DTMF (DUAL TONE MULTI-FREQUENCY) HOME POWER MANAGEMENT

OLUGBILE ADEYEMI OLUTOSIN

2006/24380EE

A PROJECT SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING, FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA.
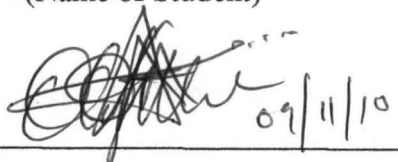
NOVEMBER, 2010.

i

DEDICATION

This project is dedicated to my parents; Prince and Reverend (Mrs) Segun Bola Olugbile. My Siblings; Olugbile Adeyinka,Olugbile Oluwayomade and Olugbile Oluwayoyinsola.

DECLARATION

I, OLUGBILE ADEYEMI OLUTOSIN, declare that this work was done by me and has never presented elsewhere for the award of degree. I also here by relinquish the copyright to Federal University Of Technology, Minna.
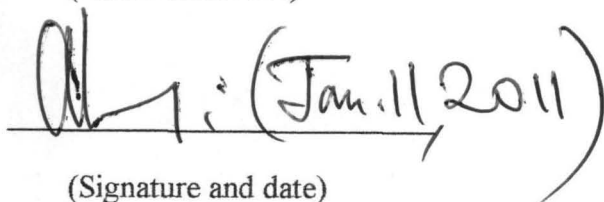

OLUGBILE ADEYEMI OLUTOSIN   MR. TOLA OMOKHAFE

_____   _____

(Name of Student)   (Name of Supervisor)

_____   _____

(Signature and date)   (Signature and date)

ENGR. A.G. RAJI

_____   _____
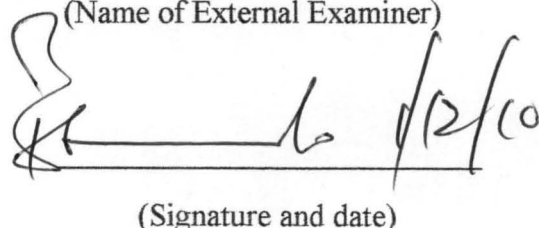
(Name of H.O.D.)   (Name of External Examiner)

_____   _____

(Signature and date)   (Signature and date)

ACKNOWLEDGEMENT

ABSTRACT

This paper describes how home appliances can be controlled remotelly. This is achieved with a mobile phone of a user located far away or anywhere from home calling a stationary mobile phone connected to this device at home. This is done through the GSM (Global System for Mobile communication) network. However, this project is DTMF based that is Dual Tone Multi-Frequency. Which is the signal generated from keypads of any mobile phone. It is dual tone in the sense that each key on the keypad is composed of two tones (a high and a low tone). To control home appliance at home remotely or ubiquitously, a SIM(Subscriber Identity Module) card is inserted in the stationary mobile phone connected to this device at home.So, the home owner calls the stationary mobile phone and when the phone rings, it automatically picks up the call through an headset connected to the mobile phone. Then the user can input the necessary command to control any appliance connected to this device. If the command is executed successfully a feedback tone will be sent twice and if it is a wrong command a long feedback tone will also be sent. So one can be sure if the device carries the operation one intends for it to carry out. Moreover, this project will help users to control their home appliances from anywhere they are at anytime if either they forgot to put it off before going out or they wish to put it on before they get home.

TABLE OF CONTENTS

Chapter Five:

LIST OF FIGURES

LIST OF TABLES

x

CHAPTER ONE

1.0   INTRODUCTION

When an individual is away from home to attend to important issues like go to work, church, ceremonies etc, there are high tendencies that the mind of the individual will be crowded with doubts and questions that are difficult to answer at that point. Questions like; did I turn off the Electric Cooker? Is the Heater off? And so on. [1]

With this project, there is nothing to worry about. You can stay wherever you are at anytime, anywhere or everywhere to control, monitor and check the status of your home appliance. Also, someone can prevent intruders from getting in to your house while you are not there.

However, Home Appliance can be controlled using various technologies. They are:

- GSM-VOICED Technology

- GSM-SMS Technology

- Internet Technology

- Bluetooth Technology

- Infra-red Technology

- DTMF(dual tone multi frequency spectrum) Technology signal

This project makes use of DTMF Technology in controlling of home appliance and prevention of intrusion against security ubiquitously. An

individual from anywhere can press buttons from his mobile phone to call a stationary mobile phone connected to these device to put off, put on or check the status of any device at home. For example, while coming back from work you can switch on your heater to heat water to take a nice shower before you get home, and turn it off after sometime as you wish.

1.1   OBJECTIVES

This project is design and constructed to develop a cost effective solution that will control home appliances remotely/ubiquitously.

1.2   METHODOLOGY

8051 microcontroller was used to interface with the mobile phone with dtmf decoder in this device. With this the microcontrollers behave as the master which will produce signals to control the transfer of information to the stationary phone at home. This is achieved through the DTMF signal generated from the remote mobile phones when keys are pressed on the phone. Relays are also connected to the microcontroller, they are the major components used for the controlling of the home appliances that will be connected to this device.

1.3   SCOPE OF THE PROJECT

This project employs the use of a microcontroller 8052, a dtmf decoder, a relay, crystal oscillator and a power supply unit. A programmer to program the microcontroller. An Assembly language program is used for the project. Two mobile phones, one stationary mobile phone to interface with the microcontroller and the other mobile phone can be any model.

1.4   SOURCES OF MATERIAL

The major source of material for this project was from the Internet. Also, the Departmental library and laboratory was very useful in contributing to how this project was implemented. Then, in carrying out the construction components that are needed were purchased from component vendors.

1.5   CONSTRAINT

The system functionality is based on GSM technology so the technological constraints must be kept in mind. The system is vulnerable to power failure but this disruption can be avoided by attaching the voltage source thus allowing users to avail the great advantage of this system. Also, due to the fact that the project is dependent on the stability of telecommunication network, fluctuation of signals in the network is also a constraint.

1.6   IMPORTANCE OF THE PROJECT

This project can be used to manage home appliances ubiquitously by pressing the buttons on your mobile phone without leaving wherever you are. It also means that wherever you are as a home owner, you can control, monitor home appliances that are in your home without physically coming in contact with them.

Moreover, for security purpose, someone can watch how several activities that are going on in your compound while you are either away from home or on your bed at night. With this you can prevent intrusion against security from happening in your house.

Also, for disables and the very old adult, this project will be of help to them. As many of them may have restricted movement either for health reasons or disability.

1.7   PROJECT LAYOUT

This project is composed of five chapters. The first chapter talks about the introduction, scope of the project, importance of the project, justification of the project, methodology, objectives of the project and the project layout. Chapter two contains the literature review in which it has the historical background of the project and the description of each component used in the project. Chapter three talks about design and implementation of the project. Chapter four is construction and testing of the project. Chapter five has the conclusion about the project and recommendation of the project.

CHAPTER TWO

2.0 LITERATURE REVIEW/THEORETICALBACKGROUND

2.1   THEORETICAL BACKGROUND

The gradual development of microelectronics and communications technologies has led to the idea of Automation. This Automation comes in various forms; they are home automation, car- theft tracking system, anti-theft immobilizer and so on. [2] The study of microelectronics and communications technologies has also given rise to what is called EMBEDDED SYSTEMS. An embedded system is a system in which involves the use of microcontroller embedded in machines such as automobiles, telephones, appliances and peripherals for computer system to carry out some specific task. This use a microcontroller to interface with some output devices.[3] Typical input and output devices include switches, relays, solenoids, LEDs, frequency devices, and sensors for data such as temperature, humidity, light level etc. [3]

Moreover, the controlling and monitoring of home appliances generally is done through the use of dtmf signal emitted from dtmf encoder in a stationary or mobile phone. All this can be achieved by uploading some set of instructions or codes called programs on to the microcontroller to carry-out the objectives of this device.

Microcontroller programs must fit in the available on-chip program memory, since it would be costly to provide a system with external, expandable, memory. Compilers and assemblers are used to turn high-level language and assembler language codes into a compact machine code for storage in the microcontroller's memory. Depending on the device, the program memory may be permanent, read-only memory that can only be programmed at the

5

factory, or program memory may be field-alterable flash or erasable read-only memory[4].

Microcontrollers come with different flavours of integrated program memory, either ROM, OTP (one-time-programmable), or FLASH. With the later becoming a very popular choice with designers in recent years and FLASH technology steadily evolving. In this part of the project report, a closer look at the significantly different microcontroller FLASH implementations on the market to highlight some points you may want to consider when choosing your next FLASH micro will be looked into. [3]

2.1.1 The Benefits of FLASH

The benefit **any** FLASH microcontroller offers over the older ROM and OTP (one-time-programmable) technologies is quickly summarized in one word: **re-programmability**.

Eliminate Scrap: Re-programmability eliminates the scrap of obsolete material associated with ROM and OTP (one-time-programmable). It can save you literally thousands of dollars in the early production ramp-up phase. [5]

Of course, there are much more benefits that FLASH offers, but whether you can utilize those or not depends on the particular FLASH implementation on your microcontroller of choice. In the following paragraphs, I'm going to take a closer look at those different flavours and the additional benefits they hold. [5]

My Generation?

Since its introduction in the early 90's, FLASH memory based microcontroller technology has evolved over three distinct generations, all of which are still in the market today. So in selecting your FLASH microcontroller, it is important to understand the differences this evolution has created and the implications it might have for your particular application.

