IJARST
International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

HOME AUTOMATION USING BLUETOOTH APP

K. Madhavi[1], Kranthi kumar[2]
Assistant Professor[1,2]
Department of ECE
Malla Reddy Engineering College

Abstract- Electronic devices and appliances have become very common in this recent year of technology especially with fast development in smartphones. In this project, the design of Home Automation System compatibly with Local housing and good features for home automation via remote access are presented.Bluetooth Based Home Automation System Using Android and Arduino is design and implemented. In this project work a part of smart home technology which using Bluetooth in a mobile device is used, so it will cheap and efficient to use. This project describes about home automation system which would be used to enable home lighting, garage door motor, water pumping motor and smoke detection using a smart phone application with Bluetooth wireless technology.The system included three main components: an Arduino microcontroller for connecting the appliances, a Bluetooth module for signal transfer, and a smartphone with the Android application to control home appliances. Bluetooth communication technology and controlled system is that the operating range is low but it can controlled from anywhere inside of home, By using smart phone application we can control household appliances and provide security to decrepit people.

The main idea of this project is to control home appliances to avoid the dangerous of electric shock and convenience of decrepit and physically disable people, who can easily access and control the home appliances by staying at particular place and access them remotely without the help of other people. By using this system, our home automation works smartly by providing increased quality of life, and comforts to users.

Index Terms- Home Automation, Bluetooth Automation, Android based automation

I. INTRODUCTION

Nowadays,we have remote controlsfor our televisionsetsandother electronicsystems,whichhavemadeourlivesrealeasy.ArduinobasedHomeAutomationusingRemotecontrolistheExistingsystemandasimpleproject,whereanoldTVRemoteisusedtocontroldifferent appliances. Have you ever wondered about home automation which would give thefacility of controlling tube lights, fans and other electrical appliances at home using a remotecontrol?Off-course,Yes!But,aretheavailableoptionscost-effective?IftheanswerisNo,wehavefound asolution to it. We have come up with a new system called Arduino based home automation usingBluetooth. This system is super-cost effective and can give the user, the ability to control anyelectronic device without even spending for a remote control. This project helps the user tocontrol all the electronic devices using his/her smartphone. Time is a very valuable thing.Everybodywants to savetime as much as they can. Newtechnologiesarebeingintroducedto saveourtime.Tosavepeople’stimeweareintroducing Home Automation system using Bluetooth.Home Automation is a concept wherea single device is used to control many aspects of a home like switching on and off differentappliances, monitoring temperature, fire alarms, garage doors etc. In this project, a remotecontrolis used to control switches.With the help of this system you can control your home appliances from your mobilephone.You can turn on/offyourhomeapplianceswithinthe rangeof Bluetooth.

Objectiveoftheproject

The Objective of this project is to implement a low cost, reliable and scalable homeautomationsystemthat canbeused toremotelyswitch onoroff anyhouseholdappliance.ThedesignofproposedmethodisbasedonArduinoboard,Bluetoothmodule,sensorsand smartphone application. Bluetooth module HC-05 is interfaced with Arduino board andhome appliances are connected with Arduino board via relay. Single device like smart phoneoperates all the appliances in thehomeusing Bluetooth

EmbeddedSystems

The Embedded systems are the computer systems that are designed for anyparticularfunctionortask.Anembeddedsystemisaspecial-purposecomputersystemdesigned to perform one or a few dedicated functions, sometimes with real-time computingconstraints. It is usually embedded as part of a complete device including hardware andmechanical parts. In contrast, a general-purpose computer, such as a personal computer, cando many different tasks depending on programming. Embedded systems have become veryimportant today as they control many of the common devices we use. Since the embeddedsystem is dedicated to specific tasks, design engineers can optimize it, reducing the size andcostoftheproduct,orincreasingthereliabilityandperform ance.Someembeddedsystemsaremass-produced,benefiting from economies of scale.

