Home automation using DTMF

(Dual Tone Multi Frequency) Module.

Submitted in partial fulfilment for the completion of Internship on Embedded System at ECIL(Hyderabad).

BY:
Jagadam Shravan(CBIT)
Abhishek Gupta(MGIT)

CHAPTER 1
OVERVIEW

1.1 Introduction:

Conventionally, electrical appliances in a home were controlled via switches that regulate the electricity to these devices. Today, we have entered the era of technology. Gone are the days were manual operation were performed. Home automation is becoming more popular around the world and is becoming a common practice. Smart home automation becomes important, because it provides the user the comfort and easy access to the home appliances. The process of home automation works by making everything in the house automatically controlled, using technology to control and do the jobs that we would normally do manually. Home automation takes care of a lot of different activities in the house.

In this project, we propose a unique System for Home automation utilizing Dual Tone Multi Frequency (DTMF) that is paired with a wireless module to provide seamless wireless control over many devices in a house. We can operate our system from any distant or remote area. It is a wireless system but instead of using a separate wireless module (transmitter and receiver) we are using the cell phones for this purpose. Cell-phone operated system is having a wide range (service provider range), less fear of interference as every call is having a unique frequency and moreover it has more control keys.

The principle used for cell-phone controlled system is the decoding of DTMF tone, .DTMF stands for Dual Tone Multi Frequency. The main components used here are the DTMF Decoder, Arduino UNO and the relay module. With the inter-connection of these components and with use of a cell phone as a remote we were successful in controlling various home appliances. For this the system we have used lamps to demonstrate AC loads. The advantages of using this technology are many. One can control home appliances from anywhere in the world. It helps in reducing the wastage of electricity. The cost for this system is also less as compared to other technologies like GSM.

1.2 AIM OF THE PROJECT

The objective of this project is to implement a low cost, reliable and scalable home automation system that can be used to remotely switch on or off any household appliance, using a microcontroller to achieve hardware simplicity and voice dial from any phone to toggle the switch state.

1.3 METHODOLOGY

The DTMF is a signaling system for recognizing the keys or on the dialed number of a push button or simply we can say it as a DTMF keypad. It is a combination of two sine wave tones to constitute a key. The tones are called as a Rows & Column frequencies and they are represented on the DTMF keypad.

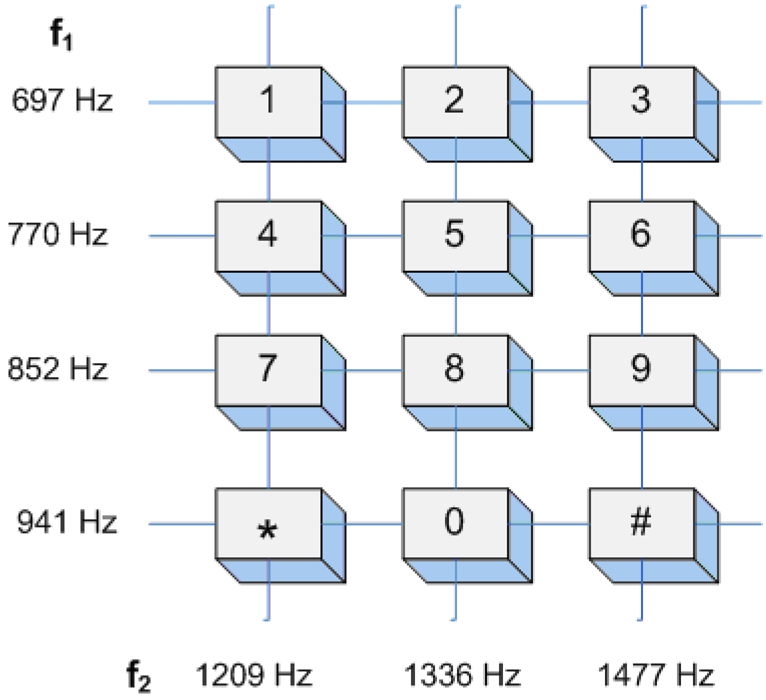

Figure 1.1

In the dual tone multi-frequency, if we press any key on your phone, then it generates two tones of specific frequencies, the first specific frequency is a high frequency of tone and the second one is a low frequency of the tone.

The following table shows the signals that you send when you press your touch tone phone keys.

NUMBER	FREQUENCIES
1	697 & 1209
2	697 & 1336
3	697 & 1477
4	770 & 1209
5	770 & 1336
6	770 & 1477
7	852 & 1209
8	852 & 1336
9	852 & 1477
*	941 & 1209
0	941 & 1336
#	941 & 1477

Table 1.1

1.4 SIGNIFICANCE

We know that in industries, loads are distributed over large distances, and it is not convenient to operate all those equipment time to time and it is also time to consume.

Also in agricultural fields, water pumps and other electric systems are spread over the large area and it is hard to operate these systems to farmers. We can use this circuit in the home too.

1.5 ORGANIZATION OF REPORT

In the first chapter we discuss introduction to the project, aim of the project, methodology involved in it and its significance.

In the next chapter we discuss the ideology, methodology involved in it and the solution for a problem.

In the third chapter we analyze the project, also discuss about the automation and home automation.

In the fourth chapter we discuss the components used in this project i.e. the microcontroller, DTMF decoder module and the relay.

In the fifth chapter we discuss the block diagram of the project, the project setup, algorithm for the code and its working.

In the last chapter we included the results and conclusion.

CHAPTER 2

IDEAOLOGY

2.1 INTRODUCTION

According to recent statistics, there is an increasing growth in the percentage of older persons in almost every part of the world today is recorded. The rate at which the proportion of older people increases in the population of developing countries is much faster than what is experienced in developed regions. Specifically, this trend is more predominant in developing economies with relatively much lower level of
Socio - economic development, particularly in India. In response to this, modern technological advances in semiconductor technology and wireless communication can be significantly exploited to drive the economic and social paradigm shifts associated with population ageing towards achieving the sustainable development goals (SDGS) in India. In this project, we designed and implemented a cost-effective smart assistive DTMF home automation system that utilizes a tele-remote circuit to control home appliances via existing cellular communication networks.

Dual tone multi frequency (DTMF) tones generated from keypads of mobile cell phones remotely control home devices and appliances. An integrated DTMF receiver decodes the tones and processes the information to control several devices using a relay switching system. The digitally controlled system overcomes the limited range of infrared and radio remote controls with the aid of available cellular communication systems. Therefore, older people of the populace can be provided with better ease of living at home by minimizing their movement and dependency at affordable cost.

2.2 METHODOLOGY

The DTMF is a signaling system for recognizing the keys or on the dialed number of a push button or simply we can say it as a DTMF keypad. It is a combination of two sine wave tones to constitute a key. The tones are called as a Rows & Column frequencies and they are represented on the DTMF keypad.

