**REPORT**

ON

**ANDROID CONTROLLED CAR WITH SPEED**

**BY:**

**SANCHIT KULKARNI**

ABSTRACT

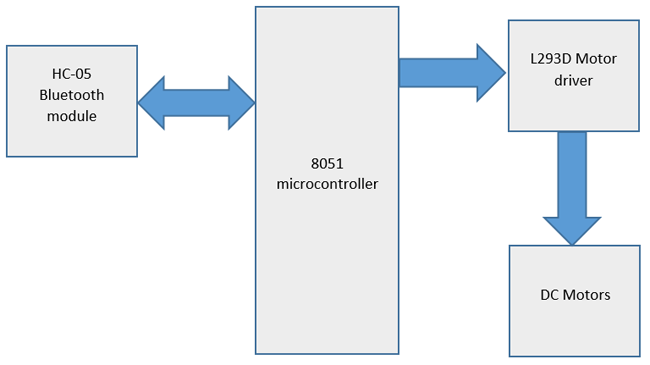
The aim of this project is to make an Android-based smartphone or tablet to control the automobiles. Android is an operating system in Linux language. It was developed from the ground to help the developers to create a mobile application for effective utilization of mobile with touch gestures, voice command etc.

An android controlled automobile allows the user to control a battery power automobile wirelessly through an Android device. This system has a Bluetooth module as a medium of data transfer between the automobile and the Android device. The data received by the Bluetooth are processed by 8051 microcontroller which performs the desired action.

The Android device has an application which provides the user with a GUI (Graphic User Interface) to send the commands using their touch gesture. The commands are sent to the automobile through an active Bluetooth device in the form of string variables.

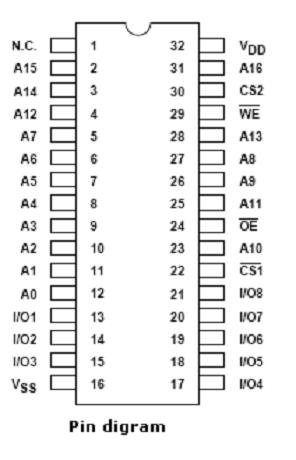
In this project we are going to build an **Android Phone controlled robot using 8051 microcontrollers** and Bluetooth module. The robot is designed using DC motors and the direction of DC motors will be controlled by the commands received from the android application. The status of the robot is sent back to the Android app. This project will also help for interfacing of HC-05 Bluetooth module with 8051 microcontrollers.

BLOCK DIAGRAM

****

8051 MICROCONTROLLER:

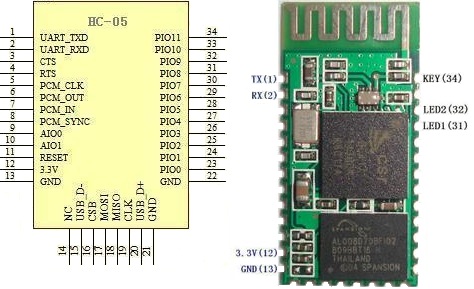
8051 microcontroller is an 8-bit microcontroller which has 128 bytes of on chip RAM, 4K bytes of on chip ROM, two timers, one serial port and four 8bit ports. 8052 microcontroller is an extension of 8051 microcontroller. In this project we are using **AT89S52**microcontroller. The below figure shows pin diagram of 8051 microcontroller.



HC-05 Bluetooth Module:

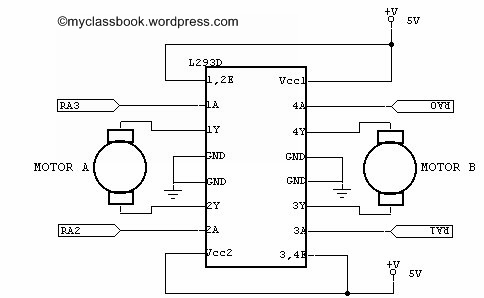
HC-05 is a serial Bluetooth module. It can be configured using AT commands. It can work in three different configurations (Master, Slave, Loop back). In our project we will be using it as a slave. The features of HC-05 module includes,

* Typical -80dBm sensitivity.
* Default baud rate: 9600bps, 8 data bits, 1 stop bit, no parity.
* Auto-pairing pin code: “1234” default pin code
* It has 6 pins.
* Vcc and Gnd pins are used for powering the HC-05.
* Tx and Rx pins are used for communicating with the microcontroller.
* Enable pin for activating the HC-05 module. when it is low , the module is disabled
* State pin acts status indicator. When it is not paired/connected with any other Bluetooth device, LED flashes continuously. When it is connected/paired with any other Bluetooth device, then the LED flashes with the constant delay of 2 seconds.

