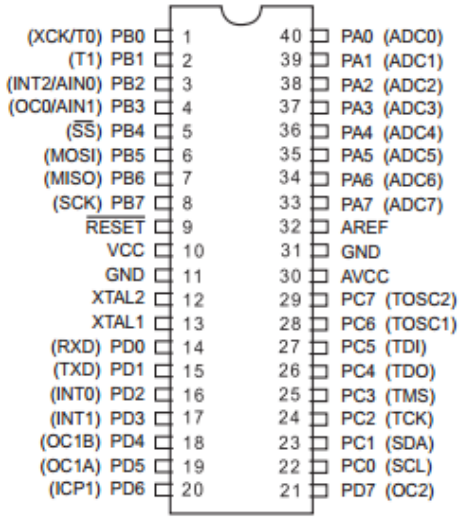


Figure 2.36 Atmega16 Pin Diagram

Port B (PB7..PB0): Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port C (PC7..PC0): Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability.

Port D (PD7..PD0): Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

RESET: Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running.

XTAL1: Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

XTAL2: Output from the inverting Oscillator amplifier.

2.2.6. **Infrared Remote**

The remote is made via cloning the already existing IR codes used via the TV, ACs and the music systems. The codes are received using the IR receiver and stored in the library, so that the same can be transmitted when that function is used from our application. The IR LED is directly connected to the Atmega16 which is giving the encoded hex code in the form of a pulse to the led which is received by the music system and function is performed.

We are using the NEC Protocol. The NEC IR transmission protocol uses pulse distance encoding of the message bits. Each pulse burst (mark – RC transmitter ON) is 562.5µs in length, at a carrier frequency of 38 KHz (26.3µs). Logical bits are transmitted as follows:

- Logical '0' – a 562.5µs pulse burst followed by a 562.5µs space, with a total transmit time of 1.125ms
- Logical '1' – a 562.5µs pulse burst followed by a 1.6875ms space, with a total transmit time of 2.25ms

When a key is pressed on the remote controller, the message transmitted consists of the following, in order:

- a 9ms leading pulse burst (16 times the pulse burst length used for a logical data bit)
- a 4.5ms space
- the 8-bit address for the receiving device
- the 8-bit logical inverse of the address
- the 8-bit command
- the 8-bit logical inverse of the command
- A final 562.5µs pulse burst to signify the end of message transmission.

The four bytes of data bits are each sent least significant bit first. Figure 1 illustrates the format of an NEC IR transmission frame, for an address of 00h (00000000b) and a command of ADh (10101101b).

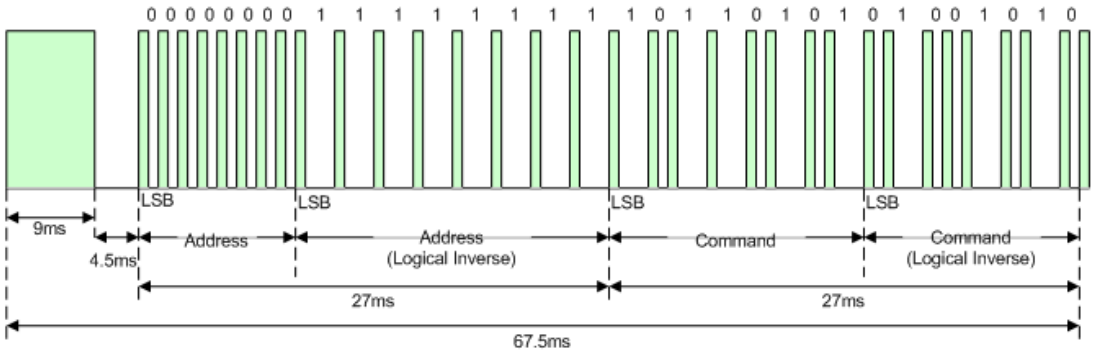


Figure 2.37 IR Remote Signal

Notice from Figure that it takes:

- 27ms to transmit both the 16 bits for the address (address + inverse) and the 16 bits for the command (command + inverse). This comes from each of the 16 bit blocks ultimately containing eight '0's and eight '1's - giving (8 * 1.125ms) + (8 * 2.25ms).
- 67.5ms to fully transmit the message frame (discounting the final 562.5µs pulse burst that signifies the end of message).

2.2.7. Algorithm

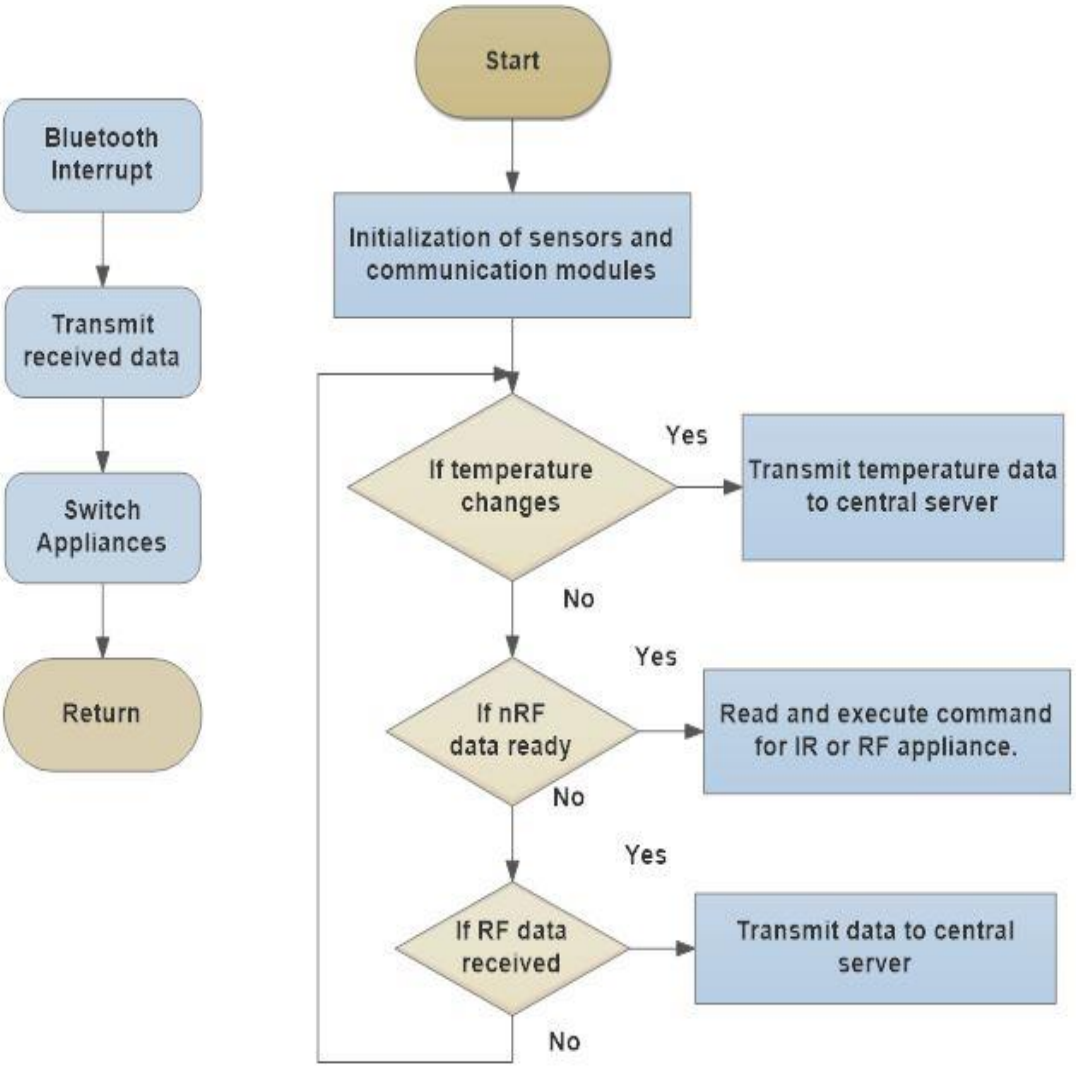


Figure 2.38 Local microcontroller algorithm

2.3. Washing Machine

- Washing machines get clothing clean by plunging the clothes through the water and detergent mixture. It is the motion that really helps to loosen dirt. In the old days, they used to beat wet clothes against a rock to get them clean. In today's world of advanced technology, different types of washing machines are available in the market, two of the main types of washing machines are fully automatic or semi-automatic. Fully automatic machines are more convenient, but semi-automatic machines allow us more control (much like you have with hand washing), and they also tend to be a little more budget-friendly.

- Fully automatic machines change from one cycle to another, taking the clothes right through from start to finish without any additional input from you. Each cycle clicks over to the next automatically, making the fully automatic washing machine one of the most advanced washing machines on the market.

- The working principle of a semi-automatic washing machine is quite different. While the cycles are quite similar, the way the machine works contrasts with the workings of the fully automatic machine. These machines work due to manual input – you tell the machine exactly what to do. This does have its advantages, such as allowing you to use only the water you need, and it gives you more control over how your clothes are washed. Once you've added your detergent and water, you use the settings to tell the machine to agitate, which acts as the semi-automatic washing machine's version of the wash cycle.

- The design of washing machines vary by manufacturer, but the general principles are essentially the same. The main aim of our project is to implement and show how the server communicates with the machine in real-time. In simple words, we have to show how we can easily operate each function on the integrated control panel of the machine from our android application. And hence the hardware includes the integrated control panel as available on the washing machine. As already explained, since semi-automatic washing machines give us more control and are more budget friendly, we have designed our hardware and software working on the principle of the semi-automatic washing machine.

Circuit Working:

- A simple PCB design to show the working of the controller for the washing machine.

- The integrated control panel consists of a microcontroller Atmega8 based board with I/O interfaces and a control algorithm running in it. Input interface consists of LEDS and push buttons for selecting the desired control inputs. It consists of 3 push buttons for selecting 3 control inputs.

- The 1st button allows selecting of any of the two options namely *fill* and *drain*. While *fill* option is selected, the 2nd button is used to go to water level selector namely *low*, *medium* and *high* .The 3rd button is for the wash type selector namely *wash* and *dry*. Also the control board consists of a switch for locking the selected input options. Once the switch is closed after the desired input options are selected, and the power LED glows, it signifies

that the washing machine is in function now, and hence the control inputs cannot be changed.

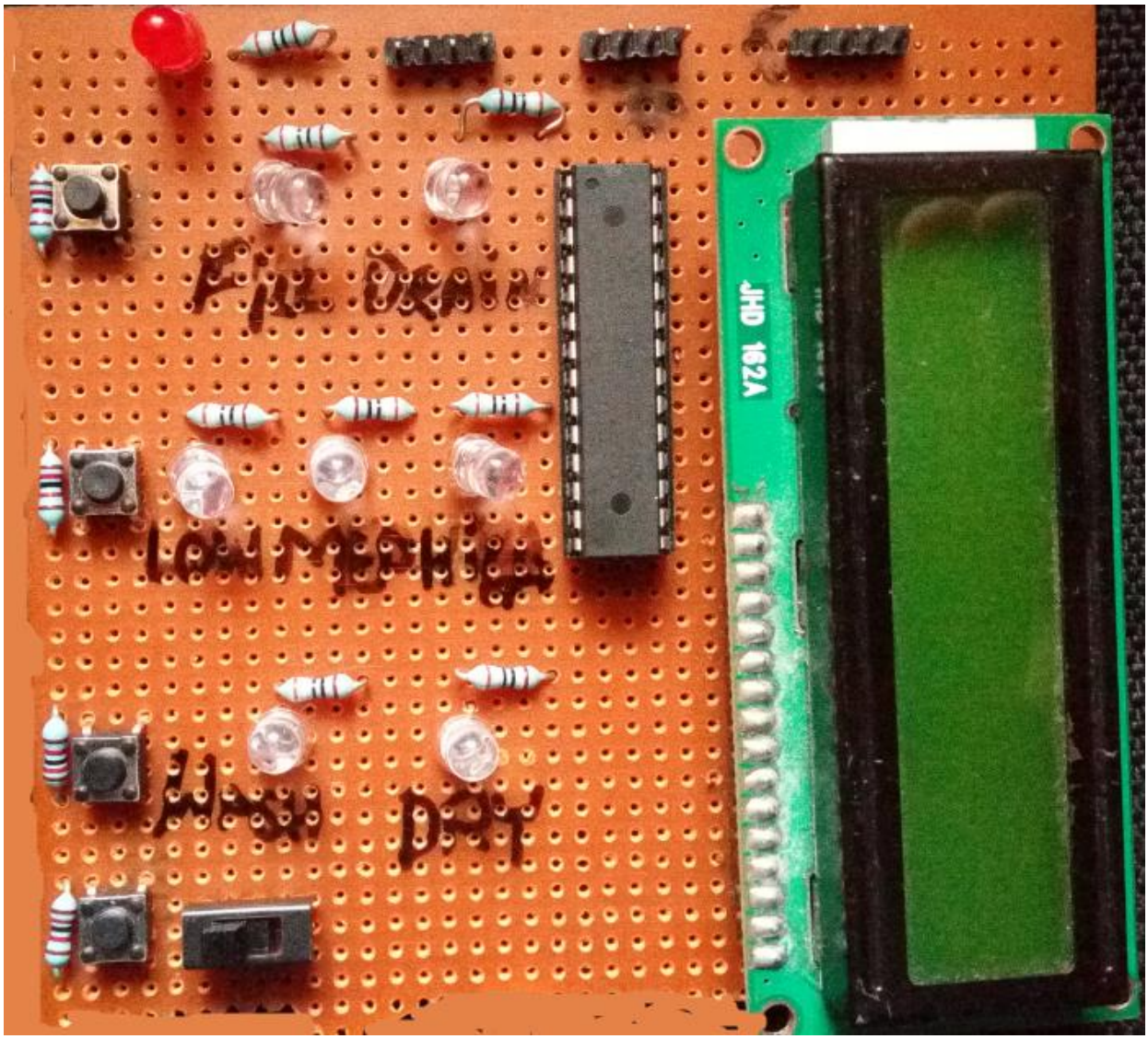


Figure 2.39 Washing Machine Control Unit Circuit

The output interface consists of LCD display, status indication LEDs and the power LED, connected to the microcontroller. The LCD display gives the status indicating what operation has been performed by the washing machine. It also shows the time remaining for a particular operation to get completed.

