**ABSTRACT**

This project demonstrates a simple home automation system which contains a remote mobile host controller and several home appliances. The home appliances communicate with the host controller through a wireless device such as a Bluetooth enabled mobile phone, in this case, an android based Smart phone. Bluetooth has a range of 10-15 meters, so that we can switch On and Off any electronic appliances within the range. Here the microcontroller can turn On and Off the home appliances accordingly.8051 Microcontroller is used for controlling the whole process of this project. An android application should be installed in user’s mobile or tablet to control the electrical loads. Using this android application user can send the commands to the Bluetooth module to control the electrical loads. Home appliances will be turned On and Off when user touch the button in the Bluetooth mobile app in Android mobile phone. The program which is written to the 8051 microcontroller communicates with Bluetooth module serially to receive the commands. When the user touch any button in Bluetooth controller app then Android phone sends a value to Bluetooth module, after receiving this value, Bluetooth module sends the received value to the microcontroller and then microcontroller reads it and compare it with predefined value. Microcontroller switches the electrical loads automatically based on the commands received from the Bluetooth. If any match is occurred then microcontroller performs relative operation. Same operation will performed each time when button pressed. Operating conventional wall switches is difficult for physically handicapped or elder people. This project provides the solution to this problem by integrating all the electrical appliances to a control unit that can be operated by an Android application device (Android smart phone or Tablet). It helps in creating a whole system for the house and can connect all the AC appliances to the 8051 microprocessor using Relays. Also the project is feasible because the cost of the project is very less as compared to the expensive Wi-Fi based home control systems presently available in the market which require an additional cost of internet services.

**LIST OF FIGURES**

**Fig no. Description Page no.**

1. Pin diagram of AT89C51 6
2. HC-05 Bluetooth module 9
3. IC 7805 pin diagram 10
4. 5V relay module 11
5. Pin diagram of ULN2003A 12
6. Block diagram 18
7. Circuit diagram 19
8. Results 24

**CHAPTER 1**

**INTRODUCTION**

Bluetooth Controlled Home Automation System using 8051 Microcontroller is a wireless Bluetooth technology to control the Home Electronic Appliances through an Android Phone.

In past few years there have been so many inventions in the field of consumer electronics such as cellular phone, air conditions, home security devices and home theatres. All these appliances can be easily controlled by a single controller, using personal area network in a home environment. And the market is going towards the home automation and networking and Bluetooth is an ideal solution for this purpose.

This report demonstrates a simple home automation system which consists of remote mobile, host controller, and several home appliances. The client module can communicate with host controller through a wireless device, in this case a Bluetooth enabled mobile phone.

Presently, conventional wall switches located in different parts of the house makes it difficult for the user to go near them to operate. It becomes even more difficult for the elderly or physically handicapped people to do so. Remote controlled home automation system provides a simpler solution with Android application technology.

Home automation is not a new thing but the most advanced home automation systems in existence today require a big and expensive infrastructure. This means that it often is not feasible to install a home automation system in an existing building. This project however, proposes a low cost and flexible automation system that can easily control lights, fans, and TV from an android based mobile phone. Here HC-05 Bluetooth module and 8-bit microcontroller 8051 are used for switching.

**CHAPTER 2**

**PROJECT BRIEF**

Home automation system using microcontroller 8051 and a Bluetooth module controls several electronic home appliances with the range of Bluetooth, using an android phone with a Bluetooth app.

AIM:

The Project aims at controlling the home electronic appliances through an Android phone using 8051 microcontroller and Bluetooth technology.

OBJECTIVES:

* To have a basic idea about microcontroller based home automation system.
* To get a basic idea about Bluetooth module and their working.
* To get an idea about interfacing of Bluetooth module with 8051 microcontrollers.

PROJECT RELEVENCE:

This project finds application and relevance where one desires to control the basic home systems and appliances using his smart phone and to access them from remote point. Sometimes we find it difficult to switch ON and OFF the devices when the switches are located far away from where we sit. And the condition is even worse for the elderly and those who are handicapped. With home automation, convenient control of our home is at our fingertips. And using a Bluetooth technology rather than a Wi-Fi based system reduces the overall cost of the system making it available to the common people.