NUMBER	FREQUENCIES	OUTPUT SIGNAL
1	697 & 1209	sin(2*pi*697*t)+sin(2*pi*1209*t)
2	697 & 1336	sin(2*pi*697*t)+sin(2*pi*1336*t)
3	697 & 1477	sin(2*pi*697*t)+sin(2*pi*1477*t)
4	770 & 1209	sin(2*pi*770*t)+sin(2*pi*1209*t)
5	770 & 1336	sin(2*pi*770*t)+sin(2*pi*1336*t)
6	770 & 1477	sin(2*pi*770*t)+sin(2*pi*1477*t)
7	852 & 1209	sin(2*pi*852*t)+sin(2*pi*1209*t)
8	852 & 1336	sin(2*pi*852*t)+sin(2*pi*1336*t)
9	852 & 1477	sin(2*pi*852*t)+sin(2*pi*1477*t)
*	941 & 1209	sin(2*pi*941*t)+sin(2*pi*1209*t)
0	941 & 1336	sin(2*pi*941*t)+sin(2*pi*1336*t)
#	941 & 1477	sin(2*pi*941*t)+sin(2*pi*1477*t)

Table 2.1

When a key is pressed in the keypad, the output signal as mentioned in the above table is generated using switched networks in Tele - Communication as per our project is concerned. Here are some output signals that are generated when a certain key is pressed in the keypad.

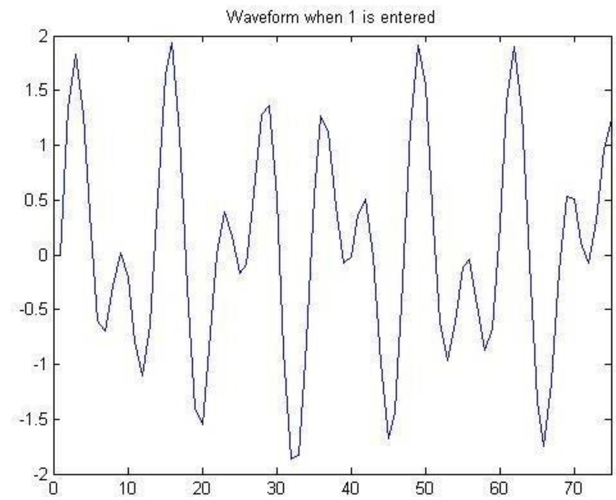

Figure 2.1

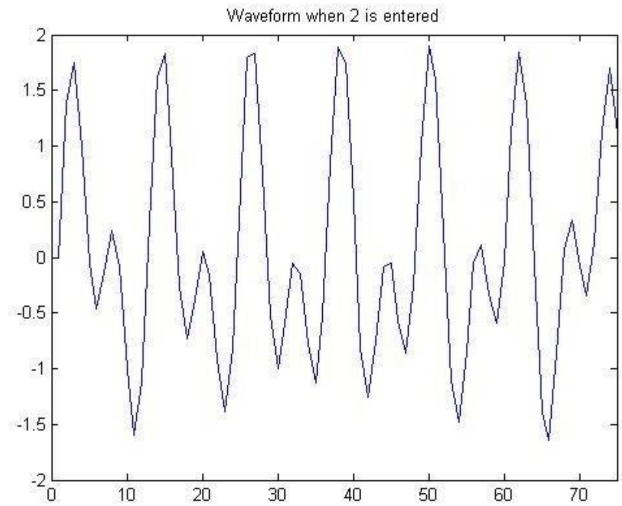

Figure 2.2

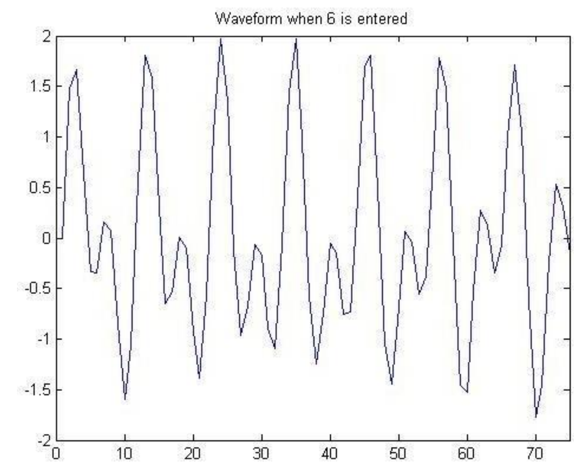

Figure 2.3

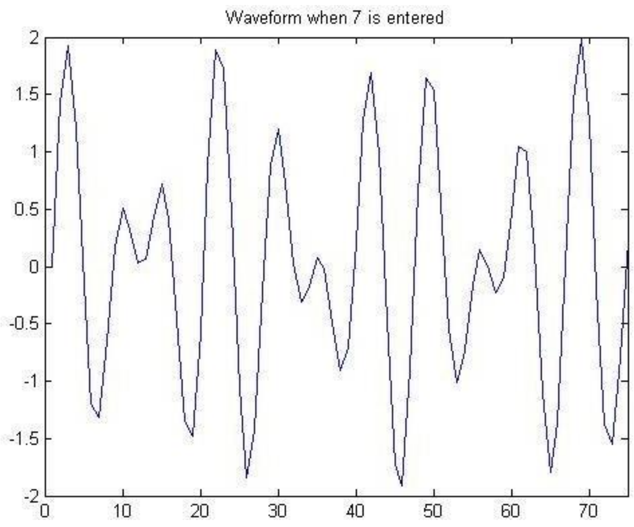

Figure 2.4

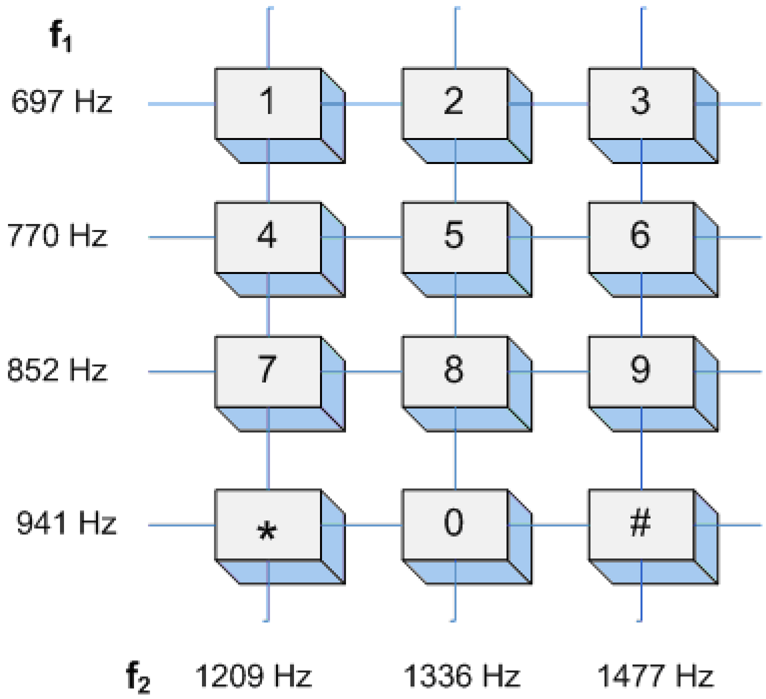

Figure 2.5

2.3 SOLUTION

A smart home can be viewed as an integration system which takes advantage of a range of techniques such as computers, network communication as well as synthesized wireless technology to connect all indoor subsystems that attach to home appliances and household electrical devices. In this way, smart home technology enables households to effectively centralize the management and services in a home, provide them with all-round functions for internal information exchange and help to keep in instant contact with the outside world. Smart home applications employ microcontrollers to monitor ovens, washing machines, lighting, refrigerators, and HVAC facilities (Heating/Ventilation/Air-Conditioning) with respect to temperature or humidity and to adjust accordingly to meet the home owner’s requirement.