Physically,embeddedsystemsrangefromportabledevicessuchasdigitalwatchesandMP3 players, to large stationary installations like traffic lights, factory controllers, or thesystemscontrollingnuclearpowerplants.Complexityvariesfromlow,withasinglemicrocontroller chip, to very high with multiple units, peripherals and networks mountedinsidealargechassis or enclosure.

In general, "embedded system" is not an exactly defined term, as many systems havesome element of programmability. For example, Handheld computers share some elementswithembeddedsystemssuchastheoperatingsystemsandmicroprocessorswhichpowerthembutarenottrulyembeddedsystems,becausetheyallowdifferentapplicationstobeloadedandperipheralsto beconnected.

The uses of embedded systems are virtually limitless, because every day new products areintroduced to the market that utilizes embedded computers in novel ways. In recent years,hardware such as microprocessors, microcontrollers, and FPGA chips have become muchcheaper. So when implementing a new form of control, it's wiser to just buy the generic chipand write your own custom software for it. Producing a custom-made chip to handle aparticulartask or set of tasks costs far moretime and money.

Theusesofembeddedsystemsarevirtuallylimitless,becauseeverydaynewproductsareintroduced tothe marketthatutilizes embeddedcomputers innovel ways.In recentyears,hardware such as microprocessors, microcontrollers, and FPGA chips have become muchcheaper. So when implementing a new form of control, it's wiser to just buy the generic chipand write your own custom software for it. Producing a custom-made chip to handle aparticulartaskorsetoftaskscostsfarmoretimeandmoney.Manyembeddedcomputersevencomewithextensivelibraries,sothat"writingyourownsoftware"becomesaverytrivialtaskindeed. From an implementation viewpoint, there is a major difference between a computerandanembeddedsystem.EmbeddedsystemsareoftenrequiredtoprovideReal-Timeresponse. The main elements that make embedded systems unique are its reliability and easein debugging.

Debugging:

Embedded debugging may be performed at different levels, depending on thefacilities available. From simplest to most sophisticate they can be roughly grouped into thefollowingareas:

- Interactive resident debugging, using the simple shell provided by the embeddedoperatingsystem (e.g.Forth and Basic)
- ExternaldebuggingusingloggingorserialportoutputtotraceoperationusingeitheramonitorinflashorusingadebugserverliketheRemedyDebuggerwhich evenworksforheterogeneous multi coresystems.
- An in-circuit debugger (ICD), a hardware device that connects to the microprocessorvia a JTAG or Nexus interface. This allows the operation of the microprocessor to becontrolledexternally,butistypicallyrestrictedtospecificdebuggingcapabilitiesintheprocessor.
- Anin-circuitemulatorreplacesthemicroprocessorwithasimulatedequivalent,providingfull control over all aspectsof the microprocessor.
- Acompleteemulatorprovidesasimulationofallaspectsofthehardware,allowingallofit tobecontrolled andmodified and allowingdebugging on anormal PC.

Unless restricted to external debugging, the programmer can typically load and runsoftwarethroughthetools,viewthecoderunningintheprocessor, andstartorstopitsoperation.Theview ofthecodemay beas assembly codeor source-code.

II. Existing Work or Literature Survey

HomeAutomationUsingBluetooth providestheautomaticcontrol onthehomeappliancesusing Smart phones which containstheBlynk application.This project describes about home automation system which would be used to enablehome lighting, garage door motor, water pumping motor and smoke detection using a smartphoneapplicationwithBluetoothwirelesstechnology. Thesystemincludedthreemaincomponents: an Arduino microcontroller for connecting the appliances, a Bluetooth moduleforsignaltransfer,andasmartphonewiththeAndroidapplicationtocontrolhomeappliances.TheblockdiagramofthisHomeautomationusingBluetoothwithalltheiressentialcomponentsis shown in figurebelow.