**CHAPTER 3**

**DESCRIPTION**

**3.1 MICROCONTROLLER – AT89C51**

DESCRIPTION

The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4 Kbytes of Flash Programmable and Erasable Read Only Memory (PEROM). The device is manufactured using Atmel’s high density nonvolatile memory technology and is compatible with the industry standard MCS-51 instruction set and pinout.

The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcomputer which provides a highly flexible and cost effective solution to many embedded control applications.

 The AT89C51 provides the following standard features: 4Kbytes of Flash, 128 bytes of RAM, 32 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator and clock circuitry. In addition, the AT89C51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes.

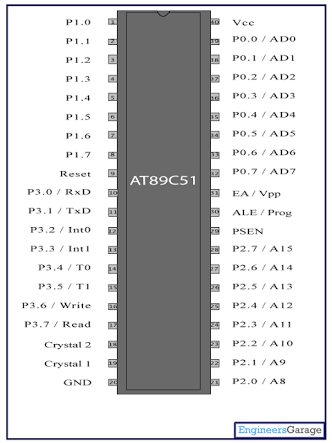


Fig 1 – Pin Diagram of AT89C51 Microcontroller.

PIN DESCRIPTION

**Pins 1-8:**Port 1

Each of these pins can be configured as an input or an output.

**Pin 9:** RESET

A logic one on this pin disables the microcontroller and clears the contents of most registers. In other words, the positive voltage on this pin resets the microcontroller. By applying logic zero to this pin, the program starts execution from the beginning.

**Pins10-17:** Port 3

Similar to port 1, each of these pins can serve as general input or output. Besides, all of them have alternative functions.

**Pin 10:** RXD

Serial asynchronous communication input or Serial synchronous communication output.

**Pin 11:** TXD

Serial asynchronous communication output or Serial synchronous communication clock output.

**Pin 12:** INT0

 Interrupt 0 input.

**Pin 13:** INT1

Interrupt 1 input.

**Pin 14:** T0

Timer 0 clock input.

**Pin 15:** T1

Timer 1 clock input.

**Pin 16:** WR

Write to external (additional) RAM.

**Pin 17:** RD

Read from external RAM.

**Pin 18, 19:**  X2, X1

Internal oscillator input and output. A quartz crystal which specifies operating frequency is usually connected to these pins. Instead of it, miniature ceramics resonators can also be used for frequency stability. Later versions of microcontrollers operate at a frequency of 0 Hz up to over 50 Hz.

**Pin 20:** GND

Ground.

**Pin 21-28:** Port 2

If there is no intention to use external memory then these port pins are configured as general inputs/outputs. In case external memory is used, the higher address byte, i.e. addresses A8-A15 will appear on this port. Even though memory with capacity of 64Kb is not used, which means that not all eight port bits are used for its addressing, the rest of them are not available as inputs/outputs.

**Pin 29:** PSEN (Program Store Enable)

If external ROM is used for storing program then a logic zero (0) appears on it every time the microcontroller reads a byte from memory.

**Pin 30:** ALE (Address Latch Enable)

Prior to reading from external memory, the microcontroller puts the lower address byte (A0-A7) on P0 and activates the ALE output. After receiving signal from the ALE pin, the external register (usually 74HCT373 or 74HCT375 add-on chip) memorizes the state of P0 and uses it as a memory chip address. Immediately after that, the ALU pin is returned its previous logic state and P0 is now used as a Data Bus. As seen, port data multiplexing is performed by means of only one additional (and cheap) integrated circuit. In other words, this port is used for both data and address transmission.

**Pin 31:** EA  (External Access)

By applying logic zero to this pin, P2 and P3 are used for data and address transmission with no regard to whether there is internal memory or not. It means that even there is a program written to the microcontroller, it will not be executed. Instead, the program written to external ROM will be executed. By applying logic one to the EA pin, the microcontroller will use both memories, first internal then external (if exists).