CHAPTER 3

PROJECT DESCRIPTION

3.1 INTRODUCTION

Imagine how helpful it will be to be able to switch on your air conditioning system ten minutes before you get home on a hot afternoon in June. How about having a security system that will detect smoke, excessive electrical power usage, burglar attempts and unauthorized movements in your house and alert you? This is what home automation is about and there is no end to its application. In fact, sophisticated home automation systems are now being developed that can maintain an inventory of household items and prepare a shopping list or automatically order replacements.

Home Automation has made it possible to have what is often referred to as a 'smart home', a home that can detect and identify you, automatically adjust the lighting to your predefined taste, open doors automatically, play your favourite music, water your flowers in the morning, switch on the security lights at night and switch them off in the morning, heat water for bathe and tea, stream to you anywhere in the world via the internet a live video of what is happening in and around your house. It makes it possible to link lighting, entertainment, security, telecommunications, heating, and air conditioning into one centrally controlled system. This allows you to make your house an active partner in managing your busy life.

Nowadays in foreign land, you can hardly find a house without a home automation system which can range from the remote for the television, burglar alarm and hi-tech security gates, to an automated air conditioning system that maintains the temperature at a predefined value.

3.2 AUTOMATION

Automation is the use of control systems and information technology to control equipment, industrial machinery and processes, reducing the need for human intervention. In the scope of industrialization, automation is a step beyond mechanization. Mechanization provided human operators with machinery to assist them with the physical requirements of work while automation greatly reduces the need for human sensory and mental requirements as well. Automation plays an increasingly important role in the global economy and in daily experience. Engineers strive to combine automated devices with mathematical and organizational tools to create complex systems for a rapidly expanding range of applications and human activities. Many roles for humans in industrial processes presently lie beyond the scope of automation. Human- level pattern recognition, language recognition, and language production ability are well beyond the 12 capabilities of modern mechanical and computer systems.

Tasks requiring subjective assessment or synthesis of complex sensory data, such as scents and sounds, as well as high-level tasks such as strategic planning, currently require human expertise. Automation has had a notable impact in a wide range of highly visible industries beyond manufacturing. Once ubiquitous telephone operators have been replaced largely by automated telephone switchboards and answering machines. Medical processes such as primary screening in electrocardiograph or radiography and laboratory analysis of human genes, blood plasmas, cells, and tissues are carried out at much greater speed and accuracy by automated systems.

Automated teller machines have reduced the need for bank visits to obtain cash and carry out transactions. In general, automation has been responsible for the shift in the world economy from agrarian to industrial in the 19th century and from industrial to services in the 20th century.

3.3 HOME AUTOMATION

Home automation may designate an emerging practice of increased automation of household appliances and features in residential dwellings, particularly through electronic means that allow for things impracticable, overly expensive or simply not possible in recent decades. Home automation includes all that a building automation provides like climate controls, door and window controls, and in addition control of multimedia home theatres, pet feeding, plant watering and so on. But there exists a difference in that home automation emphasizes more on comforts through ergonomics and ease of operation.

DTMF Home Automation lets you operate your home appliances like lights and water pump from your office or any other remote place. So if you forgot to switch off the lights or other appliances while going out, it helps you to turn off the appliance with your cell phone. **Your cell phone works as remote control to your home appliances. You can control the desired appliance by presenting the corresponding key.**

CHAPTER 4

COMPONENTS

4.1 MICRO CONTROLLER

A microcontroller is a compact microcomputer designed to govern the operation of embedded systems in motor vehicles, robots, office machines, complex medical devices, and various other devices. A typical microcontroller includes a processor, memory, and peripherals. The simplest microcontrollers facilitate the operation of the electromechanical systems found in everyday convenience items. Originally, such use was confined to large machines such as furnaces and automobile engines to optimize efficiency.

In recent years, microcontrollers have found their way into common items such as ovens, refrigerators, toasters, Microcomputers are also common in office machines such as photocopiers, scanners, and printers. The most sophisticated microcontrollers perform critical functions in aircraft, spacecraft, oceangoing vessels, life -support systems, and robots of all kinds. Medical technology offer s especially promising future roles. For example, a microcontroller might regulate the operation of an artificial heart, artificial kidney, or other artificial body organ. Microcomputers can also function with prosthetic devices (artificial limbs). A few medical -science futurists have suggested that mute patients might somedaybe able, in effect, to speak out loud by thinking of the words they want to utter, while a microcontroller governs the production of audio signals to drive an amplifier and loudspeaker.

Here the microcontroller used is ARDUINO UNO.

Figure 4.1

Arduino is an open source electronics prototyping platform that is flexible, easy -to-use hardware and software. It is designed for artists, designers, hobbyists and anyone interested in creating interactive objects or environments. Arduino Uno is basically based on ATmega328 microcontroller (MCU). It consists of 14 digital input/output pins, six analog inputs, a USB connection used for programming the onboard MCU, a power jack, an ICSP header and a reset button. It is operated with the help of a 16MHz crystal oscillator and contains everything needed to support the MCU. It is very easy to use as we simply need to connect it to a computer using a USB cable, or power it with an AC-to-DC adaptor or battery to get started. The MCU onboard is programmed in Arduino programming language using Arduino IDE.

4.1.1 ARDUINO SPECIFICATIONS

Microcontroller	ATmega328
Clock Speed	16MHz
Operating Voltage	5V
Maximum supply Voltage (not recommended)	20V
Supply Voltage (recommended)	7-12V
Analog Input Pins	6
Digital Input/Output Pins	14
DC Current per Input/Output Pin	40mA
DC Current in 3.3V Pin	50mA
SRAM	2KB
EEPROM	1KB
Flash Memory	32KB of which 0.5KB used by boot loader

Table 4.1

4.1.2 POWER

The Arduino/Genuino Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non - USB) power can come either from an AC -to-DC adapter (wall -wart) or battery. The adapter can be connected by plugging a 2.1mm center -positive plug into the board's power jack. Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector. The board can operate on an external supply from 6 to

20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts. The power pins are as follows:

- **Vin:** The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. 18
- **5V:** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3:** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND**: Ground pins. IOREF. This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.