- **Algorithm:**

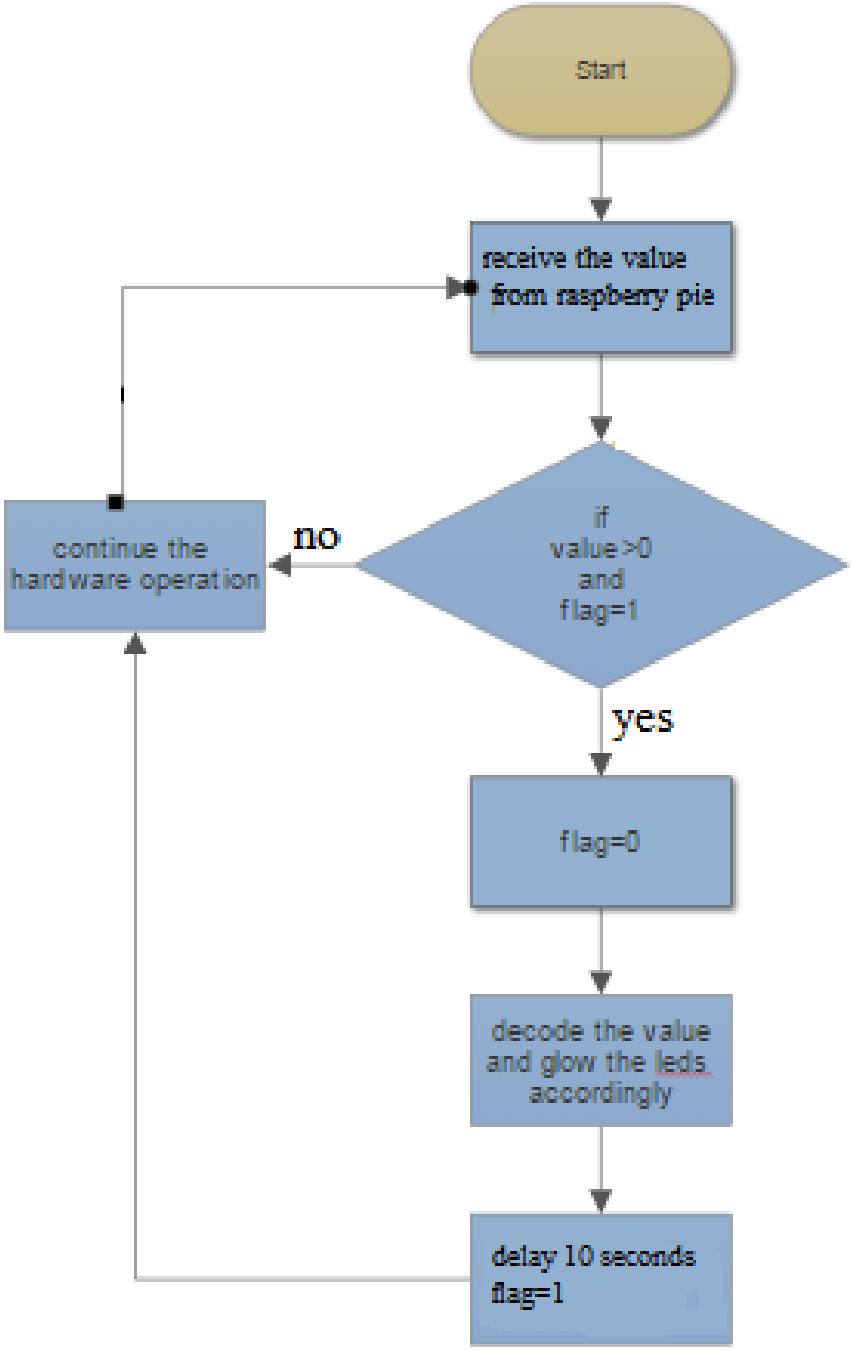


Figure 2.40 Washing Machine algorithm

- ATMEGA8 Microcontroller

  A microcontroller is a computer control system on a single chip. It has many electronic circuits built into it, which can decode written instructions and convert them to electrical signals. The microcontroller will then step through these instructions and execute them one by one. Microcontrollers are now changing electronic designs. Instead of hard wiring a number of logic gates together to perform some function we now use instructions to wire the gates electronically. The list of these instructions given to the microcontroller is called a program.

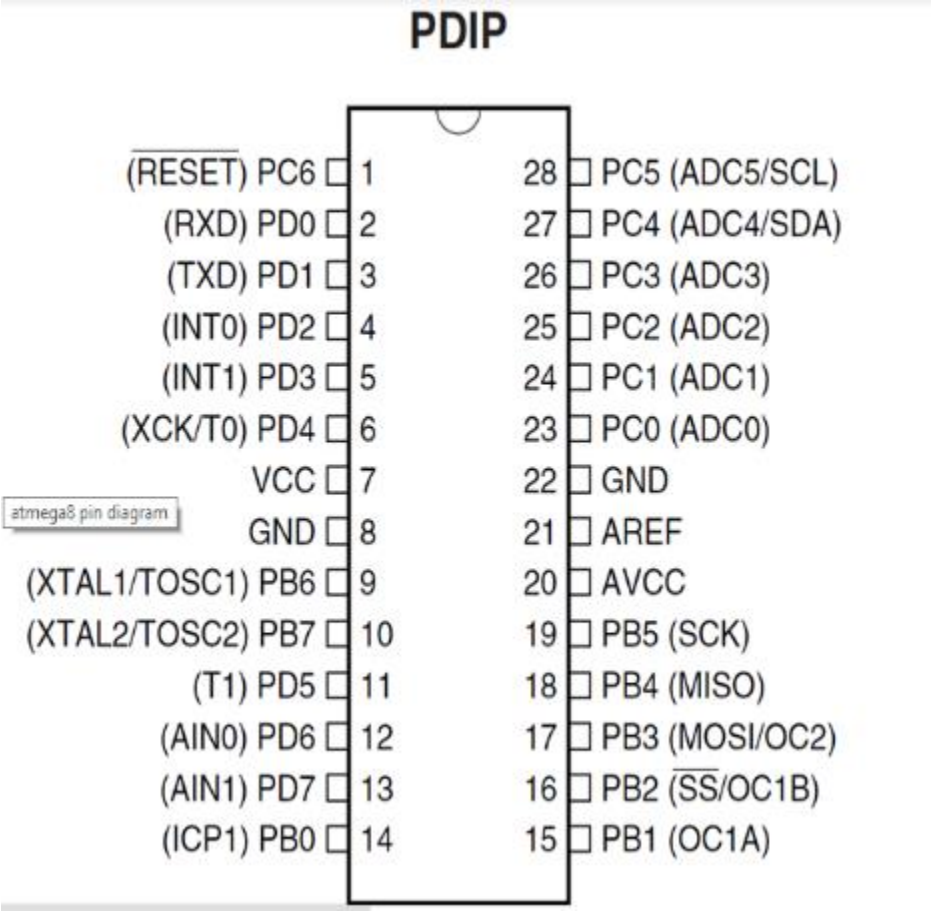


Figure 2.41 Atmega8 Pin Diagram

Memory: It has **8 Kb** of Flash program memory (10,000 Write/Erase cycles durability), **512 Bytes** of EEPROM (100,000 Write/Erase Cycles). **1Kbyte** Internal SRAM

I/O Ports: 23 I/ line can be obtained from three ports; namely Port B, Port C and Port D.

Interrupts: Two External Interrupt source, located at port D. 19 different interrupt vectors supporting 19 events generated by internal peripherals.

Timer/Counter: Three Internal Timers are available, two 8 bit, one 16 bit, offering various operating modes and supporting internal or external clocking.

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

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

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

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

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

2.4. Central Server - Raspberry Pi 2

Raspberry Pi 2 is a small, powerful single-board computer. It is used as a central data processing and storage unit. It acts as a central server for getting data and intercommunication between different processes[19].

Figure 2.42 Raspberry Pi 2 with nRF24L01+ module

Key Specifications:

- 900 MHz quad-core ARM Cortex-A7 CPU
- 1 GB RAM
- 4 USB ports
- 40 GPIO pins
- Ethernet port
- Micro SD card slot
- VideoCore IV 3D graphics core
- Capable of running full range GNU/Linux distributions.

2.4.1. Initial Setup

- Setup of server name, user-id and password for security.
- Network settings such as wlan configuration, IP addressing and proxy settings are set.
- OS updates and upgrades are performed.

- Following software packages are installed for software development and hardware control.
    - Python2.7, cython, scipy. – Python Libraries
    - OpenCV 2.4.1, ffmpeg, libtbb, cmake, git, numpy. – OpenCV and support libraries for image processing.
    - Mysql-server, mysql-client – database management libraries.
    - Wiringpi – Raspberry pi 2 I/O library.
    - Similar support libraries were installed for various applications and is described in detail in the software section.
- Setup the I/O pins for serial and SPI communication in the system.
- Hardware drivers for camera, wireless LAN adapter and other external hardwares installed.
- A shell script is put in the CRON job list to periodically check for disruptions in network connectivity and reconnection.
- Firewall and encryption-key setup to allow only registered users.

2.4.2. Local microcontroller communication

nRF module is connected to the SPI interface of the raspberry pi 2 and is used to exchange data with the local microcontroller units in the rooms. The nRF module used can be seen in the FIGURE .The data communication is similar to that explained in SECTION. Events triggering transmission and reception events are explained in SECTION.

2.4.3. Wi-Fi communication

Wi-Fi communication is used to communicate with the end-users and isolated appliances like washing machine.

2.4.3.1. Appliances:

- Server

  For receiving data a server is setup on the port 8525 which listens for incoming TCP connections from appliances. Upon receiving a connection from registered device or appliance the data is extracted from a buffer and updated in database. Netcat application is used to create the server.

  Command: nc –l 8525 >abc

- Client

Boost asio library is used to connect to the server running on the appliances. The IP address and the remote port is specified for connection. Data to be transmitted is stored in a buffer and sent via WiFi adapter to the remote host in the local network.

2.4.4. Database

MySQL database is used to maintain device and system information. All the data from users are updated in the database and this data is read and transmitted to the respective units. The User ID data is also maintained here for authentication purposes. Code for updating and creating the database is explained in detail in the software section. The description of key tables is as follows:

- Devices Table

  This table contains the Device ID, Status, and flags for status change and notifying change in data.
- Users Table

  This table contains the registered user IP address, MAC address and corresponding password for user authentication.
- Image Table

  This table contains the path to the images for intruder detection system which is to be sent to the user's smartphone.
- Notification Table

  This table contains the emergency notifications to be sent to the user like smoke alarm or LPG gas detection and system failure.

- If a status is changed in Devices table by a user, the status change flag is set and the data is transmitted to the appliance using nRF or Wi-Fi.
- When data is received from a local microcontroller or appliance the flag is set and the data is updated when the android application is opened again.

2.4.5. Algorithm

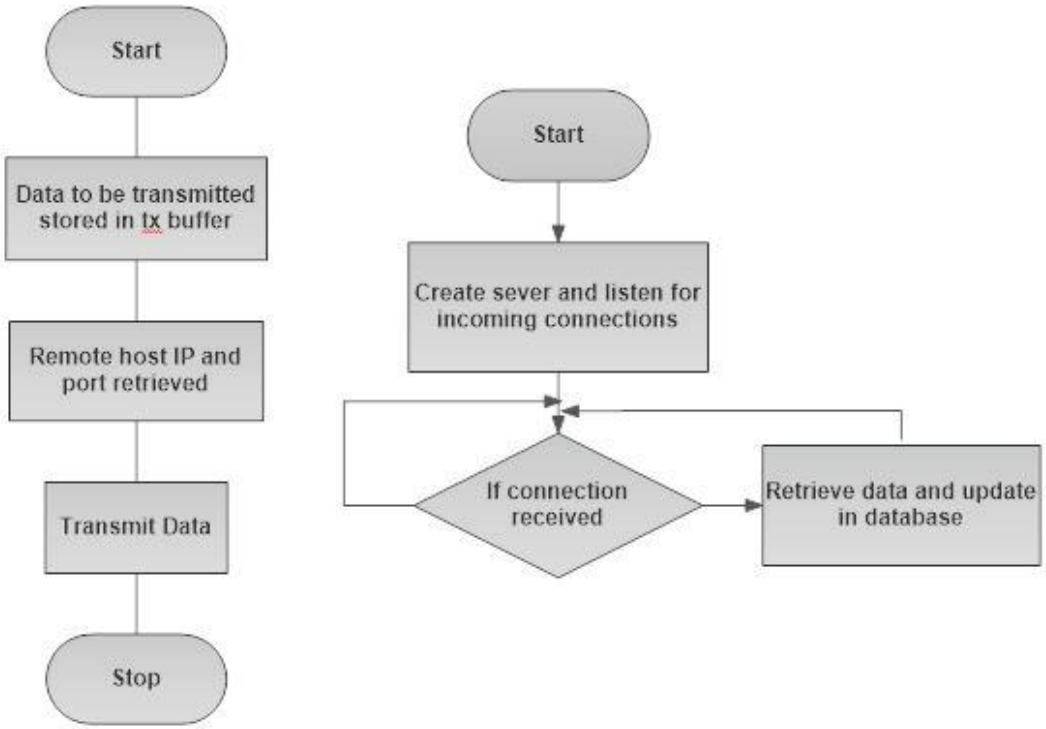


Figure 2.43 Raspberry Pi 2 Wi-Fi algorithm

2.4.6. Monitoring the System

- Database can be either monitored remotely by logging in to the database server using the link: 192.168.0.1/phpmyadmin/
- A portable display can be connected to the raspberry pi's HDMI port.
- Remote SSH connection can be established with the raspberry pi using its IP address and encryption private key.

3. Video Surveillance

3.1 Introduction

3.1.1 What is video surveillance?

Surveillance basically means observing the activities, behavior, or other changing information, of people for the purpose of protection or security [20]. This can include observation from a distance by means of sensors, electronic equipment (such as CCTV cameras), or interception of electronically transmitted information (i.e. monitoring emails, calls, etc.); and it can include simple, relatively no or low-technology methods such as human intelligence agents and postal interception.

Video surveillance is specifically that method used for surveillance which uses electronic equipment and devices such as CCTV cameras. Depending on the size of the region being surveyed, one may need a single or more cameras. The traditional system using CCTV cameras stores the video footage for accessing at a later time. Hence, these traditional surveillance systems need a large amount of memory to store all of the security footage captured day by day. Since it is not possible to store all of the acquired footage for an indefinite amount of time, the security footage is stored for an average of 30 days, and then it is discarded.

But with advancement in the field of image and video processing, the surveillance system has been developed further, and its functionality has now extended from storing video footage for manual observation to applying image processing algorithms to extract out meaningful information from the obtained video and give the processed information as output. Such type of surveillance system is known as a Smart Surveillance System [21]. Smart Surveillance Systems need the system to be responding to the data being obtained in real time. Very fast and robust analysis is required to process something in real time.

In recent years, there has been an effort to constantly produce newer and better systems for video surveillance, while decreasing their cost, hardware and processing requirement and delays. There has been a focus on making systems real time in their implementation and processing. Also, the systems are expected to have a low maintenance cost. As a result, different types of video surveillance systems are available, providing a wide range of functionalities depending on the businesses where they are used.

The systems which are being developed in the recent years are designed in such a way that they need no human intervention [22]. The main focus of these systems is to provide a completely

automated surveillance with the use of sophisticated algorithms and efficient memory management techniques. High end applications also involve cameras which have certain degrees of freedom so that they can track an object of interest.

3.1.2 Need for automated video surveillance?

Security is of prime importance in today's world and surveillance can form an important component of ensuring that security. The need for effective and efficient surveillance systems is now more than ever, given that the rate of crimes is constantly on the rise. The traditional surveillance systems contain recording cameras which constantly record the video footage to be played at a later instant of time, in cases of emergency or some incident.

The reason why we cannot rely on this kind of model of video surveillance is that there are a lot of loop holes in its basic functioning. The human monitored system is endangered by human errors like lack of attention to details, fatigue and also the large cost and sometimes the unavailability of manual labor. Also, in order to spot a particular event, we would need to go through hours of recording of the video footage and still might not be able to obtain worthwhile results. Hence, an automated video surveillance system is necessary to curtail these problems and provide a highly effective and efficient surveillance and security.

3.2 Choice of Image processing tool

3.2.1 What is OpenCV?

OpenCV [OpenCV] is an open source [23] computer vision library available from http://SourceForge.net/projects/opencvlibrary. The library is written in C and C++ and runs under Linux, Windows and Mac OS X. There is active development on interfaces for Python, Ruby, Matlab, and other languages.

OpenCV was designed for computational efficiency and with a strong focus on real time applications. OpenCV is written in optimized C and can take advantage of multicore processors. If you desire further automatic optimization on Intel architectures [Intel], you can buy Intel's Integrated Performance Primitives (IPP) libraries [IPP], which consist of low-level optimized routines in many different algorithmic areas. OpenCV automatically uses the appropriate IPP library at runtime if that library is installed.