**Pin 32-39:** Port 0

Similar to P2, if external memory is not used, these pins can be used as general inputs/outputs. Otherwise, P0 is configured as address output (A0-A7) when the ALE pin is driven high (1) or as data output (Data Bus) when the ALE pin is driven low (0).

**Pin 40:** VCC

+5V power supply.

**3.2 HC-05 BLUETOOTH MODULE**

HC-05 Bluetooth module consists two things, one is Bluetooth serial interface module and a Bluetooth adaptor. Bluetooth serial module is used for converting serial port to Bluetooth. Default baud rate of new Bluetooth module is 9600 bps. To operate, connect Rx and Tx to controller or serial converter and give 5 volt dc regulated power supply to module. Bluetooth module has two modes one is master mode and second one is slave mode. User can set either mode by using some AT commands.

DIAGRAM



Fig 2 – HC-05 bluetooth module

PIN DESCRIPTION

1. STATE  → Open
2. Rx  → Serial receiving pin
3. Tx   → Serial transmitting pin
4. GND   → ground
5. Vcc      → +5volt dc
6. EN        → to enter in AT mod

**3.3 IC 7805**

7805 is a voltage regulator integrated circuit. It is a member of 78xx series of fixed linear voltage regulator ICs. The voltage source in a circuit may have fluctuations and would not give the fixed voltage output. The voltage regulator IC maintains the output voltage at a constant value. The xx in 78xx indicates the fixed output voltage it is designed to provide. 7805 provides +5V regulated power supply. Capacitors of suitable values can be connected at input and output pins depending upon the respective voltage levels.

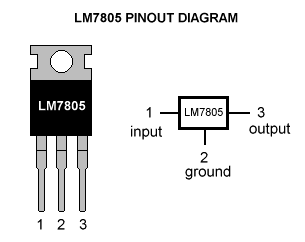


Fig 3- IC7805 pin diagram

PIN DESCRIPTION

|  |  |  |  |
| --- | --- | --- | --- |
| PIN NO | PIN | Function | DESCRIPTION |
| 1 | INPUT | Input voltage (7V-35V) | In this pin of the IC positive unregulated voltage is given in regulation. |
| 2 | GROUND | Ground (0V) | In this pin where the ground is given. This pin is neutral for equally the input and output. |
| 3 | OUTPUT | Regulated output; 5V (4.8V-5.2V) | The output of the regulated 5V volt is taken out at this pin of the IC regulator. |

**3.4 5V RELAY MODULE**

It has three pins, the VCC, GND and Signal. It can act as switch if the circuit and the load circuit have different supply voltage. It is commonly use if the load circuit is AC. It is a switch used to connect isolated connection from the circuit using a circuit signal. It has red LED that turns on every time the coil is energized or the signal pin has a high input.