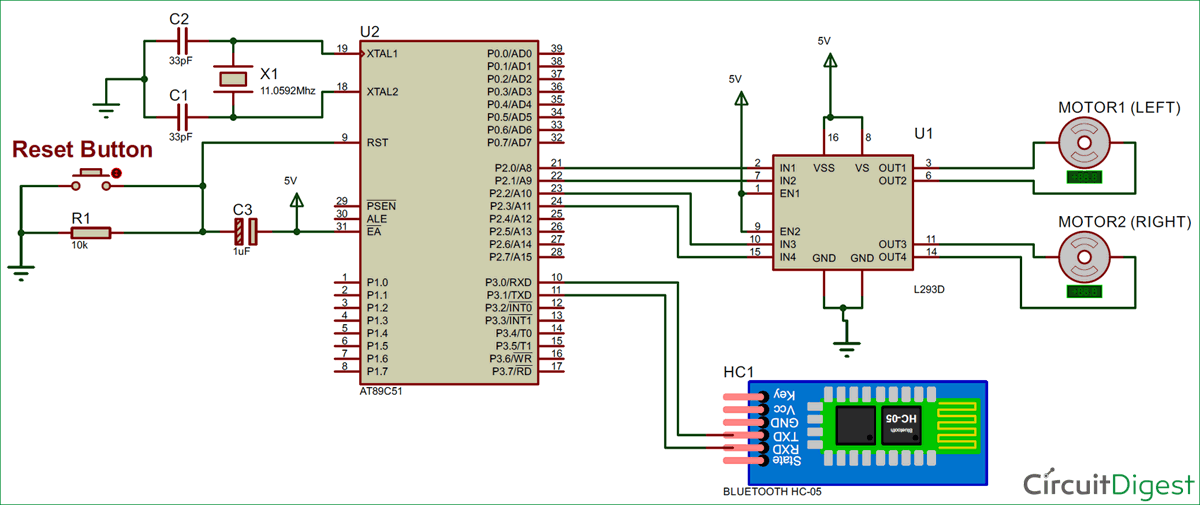


### **L293D Motor Driver IC:**

L293D is a dual H-bridge motor driver IC. This acts as a current amplifier, the output of L293D drives the DC Motors. It contains two inbuilt H-bridge circuits. In common mode of operation , it can drive two dc motors simultaneously in both the directions. The below table shows the pin description of L293D IC. Here are some [projects using L293D Motor Driver](https://circuitdigest.com/tags/motor-driver).



SCHEMATIC DIAGRAM



In this **Smart Phone controlled Robot**, the user of android app sends the data to 8051 microcontroller through HC-05 module. The received data is compared in 8051 microcontroller and the decision is made accordingly.

COMPONENTS REQUIRED

HARDWARE REQUIRED:

* 8051 microcontroller (AT89S52)
* HC-05 Bluetooth module
* L293D Motor Driver
* Robot chassis
* DC Motors (2)
* Wheels (4)
* Jumper wires

 SOFTWARE REQUIRED:

* Keil UVision 5
* Bluetooth Terminal Android app

### **Working of Android Phone Controlled Robot:**

In this **Smart Phone controlled Robot**, the user of android app sends the data to 8051 microcontroller through HC-05 module. The received data is compared in 8051 microcontroller and the decision is made accordingly. The below table shows the direction of motors and status of robot for different received characters.

|  |  |  |  |
| --- | --- | --- | --- |
| **Received character** | **Motor 1** | **Motor 2** | **Status of robot** |
| f | Forward | Forward | Moves forward |
| b | Backward | Backward | Moves backward |
| r | Forward | Backward | Moves Right |
| l | Backward | Forward | Moves left |
| s | Off | Off | Stopped |
| a | Forward | Forward | accelerates |
| R | Forward | off | Forward right |
| L | off | forward | Forward left |
| m | forward | forward | Medium speed |
| h | forward | forward | High speed |

The **Bluetooth terminal**app allows us to emulate a Bluetooth terminal. This app supports bidirectional communication and this app is compatible with most of the devices.