One of OpenCV's goals is to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly. The OpenCV library contains over 500 functions that span many areas in vision, including factory product inspection, medical imaging, security, user interface, camera calibration, stereo vision, and robotics. Because computer vision and machine learning often go hand in hand, OpenCV also contains a full, general-purpose Machine Learning Library (MLL). This sub-library is focused on statistical pattern recognition and clustering. The MLL is highly useful for the vision tasks that are at the core of OpenCV's mission, but it is general enough to be used for any machine learning problem.

3.2.2 Why OpenCV?

OpenCV is mainly used due to its high computational efficiency and suitability for real time applications. It provides a large range of functions for operations on images and also provides a high degree of flexibility. It is versatile enough to be used with many languages and on almost any embedded platform.

3.2.3 MATLAB vs OpenCV

1. Speed: Matlab is built on Java, which in turn is built upon C. So when a Matlab code runs, it is first interpreted and converted to Java. Then the code is finally executed. OpenCV, on the other hand, is basically a library of functions written in C/C++, and it also works well with many other languages. There is less amount of interpretation needed in this case. So ultimately you get more image processing done for your computers processing cycles, and not more interpreting. As a result of this, programs written in OpenCV run much faster than similar programs written in Matlab. OpenCV has a very high speed of execution.
2. Resources needed: Due to the high level nature of Matlab, it uses a lot of systems resources, needed over a gigabyte of RAM to process video frames. In comparison, OpenCV programs only require only around 70mb of RAM to run in real-time. This leaves the processor free for other tasks.
3. Cost: MATLAB is distributed with a single user license (commercial version) and needs to be purchased. OpenCV on the other hand is free.
4. Development Environment: The Matlab development environment is based on Java and hence when installing Matlab, the programming environment and IDE are installed as well. For

OpenCV, there is no particular IDE that can be used. Instead, there is a choice of using any IDE depending on the programming language and the operating system used.

5. Memory Management: OpenCV is based on C and hence whenever a chunk of memory is allocated, it will have to be released. If the code runs in a loop where a chunk of memory is allocated in every loop and the programmer forgets to release it, the program will continue to use increasing amounts of memory until it eventually crashes. This is called a 'leak'. Matlab on the other hand is efficient in memory management and it automatically allocates and releases memory in the program. Although OpenCV has a very small overall memory footprint in the beginning, in case of a memory leak, it can be a huge problem and might also be difficult to track down and debug.
6. Portability: Both Matlab and the OpenCV can be installed and run on Windows, Linux and OS X. However, OpenCV runs efficiently on almost all of the embedded platforms, and this makes it suitable for use in embedded system applications. Also, since OpenCV runs without a lot of overhead, it is used on embedded platforms where real time processing is needed.
7. Debugging: Many of the standard debugging operations can be used with both Matlab and OpenCV: breakpoints can be added to code, the execution of lines can be stepped through, variable values can be viewed during code execution etc. Matlab however, offers a number of additional debugging options over OpenCV. One great feature is that if you need to quickly see the output of a line of code, the semi-colon at the end can be omitted. Also, as Matlab is a scripting language, when execution is stopped at a particular line, the user can type and execute their own lines of code on the fly and view the resulting output without having to recompile and link again. Added to this is are Matlab's powerful functions for displaying data and images, resulting in Matlab being our choice for the easiest development environment for debugging code.
8. Help and Sample Code: Matlab features a very elaborative and informative 'help' section in their software. This section has a list of all functions along with their outlines and usage, along with some sample code which demonstrates its use. In contrast, OpenCV does not have a lot of documentation available, but it has plenty of sample codes available which can enable a user carry out pretty much any operation using the OpenCV functions. When a new image processing method is developed, a new implementation in OpenCV is soon made available for download.

3.3 Conventional Approach

3.3.1 Background Subtraction

Background subtraction, also known as Foreground Detection, is a technique in image processing and computer vision where an image's foreground is extracted for further processing (object detection, motion tracking, object recognition etc.) [24] Generally, the regions of interest in an image are objects (humans, cars, text etc.) in its foreground. After the preprocessing on image in completed (which may include image denoising, morphology etc.) object localization is needed and this may make use of the technique of background subtraction. Background subtraction is a widely used approach for detecting moving objects in videos from static cameras.

The logic behind this approach is that the background image and the foreground image are used and compared to detect moving objects in the foreground image frame. Background subtraction is done mostly when the input is a continuous video stream where the current frame changes when there is motion of objects in the video. Background subtraction provides important information for a large number of applications in computer vision. One such example is the tracking of objects for surveillance. However, the general assumption following the use of the background subtraction algorithm is that the background remains entirely static and the objects of our interest move in the foreground. However, this is not the case for real environments in which even the objects in the background can keep moving continuously. In indoor regions, shadows, reflections or moving objects on screens lead to background changes. In a same way, due to wind, dust, rain or illumination changes brought by weather, static backgrounds methods have difficulties with outdoor scenes.

3.3.2 Pixel to pixel matching

In this method, the foreground and the background images are compared pixel by pixel to find out the difference between these two images. The difference obtained is then processed further to find out the necessary information [25].

3.4 Problems with conventional approach

3.4.1 Illumination/ Shadow

Irregular Illumination or varying brightness in an image is one of the most important problems in video surveillance and it has an adverse effect on the performance of the system. Illumination changes create problems if our regions of interest for image processing are edges of objects in the image or image intensities. Shadows produce false edges and dark regions and varying illumination conditions can cause texture changes which cannot be accounted for. Irregular illumination of the image may be caused by faults in the sensor, non-uniform illumination of the scene, or the orientation of the object's surface or light source.

Also, these illumination conditions and shadows may change dynamically in every frame of a video and hence falsely give the notion of a moving object in the video. Hence, illumination and shadow correction is very essential if we are to detect moving objects in a video.

3.4.2 Noise

Image noise [26] is a random variation of brightness or color intensities in images, and is usually a component of electronic noise. The noise does not carry any information in the image and occurs randomly. Images can have varying amounts of noise, and an image captured in good light with a good quality camera will have a very small amount of noise which an optical or radio astronomical image will contain huge amounts of noise from which only a small amount of information can be extracted by sophisticated processing algorithms. Noise beyond a certain level can be unacceptable in the image because it might lead to a complete loss of the features of interest in the image, and this loss might be irrecoverable. The three kinds of noise most commonly found in images are –

a) Impulse noise,

b) Additive noise and

c) Multiplicative noise.

Every kind of noise has its own characteristics and properties which differentiate it from other kinds of noise. Also these properties are taken into account while devising methods to remove this noise from images. The most commonly occurring noise in images is the random Gaussian noise and it occurs mostly due to improper lighting conditions.

SOURCES OF NOISE IN IMAGES

Any kind of signal is expected to contain two kinds of noise – one, the noise added while the signal is captured and two, the one the signal picks up while it is being transmitted through a medium. There are various factors which are responsible for introducing noise in the image. An image can contain various amounts of noise. The number of pixels which have been affected by the noise will decide the amount of noise which has been added. The major sources of noise in the digital image are:

a) Effect of environmental conditions on the sensor (camera) capturing the image.

b) Inadequate illumination at the scene and fluctuations in the temperature of the sensor.

c) Noise introduced in the transmission channel due to interference from other sources.

d) Blurring of the camera screen due to moisture, dust particles and so on.

TYPES OF NOISE IN IMAGES

Noise is an undesirable effect which is produced in the image and is mostly a result of non-ideal conditions. It is not desirable since the noise does not carry any information and only works to deteriorate the quality of the image leading to loss of information. Several factors are responsible for the introduction of noise in images during acquisition and transmitting. The algorithms used for removal of noise are developed keeping in mind the kind of noise they need to work on and hence depend on the specific characteristics and properties of that noise. So our focus is to first identify the kind of noise present in the image so that the appropriate technique can be used for its removal. Image noise can be classified as follows:

A. Impulse Noise (Salt and Pepper Noise)

The salt and pepper noise is an impulse noise, meaning that the effects of this noise appear as impulses of an intensity that is different from that of the background pixels in parts of the image [108]. It is also known as random noise or spike noise. The name salt and pepper has been given since as a result of this noise, black and white dots appear in the image as noise. This noise arises in the image because of sharp and sudden changes in image signal. Dust particles in the image acquisition source or over heated faulty components can cause this type of noise []. Image is corrupted to a small extent due to noise. Figure 3.1 shows an image containing salt and pepper noise and the original noise free image for comparison.

Figure 3.1(a) Original noise free Image

Figure 3.1 (b) Image with Speckle noise

B. **Gaussian Noise (Amplifier Noise)**

This is the most commonly found noise in images. Gaussian noise model is additive in nature [108] and follows the Gaussian distribution. It means that each pixel in the noisy image is the sum of the true pixel value and a random, Gaussian distributed noise value. The noise is independent of intensity of pixel value at each point. The PDF of Gaussian random variable is :

$$P(x) = \frac{1}{\sigma(\sqrt{2\pi})} e^{\frac{-1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Where: $P(x)$ is the Gaussian distribution noise in image; μ is the mean and σ is the standard deviation of the Gaussian function. Figure 3.2, shows the effect of adding Gaussian noise of zero mean to an image.

C. **Poisson Noise**

Poisson or shot photon noise is the noise that occurs in the image when the number of photons sensed by the sensor is not sufficient to provide detectable statistical information [27]. This noise has a root mean square value proportional to square root intensity of the image. Different pixels have different noise values and the noise distribution at each pixel is independent of the noise at any other pixel.

Figure 3.2 (a) Original noise free Image

Figure 3.2 (b) Image with Gaussian noise

Figure 3.3 (a) Original noise free image

Figure 3.3 (b) Image with Poisson noise

D. Quantization noise (uniform noise)

The noise which results after the image pixel intensities are quantized into discrete levels is known as quantization noise. It has approximately uniformly distributed. The noise can be signal dependent and mainly occurs because the quantization process leads to loss of some useful information.

Figure 3.4 (a) Original noise free image *Figure 3.4 (b) Image with Quantization noise*

E. Speckle Noise

This noise can be modeled by random value multiplications with pixel values of the image and can be expressed as J = I + n*I Where, J is the speckle noise distribution image; I is the input image and n is the uniform noise image by mean o and variance v [27]. This noise is originated because of coherent processing of back scattered signals from multiple distributed points. In conventional radar system this type of noise is noticed when the returned signal from the object having size less than or equal to a single image processing unit, shows sudden fluctuations. Mean filters are good for Gaussian noise and uniform noise.

Figure 3.5 (a) Original noise free image *Figure 3.5 (b) Image with speckle noise*

F. Shot noise

The dominant noise in the lighter parts of an image is typically that caused by statistical quantum fluctuations, that is, variation in the number of photons sensed at a given exposure level. This noise is known as photon shot noise [27]. Shot noise has a root-mean-square value proportional to the square root of the image intensity, and the noises at different pixels are independent of one another. Shot noise follows a Poisson distribution, which is usually not very different from Gaussian.

Apart from deteriorating the quality of the image and giving it an unpleasant look, the image noise also works to reduce or overlap some of the important features in the image due to loss of visibility. This problem is particularly prevalent for objects which have a low contrast. More is the noise in the image; more will be the difficulty in viewing objects with low contrast. There are also some objects which might be visible in low noise levels, but as the noise starts increasing, they are no more visible. Hence, there is a concept of visibility threshold, which is the maximum amount of noise up to which the object is visible and this threshold is very noise dependent, especially for low-contrast objects. In principle, when we reduce image noise, more of the low-contrast objects within the body become visible.

Similar to illumination and shadow, the noise in an image can also vary with every frame and is introduced randomly in an image. Noise varying with every frame might be mistakenly considered as a moving object and might interfere with the working of the algorithm.

3.4.3 Little continuously moving objects

A background image can be considered as an image containing objects which remain inactive in the frame [ ]. In an indoor environment, it may consist of the furniture like chairs, tables and appliances. These are non-living things and are not capable of changing their position on their own. In an outdoor environment, it may consist of buildings, trees, vehicles and poles which are also non-living objects. These can be considered as static objects in the background image.

But the appearance of the background image may change with time. Sometimes, the indoor environment might contain the movement of certain objects like curtains and fans in the background. In an outdoor environment, the system might capture movement of leaves on trees due to flowing breeze. All of these movements are also detected by the system. But since they are

of no significance our desired application, detecting the movements of these objects unnecessarily adds to the processing requirements of our system and slows it down, while also acquiring unnecessary information.

3.4.4 Real time

In general, real time processing implies processing instantly as and when the data is available. This type of processing is usually needed in systems that use computer control and which need to make fast decisions based on an event which has occurred right when the event is occurring. This processing method is used when it is essential that the input request is dealt with quickly enough so as to be able to control an output properly. This is called the 'latency'.

For example, the computer inside the Engine Control Unit in a car has to manage the engine at every moment based on what the driver wants to do. Real time processing has to be programmed very carefully to ensure that no input events are missed.

Due to noise and other problems such as detecting unwanted images, the amount of operations which a system needs to perform increase significantly and this greatly increases the required processing power. A real time system is a combination of both a fast processor and an optimum algorithm.

3.5 Algorithm and Implementation

3.5.1 Illumination correction

The RGB (Red, Green Blue) color space is represented as a cube where all the three parameters R, G and B have values ranging from 0 to 255. This can be seen from figure 3.6. On the other hand, the HSV color space is represented as a cone, with H having a range of 0 to 360, S of 0 to 100 and V of 0 to 255. This is depicted in figure 3.7.

The RGB color space has very little tolerance for changes in illumination. Hence, to counter the effects of illumination, we use the HSV color space (Hue, Saturation and Value (Illumination).

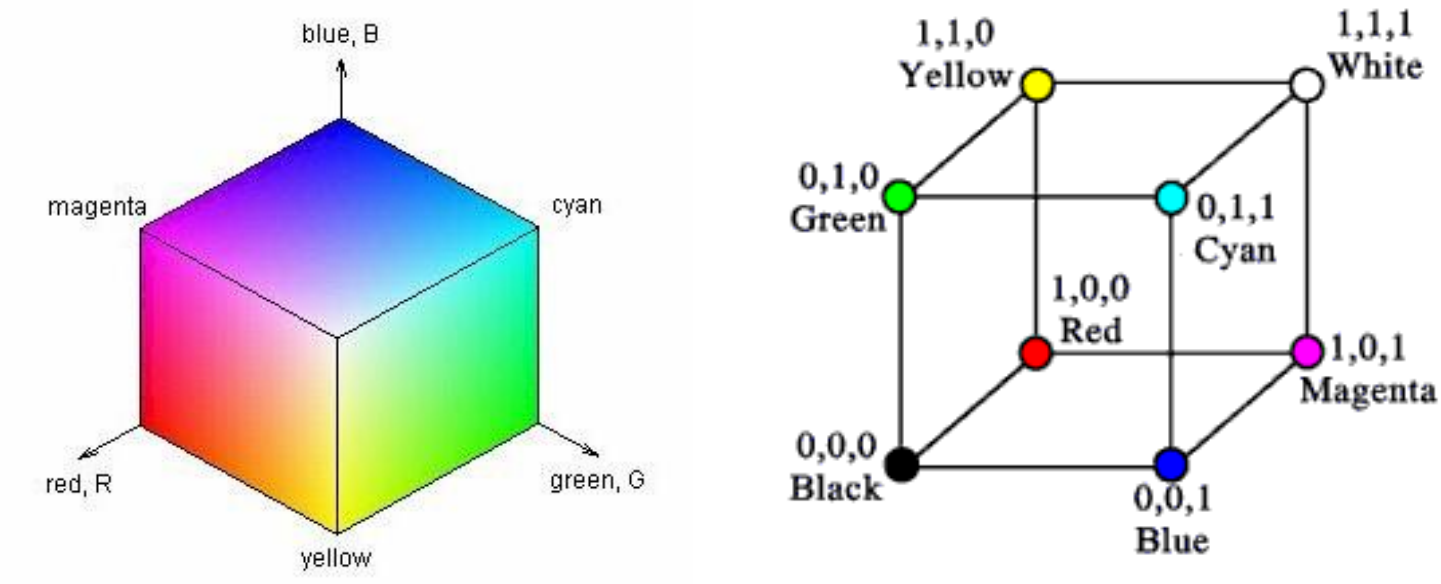


Figure 3.6 RGB (Red, Green, Blue) color space represented as an RGB cube.