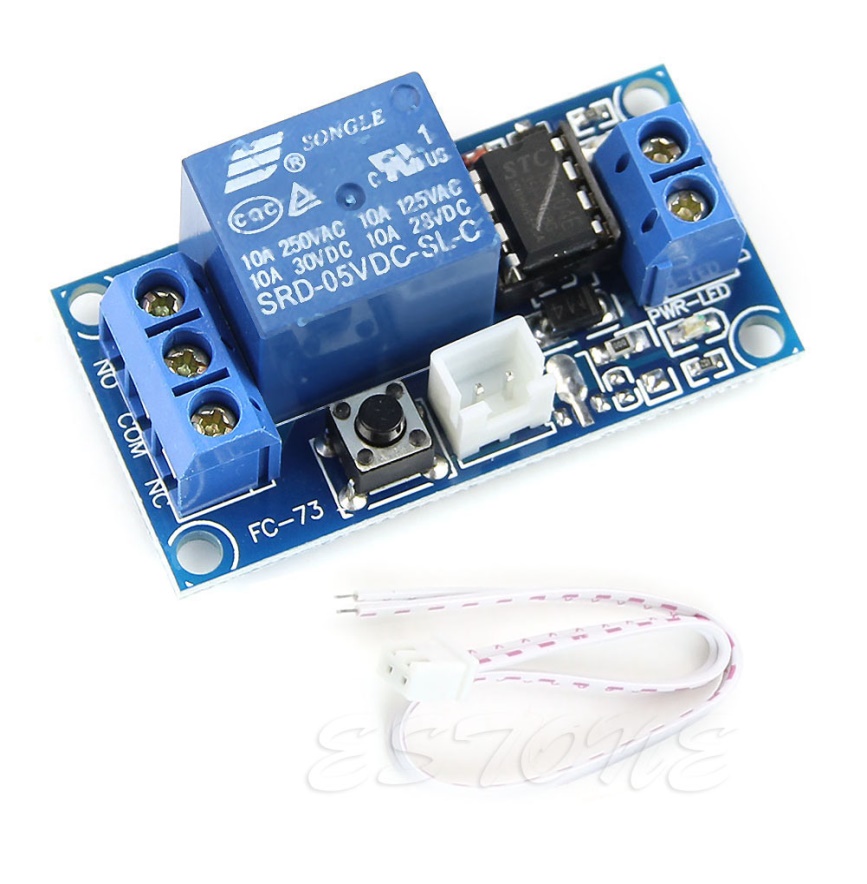


Fig 4 -5V relay module

PIN CONFIGURATION

+ : 5V power supply

- : Ground

S : Signal from the Controller

NC : Normally Closed

NO : Normally Open

COMMON : Common

**3.5 ULN2003A RELAY DRIVER**

ULN2003 is a high voltage and high current Darlington array IC. It contains seven open collector darlington pairs with common emitters. A darlington pair is an arrangement of two bipolar transistors.

ULN2003 belongs to the family of ULN200X series of ICs. Different versions of this family interface to different logic families. ULN2003 is for 5V TTL, CMOS logic devices. These ICs are used when driving a wide range of loads and are used as relay drivers, display drivers, line drivers etc. ULN2003 is also commonly used while driving [Stepper Motors](http://www.engineersgarage.com/articles/stepper-motors).

Each channel or darlington pair in ULN2003 is rated at 500mA and can withstand peak current of 600mA. The inputs and outputs are provided opposite to each other in the pin layout. Each driver also contains a suppression diode to dissipate voltage spikes while driving inductive loads.

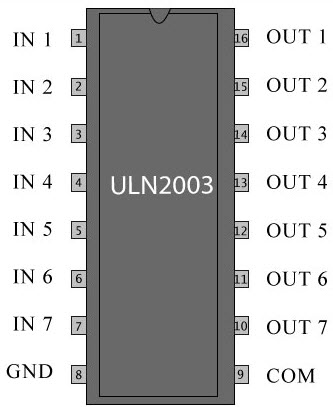


Fig 5 – pin diagram of ULN2003A

PIN DESCRIPTION

|  |  |  |
| --- | --- | --- |
| **Pin No** | **Function** | **Name** |
| 1 | Input for 1st channel | Input 1 |
| 2 | Input for 2nd channel | Input 2 |
| 3 | Input for 3rd channel | Input 3 |
| 4 | Input for 4th channel | Input 4 |
| 5 | Input for 5th channel | Input 5 |
| 6 | Input for 6th channel | Input 6 |
| 7 | Input for 7th channel | Input 7 |
| 8 | Ground (0V) | Ground |
| 9 | Common free wheeling diodes | Common |
| 10 | Output for 7th channel | Output 7 |
| 11 | Output for 6th channel | Output 6 |
| 12 | Output for 5th channel | Output 5 |
| 13 | Output for 4th channel | Output 4 |
| 14 | Output for 3rd channel | Output 3 |
| 15 | Output for 2nd channel | Output 2 |
| 16 | Output for 1st channel | Output 1 |

