PROJECT REPORT

CSD 328 / EE304 : Communication Networks
Course Project Proposal

Bhuvnesh Bhatia
AAA0142, BHUVNESH-ECE15@snu.edu.in

Ankita Srivastav
AAA0078, ANKITA-CSE15@snu.edu.in

May 08, 2014

Abstract

We propose to implement a communication network between two robots. Large number of simple robots can perform complex tasks in a more efficient way than a single robot, giving robustness and flexibility to the group. The robots can work in different group sizes, from few individuals to thousands of them. The communication could be in any form such as the robots trying to follow a particular path, or simply mimicking the other, via communicating with each other by transmitting data over radio frequencies(RF) and direction of motion being controlled using Bluetooth via an external device.

Introduction

Our motive is to implement the fundamental concept of wireless communication on a small scale. We intend to show communication between two robots in a simplex mode using Radio Frequencies. We chose to go with Radio Frequency (abbreviated as RF) module considering the price factor when the task could be achieved using an Xbee module as well. The communication involves simple directional commands transmitted wirelessly through a robot to the other robot. The slave robot then decodes the transmitted data to move in the direction specified.

Figure 1 shows a schematic diagram of the communication.

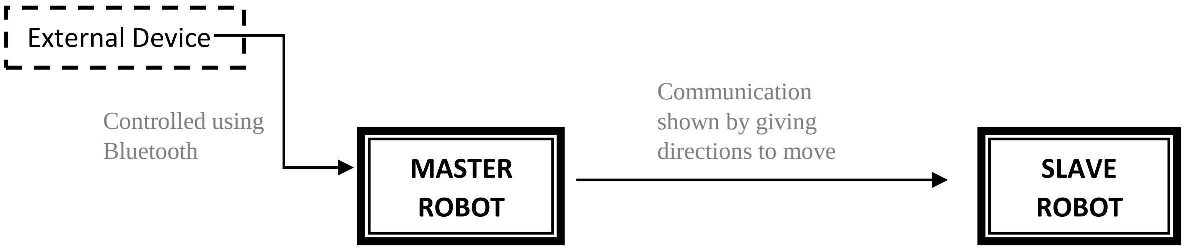

External Device
Controlled using Bluetooth
MASTER ROBOT
Communication shown by giving directions to move
SLAVE ROBOT

Figure 1

A. Designing individual robots

- <u>Master Robot</u>

  The master robot is constructed using Arduino UNO board. Small DC motors for the motion of the robot are interfaced with the microcontroller using L293D motor driver IC. Figure 2 shows the pin diagram of Arduino UNO and interfacing of the same with motor controller IC.

