

RoboBall – Semesterstand:

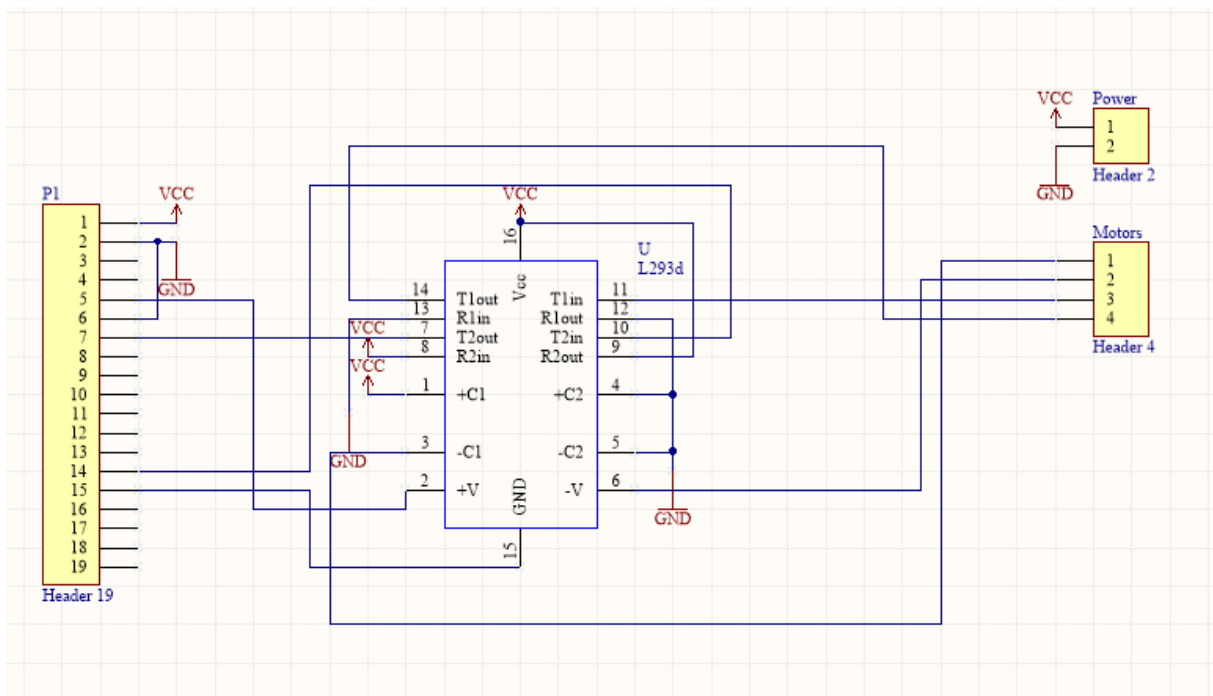
Was schon funktioniert:

Der unter Schematic gezeigte Schaltplan wurde schon auf dem Steckbrett aufgebaut und die Ansteuerung der Motoren über den ESP32 funktioniert auch schon. Die Auflade Schaltung ist vollständig verlötet (Ladeplatine und Akku). Über Bluetooth lassen sich schon vorwärts, rückwärts, links und rechts ansteuern.

Was noch fehlt:

Zu tun ist vor allem noch, dass alles zusammengefügt gehört, die Versorgungsspannung der Motoren gehört größer, dafür ist schon ein dc-dc Wandler in Planung. Die Ansteuerung der Motoren werden wir noch mit PWM verbessern um ein schönes Rollen zu erzeugen. Als Steuerung arbeiten wir auch noch an der App.

Schematic:



Programm:

```
#include "BluetoothSerial.h"

#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
#endif

const int frequency = 1000;
const int resolution = 16;
int lastchannel = 0;
BluetoothSerial SerialBT;

class Motor {
private:
    int pin1;
    int pin2;
```

```

    int ch1;
    int ch2;
    int Speed;

public:
    Motor() {}
    Motor(int, int, int, int);
    void drive();
    void setspeed(int);
    void backgroundActivity();
};

Motor::Motor(int p1, int p2, int freq, int res) {
    pin1 = p1;
    pin2 = p2;
    ch1 = lastchannel + 1;
    ch2 = lastchannel + 2;
    lastchannel += 2;

    ledcSetup(ch1, freq, res);
    ledcSetup(ch2, freq, res);
    ledcAttachPin(pin1, ch1);
    ledcAttachPin(pin2, ch2);
}

void Motor::drive() {
    if ( Speed == 0 ) {
        digitalWrite(pin1, LOW);
        digitalWrite(pin2, LOW);
    } else if (Speed > 0) {
        digitalWrite(pin2, LOW);
        ledcWrite(ch1, Speed);
    } else if (Speed < 0) {
        digitalWrite(pin1, LOW);
        ledcWrite(ch2, Speed * -1);
    }
}

void Motor::setspeed(int s)
{
    Speed = s;
}

void Motor::backgroundActivity()
{
    this->drive();
    this->Speed -= 1;
}

//Motor Right(35, 34, frequency, resolution);
//Motor Left(13, 12, frequency, resolution);
char command[30] = "";
int cmdp = 0;
int cmdavailable = false;

void setup() {
    pinMode(13, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(25, OUTPUT);
    pinMode(35, OUTPUT);

    Serial.begin(115200);
    SerialBT.begin("ESPDavid"); //Bluetooth device name
    Serial.println("The device started, now you can pair it with bluetooth!");
}

void loop() {
    int cmd;

```

```

if (SerialBT.available() && !cmdavailable ) {
    cmd = SerialBT.read();
    if (cmd == 10) {
        command[cmdp] = '\0';
        cmdp = 0;
        cmdavailable = 1;
    } else {
        command[cmdp++] = (char)lowByte(cmd);
    }

    if (cmdavailable) {
        cmdavailable = 0;
        Serial.println(command);
        if (command[0] == 'f') {
            digitalWrite(25, HIGH);
            digitalWrite(12, HIGH);
        } else if (command[0] == 'l') {
            digitalWrite(12, HIGH);
        } else if (command[0] == 'r') {
            digitalWrite(25, HIGH);
        } else if (command[0] == 'b') {
            digitalWrite(35, HIGH);
            digitalWrite(13, HIGH);
        } else if (command[0] == 's') {
            digitalWrite(25, LOW);
            digitalWrite(12, LOW);
            digitalWrite(35, HIGH);
            digitalWrite(13, LOW);
        }

        for(int i = 0; i < 30;i++) {
            command[i] = '\0';
        }
    }

    /*if(cmd == "stop") {
        Right.Speed(0);
        Left.Speed(0);
    }*/
}
//Right.backgroundActivity();
//Left.backgroundActivity();
}

```

Materialien:

- Spielzeugroboter
- Ladeplatine
- H-Brücken IC
- LiPo Akku 2500mAH
- ESP32