**CHAPTER 4**

**SERIAL DATA TRANSMISSION AND RECEPTION**

|  |
| --- |
| **Serial Interface** |
| The serial port of 8051 is full duplex, i.e., it can transmit and receive simultaneously.  The register SBUF is used to hold the data. The special function register SBUF is physically two registers. One is, write-only and is used to hold data to be transmitted out of the 8051 via TXD. The other is, read-only and holds the received data from external sources via RXD. Both mutually exclusive registers have the same address 099H. |
| **Serial Port Control Register (SCON)** |
| Register SCON controls serial data communication.  Address: 098H (Bit addressable) |
| http://nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/microcontrollers/micro/lecture11/images/fig1.1.gif |
| Mode select bits |
| http://nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/microcontrollers/micro/lecture11/images/table.gif |
| SM2: multi processor communication bit  REN: Receive enable bit  TB8: Transmitted bit 8 (Normally we have 0-7 bits transmitted/received)  RB8: Received bit 8  TI: Transmit interrupt flag  RI: Receive interrupt flag |
| **Power Mode control Register** |
| Register PCON controls processor powerdown, sleep modes and serial data bandrate. Only one bit of PCON is used with respect to serial communication. The seventh bit (b7)(SMOD) is used to generate the baud rate of serial communication. |
| Address: 87H |
| http://nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/microcontrollers/micro/lecture11/images/pcon.gif |
| SMOD: Serial baud rate modify bit  GF1: General purpose user flag bit 1  GF0: General purpose user flag bit 0  PD: Power down bit  IDL: Idle mode bit |
| **Data Transmission** |
| Transmission of serial data begins at any time when data is written to SBUF. TI is set to 1 when data has been transmitted. This signifies that SBUF is empty so that another byte can be sent. |
| **Data Reception** |
| Reception of serial data begins if the receive enable bit is set to 1 for all modes. Receive interrupt flag, RI, is set after the data has been received in all modes. The data gets stored in SBUF register from where it can be read. |
| **Serial Data Transmission Modes:** |
| Mode-0**:**  In this mode, the serial port works like a shift register and the data transmission works synchronously with a clock frequency of fosc /12. Serial data is received and transmitted through RXD. 8 bits are transmitted/ received at a time. Pin TXD outputs the shift clock pulses of frequency fosc /12, which is connected to the external circuitry for synchronization. The shift frequency or baud rate is always 1/12 of the oscillator frequency. |
| Mode-1 (standard UART mode) : |
| In mode-1, the serial port functions as a standard Universal Asynchronous Receiver Transmitter (UART) mode. 10 bits are transmitted through TXD or received through RXD. The 10 bits consist of one start bit (which is usually '0'), 8 data bits (LSB is sent first/received first), and a stop bit (which is usually '1'). Once received, the stop bit goes into RB8 in the special function register SCON. The baud rate is variable.  Bit time= 1/fbaud. In receiving mode, data bits are shifted into the receiver at the programmed baud rate. The data word (8-bits) will be loaded to SBUF if RI is zero and mode bit SM2 is zero or stop bit is 1. After the data is received and the data byte has been loaded into SBUF, RI becomes one. |
|  |

