HOME CONTROL SWITCH AUTOMATION USING GSM COMMUNICATION

BY

CHUKWUKERE BRIDGET

EE/2008/276

DEPARTMENT OF ELECTRICAL ELECTRONIC ENGINEERING

FACULTY OF ENGINEERING, CARITAS UNIVERSITY,

AMORJI-NIKE,

ENUGU.

AUGUST 2013

HOME CONTROL SWITCH AUTOMATION USING GSM COMMUNICATION

BY

CHUKWUKERE BRIDGET

EE/2008/276

BEING A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR

THE AWARD OF BACHELOR OF ENGINEERING (B.ENG) DEGREE IN

ELECTRICAL ELECTRONIC ENGINEERING

FACULTY OF ENGINEERING

CARITAS UNIVERSITY, AMORJI-NIKE ENUGU.

AUGUST 2013

CERTIFICATION

This is to certify that this project was carried out by Chukwukere Bridget with registration number EE/2008/276, department of Electrical/Electronic Engineering, Caritas University Amorji-Nike Enugu.

----------------------------- -----------------------------

Engr. E.C Aneke Date

(Project Supervisor)

------------------------------ -----------------------------

Engr. C.O. Ejimofor Date

(H.O.D) Electrical Electronics

------------------------------- ---------------------------

External Examiner Date

DEDICATION

I dedicate this project report to God almighty for making it a successful one. May his name be praise now and forever. Amen

ACKNOWLEDGEMENT

My profound gratitude goes to my project supervisor ENGR.E.C ANEKE who always being my guidance throughout the duration of the project. My special appreciation and thanks to my family that always stands by me no matter what happens. Their full support and encouragement were such a boost for my capabilities and confidence to undergo this period. I want also to thank my friends for their invaluable assistances towards this project.

TABLE OF CONTENT **PAGES**

CHAPTER ONE

CHAPTER TWO

CHAPTER FIVE

ABSTRACT

The advantage of cellular communication like GSM technology is a potential solution for such remote controlling activities. GSM-SMS technology can be used to control household appliances from remote places. Remotely, the system allows the home owner to monitor and control his house appliances via his mobile phone set by sending commands in the form of SMS message and receiving the appliances status as well. This system provides ideal solution to the problems caused in situations when a wired connection between a remote appliance device and the control unit might not be feasible. The system is wireless and uses the user's mobile handset for control and therefore the system is more adaptable and cost effective. The system uses GSM technology thus providing ubiquitous access to the system for appliance control. With this hypothesis, three different approaches have been proposed and implemented in this project; FIRSTLY, home control switch/household appliances are controlled by server mobile which acts as remote control. SECONDLY, GSM-SMS messaging technology is used to control them from remote places and FINALLY, to provide a multiple agents environment.

CHAPTER ONE

INTRODUCTION

Nowadays, people's expectations in their life quality are increasing as the technology is improving rapidly. People need an affordable system (smart) that can make their lives easier, more comfortable, and offer more safety.GSM based home automation system is an electrical and electronic system designed to control home appliance with a mobile phone. The two main technologies applied in the system are GSM and PLC. GSM stands for global system for mobile communication and PLC stands for power line communication. In this study, Nokia 6100 is chosen as the GSM phone that receives command and transfers it to controller. Due to rapidly advancing of mobile communication technology and widely availability make it possible to incorporate mobile technology into home automation systems. ''AT'' command is used as communication between GSM phone and microcontroller by passing through a serial interface RS232 with the help of its driver.

The project is based on the principle of GSM network, which enables the user to remotely control the operations of the appliances by using a mobile phone. In other word it would transform a normal home into an intelligent home. The controlling circuit is the most important component in communication and interface between home appliances. It was implemented by using peripheral interface controller (PIC) interfaced to mobile phone. The user can perform ON/OFF operations of the appliances just by pressing keypad of mobile phone.

The user sends GSM data in the form of SMS (short message service) message to switch ON or OFF any appliances at home such as lamp, air condition, fan and etc. the appliances may also provide the user with its current ON/OFF status. The system can be improved to provide the user with information about the status of each appliance.

PROJECT OBJECTIVES

The objectives of this project are as follows;

1. To design and develop home appliances control system over GSM network by using a mobile phone.
2. To determine and understand on how the GSM works.
3. To understand the architecture and programming of the PIC.
4. To learn the troubleshooting and techniques.

SCOPE OF THE PROJECT

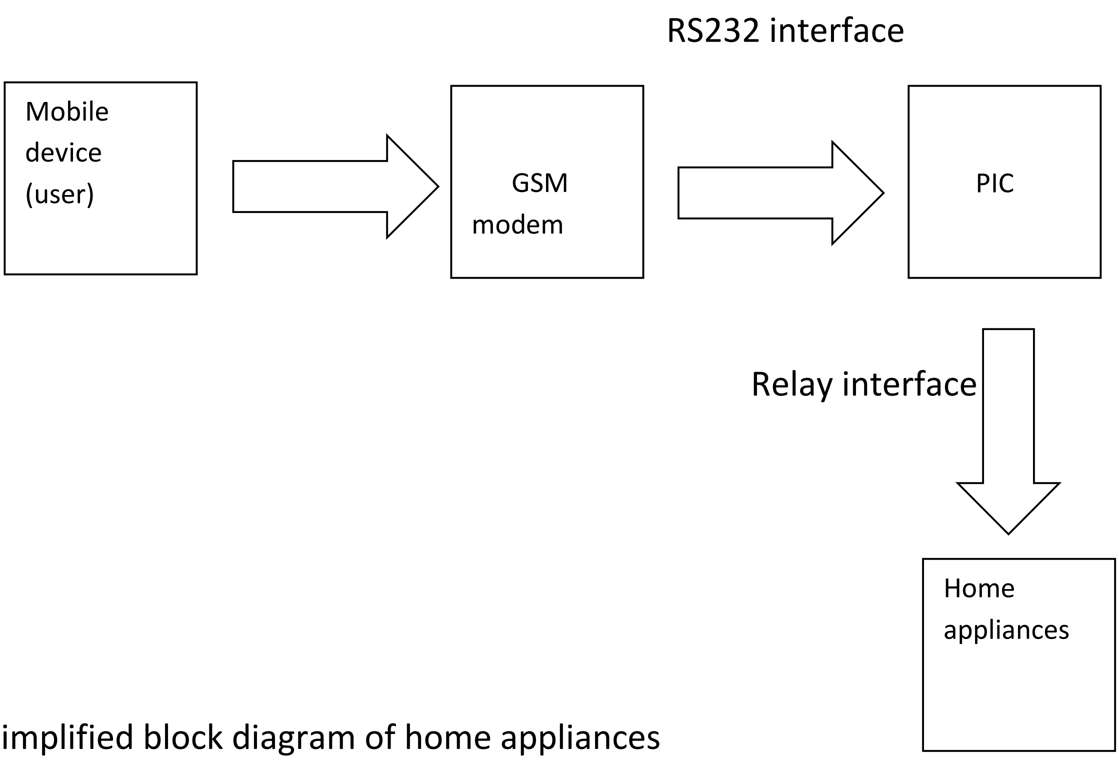
The scopes of works in this project are;

1. Mobile phone with SIM card enables the user to remotely control the operations of the appliances by communication to the GSM modem.
2. GSM modem allows the capability to send and receive SMS to and from the mobile phone.
3. RS-232 is the basic serial communication cable that will be used to interface between GSM modem and PIC microcontroller.
4. PIC microcontroller will be interfaced to the control circuit that is connected to the appliances. It contains the software components through which the appliances are controlled and monitored.
5. Relay and control circuit used to detect and control the condition of the home appliances.

OPERATION OF THE SYSTEM

The system works as a remote sensing for the electrical appliances at home or industry to check whether it is ON or OFF, at the same time the user can control the electrical appliances at home by sending SMS message into the system, for example, turning on the AC before returning home. in case of fire/security the chip will receive signals from the different sensors in the monitoring place and acts according to the received signal by sending an SMS message to user's mobile phone, it also works as automatic and immediate reporting to the user in case of emergency for home security, as well as immediate and automatic reporting to the fire brigade and police station according to activated sensor to decrease the time required for tracking action.

A sketch of overall design of the home appliances control using GSM is shown in the figure below.



FIG; Simplified block diagram of home appliances

Control using GSM.

ARCHTECTURE OF THE SYSTEM

It can be implemented to any levels of the security system. The architecture of the system mainly consists of three components the GSM MODEM and the interface circuit that include the different sensors used. The function of the GSM MODEM is the remote communication between the user and the controller through the RS232 serial communication standard. The function of the controller is to continuously check the inputs coming from the different sensor and send message through the GSM network in case of emergency such that it acts as a 24hours monitoring and continuously checking for any received message from the user through the GSM MODEM to switch on the AC for example.