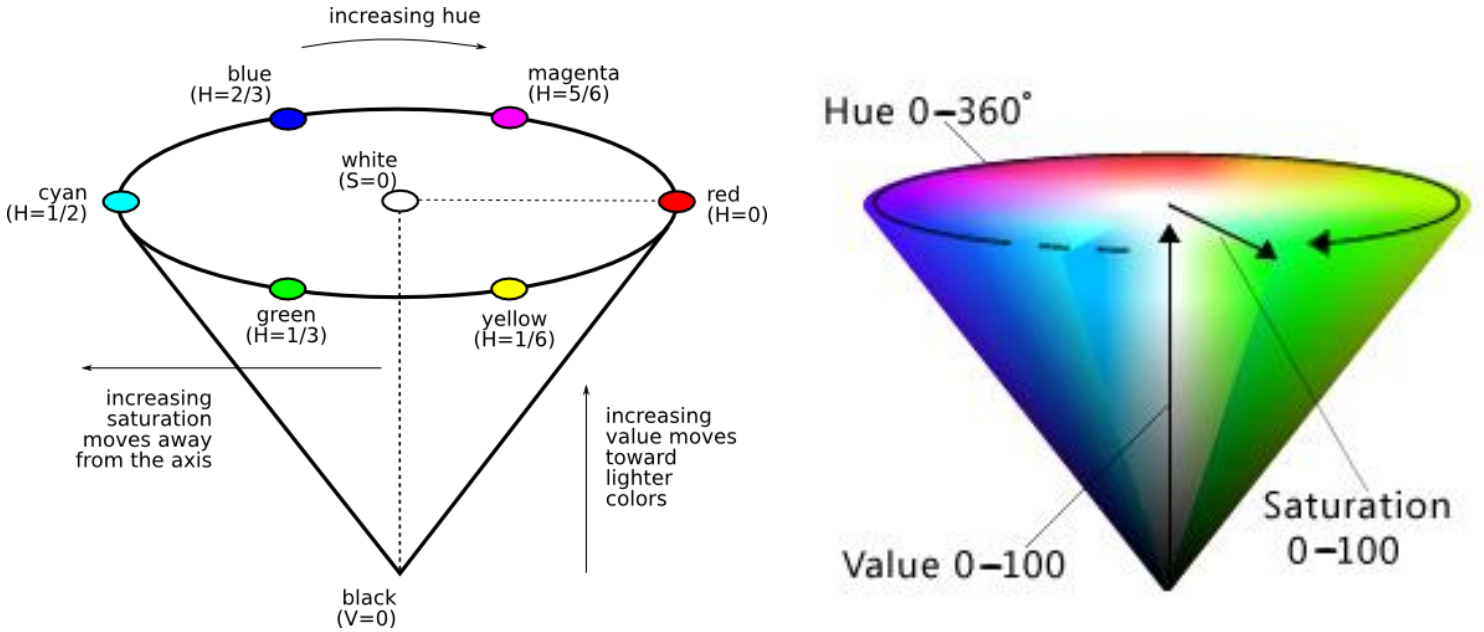


Figure 3.7 HSV (Hue, Saturation and Value) color space represented as an HSV cone.

The value/illumination plane in the HSV color space is similar to a grayscale image and is responsible for the illumination effects in the image. Hence, a conventional approach which can be followed during illumination correction is setting all the pixel values in the illumination plane to some constant integer arbitrarily. This way, the effect of lighting and illumination can be removed. However, changing the value in the illumination plane also affects the Hue and the Saturation planes. So if the lighting conditions are extreme, i.e. if our observation space is either very dark or very bright, setting a constant arbitrary value in the illumination plane may lead to errors in our readings of Hue and Saturation planes.

Figure 3.8 Effect of change in illumination on Hue and Saturation values

Figure 3.8 shows the same building under the effect of different levels of illumination at different times of the day. The change in lighting conditions or illumination has an effect on the overall color of the building. This shows that as the values in the illumination plane change, they also have an effect on the Hue and the Saturation planes, which lead to a change in the color of the object.

Hence, instead of assigning an arbitrary value to the illumination plane, we need to follow an adaptive approach which depends on the current lighting and illumination conditions of the observation space. In this project, we have proposed a way to use weighted threshold of the values in the illumination plane. This can be done by analyzing the histogram of the initial frames at the beginning of the video surveillance, while the background is being set. For illumination correction, we operate on the illumination plane of the HSV image frame.

Histogram

A histogram is a graphical representation of the frequency of numerical data. It is an estimate of the probability distribution of a continuous variable (quantitative variable) and was first introduced by Karl Pearson [ ].

Histograms are useful data summaries that convey the following information:

- The general shape of the frequency distribution (normal, chi-square, etc.)
- Symmetry of the distribution and whether it is skewed
- Modality – unimodal, bimodal or multimodal

The histogram of the frequency distribution can be converted to a probability distribution of the data. In cases in which this distribution is known, a histogram that does not fit the distribution may provide clues about a process and measurement problem.

In terms of image processing, a histogram is a graph between the pixel intensities and their corresponding frequencies of occurrence in the image. Looking at a histogram of an image, we can clearly tell which pixel intensities are occurring more frequently in an image. For a grayscale image, the x axis of the histogram will contain intensities from 0 to 255, corresponding to all the grayscale intensities which can be present in the image. An RGB image contains three planes – Red, Green and Blue. Hence a histogram for an RGB image will contain three graphs, one for each plane and for each graph the values on the x axis will vary from 0 to 255.

For illumination correction, we find the histogram of the illumination plane in the HSV color space. The illumination plane has pixel intensities varying from 0 to 255. In fact, the value plane of an image in the HSV color space is similar to the grayscale equivalent of the image. The histogram of this illumination plane is used to analyze the values of the intensities of pixels in the image which contribute to its brightness.

Weighted Average

Our main aim is to obtain that value of illumination intensity which will lead to illumination correction without causing much change in the Hue and Saturation values. We achieve this by calculating the weighted average of the illumination values of the image, and then replacing all the values in the illumination plane with the weighted average value calculated above.

Threshold

Now there are some pixel intensities in the image which occur very less frequently as compared to other intensities. These might correspond to shadows of some small objects or reflections from the surfaces of objects. These pixels, although a part of the illumination plane, do not decide the overall illumination of the image. Hence, pixel intensities whose frequency of occurrence is less than a particular threshold are not taken into account while calculating the weighted average. In other words, when calculating the weighted average, only those pixel intensities are considered whose probability of occurrence is more than a certain threshold.

This ensures that minor zones or regions in the image with very high or very low brightness do not affect the overall weighted average of the illumination value. Also, this reduces the errors related to noise pixels and shadows.

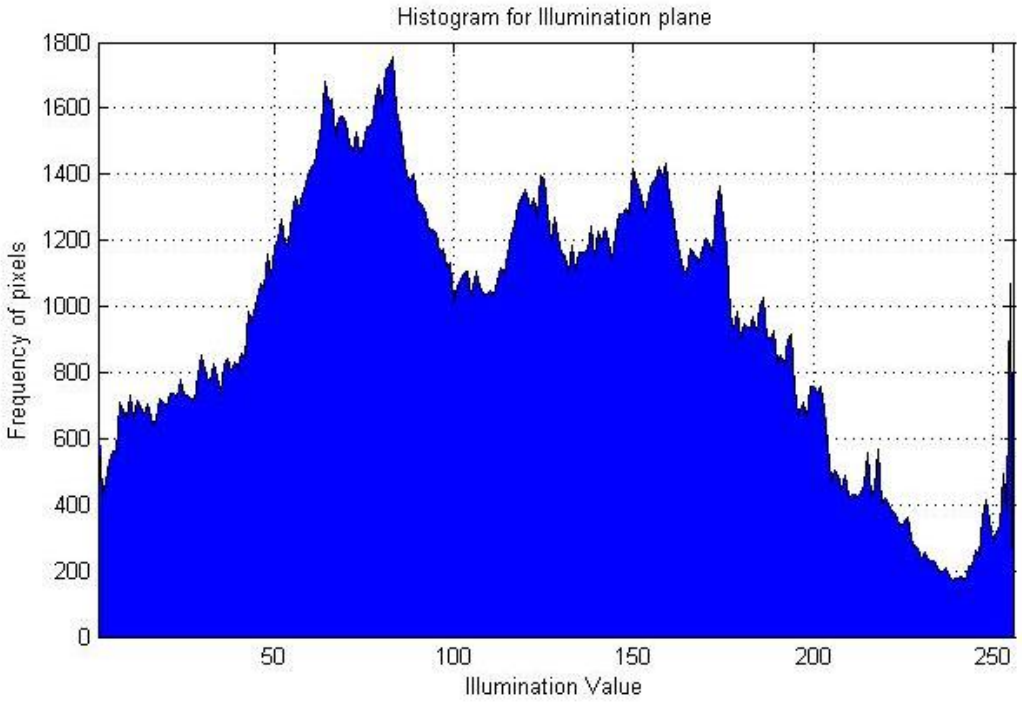


Figure 3.9 Histogram for illumination plane

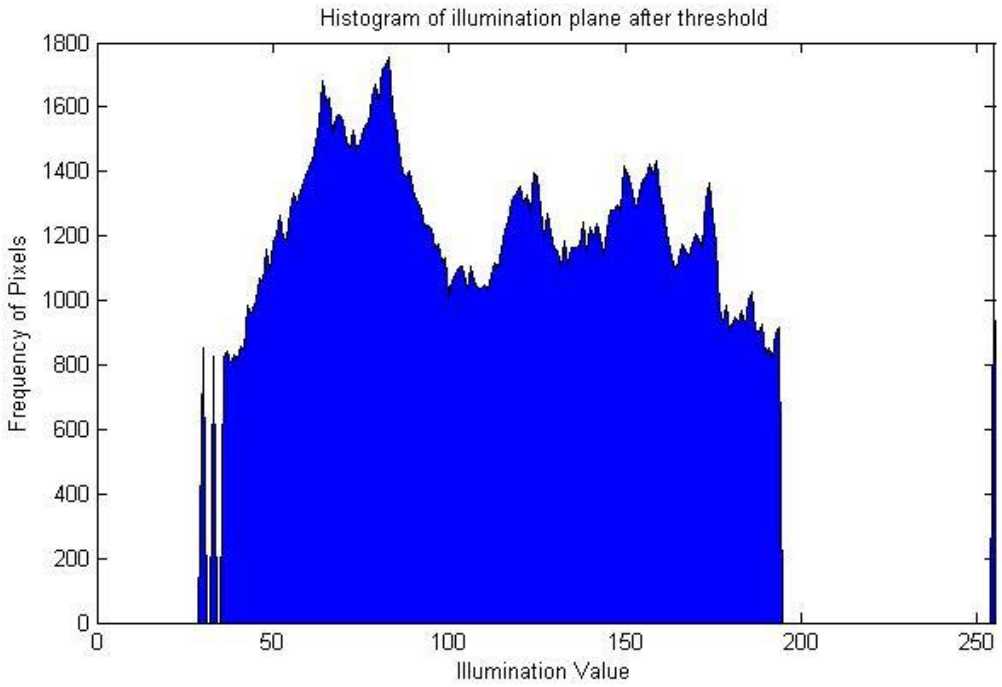


Figure 3.10 Histogram for illumination plane after threshold

3.5.2 Noise removal

In the previous section, we saw the effects of noise on the image. In addition to decreasing the quality of the image, the noise in the image also leads to additional contours at later stages of this method, and hence removing noise is the first of the pre-processing operations which need to be performed on the image.

The noise found in the image is mostly additive white Gaussian noise. There are various types of noise removal techniques which are available for reducing noise in the image to the lowest possible value. The most common is filtering the noisy image using certain filters so that the noise is removed. The filters might be linear or non-linear, and the choice of filter to be applied depends on the type of noise present in the image. Filtering can be done in both spatial and temporal domains and there are certain parameters associated with each filter.

There are various types of filters and techniques which can be used for noise removal –

i. Mean filter
ii. Median filter
iii. Weiner filter
iv. Gaussian filter

The Gaussian filter is the most commonly used noise filter for removing noise in images.

The purpose of the Gaussian filter is to cause blurring since it depends on the standard deviation. Hence, the Gaussian filter does not preserve edges in the images. In fact, it leads to the blurring of edges too. Though this might be considered a disadvantage of the filter, but for the current application, our purpose is to detect motion and hence the object edges are not of much significance to us. Also, using other higher level filters like the Bilateral filter will increase the processing time needed. We cannot allow this since the operation of our system needs to be real time.

3.5.3 Setting the background

The processing for the video surveillance has to be real time, i.e. the frames need to be processed as they come in without any delay. However, the first 10-20 frames can be used for setting the background which is used in the further algorithm. The algorithm is implemented in such a way that after the illumination intensity of the incoming video feed is set, the first 10-20 frames are captured and their average is used as the background image. This background is mostly considered to be static and all the other incoming frames are compared against this background for possible motion of objects in the frame or the entering of foreign objects from outside.

3.5.4 Comparing foreground and background frames

We need some algorithm to compare the reference frame and the current frame. There are various methods to do this. The first approach is to directly calculate the difference between the reference image and the current image by subtracting one from the other.

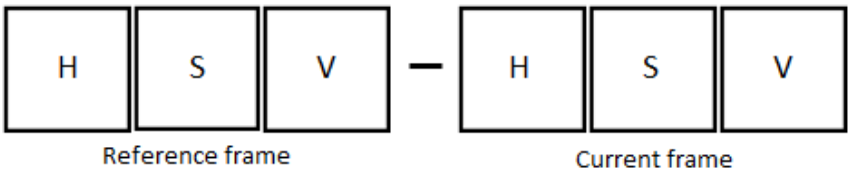


Figure 3.11 Difference image

However, this kind of subtraction will always give a non-zero value and it is difficult to interpret anything from this, because this non-zero value will be a result of changes in the pixel intensities due to both significant and insignificant movement in the current frame.

We need a method so that we can be able to separate the desirable changes from the ones occurring due to noise, slight changes in lighting conditions and small movements which can be neglected. For ensuring this, we use the concept of threshold.