The steps below show how to install and use this app.

1. Download and install Bluetooth terminal app on your android phone. The app can be downloaded from the Google play store.

2. After installing the app, open the app and turn on Bluetooth.

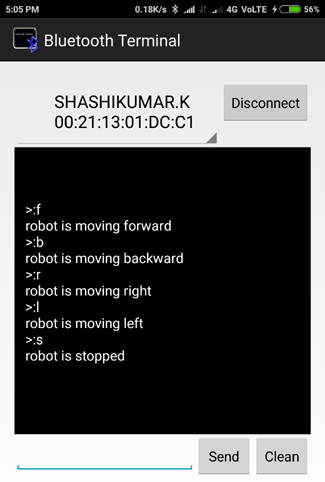
3. Select the device and click on connect option. After successful connection, we  can start sending data to HC-05 module.

In our project we have used the concept of pulse width modulation so that our robot can accelerate

And can move in different speeds depending on the character sent by Bluetooth terminal.

Ex: ‘l’ for low speed

‘a’ for acceleration



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| m1f | m1b | m2f | m2b | Motor 1 rotation | Motor 2 rotation | Status of robot |
| 1 | 0 | 1 | 0 | forward | forward | Moving forward |
| 0 | 1 | 0 | 1 | reverse | reverse | Moving backward |
| 1 | 0 | 0 | 1 | forward | reverse | Immediate right |
| 0 | 1 | 1 | 0 | reverse | forward | Immediate left |
| 0 | 0 | 0 | 0 | stopped | stopped | Stopped |
| 1 | 0 | 0 | 0 | forward | stopped | Normal right |
| 0 | 0 | 1 | 0 | Stopped | forward | Normal left |

CODE EXPLANATION

#include<reg51.h>

unsigned char ch1;

unsigned char s;

sbit m1f=P2^0;

sbit m1b=P2^1;

sbit m2f=P2^2;

sbit m2b=P2^3;

sbit en=P2^4;

void delay(unsigned int);

char rxdata(void);

void txdata(unsigned char);

void speedl();

void speedm();

void speedh();

void delayt();

void main(void)

{

unsigned char i;

unsigned char msg1[]={"low speed=1.2"};

unsigned char msg2[]={"medium speed"};

unsigned char msg3[]={"high speed"};

unsigned char msg4[]={"very high speed"};

unsigned char msg5[]={" stopped"};

unsigned char msg6[]={" reverse"};

unsigned char msg7[]={" acc=0.8"};

unsigned char msg8[]={" right"};

unsigned char msg9[]={" left"};

unsigned char msg10[]={" forward right"};

TMOD=0x21;

SCON=0x50; //8bit data , 1 stop bit , REN enabled

TH1=0xfd; //timer value for 9600 bits per second(bps)

TR1=1;

while(1)

{

s=rxdata();

if(s=='f')

{

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

for(i=0;msg1[i]!='\0';i++)

{

txdata(msg1[i]);

}

while(s=='f')

{

speedl();

s=rxdata();

}

}

else if(s=='m')

{

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

for(i=0;msg2[i]!='\0';i++)

{

txdata(msg2[i]);

}

while(s=='m')

{

speedm();

s=rxdata();

}

}

else if(s=='h')

{

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

for(i=0;msg3[i]!='\0';i++)

{

txdata(msg3[i]);

}

while(s=='h')

{

speedh();

s=rxdata();

}

}

else if(s=='v')

{

for(i=0;msg4[i]!='\0';i++)

{

txdata(msg4[i]);

}

while('v'==rxdata())

{

en=1;

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

}

}

else if(s=='s') //stop

{

for(i=0;msg5[i]!='\0';i++)

{

txdata(msg5[i]);

}

while(s=='s')

{

m1f=0;

delay(1);

m1b=0;

delay(1);

m2f=0;

delay(1);

m2b=0;

delay(1);

s=rxdata();

}

}

else if(s=='b') //back

{

for(i=0;msg6[i]!='\0';i++)

{

txdata(msg6[i]);

