EE 40 Final Project Report

Charles Lee, Xinhe Ren

1. Feature Descriptions
   a. *H-Bridge*

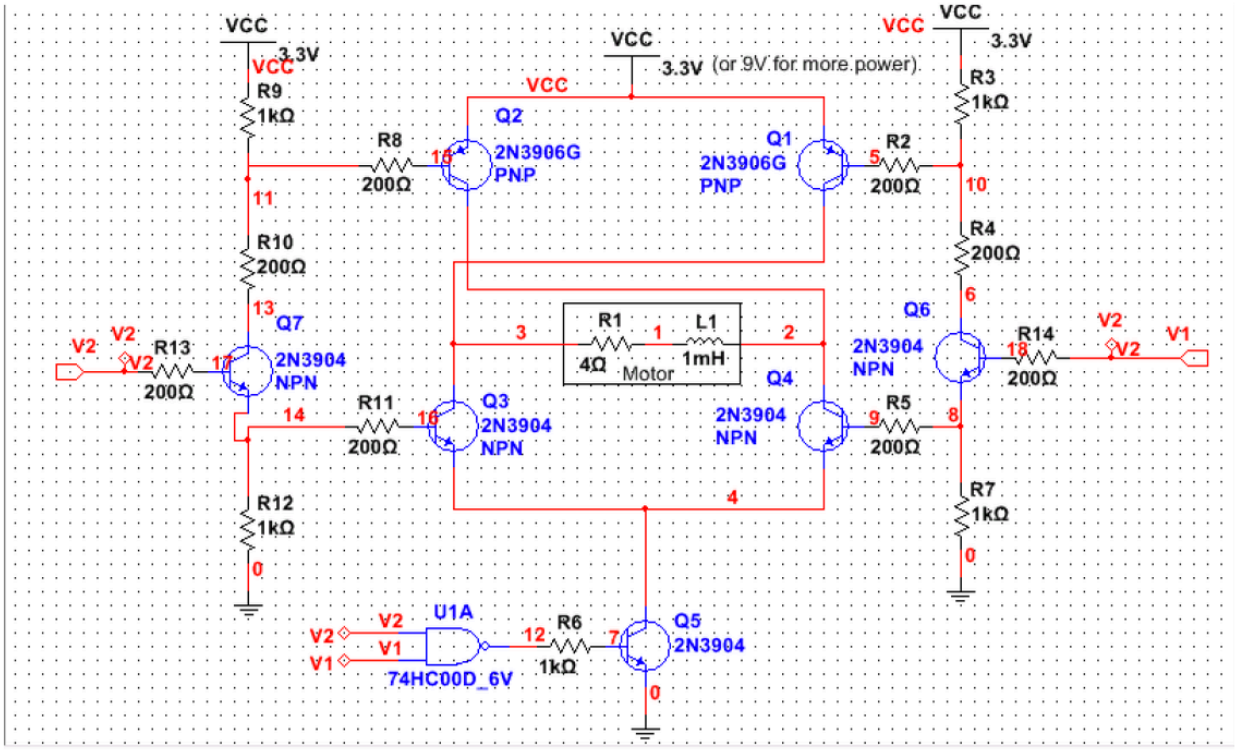
      Two H-bridges (one for each wheel) control the steering of the robot car. The H-bridge consists of 2 PNPs and 4 NPSs. The input V_{cc} is 9V, and is used to power the motors. The two control signal inputs are from MSP430, and are at 3.3V at HIGH, 0V at LOW.

   b. *Myo Armband + Bluetooth*

      A Myo armband uses user's EMG data to control the motion of the car. Five different hand gestures are mapped to different driving commands (forward, left turn, emergency stop, etc). A HC-05 Bluetooth module is used with MSP430 to receive wireless commands from laptop, which reads inputs from Myo.

2. Circuit Schematics

   H-Bridge (courtesy of Michael):



3. MSP430 Code

```
const int LEFT_L = P1_4;                    // Constants for H-bridge inputs.
const int LEFT_R = P1_5;
const int RIGHT_L = P2_5;
const int RIGHT_R = P2_4;

void setup()                                // Initial pin setup.
{
  pinMode(LEFT_L, OUTPUT);
  pinMode(LEFT_R, OUTPUT);
  pinMode(RIGHT_L, OUTPUT);
  pinMode(RIGHT_R, OUTPUT);
  pinMode(RED_LED, OUTPUT);
  pinMode(GREEN_LED, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  if (shortCircuit())                       // Digitally check for shorting
    stopAll;                                Vcc to Gnd.

  int buf_size = Serial.available();        // Reading inputs from Bluetooth
  if (buf_size > 0) {                       serial.
    char input = Serial.read();
    if (input == '0') {
      stopAll();
      digitalWrite(RED_LED, HIGH);
      digitalWrite(GREEN_LED, LOW);
    } else if (input == 'U') {              // Drives car forward.
      leftForward();
      rightForward();
      digitalWrite(RED_LED, LOW);
      digitalWrite(GREEN_LED, HIGH);
    } else if (input == 'D') {              // Drives car backward.
      leftBackward();
      rightBackward();
       digitalWrite(RED_LED, LOW);
      digitalWrite(GREEN_LED, HIGH);
    } else if (input == 'L') {              // Turns car counterclockwise.
      leftBackward();
      rightForward();
       digitalWrite(RED_LED, LOW);
      digitalWrite(GREEN_LED, HIGH);
    } else if (input == 'R') {              // Turns car clockwise.
      leftForward();
      rightBackward();
       digitalWrite(RED_LED, LOW);
```

```
      digitalWrite(GREEN_LED, HIGH);
    }
  }
}

void leftForward() {                          // Turns left wheel forward.
  digitalWrite(LEFT_L, HIGH);
  digitalWrite(LEFT_R, LOW);
}

void leftBackward() {                         // Turns left wheel backward.
  digitalWrite(LEFT_L, LOW);
  digitalWrite(LEFT_R, HIGH);
}

void rightForward() {                         // Turns right wheel forward.
  digitalWrite(RIGHT_L, HIGH);
  digitalWrite(RIGHT_R, LOW);
}

void rightBackward() {                        // Turns right wheel backward.
  digitalWrite(RIGHT_L, LOW);
  digitalWrite(RIGHT_R, HIGH);
}

boolean shortCircuit() {                      // Returns TRUE if both inputs
  return LEFT_L == HIGH && LEFT_R == HIGH ||  for a H-bridge are HIGH
    RIGHT_L == HIGH && RIGHT_R == HIGH;       (shorting Vcc to Gnd).
}

void stopAll() {                              // Kill all motor motions.
  digitalWrite(LEFT_L, LOW);
  digitalWrite(LEFT_R, LOW);
  digitalWrite(RIGHT_L, LOW);
  digitalWrite(RIGHT_R, LOW);
}
```

4. Picture