CHAPTER TWO

LITERATURE REVIEW

SYSTEM DESCRIPTION

The system has two parts, namely; hardware and software. The hardware architecture consists of a stand-alone embedded system that is based on 8-bit microcontroller (AT89C51), a GSM handset with GSM modem and a driver circuit. The GSM modem provides the communication media between the homeowner and the system by means of SMS message. The SMS message consists of commands to be executed. The format of the message is predefined. The SMS message is sent to the GSM modem via the GSM public network as a text message with a definite predefined format. Once the GSM modem receives the message, the commands sent will be executed and executed by the microcontroller. The system will interpret the commands and turn the appliance ON/OFF accordingly via the switching module. The detail description of individual modules in the system is as follows.

A. **USER GSM MOBILE HANDSET**

Cellular phone containing SIM (Subscriber's Identifying Module) card has a specific number through which communication takes place.

B. **RECEIVER GSM HANDSET**

This receiver GSM handset is used to receive the SMS sent by the user and then to transmit an acknowledgement or status to the user's mobile. The receiver handset has to be equipped with an AT Modem and a valid SIM card. In our design we have used a Nokia GSM handset model 1616. The handset has a built in AT modem Receiver GSM Handset (in the System) User GSM Handset Microcontroller Board

UART Terminal, Switching Module, Device 1, Device 2, Device 3,Advanced Computing and Communication Technologies (ACCT 2011) with UART interface and supports most of the AT command instructions. This handset is attached with the microcontroller used to control the appliance through UART. AT Modem is a Modem which supports AT commands, also known as Hayes command. The Hayes command set is a specific command language originally developed for the Hayes Smart modem. The command set consists of a series of short text strings which combine together to produce complete commands for operations such as dialling, hanging up, and changing the parameters of the connection. Most modems follow the specifications of the Hayes command set. AT commands are instructions used to control a modem. AT is the abbreviation of Attention. Every command line starts with "AT" or "at".

MICROCONTROLLER BOARD

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of in-system programmable Flash memory. The device is manufactured using Atmel's high-density non-volatile memory technology and is compatible with the industry- standard 80C51 instruction set and pin out.

Features:

8K Bytes of In-System Programmable (ISP) Flash Memory

Endurance: 1000 Write/Erase Cycles

4.0V to 5.5V Operating Range

256 x 8-bit Internal RAM

32 Programmable I/O Lines

Full Duplex UART Serial Channel

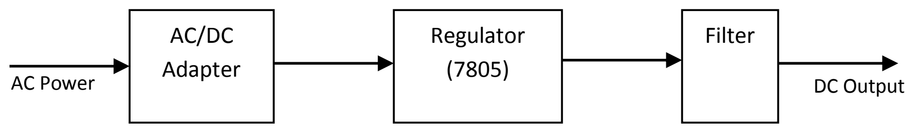
Fully Static Operation: 0 Hz to 33 MHz

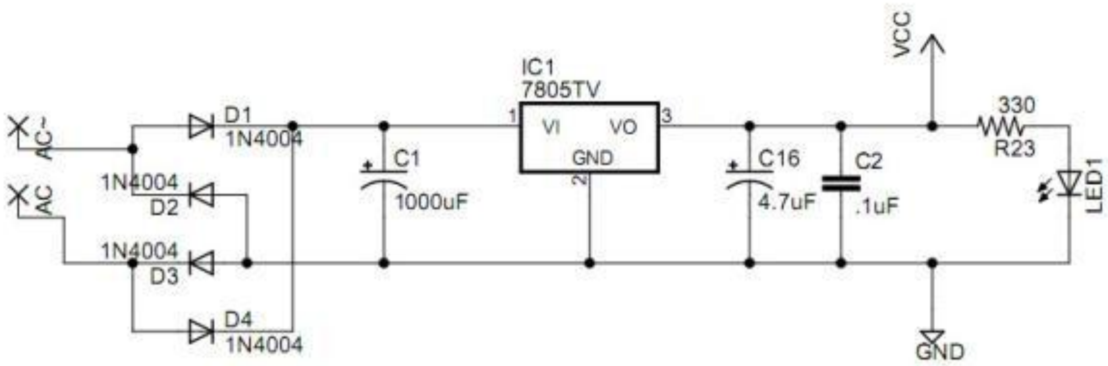
This contains the micro-controller (AT89C51) and a timeout generator circuit. This is the main module of the system. On receipt of the SMS message, text words are checked with predetermined format which includes desired device ON/OFF commands. To read a message the microcontroller sends the appropriate AT command to the Receiver GSM Modem through UART. The

Modem then responds with the message and the microcontroller stores the message in the RAM. When the message ends there is no way to know by the microcontroller. The time-out generator circuit performs the vital function of providing the microcontroller board with the ability to detect the end of a message from the receiver GSM mobile. The output of the time-out generator circuit (connected to port1_3 of the microcontroller) is low until the message is being received and becomes high at the end of the message. The microcontroller then processes the command and sends the appropriate controlling signal to the switching module.

POWER SUPPLY

The power supply is designed with the normal transformer based supply. The 230V AC line voltage is step down to the 9V AC using a step down transformer. The step downed voltage is driven to a full wave bridge rectifier, which consists of 4 1N4007 diodes (D1-D4). The DC voltage from the rectifier is connected through a capacitor C1, 1000uF. The capacitor acts as a filter by removing the ripples/ ac contents in the supply. The filter voltage is connected to the 1st pin of the LM7805, a 5V regulator IC. The regulated 5V out is taken from the 3rd pin and used for the systems power supply. Capacitors C2 and C16 are used as second stage filter for removing the transients from the supply. A LED is connected through a resistor for the indication of the power. The microcontroller and other devices get power supply from AC to Dc adapter through 7805, 5 volts regulator. The adapter output voltage will be 12V DC non-regulated. The 7805/7812 voltage regulators are used to convert 12 V to 5V/12V DC.





Vital role of power supply in ‘CORPORATE SECURITY SYSTEM BASED ON GSM’

The adapter output voltage will be 12V DC non-regulated. The 7805/7812 voltage regulators are used to convert 12 V to 5V/12V DC.

RELAY DRIVER

The relay driver is used to drive the relay that is used in the circuit. The relay act similar to the hook switch in the normal phone. When then line is in idle mode, the relay will be in OFF state which is similar to the ON-Hook in the Phones. In On-hook state, through the relay path the telephone line is connected to the ring indicator circuit. When a ring occurs, the ring indicator will interrupt the controller and the controller will send a signal to the relay driver, for driving the relay. The driving of the relay to the ON state will change the device to the OFF-hook state and the call will be connected.

RELAY BASICS

A relay is a simple electromechanical switch made up of an electromagnet and a set of contacts. It consists of a coil of wire

Surrounding a soft iron core, an iron yoke, which provides a low reluctance path for magnetic flux, a movable iron armature, and a set, or sets, of contacts. The armature is hinged to the yoke and mechanically linked to a moving contact or contacts. It is held in place by a spring so that when the relay is de-energized there is an air gap in the magnetic circuit. In this condition, one of the two sets of contacts in the relay pictured is closed, and the other set is open. Other relays may have more or fewer sets of contacts depending on their function. The relay also has a wire connecting the armature to the yoke. This ensures continuity of the circuit between the moving contacts on the armature, and the circuit track on the printed circuit board (PCB) via the yoke, which is soldered to the PCB. When an electric current is passed through the coil, the resulting magnetic field attracts the armature and the consequent movement of the movable contact or contacts either makes or breaks a connection with a fixed contact. If the set of contacts was closed when the relay was De-energized, then the movement opens the contacts and breaks the connection, and vice versa if the contacts were open. When the current to the coil is switched off, the armature is returned by a force, approximately half as strong as the magnetic force, to its relaxed position. Usually this force is provided by a spring, but gravity is also

used commonly in industrial motor starters. Most relays are manufactured to operate quickly. In a low voltage application, this is to reduce noise. In a high voltage or high current application, this is to reduce arcing. If the coil is energized with DC, a diode is frequently installed across the coil, to dissipate the energy from the collapsing magnetic field at deactivation, which would otherwise generate a voltage spike dangerous to circuit components. Some automotive relays already include a diode inside the relay case. Alternatively a contact protection network, consisting of a capacitor and resistor in series, may absorb the surge. If the coil is designed to be energized with AC, a small

Copper ring can be crimped to the end of the solenoid. This "shading ring" creates a small out-of phase current, which increases the minimum pull on the armature during the AC cycle. The whole circuit works on the 5V, DC power. A transformer based power supply is used in this project, to step down the direct 230V AC to 9VAC. The step downed voltage is applied to the rectifier circuit which will convert the AC to DC. An electrolyte capacitor will act as a filter for smoothening the dc. The popular 5V regulator, LM7805 is used for regulate the 9VDC to 5VDC.

SWITCHING MODULE

This module drives (switches ON/OFF the appliance according to the command sent in the SMS. The switching module is controlled by the microcontroller. The switching module may be in the form of a relay which allows a low power circuit to switch a relatively high current on or off for example a bulb connected to the 220V mains supply.