4.1.3 MEMORY

The ATmega328 has 32 KB (with 0.5 KB occupied by the boot-loader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

4.1.4 INPUT AND OUTPUT

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial:** 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts:** 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM:** 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.
- **SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- **LED:** 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

- **TWI:** A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library. There are a couple of other pins on the board:
- **AREF:** Reference voltage for the analog inputs. Used with analogReference().
- **Reset:** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

4.1.5 COMMUNICATION

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed.

However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A SoftwareSerial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

4.1.6 PROGRAMMING

The Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board). For details, see the reference and tutorials. The ATmega328 on the Arduino Uno comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files). You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details. The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available .

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

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

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

4.2 DTMF DECODER MODULE

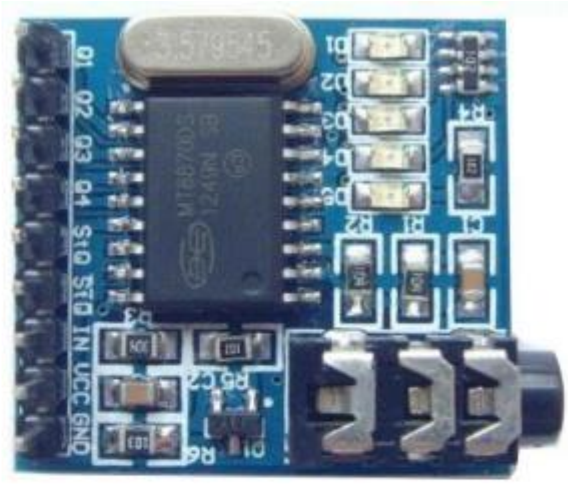

Figure 4.2

4.2.1 DESCRIPTION

The MT8870D/MT8870D-1 is a complete DTMF receiver integrating both the band split filter and digital decoder functions. The filter section uses switched capacitor techniques for high and low group filters; the decoder uses digital counting techniques to detect and decode all 16 DTMF tone-pairs into a 4-bit code. External component count is minimized by on chip provision of a differential input amplifier, clock oscillator and latched three-state bus interface.