Generation 1: Out-of-System Programming

First generation FLASH devices use a separate high voltage supply for programming. Those devices are very much like OTPs (one-time-programmable) in every aspect, with the added advantage of being re-programmable. The drawback of using an external high voltage programming supply is that the device needs to be removed from the circuit for reprogramming. Obviously, to be able to remove the part, it needs to be socketed, which severely limits your choice of feasible packages and adds additional cost to your application. If you choose to solder the part onto the board, you've either got an OTP (one-time-programmable) or you're willing to pay the re-work cost to de-solder the part if necessary.

Generation 2: In-System-Programming

Second generation FLASH microcontrollers feature in-system programming and single voltage FLASH memory. This technology allows programming the microcontroller without removing it from the printed circuit board. However, it requires some additional components on the printed circuit board and/or

operator intervention to program the part. The additional components are typically used for entering/exiting the FLASH programming mode.

Increased Flexibility, Shorter Time-to-Market

In-System programming brings some additional benefits into the game. First of all, the parts can be soldered onto the board and the boards can be completely assembled, tested and put on the shelf even before software development is complete. It is now possible to program and re-program the microcontroller anytime, for example at the end-of-line in production or even just shortly before the board is shipped out. This adds a new level of flexibility to your production line, parts procurement and warehouse logistics. This approach also significantly cuts down your time-to-market, as with both OTP (one-time-programmable) and ROM you typically needed to have the programmed part before you started assembly of your boards. For ROM in particular it meant a 6...12 weeks time delay between the time you finish your software and the time you got your ROMed parts delivered.

A short-coming of the in-system approach is that no remote program memory updates are possible or are only possible with substantial additional control logic (like another microcontroller). Programming of the FLASH is only possible within a very narrowly defined voltage and temperature range. For example, your application might be running at 3V, but programming the FLASH is only possible at 5V.

Generation 3: Self-Programming FLASH
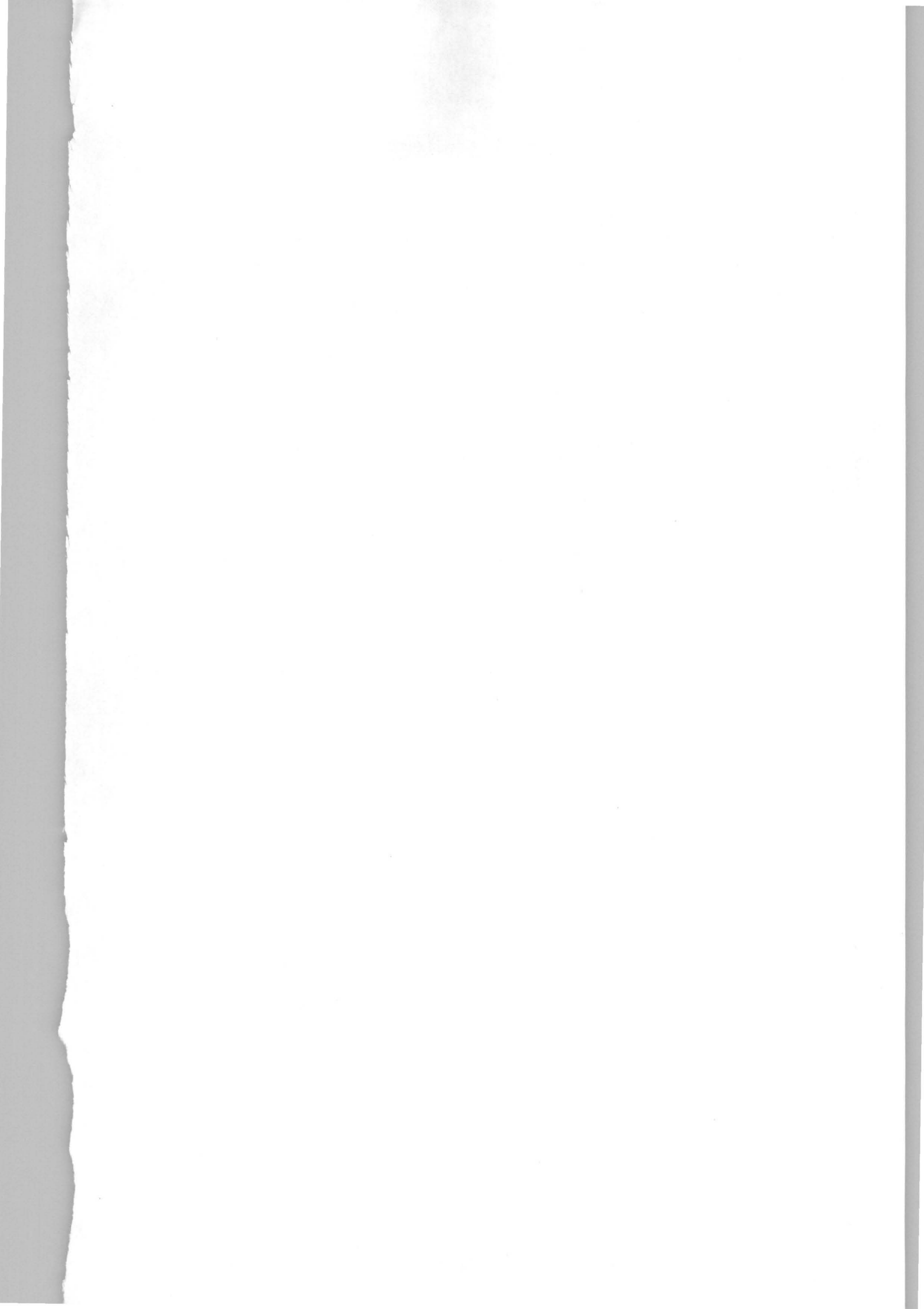
The latest generation of FLASH microcontrollers offers all of the other generations' benefits plus some more. It features what is often referred to as "self-programming" capability, some manufacturers however also call this "true in-system programming"- so it can be easily confused with the second generation technology. Self-programming capability is achieved by adding either a dedicated ROM, FLASH or SRAM boot memory to the microcontroller. When executing the boot memory code the microcontroller can receive new code via a serial communication channel (UART, SPI, USB, CAN, etc.) and program that code into its own FLASH program memory. A separate boot memory is necessitated by the fact that you cannot read (necessary to execute the FLASH programming code) and write (necessary to program) to the same physical FLASH memory at the same time. Self-programming usually works over the entire specified voltage and temperature range of the microcontroller. [5]

Eliminate External/Embedded EEPROM and Save Board Space

Another benefit of self-programming is that applications, which so far used EEPROM to store large data tables that are rarely modified, can now use the self-programming FLASH memory to store those tables – saving the cost of external or embedded EEPROM and reducing board space. [5]

2.1.2 Types of Boot Memory:

There are three approaches to implementing boot memory on microcontrollers: FLASH, ROM and SRAM memory. Of those 3 clearly

9

SRAM is the most "dangerous" choice for remote upgrades, as the FLASH programming code has to be loaded from FLASH into SRAM and then the FLASH is being erased. Imagine what happens if you have a power failure in the middle of updating your FLASH? Your programming code in the SRAM is gone and your FLASH is already erased – not recommended for remote updates. This potential hazard of volatile SRAM is overcome by using either non-volatile FLASH or ROM boot memory. The advantages of having a FLASH boot block versus ROM is that you can change the code to implement different communication channels for FLASH programming, whereas with a ROM boot code you are usually restricted to a single, manufacturer pre-determined programming interface. ROM offers an added level of security and reliability as it can not be accidentally erased or corrupted. So the choice comes down to flexibility versus security and reliability. There are however, workarounds for the inflexibility of boot ROM. Some manufacturers let you put your own programming routines into the standard FLASH program memory, which then jump to the boot memory's write routines to program the FLASH. This way you can still implement programming via your communication interface of choice, but it requires greater care when modifying the FLASH memory so you don't accidentally erase your own programming routines. [5]

FLASH Endurance: Now another characteristic of FLASH becomes important: The minimum guaranteed number of FLASH erase/write cycles. When you're using FLASH to store program information only, 1000 erase/write cycles is plenty, but if you want to store non-volatile data that needs to be updated frequently, you might want more. FLASH micros on the

10

market today offer anything between 1000 erase/write cycles to over 100 000 erase/write cycles. There is also plenty of creative marketing at work in specifying those numbers, so read those datasheets carefully. Some manufacturers specify their "100k" erase/write cycles only at room temperature and in the fine print this number goes down to 10k over temperature, while others guarantee 100k cycles over the entire temperature range. [5]

2.2   HISTORICAL BACKGROUND

Traditionally, homes have been wired for four systems: electrical power, telephones, TV outlets (cable or antenna), and a doorbell. Typically, components and wiring for these are kept within a closet, power metering box or a patch panel. [2]

A remote control for moving vessels and vehicles was first patented by Nikola Tesla in 1898[1], and he announced wireless control of boats in 1909. With the invention of the electronic microcontroller and the widespread uptake of digital communication technology, the cost of electronic control fell rapidly and reliability improved. Remote and intelligent control technologies were adopted by the building services industry and appliance manufacturers worldwide, as they offer the end user easily accessible and/or greater control of their products.

Home automation (also called **domotics**) designates an emerging practice of increased automation of household appliances and features in residential dwellings, particularly through electronic means that allow for things

11

impracticable, overly expensive or simply not possible in recent past decades. The term may be used in contrast to the more mainstream "building automation", which refers to industrial uses of similar technolgy, particularly the automatic or semi-automatic control of lighting, doors and windows, Heating, Ventilation and Air Conditioning, and security and surveillance systems.