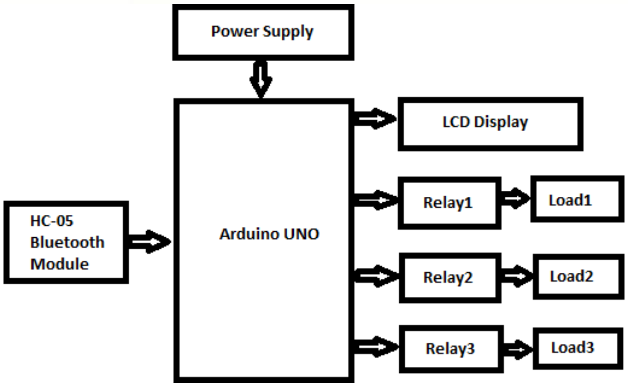

Fig2.1.0BlockDiagram of Project

III. Proposed Work

ARDUINO UNO

The Arduino is a family of microcontroller boards to simplify electronic design, prototyping and experimenting for artists, hackers, hobbyists, but also many professionals. People use it as brains for their robots, to build new digital music instruments, or to build a system that lets your house plants tweet you when they're dry

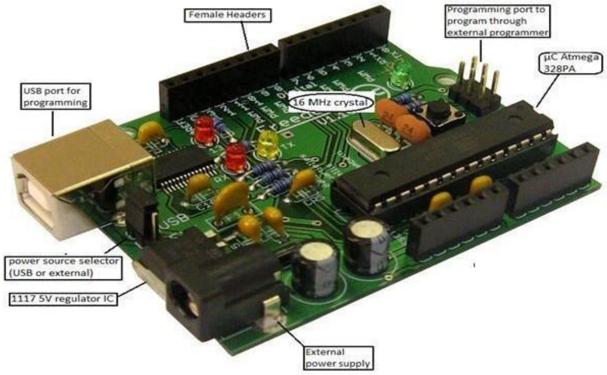

Fig 2.2.1: Arduino UNO

The Arduino Uno is an open source microcontroller based on the Microchip ATmega328P microcontroller and developed by Arduino.cc The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansionboards (shields) and other circuits.Arduinos (we use the standard Arduino Uno) are built around an ATmega microcontroller essentially a complete computer with CPU, RAM, Flash memory, and input/output pins, all on a single chip. Unlike, say, a Raspberry Pi, it's designed to attach all kinds of sensors, LEDs, small motors and speakers, servos, etc. directly to these pins, whichcan read in or output digital or analog voltages between 0 and 5 volts. The Arduino connects to your computer via USB, where you program it in a simple language (C/C++, similar to Java) from inside the free Arduino IDE by uploading your compiled code to the board. Once programmed, the Arduino can run with the USB link back to your computer, or stand-alone without it no keyboard or screen needed, just power.

Starting clockwise from the top center:
Analog Reference pin (orange)
Digital Ground (light green)
Digital Pins 2-13 (green)
Digital Pins 0-1/Serial In/Out - TX/RX (dark green) - These pins cannot be used for digital i/o (Digital Read and Digital Write) if you are also using serial communication (e.g: Serial.begin).
Reset Button - S1 (dark blue)
In-circuit Serial Programmer (blue-green)
Analog In Pins 0-5 (light blue)
Power and Ground Pins (power: orange, grounds: light orange)
External Power Supply In (9-12VDC) - X1 (pink)
Toggles External Power and USB Power (place jumper on two pins closest to desired supply) - SV1 (purple)
USB (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board) (yellow)

The Arduino is a family of microcontroller boards to simplify electronic design, prototyping and experimenting for artists, hackers, hobbyists, but also many professionals. People use it as brains for their robots, to build new digital music instruments, or to build a system that lets your house plants tweet you when they're dry. Arduinos (we use the standard Arduino Uno) are built around an ATmega microcontroller — essentially a complete computer with CPU, RAM, Flash memory, and input/output pins, all on a single chip. Unlike, say, a Raspberry Pi, it's designed to attach all kinds of sensors, LEDs, small motors and speakers, servos, etc. directly to these pins, which can read in or output digital or analog voltages between 0 and 5 volts. The Arduino connects to your computer via USB, where you program it in a simple language (C/C++, similar to Java) from inside the free Arduino IDE by uploading your compiled code to the board. Once programmed, the Arduino can run with the USB link back to your computer, or stand-alone without it — no keyboard or screen needed, just power.