**CHAPTER 5**

**BLOCK DIAGRAM AND WORKING EXPLANATION**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| In this project we have used8051 microcontroller for controlling the whole process of this project. And a Bluetooth module is used for controlling the home appliances wirelessly. Home appliances will turned ON and OFF when user will touch button in the Bluetooth mobile app in Android mobile phone. To run this project, first we need to download Bluetooth app form Google play store. We can use any Bluetooth app that can send data using Bluetooth. Here are some apps name that can be used:   1. Bluetooth Spp pro 2. Bluetooth controller   After installing the App, you need to open it and then search Bluetooth device and select HC-05 Bluetooth device. And then configure keys.  Here in this project we have used Bluetooth controller app.  1.Download and install Bluetooth Controller.  2.Turn ON mobile Bluetooth.  3.Now open Bluetooth controller app  4.Press scan  5.Select desired Bluetooth device (Bluetooth Module HC-05).  6.Now set keys by pressing set buttons on screen.  To set keys we need to press ‘set button’ and set key as given below:   |  |  |  | | --- | --- | --- | | **Button** | **Data** | **Operation** | | Fan On | 1 | Fan Turned On | | Fan Off | 2 | Fan Turned Off | | Light On | 3 | Light Turned On | | Light Off | 4 | Light Turned Off | | TV On | 5 | TV Turned On | | TV Off | 6 | TV Turned Off | |
| After setting keys press ok.  You can see that there are 9 buttons in which first row is for fan controlling, second one is for light controlling and last one is for TV controlling.  Means First row’s ON and OFF buttons are used to ON and OFF the fan, second row’s buttons are for Light and third ones are for TV. We have used three bulbs of different colors instead of TV and fan, for demonstration purpose.  Now, when we touch any button in Bluetooth controller app then Android phone sends a value to Bluetooth module, after receiving this value, Bluetooth module sends the received value to the microcontroller and then microcontroller reads it and compare it with predefined value. If any match is occurred then microcontroller performs relative operation. Same operation will performed each time when button pressed.  Bluetooth Controlled Home Automation Block diagram  Fig 6 – Block diagram  Now, when user touch ‘Fan On’ button in Bluetooth controller app then microcontroller receives ‘1’ via Bluetooth module and then controller Switch ‘On’ the Fan by using relay driver and relay. And when user touch ‘Fan Off’ button in Bluetooth controller app then microcontroller receives ‘2’ via Bluetooth module and then controller Switch ‘Off’ the Fan by using relay driver and relay.  Likewise 3,4,5,6 numbers are sent by Android Phone, when Light On, Light Off, TV On, TV Off button has been touched respectively |

**CHAPTER 6**

**CIRCUIT DIAGRAM AND EXPLANATION**

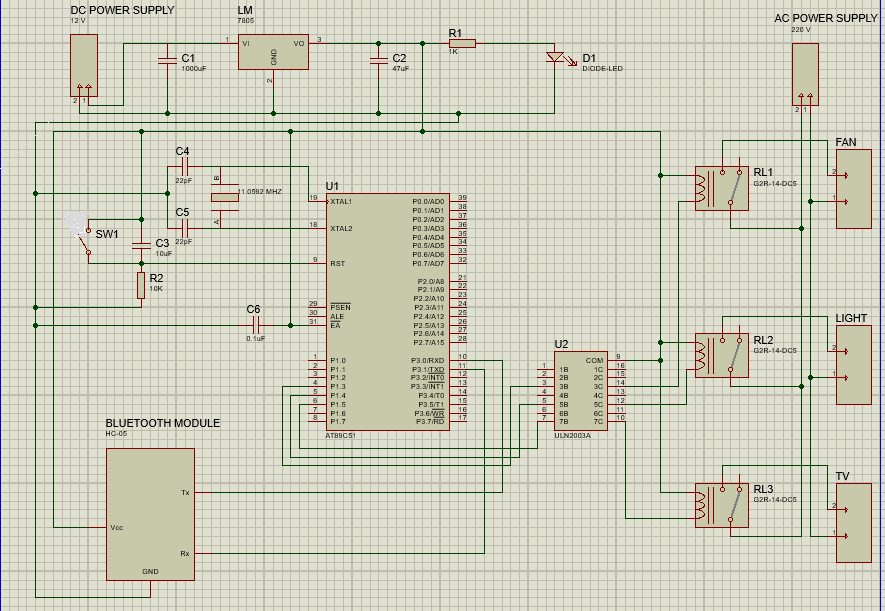
Circuit connections of this project are very simple. Bluetooth module’s Rx and Tx pins are directly connected to the Tx and Rx pins of Microcontroller. Three 5 volt relays are used as a switch for turning ON and OFF the home appliances running on AC mains. And a relay driver ULN2003A is used for driving relays. Fan, Light and TV are connected at P1.3, P1.4 and P1.5 through relays and relay driver. An 11.0592 MHz crystal oscillator is used in this circuit for generating clock signal for microcontroller. And a 5 volt voltage regulator LM7805 is used for provide 5 volt for the whole circuit.