4.2.2 INTERNAL CIRCUIT

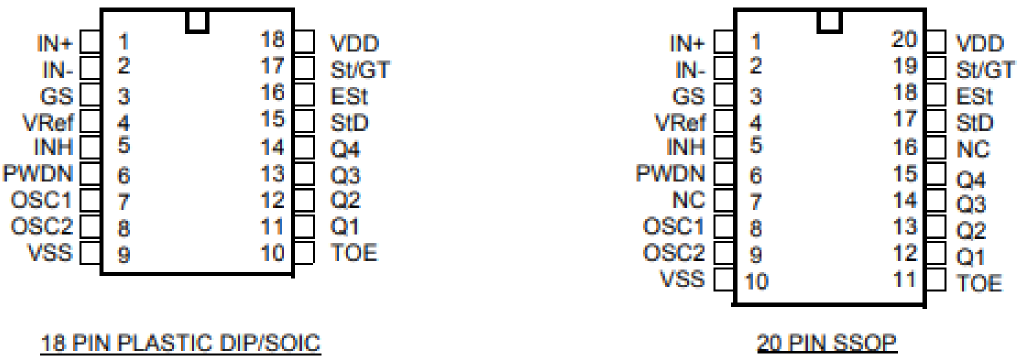

Figure 4.3

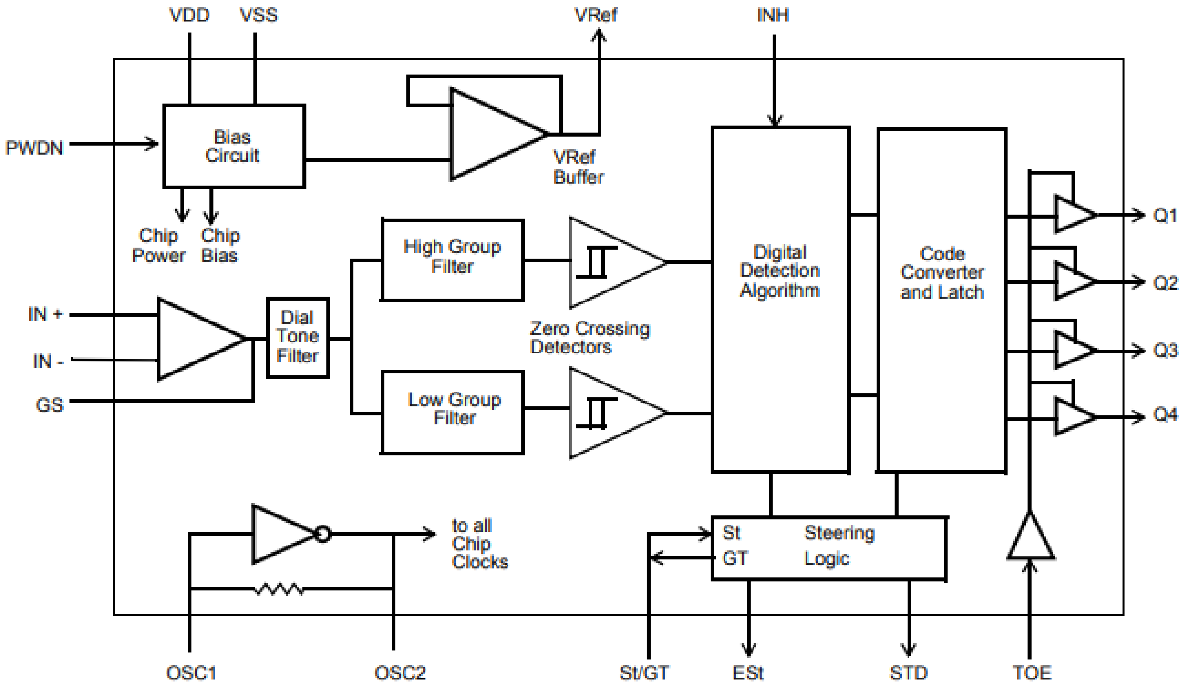

Figure 4.4

Pin #		Name	Description
18	20		
1	1	IN+	**Non-Inverting Op-Amp (Input)**.
2	2	IN-	**Inverting Op-Amp (Input)**.
3	3	GS	**Gain Select.** Gives access to output of front end differential amplifier for connection of feedback resistor.
4	4	V_{Ref}	**Reference Voltage (Output).** Nominally $V_{DD}/2$ is used to bias inputs at mid-rail (see Fig. 6 and Fig. 10).
5	5	INH	**Inhibit (Input).** Logic high inhibits the detection of tones representing characters A, B, C and D. This pin input is internally pulled down.
6	6	PWDN	**Power Down (Input).** Active high. Powers down the device and inhibits the oscillator. This pin input is internally pulled down.
7	8	OSC1	**Clock (Input)**.
8	9	OSC2	**Clock (Output)**. A 3.579545 MHz crystal connected between pins OSC1 and OSC2 completes the internal oscillator circuit.
9	10	V_{SS}	**Ground (Input)**. 0 V typical.
10	11	TOE	**Three State Output Enable (Input).** Logic high enables the outputs Q1-Q4. This pin is pulled up internally.
11-14	12-15	Q1-Q4	**Three State Data (Output).** When enabled by TOE, provide the code corresponding to the last valid tone-pair received (see Table 1). When TOE is logic low, the data outputs are high impedance.
15	17	StD	**Delayed Steering (Output).**Presents a logic high when a received tone-pair has been registered and the output latch updated; returns to logic low when the voltage on St/GT falls below V_{TSt}.
16	18	ESt	**Early Steering (Output).** Presents a logic high once the digital algorithm has detected a valid tone pair (signal condition). Any momentary loss of signal condition will cause ESt to return to a logic low.

Table 4.2

Pin #		Name	Description
18	20		
17	19	St/GT	**Steering Input/Guard time (Output) Bidirectional.** A voltage greater than V_{TSt} detected at St causes the device to register the detected tone pair and update the output latch. A voltage less than V_{TSt} frees the device to accept a new tone pair. The GT output acts to reset the external steering time-constant; its state is a function of ESt and the voltage on St.
18	20	V_{DD}	**Positive power supply (Input).** +5 V typical.
	7, 16	NC	No Connection.

Table 4.3

4.2.3 FUNCTIONAL DESCRIPTION

The MT8870D/MT8870D-1 monolithic DTMF receiver offers small size, low power consumption and high performance. Its architecture consists of a band split filter section, which separates the high and low group tones, followed by a digital counting section which verifies the frequency and duration of the received tones before passing the corresponding code to the output bus.

A. FILTER SECTION

Separation of the low-group and high group tones is achieved by applying the DTMF signal to the inputs of two sixth-order switched capacitor band pass filters, the bandwidths of which correspond to the low and high group frequencies. The filter section also incorporates notches at 350 and 440 Hz for exceptional dial tone rejection (see Figure 3). Each filter output is followed by a single order switched capacitor filter section which smoothens the signals prior to limiting. Limiting is performed by high-gain comparators which are provided with hysteresis to prevent detection of unwanted low-level signals. The outputs of the comparators provide full rail logic swings at the frequencies of the incoming DTMF signals.

B. DECODER SECTION

Following the filter section is a decoder employing digital counting techniques to determine the frequencies of the incoming tones and to verify that they correspond to standard DTMF frequencies. A complex averaging algorithm protects against tone simulation by extraneous signals such as voice while providing tolerance to small frequency deviations and variations. This averaging algorithm has been developed to ensure an optimum combination of immunity to talk-off and tolerance to the presence of interfering frequencies (third tones) and noise. When the detector recognizes the presence of two valid tones (this is referred to as the "signal condition" in some industry specifications) the "Early Steering" (ESt) output will go to an active state. Any subsequent loss of signal condition will cause Est to assume an inactive state (see "Steering Circuit").

C. STEERING CIRCUIT

Before registration of a decoded tone pair, the receiver checks for a valid signal duration (referred to as character recognition condition). This check is performed by an external RC time constant driven by Est. A logic high on Est causes vc (see Figure 4) to rise as the capacitor discharges. Provided signal condition is maintained (Est remains high) for the validation period (tGTP), vc reaches the threshold (VTSt) of the steering logic to register the tone pair, latching its corresponding 4-bit code (see Table 1) into the output latch. At this point the GT output is activated and drives vc to VDD. GT continues to drive high as long as Est remains high.

Finally, after a short delay to allow the output latch to settle, the delayed steering output flag (StD) goes high, signaling that a received tone pair has been registered. The contents of the output latch are made available on the 4-bit output bus by raising the three state control input (TOE) to a logic high. The steering circuit works in reverse to validate the inter digit pause between signals. Thus, as well as rejecting signals too short to be considered valid, the receiver will tolerate signal interruptions (dropout) too short to be considered a valid pause. This facility, together with the capability of selecting the steering time constants externally, allows the designer to tailor performance to meet a wide variety of system requirements.

D. CRYSTAL OSCILLATOR

The internal clock circuit is completed with the addition of an external 3.579545 MHz crystal and is normally connected as shown in Figure 10 (Single-Ended Input Configuration). However, it is possible to configure several MT8870D/MT8870D-1 devices employing only a single oscillator crystal. The oscillator output of the first device in the chain is coupled through a 30 pF capacitor to the oscillator input (OSC1) of the next device. Subsequent devices are connected in a similar fashion. The problems associated with unbalanced loading are not a concern with the arrangement shown, i.e., precision balancing capacitors are not required.

Button	Low DTMF frequency (Hz)	High DTMF frequency (Hz)	Binary coded output			
			Q1	Q2	Q3	Q4
1	697	1209	0	0	0	1
2	697	1336	0	0	1	0
3	697	1477	0	0	1	1
4	770	1209	0	1	0	0
5	770	1336	0	1	0	1
6	770	1477	0	1	1	0
7	852	1209	0	1	1	1
8	852	1336	1	0	0	0
9	852	1477	1	0	0	1
0	941	1336	1	0	1	0
*	941	1209	1	0	1	1
#	941	1477	1	1	0	0

Table 4.4

4.3 RELAY

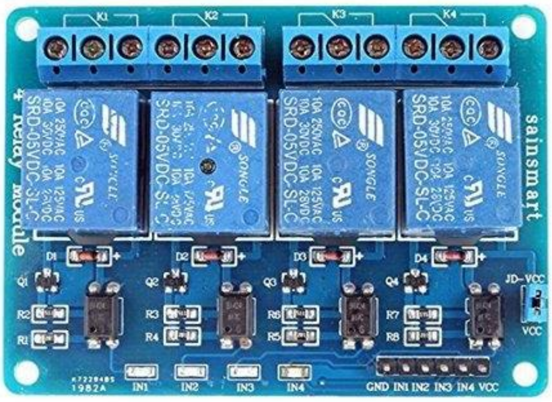

Figure 4.5

4.3.1 DESCRIPTION

The relay module is an electrically operated switch that allows you to turn on or off a circuit using voltage and/or current much higher than a microcontroller could handle. There is no connection between the low voltage circuit operated by the microcontroller and the high power circuit. The relay protects each circuit from each other.

The each channel in the module has three connections named NC, COM, and NO. Depending on the input signal trigger mode, the jumper cap can be placed at high level effective mode which 'closes' the normally open (NO) switch at high level input and at low level effective mode which operates the same but at low level input.

