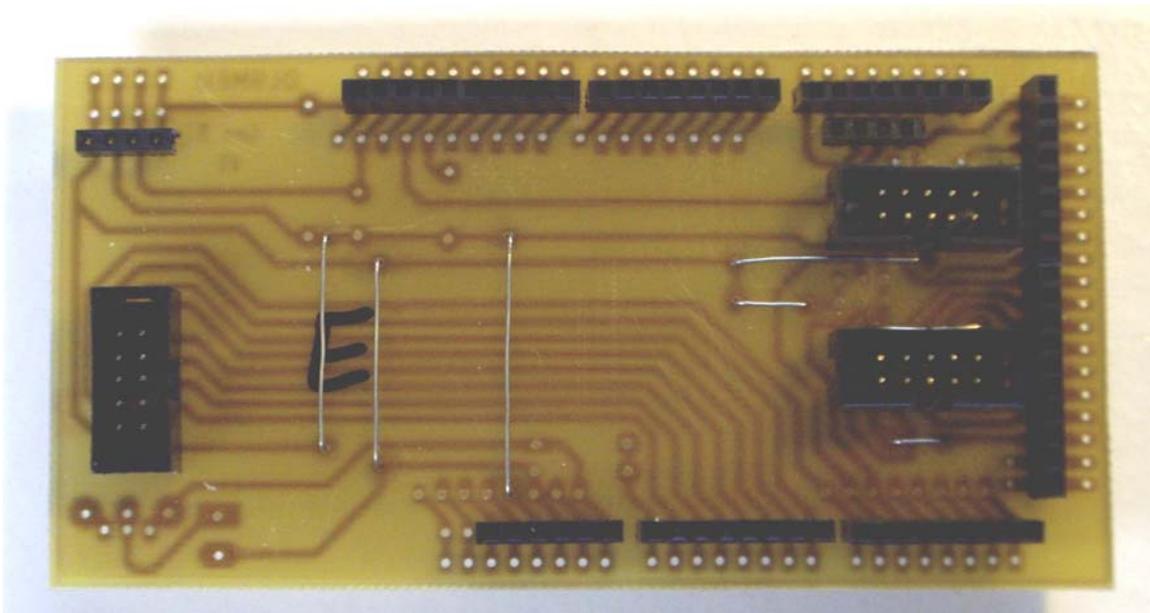


5. Die drei vierpoligen Buchsen für 2 x I2C-Bus und Bluetooth werden eingelötet.
6. Nun folgen Kippschalter und Piezo-Buzzer. Die bedruckte Seite beim Buzzer muss nach außen zeigen.
7. Jetzt wird eine vierpolige Stiftleiste für den Anschluss der Batterie eingelötet.
8. Und zuletzt die sechs Pfostenverbinder zum Arduino MEGA.
ACHTUNG: Die Pfostenverbinder so einsetzen wie das folgende Foto zeigt:



Die äußeren Lötzeihen oben und unten bleiben zunächst leer. Ebenso die beiden äußeren Lötzeihen rechts!

Bitte beim Löten darauf achten, dass kein Zinn oder Flussmittel an die Stifte kommt, die die Verbindung zum Arduino sicherstellen.

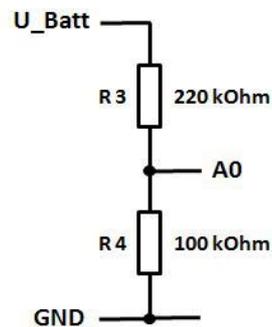
Nicht verwirren lassen: Foto zeigt eine ältere Version der Zwischenplatine.

ACHTUNG: Rechts an der Stelle, wo sich 8-poliger und 10-poliger Pfostenverbinder berühren, muss an beiden in der Mitte, wo sie sich berühren, mit einer Feile ca. 0,4 mm jeweils abgenommen werden.

9. Wenn die Zwischenplatine fertig ist, nehmen wir sie schrittweise in Betrieb. Als Erstes testen wir die Messung der Versorgungsspannung über den Spannungsteiler R3 = 220 kOhm / R4 = 100 kOhm.