The techniques employed in home automation include those in building automation as well as the control of home entertainment systems, houseplant watering, pet feeding, changing the ambiance "scenes" for different events (such as dinners or parties), and the use of domestic robots.

Typically, it is easier to more fully outfit a house during construction due to the accessibility of the walls, outlets, and storage rooms, and the ability to make design changes specifically to accommodate certain technologies. Wireless systems are commonly installed when outfitting a pre-existing house, as they obviate the need to make major structural changes. These communicate via radio or infrared signals with a central controller.

2.3 Examples of Activities of Home Automation

2.3.1 HVAC

Heating, Ventilation and Air Conditioning (HVAC) solutions include temperature and humidity control. This is generally one of the most important aspects to a homeowner. An Internet-controlled thermostat, for example, can both save money and help the environment, by allowing the homeowner to control the building's heating and air conditioning systems remotely.

12

2.3.2 Lighting

Lighting control systems can be used to control household electric lights in a variety of ways:

Extinguish all the lights of the house

Replace manual switching with Automation of on and off signals for any or all lights

- Regulation of electric illumination levels according to the level of ambient light available

- Change the ambient colour of lighting via control of LEDs or electronic dimmers.

2.3.3 Natural lighting

Natural lighting control involves controlling window shades, LCD shades, draperies and awnings. Recent advances include use of RF technology to avoid wiring to switches and integration with third party home automation systems for centralized control.

2.3.4 Security

Control and integration of security systems.

With Home Automation, the consumer can select and watch cameras live from an Internet source to their home or business. Security cameras can be controlled, allowing the user to observe activity around a house or business

right from a Monitor or touch panel. Security systems can include motion sensors that will detect any kind of unauthorized movement and notify the user through the security system or via cell phone.

This category also includes control and distribution of security cameras (see surveillance).

- Detection of possible intrusion
- Sensors of detection of movement
- Sensors of magnetic contact of door/window
- Sensors of glass breaking
- Sensors of pressure changes
- Simulation of presence
- Detection of fire, gas leaks, water leaks (see fire alarm and gas alarm).
- . Medical alert. Tele assistance.
- Precise and safe closing of blinds.

Using special hardware, almost any device can be monitored and controlled automatically or remotely. Including:

- Coffee pot
- Garage door
- Pet feeding and watering
- Plant watering
- Pool pump(s) and heater.

2.4   PREVIOUS WORKS

Home Automation due to its importance to people in saving cost, time and security has made many Engineers, Technologist and Scientist to conduct various types of researches and experiment on it. Below is the list of some works that has been as regards Home Automation:

Delgado, Picking, and Grout (2006) consider the problems with the implementation of home automation systems. Furthermore the possible solutions are devised through various network technologies. Several issues affecting home automation systems such as lack of robustness, compatibility issue and acceptability among the old and disabled people are discussed. [6]

Ciubotaru-Petrescu, Chiciudean, Cioarga, and Stanescu (2006) present a design and implementation of SMS based control for monitoring systems. The paper has three modules involving sensing unit for monitoring the complex applications. A processing unit that is microcontroller and a communication module that uses GPRS modem or cell phone via serial port RS-232. The SMS is used for status reporting such as power failure. [7]

In their paper, Conte and Scaradozzi (2003) view home automation systems as multiple agent systems (MAS). In the paper home automation system has been proposed that includes home appliances and devices that are controlled and maintained for home management. The major task is to improve performance. [8]

In their paper, Alkar and Buhur (2005) propose an Internet Based Wireless Home Automation System for Multifunctional Devices. This paper proposes a

15

low cost and flexible web-based solution but this system has some limitations such as the range and power failure. [9]

Murthy (2008) explores primary health-care management for the rural population. A solution proposes the use of the mobile web-technologies providing the PHC services to the rural population. The system involves the use of SMS and cell phone technology for information management, transactional exchange and personal communication. [10]

Jawarkar, Ahmed, Ladhake, and Thakare (2008) propose remote monitoring through mobile phone involving the use of spoken commands. The spoken commands are generated and sent in the form of text SMS to the control system and then the microcontroller on the basis of SMS takes a decision of a particular task. [11]

Potamitis, Georgila, Fakotakis, and Kokkinakis, G. (2003) suggest the use of speech to interact remotely with the home appliances to perform a particular action on behalf of the user. The approach is inclined for people with disability to perform real-life operations at home by directing appliances through speech. Voice separation strategy is selected to take appropriate decision by speech recognitions. [12]

CHAPTER THREE

3.0 DESIGN AND IMPLEMENTAION

The DTMF home power management device was designed around the following subsystems:

I.   5-volt system power supply.

II.   MT-8880 DTMF transceiver.

III.   24C04 512-byte memory.

IV.   AT8952 8-bit microcontroller.

V.   4-normally open 6v, 10A contact current relay.

3.1   SYSTEM POWER SUPPLY

The system was powered by a regulated 5v-Dc obtained from AC mains via a 12-volt, 0.5A step-down transformer. The low voltage AC was rectified into a pulsating DC voltage of amplitude given by the relation:

$$V_{ac} = V_{rms}\sqrt{2} - 1.4$$

V_{rms} = RMS secondary Voltage of the transformer.

$\sqrt{2}$ = RMS to peak scaling factor

1.4 = voltage drop across two diodes with bridge rectifier.

For a 12v Ac input, $V_{dc} = 12\sqrt{2} - 1.4 = 15.5$

This DC voltage was smoothened by a capacitance evaluated from the expression:

$C = It / V$

I= load current

t=1 /f= 1/2f (full wave bridge rectifier)

v = peak to peak Ac ripple voltage.

The load current was evaluated from the summation of the total current drawn by the different subsystem.

LED= 10mA

MT-8880 = 10mA

24C04= 5mA

89C52= 10mA

Relays = 4* 30mA= 120mA
_____

Total=   155mA

The peak to peak ripple voltage was deduced by the determining the minimum input voltage required by the 5-volt regulator to maintain regulation.

For a regulated 5-volt output, the input voltage must be greater than 7v.

On a 15.5v DC input, this translates into a maximum peak to peak voltage of :

15.5v –7v = 8.5v.

Inserting these parameters into the equation yields:

$$C = \frac{\left[0.155A*\left(\frac{1}{2}*50Hz\right)\right]}{8.5v} = \frac{0.00155}{8.5v}$$

The above value of capacitance is the minimum value required to guarantee system operation under worst scenario. The value was amplified to 2200μF at a capacitor working voltage of 25v.

The smoothened DC was fed into a 7805, 5-volt regulator and then stabilized by a second 2200μF capacitor before driving the circuit.
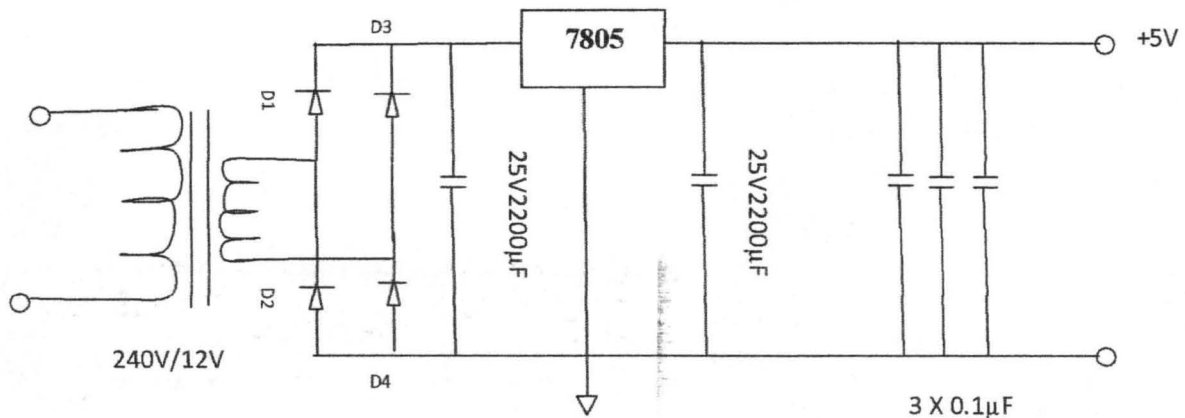
FIG 3.1: SYSTEM POWER SUPPLY UNIT

3.2 MT-8880 DTMF transceiver

To decode the DTMF tones received over the mobile phones headset, a dtmf decoder was required. An MT-8880 component was chosen as it could transmit as well as received. [13] The part has the pin out shown in FIG 3.2

19
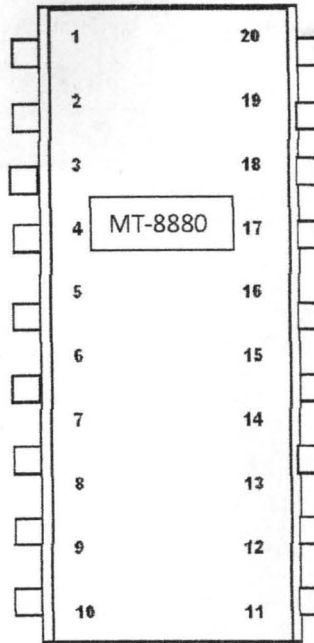
FIG3.2: DUAL TONE MUTI-FREQUENCY (DTMF) TRANSCIEVER

PINOUT

PIN DEFINITION:

1: Non-inverting op-amp input   6: DTMF Oscillator   11: Register select

2: Inverting op-amp input   7: Clock Input   12: System Clock Input

3: Gain select   8: Tone output   13: Interrupt request

4: Reference voltage   9: Read/Write   14-17: Microcontroller

data bus

5: Ground Input   10: chip select   18: Early Steering

19: Steering input/Guard time   20: Positive power supply input

The transceiver was configured for operation by writing specific bit pattern to the two control register CRA and CRB via software.