4.3.2 SPECIFICATIONS

- On-board EL817 photoelectric coupler with photoelectric isolating anti-interference ability strong
- On-board 5V, 10A / 250VAC, 10A / 30VDC relays

• Relay long life can absorb 100000 times in a row

• Module can be directly and MCU I/O link, with the output signal indicator

• Module with diode current protection, short response time

• PCB Size: 45.8mm x 32.4mm

4.3.3 PIN CONFIGURATION

- VCC
- GND
- IN1
- IN2
- COM
- NC
- NO

4.3.4 WORKING PRINCIPLE

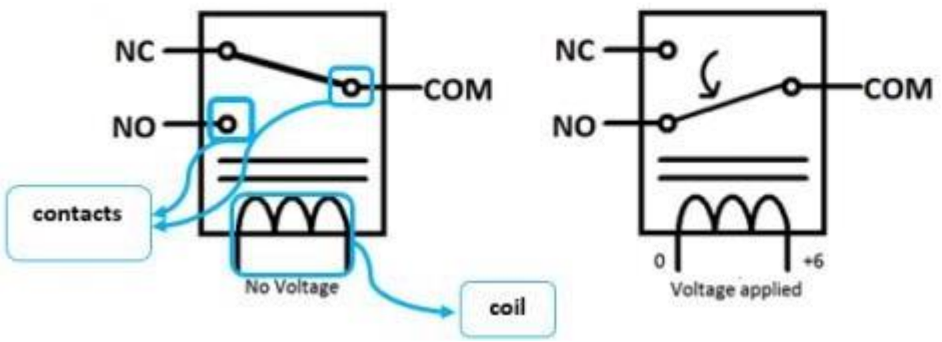

Figure 4.6

Relays consist of three pins normally open pin , normally closed pin, common pin and coil. When coil powered on magnetic field is generated the contacts connected to each other.

