Hrutvik Patel

Karan Patel

Mr. Rosen

TEJ4M1

June 7 2016

Computer Engineering Culminating: R/C Car

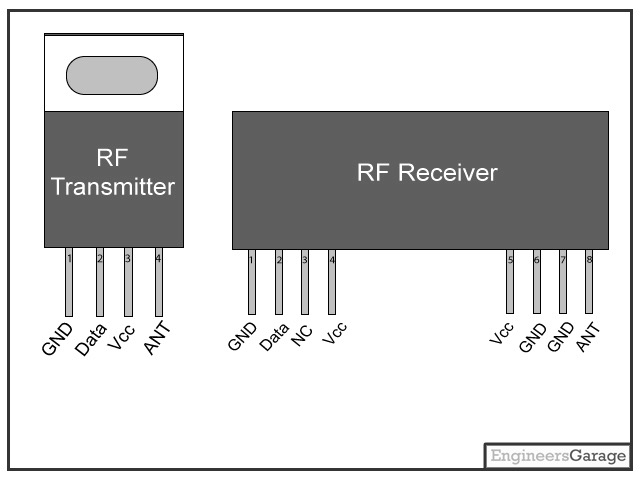
****



Table of Contents

TEJ4M1 Summative Project Proposal 3

An introduction/ Project Description: 4

• What is it? 4

• What does it do? 4

• How do you operate it? 4

Theory of Operation: 5

• How does it works electrically? 5

• How does it works mechanically? 5

Construction Details & Results: 6

• How we constructed it and trouble-shot it: 6

• Problems we encountered and how we overcame them: 6

• How it turned out in the end: 7

Schematic: 8

Code: 10

Picture of final product: 16

Reflection: 17

# TEJ4M1 Summative Project Proposal

**Project Idea:** Remote Controlled Car using the RF Transmitter And Receiver

**Date:** May 12th, 2016

**Team Members:** Hrutvik Patel, Karan Patel

**Project Description/Criteria :**

**Description:**

Using our knowledge in microcontrollers and what we’ve learned this year, for our culminating project we will be making a Remote Control Car using two arduino uno boards. One board will have an RF Transmitter and buttons that will transmit the signal from the remote controller and the other board will have the RF Receiver that will operate the motors on the car.

**Radio Control logic:**

By setting the two boards at the same frequency and using the RF Transmitter and RF receiver we will be able to use radio signals to communicate between the remote controller and the car body.

**Car mechanics:**

The controller will consist of 4 buttons that will each control 1 motor on our car.

To move the car forward the backmost motors will work simultaneously.

To steer to the right the bottom left motor will turn on causing the car to shift rightwards.

To steer to the left the bottom right motor will turn on causing the car to shift leftwards.

To move the car backwards the topmost motors will work simultaneously.

**Schematic:** Attached with proposal

**Sketch:** Attached with proposal

**Timeline**:

* May 24th - Complete construction of the RC Car along with its controller
* May 30th - Wire the RC car and the controller to the arduino UNO board and test each part
* June 1st - Finale diagnosis of the hardware for functionality
  + - Test the motors and the buttons of the controller
* June 3rd - Star testing our code
* June 9th - We will have the complete working hardware and software part of project ready to present for marking during class

# An introduction/ Project Description:

## • What is it?

`Our project is a wireless (Radio controlled) car. It uses a transmitter and receiver to communicate with 2 Arduino boards, one for the controller (transmitter) and one for the car body (receiver). The backbone of our car is the Arduino microcontroller, which we wired to allow the controls to work and programmed to operate the car.

## • What does it do?

We made our car so it can reach a destination without physically touching it. Like any other wireless car, our car can accelerate forwards/backwards and turn left/right as the appropriate switches are pressed on the controller.

## • How do you operate it?

To operate our car we have a controller that controls the movement of our car. In total we have 4 switches on our controller. A switch for accelerating forward, a switch for accelerating backwards, a switch for turning right and finally a switch for turning left.