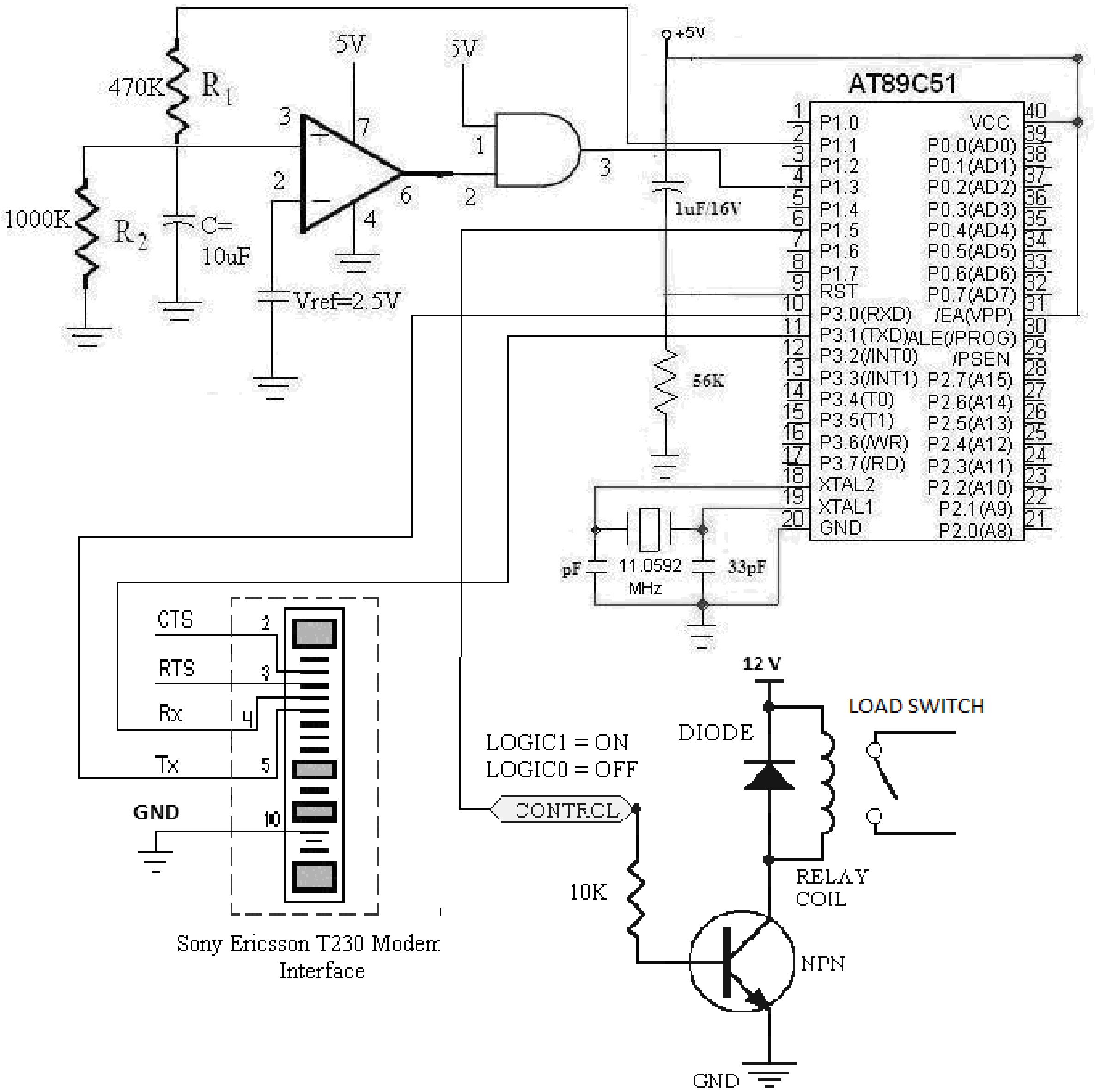


FIG. 2 CIRCUIT DIAGRAM OF THE COMPLETE SYSTEM

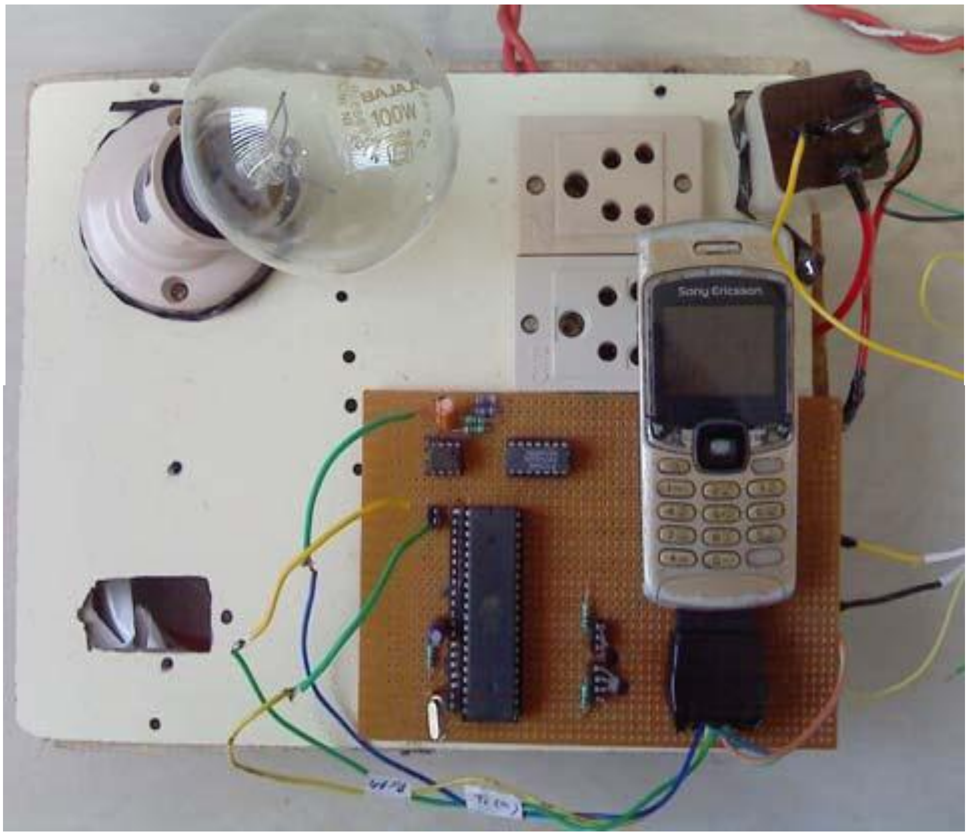


FIG. 4 PHOTO OF THE COMPLETE SYSTEM

CHAPTER THREE

SOFTWARE DESIGNING

There are hundreds of programming languages each was developed to solve a particular type of problem. Most traditional languages such as basic, C, COBOL, FOTRAN, PL/1, and PASCAL are procedural languages. That is the program sequence determines the exact sequence of operations. Programming language is a free field language. Precedence of the operator determines the order of operation. Comments are used to document the software; pre-process or directives are special operations that occur first. A global declaration provides modular building blocks. Declarations are the operations. Function declarations allow for one routine to call another. Compound statements are the more complex operations. Global variables are permanent and can be shared. Local variables are temporary and are private source a file makes it easier to maintain large projects. The software was designed using CRIMSON EDITOR C complier and SDCC (small device C complier) which contains the header file of the Microcontroller (AT89C51). It used virtual conventional C programming language keywords and syntax. The program environment is where the code is written, compile and debug. It will generate and Intel ex file (content of the system Rona) which is transfer into the decontrolled via a computer interfaced programming device.

HARDWARE DESIGNING

The hardware parts where described as unit below

TRANSFORMER

Transformers transforms an alternating current voltage (ac) and electric signal which is applied to its primary voltage winding (an inductor) which by mutual induction output voltage with respect to the applied voltage and it's conductor turn ratio with little loss of power. Transformers work only with AC and this is one of the reasons why mains electricity is AC. Step-up transformers increase voltage, step-down transformers reduces voltage. Most power supplies use a step-down transformer to reduce the dangerously high mains voltage (220v in Nigeria) to a safer low voltage. The input coil is called the primary and the output coil is called the secondary. There are no electrical connections between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. The two lines in the middle of the circuit symbol represent the sore. Transformers waste very little power so the power out is (almost) equal to the power in. note that as voltage is stepped down current is stepped up.

The ratio of the number of turns on each coil called the turn ratio determines the ration of the voltages. A step-down transformer has a large number of turns on its primary (input) coil which is connected to the high voltage mains supply and a small number of turns on its secondary (output) coil to give a low output voltage. Turn ratio = Vp/ Ns = Np/Ns and power = Vs * Is,

Vp = primary (input) voltage

Np = number of turns on primary coil. Ip = primary (input) current, Vs = secondary (output) voltage, Ns = number of turns on secondary coils = secondary (output) current.