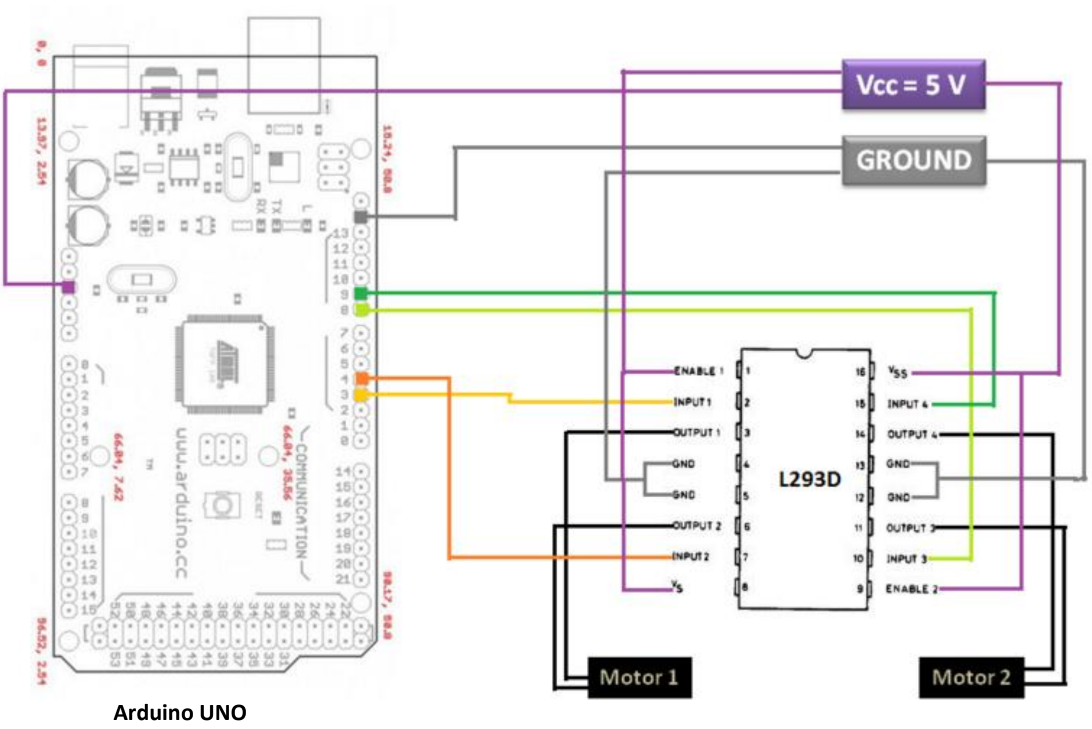


Figure 2

- <u>Slave Robot</u>

  The slave robot is attached with two DC motors and a motor driver IC to control its motion.

B. Implementing communication between the robots

The master robot is equipped with a RF transmitter module (433 MHz). The module consists of a transmitter and an encoder IC (HT12E).
Figure 3 shows the pin diagram of a RF transmitter interfaced with HT12E, L293D and Arduino UNO.
Refer appendix B for specifications of Bluetooth and RF module.

