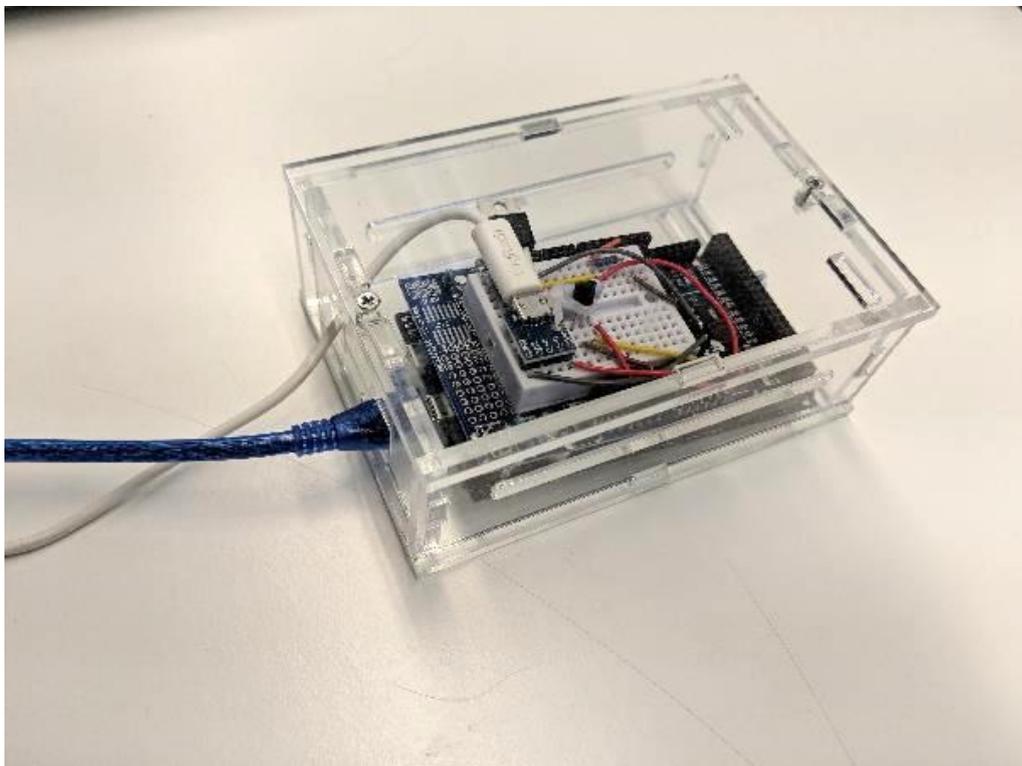
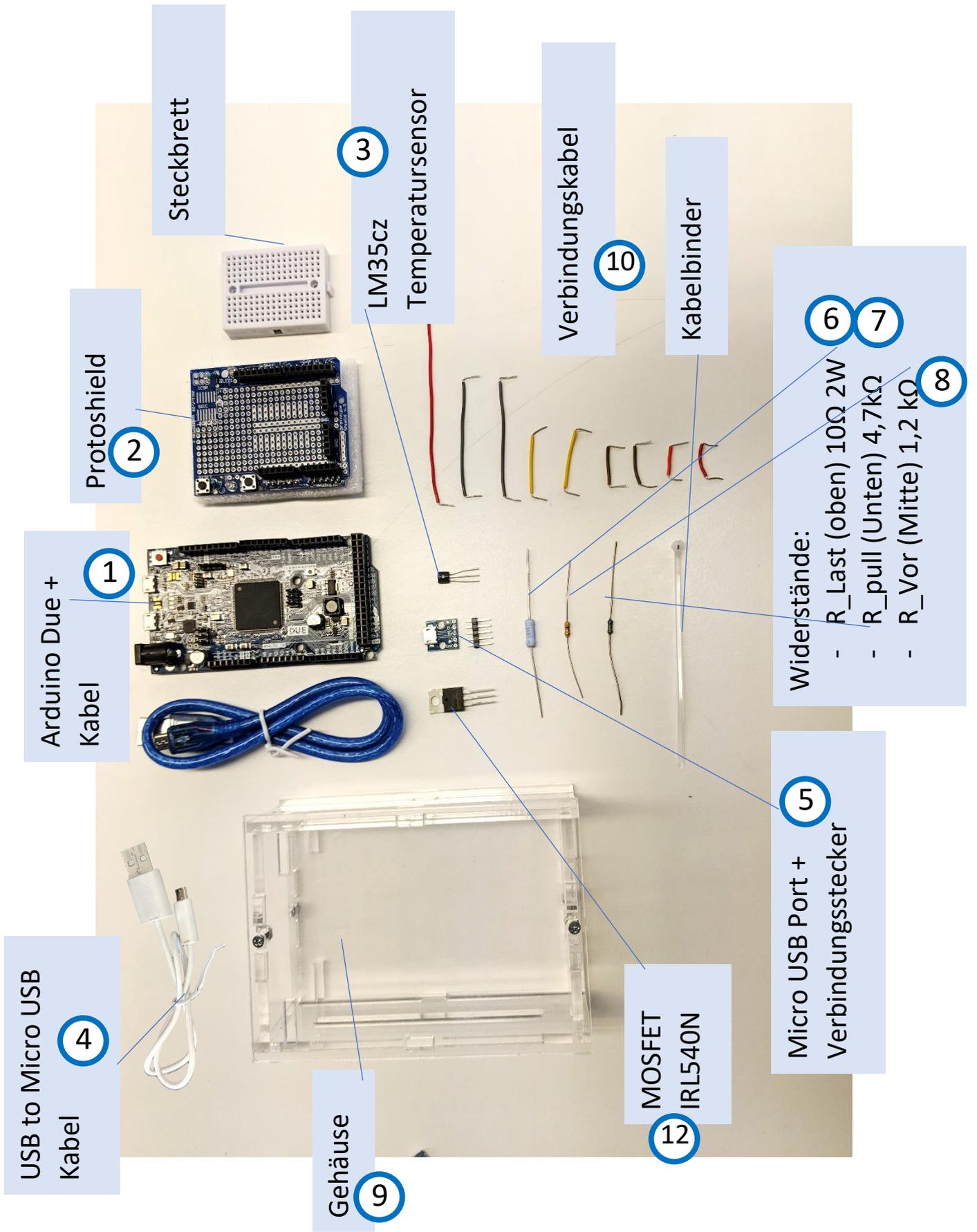


# ANLEITUNG ZUM AUFBAU EINES MOBILEN TEMEPERATUR- REGELUNGSVERSUCHS

13.03.2018, Version 1

Sebastian Fischer



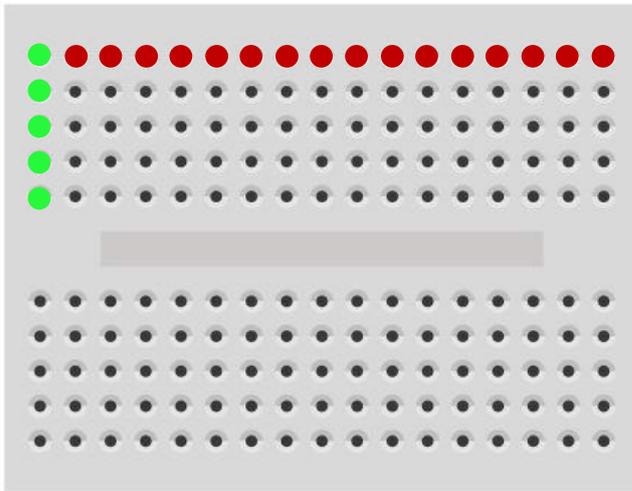


Anbei eine Stückliste aller Teile, Simulink Modelle und Matlab Files inkl. Kauf-Link:

| Nr    | Bezeichnung  | Anzahl | Preis   | Link                                 |
|-------|--|--------|---------|--------------------------------------|
| 1     | Arduino Due R3 SAM3X8E 32-bit ARM Cortex-M Control Board Module +Cable New | 1      | 13,42 € | <a href="#">Arduino Due</a>          |
| 2     | Protoshield + Steckbrett   | 1      | 6,29 €  | <a href="#">Shield</a>               |
| 3     | LM35cz Temperatur Sensor   | 1      | 3,35 €  | <a href="#">LM35cz</a>               |
| 4     | USB auf Micro USB Kabel  | 1      | 1,15 €  | <a href="#">USB / Micro USB</a>      |
| 5     | Micro USB port   | 1      | 3,97 €  | <a href="#">Micro USB Breakboard</a> |
| 6     | Leistungswiderstand Panasonic ERG-2SJ100 10R2W                             | 1      | 0,35 €  | <a href="#">Widerstand</a>           |
| 7     | 4,7 K Widerstand   | 1      | 0,10 €  |                                      |
| 8     | 1,2 K Widerstand   | 1      | 0,10 €  |                                      |
| 9     | Gehäuse  | 1      | 21 €    | <a href="#">Gehäuse</a>              |
| 10    | Verbindungskabel   | 1      | 3,89 €  | <a href="#">Kabel</a>                |
| 11    | MOSFET IRL 540N  | 1      | 0,70 €  | <a href="#">Mosfet irl 540N</a>      |
| 12    | Ventilator   | 1      | 14,03€  | <a href="#">Ventilator</a>           |
| 13    | Matlab 2018a Lizenz  | 1      | -       |                                      |
| 14    | SIMULINK Support Package for Arduino Hardware                              | 1      | -       |                                      |
| 15    | Simulink Support Package for Arduino Hardware                              | 1      | -       |                                      |
| 16    | Embedded Coder (Version 7.0)   | 1      | -       |                                      |
| 17    | System Identification Toolbox (Version 9.8)                                | 1      | -       |                                      |
| 18    | MATLAB Support Package for Arduino Hardware (Version 18.1.1)               | 1      | -       |                                      |
| 19    | MATLAB Support for MinGW-w64 C/C++ Compiler (Version 18.1.0)               | 1      | -       |                                      |
| 20    | Curve Fitting Toolbox (Version 3.5.7)                                      | 1      | -       |                                      |
| 21    | Bioinformatics Toolbox (4.10)  | 1      | -       |                                      |
| 22    | Uebertragungsfunktion.slx  | 1      | -       |                                      |
| 23    | Projektarbeit2_Steuerung.slx   | 1      | -       |                                      |
| 24    | Projektarbeit2_SteuerungMitGW.slx  | 1      | -       |                                      |
| 25    | Projektarbeit2_Regelung.slx  | 1      | -       |                                      |
| 26    | Projektarbeit2_DisturbanceCompensation.slx                                 | 1      | -       |                                      |
| 27    | Frequenzgang_Messung.slx   | 1      | -       |                                      |
| 28    | StartConditions_Tf.m   | 1      | -       |                                      |
| 29    | Bodeplot.m   | 1      | -       |                                      |
| 30    | Execute_FreqRespMeas.m   | 1      | -       |                                      |
| 31    | Exectue_Transferfunction.m   | 1      | -       |                                      |
| 32    | FrequResponse.m  | 1      | -       |                                      |
| 33    | Plots.m  | 1      | -       |                                      |
| 34    | Transfcn.m   | 1      | -       |                                      |
| Summe |  |        | 68,3 €  |                                      |