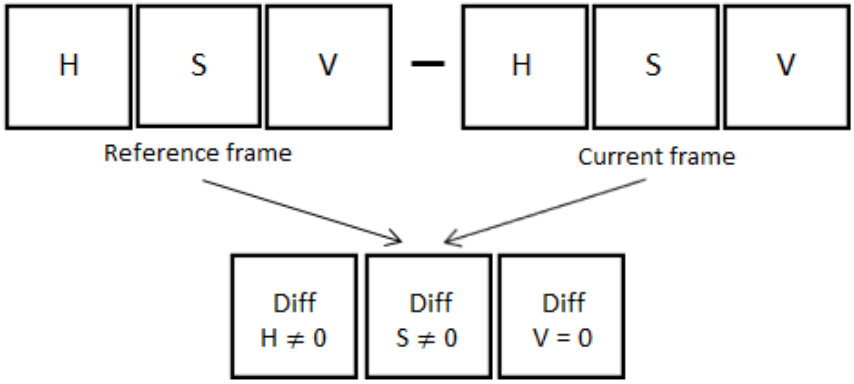


Figure 3.12 Threshold on difference image

Since the value of all pixels in the illumination plane has been set constant for illumination correction, the difference value of the illumination plane will be zero. Hence for detecting motion, we need to analyze the Hue and the Saturation plane. After subtracting the background image from the current image, we have obtained a range of values in the hue and saturation planes. Some of these values are a result of noise and other errors. But some of these also contain the information about motion of objects in the current frame with respect to the background frame. Hence, to separate these desirable pixels from noise, the difference image is binarized using a threshold. Now this binarization has to be done by taking into account both the Hue and the Saturation values. Let d_h, d_s and d_v be the difference matrices of the hue, saturation and value planes respectively. Hence the threshold for binarization is set as:

$$\text{Threshold} = \sqrt{(d_h^2 + d_s^2 + d_v^2)}$$

The pixels which cross this threshold are set as 255 and the ones which fall within this threshold are set to '0'. At the end of thresholding, the pixels which are white should detect the moving object in the background.

3.5.5 Processing

Now that we have obtained a binary image after threshold, it contains the pixel representing motion. However, this representation is still very crude and it needs to be converted to a form in which it can be seen, understood and interpreted by the user. Hence it needs to be processed further to extract out meaningful information about motion in the foreground image from it.

For this, we first –

1. <u>Detect edges in the resulting image using the Canny edge detector</u>

The Canny Edge detector [ ] also known as the optimal detector aims to satisfy three main criteria:

- Low error rate: It detects only real edges and does not take into account false edges.
- Good localization: There is a very minor difference between the pixels marked as edges and the actual edge pixels
- Minimal response: Only one detector response per edge.

<u>Steps to implement the Canny Edge Detector</u>

i. Smoothening: This means filtering out noise. The filter used for this purpose is the Gaussian filter. An example of a Gaussian kernel of size=5 that might be used is shown below:

$$K = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

ii. Finding gradients: We use a process analogous to Sobel for finding intensity gradients in the image.

   a. Apply a pair of convolution masks (in x and y directions):

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \qquad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$

   b. Find the gradient strength and direction with:

$$G = \sqrt{G_x^2 + G_y^2}$$
$$\theta = \arctan\left(\frac{G_y}{G_x}\right)$$

The direction is rounded to one of four possible angles (namely 0, 45, 90 or 135)

iii. The next step is to apply non-maximum suppression to remove pixels that are not considered to be part of an edge. After this step, only thin lines will be left.

iv. Hysteresis: This is the final step where the Canny uses two thresholds (upper and lower):

a. If a pixel gradient is higher than the upper threshold, the pixel is accepted as an edge

b. If a pixel gradient value is below the lower threshold, then it is rejected.

c. If the pixel gradient is between the two thresholds, then it will be accepted only if it is connected to a pixel that is above the upper threshold.

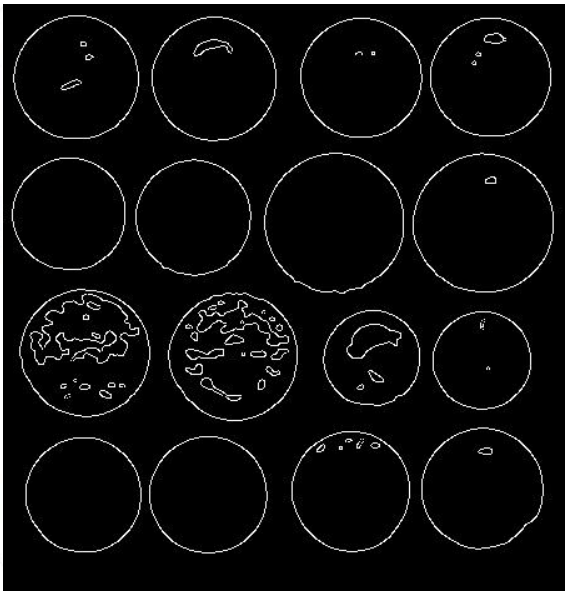


Figure 3.13 (a) Original Image *Figure 3.13 (b) After applying Canny edge detector*

2. Dilate the edges

Dilation is one of the basic operations in mathematical morphology [ ]. Originally developed for binary images, it has been expanded first to grayscale images, and then to complete lattices. The dilation operation usually uses a structuring element for probing and expanding the shapes contained in the input image.

When the dilate function is applied after the Canny Edge detector, it dilates the white edges in the image against the black background. The significance of this operation is that the very small edges detected by the Canny algorithm which are very close to each other merge into each other and form one blob.

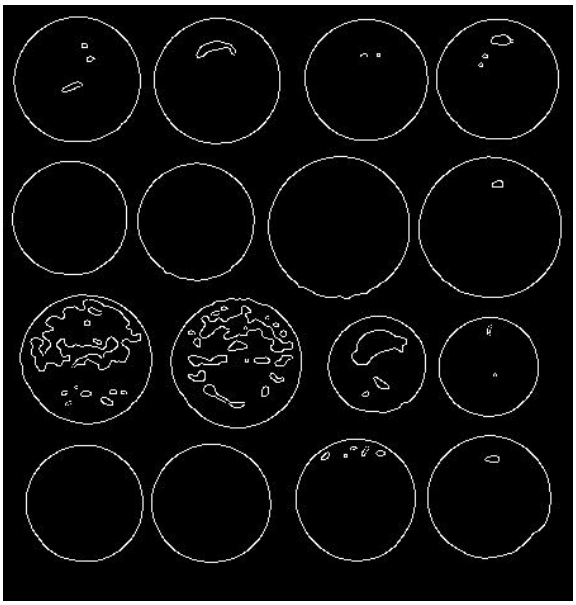


Figure 3.14 (a) Edge detected image

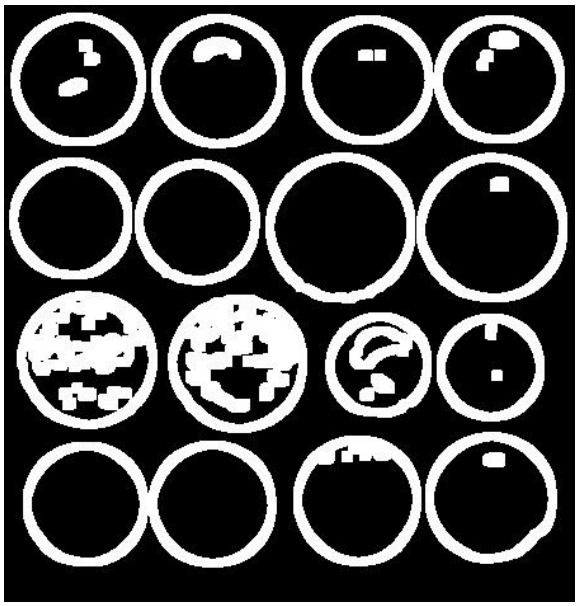


Figure 3.14 (b) Dilated image

3. Detect contours in the image

Contours can be explained simply as a curve joining all the continuous points (along the boundary), having same color or intensity [ ]. The contours are a useful tool for shape analysis and object detection and recognition.

A contour is different from an edge is a way that an object-boundary must and should encompass an object (like cup, face, eyes), whereas for a contour there is no such strict-constraint. A contour could encompass an object (like cup, face, eyes) or a texture or a feature (like corners).

In other words, an object boundary denotes the edges of an object. A boundary can be a contour (but can also be a rectangle or a straight edge; however, an image may have multiple contours.

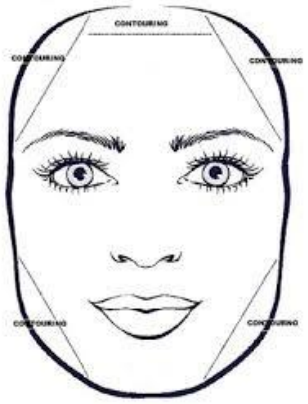


Figure 3.15 Difference between edges and contours

In this image, there is the face boundary, but also contours for the eyes and lips (which can also be viewed as separate objects).

The approach of using contours to detect moving objects has a lot of advantages –

i. Since contour detection is an inbuilt feature in OpenCV, it is highly efficient and capable of computing contours in real time.
ii. The feature also provides the ability to compute contour area i.e. the area enclosed inside a closed curve.
iii. The contour detection function has a parameter which can set the hierarchy of the contours.
iv. We can calculate the centroid or other parameters of the contour to keep track of the object if the camera is moving.

The small amount of noise or illumination effects still present in the image might lead to the forming of very small contours spread over an area instead of one large contour of an object. For e.g. in the figure 3.14 (a) and (b), after applying the canny edge detector function, there are a few unwanted edges formed due to noise or other minor image features. We do not need to take these into account for further processing. This problem can be overcome using the dilate function. Dilation is a morphological operation due to which the white edges detected using Canny edge detector dilate around the black background i.e. the contour boundaries thicken, thus providing one large contour instead of many smaller ones. This can be seen from figure 3.16, in which the noisy edges merge into each other due to dilation, and hence they do not show up in the final result after drawing the contours.

Contour hierarchy

In some cases, some shapes are inside other shapes. Just like nested figures. In this case, we call outer one as parent and inner one as child. This way, contours in an image have some relationship to each other. And we can specify how one contour is connected to each other, like, is it child of some other contour, or is it a parent etc. Representation of this relationship is called the Hierarchy.

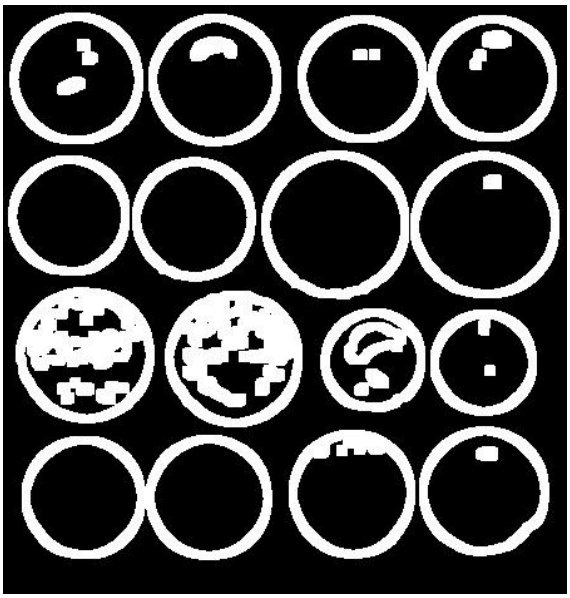


Figure 3.16 (a) Dilated image

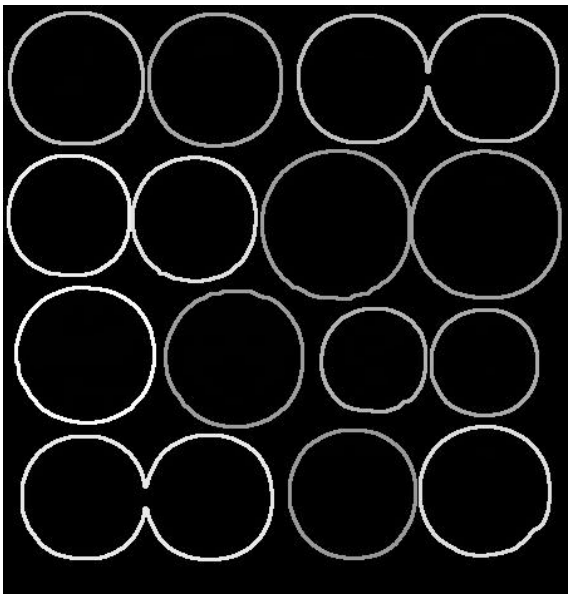


Figure 3.16 (b) Detected contours

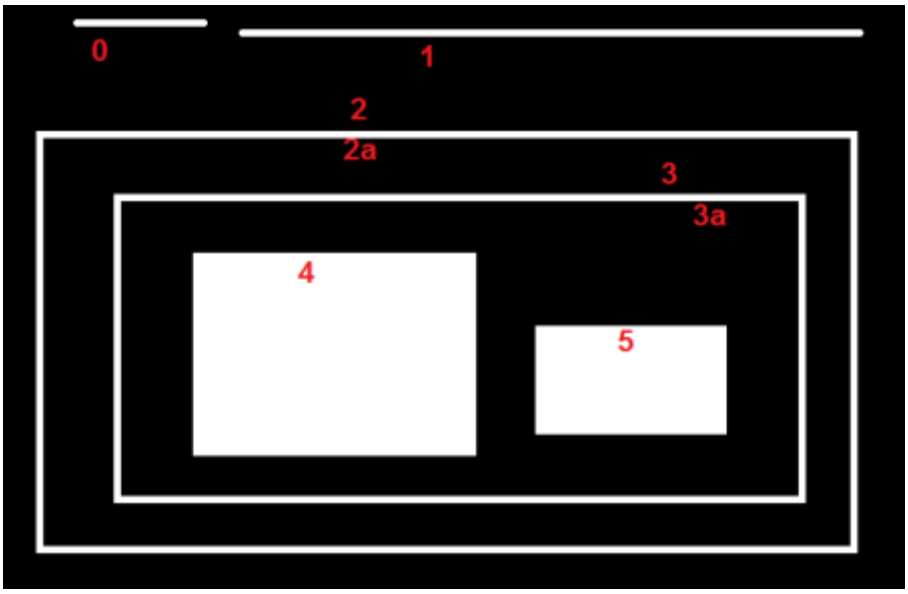


Figure 3.17 Contour hierarchy

In figure 3.17, the contours which are not enclosed by any other contours are the parent contours or the external contours. 0, 1 and 2 are examples of parent contours. They are in the same level of the hierarchy.

Contour-2a can be considered as a child of contour-2 (or in opposite way, contour-2 is parent of contour-2a). So let it be in hierarchy-1. Similarly contour-3 is child of contour-2 and it comes in next hierarchy. Finally contours 4 and 5 are the children of contour-3a, and they come in the last hierarchy level.

3.5.6 Area thresholding

For Video surveillance, it is important to detect only those objects which are of interest from the point of view of security. The background used for surveillance might contain small objects which are in motion like the movement of a curtain or the movement of plants and leaves due to wind. This kind of motion needs to be ignored by our algorithm in order to track the motion of a foreign object in the observation space.

This is done by applying an area threshold over the contours after they are detected. Only the contours whose area is greater than the set threshold will be interpreted as foreign objects by the system. All other contours will be ignored.

The value of this area threshold also depends on our observation space and hence there should be a provision to set this threshold depending on the surroundings where we want the surveillance to operate. Hence, a track bar is provided, using which one can easily change the value of this area threshold.

3.5.7 Display of output

The objects which show up after the area thresholding are treated as moving objects in our observation area and they are shown in the output. They are displayed as enclosed in a bounded rectangle and are also stored in the database and can be accessed by the user.

4. Software

4.1 Raspberry Pi Android App Communication

4.1.1 LAMP Server on Raspberry Pi:-

- **LAMP** Server stands for a **Linux, Apache, MySQL and PHP** server.

- The **Linux** version we are using is **Raspbian**. Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi.
  It's called Raspbian because it's a portmanteau of Raspberry and Debian; much in the same way as RaspBMC or Pidora[28].