Fig 6- Circuit diagram

**CHAPTER 7**

**PROGRAM**

//including header files and defines input, output pins and variables

#include<reg51.h>

sbit Fan=P1^3;  
sbit Light=P1^4;  
sbit TV=P1^5;

 char str;  
 char Charin=0;

//creating a function for delay

void delay(int time)  
{  
 unsigned int i,j;  
 for(i=0;i<time;i++)  
 for(j=0;j<1275;j++);  
}

//configuring 9600bps baud rate at 11.0592MHz Crystal Frequency

void Serialwrite(char byte)  
{  
  SBUF=byte;  
  while(!TI);  
  TI=0;  
}

void Serialprintln(char \*p)  
{  
  while(\*p)  
  {  
    Serialwrite(\*p);  
    p++;  
  }  
  Serialwrite(0x0d);  
}

void Serialbegin()  
{  
   TMOD=0x20;  
   SCON=0x50;  
   TH1=0xfd;  
   TR1=1;  
}

// initialized UART and monitored the SBUF register for receiving the data

// then data is matched and compared with predefined values and relative operation is performed

void main()  
{  
  P1=0x00;  
  Serialbegin();  
  Serialprintln("System Ready...");  
  delay(50);  
  while(1)  
  {  
    while(!RI);  
    Charin=SBUF;  
    str=Charin;  
    RI=0;  
      if(str=='1')  
      {  
        Fan=1;  
        Serialprintln(" Fan ON");  
        delay(50);  
      }  
      else if(str=='2')  
      {  
        Fan=0;  
        Serialprintln(" Fan OFF");  
        delay(50);  
      }

       else if(str=='3')  
      {  
        Light=1;  
        Serialprintln(" Light ON");  
        delay(50);  
      }

       else if(str=='4')  
      {  
        Light=0;  
        Serialprintln(" Light OFF");  
        delay(50);  
      }

       else if(str=='5')  
      {  
        TV=1;  
        Serialprintln(" TV ON");  
        delay(50);  
      }

       else if(str=='6')  
      {  
        TV=0;  
        Serialprintln(" TV OFF");  
        delay(50);  
      }  
      str=0;  
  }  
}

So that’s how we can create a whole system for the house and can connect all the AC appliances to the 8051 microcontroller using Relays. And this Bluetooth controlled home automation system can be operated from a Smart phone.

**CHAPTER 8**

**COST ESTIMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **SL.NO** | **COMPONENTS** | **QUANTITY** | **COST(in Rs.)** |
| **1.** | Atmel 89C51 | 1 | 70/- |
| **2.** | Bluetooth module HC-05 | 1 | 350/- |
| **3.** | Relay | 1 | 300/- |
| **4.** | ULN2003 | 1 | 20/- |
| **5.** | Bulb | 3 | 15/- |
| **6.** | Holder | 3 | 80/- |
| **7.** | Wire | 5 metres | 50/- |
| **8.** | IC 7805 | 1 | 85/- |
| **9.** | Burning kit | 1 | 1500/- |
| **10.** | 10µF capacitor | 1 | 5/- |
| **11.** | 1000µF capacitor | 1 | 5/- |
| **12.** | 0.1µF capacitor | 1 | 5/- |
| **13.** | 47µF capacitor | 1 | 5/- |
| **14.** | 22Pf capacitor | 2 | 5/- |
| **15.** | 1K resistor | 1 | 5/- |
| **16.** | 10K resistor | 1 | 5/- |
| **17.** | 230/12V Transformer | 1 | 100/- |
| **18.** | Diode 1N4007 | 4 | 4/- |
| **19.** | 11.0592 MHz crystal oscillator | 1 | 20/- |
| **20.** | LED | 1 | 2/- |
| **TOTAL COST** | | | **Rs.3,008/-** |

**CHAPTER 9**

**RESULTS**

The project “home automation using AT89C51” has been successfully designed and tested. The project designed is very practical in nature because everything can be controlled with the help of just a mobile phone which is widely available nowadays and also proves to be handy.