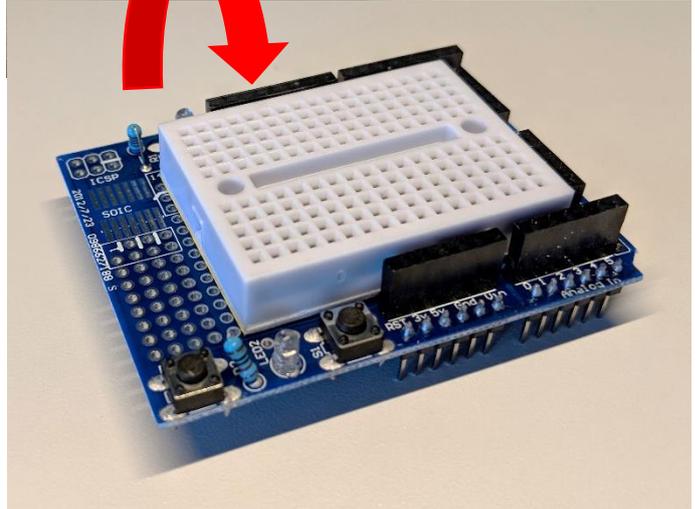
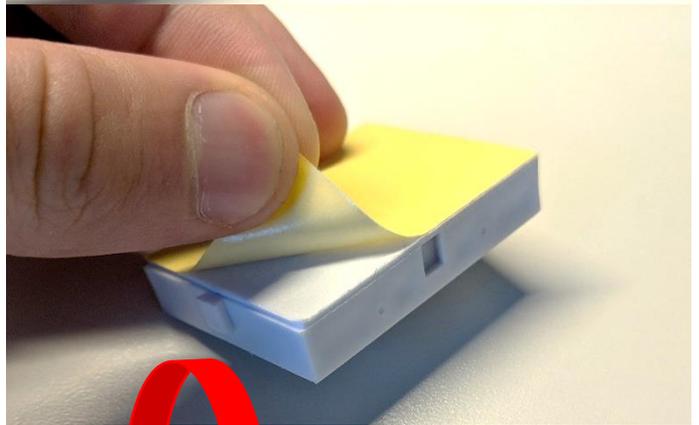
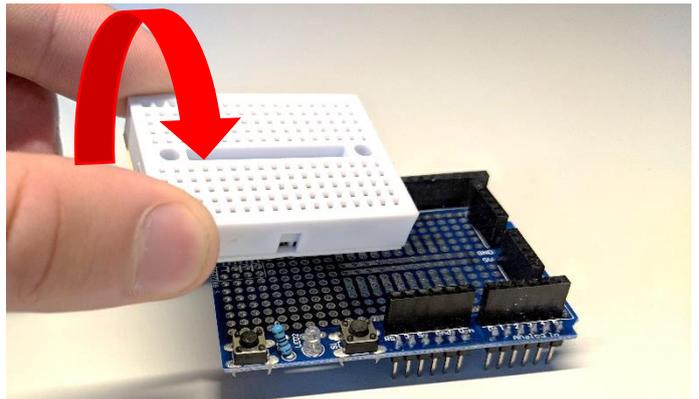
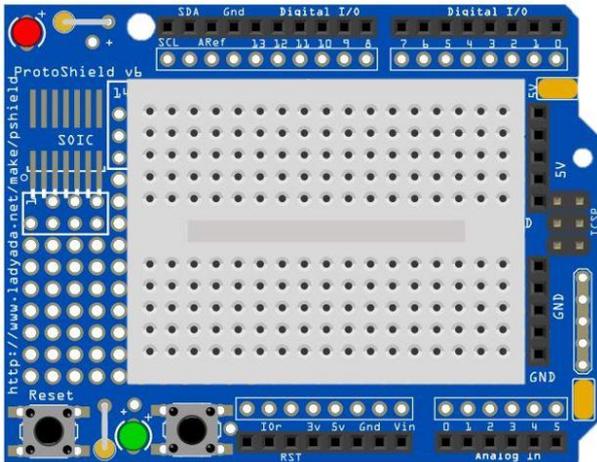
# Steckbrett



Die Verbindungen zwischen den Steckbrettlöchern verlaufen, in dieser Ansicht vertikal. Die grünen Punkte veranschaulichen dies zum besseren Verständnis. Löcher, welche sich horizontal voneinander befinden, sind nicht miteinander verbunden (siehe rote Punkte). Zudem ist die obere Hälfte des Steckbrettes nicht mit der unteren Hälfte verbunden.

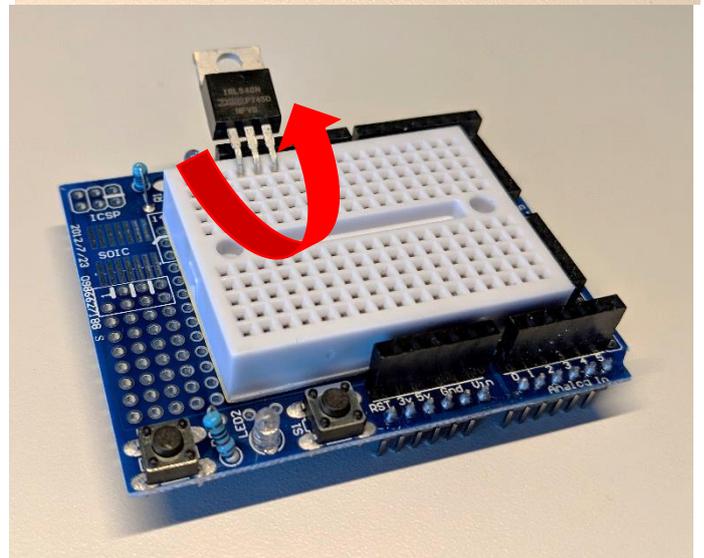
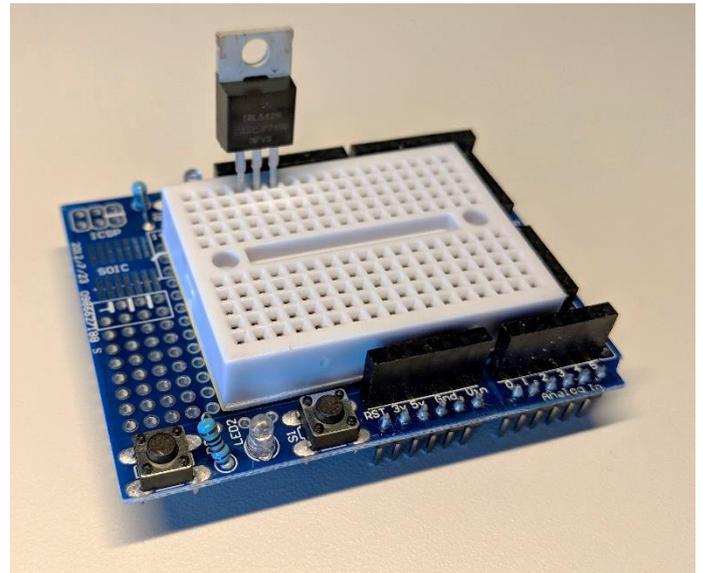
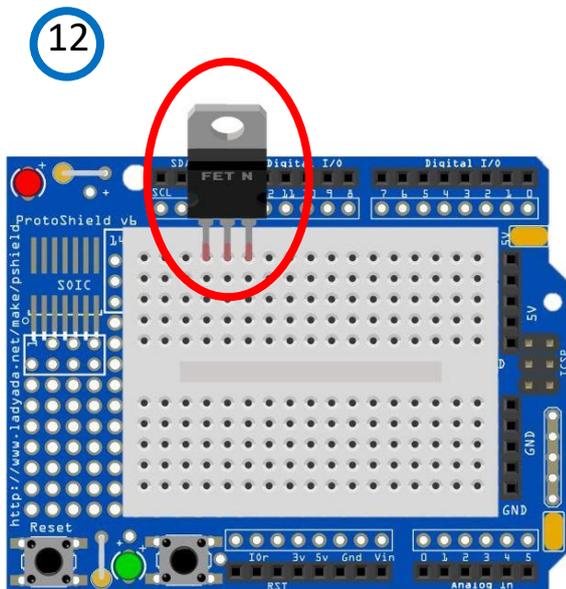
# Schritt 1