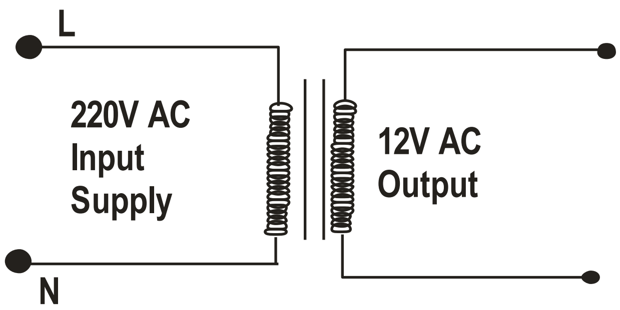


Fig. Transformer symbol

TRANSFORMER CALCULATION

Assure the couple coefficient of the transformer K = 1. The power factor can then be calculated using this formula. A transformer rated 24ov to 12volt (step-down), the ratio is 20:1, if the secondary turns is 48 then the primary turn can be simply derived using two methods;

Method 1: Using the turn ratio;

Since the turn ration is 20:1

Then the primary turn are 48 x 20 = 960turns

Method 2:

Using Np/Ns = Vp/Vs

Make Np the subject

Np = (Vp x Ns)/VsNP = (240 x 48)/12 = 960turns

RECTIFIERS

There are several ways of connecting diodes to make a rectifier (convert AC to DC). The bridge rectifier is the most important and it produces full-wave varying DC. A full-wave rectifier can also be made from jest two diodes if a centre-tap transformer is used, but this method is rarely used now that diodes are cheaper. A single diode can be used as a rectifier but it only uses the positive (x) parts of the AC wake to produce half-wave varying DC.

BRIDGE RECTIFIER

A bridge rectifier can be made using four individual diodes, but it is also available in special packages containing the four diodes required. It is called a full-wave rectifier because it uses the entire AC wake (both positive and negative sections.

1.4v is used up in the bridge rectifier because each diode uses 0.7v when conducting and there are always two diodes conducting, as shown in the diagram below

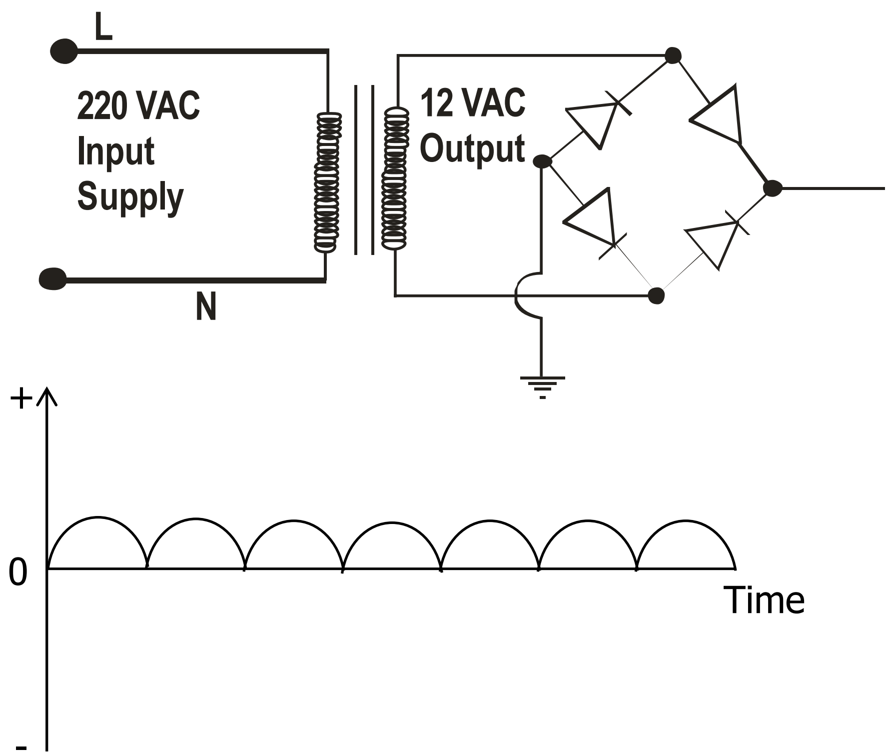
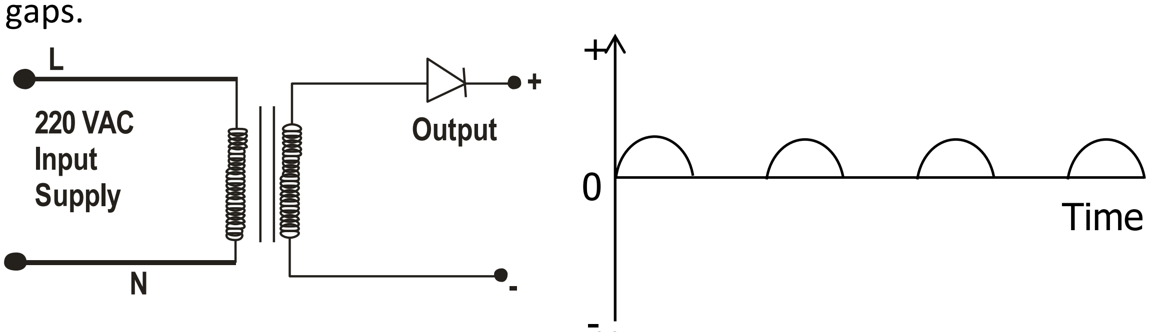


Fig. Full wave rectifier (bridge rectifier) and wave form

HALF WAVE RECTIFIER (single diode rectifier)

A single diode can be used as a rectifier but this produces half-wave

varying DC which has gaps when the AC is negative. It is hard to smooth this sufficiently well to supply electronic circuits unless they require a very small current so the smoothing capacitor does not significantly discharge during the gaps.



3.18 CAPACITORS (SMOOTHING)

Smoothing is performed by a large value electrolytic capacitor connected across the DC supply to act as a reservoir, supplying current to the output when varying DC voltage from the rectifier is falling. The diagram shows the unsmoothed varying DC (dotted line) and the smoothed DC (solid line). The capacitor charges quickly near the peak of the varying DC, and then discharges as it supplies current to the output.

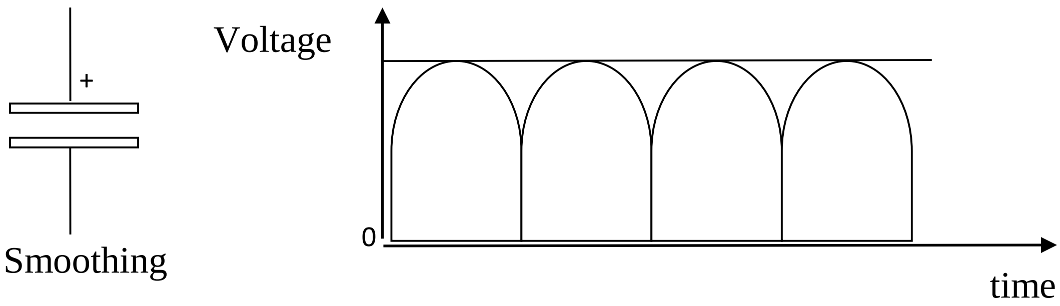


Figure 3.4.1 capacitor charging and discharging

Note that smoothing significantly increases the average DC voltage to almost the peak value (1.4 x RMS value). For example 6v RMS AC is rectified to full wave DC of about 4.6v RMD (1.4v is lost in the bridge rectifier). With smoothing this increases to almost the peak value giving 1.4 x 4.6 = 6.4v smooth DC.

Smoothing is not perfect due to the capacitor voltage falling a little as it discharges, giving a small ripple voltage. For many circuits a ripple which is 10% of the supply voltage is satisfactory and the equation below gives the required value for the smoothing capacitor. A larger capacitor will give fewer ripples. The capacitor value must be doubled when smoothing half-wave DC. Smoothing capacitor for 10% efficiency, C = 5 x Io/Vs x f

C = smoothing capacitance in farads (f), Io = output current from the supply in amps (A), Vs = supply voltage in volts (v), this is the peak value of the unsmoothed DC. f = frequency of the AC supply in hertz (Hz), 50Hz in the Nigeria.