The device was connected to a 3.58MHz crystal to guarantee an error free to detection. The device was interfaced with the system controller via port 1 as shown in FIG 3.3.

FIG 3.3: MT-8880-MICROCONTROLLER INTERFACE

Three I/O pins were dedicated to generating the Read/Write(R/W), Chip Select (CS) and Register Select (RS) control strobes. The transceiver 4-bit bus was interfaced with I/O pins of the microcontroller (P1.0-P1.3).

To reduce coding complexity, the transceiver was used in the polled mode to check for the prescence of a new tone pair. If a valid DTMF signal is thus received the valve is read and stored in the microcontroller data memory.

21

3.3　　24C04 512-byte EEPROM

For retention of load states and homeowner password, an EEPROM device was incorporated in the design. A 512-byte I^2C EEPROM device was used as this was the smallest capacity device available at design time. [16]

The two-wire interface was emulated in software by bit-banging the serial data over P3.5 and P3.6 as SCL and SDA respectively.

Five memory locations were utilized by the Firmware to hold a 4-digit PIN and a 1-byte load status data. At system power-up, these data are loaded from memory and placed in data memory for easy access. [14]

The 24C04 device was hardwired for write access at slave address 0XA0, and read access at slave address 0XA1.
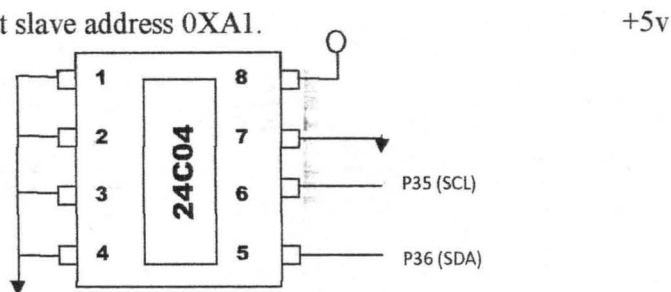
FIG 3.4: EEPROM/CONTROLLER INTERFACE

3.4　　AT89S52 8-bit Microcontroller

To coordinate system activities and endowed flexibility, a microcontroller driven design approach was adopted. An easily available microcontroller-AT89S52 part was used. The device was programmed to execute a customized firmware that retrieved data from the transceiver, compared the received data

with stored data and take appropriate actions based on the output of the pre-programmed decisions. [15]

The microcontroller was interfaced to the different system components as diagrammed below:
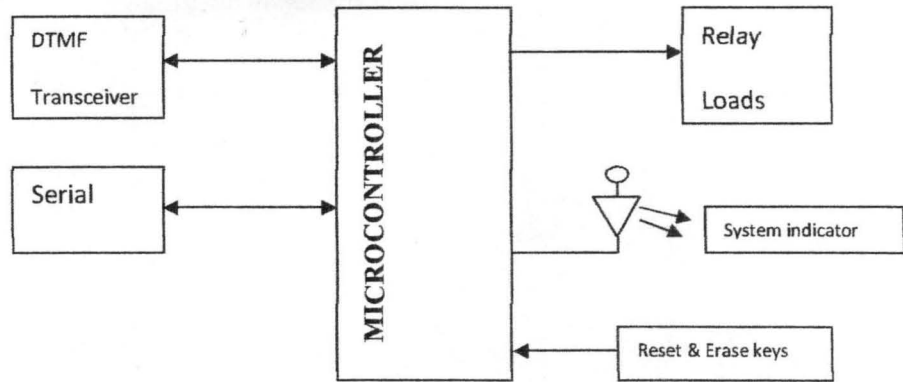
FIG3.5: Controller-System Interface

The microcontroller was interfaced with the MT-8880 over Port 1, the serial memory over Port 3, and the relay switches over port 0. An LCD debug output was provided on Port 2 to facilitate the development process.

Two N/O push buttons were also corrected to the microcontroller. The two buttons (RESET and CLEAR) enable system re-initialization in the event of a password loss.

Activating the re-initialization sequences overwrites the user pin to default pin of "1234" ; while also turning the four switched output off.

3.5   RELAY SWITCHES

To control loads, four normally-open relays used. The relays were driven by PNP driver transistors as shown in FIG 3.6.

23
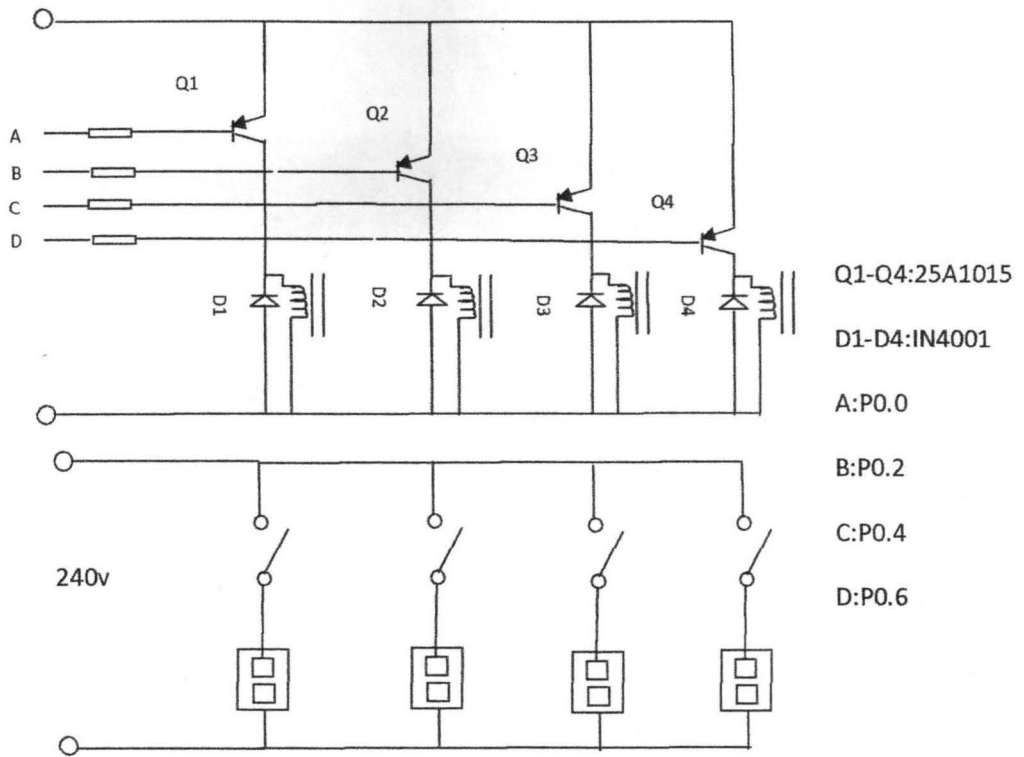
FIG 3.6: RELAY POWER SWITCH

The transistors were biased on by resistances calculated from the relation:

$R_B = V_B - V_{BE} / I_B.$

$I_B = I_C / H_{FE.}$
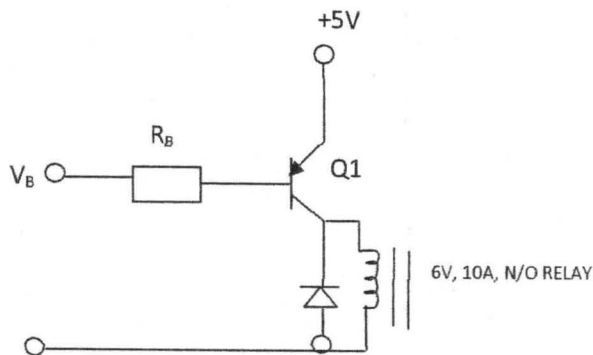
FIG 3.7: A 6V, 10A, N/O RELAY

RELAY DRIVER BIASING CALCULATION

24

The relays used had a coil resistance of 150Ω. On a 6v coil voltage, the relay current is thus:

$6v/150Ω = 40mA$.

However, the relays were run on 5v slightly reducing the current to 33mA.

The driver transistors have a DC gain of about 100

Therefore, $I_C = I_{COIL} = 33mA$

$I_B = I_C / H_{FE} = 0.033 / 100 = 0.000033A = 330\mu A$

$R_B = V_B - V_{BE} / I_B = 5 - 0.7 / 0.00033 = 4.3 / 3.3*10^{-4} = 1.3 * 10^4 Ω$

The value was reduced by a factor of two to guarantee relay switching under reverse conditions. The transistors were protected by free- wheeling diodes against inductive kick backs generated by the relay's de-energising.

3.6   CONTROL SOFTWARE

The system control software was modulated for easy testing and correction of faulty modules. The basic execution flow is detailed below:

i.   Initialize system

ii.   Restore loads status from memory and energize or de-energize as indicated by the load status control byte.

iii.   Poll MT 8880 for DTMF tones'

iv.   If DTMF tone is received, execute user command.

v.   Go back to (iii).

25

Five commands were implemented over the air interface. These commands are shown in the table below:

TABLE 3.1: Commands, types of operation to be carried out and the action taken.

