Design, Analysis And Fabrication Of Android To Control Household Appliances

L. Naveen[1], , S. K. Das[*] and J. K. Moharana[**]
Dept.of EEE, GITA, Bhubaneswar, Odisha

***Abstract*:** Presently modern houses are packed with electric power driven gadgets, may it for cooking, comfort living, hobby or entertainment etc. Along with the advancement of time, engineers & hobbyists are coming with different varieties of control methodology for controlling home appliances. Presently big business & research houses are entering the field such as Google, Microsoft etc. Though enough work has been done in this field conceptually and experimentally, a viable & reliable product yet to reach a common man's house with affordable price. The much hyped "Google Glass" is yet to hit the market. The trend of thinking in this field still going to use mobile hand set to control appliances or network of appliances. In this paper we are going to present a simple and affordable concept of wirelessly controlling household appliances using a android enables mobile phone, few electronics & electrical components connected wirelessly by Bluetooth.

Keywords: Android, Bluetooth, home appliances, DTMF, GSM and SMS

I.INTRODUCTION

A lot of work has been done for remote control of home & office appliances. Many different approaches have been made to achieve the goal. In [1] the basic methodology is to use infrared remotes to control the various appliances. Different appliances provide their controlling remote along with their products. Also handhelds are available for making fans & lights on/off. The companies like Legrand and Gold Medal already started these kinds of control system and they are at present available in the market. In this method the control of home appliances can be done even though when we are elsewhere just by using the DTMF tone generated when the user pushes mobile phone keypad buttons or when connected to a remote mobile. In [3] this is another type of home appliance control system where the person must be present in sight to the appliance that is needed to be controlled and a predefined gesture must be used to turn on the device and another gesture must be used by us to turn off the device. The performance of the proposed system is done with a hardware embedded in that particular device. In this system, the control of home appliances can be done from a remote area with GSM & SMS communication system. This system is accomplished by personal computers, interface cards, radio transmitters and receivers, microprocessors, ac phase control circuits, along with window-type software and microprocessor control software. The main aim of this system [4] [5] is to establish a wireless connection between a client computer and a server in a real application. Via the wireless communication system, this system utilizes the remote voice *recognisation* server to translate the voice input received from a serviced client computer in to a symbolic data file to be processed by the client's computer. The Wi-Fi network and the computer server utilize the GUI manager for implementing the speed reorganization system. [2] and [6] in this method the devise control the home appliances through voice commands. For that purpose a mobile application is developed that convert the user voice command into SMS and send through GSM network. Such application is developed using java for mobile technology and MPLAB for microchip family of controller. This proposed system is affordable to everyone, cheap and easy to install and useful for disable / old peoples. Apart from these [4] there are many proposed methodology of controlling home appliances involving complex networking of appliances and embedded hardware. We are here proposing and experimenting a simple concept which can be implemented using a android enabled mobile phone, some simple electronics components, relays and converted DC power supply.

II.DESIGN ASPECT & COMPONENT CHOICE

For remote control of home appliances first we have to choose a android enabled mobile phone. We have chosen android as the operating system following reasons

i. About Android: World is contracting with the growth of mobile phone technology. As the number of users is increasing day by day, facilities are also increasing. Starting with simple regular handsets which were used just for making phone calls, mobiles have changed our lives and have become part of it. Now they are not used just for making calls but they have innumerable uses and can be used as a Camera, Music player, Tablet PC, T.V. , Web browser etc and with the new technologies, new software and operating systems are required. Operating Systems have developed a lot in last 15 years. Starting from black and white phones to recent smart phones or mini computers, mobile OS has come far away. Especially for smart phones, Mobile operating system