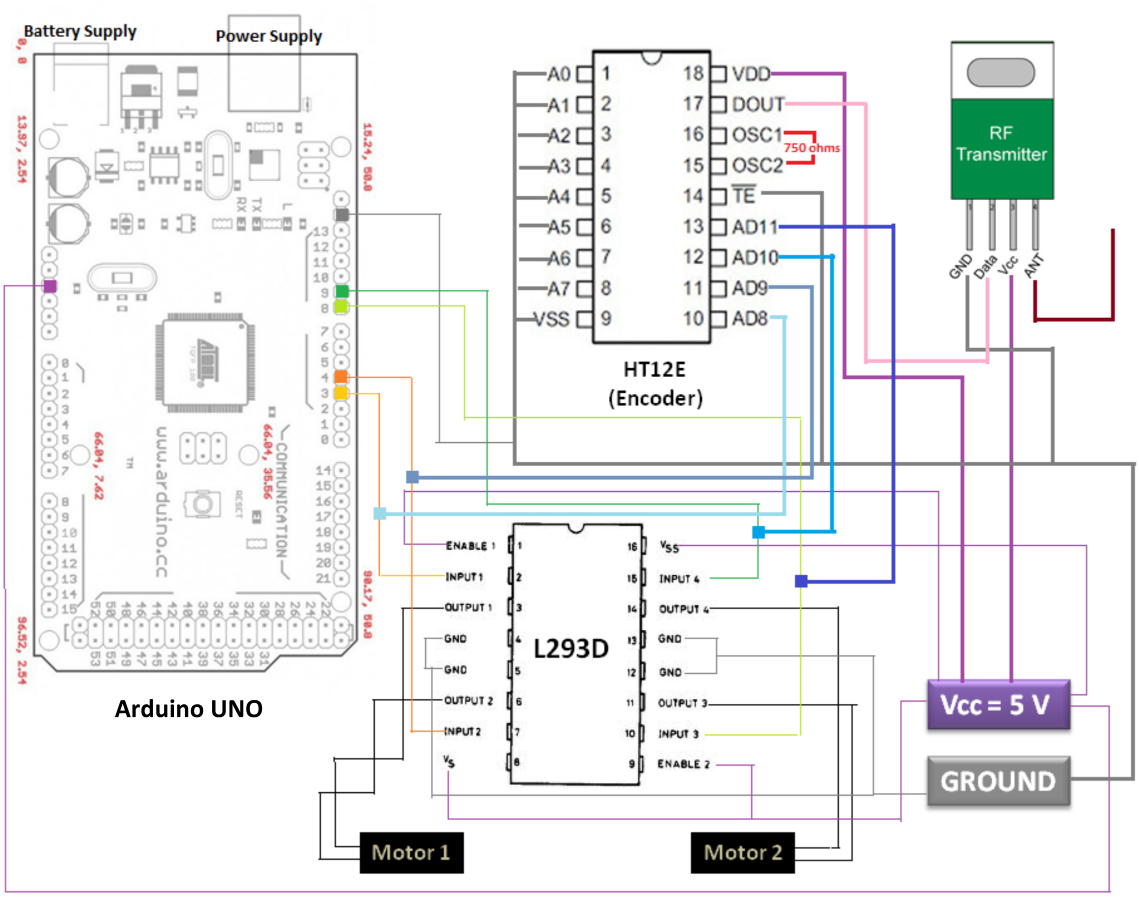


Figure 3

The slave robot is equipped with a RF receiver module consisting of a receiver and a decoder IC (HT12D), an L293D IC and two motors.
Figure 4 shows the interfacing of the individual modules.

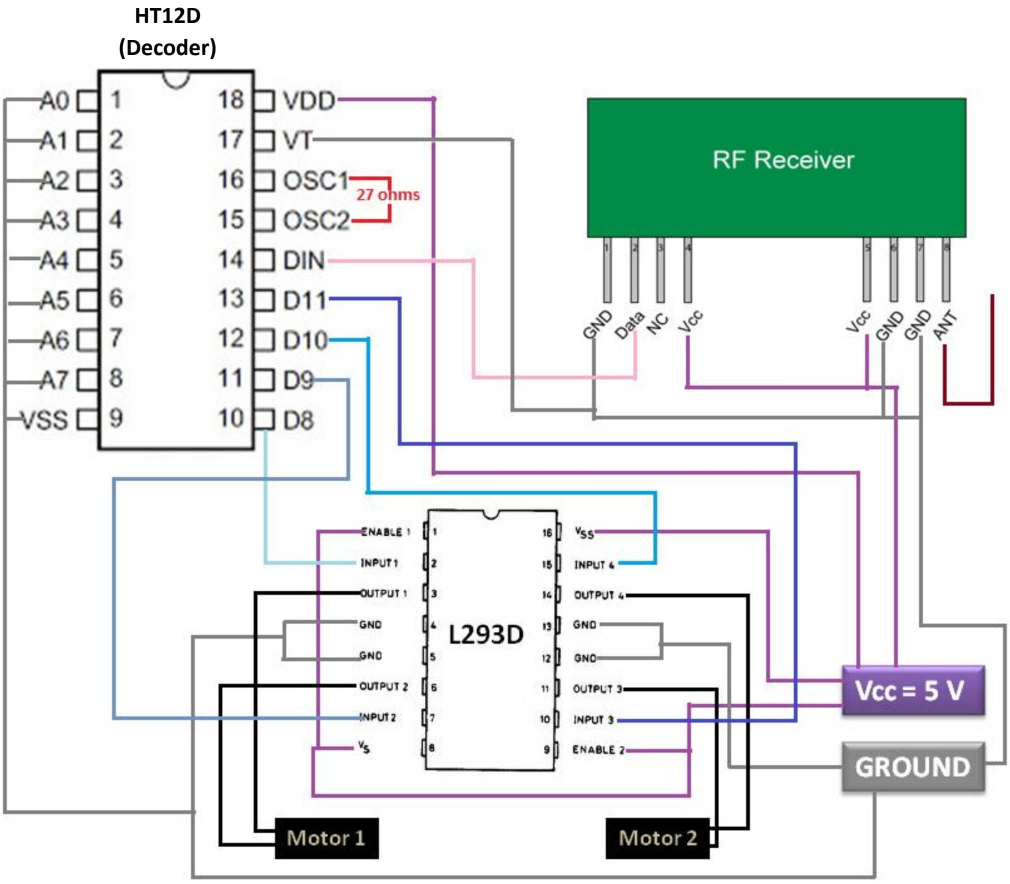


Figure 4

The Radio Frequency module (transmitter and receiver) uses ASK modulation and works on 433 MHz frequency band. It supports up to 10 kbps of signal rate and can transmit up to 90 m in an open area.[1]

C. Working of the system

The command to the Master Robot is given by cell phone using Bluetooth communication. This is achieved using the Bluetooth module (HC-05) interfaced with the Arduino board. The Master Robot follows the given command and transmits the same command wirelessly to the Slave Robot.

Figure 5 shows the interfacing of HC-05 with the Arduino UNO.