Command	Type	Action
PPPP10#	LOAD	TURN OFF
PPPP11#	CONTROL	LOAD 1
	LOAD	TURN ON
	CONTROL	LOAD 1
PPPP20#	LOAD	TURN OFF
PPPP21#	CONTROL	LOAD 2
	LOAD	TURN ON
	CONTROL	LOAD 2
PPPP30#	LOAD	TURN OFF
PPPP31#	CONTROL	LOAD 3
	LOAD	TURN ON
	CONTROL	LOAD 3
PPPP40#	LOAD	TURN OFF
PPPP41#	CONTROL	LOAD 4
	LOAD	TURN ON
	CONTROL	LOAD 4
PPPPZZZZ#	PASSWORD	CHANGE
	CONTROL	PASSWORD

FLOW CHART
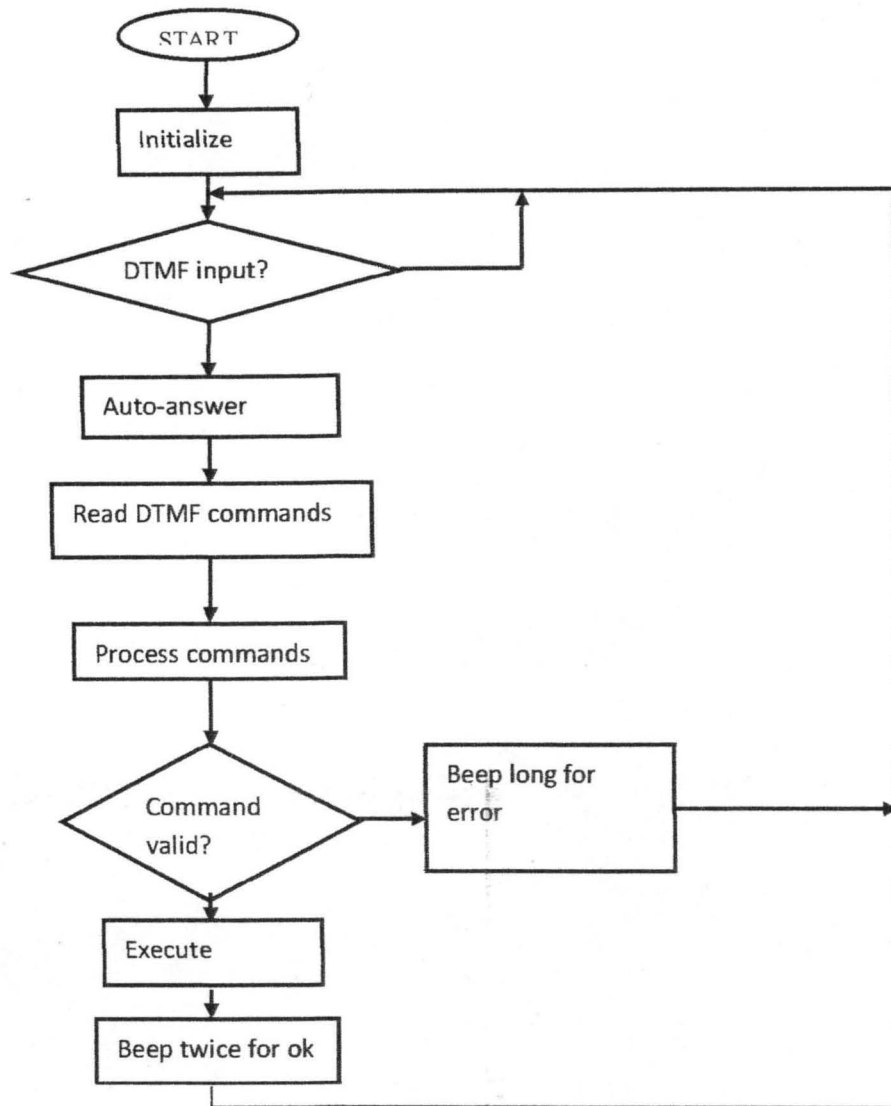
FIG 3.8: FLOW CHART FOR THE MODE OF OPERATION OF THE

DEVICE

The firmware was prepared to return audio beeps to the human operator via the microphones in the headset, but the mobile phone could not send the beeps back to the operator.

3.7   SYSTEM INDICATOR

A light emitting diode (LED) was connected to the microcontroller to give a visual display of system state. At power up, the LED is pulsed three times if there is a memory access error; thereafter the LED is continuously turned ON.

3.8   SYSTEM IMPLEMENTATION

The physical realization of the project was of uttermost importance considering its applicability. After carrying out the designing of the different modules, the circuit diagrams had to be converted to a working project. The AT89S52 microcontroller was programmed and then implementation firstly carried out on a breadboard. This helps to make changing of circuit components easy while attempting to obtain the desired result. The power supply was derived from the bench power supply. Stage by stage testing was done on the breadboard before carrying out the final soldering on the veroboard. After soldering on the veroboard, stage by stage testing was also done to ensure that the circuit was still functioning as desired and no error in connection had been made during the soldering process.

In carrying out this project, each module in the circuit diagram was attended to one after the other.

3.8.1 FACTORS TO BE CONSIDERED TO IMPLEMENT THE SYSTEM

1) The receiver must reside in a location where a signal with sufficient strength can be obtained.

2) The only person who can communicate with the control module is the person who will be successfully authenticated.

3) Only devices with electrical controlling input ports will be possible targets for controlling.

4) The receiver must have a power source (12V) attached at all times.

5) Operation of the controlling unit is only possible through a cell phone with DTMF capabilities and voice message.

This makes the implementation of this project to be in various stages which are discussed below:

3.8.1.1 Testing the mobile phone to be used

For successful operation of this project, a GSM phone with enhancement that has the ability to pick up calls automatically (auto-answer) is required. This is because the project does not require anybody to pick up the call before it carries its necessary objectives.

However, a nokia phone and a hands-free were bought to do the work. The hands-free is connected to its port on the phone and a call was initiated from another GSM phone to ascertain for its auto-answer capability.

3.8.1.2 Programming the microcontroller

Programming varies from one microcontroller to another. Eight-bit microcomputers can be programmed using binary or hexadecimal numbers (machine language) and semi-English language statements (assembly language). Regardless of what type of language is used to write the programs, the microcontroller understands only binary numbers. Thus, all programs must eventually be translated into their appropriate binary forms. The assembly language was used in the programming of the microcontroller for this project and the program for the microcontroller is in Appendix.

29

3.8.1.3 The bread board stage

The circuit components were then arranged on the breadboard accordingly. Proper and adequate care was taken to make sure that the configuration suits the circuit and gives the expected output. This stage helps to make changing of circuit components easy while attempting to obtain the desired result. At this stage, measurement of various parameters like voltage drops, input impedances, base current, etc, were made so as to ensure accurate results. Stage by stage testing was done before carrying out the final soldering on the veroboard.

However, after all modules have connected and arranged on the breadboard. The mobile phone is connected through the hands-free for testing.

3.8.1.4 SOLDERING ON THE VEROBOARD

The components were soldered onto the veroboard in accordance with the working circuit on the breadboard. This was done with the use of a soldering iron and soldering lead. Soldering on the veroboard was a task that required a lot of time and accuracy. Since all the holes are connected, every connection that was not needed had to be etched out using sand paper. It was also ensured that the solder did not spill over onto other connection point on the veroboard that could result in a short circuit.

A COMPLETE CIRCUIT DIAGRAM
OF A DUAL TONE MULTI-
FREQUENCY (DTMF) HOME
POWER MANAGEMENT DEVICE
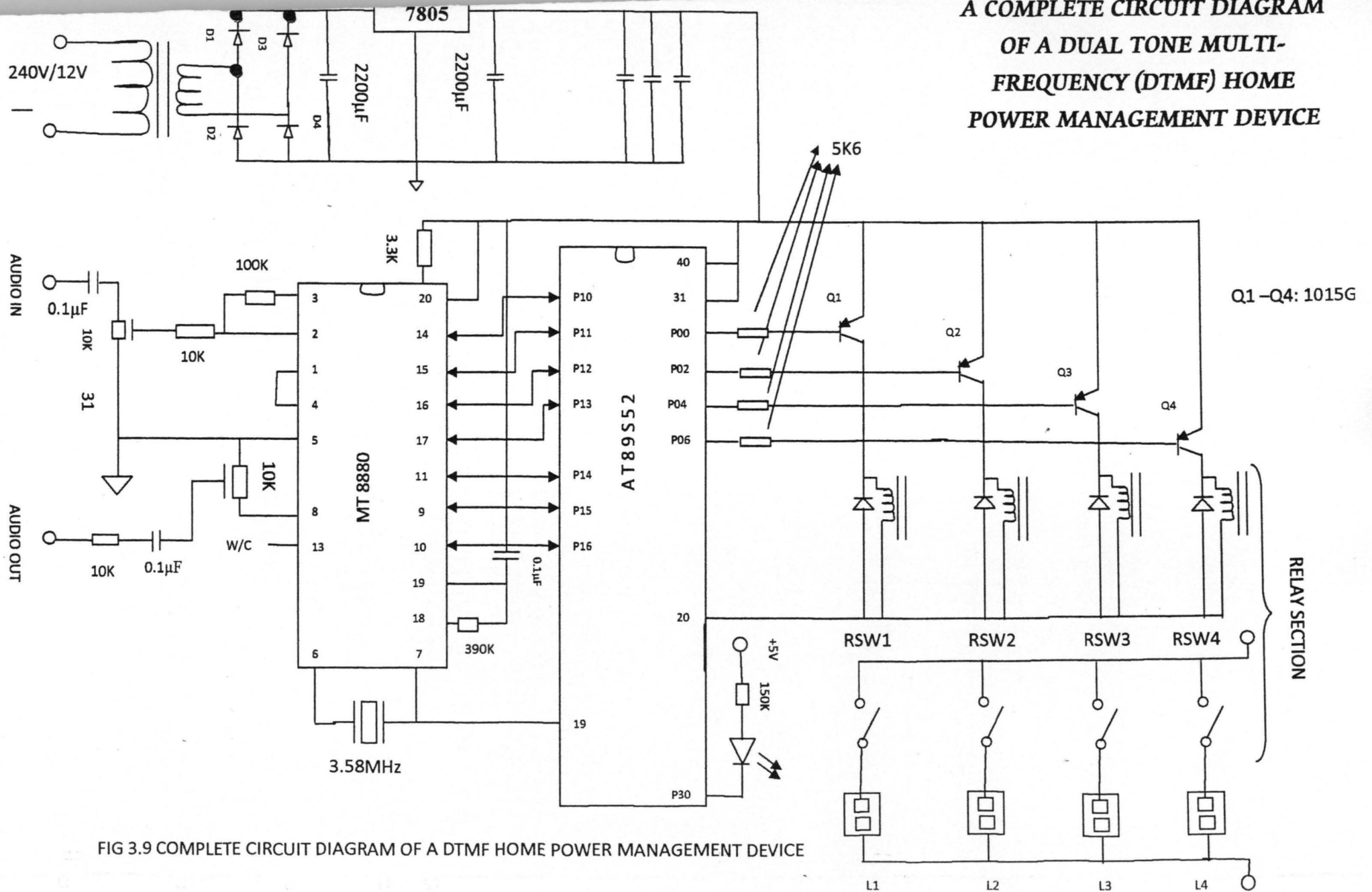
FIG 3.9 COMPLETE CIRCUIT DIAGRAM OF A DTMF HOME POWER MANAGEMENT DEVICE