The 78xx, 78Mxx and 78Sxx regulators all have the pin-out shown in the left of figure 4.5 and are normally supplied in a case style known as TO-220. The 78Lxx series, shown in the right of figure 4.51 also has the same pin-out but out has a case style known as TO-92. They are all connected to the rest of the power supply in the same way, as shown below

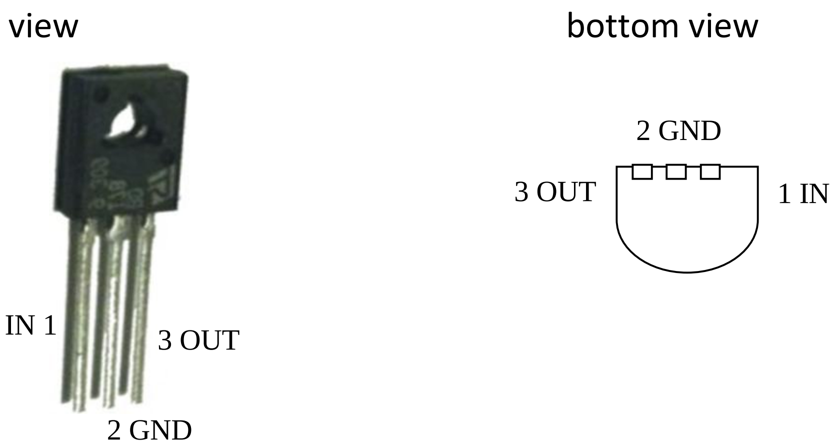


Fig 3.5

Pin out diagram of the 78xx series of regulator ICs

IN 1 3 OUT

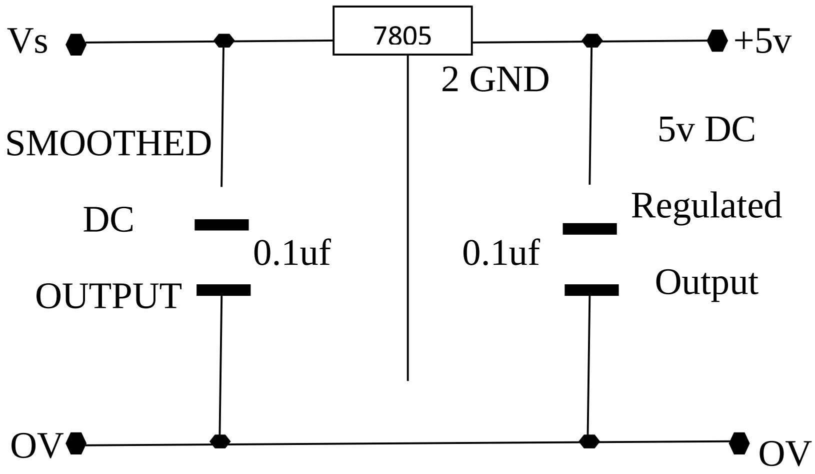


Fig 3.6 wiring up a regulator IC

The diagram above shows the Voltage Regulator pin-out and wiring.

DIODES

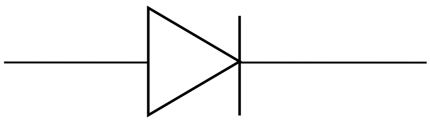


Fig 3.7 Diodes and its circuit symbol

FUNCTIONS OF DIODE

Diodes allow electricity to flow in only one direction. The arrow of the circuit symbol shows the direction in which the current can flow. Diodes are the electrical version of a valve and early diodes were actually called valves.

RESISTORS

Resistors restrict the flow of electric current, for example a resistor is placed in series with a light-emitting diode (LED) to limit the current passing through the LED.

CONNECTING AND SOLDERING OF RESISTOR

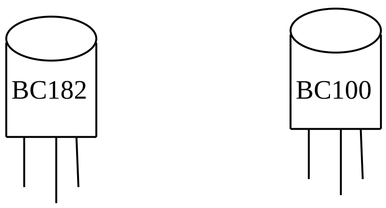
Resistors may be connected either way round. They are not damaged by heat when soldering.

TRANSISTORS

Transistors amplify current, for example they can be used to amplify the small output current from a logic chip so that it can operate a relay or other high current device. In many circuits a resistor is used to convert the changing current to a changing voltage, so the transistor is being used to amplify voltage.

A transistor may be used as a switch (either fully on with maximum current, or fully off with no current) and as an amplifier (always partly on).

The amount of current amplification is called the current gain, symbol h_{fE}



TYPES OF TRANSISTORS

There are two types of standard transistors, NPN and PNP, with different circuit symbols.

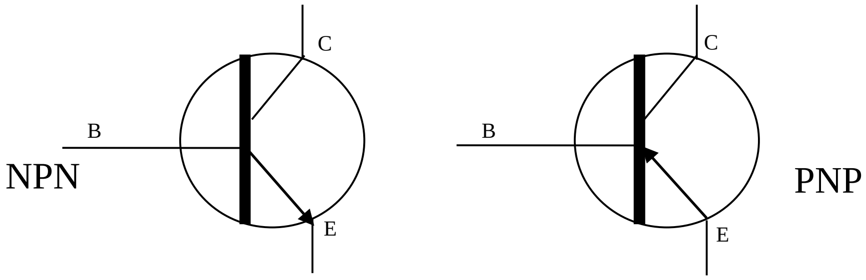


Fig 4.8

The letters refer to the layers of semiconductor material used to make the transistor. Most transistors used today are NPN because this is the easiest type to make from silicon. If you are new to electronics it is best to start by learning how to use NPN transistors.

The leads are labelled base (B), collector (C) and emitter (E).

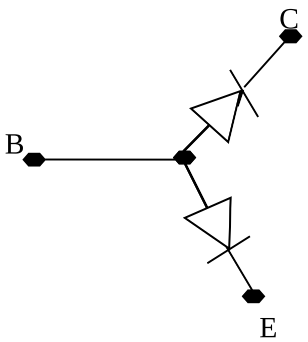
These terms refer to the internal operation of a transistor but they are not much help in understanding how a transistor is used, so just treat them as labels!

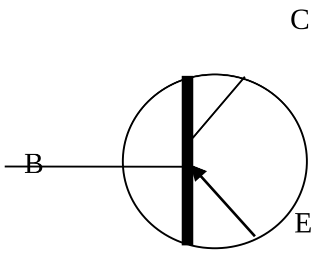
A Darlington pair is two transistors connect together to give a very high current gain.

TESTING A TRANSISTOR

Transistors can be damaged by heat when soldering or by misuse in a circuit. If you suspect that a transistor may be damaged there are two easy ways to test it.

Testing with multimeters: use a multimeter or a simple tester (battery, resistor and LED) to check each pair of leads for conduction. Set a digital multimeter to diode test and an analogue multimeter to a low resistance range.





TESTING AN NPN TRANSISTOR

Test each pair of leads both ways:

1, the base-emitter (BE) junction should behave like a diode and conduct one way only.

2, the base-collector (BC) junction should behave like a diode and conduct one way only

3, the collect-emitter (CE) should not conduct either way.

The diagram shows how the junctions behave in an NPN transistor. The diodes are reversed in a PNP transistor but the same test procedure can be used.

Testing in a simple switching circuit:

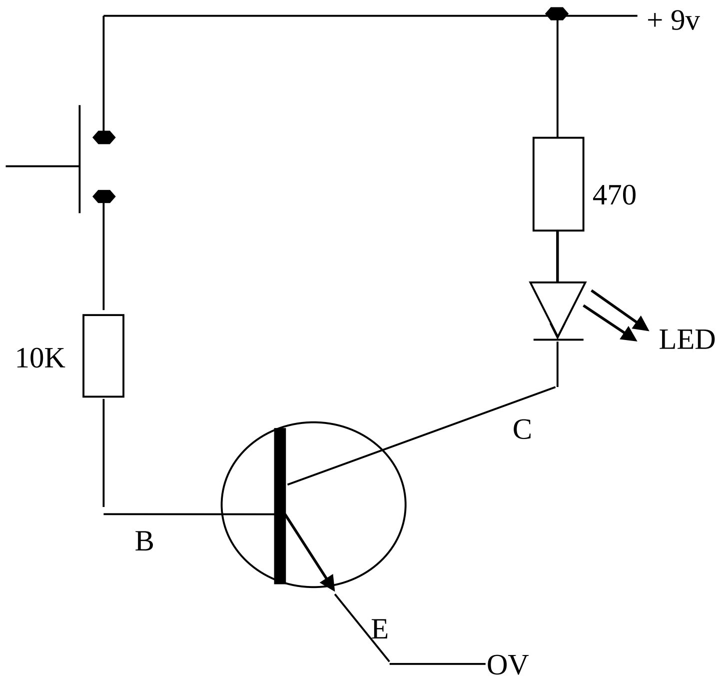


Fig A simple switching circuit to test an NPN transistor

Connect the transistor into the circuit as shown above, which uses the transistor as shown above, which uses the transistor as a switch. The supply voltage is not critical; anything between 5 and 1 is suitable. This circuit can be quickly built on breadboard. Careful to include the lokol resistor in the base connection or you will destroy the transistor as you test it. If the transistor is OK the LED should light when the switch is pressed and not light when the switch is released.

To test a PNP transistor use the same circuit but reverse the LED and the supply voltage.