(OS) has greatly evolved from Palm OS in 1996 to Windows pocket PC in 2000 then to Blackberry OS and Android. One of the most widely used mobile OS these days is **ANDROID. Android** does a software bunch comprise not only operating system but also middleware and key applications. Android Inc. was founded in Palo Alto of California, U.S. by Andy Rubin, Rich miner, Nick sears and Chris White in 2003. Later Android Inc. was acquired by Google in 2005. After original release there have been number of updates in the original version of Android. Unveiling of the Android platform was announced on 5 November 2007 as an open-source software stack for mobile devices, and a corresponding open-source project led by Google. Its goal was to make sure that there was no central point of failure, so that no industry player can restrict or control the innovations of any other. Google had lined up a series of hardware component and software partners and signalled to carriers that it was open to various degrees of cooperation on their part. On November 5, 2007, the Open Handset Alliance, a consortium of technology companies including Google, device manufacturers such as HTC and Samsung, wireless carriers such as Sprint Nextel and T-Mobile, and chipset makers such as Qualcomm and Texas Instruments, unveiled itself, with a goal to develop open standards for mobile devices. That day, Android was unveiled as its first product, a mobile device platform built on the Linux kernel version 2.6. The first commercially available phone to run Android was the HTC Dream, released on October 22, 2008. Since 2008, Android has seen numerous updates which have incrementally improved the operating system, adding new features and fixing bugs in previous releases. Each major release is named in alphabetical order after a dessert or sugary treat; for example, version 1.5 Cupcake was followed by 1.6 Donut. The latest release is 4.2 Jelly Bean. In 2010, Google launched its Nexus series of devices - a line of smartphones and tablets running the Android operating system, and built by a manufacturer partner. HTC collaborated with Google to release the first Nexus Smartphone, the Nexus One. The series has since been updated with newer devices, such as the Galaxy Nexus phone and Nexus 7 tablet, made by Samsung and Asus respectively. Google releases the Nexus phones and tablets to act as their flagship Android devices, demonstrating Android's latest software and hardware features.

ii.Android Features

1. UI refinements for simplicity and speed: The user interface is refined in many ways across the system, making it easier to learn, faster to use, and more power-efficient. A simplified visual theme of colors against black brings vividness and contrast to the notification bar, menu and other parts of the UI.

2. Faster, more intuitive text input: The Android soft keyboard is redesigned and optimized for faster text input and editing. The keys themselves are reshaped and repositioned for improved targeting, making them easier to see and press accurately, even at high speeds. The keyboard also displays the current character and dictionary suggestions in a larger, more vivid style that is easier to read and adds the capability to correct entered words from suggestions in the dictionary.

3. One-touch word selection and copy/paste: When entering text or viewing a web page, the user can quickly select a word by press-hold, then copy to the clipboard and paste. With both the selection and cursor modes, no use of a trackball is needed.

4. Improved power management: The Android system takes a more active role in managing apps that are keeping the device awake for too long or that are consuming CPU while running in the background. By managing such apps — closing them if appropriate — the system helps ensure best possible performance and maximum battery life. The Application settings provide an accurate overview of how the battery is being used, with details of the usage and relative power consumed by each component or application.

5. Control over applications: A shortcut to the Manage Applications control now appears in the Options Menu in the Home screen and Launcher, making it much easier to check and manage application activity. Once the user enters Manage Applications, a new Running tab displays a list of active applications and the storage and memory being used by each. The user can read further details about each application and if necessary stop an application or report feedback to its developer.

6. New ways of communicating, organizing: An updated set of standard applications lets the user take new approaches to managing information and relationships**.**

7. Internet calling: The user can make voice calls over the internet to other users who have SIP accounts. The user can add an internet calling number (a SIP address) to any Contact and can initiate a call from Quick Contact or Dialer. To use internet calling, the user must create an account at the SIP provider of their choice — SIP accounts are not provided as part of the internet calling feature. Additionally, support for the platform's SIP and internet calling features on specific devices is determined by their manufacturers and associated carriers.