2



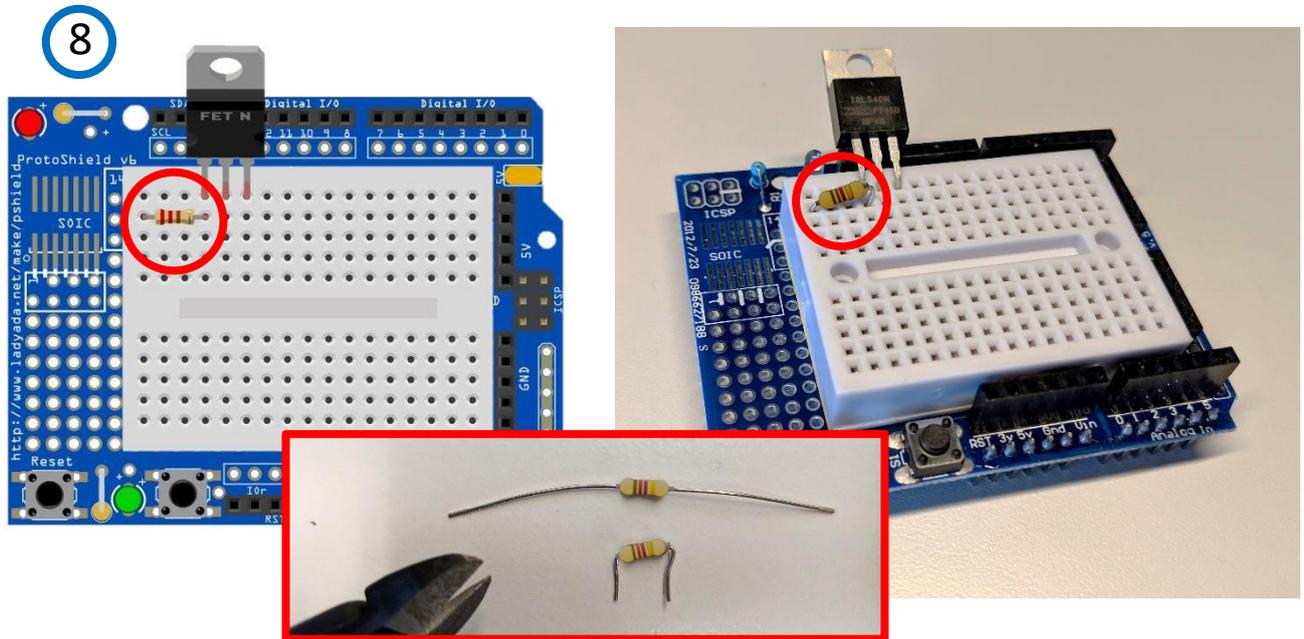
Zu aller erst muss die Folie auf der Rückseite des Steckbretts entfernt werden und das Steckbrett anschließend auf dem Protoshield wie in der Abbildung dargestellt draufgeklebt werden.

# Schritt 2



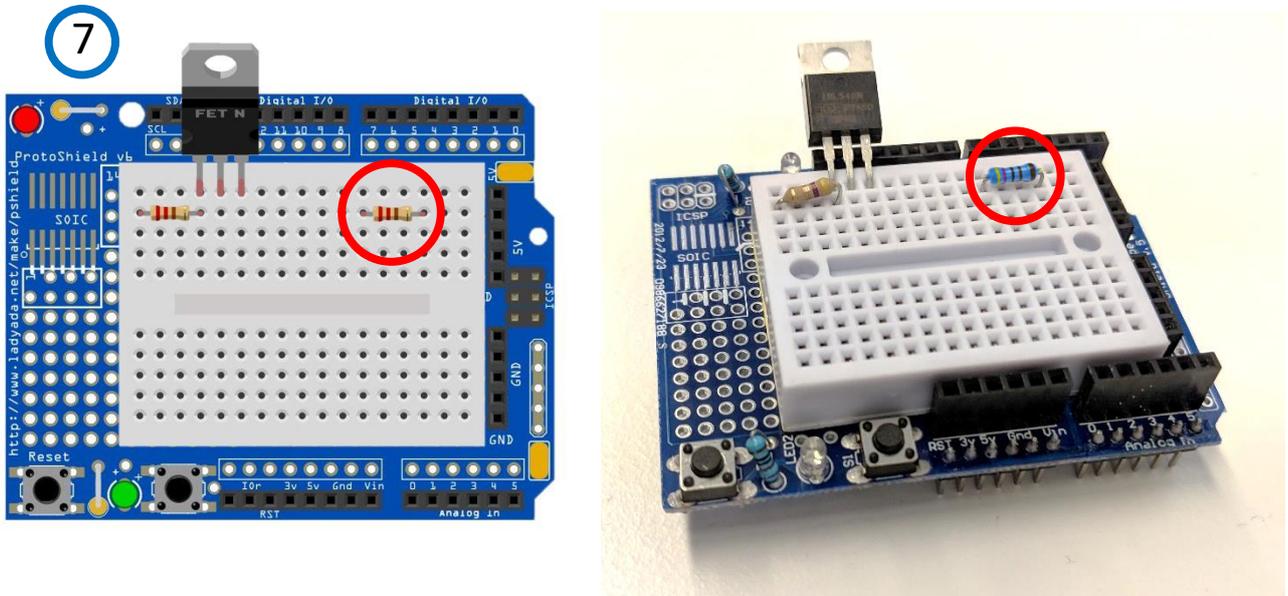
Hier sei darauf zu achten, dass der Mosfet richtig herum auf das Steckbrett gesteckt wird. Sprich die Fläche mit dem Loch ist hinten und die dunkle Fläche mit der Beschriftung vorne. Anschließend den Mosfet noch um ca. 90° nach hinten biegen, damit dieser später ins Gehäuse passt.

# Schritt 3



Die „Beine“ des 1,2 kΩ Widerstand (Farbcode: braun – rot – rot) müssen zunächst mit einem Seitenschneider gekürzt und umgebogen werden bevor er an gezeigter Stelle platziert wird.

# Schritt 4

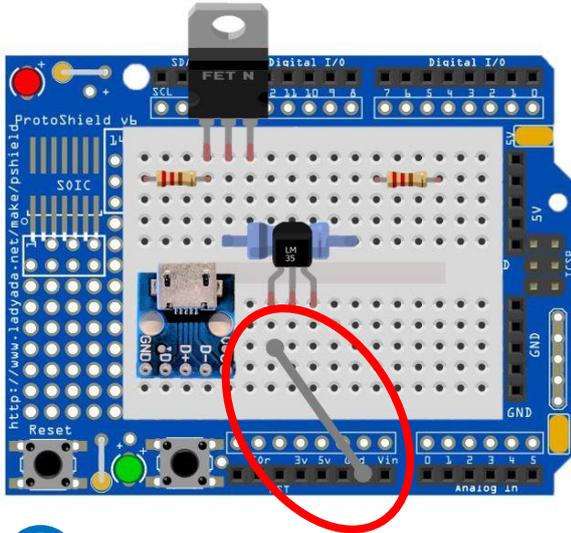


Auch die Beine des 4,7 kΩ Widerstandes (Farbcode: gelb – violett – schwarz – braun) müssen gekürzt und gebogen werden, bevor dieser an gezeigter Stelle aufgesteckt wird.



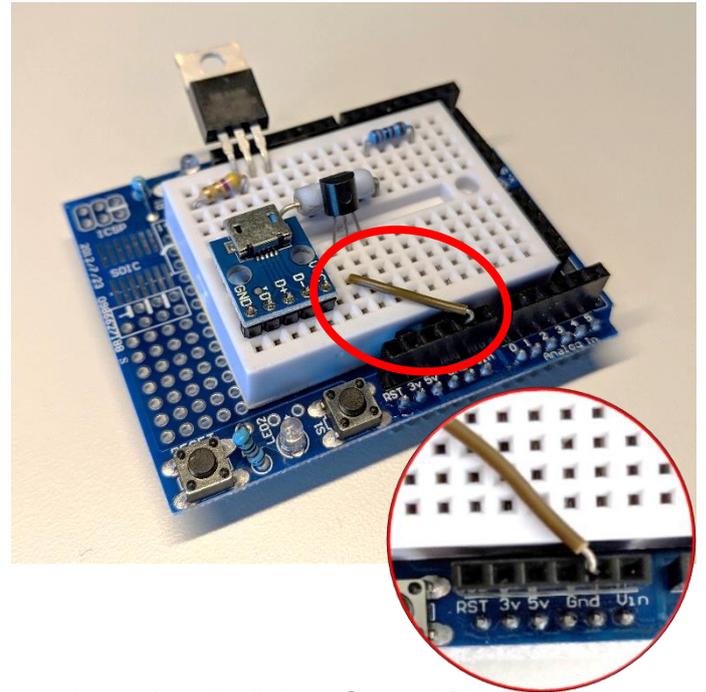


# Schritt 9

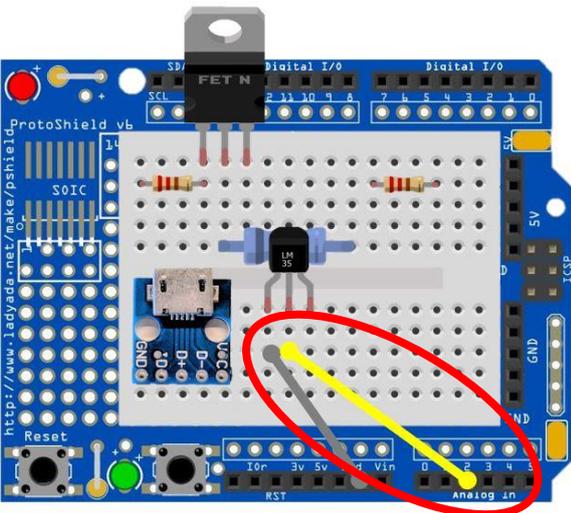


10

Der „Ground“ Fuß des Temperatursensors muss entsprechend mit dem Ground Pin des Proto Shields verbunden werden. Der zweite und dritte Pin von rechts sind beides GND Pins und somit frei wählbar.

