**Home Automation**

**Abstract**

Implementation of Home Automation using the latest technology gives us more convenience, security and safety.

Smartphone affordability increases every year and smartphones have begun to play important roles in our daily lives due to their size and portability. Google’s Android operating system (OS) is one of the leading and most preferred smartphones.

Controlling home appliances by using an Android phone gives users the ability to control their home appliances anywhere and at any time while at home and saves time spent in searching for the remote-control unit of Home Automation Systems since the user’s phone is usually kept close at hand.

This project presents the design and implementation of a low-cost prototype of a Bluetooth-based Home Automation System using an Android Phone.

The design uses an Atmel 8051 Microcontroller board, and the home appliances are physically connected to input/output ports of this board via relays.

**Keywords**

Home Automation, Android, Bluetooth,

Relays, 8051.

**Introduction**

Home Automation Systems are implemented to increase user convenience in the control of home appliances either via wired or wireless communication.

The word “wired” means that the home appliances are physically connected to a server or central controller to the Home Automation System, while “wireless” means that home appliances are connected wirelessly to a server or central controller or remote-control unit.

Home appliances such as TVs, DVD players, air conditioning and other devices come with their respective remote-control units.

When a Home Automation System is installed, there is the addition of at least one more remote-control unit.

Nowadays, people are inseparable from their smartphones.

A total of 5000 people from the US, UK, China, India, South Korea, South Africa, Indonesia and Brazil took a smartphone survey which was organized by Time magazine. This survey found that

the majority were highly attached to their phones, whereby 84% claimed that they could not go a single day without their phones. Android has 75% worldwide of the smartphone market share and a total of 136 million Android phones.

**Relays**

Relays are switches that open and close circuits electromechanically or electronically. Relays control one electrical circuit by opening and closing contacts in another circuit. As relay diagrams show, when a relay contact is normally open (NO), there is an open contact when the relay is not energized.

When a relay contact is Normally Closed (NC), there is a closed contact when the relay is not energized. In either case, applying electrical current to the contacts will change their state. 

Relays are generally used to switch smaller currents in a control circuit and do not usually control power consuming devices except for small motors and Solenoids that draw low amps. Nonetheless, relays can "control" larger voltages and amperes by having an amplifying effect because a small voltage applied to a relays coil can result in a large voltage being switched by the contacts.

Protective relays can prevent equipment damage by detecting electrical abnormalities, including overcurrent, undercurrent, overloads and reverse currents. In addition, relays are also widely used to switch starting coils, heating elements, pilot lights and audible alarms.

**Type of Relays:**

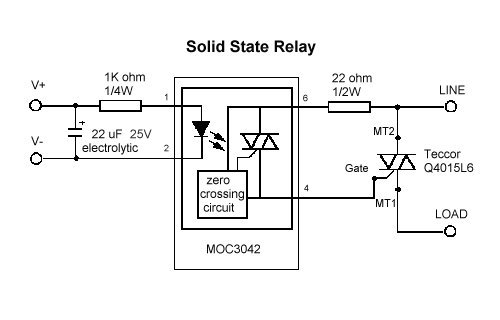
Relays are either electromechanical relays or solid-state relays.

In electromechanical relays (EMR), contacts are opened or closed by a magnetic force. With solid-state relays (SSR), there are no contacts and switching is totally electronic. The decision to use electromechanical or solid-state relays depends on an application's electrical requirements, cost constraints and life expectancy. Although solid-state relays have become very popular, electromechanical relays remain common. Many of the functions performed by heavy-duty equipment need the switching capabilities of electromechanical relays. Solid State Relays switched the current using non-moving electronic devices such as silicon controlled rectifiers.

These differences in the two types of relays result in advantages and disadvantages with each system.

Because solid state relays do not have to either energize a coil or open contacts, less voltage is required to "turn" Solid State Relays on or off. Similarly, Solid State Relays turn on and turn off faster because there are no physical parts to move.