8. Near-field communications**:** An NFC Reader application lets the user read and interact with near-field communication (NFC) tags. For example, the user can “touch” or “swipe” an NFC tag that might be embedded in a poster, sticker, or advertisement, then act on the data read from the tag. A typical use would be to read a tag at a restaurant, store, or event and then rate or register by jumping to a web site whose URL is included in the tag data. NFC communication relies on wireless technology in the device hardware, so support for the platform's NFC features on specific devices is determined by their manufacturers.

9. Downloads management: The Downloads application gives the user easy access to any file downloaded from the browser, email, or another application. Downloads is built on a completely new download manager facility in the system that any other applications can use, to more easily manage and store downloads.
10. Network connectivity*:* It supports wireless communications using Bluetooth, GSM mobile-phone technology, 3G, Edge and 802.11 Wi-Fi networks. ANDROID has an ability to report the location of Wi-Fi access points, encountered as phone user's move around, builds databases containing the physical locations of hundreds of millions of such access points. These databases form electronic maps to locate smartphones, allowing them to run apps like Foursquare, Google Latitude, Face book Places, and to deliver location-based ads.
11. Security and privacy: Android is a multi-process system, in which each application (and parts of the system) runs in its own process. Most security between applications and the system is enforced at the process level through standard Linux facilities, such as user and group IDs that are assigned to applications. Android applications run in a sandbox, an isolated area of the system that does not have access to the rest of the system's resources, unless access permissions are explicitly granted by the user when the application is installed. Before installing an application, the Play Store displays all required permissions: a game may need to enable vibration or save data to an SD card, for example, but should not need to read SMS messages or access the phonebook. After reviewing these permissions, the user can choose to accept or refuse them, installing the application only if they accept.
12. Malware antivirus and virus: Several security firms, such as Lookout Mobile Security, AVG Technologies, and McAfee, provide antivirus for Android devices. This software is ineffective as sandboxing also applies to such applications, limiting their ability to scan the deeper system for threats.

iii.Programming Software :

For controlling various home appliances we will need a software programme for creating user interface, controlling data, connectivity of the mobile phone to appliances to be controller. This is for sender or controlling side. The software requirement has been categorized as follows

i. Android SDK: A software development kit (SDK or "devkit") is typically a set of software development tools that allows for the creation of applications for a Android software package, software framework, hardware platform, computer system, video game console or operating system

ii. Eclipse IDE: An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of a source code editor, build automation tools and a debugger.

iii. Java JDK: Java – officially supported but C/C++ is also possible but not supported

All these programming software are available free in the internet to be downloaded.

iv.Wireless connectivity

As stated above Android has connectivity functionality both of Wi-Fi & Bluetooth. For our requirement both connectivity protocols will work. But for economic and off-the-self availability of the transreceiver component we have chosen Bluetooth as our wireless connectivity protocol.

v.Bluetooth

Bluetooth operates in the range of 2400–2483.5 MHz (including guard bands). This is in the globally unlicensed Industrial, Scientific and Medical (ISM) 2.4 GHz short-range radio frequency band. Bluetooth uses a radio technology called frequency-hopping spread spectrum. The transmitted data is divided into packets and each packet is transmitted on one of the 79 designated Bluetooth channels. Each channel has a bandwidth of 1 MHz. The first channel starts at 2402 MHz and continues up to 2480 MHz in 1 MHz steps. It usually performs 1600 hops per second, with Adaptive Frequency-Hopping (AFH) enabled.

Originally Gaussian frequency-shift keying (GFSK) modulation was the only modulation scheme available; subsequently, since the introduction of Bluetooth 2.0+EDR, π/4-DQPSK and 8DPSK modulation may also be used between compatible devices. Devices functioning with GFSK are said to be operating in basic rate (BR) mode where an instantaneous data rate of 1 Mbit/s is possible. The term Enhanced Data Rate (EDR) is used to describe π/4-DPSK and 8DPSK schemes, each giving 2 and 3 Mbit/s respectively. The combination of these (BR and EDR) modes in Bluetooth radio technology is classified as a "BR/EDR radio".