}

while('b'==rxdata())

{

en=1;

m1f=0;

delay(1);

m1b=1;

delay(1);

m2f=0;

delay(1);

m2b=1;

delay(1);

}

}

else if(s=='r') //right

{

for(i=0;msg8[i]!='\0';i++)

{

txdata(msg8[i]);

}

while('r'==rxdata())

{

en=1;

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=0;

delay(1);

m2b=1;

delay(1);

}

}

else if(s=='l') //left

{

for(i=0;msg9[i]!='\0';i++)

{

txdata(msg9[i]);

}

while('l'==rxdata())

{

en=1;

m1f=0;

delay(1);

m1b=1;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

}

}

else if(s=='R') // F right

{

for(i=0;msg10[i]!='\0';i++)

{

txdata(msg10[i]);

}

while('R'==rxdata())

{

en=1;

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=0;

delay(1);

m2b=0;

delay(1);

}

}

else if(s=='L') // F left

{

for(i=0;msg9[i]!='\0';i++)

{

txdata(msg9[i]);

}

while('L'==rxdata())

{

en=1;

m1f=0;

delay(1);

m1b=0;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

}

}

else if (s=='a')//acceleration if s=a then acceleration is produced

{

for(i=0;msg7[i]!='\0';i++)

{

txdata(msg7[i]);

}

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

for(i=0;i<8;i++)

{

speedl();

}

for(i=0;i<3;i++)

{

speedm();

}

for(i=0;i<3;i++)

{

speedh();

}

while(s=='a')

{

en=1;

m1f=1;

delay(1);

m1b=0;

delay(1);

m2f=1;

delay(1);

m2b=0;

delay(1);

s=rxdata();

}}}}

char rxdata()//function recieves data and sends to main

{

if(RI=0){

ch1=SBUF;

return ch1;}

else if (RI=1)

{

RI=0;

ch1=SBUF;

return ch1;

}

}

void txdata(unsigned char x)//function to transmit data

{

SBUF=x;

while(TI==0);

TI=0;

}

void delay(unsigned int z)//creates dealay

{

unsigned int p ,q;

for(p=0 ; p<z ; p++)

{

for(q=0 ; q<1375 ; q++);

}

}

void speedl()//low speed function

{

int j;

en=1;

for(j=0;j<3;j++)

{

delayt();

}

en=0

for(j=0;j<3;j++)

{

delayt();

}

}

void speedm()//medium speed function

{

int j;

en=1;

for(j=0;j<2;j++)

{

delayt();

}

en=0;

for(j=0;j<2;j++)

{

delayt();

}}

void speedh()//high speed function

{

en=1;

delayt();

en=0;

delayt();}

void delayt()//delay function

{

TMOD=0x21;

TL0=0x00;

TH0=0x00;

TF0=0;

TR0=1;

while(TF0==0);

TR0=0;

TF0=0;

}

CONCLUSION

Android controlled robots are one of the most useful projects that can be developed by a 8051 microcontroller .One can use Arduino also for this project since Arduino’s are more advanced version of microcontrollers.

Some advantages of this robot include:

1. The robot is small in size so can be used for spying.

2. With few additions and modifications, this robot can be used in the borders for detecting and disposing hidden land mines.

3. The robot can be used for surveillance.

4. We can interface sensors to this robot so that it can monitor some parameters.

5. We can add wireless camera to this robot.

FUTURE SCOPE

Since robotics is considered as one of the most useful concepts for the future of mankind and so this concept of android controlled robot has many scopes. Since movement is a very very important part of controlling of the robots and since motors are used for this purpose controlling of motors become very important part. If we are able to control the motors in the way we want it to do that becomes really helpful for us to control robots. Programmable microcontrollers help us lot in doing this and instead of Bluetooth terminal apps we can give voice command type inputs which is processed as a command like move forward or backward etc. In our project we modified by using concept of pulse width modulation and so we can accelerate our robot and lower or increase speed according to our needs which is addition to what we could find in different sources so with more time and budget this project can be enhanced and manipulated for many uses like that we have discussed in the section of conclusion.

REFERENCES:

1.www.circuitdigest.com

2.www.electronicsforu.com

3.’The 8051 microcontroller and embedded c’ by Muhammad ali mazidi