CHAPTER FOUR

4.0 TESTS, RESULTS AND DISCUSSION

4.1 TESTING

While implementing the project and after implementation, several tests were carried out to ascertain the actual capability of the final project and ensure that the construction meets the desired specification and design concept. The process of testing at various stages involved the use of a multimeter to measure the various electrical parameters such as voltage, current and resistance values of the components and their electrical status and also to measure continuity among the components. The following tests were carried out:

a. Unit test: testing a single component. Each single component required for the circuit design was tested.

b. Integration test: testing a combination of different components.

c. System test: testing the system as a whole. The entire system at the end of soldering was tested to make sure that it conformed to desired objective.

d. User acceptance test: is performed to check if the requirements have been met.

The user acceptance test was carried out after soldering and casing. The remote GSM phone was used to call the stationary GSM phone. After the first ring, the stationary GSM phone auto-pick the call. Thereafter, the commands were being issued and the respective devices connected to each socket were controlled without touching any of the devices.

32

However, further test were conducted to see if the device can control more than one device at a time but one after the other.

4.2 RESULTS

Below is the summary of the result from the test carried out on the whole device after soldering.

NOTE:

is to terminate command

1234 is a default pin/password, NNNN is for new password

1=ON, 0=OFF

TABLE 4.1: SHOWING THE RESULTS OF COMMAND CARRIED OUT

	COMMAND	RESULTS
SOCKET 1	123410#	NO OUTPUT
	123411#	OUTPUT
SOCKET 2	123420#	NO OUTPUT
	123421#	OUTPUT
SOCKET 3	123430#	NO OUTPUT
	123431#	OUTPUT
SOCKET 4	123440#	NO OUTPUT
	123441#	OUTPUT
PASSWORD	1234NNNN#	PASSWORD SUCCESSFULLY CHANGED

MULTIPLE CONTROL	123410#1234 21#123441#	SOCKET 1 IS OFF WHILE SOCKET 2and 4 IS ON

Pictures of the actions carried out by the DTMF HOME

POWER MANAGEMENT DEVICE

(a)
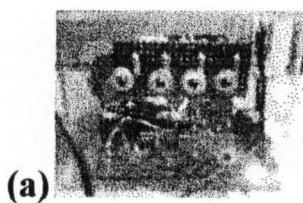
The green LED indicates that the device is ON and the red LEDs shows that no load is ON.

(b)

A Load is connected to the device. (multi-purpose charger).

(c)

The multi-purpose charger connected is charging the battery connected to it with supply from the device.

(d)
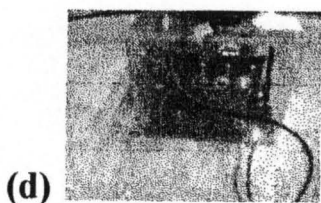
The red LED indicates that one of the socket is swithed ON.

(e)
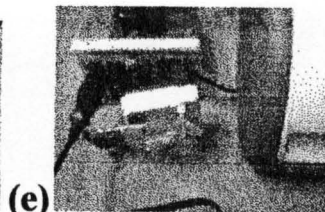
For muti-appliance control, a printer is added to the multi-purpose charger.Status=OFF

(f)

Now, the two appliances are on ON state
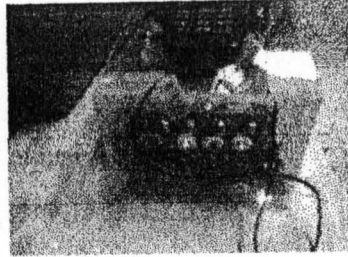
(g) The two red LEDs are inicating that two sockets are already switched ON.
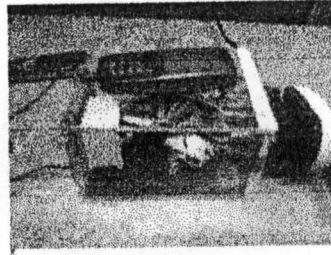
(h) Side View of the device

FIG 4.1(a) - (h): Pictures of actions performed with the DTMF

4.3   DISCUSSION OF RESULTS

From the table above, there will be no output when this kind of command is given:

PASSWORD SOCKET TO BE CONTROLLED AND THE INTENDED STATUS #. For this kind of command, one can choose to use the default password or change the password before initiating any command. The socket could be 1, 2, 3 or 4 and the intended status could either be ON (1) or OFF (0) as the case may be. The # is to tell the microcontroller that this is the end of the major command.

Nevertheless, the essence of the password is to make sure that it is only the homeowner that control the device not anybody or everybody. So in changing the from either the default password or the old password to the new password, the command is like: **password (old or default) new password#.** This command is similar to the previous control command discussed. It is just the action to be performed that is different.

Conversely, there are other test that were carried out. There results are shown in Table 4.2.

TABLE 4.2: RESULTS OF COMPONENT TEST

TEST	RESULTS
Transformer primary was connected to the mains and readings were taken from its output (Transformer Secondary) via a multimeter	11.75Vac
The power supply units	4.89VDc
Socket Outlet	220V Ac
Inputs to the normally-open relay	5.7V

4.4   SHORTCOMINGS/LIMITATIONS

1. The major shortcoming of this project work is that it is dependent on the GSM network. So if the device is to be used in areas that are of poor network coverage, there will be problem in carrying out the objectives of this project.

2. This project is likely to have problem when there is power outage as it depend solely on power supplied by the power authority of the country.

3. Moreover, commands to control each device can only be entered one after the other and not at the same time that is a command to control two devices at the same time.

4.   This device can only control but cannot monitor the status of home appliance.

5.   This device can only control four appliamces.

4.5   REMEDIES

However, base on the shortcomings that are listed above. The following remedies may be put into consideration:

1. Since the device is not configured to use a particular network, any network can be used with the stationary mobile phone. So, the problem of poor network can be settled by making use of the very best network in any locality that the project will be used.

2. The problem of power outage can be corrected by connecting the stationary mobile phone directly to the power supply. Also, a charging circuit can be constructed alongside with the device to charge the circuit.

3. Since the microcontroller is a Flash In-system programming microcontroller. The microcontroller can be re-programmed to take in more than one command to control more than one device at the same time.

4. The number of home appliance that can be controlled by this device can be increased by modifying the programme written into the microcontroller to control more devices. As the microcontroller has 32-input/output pins (4ports with 8-pins per port), more devices can be controlled.

CHAPTER FIVE

5.0 CONCLUSION AND RECOMMENDATION

5.1   SUMMARY OF WORKDONE

All that has been written since the beginning of this report is to elaborate on what DTMF home power management is all about. Also, the methodology and problems encountered in carrying out this project has been discussed.

Moreover, the necessary calculation that gave rise to the value of each component used can also be found in this project report.

5.2   PROBLEMS ENCOUNTERED

1. Since the device requires a GSM mobile phone with auto-answer capability which also requires that a headset or hands-free be connected to the GSM mobile phone. It was difficult to get a headset or hands-free that can do the job.

2. Due to piracy of commodities in Nigeria, most of the headset or hands-free gotten were found not to follow the colour code for cables that made up the connections in the headset or hands-free. This makes it difficult for connection of the headset or hands-free from the GSM mobile phone to the device.

3. The problem of undependable and irregular power supply is also a major difficult as it made soldering of components that made up the device to be delayed and uneasy.

5.2   POSSIBLE IMROVEMENTS/ADDITIONS

1. The number of devices to be controlled can be increased by modifying the sets of instruction written into the microcontroller to control more devices.