Bluetooth is a packet-based protocol with a master-slave structure. One master may communicate with up to 7 slaves in a piconet; all devices share the master's clock. Packet exchange is based on the basic clock, defined by the master, which ticks at 312.5 µs intervals. Two clock ticks make up a slot of 625 µs; two slots make up a slot pair of 1250 µs. In the simple case of single-slot packets the master transmits in even slots and receives in odd slots; the slave, conversely, receives in even slots and transmits in odd slots. Packets may be 1, 3 or 5 slots long but in all cases the master transmits will begin in even slots and the slave transmits in odd slots. Bluetooth provides a secure way to connect and exchange information between devices such as faxes, mobile phones, telephones, laptops, personal computers, printers, Global Positioning System (GPS) receivers, digital cameras, and video game consoles. It was principally designed as a low-bandwidth technology.

vi.Communication and connection

A master Bluetooth device can communicate with a maximum of seven devices in a piconet (an ad-hoc computer network using Bluetooth technology), though not all devices reach this maximum. The devices can switch roles, by agreement, and the slave can become the master (for example, a headset initiating a connection to a phone will necessarily begin as master, as initiator of the connection; but may subsequently prefer to be slave).

The Bluetooth Core Specification provides for the connection of two or more piconet to form a scatternet, in which certain devices simultaneously play the master role in one piconet and the slave role in another.

At any given time, data can be transferred between the master and one other device (except for the little-used broadcast mode. The master chooses which slave device to address; typically, it switches rapidly from one device to another in a round-robin fashion. Since it is the master that chooses which slave to address, whereas a slave is (in theory) supposed to listen in each receive slot, being a master is a lighter burden than being a slave. Being a master of seven slaves is possible; being a slave of more than one master is difficult. The specification is vague as to required behaviour in scatternets. Many USB Bluetooth adapters or "dongles" are available, some of which also include an IrDA adapter. Older (pre-2003) Bluetooth dongles, however, have limited capabilities, offering only the Bluetooth Enumerator and a less-powerful Bluetooth Radio incarnation. Such devices can link computers with Bluetooth with a distance of 100 meters, but they do not offer as many services as modern adapters do.

III. HARDWARE REQUIREMENT

Until now we have discussed the mobile phone programming & its connectivity to the controlling device to be attached to the appliances to be controlled. First thing is that the controlling component will receive the information sent from the mobile phone through Bluetooth, the decode it and through a microcontroller send the appropriate signal to the relay to be implemented. Also primarily we will need a power supply for the controlling unit. For all these the hardware and software requirement will be as below

1. Step-down transformer-(E10048-19F, 220/240V, 12V, 500mA, 1:10)
2. Diode- (1N4007, 30V, 1A)
3. Capacitor- (1000 Microfarad, 25V, 820mA)
4. Voltage regulator- (IC LM7805, 5V, 2.2A)
5. Resister- 1K
6. LED- 1.5V
7. n-p-n Transistor-(BC547, 50V, 0.1A, 1.5W)
8. Relay- (SPDT Type, 12Vdc, 5V)
9. Bluetooth transreceiver
10. ATmega8 Microcontroller- (28 PIN, 5 V I/P, 8 KB)

IV.SOFTWARE FOR MICROCONTROLLER PROGRAMMER

i.AVR STUDIO 4: Assembly languages are a type of low level language for programming computers, microprocessors, microcontroller, and other (usually) integrated circuits. They implement a symbolic representation of the numeric machine codes and other constants needed to program a particular CPU architecture. This representation is usually defined by the hardware manufacturer, and is based on abbreviations (called mnemonics) that help the programmer remember individual instructions, registers, etc. An assembly language family is thus specific to certain physical (or virtual) computer architecture. This is in contrast to most high-level languages, which are (ideally) portable.

ii.WIN AVR : The WINAVR tool kit includes three main tools, assembler, compiler and linker. An assembler is used to assemble the ARM assembly program. A compiler is used to compile the C source code into an object file. A linker is used to create an absolute object module suitable for our in-circuit emulator.

iii.SINA PROG: Sina programmer is a software to compile the program.