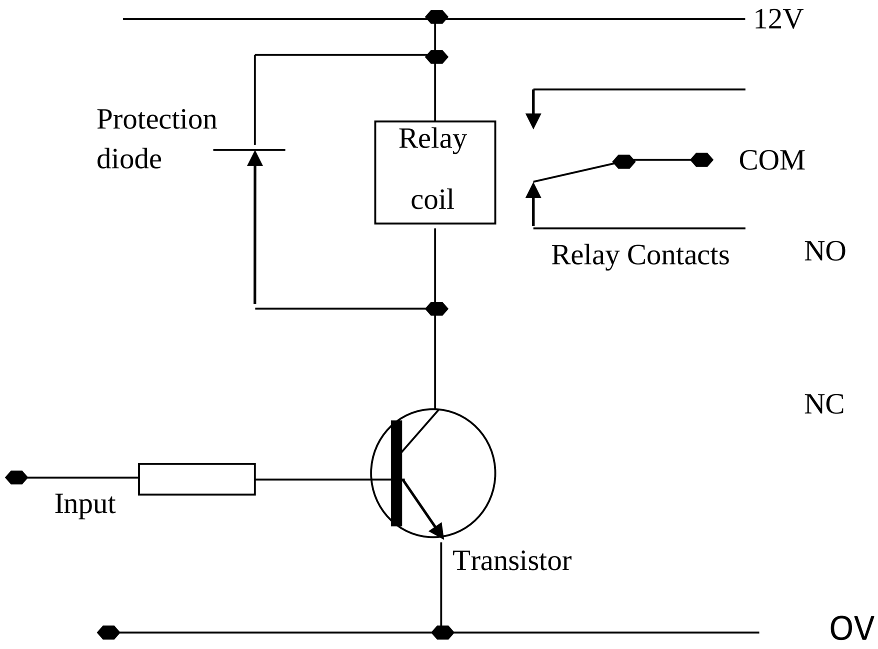
Some multimeters have a transistor test function which provides a known base and measures the collector current. So as to display the transistor DC current gain H_{fE}

RELAY

Relay is an electromechanical device with two functional parts the mechanical contact part is the switch while electrical part is the solenoid, when voltage is applied across the solenoid coil (inductor) it becomes a magnet, this causes the switch contact at normally close to drift to the normally open position. If the voltage across the inductor is removed it returns to its normal position.

PROTECTION DIODES FOR RELAYS

Signal diodes are also used with relays to protect transistors and integrated circuits from brief high voltage produced when the coil is switched off. The diagram below shows how protection diode is connected across the coil, note that the diode is connected 'backwards' so that it will normally conduct. Conduction only occurs when the relay coil is switched off, at this moment current tries to continue flowing through the coil and it is harmlessly diverted through the diode. Without the diode no current could flow and the coil would produce a damaging high voltage 'spark' in its attempt to keep the current flowing.



How protection diode is being connected across relay coil

MICROCONTROLLER

This is the bedrock of the system, its functionality is based on the ability of the designer to arrange his instruction or generate the code which will generate the machine language, which is the content of the ROM (read only memory) of the microcontroller.

FEATURES OF THE MICROCONTROLLER

A, Compatible with MCS-51™ products

B, 4k bytes of in-system reprogrammable flash memory-endurance:

1, 000 writ/erase cycles

C, Fully static operation: 0Hz to 24MHz

D, Three-level program memory lock

E, 128 x 8-bit internal RAMS

F, 32 programmable 1/0 lines

G, Two 16-bit timer/counters

H, six interrupt sources

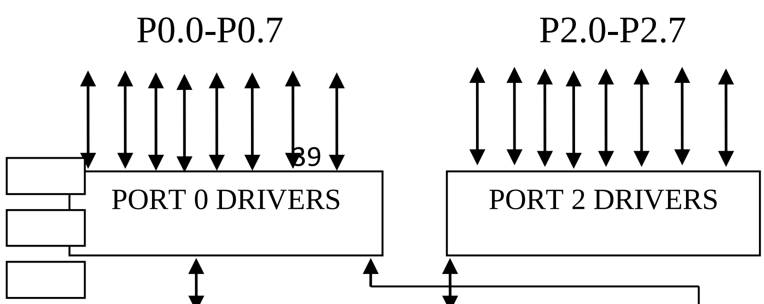
I, Programmable serial channel

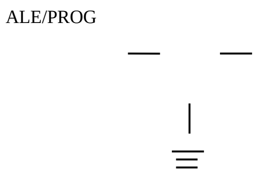
J, Low-power idle and power-down modes

DESCRIPTIONS

The AT89C51 is a low-power; high performance CMOS 8-bit microcontroller with 4k bytes of flash programmable and erasable read only memory (PE ROM). The device is manufactured using Atmel's high-density non-volatile memory technology and is compatible with the industry-standard MCS-51 instruction set and pin-out. The on-chip flash allows the program memory to be reprogrammed in-system or by a conventional non-volatile memory programmer. By combining a versatile 8-bit CPU with flash on a monolithic chip, the Atmel AT89C51 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control application.

Fig Internal block diagram of microcontroller AT89C51





The AT89C51 provides the following standard feature 4kbytes of flash, 128bytes of a RAM, 32 1/0 lines, two 16-bit timer/counters, five vector two level interrupt architecture, a full duplex serial port, and on-chip oscillator and clock circuitry.

In addition,ed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The idle mode stops the CPU while allowing the RAM timer/counters, serial port and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset.

PIN DESCRIPTION

PORT 0

Port 0 is an 8-bit open-drain bi-directional 1/0 port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high impedance inputs. Port 0 may also be configured to be the multiplexed low order address/data bus during accesses to external program and data memory. In this mode port 0 has internal pull-ups. Port 0 also receives the code bytes during flash programming, and outputs the code bytes during program verification. External pull ups are required during program verification.

PORT 1

Port 1 is an 8-bit bi-directional 1/0 port with internal pull-ups. The port 1 output buffers can sink/source four TTL inputs. When 1s are written to port 1 pins they are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally being pulled low will source current (IIL) because of the internal pull-ups. Port 1 also receives the low-order address bytes during flash programming and verification.

PORT 2

Port 2 is an 8-bit bi-directional 1/0 port with internal pull-ups. The port 2 output buffers can sink/source four TTL inputs. When 1s are written to port 2 pins they are pulled high by the internal pull-ups and can be used as inputs.

Port 2 pins that are externally being pulled low will source current (IIL) because of the internal pull-ups. Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that uses 16-bit addresses (MOVX@DPTR), port 2 emits the contents of the P2 special function register. Port 2 also receives the high-order address bit and some control signals during flash programming and verification.

PORT 3

Port 3 is an 8-bit bi-directional 1/0port with internal pull-ups. The port 3 output buffer can sink/source four TTL inputs. when 1s are written to port 3 pins they are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current (IIL) because of the pull ups.

CHAPTER FOUR

TESTING ANALYSIS AND PACKAGING

TESTING:

In every engineering outfit projected work must undergo series of test before the project will be satisfied okay. In production chain it will be tagged with a label, Q.C passed.

In small system designs, testing process is also applicable. There are sequences of test needed to undergo for any successful project.

1, Testing of the individual components.

2, Unit by unit testing.

3, System testing.

TESTING OF THE INDIVIDUAL COMPONENTS

Components should be tested individually before fiddling with it, so as to remove the bad ones. This test will be satisfied test measures for individual components which are basically used of the multimeter.

Testing of transistor, diodes, LED, LDR and every other component that will be used in the project.

UNIT BY UNIT TESTING

These involve wiring up circuitry and testing to confirm its functions before soldering. Examples are bread boarding of power supply stage of the project, testing it and confirming that is functional before soldering.

SYSTEM TESTING

This involves the testing of the entire circuitry and cross examines it for errors like short circuits, lead flux, joining unwanted links. Proper insertion of IC pin layout and also checking if ICs of these pin number are slotted in their proper base. After this check, cross check again before powering the system.

CAUTIONS

You must have a standby fire extinguisher in case of fire outbreak. Protective gadgets like goggle should be worn while performing these tests, more especially on power test. Any wrong connection made on circuit might cause an explosion. Work on ventilated environment and do not conduct any form of play while working. Apply the necessary rules, to abstain endangering your life and others as well.

INTEGRATION

The integration of the units that made up the circuit and testing them now as unified system. Here if the project is not working as assumed, necessary changes can now be made until a desirable outcome is attained.

PACKAGING

Every quality and good product is often determined by packaging. Credit is awarded to properly packaged project. After the integrating and final testing of the project, I now made choice of package considering cost as well as durability elegance. I then choose to embark on metallic casing where the said project was housed.

Before construction of this package I considered the size of the project and maintenance factor which need may arise. There are some other ways the project can be packaged namely,

1, Plastic packaging

2, Wooden packaging.

LIST OF THE SYSTEM COMPONENT.