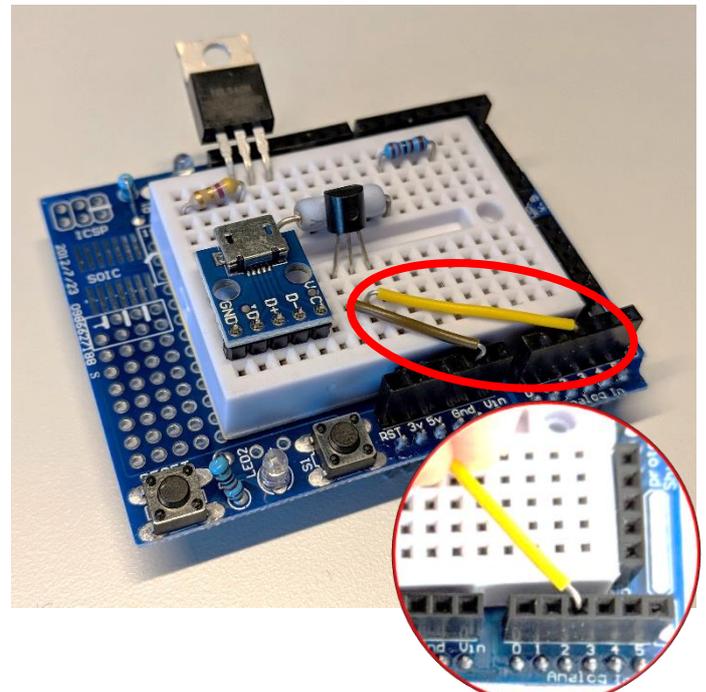


# Schritt 10

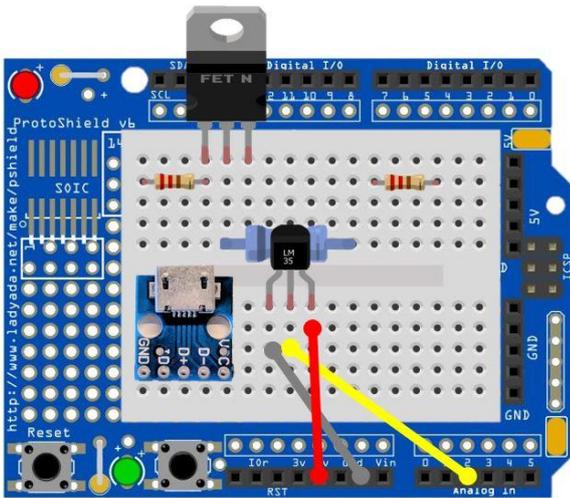


10

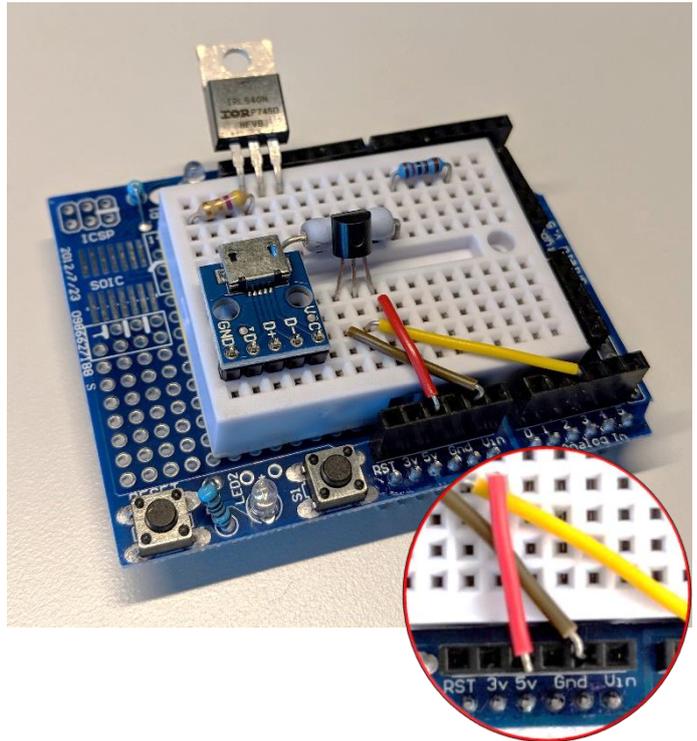
Der mittlere „Fuß“ des Temperatursensors wird mit dem Analog Pin 2 (dritter von links) verbunden.



# Schritt 11

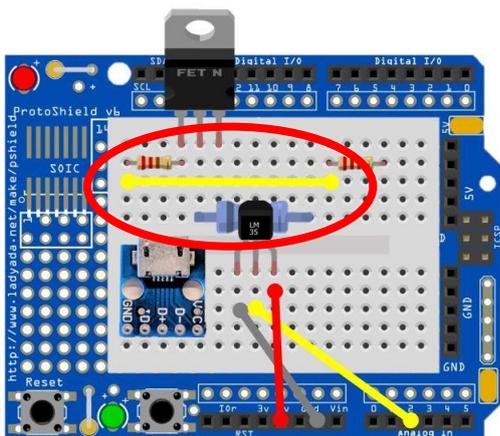


10

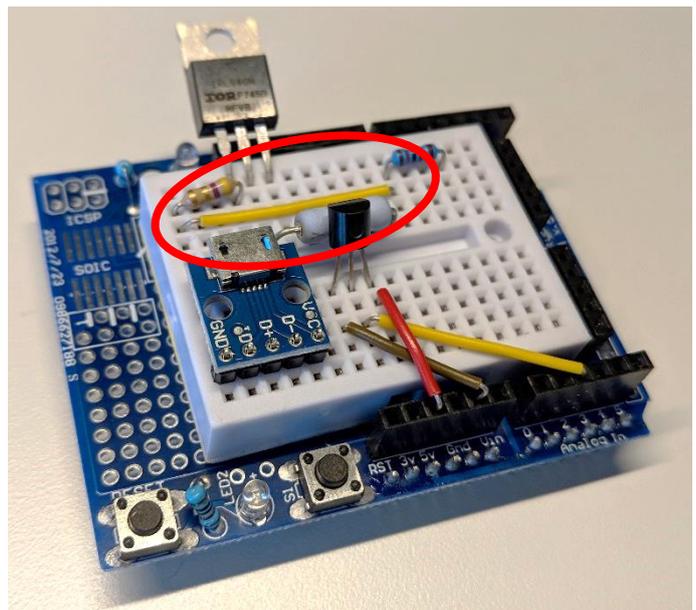


Der rechte „Fuß“ des Temperatursensors wird mit dem 5V Pin des Protoshields verbunden (vierter Pin von rechts).

# Schritt 12

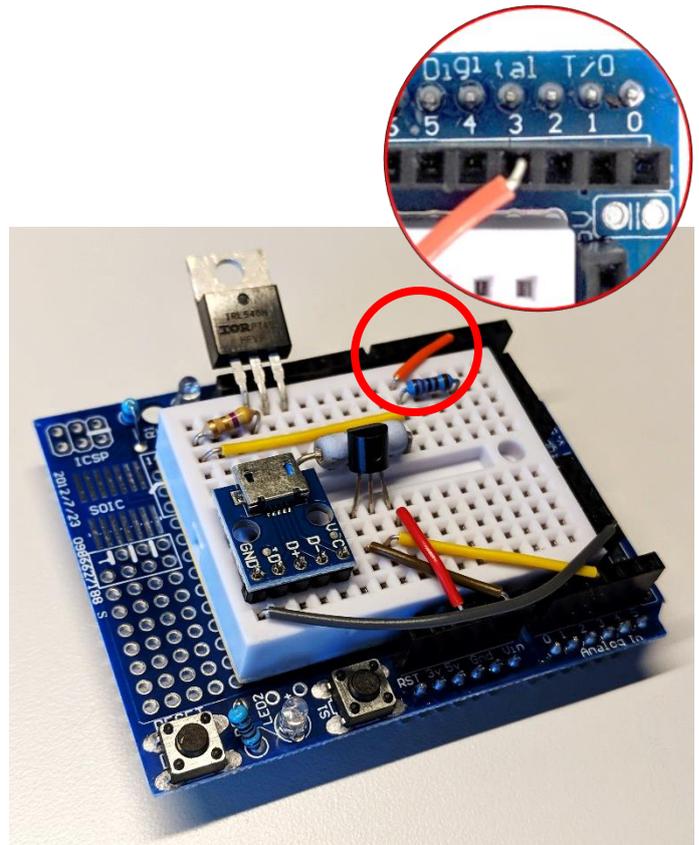
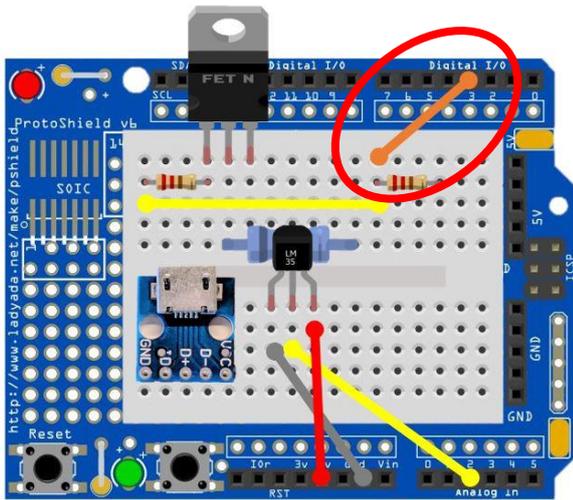