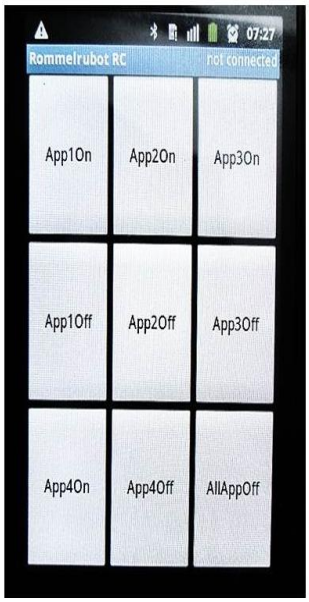
*iv.EMBEDDED C***:** Because most embedded projects have severe cost constraints, they tend to use low-cost processors like the 8051 family of devices considered in this book. These popular chips have very limited resources available most such devices have around 256 bytes (not megabytes!) of RAM, and the available processor power is around 1000 times less than that of a desktop processor. As a result, developing embedded software presents significant new challenges, even for experienced desktop programmers. If you have some programming experience - in C, C++ or Java - then this book and its accompanying CD will help make your move to the embedded world as quick and painless as possible. It should be noted that many SOC (System on a chip) for this are available such as CC2541 from Texas Instrument which have a 51 microcontroller along with blue tooth transreceiver. But for simplicity we have chosen a Bluetooth receiver and a general purpose microcontroller.

V.EXPERIMENTAL SETUP

Step 1- First the android enabled cell phone (as given in Fig.1) is programmed as per our requirement. We developed a GUI from where we can communicate & control the appliances. The Fig.1 shows the screenshot of the user interface. Also we have to send a message from the GUI to the control section by blue-tooth. A step by step procedure for developing and GUI and attaching to various functionality such a as blue tooth transmission is provided in the web site www.developer.android.com. This provides the process of how to install android & eclipse etc. related software in your computer. Following the steps

provided any one can develop an android programme and download it to the target handset.

Step 2 – For the controlling unit first a power supply system is developed whose circuit diagram is shown below as Fig.2. Then microcontroller is written with developed code to do the need full. The output of the blue tooth transreceiver is set to the microcontroller to be processed and activate the corresponding relay switch to be made on or off. The main circuit having the controlling unit and blue troth receiver, microcontroller, relays is shown in Fig.3. The detail circuit for relay operation is shown as Fig.4.