  Debian is a line of quite popular Linux distributions that is the base of many famous distros; Ubuntu and its offshoots, Tails, SteamOS, wattOS and many more. It's very easy to use and contains a large repository of software, so it's easy to extend and customise it exactly how you want it.

  The different versions of Linux are called distributions or distros. They take the Linux kernel and its other technologies and package it with their own software, software repositories, branding and perhaps even a desktop environment. Debian is an example of one of these distros.

  What you will notice is its package management system, Aptitude, which is a major part of Debian and its various flavours and offshoots for example, commands like sudo apt-get. There's also stuff like the way boot time programs are loaded and some necessary tools for compiling software.

  The Raspberry Pi uses ARM v6, while Ubuntu specifically uses a newer version. There are also special tools that make better use of the Raspberry Pi hardware and are integrated into the major Raspberry Pi distros.

- **Apache HTTP** Server is a popular web server application you can install on the Raspberry Pi to allow it to serve web pages[29]. On its own, Apache can serve HTML files over HTTP, and with additional modules can serve dynamic web pages using scripting languages such as PHP.

  Instead of implementing a single architecture, Apache provides a variety of Multi Processing Modules(MPMs), which allow Apache to run in a process-based, hybrid (process and thread) or event-hybrid mode, to better match the demands of each particular infrastructure. This implies that the choice of correct MPM and the correct configuration is important.

  Where compromises in performance need to be made, the design of Apache is to reduce latency and increase throughput, relative to simply handling more requests, thus ensuring consistent and reliable processing of requests within reasonable time-frames.

  Apache features configurable error messages, DBMS-based authentication databases, and content negotiation. It is also supported by several graphical user interfaces (GUIs).It supports password authentication and digital certificate authentication. Because the source code is freely available, anyone can adapt the server for specific needs, and there is a large public library of Apache add-ons.

- **PHP** is a pre-processor; it's code that runs when the server receives a request for a web page. It runs, works out what needs to be shown on the page, then sends that page to the browser. Unlike static HTML, PHP can show different content under different circumstances. PHP is a very popular language on the web; large projects like Facebook and Wikipedia are written in PHP.

  PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks[30]. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any

type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

- **MySQL** (pronounced My Sequel or My S-Q-L) is a popular database engine. Essentially, where computers are concerned, a database is just a collection of data. Specialised database software, like MySQL, are just programs that lets you store and retrieve that data as efficiently as possible[31].

  Many computer programs, including web-based programs like blogs, photo galleries and content management systems need to store and retrieve data. For example, blog software need to store the posts (ie, articles) you write, and retrieve them when a visitor goes to your site. Similarly, photo galleries store information about their pictures (for example, for sites that allow users to rate the photos, the numerical rating for each picture is stored in a database). Instead of reinventing the wheel and implementing their own system of storing and retrieving data, these software simply use the specialised database programs I mentioned earlier.

  To make it easy for other programs to access data through them, many database software support a computer language called "SQL" (often pronounced as "sequel"). SQL was specially designed for such a purpose. Programs that want the database software to handle the low-level work of managing data simply use that language to send it instructions. There are many databases that support the use of SQL to access their data, among them MySQL. In other words, MySQL is just the brand of one database software, one of many.

4.1.2 JSON Data Format

- **JSON** is an open-standard format that uses human-readable text to transmit data objects consisting of attribute–value pairs**.** JSON stands for **J**ava**S**cript **O**bject **N**otation[32]. JSON is a lightweight data-interchange format. JSON is language independent. JSON is "self-describing" and easy to understand. The JSON filename extension is .json.

JSON example:

```
{"employees":[
        {"firstName":"John", "lastName":"Doe"},
        {"firstName":"Anna", "lastName":"Smith"},
        {"firstName":"Peter", "lastName":"Jones"}
]}
```

- JSON's basic data types are:
  1. Number: a signed decimal number that may contain a fractional part and may use exponential E notation, but cannot include non-numbers like NaN. The format makes no distinction between integer and floating-point. JavaScript uses a double-precision floating-point format for all its numeric values, but other languages implementing JSON may encode numbers differently.
  2. String: a sequence of zero or more Unicode characters. Strings are delimited with double-quotation marks and support a backslash escaping syntax.
  3. Boolean: either of the values true or false
  4. Array: an ordered list of zero or more values, each of which may be of any type. Arrays use square bracket notation with elements being comma-separated.

5. Object: an unordered collection of name/value pairs where the names (also called keys) are strings. Since objects are intended to represent associative arrays, it is recommended, though not required, that each key is unique within an object. Objects are delimited with curly brackets and use commas to separate each pair, while within each pair the colon ':' character separates the key or name from its value.
6. Whitespace is allowed and ignored around or between syntactic elements (values and punctuation, but not within a string value). Four specific characters are considered whitespace for this purpose: space, horizontal tab, line feed, and carriage return. JSON does not provide any syntax for comments.

4.2. Android Application

4.2.1. Overview:

- The front-end of an application (app) has a lot of interconnected components. The JavaScript for the application falls into three main parts: libraries (such as jQuery, D3 etc.), the main application core, and widgets (each widget in the application is modularized and has its own code).
- Phonegap/Corona/Java can be used to develop frontend of android/i-phone app.
- For *native mobile application* front-ends, most popular would probably be:
  1. Objective C (for iOS apps)
  2. Java (Android apps)
- The back end of an application consists of a server, an application, and a database. A back-end is the one which builds and maintains the technology that powers those components which, together, enable the user-facing side of the app to even exist in the first place.
- In order to make the server, application, and database communicate with each other, we can use server-side languages like PHP, Ruby, Python, Java, and .Net to build an

application, and tools like MySQL, Oracle, and SQL Server to find, save, or change data and serve it back to the user in front-end code.

- A local host is setup to test the working using free servers available online (Apache-wsgi, etc.).
- The database is made using MySQL/SQLite/Oracle which makes the data organized in the server in a particular format.
- After the database is made, the communication of it with the front-end is implemented using languages mentioned above in their respective Frameworks.
- In case of an app, each platform that runs Java needs a virtual machine (VM) implementation. On Android the original VM is called Dalvik. The job of these virtual machines is to interpret the bytecode, which is really just a set of instructions similar to the machine code found in CPUs, and execute the program on the processor.
- The VMs use a variety of technologies including just-in-time compilation (JIT) and ahead-of-time compilation (AOT) to speed up the processes. This in turn is executed on the VM built-in to Android. This is different to the model used by iOS which uses a native compiler to turn Objective-C into ARM machine code.
- The next step is to learn how to use Java to create Android apps. For that you will need the Android Software Development Kit (SDK). The Android SDK provides all the API libraries and tools you need build an Android app. Google offers the Android Developer Tools (ADT) bundle which includes the SDK, a version of the Eclipse IDE with the ADT plugins, and the Android Platform-tools with the Android emulator.
- Now, we are ready to buy a domain and dump our code on it. The website is ready to use for testing and further purposes.
- Android does not review newly submitted apps right away. They'll pass by at some point and check it out but you are able to instantly add your app to Google Play.
- iOS, once again, is different here. Apple reserves the right to review and approve your app before it can go live. There is no set timeframe for this, but you can expect at least a week before you hear back from them.

4.3. App: Frontend

4.3.1. Operating System: Android

Android is a mobile operating system (OS) currently developed by **Google**, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input.

In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics.

Android has the largest installed base of all operating systems of any kind. Android has been the bestselling OS on tablets since 2013, and on smartphones it is dominant by any metric. As of July 2013, the Google Play store has had over one million Android applications ("apps") published, and over 50 billion applications downloaded[33].

Apps, which extend the functionality of devices, are written using the Android software development kit (SDK) and, often, the Java programming language that has complete access to the Android APIs.

4.3.2. IDE: Android Studio

- **IDE** stands for **I**ntegrated **D**evelopment **E**nvironment.
- To develop the Android App we have used **Android Studio**.
- **Android Studio** is the official integrated development environment (IDE) for Android platform development.

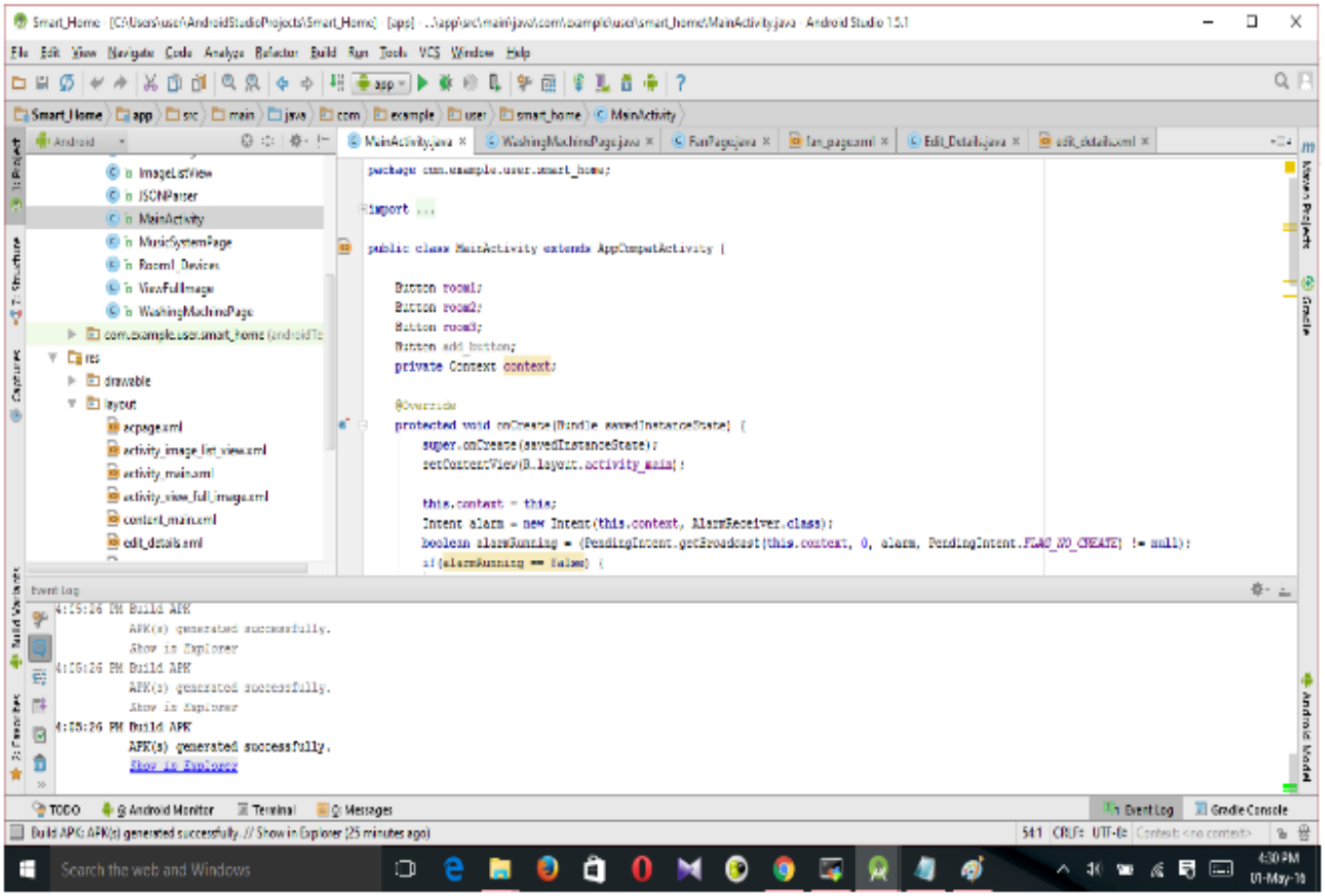


Figure 4.1 Android Studio

- A rich layout editor that allows users to drag-and-drop UI components, option to preview layouts on multiple screen configurations.
- Support for building Android Wear apps.
- Built-in support for Google Cloud Platform, enabling integration with Google Cloud Messaging and App Engine.
- Its prime competitor is the Eclipse Android Development Tools.

4.3.3. Programming Languages:

4.3.3.1. XML:

Extensible Markup Language (**XML**) is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures such as those used in web services.

XML has come into common use for the interchange of data over the Internet. XML has been used to design the layout of the app. It includes all the buttons, text fields etc.

4.3.3.2. Java:

Java is a set of computer software and specifications developed by Sun Microsystems, which was later acquired by the Oracle Corporation, that provides a system for developing application software and deploying it in a cross-platform computing environment. Java is used in a wide variety of computing platforms from embedded devices and mobile phones to enterprise servers and supercomputers. The heart of the Java platform is the concept of a "virtual machine" that executes Java bytecode programs. This bytecode is the same no matter what hardware or operating system the program is running under[34].

The JIT compiler translates the Java bytecode into native processor instructions at run-time and caches the native code in memory during execution. In most modern operating systems (OSs), a large body of reusable code is provided to simplify the programmer's job. This code is typically provided as a set of dynamically loadable libraries that applications can call at runtime. Because the Java platform is not dependent on any specific operating system, applications cannot rely on any of the pre-existing OS libraries.

Instead, the Java platform provides a comprehensive set of its own standard class libraries containing many of the same reusable functions commonly found in modern operating systems. Java has been used to provide functionality to the layout that has been

designed using XML. Using Java we have made the layout reactive to gestures such as tapping, swiping and clicking.

4.4. App: Backend

4.4.1. Database: As described in the previous sections about the MySQL, the basic overview is seen. As it is an open source software and has many implications, we have chosen MySQL as our tool to handle and manage our database. Now, we will discuss about the basic tools of MySQL used in our project as follows:

4.4.1.1. Setting up a new MySQL User Account:

For example, to add a new user to MySQL, you just need to add a new entry to user table in database MySQL.

Below is an example of adding new user guest with SELECT, INSERT and UPDATE privileges with the password guest123; the SQL query is:

```
root@host# mysql -u root -p
Enter password:*******
mysql> use mysql;
Database changed

mysql> INSERT INTO user
          (host, user, password,
           select_priv, insert_priv, update_priv)
           VALUES ('localhost', 'guest',
           PASSWORD('guest123'), 'Y', 'Y', 'Y');
Query OK, 1 row affected (0.20 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 1 row affected (0.01 sec)

mysql> SELECT host, user, password FROM user WHERE user = 'guest';
+-----------+---------+------------------+
| host      | user    | password         |
+-----------+---------+------------------+
| localhost | guest | 6f8c114b58f2ce9e |
+-----------+---------+------------------+
1 row in set (0.00 sec)
```

4.4.1.2. MySQL Basic Queries:

- To insert data into MySQL table, you would need to use **SQL INSERT INTO** command.

Syntax:

INSERT INTO *table_name* **VALUES** (*value1*,*value2*,*value3*,...);	**INSERT INTO** *table_name* (*column1*,*column2*,*column3*,...) **VALUES** (*value1*,*value2*,*value3*,...);

The first form does not specify the column names where the data will be inserted, only their values.

The second form specifies both the column names and the values to be inserted.

- The **SQL SELECT** command is used to fetch data from MySQL database.

Syntax:

```
SELECT column_name(s)
FROM table_name
LIMIT number;
```

- There may be a requirement where existing data in a MySQL table needs to be modified. You can do so by using **SQL UPDATE** command. This will modify any field value of any MySQL table.

  Syntax:

  UPDATE *table_name*

  SET *column1=value1*,*column2=value2*,...

  WHERE *some_column=some_value*;

.

- If you want to delete a record from any MySQL table, then you can use SQL command **DELETE FROM**.