With home automation, convenient control of our home is at our fingertips. And using a Bluetooth technology rather than a Wi-Fi based system reduces the overall cost of the system making it available to the common people.

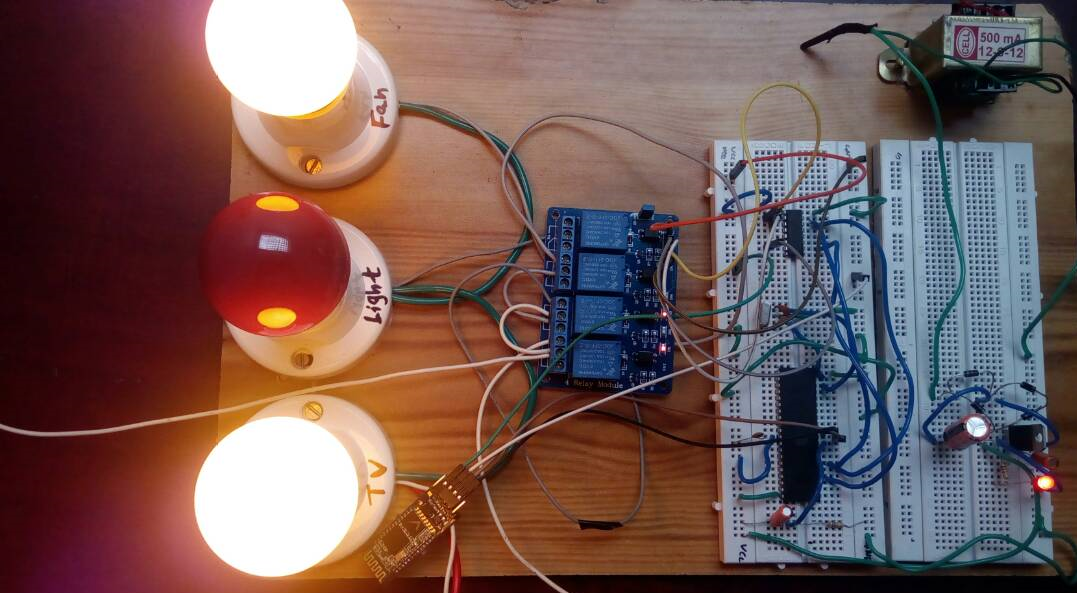
 

Fig 7 – Results

**CONCLUSION AND FUTURE SCOPE**

**CONCLUSION**

The home automation system has been experimentally proven to work satisfactorily by connecting sample appliances to it and the appliances were satisfactorily controlled from a wireless mobile device. The Bluetooth client was successfully tested on a multitude of different mobile phones from different manufacturers, thus proving its portability and wide compatibility.

Home automation conveys incredible convenience and comfort to its users. The system being discussed in the project is a robust and easy to use system. According to this project all the controls would be in the hands of the user, who is provided with the facility of monitoring all the appliances within the communication range through Bluetooth and can check the status of the appliances at whatever time of the day using his phone. Apart from increasing the comfort, the system also increases the home’s energy efficiency and savings by remotely powering off systems and appliances when they aren’t in use. Hence, home automation literally pays off.

This project will not only provide convenience to the common man but will be a boon for the elderly and disabled.

**FUTURE SCOPES**

* Using a Bluetooth controller app with more keys and increasing the number of relays we can control more number of devices in home.
* Using any other Bluetooth module that supports multi-user interface instead of HC-05 more than one users can be allowed to control the devices.
* The project can be further developed by integrating it with the internet to monitor the home while sitting in a remote area. By doing this, one can keep an eye on his or her home through internet connected to the user’s mobile phone or PC or laptop. This will not only improve the security of the one’s home in this modern day world but will also assist in conservation of energy like if one left any home appliances switched on by mistake, then he can check the status of the appliance on the graphical interface made on his mobile and can switch it of using the internet connectivity.

**BIBILIOGRAPHY**

1. Muhammed Ali Mazidi-AVR Microcontroller and Embedded Systems

2. Google-Wikipedia

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