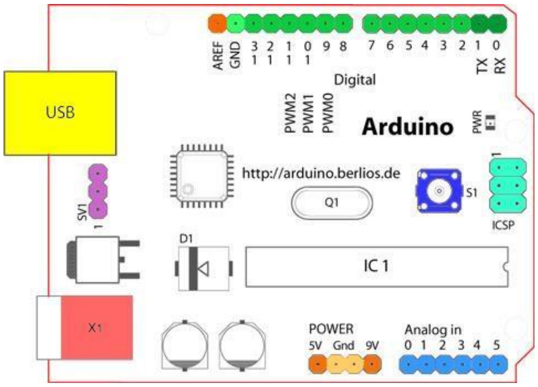

Figure 2.2.2: Structure of Arduino Board

Looking at the board from the top down, this is an outline of what you will see (parts of the board you might interact with in the course of normal use are highlighted)

Starting clockwise from the top center:
Analog Reference pin (orange)
Digital Ground (light green)
Digital Pins 2-13 (green)
Digital Pins 0-1/Serial In/Out - TX/RX (dark green) - These pins cannot be used for digital i/o (Digital Read and Digital Write) if you are also using serial communication (e.g. Serial.begin).
Reset Button - S1 (dark blue)
In-circuit Serial Programmer (blue-green)
Analog In Pins 0-5 (light blue)
Power and Ground Pins (power: orange, grounds: light orange)
External Power Supply In (9-12VDC) - X1 (pink)
Toggles External Power and USB Power (place jumper on two pins closest to desired supply) - SV1 (purple).

□ USB (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board) (yellow)

Digital Pins:

In addition to the specific functions listed below, the digital pins on an Arduino board can be used for general purpose input and output via the pin Mode(), Digital Read(), and Digital Write() commands. Each pin has an internal pull-up resistor which can be turned on and off using digital Write() (w/ a value of HIGH or LOW, respectively) when the pin is configured as an input. The maximum current per pin is 40mA.

□ Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. On the Arduino Diecimila, these pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip. On the Arduino BT, they are connected to the corresponding pins of the WT11 Bluetooth module. On the Arduino Mini and LilyPad Arduino, they are intended for use with an external TTL serial module (e.g. the Mini- USB Adapter).

□ External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt() function for details.

□ PWM: 3, 5, 6, 9, 10, and 11 Provide 8-bit PWM output with the analog Write() function. On boards with an ATmega8, PWM output is available only on pins 9, 10, and 11.

□ BT Reset: 7. (Arduino BT-only) Connected to the reset line of the bluetooth module.

□ SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

□ LED: 13. On the Diecimila and LilyPad, there is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

Analog Pins:

In addition to the specific functions listed below, the analog input pins support 10-bit analog-to-digital conversion (ADC) using the analog Read() function. Most of the analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. Analog inputs 6 and 7 (present on the Mini and BT) cannot be used as digital pins.

□ I2C: 4 (SDA) and 5 (SCL). Support I2C (TWI) communication using the Wire library (documentation on the Wiring website).

Power Pins:

□ VIN (sometimes labeled "9V"): The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. Also note that the Lily Pad has no VIN pin and accepts only a regulated input.

□ 5V: The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.

□ 3V3 (Diecimila-only) : A 3.3 volt supply generated by the on-board FTDI chip.

□ GND: Ground pins.

Other Pins:

□ AREF: Reference voltage for the analog inputs. Used with analog Reference().