# Theory of Operation:

## • How does it works electrically?

The functionality of the car is behind the electronics of the car. The electronics our car consists of are motor, servo, RF transmitter, RF receiver, transistors, switches, resistors and wires. To make the car move we have motors that turn, spinning the wheel. The radio control of the car is achieved through the use of the RF transmitter and RF receiver. The receiver and transmitter are set at the same frequency so they can communicate with each other. The transmitter is placed on the controller and the receiver is placed on the car body. The microcontroller sends a high signal to the motors to make them turn. An H-bridge is used to get the motor to turn both ways allowing forward and backwards movement. The servo is programmed to turn a specific amount so you can control turning the car left and right.

## • How does it works mechanically?

The mechanics of the car include a car body designed by us that can hold an Arduino board, an engine (motor) and an axle (servo). The axle turns the wheels by using a servo that is connect to the wheels . We programmed the servo to turn at the right amount to allow the car to move left and right on command. Unfortunately the front wheels of our car aren't the same as the back wheels so our car isn't as smooth and balanced as we'd like it to be.

# Construction Details & Results:

## • How we constructed it and trouble-shot it:

Before we attempt to construct anything we had a clear vision in our head and planned the day before what we will need and how we will construct it.

To construct our car we used materials like: plastic, nuts and bolts, popsicle stick, a thick wire (to control steering when turning).

First we built the car shape with the plastic and drilled a hole on for the chassis. Then we built a frame out of popsicle sticks to hold the chassis in place and implemented the chassis using nuts, bolts and the wire and finally we glued our electronic components (servo, engine, microcontroller).

To trouble shoot our project we would work in steps and test each step. For example our first step was testing to see if our engine worked: to trouble shoot this we simply sent a high signal to the motor to see if it turns. Another problem we trouble shot was our H-bridge: to trouble shoot it we sent a high signal at 2 different pins to see if it would turn forwards/backwards. (we got help from Mr. Rosen and he trouble shot the H-bridge for us because we weren’t able to find the mistake). Finally we trouble shot our transmitter and receiver and checked to see if we can wirelessly communicate between Arduinos.

## • Problems we encountered and how we overcame them:

The first problem we encounter was with the base of our car, initially it wasn’t strong enough and to compensate for that we switched out our Styrofoam for hard plastic.

Another ongoing problem that we encountered was lack of time due to cricket games. Karan and I both had cricket games so there was no partner that could work on the project while there was a game going on. To overcome this problem we planned what we were going to do the classes we missed beforehand and work more productively.

Another “problem” we faced was with how we were going to turn the car: I proposed to have two additional motors in the front where when one is turned on and the other is off the car will move a certain direction. But Karan had a better idea in mind with the use of a servo. This idea was more effective and turned out to be pretty simple to implement.