Although the absence of contacts and moving parts means that Solid State Relays are not subject to arcing and do not wear out, contacts on Electromechanical Relays can be replaced, whereas entire Solid-State Relays must be replaced when any part becomes defective.

Because of the construction of Solid State Relays, there is residual electrical resistance and/or current leakage whether switches are open and closed. The small voltage drops that are created are not usually a problem; however, Electromechanical Relays provide a cleaner ON or OFF condition because of the relatively large distance between contacts, which acts as a form of insulation.

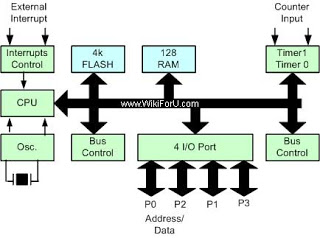
**8051 Microcontroller**

Why 8051 Over 8052?

8051 is a popular 8-bit microcontroller and has been used in many applications since Intel introduced it. Many 8051 architectures are produced by Tri-scend, Intel, Atmel, Philips, Infineon (Siemens), ISSI, and Max Corp. Today, 8051 microcontrollers may not be popular but, the 8051 architecture is still popular and employed in thousands of embedded applications.

This 8-bit architecture has been different segments such as 8052, 8051, 8751 and 8031. 8052 is the super-set of 8051 and 8031 is the memory-less microcontroller hence, it has interfaced with external ROM. Whereas, 8751 chip has only 4Kbytes of on chip UV-EPROM.

Everyone knows about the general 40-pin microcontroller i.e. 8051 introduced by Intel in 1980s and consists of serial communication pins, Timer, Interrupts, RAM, ROM. It has 4 ports and each port has 8 pins, total 32 pins and other 8 pins for other purposes.  
  
8052 is the super-set of 8051 that consists of 8K bytes of internal RAM (4K in 8051), 256 bytes of ROM (128 bytes in 8051), 3 Timers (2 timers in 8051), and 6 Interrupts (4 Interrupts in 8051) and other functionalities are same. All programs, written for 8051, can be run for 8052 but the reverse is not true.



**Home Automation Systems**

Home Automation or domestic robotics (domestics) is a combination of automation technologies and computer science.

High affordability and connectivity through smartphones and tablets has increased the popularity of Home Automation in recent years.  
Radio frequency (RF) is one of the techniques used in the Home Automation network to control and monitor home appliances.

Home Automation consists of electronic programmable controls for home appliances using wired or wireless communication. A smart home controller connects the entire house in a Home Automation System.

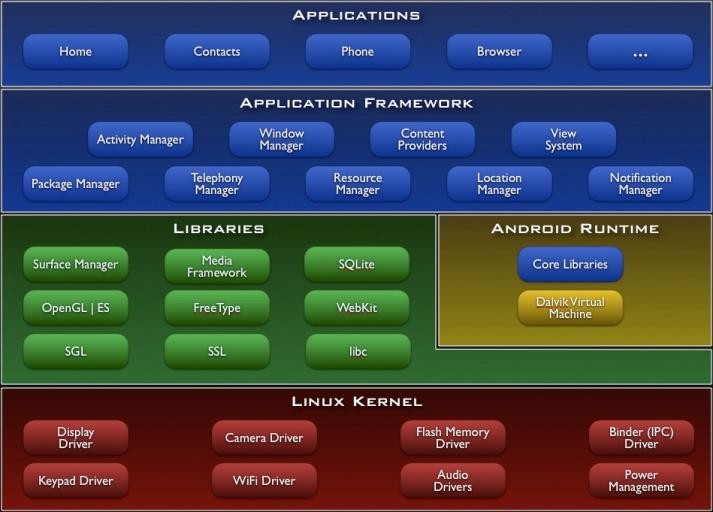
This acts like a centralized control of household systems, such as sprinkler units and inside temperature monitors. With this centralized control unit, a person does not need to leave the couch to turn on/off home appliances, which is convenient for users.