  Syntax:

  DELETE FROM *table_name*

  WHERE *some_column=some_value*;

4.4.1.3. Some important MySQL commands

- **USE *Databasename*** : This will be used to select a particular database in MySQL work area.
- **SHOW DATABASES:** Lists the databases that are accessible by the MySQL DBMS.
- **SHOW TABLES**: Shows the tables in the database once a database has been selected with the use command.
- **SHOW COLUMNS FROM *tablename*:** Shows the attributes, types of attributes, key information, whether NULL is permitted, defaults, and other information for a table.
- **SHOW INDEX FROM *tablename***: Presents the details of all indexes on the table, including the PRIMARY KEY.
- **SHOW TABLE STATUS LIKE *tablename*\G**: Reports details of the MySQL DBMS performance and statistics.

4.4.2. Server Setup:

The database is made in MySQL on the server of Raspberry pi, is accessed using PHP scripts.

```
<?php
   /*
    All database connection variables
  */
define('DB_USER', "root"); // db user
define('DB_PASSWORD', "pass123"); // db password (mention your db password here)
define('DB_DATABASE', "Smart_Home"); // database name
define('DB_SERVER', "localhost"); // db server
?>
```

We have used the file db_config.php to declare the variables related to the setup of the connection with the server.

With the help of this PHP script we have declared the variables DB_USER, DB_PASSWORD, DB_DATABASE and DB_SERVER with the values ‘root’, ‘pass123’, ‘Smart_Home’ and ‘localhost’ respectively. These variables have been declared here so as to use in the next mentioned PHP script which sets up the connection between App and the database.

4.4.2.1. Connecting to Server:

PHP has been used to connect to the server to setup a link through the WiFi connection to facilitate upload and download of data between the server and Android App.This script, db_connect.php invokes the above mentioned db_config script. The database connection variables are imported from this file and are used to verify the details entered by the user. Hence it helps to connect to the server and also select the database at the same time. This database is hosted on the Raspberry Pi server and can be accessed using phpMyAdmin.

```
<?php
    /* A class file to connect to database */
    class DB_CONNECT {
        // constructor
      function __construct() {
        // connecting to database
        $this->connect();
      }
        // destructor
      function __destruct() {
        // closing db connection
        $this->close();
      }
      /* Function to connect with database     */
      function connect() {
        // import database connection variables
        require_once __DIR__ . '/db_config.php';
        // Connecting to mysql database
        $con = mysql_connect(DB_SERVER, DB_USER, DB_PASSWORD) or
    die(mysql_error());
        // Selecting database
        $db = mysql_select_db(DB_DATABASE) or die(mysql_error()) or
    die(mysql_error());
        // returning connection cursor
        return $con;
      }
        /* Function to close db connection     */
      function close() {
        // closing db connection
        mysql_close();
      }
    }
    ?>
```

4.4.2.2. GET data from database:

The user needs receive continuous knowledge about the status of the various smart appliances in his home and also provide him with functionality to control the devices. This is done by continuously accessing the database and obtaining its details. This is again done through PHP scripting. The module get_all_products.php enables with this task.

```
// get all products from products table
$result = mysql_query("SELECT *FROM Device_AC") or die(mysql_error());
// check for empty result
if (mysql_num_rows($result) > 0) {
  // looping through all results
  // products node
  $response["products"] = array();
   while ($row = mysql_fetch_array($result)) {
     // temp user array
     $product = array();
     $product["pid"] = $row["pid"];
     $product["Status"] = $row["Status"];
     $product["Temperature"] = $row["Temperature"];
    // push single product into final response array
     array_push($response["products"], $product);
  }
  // success
  $response["success"] = 1;
  // echoing JSON response
  echo json_encode($response);
```

The entire program has been built in PHP but MySQL queries have been incorporated in it. Using MySQL all the details in the table are selected and are stored in an array. A loop is run to obtain all the details in the multiple columns of each row. When all rows have been scanned and saved the loop ends. This array is then encoded using JSON and the user receives the data as an JSON object in the app which has been developed to handle JSON objects and JSON exceptions.

4.4.2.3. POST data to database:

The Android application allows the user to change the status of the device and also control its characteristic parameters. The changes that are made must be communicated to the Raspberry Pi by updating the database which will relay these changes as instructions to the devices.

The update_product.php file allows the app to change the database in real time. The fields that have been changed are stored as variables which are further encoded as JSON objects. It "POST's" the data stored in these objects in the respective columns and then the MySQL query "UPDATE" for updating the database is used. We have also incorporated pop-up messages in the App which display messages with respect to if the updating was successful or not as the case might be.

```
// check for required fields
if (isset($_POST['pid']) && isset($_POST['Temperature']) && isset($_POST['Status'])) {
        $pid = $_POST['pid'];
       $Temperature = $_POST['Temperature'];
      $Status = $_POST['Status'];
        // mysql update row with matched pid
      $result = mysql_query("UPDATE Device_AC SET Temperature =
   '$Temperature',Status='$Status' WHERE pid = $pid");
       // check if row inserted or not
      if ($result) {
         // successfully updated
         $response["success"] = 1;
         $response["message"] = "Product successfully updated.";
          // echoing JSON response
         echo json_encode($response);   } }
```

4.4.3. phpMyAdmin:

phpMyAdmin is a free and open source tool written in PHP intended to handle the administration of MySQL or MariaDB with the use of a web browser. It can perform various tasks such as creating, modifying or deleting databases, tables, fields or rows; executing SQL statements; or managing users and permissions. It has been used as a link between the frontend Android App and the backend server that has been established on the Raspberry Pi.

4.5. App: Layout and Working

An Android App has been developed to help the user with comfort to remotely control all the devices and appliances in his home. This app is designed with multiple “activities” each of which have a designated function and objective.

4.5.1. Main Activity/ Choose Room:

Figure 4.2 displays the home page of the app. When the app is launched from the menu this is the window that one can see. It allows the user to select the room in which he wants to control a device. This allows him to sit in the living room and easily control the devices in the kitchen or his bedroom.

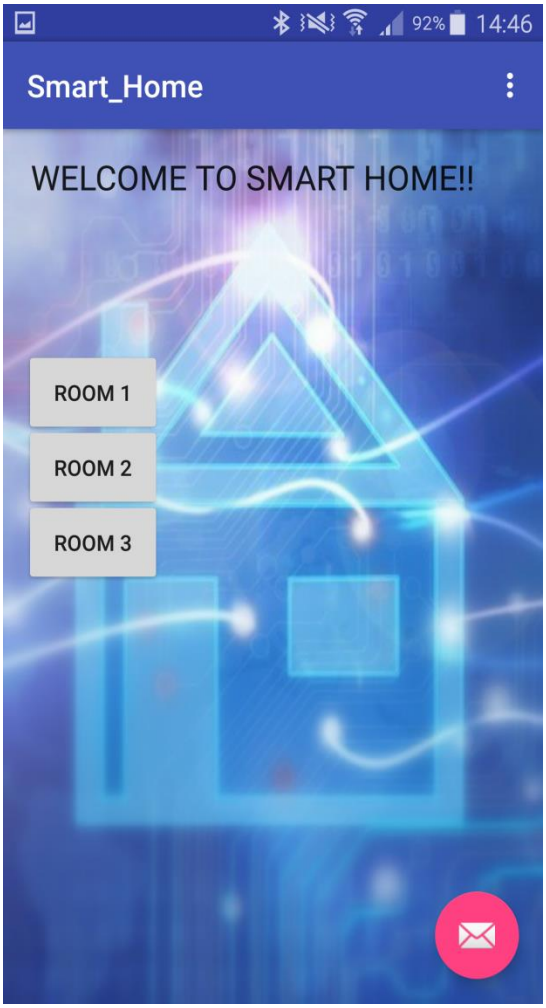


Figure 4.2 Home page

4.5.2. **Choose Device:**

This page has all the devices of the selected room listed in the form of buttons (refer fig 4.3). If one wishes to control the air-conditioner or washing machine he must click on the AC button or the washing machine button respectively. On doing this the app will redirect him to the specific page of that particular device. For quick access the user can also note the humidity and temperature of the particular room and can also make changes from there itself.

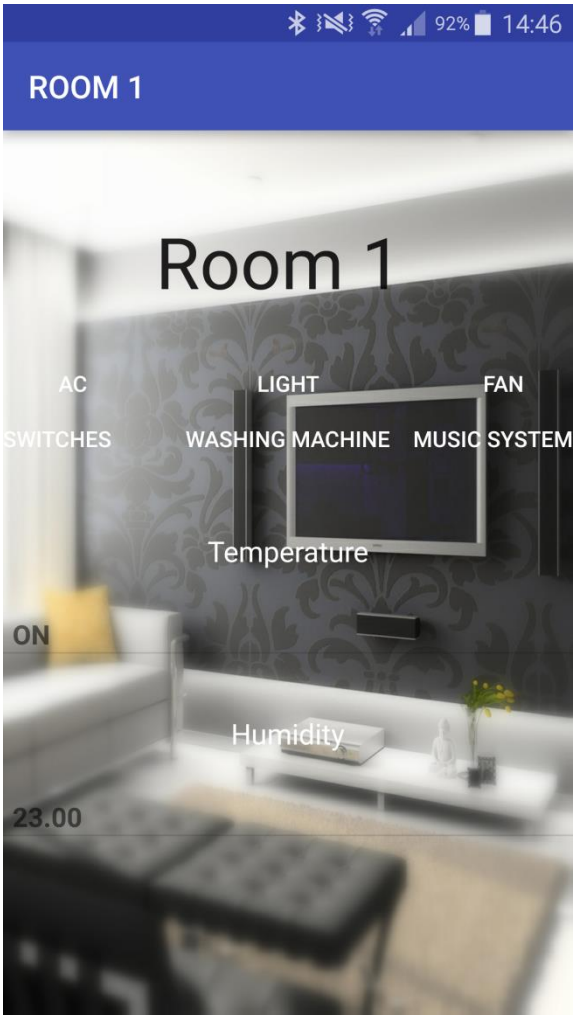


Figure 4.3 Room 1(select device from here)

4.5.3. Device:

This is the most important activity of the app. From this window the user can carry out a number of functions such as changing the temperature of the **AC**, switching it on/off, draining out the **washing machine** or filling it with water or switching on/off the lights in the room. Each device has a different page that has been designed according to its use. For example, the **washing machine** page (refer fig 4.4), provides the user to not only switch the machine on but also it allows him to fill the washing machine with water and also drain it once the cycle is over. The user also gets to control the inflow speed of the water. He has liberty to also select the type of cycle he wants to run i.e. washing or drying.

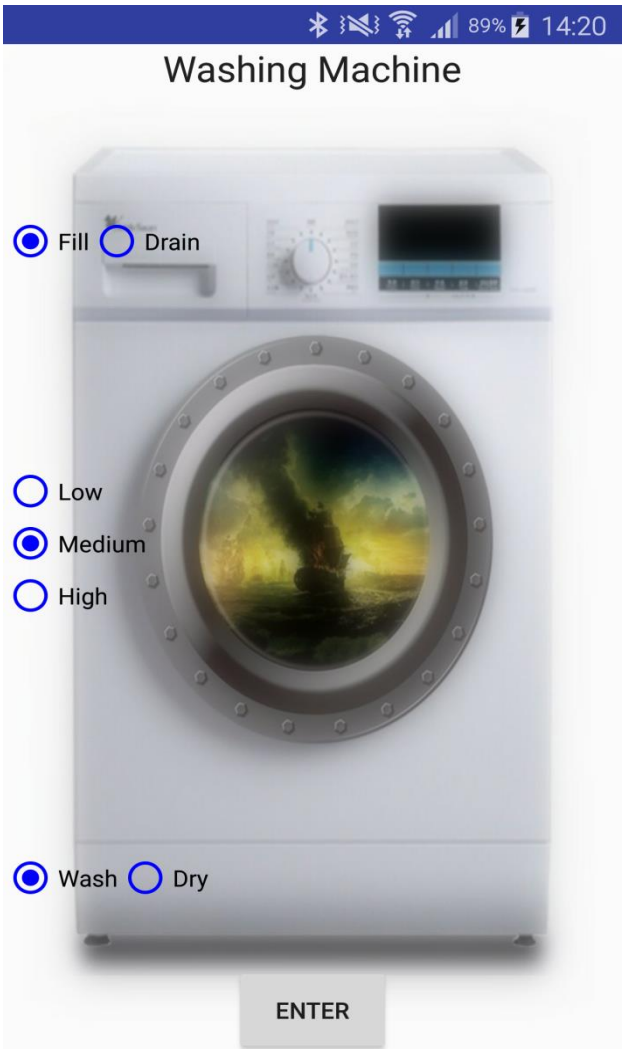


Figure 4.4 Washing Machine Control Page

One can also control the **music system** via the Android App (refer fig 4.5). Apart from switching it on and off, one can change to volume. The user can change the song he is listening to, play/pause a song all of this directly through his Android smartphone.

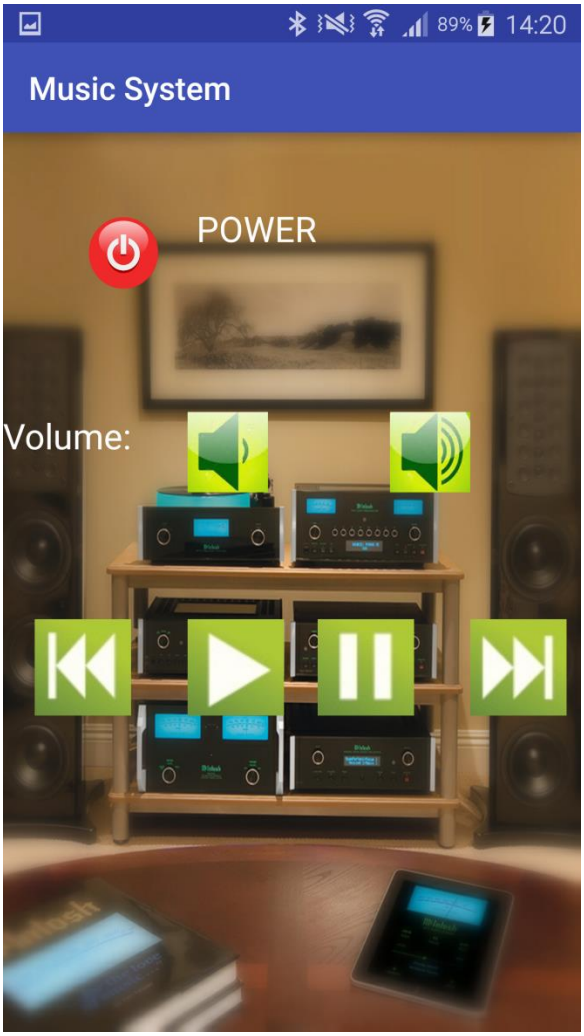


Figure 4.5 Music System Control Page

This page allows the user to control the **ceiling fan** in the room (refer fig 4.6). Apart from switching it on and off he can even regulate its speed with the help of the slider shaped regulator that we have designed. As and when he makes changes he will see a pop-up message (as shown in fig 4.6) which will notify him of the fan speed at that point of time.