1.App1On -> Appliance 1 On

2. App2On -> Appliance 2 On

3. App3On -> Appliance 3 On

4. App1Off-> Appliance 1 Off

5. App2Off-> Appliance 2 Off

6. App3Off-> Appliance 3 Off

7. App4On-> Appliance 4 On

8. App4Off-> Appliance 4 Off

9. AllAppOff-> All Appliance Off

Fig.1: Programmed cell phone

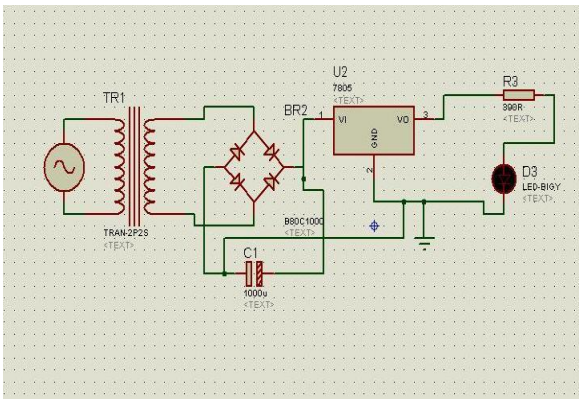


Fig.2: Circuit diagram for power supply

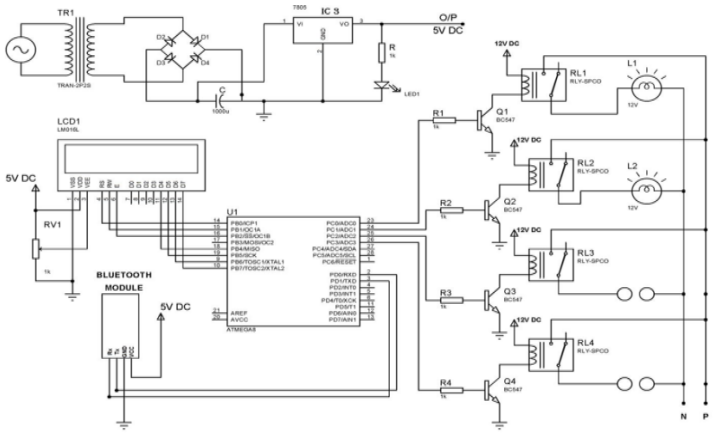


Fig.3: Circuit diagram for Relay operation

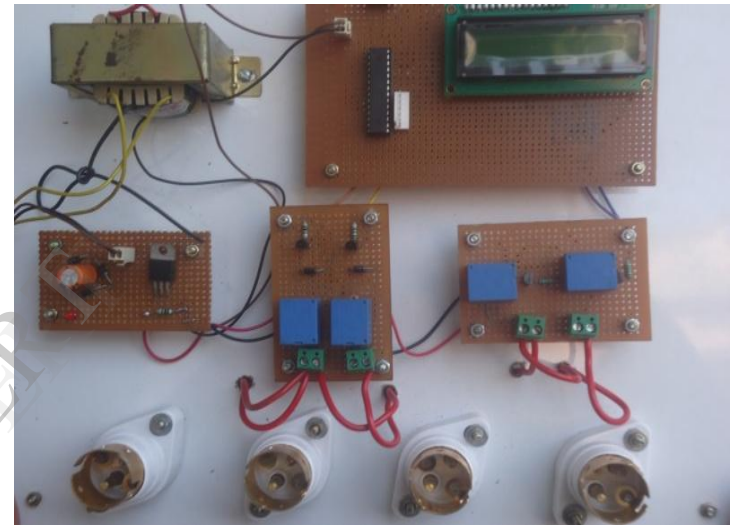


Fig.4: Experimental set up for relay portion

VI.EXPERIMENTAL RESULT

On completing the set we experimented with 5 nodes to make it on or off. It is found out that the set up worked perfectly well. The results are encouraging and validated the topology. The Fig.5 shows the total experimental set up in which all appliances, power supply with remote controller and Android controller have been included. One of the applications has been operated on remote control which has been depicted in Fig.6. Similarly two appliances operated at a time by remote control given in Fig.7.