10



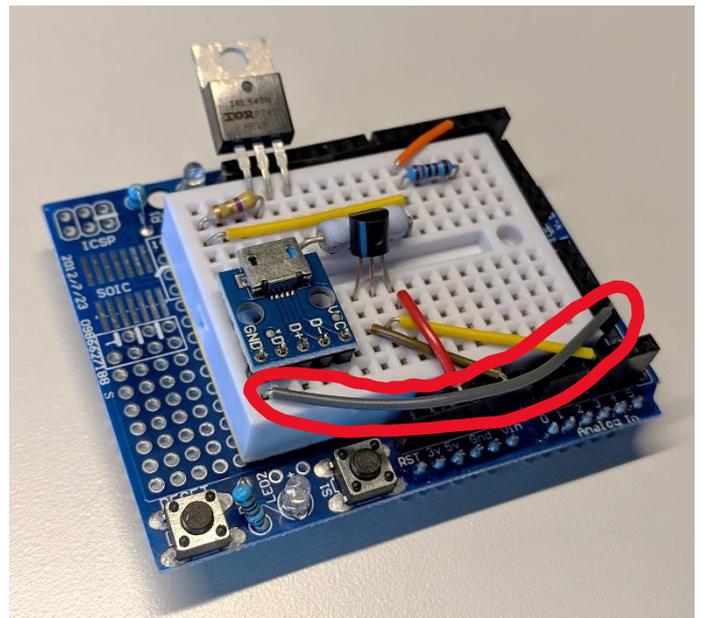
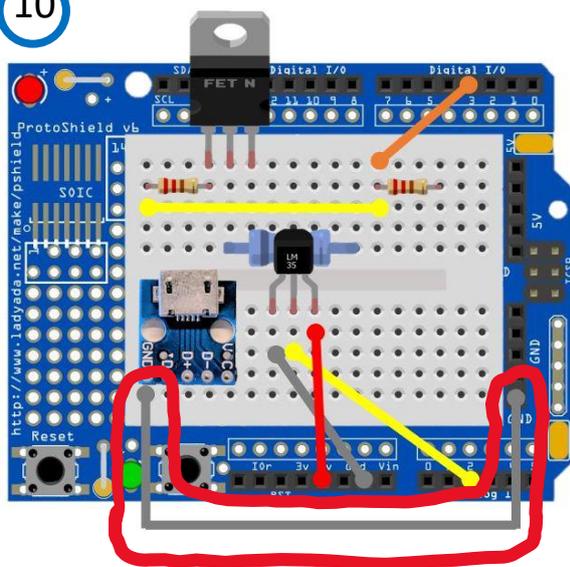
# Schritt 13

10

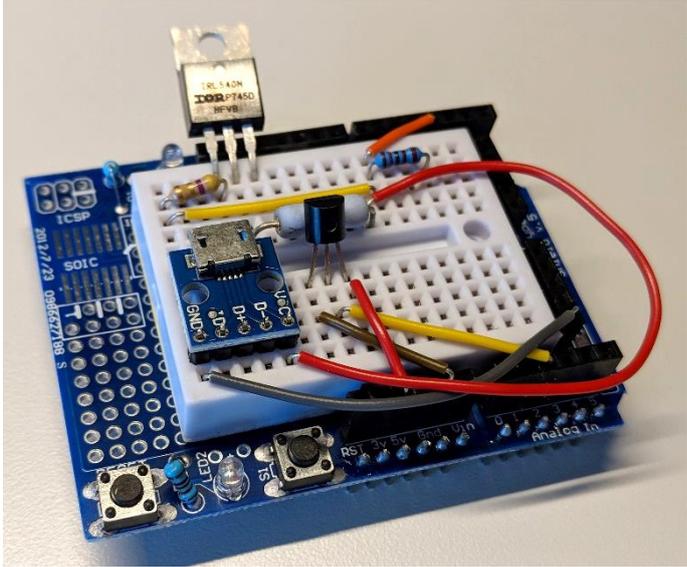
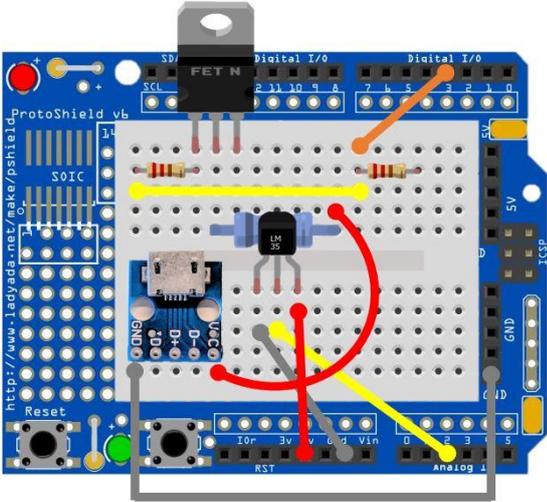


# Schritt 14

10



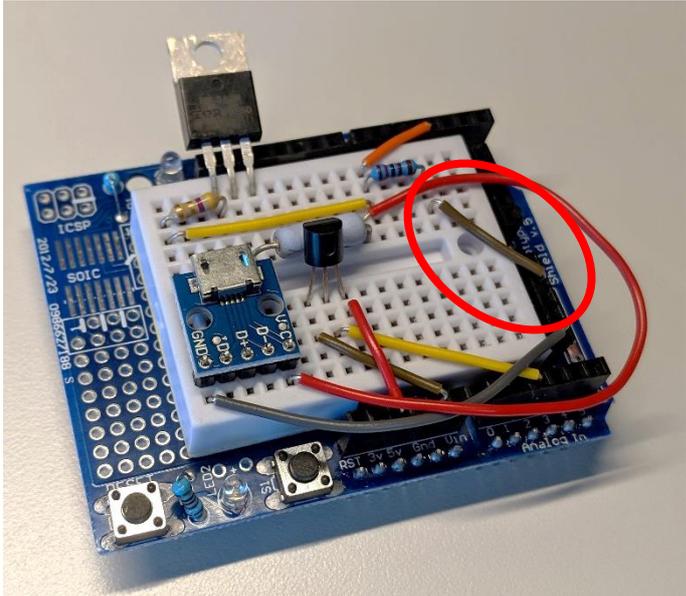
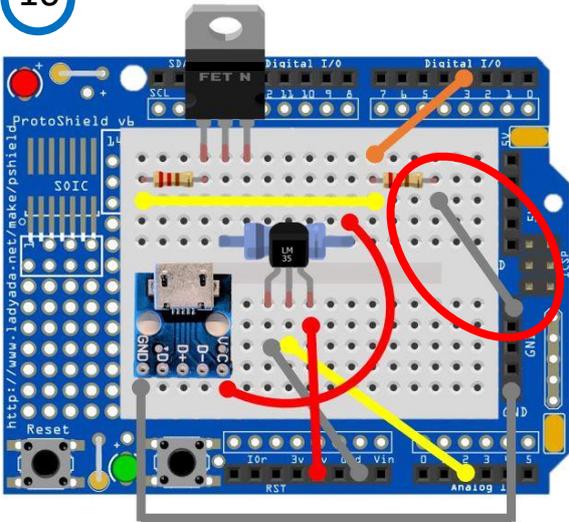
# Schritt 15



10

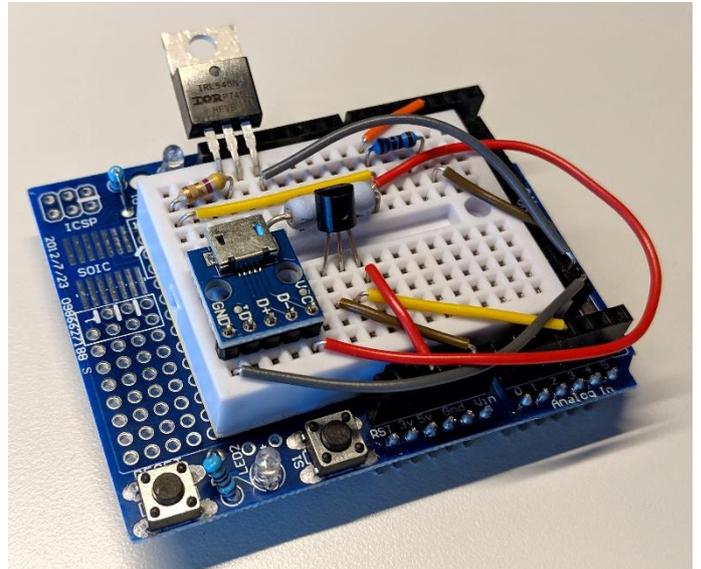
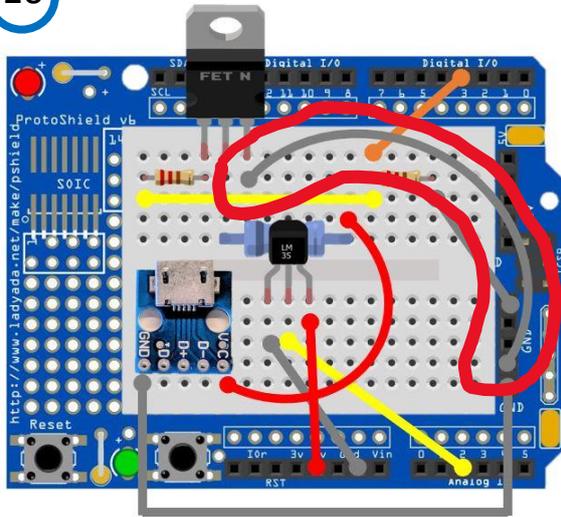
# Schritt 16