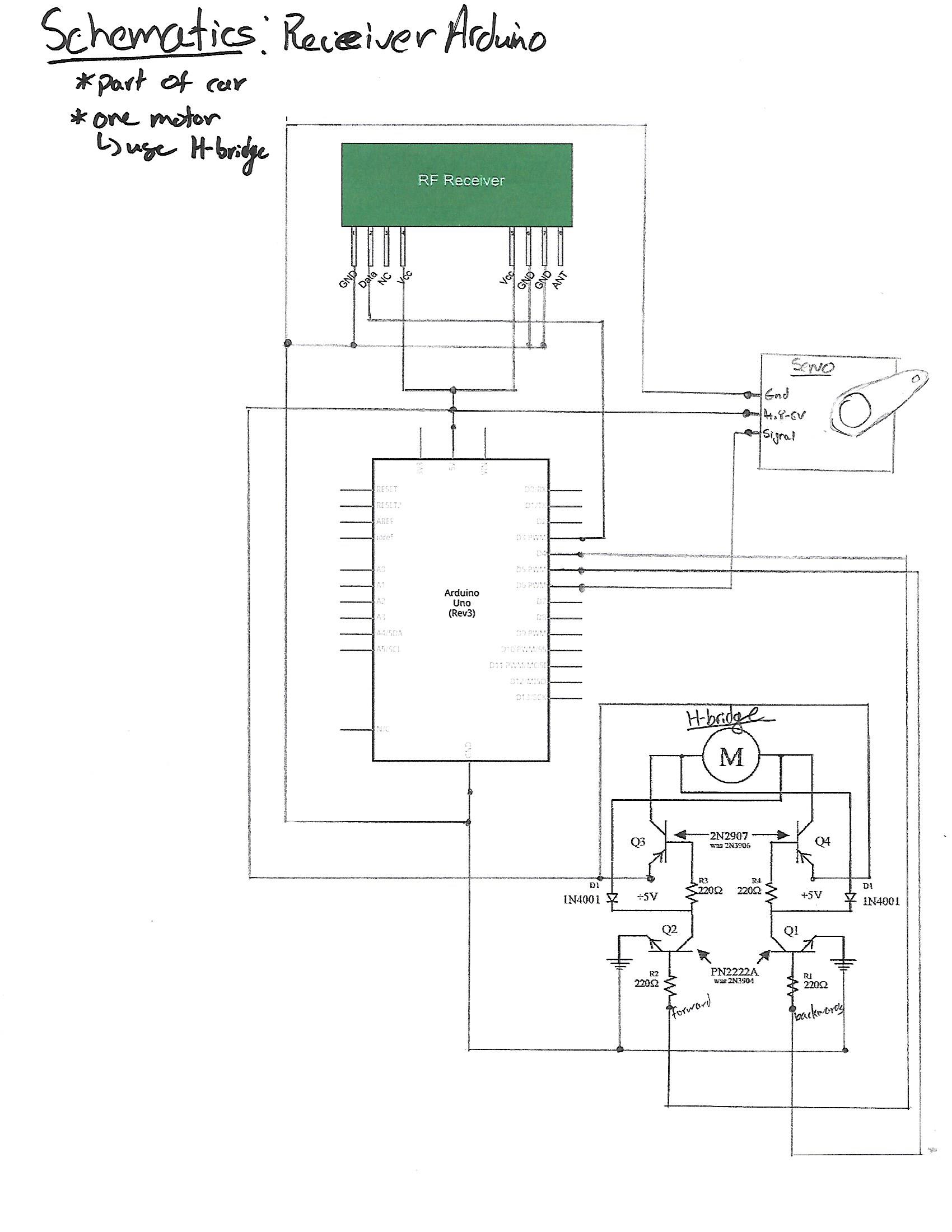
Finally we had a problem with our H-bridge which Mr. Rosen helped us fix. The problem was with the diodes as one of them wasn’t working properly.

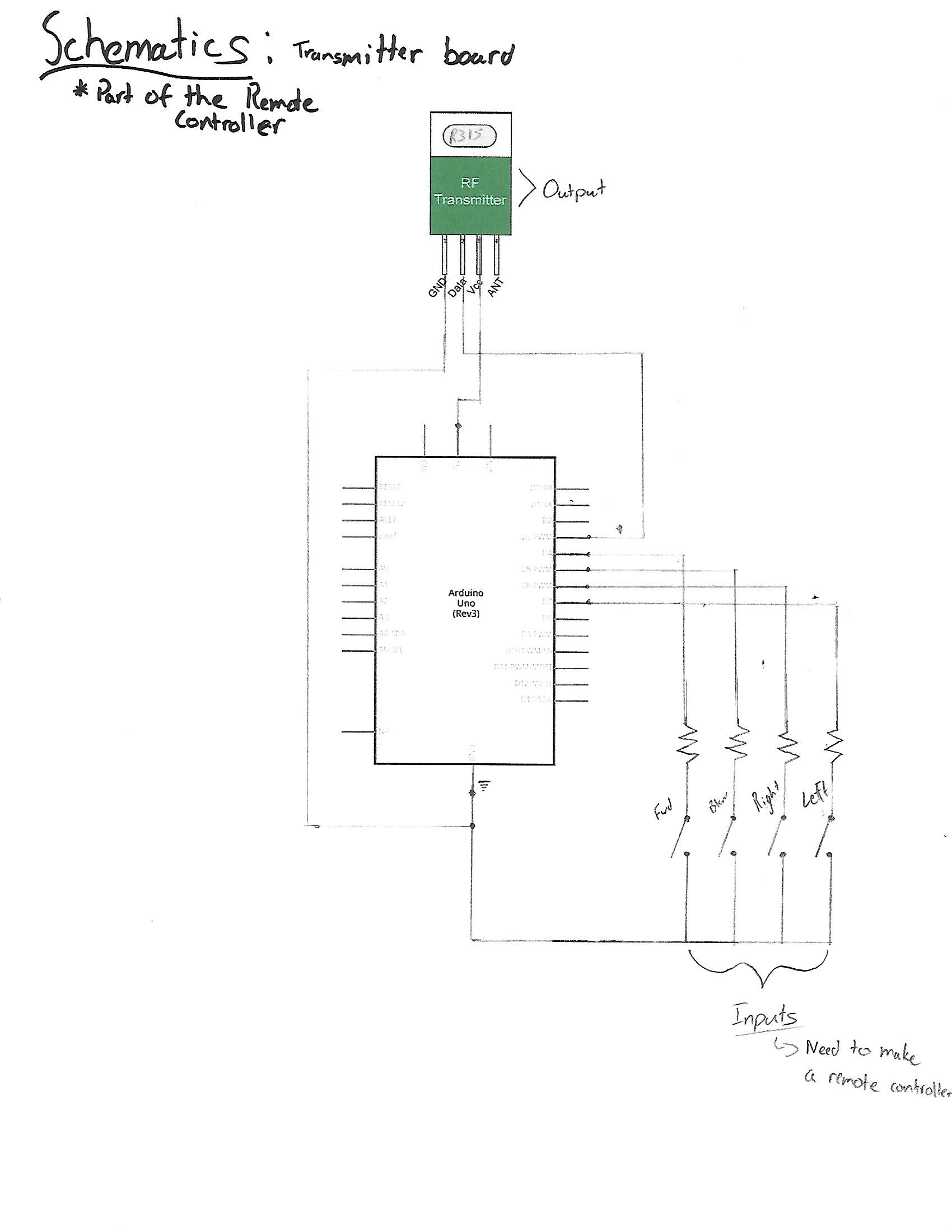
## • How it turned out in the end:

Originally our car had a base made of Styrofoam where all the individual parts were on the Styrofoam, however the Styrofoam wasn’t robust enough and we switched our material to a hard plastic-like board.

Initially we had our minds set on using 4 motors but at the end we used 1 motor and 1 servo. With the use of an H-bridge we were able to get the car to move forwards and backwards without having to use extra motors. Overall I like how our car turned out in the end because we used less material than required and still had a functioning car.

# Schematic:





# Code:

// Reciever Code for RC CAR

// Karan & Hrutvik

// 08/06/1016

#include <VirtualWire.h>

#include <Servo.h>

Servo servo;

int forward = 4;

int backwards = 5;

void setup() {

Serial.begin(9600);

pinMode(forward, OUTPUT);

pinMode(backwards, OUTPUT);

servo.attach(6);

vw\_setup(2000); //begins using the VirtualWire settings and intializes the library

vw\_set\_rx\_pin(3); // set pin for transmitting as pin 3;

vw\_rx\_start();

}

void loop() {

digitalWrite(forward, LOW);

digitalWrite(backwards, LOW);

// message of 8 bit int gets storred in a array

uint8\_t output[VW\_MAX\_MESSAGE\_LEN];

uint8\_t outputLength = VW\_MAX\_MESSAGE\_LEN; //length of message

const char F = 'f', B = 'b', R = 'r', L = 'l';

if(vw\_get\_message(output, &outputLength)){

int i;

for(i = 0; i < outputLength; i++){

Serial.print(output[i]);//check

if(output[i] == F){

digitalWrite(forward, HIGH);

}else if(output[i] == B){

digitalWrite(backwards, HIGH);

}else if(output[i] == R){

for(int servoPos = 0; servoPos < 180; servoPos ++){

servo.write(servoPos); // write to servo

delay(10);

}

}if(output[i] == L){

for(int servoPos = 180; servoPos >= 1; servoPos --){

servo.write(servoPos); // write to servo

delay(10);

}

}

Serial.println("");

}

}

}

// Transmitter Code for RC CAR

// Karan & Hrutvik

// 08/06/1016

#include <VirtualWire.h>

//\*\*\*\*FOR TRANSMITTING\*\*\*\*

int forward = 4;

int backwards = 5;

int right = 6;

int left = 7;

void setup(){

pinMode(forward, INPUT);

pinMode(backwards, INPUT);

pinMode(right, INPUT);

pinMode(left, INPUT);

digitalWrite(forward, HIGH);

digitalWrite(backwards, HIGH);

digitalWrite(right, HIGH);

digitalWrite(left, HIGH);