CHAPTER 5

PROJECT ANALYSIS

5.1 BLOCK DIAGRAM

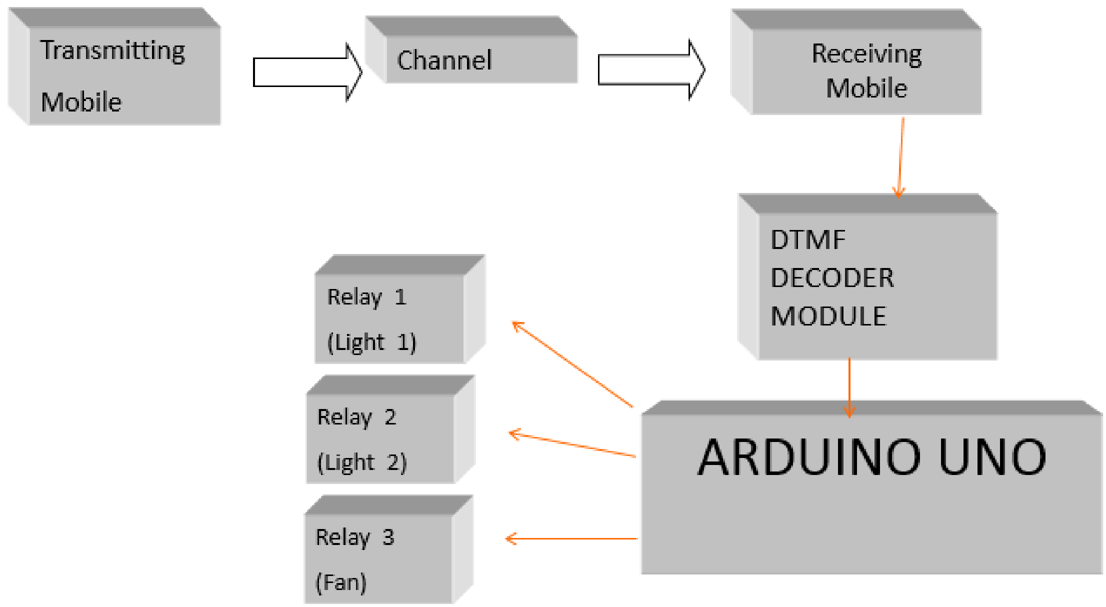

Figure 5.1

5.2 POJECT SETUP

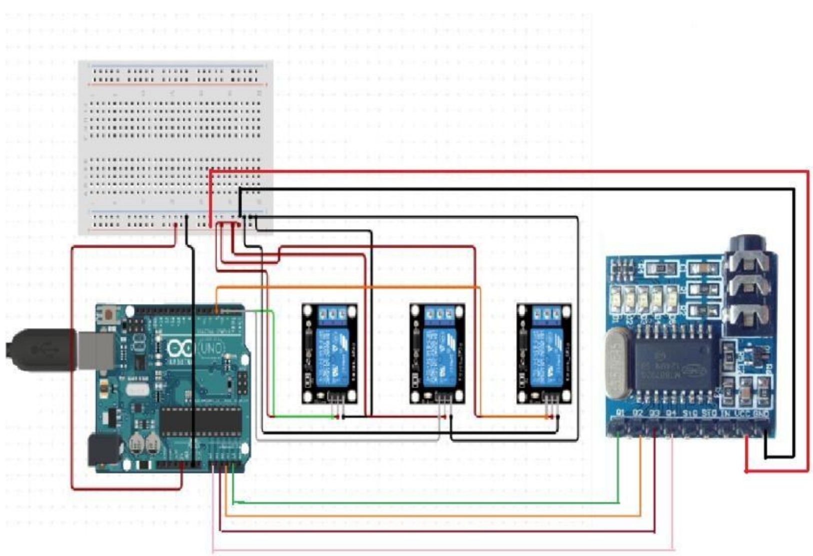

Figure 5.2

5.3 ALGORITHM

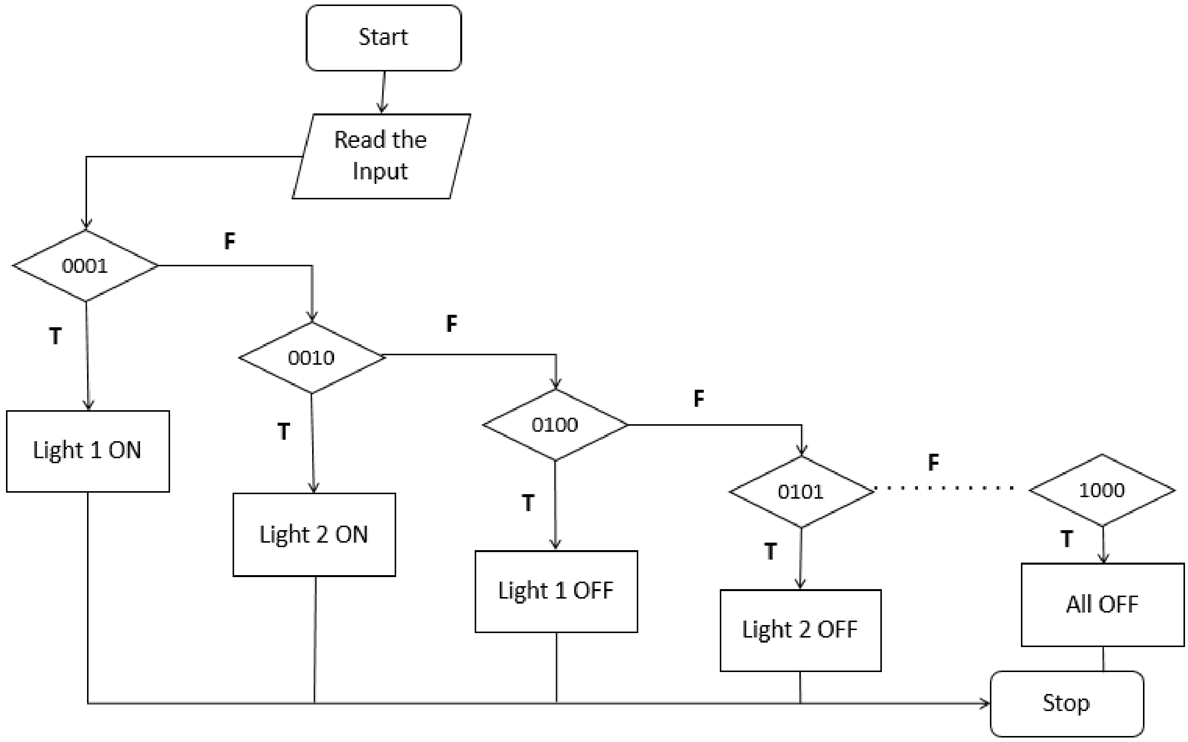

Figure 5.3

5.4 WORKING

The following are the steps corresponding to the project

1. After connecting components as per the project setup mentioned above, switch ON the power supply.
2. Make a call to the mobile connected to the DTMF decoder module and make sure that the mobile is in auto-answer mode.
3. As per the requirements, dial the number which in turn switches the load to ON and OFF. With regard to the number dialed, check the below table for the operation going to happen (i.e which load will be controlled).

KEY PRESSED	OPERATION DONE
1	Light 1 ON
2	Light 2 ON
3	Fan ON
4	Light1 OFF
5	Light2 OFF
6	Fan OFF
7	ALL ON
8	ALL OFF

Table 5.1

CHAPTER 6

RESULTS AND CONCLUSION

6.1 OUTPUT

Figure 6.1

When key 1 is pressed, relay 1 is made ON, which in turn makes the light 1 ON.

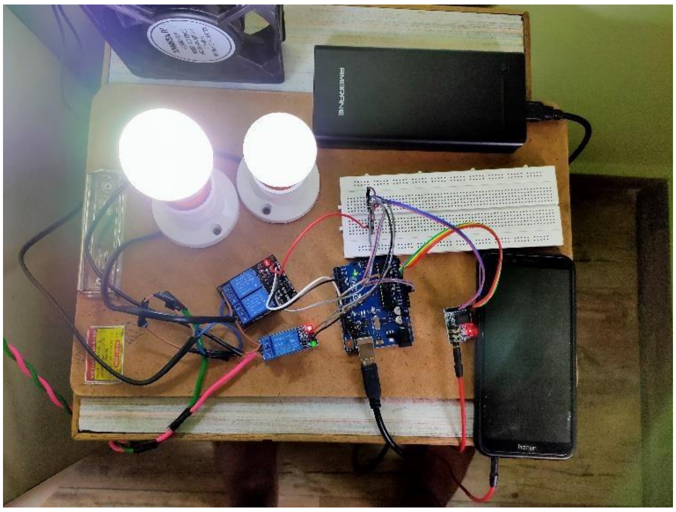

Figure 6.2

When key 2 is pressed, relay 2 is made ON, which in turn makes the light 2 ON.

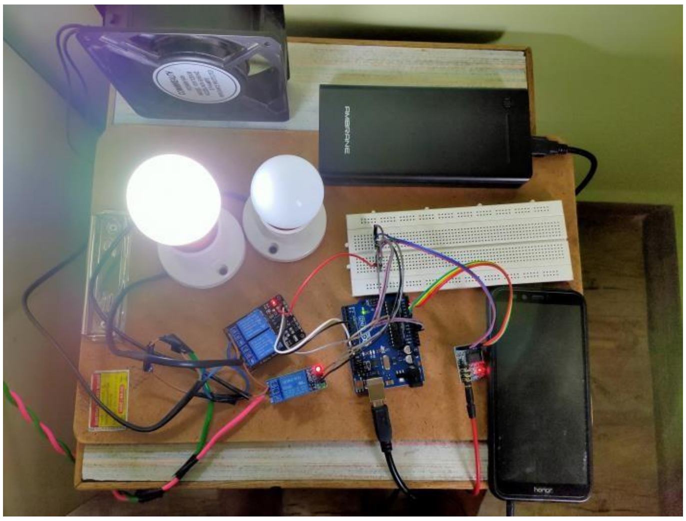

Figure 6.3

When key 4 is pressed, relay 1 is made OFF, which in turn makes the light 1 OFF.

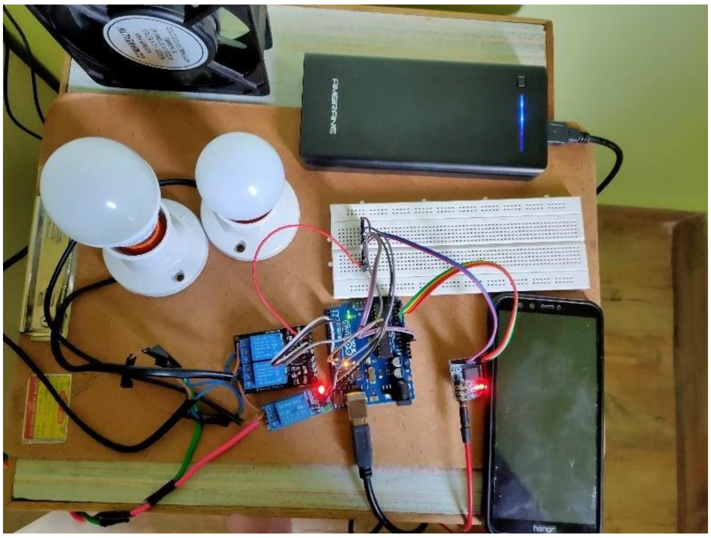

Figure 6.4

When key 5 is pressed, relay 2 is made OFF, which in turn makes the light 2 OFF.

The switching of relay when a key is pressed is irrespective of other relays, whether they are in ON or OFF position.

6.2 APPLICATIONS

1. Unique system of Automation.
2. Remote Mobile System.
3. Automated Home Appliances.
4. Assistance for Old People.

6.3 CONCLUSION

This report has discussed the development of a Semi automated load control system with the help of the dual tone multi frequency concept .The objective of our project was to develop the necessary hardware and software to have the DTMF decoder switch the load to on and off according to user application .The objective to switch the load according to user application were met by keeping track of the digit pressed, the DTMF module was able to control the loads i.e the light, fan.