10



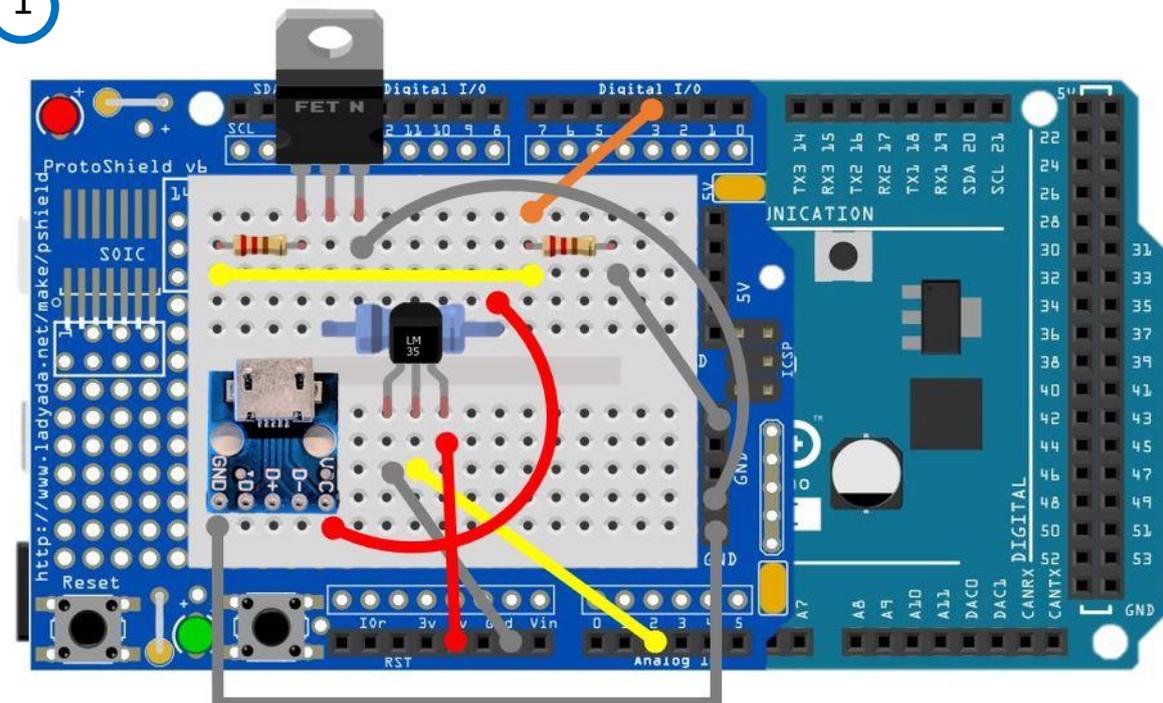
# Schritt 17

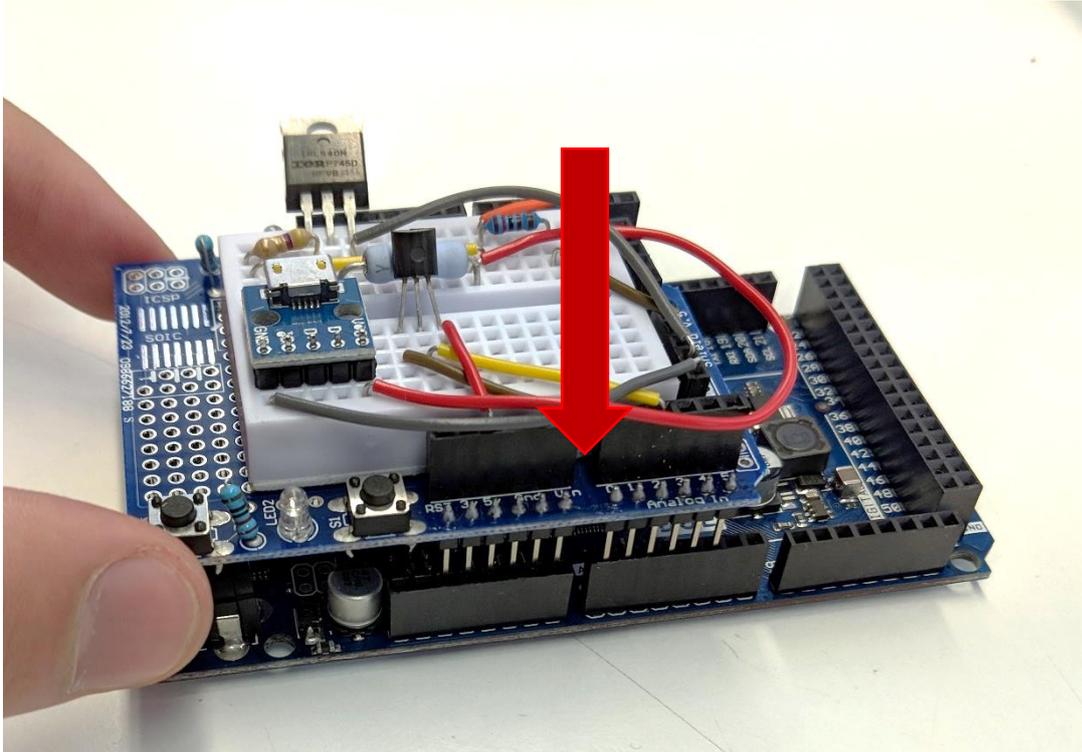
10



# Schritt 18

1

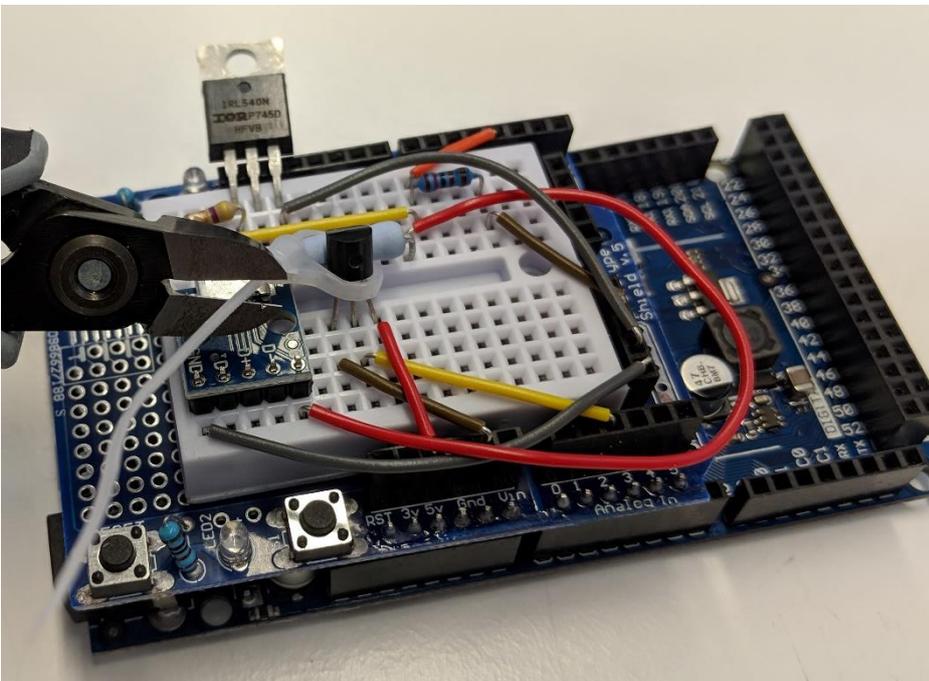
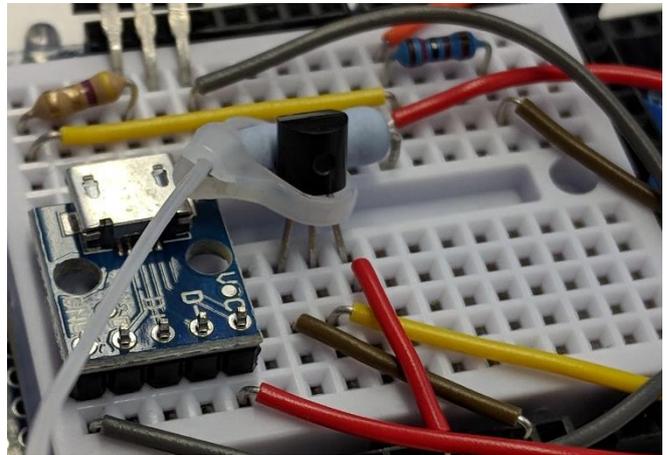
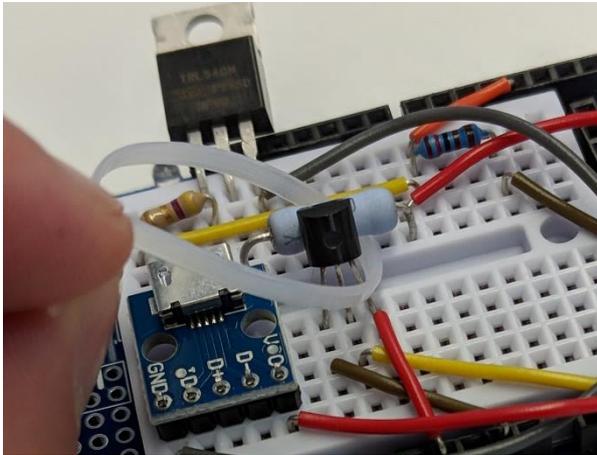




Das Protoshield so auf den Arduino Due aufsetzen, sodass die jeweiligen Pins zueinander passen und fest aufstecken.