Für die Messung der Versorgungsspannung mit dem Arduino gibt es ein kleines Demoprogramm:

Messung_U_Batt.cbp



$$\frac{U_{A0}}{U_{Batt}} = \frac{R4}{R3 + R4} = \frac{100}{220 + 100} = \frac{1}{3,2} = 0,313$$

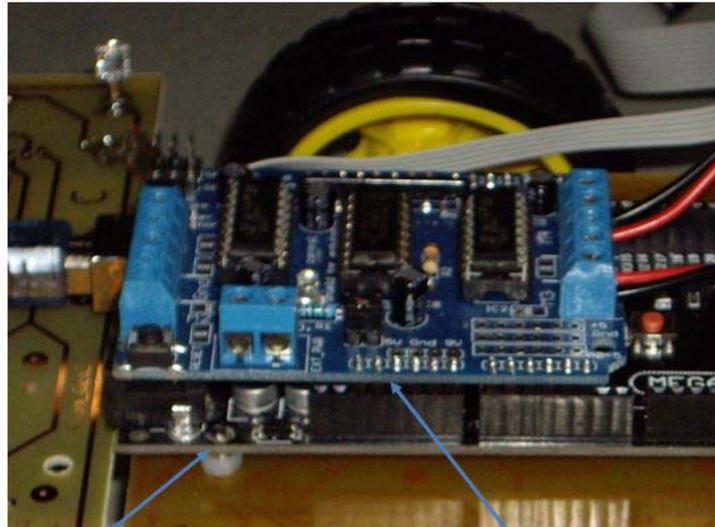
$$U_{Batt} = U_{A0} \times 3,2$$

Die Ausgabe erfolgt auf dem seriellen Monitor der Entwicklungsumgebung.

Eine Erfassung der Batteriespannung ist wichtig, da bei schwächer werdenden Batterien unser Roboter nicht mehr richtig funktioniert.

10. Jetzt machen wir die ersten Fahrversuche!

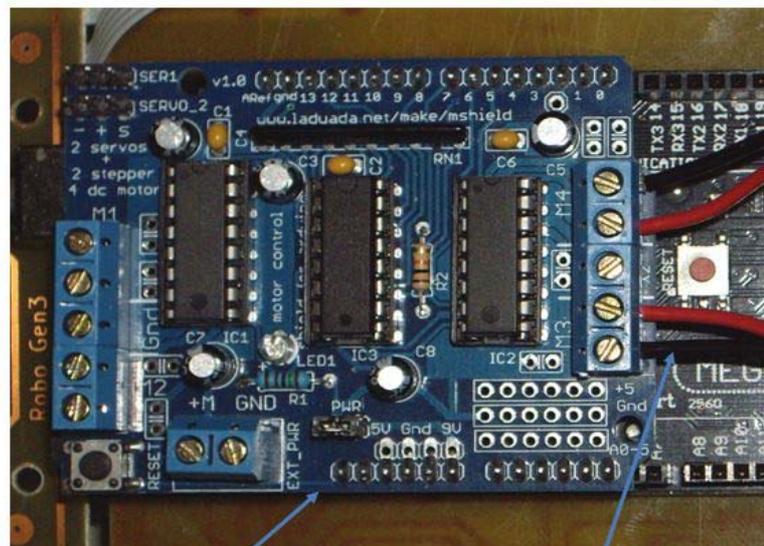
11. Dazu setzen wir die Motortreiber-Platine auf unseren Arduino MEGA:



Arduino MEGA

Motortreiber-Platine

Und schließen die Motoren an:



Motortreiber-Platine

Anschlussleitungen Motor

Jetzt die Stunde der Wahrheit. Wir laden das Testprogramm:

MotorTest_0.cbp

Die Geschwindigkeit der Motoren wird auf 200 bei Vorwärtsfahrt gesetzt

Dann fahren wir 150 ms (= 0,15 Sekunden) nach vorne. Wir halten an.

```
drive(MOTOR_RIGHT, FORWARD, 200); // Motor rechts vorwärts  
drive(MOTOR_LEFT, FORWARD, 200);  
  
delay (150); // x ms vorwärts fahren  
  
drive(MOTOR_RIGHT, RELEASE, 150); // stopp  
drive(MOTOR_LEFT, RELEASE, 150);
```

Und warten 0,4 Sekunden.

Anschließend geht es wieder für 150 ms zurück.