**Bluetooth Technology**

Bluetooth technology is a low-power-consumption short-range wireless communications technology and operates in the unlicensed industrial, scientific and medical (ISM) band at 2.4 GHz to 2.485 GHz.

The 2.4 GHz ISM band is available and unlicensed in most countries. Pairing uses two Bluetooth enabled devices known to each other. Table shows the types of

Bluetooth.



**Types of Bluetooth**

|  |  |
| --- | --- |
| Bluetooth Class | Range |
| Class 1 | 100 meters |
| Class 2 | 10 meters |
| Class 3 | 1 meter |

**Class 1 Bluetooth**

With class 1 Bluetooth devices transmitting at 100mW, which have a standard range of approximately 100 meters or 328 feet, range is comparable to that of an 802.11b WLAN device. Class 1 devices are most commonly implemented in devices where power is plentiful, such as laptop and desktop systems.

**Class 2 Bluetooth**

A Class 2 Bluetooth device refers to the specification of the chip used inside the device. Class 2 devices will have a power rating of 2.5mW (milliwatts) and an operating range of approximately 10 to 20 meters.

**Setting up Connection**

Any Bluetooth device in discoverable mode transmits the following information on demand:

Device name

Device class

List of services

Technical information (for example: device features, manufacturer, Bluetooth specification used, clock offset)

Any device may perform an inquiry to find other devices to connect to, and any device can be configured to respond to such inquiries. However, if the device trying to connect knows the address of the device, it always responds to direct connection requests and transmits the information shown in the list above if requested.

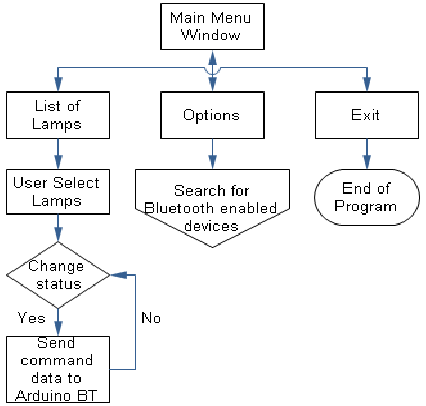
Use of a device's services may require pairing or acceptance by its owner, but the connection itself can be initiated by any device and held until it goes out of range. Some devices can be connected to only one device at a time, and connecting to them prevents them from connecting to other devices and appearing in inquiries until they disconnect from the other device.

Every device has a unique 48-bit address.

However, these addresses are generally not shown in inquiries. Instead, friendly Bluetooth names are used, which can be set by the user. This name appears when another user scans for devices and in lists of paired devices.

Most cellular phones have the Bluetooth name set to the manufacturer and model of the phone by default. Most cellular phones and laptops show only the Bluetooth names and special programs are required to get additional information about remote devices. This can be confusing as, for example, there could be several cellular phones in range named T610.





**Code**

#include<reg52.h>

#include<stdio.h>

sbit b = P0^0;

sbit c = P0^1;

unsigned int a;

void receive()

{

TMOD = 0X20;

TH1 = 0XFD;

SCON = 0X50;

TR1 = 1;

while(RI==0);

a = SBUF;

RI = 0;

}

void main()

{

P0=0X00;

while(1)

{

receive();

if(a=='a')

{

b = 0;

}

if(a=='A')

{

b = 1;

}

if(a=='b')

{

c = 0;

}

if(a=='B')

{

c = 1;

}

}

}

**MIT App Inventor for Android**

App Inventor for Android is an open-source web application originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT).

It allows newcomers to computer programming to create software applications for the Android operating system (OS). It uses a graphical interface, very like Scratch and the Star Logo TNG user interface, which allows users to drag-and-drop visual objects to create an application that can run on Android devices. In creating App Inventor, Google drew upon significant prior research in educational computing, as well as work done within Google on online development environments.