1, Microcontroller (AT89C51) Atmel product

2, 20Dip (dual in package) IC socket

3, 12volts/500 milliamps relay

4, IN400I (rectifying diode)

5, 1000 micro-farad capacitor (25v/electrolytic)

6, 10micro farad capacitor (16v/electrolytic)

7, 30pico farad capacitor (non-electrolytic)

8, C 945 (NPN transistor)

9, 78 x os (voltage regulator)

10, 1kilo ohms resistor

11, Red LED

CHAPTER FIVE

PROBLEM ENCOUNTERED, PRECAUTIONS, SUMMARY AND RECOMMENDATIONS

PROBLEM ENCOUNTERED

- Inconsistent supply of power from NEPA.(national electric power authority) this made the project go at snail pace.
- Writing, running and debugging of the programme was the most stressful. In order to develop a working program a lot of brain storming was done.
- Running around to source components for the project was really herculean as some of the components are not even around us in the east.

PRECAUTIONS

Precautions were taking in both construction techniques and during the testing period. These precautions were taken in areas as follows;

- The design was first carried out on a bread board, tested and verified before transferring it to the Vero board. This was to ensure reduced component damage.
- At soldering stage, caution was taken to avoid direct heating of the electronic components used to avoid damage since some components are easily affected by heat.

Precaution was also taken during the testing of the program stored in the microcontroller to avoid the system not working properly.

CONCLUSION

In the paper low cost, secure, ubiquitously accessible, auto configurable, remotely controlled solution for automation of homes has been introduced. The approach discussed in the paper has achieved the target to control home appliances remotely using the SMS-based system satisfying user needs and requirements. The extensive capabilities of this system are what make it so interesting. From the convenience of a simple cell phone, a user is able to control and monitor virtually any electrical device in a household. By connecting all the appliances with the system through power line communication or wireless to the system, all electrical household appliances can be controlled by sending a message from a mobile handset.

APPENDIX C

BILL of engineering measurement and evaluation

S/N	Components or materials	Quantity	Price per unit	Total cost
1.	Vero board	1	120,00	120,00
2.	3300 microfarad capacitor	1	45,00	45,00

3.	10 microfarad capacitor	1	20,00	20,00
4.	20 DIP (IC Base)	1	40,00	40,00
5.	IN400I (Rectifying Diode	4	10,00	40,00
6.	LED	13	10,00	130,00
7.	IC (AT89C51)	1	2000,00	2000,00
8.	Resistors	39	10,00	390,00
9.	Connecting wires	2 yards	150,00	300,00
10.	Soldering iron	1	300,00	300,00
11.	Soldering lead	6yards	50,00	300,00
12.	Transistor	1	50,00	50,00
13.	Relay	2	500,00	1000,00
14.	Lamp holder	1	80,00	80,00
15.	Colored bulb	1	60,00	60,00
16	Lead sucker	1	350,00	350,00
17.	Casing	1	2000,00	2000,00
18.	Internet browsing	----	3000,00	3000,00
19.	Step down transformer	1	280,00	280,00
20.	Transportation	----	2000,00	2000,00
21.	Miscellaneous	----	5000,00	5000,00
22.	Mobile phone	2	13,000,00	26,000,00
	GRAND TOTAL	**-----**	**------**	**₦ 35,475,00**

APPENDIX 3: SYSTEM SOURCE CODE

Source code.