2. The device can also be used to monitor status of home appliances in such a way that the state of any home appliance whether ON or OFF can be known remotely.

3. It can also be made to prevent intrusion at homes, fire out-break and any other factor that can militate against home security. This can be done by making sure that sensors are installed at every spot that is needed protected against any form security breach. So that when there is security breach, the input can be send to the device and device sending tone to indicate security breach to the home owner.

5.3 CONCLUSION

Mobile phones have become an indispensable part of our life. Our system uses a microcontroller and a GSM mobile phone for its operations. The systems can be used as a test bed for any application that requires on-off switching based applications. Wireless controlled home appliances in the comforts of any environment will revolutionize our way of living.

In this project low cost, secure, ubiquitously accessible, remotely controlled solution for automation of homes has been introduced. The approach discussed in this project is novel and has achieved the aim of controlling home appliances remotely using DTMF tones generated from GSM mobile phone while dialling digits on the GSM mobile phone, satisfying user needs and requirements. Implementing this project in the home provides home owners with the privilege to automate their home so that they can take advantage of the technological advancement available today. In addition, automating one's home is an excellent way to, cut energy costs, and gain increased control over

the surroundings. Home automation is an inexpensive project, and the much usefulness is only limited by one's imaginations.

This project has proven to be very interesting and challenging. It has helped to bring together all the theoretical knowledge gained during these years and an appreciation for the practicality. It has also helped in getting more informed about the world of embedtronics and domotics.

5.5   RECOMMENDATION

Based on the conclusion, importance and the advantages of DTMF home power management, it is advised that every house should have this device installed in their various apartment in order for them to be able to manage (control and monitor) their home appliances and also to prevent their home against intrusion from strangers and any form of security breach.

It is also advised that with necessary modification in this device, it can be used in other places apart from home for security and control purposes. So, Organisation, Security companies, industries and establishment can still make use of this device base on the requirements that needed to be met.

REFERENCES

1. A journal of The Leading Global Real-World Media and Entertainment Company.Copyright 2010 Discovery Communication Inc.

2. Wikimedia Foundation Incorporation (2010). A discussion on Home automation.

3. Wikimedia Foundation Incorporation (2010). A discussion on Microcontroller.

4. Heath, Steve (2003). Embedded system design._EDN series for design engineers (2 ed.). Newnes. pp. 11–12.

5. 2002-2008 Microcontroller Pros Corporation. The Evolution Of Flash Microcontrollers

6. Delgado, A. R., Picking, R., & Grout, V. (2006) Remote-controlled home automation systems with differentnetwork technologies. *Proceedings of the 6th International Network Conference (INC 2006)*, University of Plymouth, 11-14 July 2006, pp. 357-366. Retrieved from http://www.newi.ac.uk/groutv/papers/p5.pdf

7. Ciubotaru-Petrescu, B., Chiciudean, D., Cioarga, R., & Stanescu, D. (2006). Wireless Solutions for Telemetry in Civil Equipment and Infrastructure Monitoring. *3rd Romanian-Hungarian Joint Symposium on Applied Computational Intelligence (SACI)* May 25-26, 2006. Retrieved from : http://www.bmf.hu/conferences/saci2006/Ciubotaru.pdf

8. Conte, G., & Scaradozzi, D. (2003). Viewing home automat ion systems as multiple agents systems. *RoboCUP2003*, Padova, Italy. Retrieved from : http://www.robosiri.it/ROBOCUP_2003/ROBOCUPSITOSIRI/articles/pdf/Conte.pdf

9. Alkar, A. Z., & Buhur, U. (2005). An Internet Based Wireless Home Automation System for Multifunctional Devices. *IEEE Consumer Electronics, 51*(4),1169-1174.Retrieved from:http: //www.thaieei.com/embedded/pdf/Automation/20022.pdf

10. Murthy, M. V. R. (2008). Mobile based primary health care system for rural India. *W3C workshop on Roleof Mobile Technologies in Fostering Social Development*, June, 2008.

11. Jawarkar, N. P., Ahmed, V., Ladhake, S. A. & Thakare, R. D. (2008). Micro-controller based Remote Monitoring using Mobile through Spoken Commands. Journal Of Networks, 3(2), 58-63. Retrieved from : http://www.academypublisher.com/jnw/vol03/no02/jnw03025863.pdf

12. Potamitis, I., Georgila, K., Fakotakis, N., & Kokkinakis, G. (2003). An integrated system for smart-home control of appliances based on remote speech interaction. *EUROSPEECH 2003, 8th European Conference on Speech Communication and Technology*, pp. 2197-2200, Geneva, Switzerland, Sept. 1-4, 2003. Retrieved from http://www.wcl.ee.upatras.gr/ai/papers/potamit is14.pdf

APPENDIX

```c
#include <reg52.h>


//#define debug

#include <intrins.h>

#define xtal 3580000UL

#define dx_port1 P0

#define dx_port2 P2

#define tone_port P1

//************************

//************************

#define max_cmd 15

#define T0_reload 0x06

#define cmd_stop '#'

#define cmd_start '*'

#define t_base 250

#define sec_1 1193

#define timeout_count sec_1*5

#define write_delay 3000UL

#define max_char 16

//***********************

//***********************

#define slave_address 0xa0

#define write_flag 0x00

#define read_flag 0x01

#define pin_lenght_size 4

#define pin_address_start 0x00

#define presets_address pin_address_start+pin_lenght_size

//***********************
```

```c
//***********************
#define load_ctrl 8
#define pin_change 10
#define max_load '4'
#define min_load '1'
#define pin_start 1
#define pin2_start 6
#define load_off '0'
#define load_on '1'
#define load_1 '1'
#define load_2 '2'
#define load_3 '3'
#define load_4 '4'


#define freq_reload 184
//**********************
//*********************
sbit sda= P3^7;
sbit scl=P3^6;
sbit tone_dx=P3^0;
#define freq_dx tone_dx
sbit load1_dx =P2^0;
sbit load2_dx =P2^3;
sbit load3_dx =P2^6;
sbit load4_dx =P0^6;
sbit error_led =P3^5;
sbit reset= P3^5;
sbit tone_in=P3^2;
//****************
//****************
```

```c
sbit lcd_out = P0^0;

sbit lcd_clock= P0^3;

sbit lcd_load=P0^1;

sbit lcd_en=P0^2;

#define lcd_rs lcd_out
//******************
sfr T2MOD = 0xc9;
//********************
//********************
volatile unsigned int count;

unsigned char data buffer[max_cmd+2];

unsigned char preset,error,timeout;
//*************************
//*************************
code char table[]={"D1234567890*#ABC"};
//*********************
//*********************
unsigned char store_pin(unsigned char data *);

void tone_off(void);
//****************
//****************
void tf0_isr(void) interrupt 1
{
        if(!(--count)){    TR0=0;timeout=1;}
}
//********************
//********************
void start_timeout(void)
{
        TR0=0;
```

45

```c
            TF0=0;

            TL0=T0_reload;

            count=timeout_count;

            timeout=0;

            TR0=1;

}
//****************
//****************
void stop_timeout(void){TR0=0;}
//*****************
//*****************
void delay_ms(void)

{

        start_timeout();

        count=1;

        while(!timeout);            //wait for 250 machine clock tickes here

        stop_timeout();

}
//*****************
//*****************
void write_timeout(void)

{

        unsigned char z=write_delay/t_base;


        while(z){delay_ms();z--;}

}
//********************
//********************
void delay_sec(float sec)                    //maximum of 16 seconds possible here

{
```

```c
        unsigned int z=sec_1;


        z*=sec;

        while(z){delay_ms();z--;}
}
//***************
//***************
void serialize(unsigned char c,unsigned char reg_select)
{
        unsigned char z;


        lcd_load=0;
        lcd_clock=0;
        lcd_en=0;


        for(z=0;z<8;z++)
        {
                lcd_out=!(c&0x80);
                lcd_clock=1;
                lcd_clock=0;
                c<<=1;
        }
        lcd_rs=reg_select;
        lcd_load=1;
        lcd_load=0;
        lcd_en=1;
        lcd_en=0;
}
//*******************
//*******************
```

```c
void lcd_data(unsigned char c)
{
        unsigned char z=250;
        serialize(c,1);
        while(--z);
}
//***************
//***************
void lcd_cmd(unsigned char c)
{
        unsigned char z=250;
        serialize(c,0);
        while(--z);
}
//***************
//***************
void lcd_clear(void)
{
        lcd_cmd(0x01);
}
//***************
//***************
void lcd(char code *ptr)
{
        while(*ptr)lcd_data(*ptr++);
}
//***********************
//***********************
void init_lcd(void)
{
```

```
                    lcd_cmd(0x38);

                    write_timeout();

                    lcd_cmd(0x38);

                    delay_ms();

                    delay_ms();

                    lcd_cmd(0x38);

                    delay_ms();

                    delay_ms();

                    lcd_cmd(0x0c);

                    delay_ms();

                    delay_ms();

                    lcd_cmd(0x01);

                    delay_ms();

                    delay_ms();

                    lcd_cmd(0x06);

                    delay_ms();

                    delay_ms();

        }

//******************

//******************

void dly(void)
                    //

        {

                    _nop_();

                    _nop_();

                    _nop_();

                    _nop_();

                    _nop_();

                    _nop_();

        }
```