This project introduced us to the important topics of A/D conversion, D/A conversions, filter applications, telecommunications and serial communications

In the project, a D/A converter allowed us access to Digital inputs of dual tones from a digital keypad i.e dial keypad in mobile. This digital signal is converted to analog and supplied to DTMF module via AUX cable in analog form. Later, after the process in the DTMF decoder these analog signal are converted to digital signal to switch the latches that in turn reflect in switching the loads.

DTMF Based load control system has been designed and setup which makes it possible to control all home appliances automatically using our own mobile phones. The control of all appliances is possible even from a wide range.

REFERENCES

1. H. Schulzrinne and T. Taylor, RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals, IETF RFC 4733, December 2006.
2. Zarlink Semiconductors Inc, https://www.microsemi.com/document-portal/doc_view/127041-mt8870d-datasheet-oct2006.
3. Premium Farnell Ltd, https://www.farnell.com/datasheets/1682209.pdf
4. https://www.fecegypt.com/uploads/dataSheet/1480848003_2_channel_5v_10a_relay_module.pdf

APPENDIX - A

Program Code:

```
// Pins Setup
int Q1=A0; //Data pin 1 of DTMF Decoder
int Q2=A1; //Data pin 2 of DTMF Decoder
int Q3=A2; //Data pin 3 of DTMF Decoder
int Q4=A3; //Data pin 4 of DTMF Decoder
int Relay1=12;
int Relay2=13;
int Relay3=11;

void setup()
{
 pinMode(Relay1,OUTPUT);
 pinMode(Relay2,OUTPUT);
 pinMode(Relay3,OUTPUT);
 digitalWrite(Relay1,HIGH);
 digitalWrite(Relay2,HIGH);
 digitalWrite(Relay3,HIGH);
}
void loop()
{
        //when no key is pressed all the relays are in off condition
  if(digitalRead(Q4)==0 && digitalRead(Q3)==0 && digitalRead(Q2)==0 && digitalRead(Q1)==0 )
  {
    digitalWrite(Relay1,HIGH);
    digitalWrite(Relay2,HIGH);
    digitalWrite(Relay3,HIGH);
    delay(200);
  }
```

```
 // when key 1 is pressed relay1 is in on condition
 if (digitalRead(Q4)==0 && digitalRead(Q3)==0 && digitalRead(Q2)==0 &&
digitalRead(Q1)==1 )
 {
   digitalWrite(Relay1,LOW);
   delay(200);
 }
// when key 2 is pressed relay2 is in on condition
 if (digitalRead(Q4)==0 && digitalRead(Q3)==0 && digitalRead(Q2)==1 &&
digitalRead(Q1)==0 )
 {
   digitalWrite(Relay2,LOW);
   delay(200);
 }
// when key 3 is pressed relay3 is in on condition
 if (digitalRead(Q4)==0 && digitalRead(Q3)==0 && digitalRead(Q2)==1 &&
digitalRead(Q1)==1 )
 {
   digitalWrite(Relay3,LOW);
   delay(200);
 }


// when key 4 is pressed relay1 is turned off
 if (digitalRead(Q4)==0 && digitalRead(Q3)==1 && digitalRead(Q2)==0 &&
digitalRead(Q1)==0 )
 {
   digitalWrite(Relay1,HIGH);
   delay(200);
 }

    // when key 5 is pressed relay2 is turned off
 if (digitalRead(Q4)==0 && digitalRead(Q3)==1 && digitalRead(Q2)==0 &&
digitalRead(Q1)==1 )
```

```
  {
    digitalWrite(Relay2,HIGH);
    delay(200);
  }
// when key 6 is pressed relay3 is turned off
  if (digitalRead(Q4)==0 && digitalRead(Q3)==1 && digitalRead(Q2)==1 &&
digitalRead(Q1)==0 )
  {
    digitalWrite(Relay3,HIGH);
    delay(200);
  }

    // when key 7 is pressed all relays are turned on
  if (digitalRead(Q4)==0 && digitalRead(Q3)==1 && digitalRead(Q2)==1 &&
digitalRead(Q1)==1 )
  {
    digitalWrite(Relay1,LOW);
    digitalWrite(Relay2,LOW);
    digitalWrite(Relay3,LOW);
    delay(200);
  }

    // when key 8 is pressed all relays are turned off
  if (digitalRead(Q4)==1 && digitalRead(Q3)==0 && digitalRead(Q2)==0 &&
digitalRead(Q1)==0 )
  {
    digitalWrite(Relay1,HIGH);
    digitalWrite(Relay2,HIGH);
    digitalWrite(Relay3,HIGH);
    delay(200);
  }
```

APPENDIX – B

Matlab Code:

```
clc;
clear all;
close all;
number=input('Enter the Number');
fs=8000;
T=0.5;
t=0:1/fs:T;
for k=1:length(number)
    switch number(k)
        case 1
            tone=sin(2*pi*697*t)+sin(2*pi*1209*t);
            sound(tone);
            figure();
            plot(tone);
            axis([0 75 -2 2]);
            title('Waveform when 1 is entered');
        case 2
            tone=sin(2*pi*697*t)+sin(2*pi*1336*t);
            sound(tone);
            figure();
            plot(tone);
            axis([0 75 -2 2]);
            title('Waveform when 2 is entered');
        case 3
            tone=sin(2*pi*697*t)+sin(2*pi*1477*t);
            sound(tone);
            figure();
            plot(tone);
            axis([0 75 -2 2]);
            title('Waveform when 3 is entered');
```

```
case 4
   tone=sin(2*pi*770*t)+sin(2*pi*1209*t);
   sound(tone);
   figure();
   plot(tone);
   axis([0 75 -2 2]);
   title('Waveform when 4 is entered');
case 5
   tone=sin(2*pi*770*t)+sin(2*pi*1336*t);
   sound(tone);
   figure();
   plot(tone);
   axis([0 75 -2 2]);
   title('Waveform when 5 is entered');
case 6
   tone=sin(2*pi*770*t)+sin(2*pi*1477*t);
   sound(tone);
   figure();
   plot(tone);
   axis([0 75 -2 2]);
   title('Waveform when 6 is entered');
case 7
   tone=sin(2*pi*852*t)+sin(2*pi*1209*t);
   sound(tone);
   figure();
   plot(tone);
   axis([0 75 -2 2]);
   title('Waveform when 7 is entered');
case 8
   tone=sin(2*pi*852*t)+sin(2*pi*1336*t);
   sound(tone);
   figure();
   plot(tone);
   axis([0 75 -2 2]);
```

```
            title('Waveform when 8 is entered');
                otherwise
            disp('Invalid Number');
    end;
    pause(5);
end
```