□ Reset: (Diecimila-only) Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

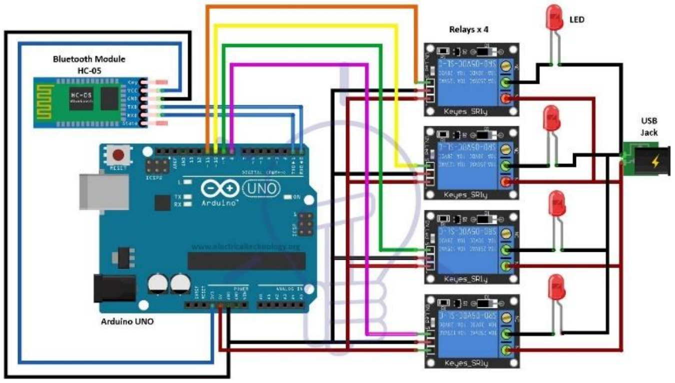

Fig3.1SchematicDiagram

Connections

Connections of this home automation system is designed which can be controlled byanysmartphone.Theautomationsystemconnectswiththes martphonethroughBluetooth.Thesmart phone sends control signals to switch home appliances ON or OFF by an android appthroughBluetoothinterface.TheprojectisbuiltonArduino UNOandisusedtocontrolLEDsand four home appliances connected to the Arduino through relays. The Arduino board isinterfacedto an HC-05 Bluetooth moduleto pair with the smart phone.An app named "Blynk" is used on the smart phone which is capable of sending textstrings to a paired device.The app will pair with the home automation system through HC-05Bluetooth Module. Every module has a unique MAC address and a password for pairing withotherdevices.LiketheBluetoothmoduleusedinthisprojec thadaMACaddress–98:D3:31:F4:18:22andhadapassword "1234" forpairingwith otherBluetoothdevices.TheArduinoboardreceivestheuserco mmandsintheformofnumbersfromthesmartphone through Bluetooth interface. These numbers are assigned to the home appliances andtheappliancesaretoggledeitherONorOFFonreceivingthe numericcommand.TheArduinosketch looks for the numeric commands from the Bluetooth module and operates relays to switch appliances.

3.1 Working

When the circuit is powered on, the Arduino loads the required libraries and switchesrelays to OFF position. Some initial messages are flashed on the LCD display and the statusofallthefourdevicesisshownOFFontheLCD .TheArduinowaitsforthenumericcommandto be received from the Bluetooth module. The four appliances are assigned numbers from 1to 4. If either number is obtained as a string from the Bluetooth module, the status of therespectiveapplianceistoggled.Bydefault,thepi nsconnectingtotherelayshaveaLOWlogicdriving therelays to switch the appliances OFF.

If an appliance is in OFF condition and number representing it is passed through theBluetooth app, the Arduino switches the logic at the respective pin to HIGH triggering therelay to switch the appliance ON. The change in the status of the appliance is updated on theLCD display and the LED indicating supply to the appliance starts glowing due to forwardbiasing.

If an appliance is in ON condition and number representing it is passed through theBluetooth app, the Arduino switches the logic at the respective pin to LOW driving the relayto switch the appliance OFF. The change in the status of the appliance is updated on the LCDdisplay and the LED indicating supply to the appliance stops glowing due to lack of forwardvoltage.

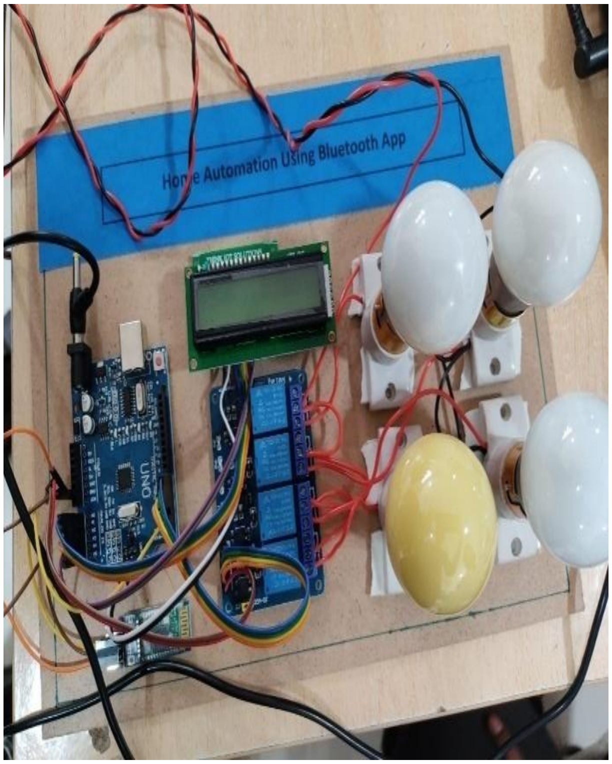

Fig4.1. ProjectKit

This is the project kit with all the connections attached.The Smart phone Pair with Bluetoothmodulewhich is used to controlthehome appliances by thecommands.