49

```c
//*******************

//*******************

void i2c_start(void)

{

        sda=1;

        scl=1;

        dly();

        sda=0;

        dly();

        scl=0;

        dly();

}

//****************************

//****************************

void i2c_stop(void)
                    //
{

        sda=0;

        dly();

        scl=1;

        dly();

        sda=1;

}

//***********************

//***********************

void no_ack(void)                                        //
{

        sda=1;

        _nop_();

        scl=1;
```

```c
        _nop_();

        scl=0;

}

//***********************

//***********************

unsigned char write_byte(unsigned char p)       //

{

        unsigned char d;

        for(d=0;d<8;d++)

        {

        sda=p&0x80;

        _nop_();

        scl=1;

        dly();

        scl=0;

        dly();

        p<<=1;

        }


        sda=1;

        _nop_();

        scl=1;

        _nop_();

        p=sda;

        scl=0;

        return p;

}

//***********************

//***********************

unsigned char read_byte(void)            //
```

```
        {
                unsigned char x,c;

                c=0x00;
                sda=1;
                for(x=0;x<8;x++)
                {
                c<<=1;
                scl=1;
                dly();
                c|=sda;
                scl=0;
                dly();
                }
                return c;
        }
//************************
//************************
unsigned char write(unsigned char address,unsigned char c)      //
{
                i2c_start();
                if(error=write_byte(slave_address|write_flag))goto write_abort;
                if(error=write_byte(address))goto write_abort;
                if(error=write_byte(c))goto write_abort;
                i2c_stop();
                write_timeout();
                return 0;


write_abort:
                i2c_stop();
```

52

```
            return 1;

    }
    //***********************

    //***********************

    unsigned char read(unsigned char address)                        //

    {

            unsigned char data_read;

            i2c_start();

            if(error=write_byte(slave_address|write_flag))goto read_abort;

            if(error=write_byte(address))goto read_abort;

            i2c_start();

            if(error=write_byte(slave_address|read_flag))goto read_abort;

            data_read=read_byte();

            no_ack();

            i2c_stop();

            error=0;

            return data_read;


    read_abort:

            i2c_stop();

            error=1;

            return 0;

    }
    //;***********************

    //***********************

    void set_presets(void)

    {

            dx_port1=0xff;

            dx_port2=0xff;

            if(preset&0x01)load1_dx=0;
```

53

```c
            if(preset&0x02)load2_dx=0;

            if(preset&0x04)load3_dx=0;

            if(preset&0x08)load4_dx=0;

    }
//*******************
//*******************
unsigned char read_dtmf(void)

{

        unsigned char z;


        if(tone_in)                    //if STD low grab data

        {

                z=tone_port&0x0f;

                while(tone_in); //wait until STD deasserted

                return table[z];

        }

        else return 0;

}
//*******************
//*******************
unsigned char get_dtmf(void)

{

        unsigned char data *ptr=buffer;

        unsigned char x=0;

        unsigned char c;


        while(1)

        {

                start_timeout();

                while((!(c=read_dtmf()))&&(!timeout));
```

54

```
if(timeout)
{
        stop_timeout();
        error=1;
        return 0;
}

if(c)
{
        stop_timeout();
        if(c==cmd_stop)
        {
                if(!x)
                {
                        error=1;
                        return 0;
                }
                error=0;
                return x;
        }
        *ptr=c;
        ptr++;
        x++;
        #ifdef debug
        lcd_data(c);
        #endif
        if(x>max_cmd)
        {
                error=1;
                return 0;
```

```
                    }

                }

            }

        }
//******************
//*******************

unsigned char load_presets(void)

{

        preset=read(presets_address);

        if(error)return 1;

        else return 0;

}
//*******************
//*******************

void init_irq(void)

{

        ET0=1;

        ET2=1;

}
//******************
//******************

void init_timer(void)

{

        TCON=0x00;

        TMOD=0x22;

        TH0=T0_reload;

        TL0=T0_reload;


        T2CON=0x00;

        T2MOD=0x00;
```

```c
        RCAP2H=(65536-freq_reload)>>8;

        RCAP2L=(65536-freq_reload);

        TH2=RCAP2H;

        TL2=RCAP2L;

}
//****************
//****************
unsigned char reset_pin(void)
{

        buffer[0]='1';

        buffer[1]='2';

        buffer[2]='3';

        buffer[3]='4';

        if(store_pin(buffer))return 0;

        else return 1;

}
//*******************
//*******************
unsigned char reset_loads(void)
{

        return(write(presets_address,0x00));

}
//******************
//******************
void show_loading(void)
{

        lcd_clear();

        lcd("loading...");

        delay_sec(1);

}
```

```c
//***************
//***************
void show_pin(void)
{
        lcd_clear();
        lcd("PIN: ");
        lcd_data(read(0x00));
        lcd_data(read(0x01));
        lcd_data(read(0x02));
        lcd_data(read(0x03));
        delay_sec(1);
}
//***************
//***************
unsigned char sys_init(void)
{
        unsigned char z=0;
        IE=0x00;
        tone_off();
        init_irq();
        init_timer();
        EA=1;
        preset=0x00;
        if(!reset)
        {
                z=reset_pin();
                z|=reset_loads();
        }
        z|=load_presets();
        #ifdef debug
```

58

```c
            init_lcd();

            show_loading();

            show_pin();

            #endif

            if(!z)set_presets();

            return z;

    }
    //*******************
    //*******************
    void beep_error(void)

    {

            unsigned char z=6;

            error_led=0;

            while(z)

            {

                    delay_sec(1);

                    error_led=!error_led;

                    z--;

            }

            error_led=1;

    }
    //*****************
    //*****************
    void tone_on(void)

    {

            TH2=RCAP2H;

            TL2=RCAP2L;

            TF2=0;

            TR2=1;

    }
```

59

```
//***************

//***************

void tone_off(void)

{
        TR2=0;
}
//****************

//****************

void tf2_isr(void)          interrupt 5

{
        if(TF2)
        {
                freq_dx=!freq_dx;
                TF2=0;
        }
}
//*****************

//*****************

/*

void tone_on(void)

{
        tone_dx=1;
}
//**************

//**************

void tone_off(void)

{
        tone_dx=0;
}
*/
```

60

```c
//**************
//**************
void send_error(void)
{
        tone_on();
        delay_sec(1.0);
        tone_off();
}
//********************
//********************
void send_ok(void)
{
        tone_on();
        delay_sec(0.8);
        tone_off();
        delay_sec(0.5);
        tone_on();
        delay_sec(0.8);
        tone_off();
}
//*******************
//****************
void turn_off(unsigned char c)
{
        switch(c)
        {
                case load_1:    preset&=~(0x01);

                                                break;
                case load_2:    preset&=~(0x02);

                                                break;
```

```c
                case load_3:      preset&=~(0x04);
                                                break;

                case load_4:      preset&=~(0x08);
                                                break;

                default:          break;

        }

}
//******************
//******************

void turn_on(unsigned char c)

{

        switch(c)

        {

                case load_1:      preset|=0x01;
                                                break;

                case load_2:      preset|=0x02;
                                                break;

                case load_3:      preset|=0x04;
                                                break;

                case load_4:      preset|=0x08;
                                                break;

                default:          break;

        }

}
//******************
//******************

unsigned char compare_pin(unsigned char data *ptr)

{

        unsigned char data s[10];
```

```c
        unsigned char address=pin_address_start;

        unsigned char x;


        for(x=0;x<pin_lenght_size;x++)

        {

                s[x]=read(address);

                if(error)return 0;

                address++;

        }

        for(x=0;x<pin_lenght_size;x++)

        {

                if(s[x]!=*ptr)return 0;

                ptr++;

        }

        return 1;

}
//*****************
//*****************
unsigned char store_pin(unsigned char data *ptr)

{

        unsigned char address;

        unsigned char c;


        for(address=pin_address_start;address<pin_lenght_size;address++)

        {

                c=*ptr;

                if(write(address,c))return 0;

                ptr++;

        }

        return 1;
```

63

```c
}
//*********************
//*********************
unsigned char  decode_dtmf(unsigned char c)
{

        if(buffer[0]!=cmd_start)return 0;
        if(buffer[5]!=cmd_start)return 0;
        if(!(compare_pin(buffer+pin_start)))return 0;


        switch(c)
        {
                case load_ctrl:

        if((buffer[6]>max_load)||(buffer[6]<min_load))return 0;

        if((buffer[7]!=load_off)&&(buffer[7]!=load_on))return 0;

        if((buffer[7])==load_off)turn_off(buffer[6]);

                                        else turn_on(buffer[6]);


                                        set_presets();

        if(write(presets_address,preset))return 0;

                                        else return 1;


                case pin_change:        return(store_pin(buffer+pin2_start));


                default:                return 0;
        }

}
```

64

```c
//******************
//******************
void main(void)
{
        unsigned char c;

        if(sys_init())beep_error();
        #ifdef debug
        while(1)
        {
                lcd_clear();
                lcd_cmd(0x80);
                while(!tone_in);
                c=get_dtmf();
                if(error)
                {
                        lcd_clear();
                        lcd("command error!");
                        delay_sec(1);
                }
                else
                {
                        if(c=decode_dtmf(c)){
                                lcd_clear();
                                lcd("cmd exec ok!");
                                delay_sec(1);
```

65

```c
                    }
            else{
                    lcd_clear();
                    lcd("cmd exec error!");
                    delay_sec(1);
                    }
        }
}


#endif
#ifndef debug
while(1)
{
        while(!tone_in);
        c=get_dtmf();
        if(error)send_error();
        else{
                    if(c=decode_dtmf(c))send_ok();
                    else send_error();
            }
}
#endif

}
```