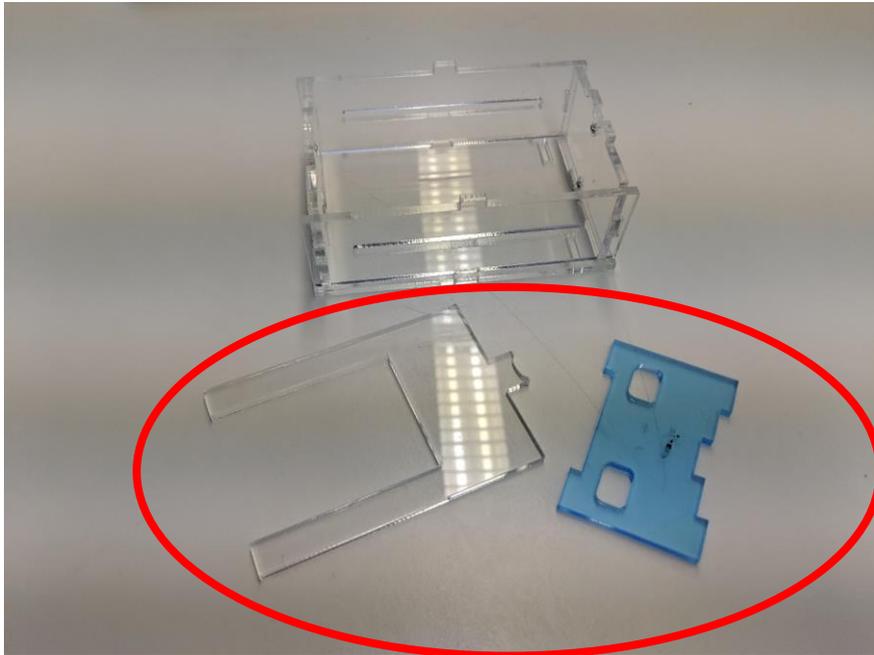
1

# Schritt 19

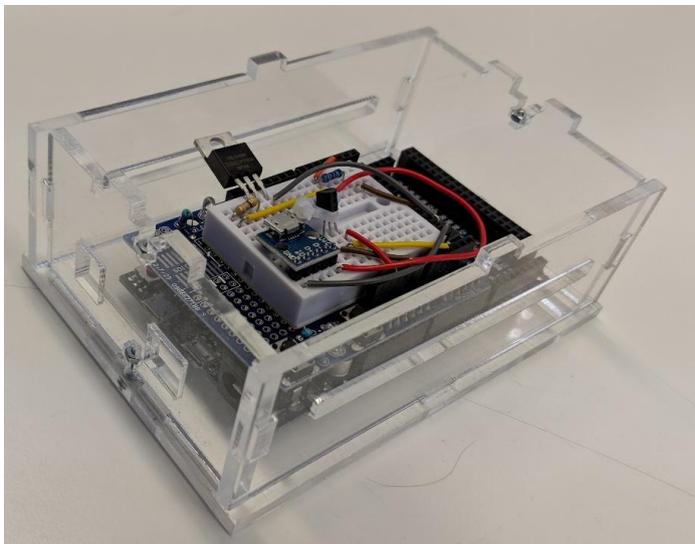


Der Temperatursensor wird nun mittels Kabelbinder an den Widerstand gedrückt. Dazu den Kabelbinder unter dem Widerstand durchführen und um den Temperatursensor festziehen. Abschließend noch den Kabelbinder mittels Seitenschneider kürzen.

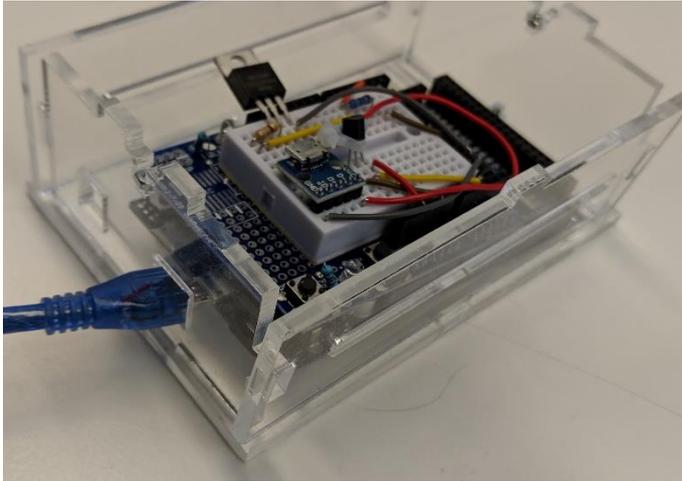
# Schritt 20



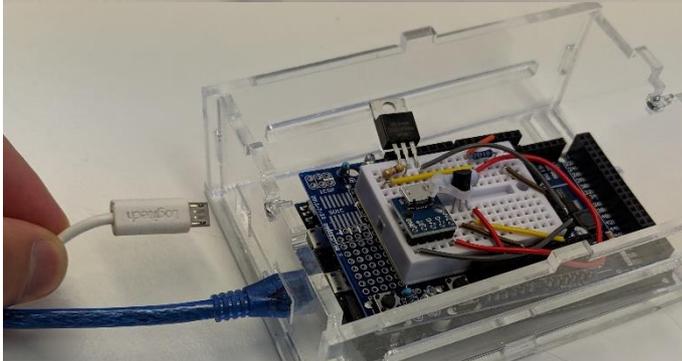
Das Gehäuse wird gemäß mitgelieferter Aufbauanleitung zusammengebaut, jedoch werden die 2 dargestellten Teile nicht verwendet.



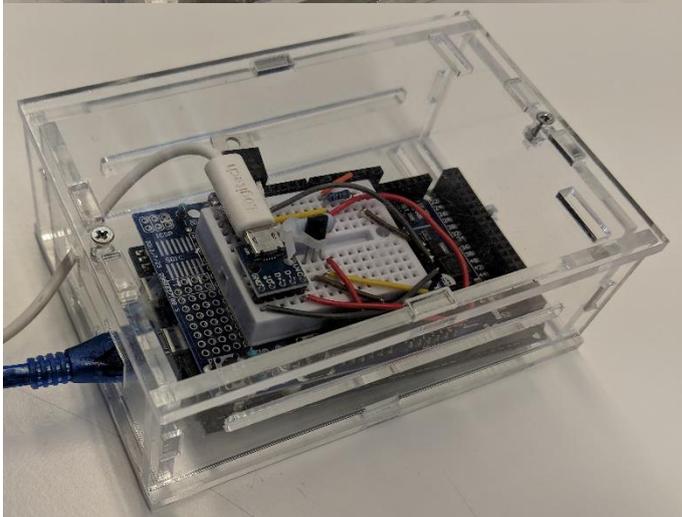
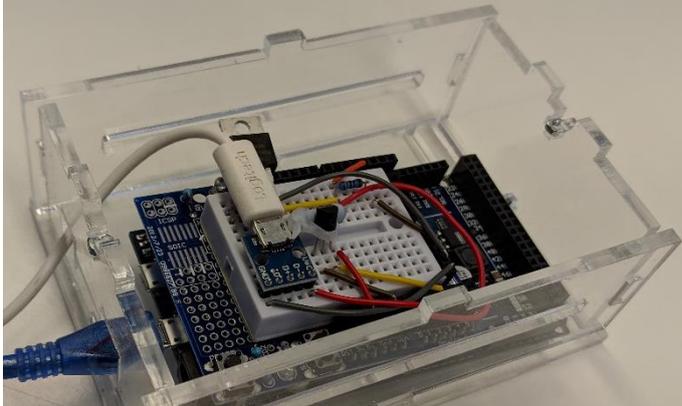
Den fertigen Hardware Aufbau in das Gehäuse legen, sodass die Arduino micro USB Anschlüsse zu den Ausschnitten des Gehäuses zeigen.



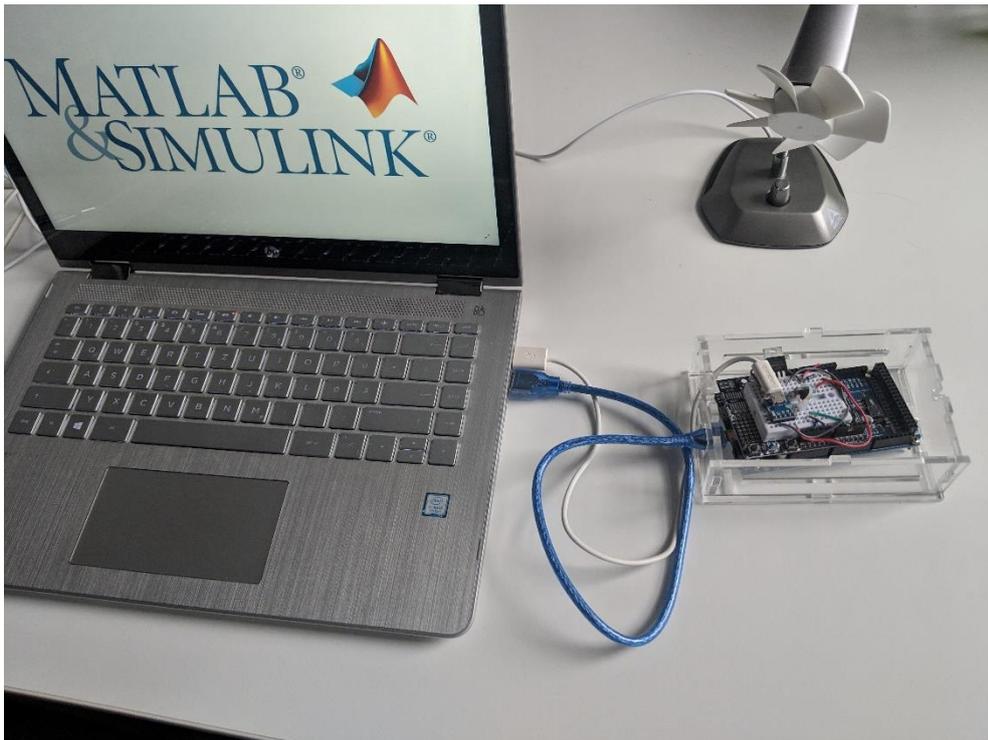
Das blaue Kabel des Arduino Due wird nun durch den Gehäuse Ausschnitt durchgeführt und mit dem unterem USB Port des Arduino Due verbunden.