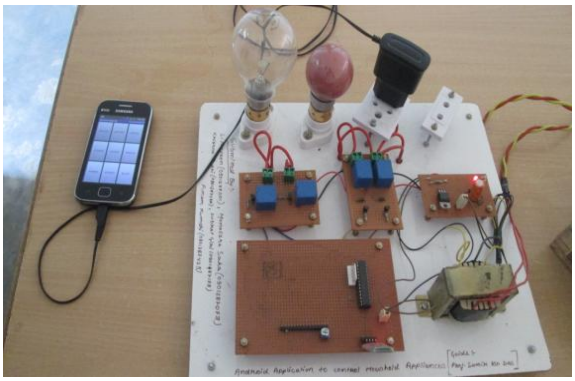


Fig.5: Experimental set up for the topology

Fig.6: One of applications is ON by remote control

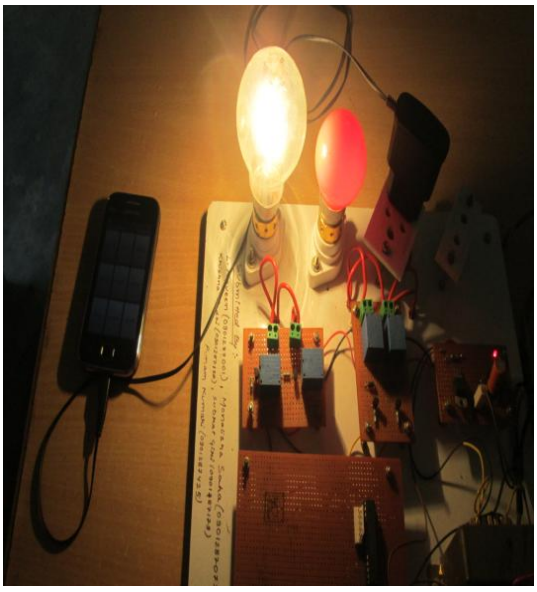


Fig.7: Two applications are ON by remote control

VII.CONCLUSIONS

As we have done it experimentally we have chosen the component available locally & cheap. Instead of blue tooth wireless communication Zigbee or Wi-fi connectivity can be used. More over many readymade modules are already available for controlling the appliances with iphone such as X10, Z-wave etc. Further work can be done in this field making it more reliable & economical. Data communication both way can be more appreciated so that user can know the status of the appliance. Actually the user always wants a stable product off the self of the shop. So this process of developing a home appliance control is the work of a consortium comprising of appliance manufacturers, abiding by a common specification. We have only shown a possible methodology which can achieve the goal

ACKNOWLEDGEMENT

This paper is based on Project done by L.Naveen, **S.Giri, M.Saha, P.Kumari, K.Padhi** at GITA, Bhubaneswar, Odisha in Electrical and Electronics Department, 2013.

REFERENCE

1.Hari Charan Tadimeti, Manas Pulipati, "Overview of Automation Systems and Home Appliances Control using PC and Microcontroller", International Journal of Science and Research (IJSR), India Online ISSN: 2319-7064, Volume 2 Issue 4, April 2013

2.Faisal Baig, Saira Beg and Muhammad Fahad Khan, "Zigbee Based Home Appliances Controlling Through Spoken Commands Using Handheld Devices", International Journal of Smart Home Vol. 7, No. 1, January, 2013

3.C. K. Das, M. Sanaullah, H. M. G. Sarower and M. M. Hassan, "Development of a cell phone based remote control system: an effective switching system for controlling home and office appliances", International Journal of Electrical & Computer Sciences IJECS Vol: 9 No: 10.

4.Tanvir A. Khan, Yasir Saleem and M. Saleem "Xtreme Application: A User Interface System for Home Appliances with Cellular Phones" International Journal of Computer Science and Telecommunications [Volume 1, Issue 1, November 2010]

5.R. Chutia, D. Sonowal and S. Sharma, "Remote Household Appliance Control System Using GSM", Proc. of the International Conference on Advanced Computing and Communication Technologies (ACCT 2011), ISBN: 978-981-08-7932-7

6.Deepali Y Loni *Assistant Professor, Department of ElectronicsD.K.T.Es' Textile & Engineering Institute,Ichalkaranji. Post code:416115, India., "DSP Based Speech Operated Home Appliances Using Zero Crossing Features"* deepaliloni@rediffmail.com

7.Ahmed ElShafee, Karim Alaa Hamed, "Design and Implementation of a WiFi Based Home Automation

System" World Academy of Science, Engineering and Technology 68 2012
8.Milan Z. Bjelica, Nikola Teslic, Zoran Jovanovic and Zoran Marceta, "Home Appliances as Home Controllers: Concepts and Set-Top Box Implementation", Telfor Journal, Vol. 2, No. 2, 2010.
9.Rifat Shahriyar, Enamul Hoque, S.M. Sohan, Iftekhar Naim,Md. Mostafa Akbar & Masud Karim Khan, "Remote Controlling of Home Appliances using Mobile Telephony", International Journal of Smart Home'Vol. 2, No. 3, July, 2008