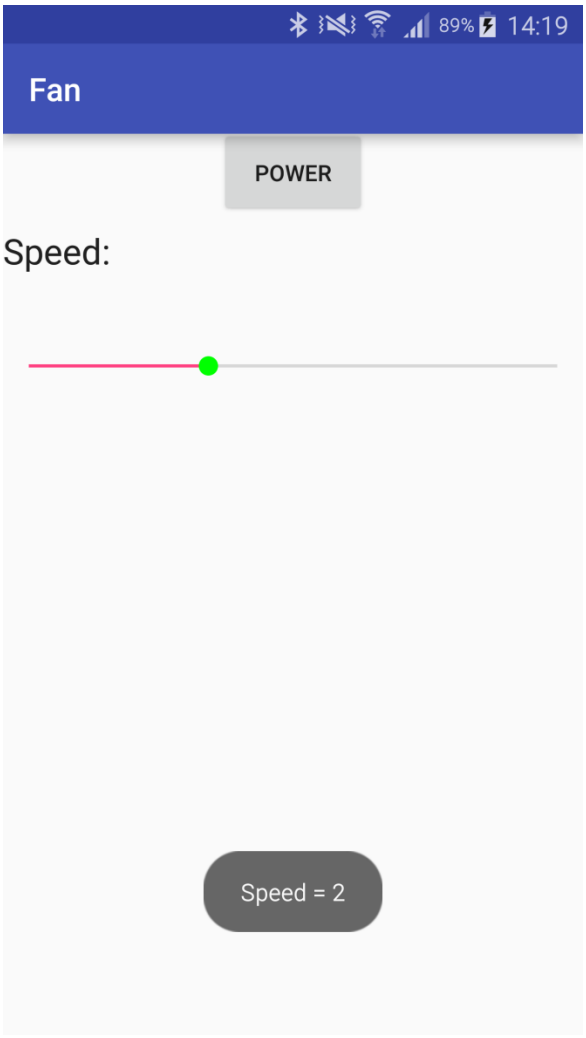


Figure 4.6 Fan Control Page

Also we have provided functionality to control the air-conditioner in the future. The **AC** page has options to change the temperature. The AC page offers the option to make changes to the working parameters of the AC (refer fig 4.7).

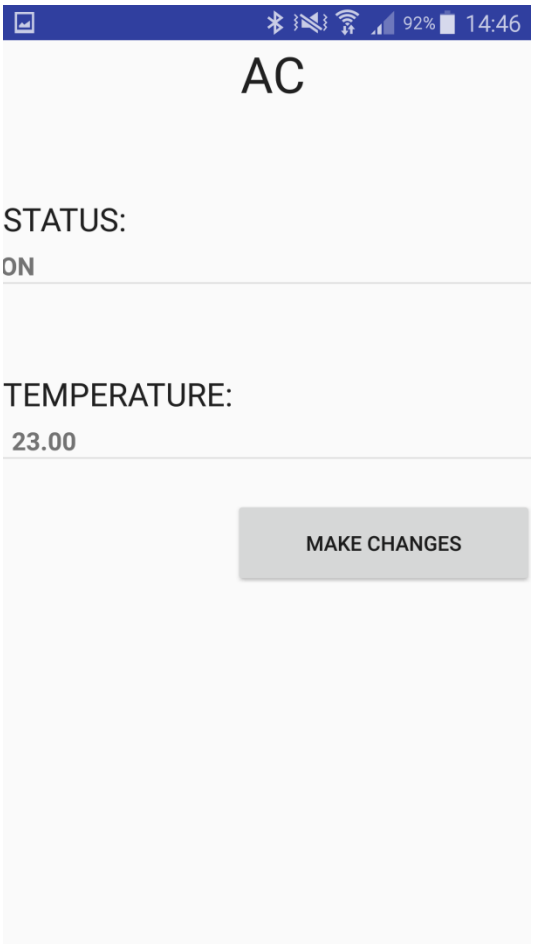


Figure 4.7 Air-Conditioner Control Page

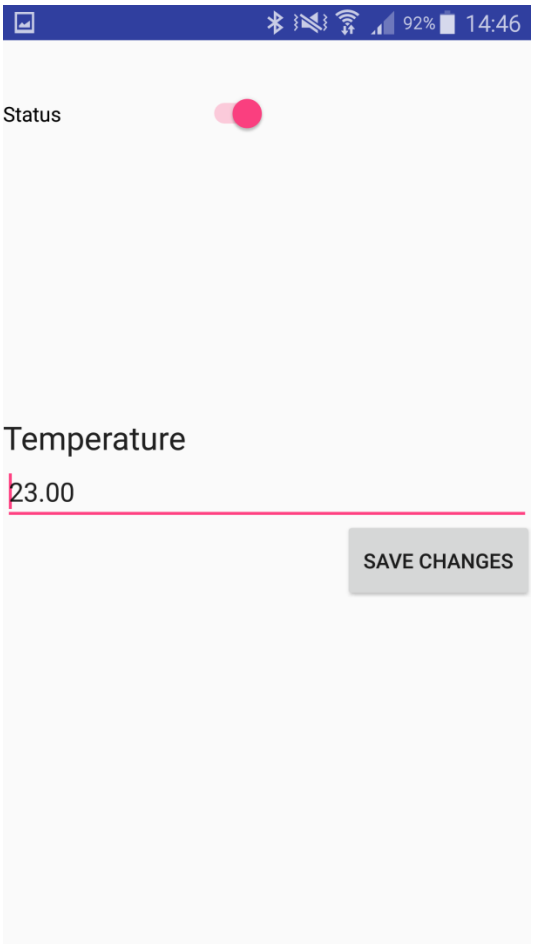


Figure 4.8 Make Changes to AC parameters

On clicking it the user can input the temperature he wants to set the air-conditioner to and even switch it on and off with the help of the status switch (fig 4.8). If the room temperature exceeds a certain value as decided by the user, he will receive a notification regarding this. By clicking this notification he will be redirected to the page to change the working parameters of the air-conditioner.

4.5.4. Surveillance Notifications:

When the user leaves his house, the surveillance system will get activated. If it notices any unwarranted movement in the house. The system will send an image of the movement it has recorded to the app. The user will receive an alert in the notification panel (fig 4.9).

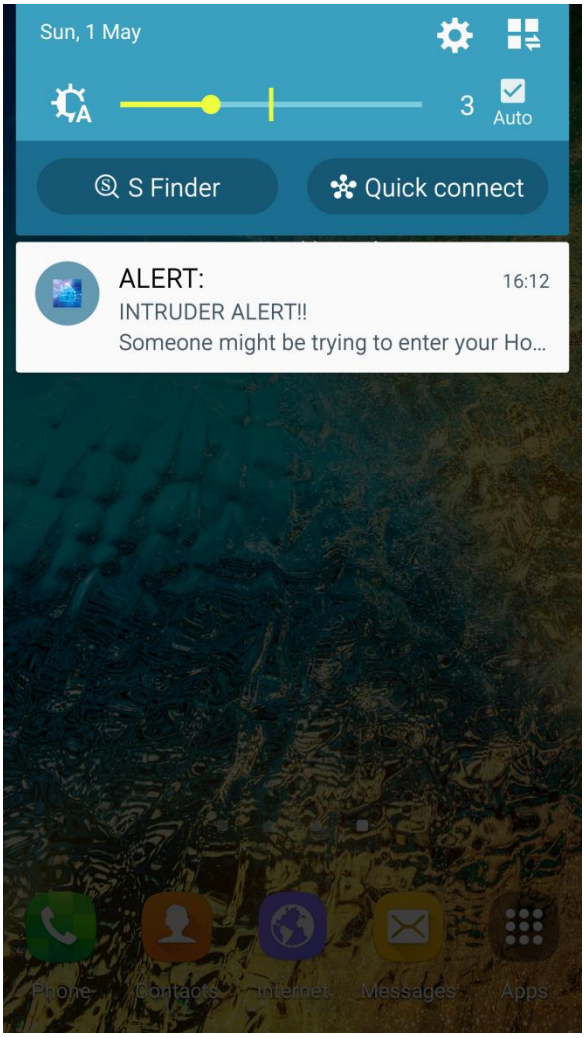


Figure 4.9 Surveillance System Notification

When he clicks this notification he will be redirected to a page (fig 4.10) which will have the links of all the new images the surveillance system has captured and sent. When one clicks on the links he will be able to see the image (fig 4.11) captured by the surveillance system.

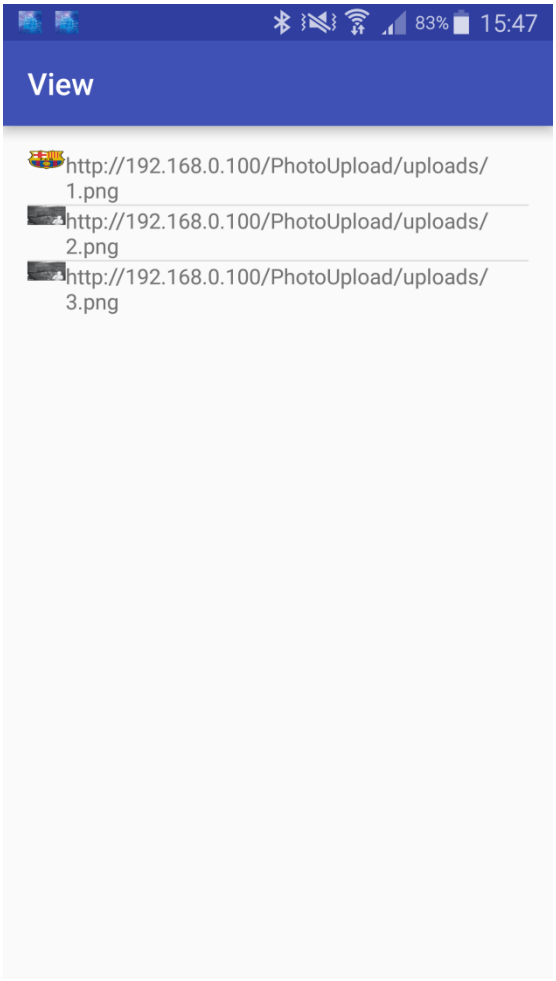


Figure 4.10 Images captured and sent by the surveillance system

Figure 4.11 Intruder Image

5. Conclusion

- Using different wireless communication technologies for different applications depending on the metrics and constraints like power consumption and data rate, the home automation system was designed.
- Power consumption was lowered and a completely wireless system was designed. The cost of development is further minimized by mass production and mass purchase of components as well as designing of dedicated embedded processing unit and operating system.
- The android application interface is intuitive and has a simplistic design with customization options. The setup of hardware is easy and hence low installation costs.
- With the help of machine learning and more sophisticated algorithms the system will become more robust and autonomous giving an actual automated home.
- Surveillance system and presence detection monitor work in tandem to provide security and intruder alarm. This feature was tested for response time and image quality in detection.
- The overall result is that there is large scope for further development as specified in the concluding section which can be implemented with time.

6. Further Scope:

The extent of involvement of the Internet and technology in our daily lives has increase exponentially in recent times. The advent of smartphones and tablets is just the beginning. Smart cars, Smart homes and classrooms are being developed by researchers all over the world. In the future it is safe to estimate that the smart homes will be highly advanced. The degree of automation will increase resulting in lesser and lesser human intervention. This will provide humans the comfort of monitoring and controlling complex appliances such as a robot butler or something as simple as the window blinds, that too while being miles away.

Soon the home automation system will be equipped with learning capabilities. This would enable the system to understand the usage patterns of the residents of the home on basis of which it would take informed decisions and perform actions without any human intervention. The biggest hurdle researchers face is ensuring the security of the entire setup not only for the software but also hardware. Since the entire system is internet based it would be very easy for someone to gain control of the system by hacking into the network. This must be countered by effective mechanism such as encrypting the data and ensuring that the data transfer lines have not been compromised. On the hardware front the whole system is based on a network of microcontrollers and raspberry pi central units. These pieces of hardware have ports to connect them to various computers or even insert memory cards with the help of which one can easily assume control of the system. Apart from this, using sophisticated softwares such as JTAG one can easily retrieve codes from IC's and thus put the whole system at risk. If the system is compromised the culprit will have unlimited access to the activities inside the home which puts the residents in grave danger.

Thus the Smart Home system has a lot of scope not only for advancing the facilities it can provide the end user but also protecting his privacy at the same time.

7. References

[1]http://www.xfinity.com/resources/home-automation.html

[2]https://www.elprocus.com/home-automation-systems-applications/

[3]https://en.wikipedia.org/wiki/Internet_of_Things

[4].http://www.webopedia.com/TERM/I/internet_of_things.html

[5]http://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT

[6]http://homesecurity.honeywell.com/home_automation.html#lighting

[7]https://www.savant.com/products

[8]ElShafee, A. and Hamed, K.A., 2012. Design and implementation of a WIFI based home automation system. *World academy of science, engineering and technology*, *68*, pp.2177-2180.

[9]Arul, S. Benjamin. "Wireless Home Automation System Using Zigbee."*International Journal of Scientific & Engineering Research* 5, no. 12 (2014): 133.

[10]Jadhav, Pankaj, Amit Chaudhari, and Swapnil Vavale. "Home Automation using ZigBee Protocol." *International Journal of Computer Science and Information Technologies* 5, no. 2 (2014): 1778-1780.

[11]Ramlee, R.A., LEONG, M.H., SARBAN SINGH, R.S.A., ISMAIL, M.M., OTHMAN, M.A., SULAIMAN, H.A., MISRAN, M.H., SAID, M. and ALICE, M., 2013. Bluetooth remote home automation system using android application.

[12]Sriskanthan, N., Tan, F. and Karande, A., 2002. Bluetooth based home automation system. *Microprocessors and Microsystems*, *26*(6), pp.281-289.

[13]Palaniappan, Satish, Naveen Hariharan, Naren T. Kesh, and S. Vidhyalakshimi. "Home Automation Systems-A Study." *International Journal of Computer Applications* 116, no. 11 (2015).

[14]Jerfia, G.N., Vaishnavi, J. and Karthick, P., CENTRALIZED SMART HOME AUTOMATION THROUGH VOICE RECOGNITION.

[15]Bergstrom, Peter, Kevin Driscoll, and John Kimball. "Making home automation communications secure." *Computer* 34, no. 10 (2001): 50-56.

[16]http://standards.ieee.org/about/get/802/802.11.html

[17]https://www.bluetooth.com

[18]http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01P

[19]https://www.raspberrypi.org/products/raspberry-pi-2-model-b/

[20]**https://en.wikipedia.org/wiki/Surveillance**

[21]Rothkrantz, L. J. M. "Smart Surveillance Systems." *Network Topology in Command and Control: Organization, Operation, and Evolution: Organization, Operation, and Evolution* (2014): 270.

[22]Lee, L., R. Romano, and G. Stein. "Introduction to the special section on video surveillance." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8 (2000): 740-745.

[23]http://opensource.org

[24]Piccardi, Massimo. "Background subtraction techniques: a review." In*Systems, man and cybernetics, 2004 IEEE international conference on*, vol. 4, pp. 3099-3104. IEEE, 2004.

[25]Keysers, D., Deselaers, T. and Ney, H., 2004. Pixel-to-pixel matching for image recognition using hungarian graph matching. In *Pattern Recognition*(pp. 154-162). Springer Berlin Heidelberg.

[26]Farooque, M.A. and Rohankar, J.S., 2013. Survey on various noises and techniques for denoising the color image. *International Journal of Application or Innovation in Engineering & Management (IJAIEM), 2*(11), pp.217-221.

[27]Mr. Rohit Verma 2013 A Comparative Study of Various Types of Image Noise and Efficient Noise Removal Techniques, *International Journal of Advanced Research in Computer Science and Software Engineering*

[28]http://linuxuser.co..uk/news/raspbian-explained

[29]https://en/wikipedia.org/wiki?Apache_HTTP_Server

[30]http://php.net/manual/en/faq.general.php

[31]http://www.w3schools.com/sql/

[32]http://www.w3schools.com/json

[33]http://www.webopedia.com/TERM/A/android_app.html

[34]https://en.wikipedia.org/wiki/Java_(programming_language)