Serial.begin(9600);

vw\_setup(2000); //begins using the VirtualWire settings and intializes the library

vw\_set\_tx\_pin(3); // set pin for transmitting as pin 3;

}

void loop() {

const char \*F = "f", \*B = "b", \*R = "r", \*L = "l";

//strlen > returns the size of the string stored in the array

// > counts the visible characters and not the null characters

//uint8\_t \* > char array of unsigned 8 bit integer

//char with \* > is pointer, it contains the address value in memory

if(digitalRead(forward) == LOW){

vw\_send((uint8\_t \*)F, strlen(F));

vw\_wait\_tx(); // wait to send message

}else if(digitalRead(backwards) == LOW){

vw\_send((uint8\_t \*)B, strlen(B));

vw\_wait\_tx(); // wait to send message

}else if(digitalRead(right) == LOW){

vw\_send((uint8\_t \*)R, strlen(R));

vw\_wait\_tx(); // wait to send message

}else if(digitalRead(left) == LOW){

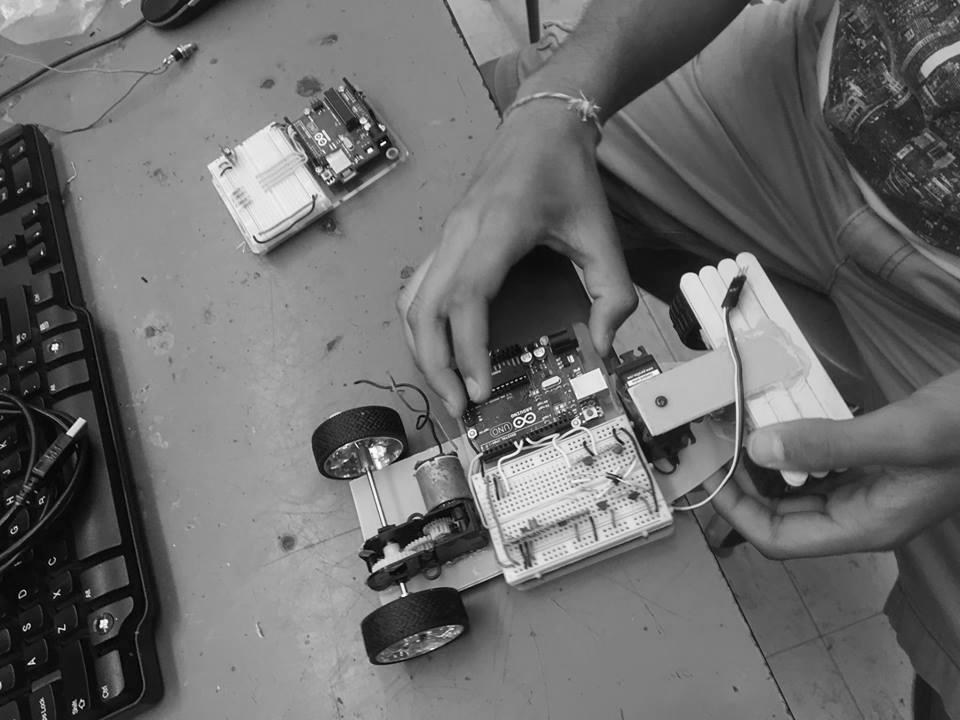
vw\_send((uint8\_t \*)L, strlen(L));

vw\_wait\_tx(); // wait to send message

}

}

# Picture of final product:





# Reflection:

From this project I learned more about microcontrollers and how to communicate between Arduinos using a RF transmitter and RF receiver. Another thing I learned was how an H-bridge can be used to make a motor spin both ways.

As a team Karan and I worked really well and managed our time wisely. Since we both had to attend cricket games we were at a disadvantage compared to the other teams, however we managed to pull through by planning ahead and working more productively in class.

This project required a lot of planning and research, followed by gathering the correct material, constructing the car, trouble shooting and finally making the code. Overall the distribution of work was even and we both agreed with how the work was distributed. For the research The radio control logic was done by me, Karan made the schematics and I made the initial isometric design. While gathering the material I had an "engine" at home (gears and a motor) from a toy car and Karan had plastic for the base. The school provided the remaining material. For the construction Karan cut out the car base and made the chassis and we both wired the car. We both troubleshot the car and Karan made the code as he skipped a cricket game to work on the project.