[1] Source: http://www.instructables.com/id/RF-315433-MHz-Transmitter-receiver-Module-and-Ardu/

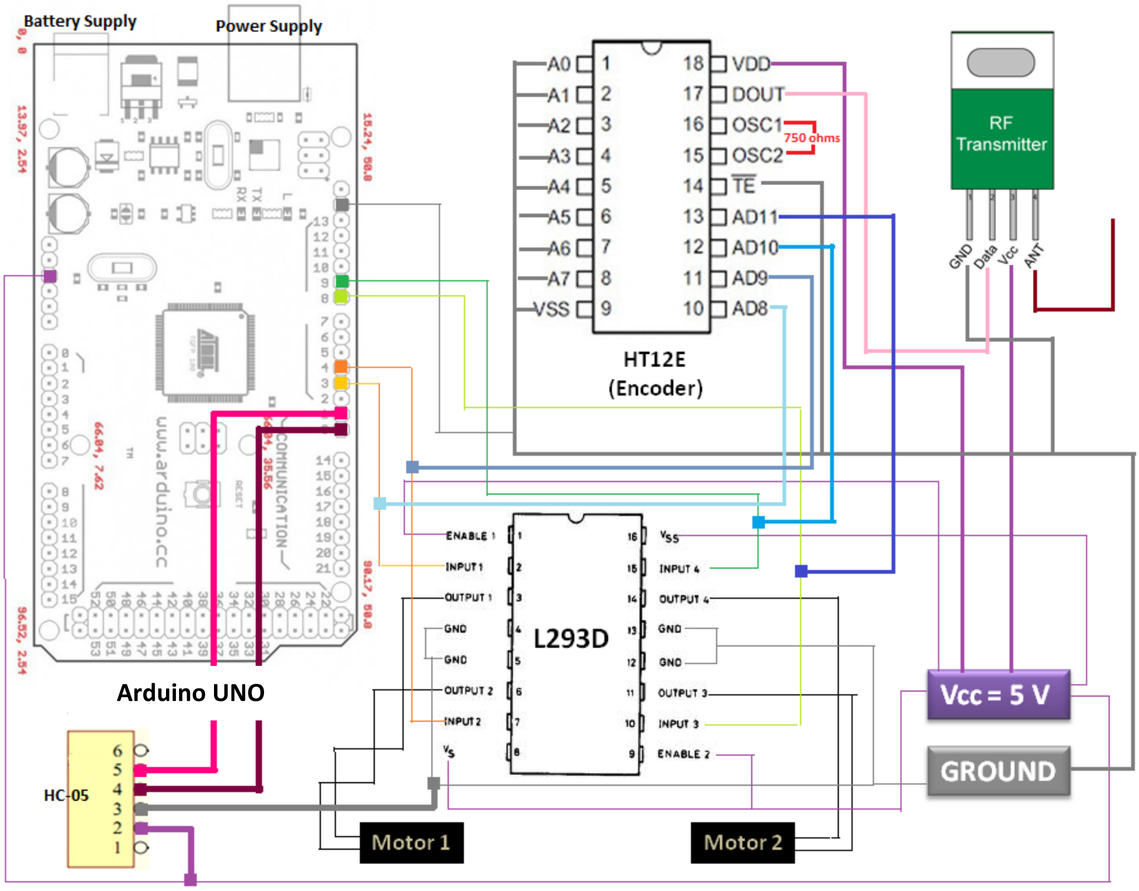


Figure 5

D. Performance evaluation

The Bluetooth module worked up to a range of 4 metres in an open area whereas the range of RF module on testing came out to be about 50 meters. However as the distance of slave robot from the master robot increased the time delay in reception of data in slave robot increased.

Since the motors used in the two robots were of different RPM, a difference is speed of movement of the robots was seen. This can be easily corrected using motors of the same RPM.