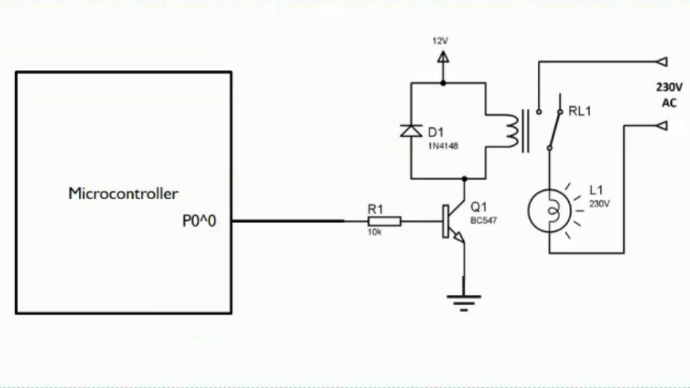
App Inventor and the projects on which it is based are informed by constructionist learning theories, which emphasizes that programming can be a vehicle for engaging powerful ideas through active learning. As such, it is part of an ongoing movement in computers and education that began with the work of Seymour Papert and the MIT Logo Group in the 1960s and has also manifested itself with Mitchel Resnick's work on Lego Mindstorms and StarLogo.

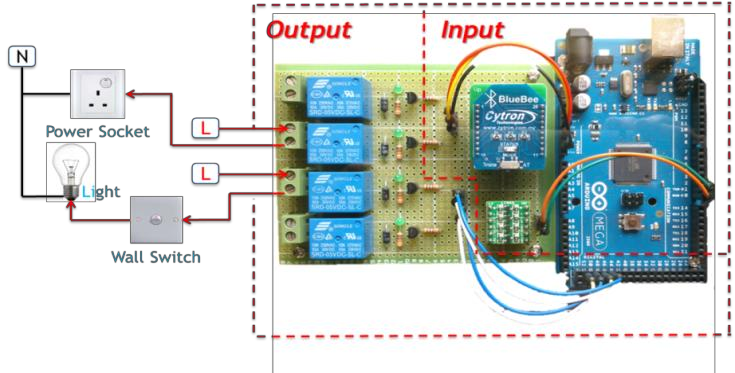
MIT App Inventor is also supported with the firebase database extension. This allows people to store data on Google’s firebase.

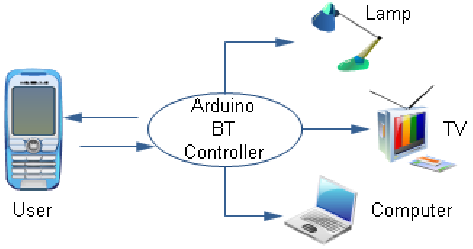
**Implementation**

This section describes the implementation of the controller.

The controller is integrated with Atmel 8051 Microcontroller, HC-05 Bluetooth Module, Two Channel Relay, and other necessary electronic components.







**Future Scope**

As Technology is increasing rapidly

and everything around us going smart.

Our device will have a huge demand in near future as Home Automation is already in the development and got huge demand in foreign countries. And it makes our works easy and operate appliances safely at their own levels.

**Conclusion**

It can be concluded that the objectives of this project were achieved, i.e., to develop a low cost and high reliability Home Automation System. In addition, a user-friendly Android application as an alternative to the remote-control unit of a Home Automation System has also been established.

**References**

1. B. MURALI KRISHNA  
    "BLUETOOTH BASED WIRELESS HOME AUTOMATION SYSTEM USING FPGA"
2. SHIRISHA TADOJU   
   "Bluetooth Remote Home Automation System using Android Application "
3. Bhavik Pandya   
   "Android Based Home Automation System Using Bluetooth & Voice

Command”

Circuit Digest –

https://circuitdigest.com/microcontroller-projects/bluetooth-controlled-home-automation-using-8051