The numbers are transferred to the interfaced Bluetooth module from the paired smartphone. The smart phone must be paired with the Bluetooth module. Bluetooth Terminalusedto pass the numericcommands

IV. RESULTS AND DISCUSSION

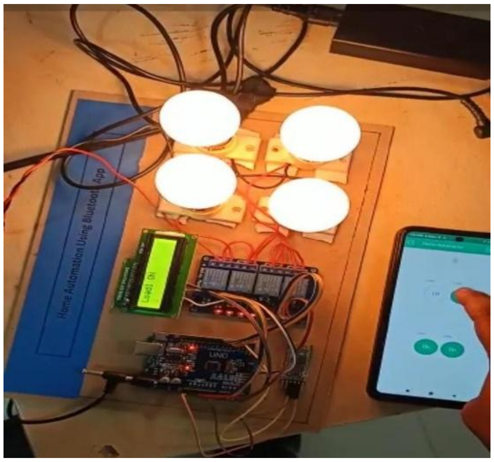

Fig4.2Output oftheproject

Output of the Home Automation Using Bluetooth App can be seen in the abovefigure. If an appliance is in ON condition and number representing it is passed through theBluetooth app, the Arduino switches the logic at the respective pin to LOW driving the relaytoswitch theapplianceOFF.Thechangeinthestatusof theapplianceis updatedon theLCDdisplay.

If an appliance is in OFF condition and number representing it is passed through theBluetooth app, the Arduino switches the logic at the respective pin to HIGH triggering therelayto switchthe applianceON. Thechangein thestatus oftheapplianceisupdated ontheLCD display.

V. CONCLUSION

Here, we have introduced design and implementation of a low cost, flexibleand wireless solution to the home automation. The system is secured for access from anyuserorintruder.Theusersareexpected toacquirepairing passwordforthe ArduinoBTandthe cell phone to access the home appliances. This adds a protection from unauthorizedusers.This system can be used as a test bed for any appliances that requires on-off switchingapplications without any internet connection.The full functionality of the home automationsystem was tested and the wireless communication between the cell phone and Arduino BTwasfoundtobelimitedinrangeof20meters answasreported tobeapplicable intheusage.This project can be further expanded by including some sensors like light sensors,temperaturesensors,safetysensorsetc.andautomaticallyadjustdifferentparameterslikeroomlightning, air conditioning, doorlocks etc.

REFERENCES

[1] Wireless Sensor Networks: Concepts, Applications, Experimentation and Analysis. 2016.p. 108. ISBN 9811004129. The use of standardized, with open standards over proprietary protocols provides the industry with the freedom to choose between suppliers with guaranteed interoperability. Standardized solutions usually have a much longer lifespan than proprietary solutions.

[2] Jump up^ "Research and Markets: Global Home Automation and Control Market 2014- 2020 - Lighting Control, Security & Access Control, HVAC Control Analysis of the $5.77 Billion Industry". Reuters. 2015-01-19. Archived from the original on 2016-05-05.

[3] Home Automation & Wiring (1 ed.). New York: McGraw-Hill/TAB Electronics. 1999- 03- 31. ISBN 9780070246744.

[4] Kenneth.J.Ayala, “The 8051 Microcontroller”,Cengage Learning,3rd Edition,2004.

[5] Microcontrollers(Theory & Applications)-A.V.Deshmuk,WTMH,2005.

[6] D.V.Hall, “Microprocessors and Interfacing”,TMH,2nd Edition,2006.

[7] Simon haykin, “Communication Systems”, wiley Publications,2nd Edition, 2007.

[8] Dr.Kamilo feher, “Wireless Digital Communications”, Prentice Hall of India, New Delhi,2003.