Conclusion

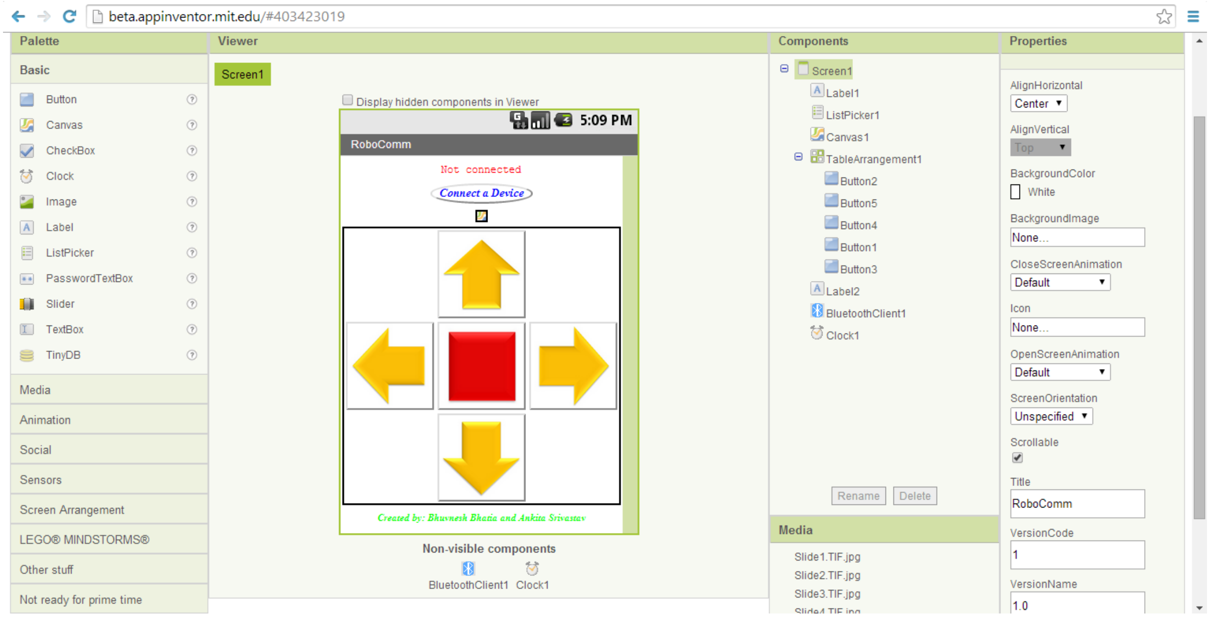
This project has given us a better understanding of how the actual communication in the real world takes place. It further gave us an insight into the interfacing hardware components with microcontroller and also helped us familiarise with the application of communication concepts.

Appendices

• Appendix A – Programming

1. The control of the master robot via a cell phone is achieved through an Android application (RoboComm.apk) developed on MIT App Inventor[2].

   Figure 6 shows a screenshot of the app in the developing process.



AlignHorizontal: Center
AlignVertical: Top
BackgroundColor: White
BackgroundImage: None...
CloseScreenAnimation: Default
Icon: None...
OpenScreenAnimation: Default
ScreenOrientation: Unspecified
Scrollable: ☑
Title: RoboComm
VersionCode: 1
VersionName: 1.0

Figure 6

2. The Arduino UNO board is burnt with the following code to intercept the Bluetooth signals and drive the motors accordingly. The code is written in embedded C language and developed in Arduino software.

```
//DEFINITIONS
int motor1_A = 3; // pin 2 of L293D
int motor1_B = 4; // pin 7 of L293D
int motor2_A = 8; // pin 10 of L293D
int motor2_B = 9; // pin 15 of L293D

int state;
int flag=0;
int stateStop=0;

void setup() {

    // setting the pins as output
    pinMode(motor1_A, OUTPUT);
    pinMode(motor1_B, OUTPUT);
    pinMode(motor2_A, OUTPUT);

    // initializing serial communication at 9600 bits per second
    Serial.begin(9600);
}
```