System Controller source

```
;
;
;*********************************************************
```

```
        list            p=pic16f84a
        include         p16f84a.inc
        __config _hs_osc & _wdt_off & _pwrte_on & _cp_off
        errorlevel      -302    ;Eliminate bank warning


;***************  Label Definition  ********************
        cblock  h'0c'
rx_status               ;SC status save area
rx_substatus            ;SC substatus save area
rx_edge                 ;Input edge check flag
last_look               ;Input last look flag
code_ck                 ;Code check flag
cont_data               ;Control Data
w_save                  ;W reg save area
s_save                  ;Status reg save area
        endc

ptn1    equ     b'11110000'     ;Pattern 1 data
ptn2    equ     b'00001111'     ;Pattern 2 data
ra1     equ     d'1'            ;RA1 bit position
rb5     equ     d'5'            ;RB5 bit position
rb7     equ     d'7'            ;RB7 bit position

;***************  Program Start  **********************
        org     0               ;Reset Vector
        goto    init
        org     4               ;Interrupt Vector
        goto    int

;*****************  Initial Process  ******************
init    bsf     status,rp0      ;Change to Bank1
```

```
        movlw   b'00011111'     ;RA4-0:IN mode
        movwf   trisa           ;Set TRISA reg
        movlw   b'00000000'     ;RB1-0:INP mode
        movwf   trisb           ;Set TRISB reg
        movlw   b'00000101'     ;RBPU/TOCS/PSA=0,PS=101
        movwf   option_reg      ;Set OPTION_REG
        bcf     status,rp0      ;Change to Bank0

        clrf    portb           ;RL1,RL2 OFF
        clrf    rx_status       ;Clear SC status
        clrf    rx_substatus    ;Clear SC substatus
        clrf    rx_edge         ;Clear Edge check flag
        incf    last_look,f     ;Set Last Look flag ON
        clrf    code_ck         ;Clear code check flag
        clrf    cont_data       ;Clear Control Data

        movlw   d'100'          ;256-10000us/64us = 100
        movwf   tmr0            ;Set 10msec to TMR0
        movlw   h'a0'           ;GIE=1,TOIE=1
        movwf   intcon          ;Interruption enable

;*********** Initial Input check Process **************
edge_check
        btfss   porta,ra1       ;Input signal ON ?
        goto    check1          ;No. Signal OFF
        btfsc   last_look,0     ;Last Look flag OFF ?
        goto    edge_check      ;No. Input NOT changed
        bcf     intcon,gie      ;Interruption disable
        incf    code_ck,f       ;Set code check flag ON
        incf    last_look,f     ;Set Last Look flag ON
        movlw   d'120           ;256-5000us/64us = 120
        movwf   tmr0            ;Set 60sec to TMR0
        bsf     intcon,gie      ;Interruption enable
```

```
wait
        btfss   rx_edge,0       ;Input edge check ?
        goto    wait            ;No. Wait interruption
        clrf    rx_edge         ;Clear edge check flag
        goto    edge_check      ;Jump to Input edge check
check1
        clrf    last_look       ;Set Last Look flag OFF
        goto    edge_check      ;Jump to Input edge check

;************  Timer Interruption Process  **************
int
        movwf   w_save          ;Save W register
        movf    status,w        ;Read STATUS reg
        movwf   s_save          ;Save STATUS reg
        bcf     status,rp0      ;Change to Bank0
        bcf     intcon,t0if     ;Clear timer int flag

        movf    dis,w           ;Read dis ON counter
        btfsc   status,z        ;Counter = 0 ?
        goto    stchk0          ;Yes
        decfsz  relay,f         ;Counter - 1 = 0 ?
        goto    stchk0          ;No
        clrf    portb           ;Read dis

stchk0
        movf    code_ck,w       ;Read code check flag
        btfsc   status,z        ;Flag ON ?
        goto    int_end         ;No. End of interruption
        movf    rx_status,w     ;Read RX status
        btfss   status,z        ;Status = 0 ?
        goto    stchk1          ;No. Next
```

```
;***********  Preamble data check Process  *************
        movf    rx_substatus,w  ;Read RX substatus
        btfss   status,z        ;Substatus = 0 ?
        goto    st00            ;No. Next
        goto    st_on           ;Input signal ON ?
st00    movlw   d'1'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 1 ?
        goto    st01            ;No.
        goto    st_off          ;Input signal OFF ?
st01    movlw   d'2'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 2 ?
        goto    st02            ;No.
        goto    st_on           ;Input signal ON ?
st02    movlw   d'3'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 3 ?
        goto    st03            ;No.
        goto    st_off          ;Input signal OFF ?
st03    movlw   d'4'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 4 ?
        goto    st04            ;No.
        goto    st_on           ;Input signal ON ?
st04    movlw   d'5'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 5 ?
        goto    st05            ;No.
        goto    st_off          ;Input signal OFF ?
st05    movlw   d'6'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
```

```
        btfss   status,z        ;Substatus = 6 ?
        goto    st06            ;No.
        goto    st_on           ;Input signal ON ?
st06    movlw   d'7'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 7 ?
        goto    st07            ;No. Substatus=8
        goto    st_on           ;Input signal ON ?
st07    btfsc   porta,ra1       ;Input signal OFF ?
        goto    illegal         ;No. Jump to illegal
        clrf    rx_substatus    ;SC substatus = 0
        incf    rx_status,f     ;SC status = 1
        clrf    cont_data       ;Clear count data
        goto    int_end         ;End of interruption

;*****************************************************
stchk1
        movlw   d'1'            ;Set check data
        subwf   rx_status,w     ;SC status - check data
        btfss   status,z        ;SC status = 1 ?
        goto    stchk2          ;No. Next
;************  Control data check Process  **************
        movf    rx_substatus,w  ;Read SC substatus
        btfss   status,z        ;Substatus = 0 ?
        goto    st10            ;No. Next
        goto    st_on           ;Input signal ON ?
st10    movlw   d'1'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 1 ?
        goto    st11            ;No.
        btfss   porta,B         ;B0-1 = 1 ?
        goto    stinc           ;Jump to Substatus + 1
```

```
        bsf     cont_data,0     ;Set B0-1 = 1
        goto    stinc           ;Jump to Substatus + 1
st11    movlw   d'2'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 2 ?
        goto    st12            ;No.
        goto    st_off          ;Input signal OFF ?
st12    movlw   d'3'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 3 ?
        goto    st13            ;No.
        goto    st_on           ;Input signal ON ?
st13    movlw   d'4'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 4 ?
        goto    st14            ;No.
        btfss   porta,ra1       ;B1 = 1 ?
        goto    stinc           ;Jump to Substatus + 1
        bsf     cont_data,1     ;Set B1 = 1
        goto    stinc           ;Jump to Substatus + 1
st14    movlw   d'5'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 5 ?
        goto    st15            ;No.
        goto    st_off          ;Input signal OFF ?
st15    movlw   d'6'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 6 ?
        goto    st16            ;No.
        goto    st_on           ;Input signal ON ?
st16    movlw   d'7'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
```

```
        btfss   status,z        ;Substatus = 7 ?
        goto    st17            ;No.
        btfss   porta,ra1       ;B2 = 1 ?
        bsf     cont_data,2     ;Set B2 = 1
        st17    movlw   d'8'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 8 ?
        goto    st18            ;No.
        goto    st_off          ;Input signal OFF ?
st18    movlw   d'9'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data

btfss   status,z        ;Substatus = 9 ?
        goto    st19            ;No.
        goto    st_on           ;Input signal ON ?
st19    movlw   d'10'           ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 10 ?
        goto    st20            ;No.
        btfss   porta,ra1       ;B3 = 1 ?
        goto    stinc           ;Jump to Substatus + 1
        bsf     cont_data,3     ;Set B3 = 1
        goto    stinc           ;Jump to Substatus + 1
st20    movlw   d'11'           ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 11 ?
        goto    st21            ;No.
        goto    st_off          ;Input signal OFF ?
st21    movlw   d'12'           ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 12 ?
        goto    st22            ;No.
        goto    st_on           ;Input signal ON ?
```

```
st22   movlw   d'13'          ;Set check data
       subwf   rx_substatus,w ;Substatus - check data
       btfss   status,z       ;Substatus = 13 ?
       goto    st23           ;No.
       btfss   porta,ra1      ;B4 = 1 ?
       bsf     cont_data,4    ;Set B4 = 1
       goto    stinc          ;Jump to Substatus + 1
st23   movlw   d'14'          ;Set check data
       subwf   rx_substatus,w ;Substatus - check data
       btfss   status,z       ;Substatus = 14 ?
       goto    st24           ;No.
       goto    st_off         ;Input signal OFF ?
st24   movlw   d'15'          ;Set check data
       subwf   rx_substatus,w ;Substatus - check data
       btfss   status,z       ;Substatus = 15 ?
       goto    st25           ;No.
       goto    st_on          ;Input signal ON ?
st25   movlw   d'16'          ;Set check data
       subwf   rx_substatus,w ;Substatus - check data
       btfss   status,z       ;Substatus = 16 ?
       goto    st26           ;No.
       btfss   porta,ra1      ;B5 = 1 ?
       goto    stinc          ;Jump to Substatus + 1
       bsf     cont_data,5    ;Set B5 = 1
       goto    stinc          ;Jump to Substatus + 1
st26   movlw   d'17'          ;Set check data
       subwf   rx_substatus,w ;Substatus - check data
       btfss   status,z       ;Substatus = 17 ?
       goto    st27           ;No.
       goto    st_off         ;Input signal OFF ?
st27   movlw   d'18'          ;Set check data
       subwf   rx_substatus,w ;Substatus - check data
```

```
        btfss   status,z        ;Substatus = 18 ?
        goto    st28            ;No.
        goto    st_on           ;Input signal ON ?
st28    movlw   d'19'           ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 19 ?
        goto    st29            ;No.
        btfss   porta,ra1       ;B6 = 1 ?
        goto    stinc           ;Jump to Substatus + 1
        bsf     cont_data,6     ;Set B6 = 1
        goto    stinc           ;Jump to Substatus + 1
st29    movlw   d'20'           ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 20 ?
        goto    st30            ;No.
        goto    st_off          ;Input signal OFF ?
st30    movlw   d'21'           ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 21 ?
        goto    st31            ;No.
        goto    st_on           ;Input signal ON ?
st31    movlw   d'22'           ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 22 ?
        goto    st120           ;No. Substatus=23
        btfss   porta,ra1       ;B7 = 1 ?
        goto    stinc           ;Jump to Substatus + 1
        bsf     cont_data,7     ;Set B7 = 1
        goto    stinc           ;Jump to Substatus + 1
st32    btfsc   porta,ra1       ;Input signal OFF ?
        goto    illegal         ;No. Jump to illegal
        clrf    rx_substatus    ;SC substatus = 0
```

```
        incf    rx_status,f     ;SC status = 2
        goto    int_end         ;End of interruption

st_on
        btfss   porta,ra1       ;Input signal ON ?
        goto    illegal         ;No. Jump to illegal
        goto    stinc           ;Jump to Substatus + 1
st_off
        btfsc   porta,ra1       ;Input signal OFF ?
        goto    illegal         ;No. Jump to illegal
        goto    stinc           ;Jump to Substatus + 1

;************** End data check Process ****************
stchk2  btfss   porta,ra1       ;Input signal ON ?
        goto    illegal         ;No. Jump to illegal
        movlw   d'2'            ;Set check data
        subwf   rx_substatus,w  ;Substatus - check data
        btfss   status,z        ;Substatus = 2 ?
        goto    stinc           ;Jump to Substatus + 1

;**************** Data check Process ******************
        movf    cont_data,w     ;Read control data
        xorlw   ptn1            ;Check Pattern1
        btfss   status,z        ;Data = Pattern1 ?
        goto    dtchk0          ;No.
        bcf     portb,rb3       ;
        bsf     portb,rb5       ;
 goto    dtchk1         ;Jump to ON counter set
dtchk0
        movf    cont_data,w     ;Read control data
        xorlw   ptn2            ;Check Pattern2
        btfss   status,z        ;Data = Pattern2 ?
```

```
        goto    illegal         ;No. Jump to illegal
        bcf     portb,rb3       ;
        bsf     portb,rb5       ;
dtchk1  movlw   d'60'           ;Set 60sec
        movwf                   ;Save  PORT counter

;*****************  Illegal Process  ********************
illegal incf    rx_edge,f       ;Edge check flag ON
        clrf    rx_substatus    ;RX substatus = 0
        clrf    rx_status       ;RX status = 0
        clrf    code_ck         ;Clear code check flag
        goto    int_end         ;End of interruption

;***********  Substatus Increment Process  **************
stinc   incf    rx_substatus,f  ;Substatus + 1

;********  End of Timer Interruption Process  ***********
int_end movlw   d'100'          ;256-10000us/60sec = 100
        movwf   tmr0            ;Set 60sec to TMR0
        movf    s_save,w        ;Read saved STATUS reg
        movwf   status          ;Recover STATUS reg
        swapf   w_save,f        ;Read saved W register
        swapf   w_save,w        ;Recover W register
        retfie                  ;End of interruption
```

REFERENCES

Alkar, A. Z. & Buhur, U. (2005). An Internet Based Wireless Home Automation System for Multifunctional Devices. *IEEE Consumer Electronics*, 51(4), 1169-1174.

Delgado, J.C.M. & Nunes, J.C. (2000). An Internet application for home Automation, Electrotechnical Conference; *MELECON 10th Mediterranean*, Vol. 1, pp. 298 -301.

Fujiyama, H. (2006). System-on-a-chip with security modules for network home electric appliances, *Fujitsu Scientific and Technical Journal, 2 System-on-a-Chip,* vol. 42 pp. 227-233.

Neng-Shiang Liang, Li-Chen Fu & Chao-Lin Wu. (2002). An integrated, flexible, and Internet-based control architecture for home automation System in the Internet era*, Proceedings ICRA. IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 1101 –1106.

Luo Ren C., Hsu Te Y., Lin Tung Y. & Su Kuo L.(2005). The development of intelligent home security robot: *Proceedings of the IEEE International Conference on Mechatronics*, ICM '05, pp. 422-427.

Rifat, S. Enamul, H., S.M. Sohan & Iftekhar Naim. (2008). Remote Controlling of Home Appliances using Mobile Telephony, *International Journal of Smart Home,* Vol. 2, No. 3.

Sriskanthan, N. & Tan, K. (2002). Bluetooth Based Home Automation Systems*, Journal of Microprocessors and Microsystems,* Vol. 26, pp. 281-289.

Yang Lili, Yang Shuang-Hua, & Yao Fang. (2007). Safety and security of remote monitoring and control of intelligent home environments, *Proceedings - IEEE International Conference on Systems, Man and Cybernetics* ,Vol. 2 pp 1149-1153.