[2] Website: http://beta.appinventor.mit.edu/

```
void loop() {

    // If the “up” button is pressed, the robot will move forward
    if (state == '1') {
        digitalWrite(motor1_A, HIGH);
        digitalWrite(motor1_B, LOW);
        digitalWrite(motor2_A, LOW);
        digitalWrite(motor2_B, HIGH);

        if(flag == 0){
          Serial.println("Move Forward"); // For debugging purpose
          flag=1;
        }
    }

    // If the “left” button is pressed, the robot will turn left
    else if (state == '2') {
        digitalWrite(motor1_A, LOW);
        digitalWrite(motor1_B, LOW);
        digitalWrite(motor2_A, LOW);
        digitalWrite(motor2_B, HIGH);

        if(flag == 0){
          Serial.println("Turn Left"); // For debugging purpose
          flag=1;
        }
        delay(1500);
        state=3;
        stateStop=1;
    }


    // If the “red” button is pressed, the robot will stop
    else if (state == '3' || stateStop == 1) {
        digitalWrite(motor1_A, LOW);
        digitalWrite(motor1_B, LOW);
        digitalWrite(motor2_A, LOW);
        digitalWrite(motor2_B, LOW);

        if(flag == 0){
          Serial.println("Stop"); // For debugging purpose
          flag=1;
        }
        stateStop=0;
    }

    // If the “right” button is pressed, the robot will turn right
    else if (state == '4') {
        digitalWrite(motor1_A, HIGH);
        digitalWrite(motor1_B, LOW);
        digitalWrite(motor2_A, LOW);
        digitalWrite(motor2_B, LOW);

        if(flag == 0){
          Serial.println("Turn Right"); // For debugging purpose
          flag=1;
        }
        delay(1500);
        state=3;
        stateStop=1;
    }

    // If the “down” button is pressed, the robot will move backward
    else if (state == '5') {
        digitalWrite(motor1_A, LOW);
        digitalWrite(motor1_B, HIGH);
        digitalWrite(motor2_A, HIGH);
        digitalWrite(motor2_B, LOW);

        if(flag == 0){
          Serial.println("Move Backward"); // For debugging purpose
          flag=1;
        }
    }

}
```

- **Appendix B – Specifications of Bluetooth and RF module**

  HC-05 Bluetooth module:-
  Bluetooth protocol: Bluetooth Specification v2.0+EDR
  Frequency: 2.4GHz ISM band
  Modulation: GFSK (Gaussian Frequency Shift Keying)
  Emission power: ≤4dBm, Class 2
  Sensitivity: ≤-84dBm at 0.1% BER
  Speed: Asynchronous: 2.1Mbps (Max) / 160 kbps, Synchronous: 1Mbps/1Mbps
  Profiles: Bluetooth serial port
  Power supply: +3.3 - +5 VDC 50mA

  RF Module:-
  Transmitter side
  Working voltage: 3V - 12VDC
  Working current: 9mA – 40 mA
  Modulation mode: ASK
  Working frequency: 433MHz
  Transmission power: 25mW (315MHz at 12V)
  Frequency error: +150kHz (max)
  Velocity : less than 10Kbps

  Receiver side:-
  Working voltage: 5.0VDC +0.5V
  Working current ≤5.5mA max
  Working method: ASK
  Working frequency: 433 MHz
  Bandwidth: 2MHz
  Transmitting velocity: <9.6Kbps (at 315MHz and -95dBm)

- **Appendix C – Work Distribution**

  Bhuvnesh Bhatia has implemented all the hardware required to the detail. His contribution to the project includes –

  - ✓ Building the individual robots
  - ✓ Interfacing Bluetooth module, Arduino UNO, RF transmitter, HT12E encoder IC and L293D IC together for the master robot
  - ✓ Interfacing RF receiver, HT12D decoder IC and L293D IC together for the slave robot

Apart from this, he can efficiently debug and/or modify the Arduino program used.

Ankita Srivastav has employed all the software required to the detail. Her contribution to the project includes –

- ✓ Learning MIT App inventor to develop the application and learning embedded C language
- ✓ Developing the RoboComm Android Application
- ✓ Writing the Arduino code and debugging it

Along with this, she can proficiently modify and/or correct the circuit, is required.

References

1. Pin diagram of Arduino UNO: http://www.hwkitchen.com/products/arduino-mega-2560/
2. Pin Diagram of L293D: http://shashank-kulkarni.blogspot.in/2011/05/driving-circuits-for-dc-motors.html
3. Pin Diagram of HT12E and HT12D: http://homemaderobo.blogspot.in/2012/05/434mhz-rf-modules-with-encoders-and.html
4. Pin Diagram of RF module: http://www.engineersgarage.com/electronic-components/rf-module-transmitter-receiver
5. Pin Diagram of Bluetooth module: http://www.elecfreaks.com/wiki/index.php?title=Bluetooth_Modem
6. Specifications of Bluetooth and RF module: http://www.instructables.com/id/RF-315433-MHz-Transmitter-receiver-Module-and-Ardu/