Das andere Kabel wird durch den anderen Gehäuseausschnitt durchgeführt und mit dem auf dem Steckbrett befindlichen micro USB Port verbunden



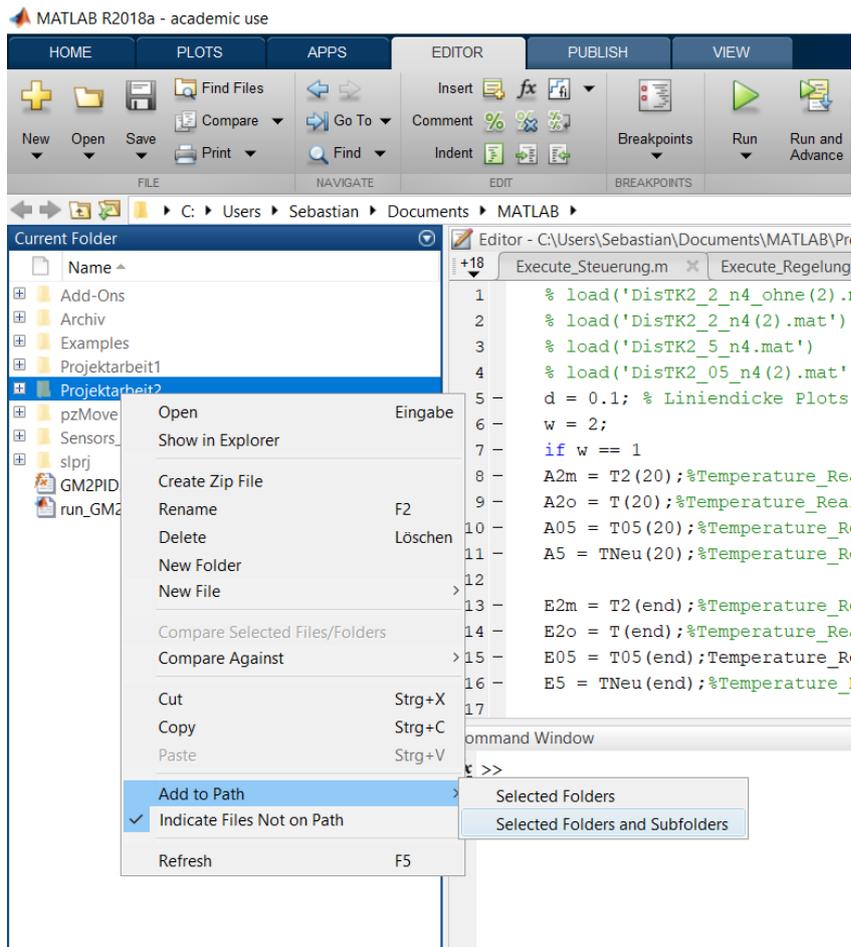
Zu guter Letzt wird nur noch der Deckel auf das Gehäuse geschraubt und fertig ist der Hardware Aufbau.



Fertiger Hardware Aufbau an den Laptop USB Ports angeschlossen.

# Softwareeinrichtung

Nachdem der Hardware Aufbau abgeschlossen ist, müssen alle nötigen MATLAB Skripte und Simulink Modelle auf dem Computer abgespeichert werden. Z.B. können alle Dateien im Matlab Verzeichnis unter einem bestimmten Ordner abgespeichert werden. Dieser Ordner muss dann in Matlab ausgewählt werden. Dazu muss der Ordner wie links im Bild mit der rechten Maustaste angeklickt werden, anschließend „Add to Path“ und „Select Folders and Subfolders“ auswählen. In meinem Beispiel heißt der Ordner in dem die Dateien abgelegt sind „Projektarbeit2“.



Nun müssen noch diverse Matlab Add-Ons installiert werden. Dazu öffnet man Matlab und wählt im Reiter „Home“ → „Environment“ die Schaltfläche Add-Ons aus. Anschließend nach dem gewünschten Package suchen und installieren.



Folgende Add-Ons müssen installiert werden:

- Simulink Support Package for Arduino Hardware
- Curve Fitting Toolbox (Version 3.5.7)

- System Identification Toolbox (Version 9.8)
- MATLAB Support Package for Arduino Hardware (Version 18.1.1)
- MATLAB Support for MinGW-w64 C/C++ Compiler (Version 18.1.0)
- Embedded Coder (Version 7.0)
- Bioinformatics Toolbox (4.10)

Außerdem muss noch ein MATLAB file ausgetauscht werden, damit der Widerstand nicht weitergeheizt wird nachdem das Simulink Modell gestoppt wird. Dazu wie folgt vorgehen:

Das File "arduino\_PWMOutput.p" muss das gleichnamige File in folgendem Ordner ersetzen:  
 C:\ProgramData\MATLAB\SupportPackages\R2018a\toolbox\target\supportpackages\arduinoobase\+codertarget\+arduinoobase\+internal<sup>1</sup>

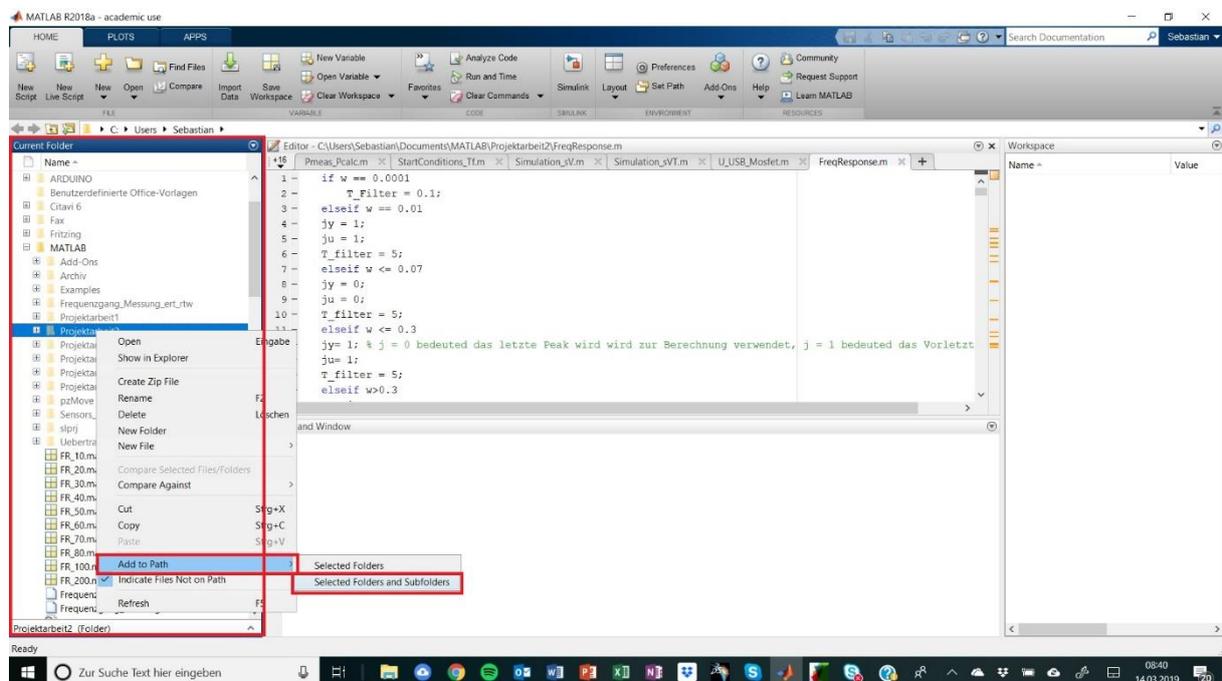
Zu Sicherheit sollte davor das bereits existierende „arduino\_PWMOutput.p“ file in „arduino\_PWMOutput\_old.p“ umbenannt werden.

Nach erfolgreichem Ersetzen des "arduino\_PWMOutput.p" File, müssen folgende Befehle im Matlab command window ausgeführt werden.

1. clear pcode
2. clear classes
3. rehash toolboxcache

Nun wird der Widerstand nach beenden des Simulink Modells nicht weiter geheizt.

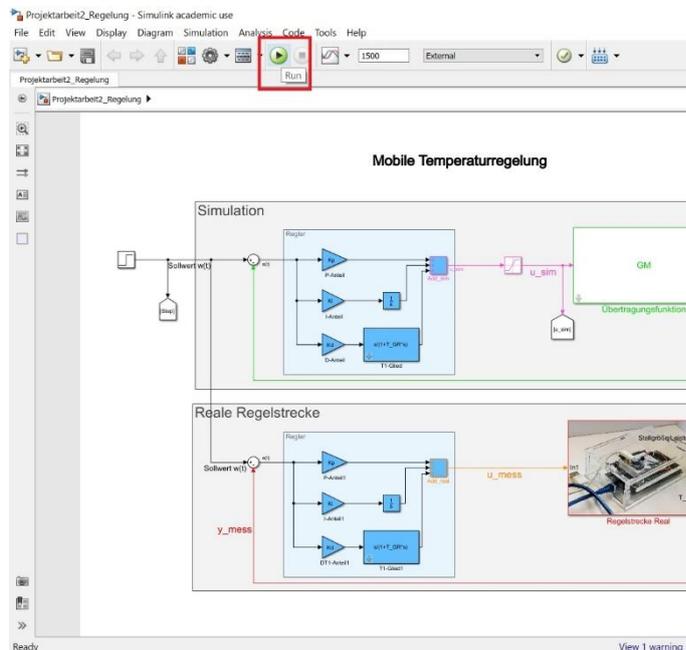
Nachdem alle Skripte auf dem Computer gespeichert, das Matlab File ausgetauscht und die nötigen Add.Ons installiert wurden, müssen der Ordner in dem die Skripte und Files auf dem PC gespeichert wurden noch zum Matlab Pfad hinzugefügt werden:



Dazu Matlab öffnen und in Current Folder Window den entsprechenden Ordner mit rechts anklicken → Add to Path → Select Folders and Subfolders.

<sup>1</sup> Hinweis des Mathworks MATLAB Support Service

Anschließend können die verschiedenen Simulink Modelle „Projektarbeit2\_Steuerung“, „Projektarbeit2\_SteuerungMitGW“, „Projektarbeit2\_Regelung“ und „Projektarbeit2\_DisturbanceCompensation“ geöffnet und ausgeführt werden. Dazu einfach den Hardware Aufbau an dem Computer über die USB Ports verbinden, das jeweilige SIMULINK Modell öffnen und auf run klicken (siehe folgende Abbildung).



Bei erstmaliger Ausführung eines Modells kann es ein wenig dauern, bis der Code fertig kompiliert wurde (ca. 1 Minute).

Zur Überprüfung ob der Hardware Aufbau sowie die Softwareeinrichtung erfolgreich sind, soll nun das Simulink Modell „Projektarbeit2\_Steuerung.slx“ geöffnet werden und durch Betätigung des „Run“ Button gestartet werden. Das Programm endet entweder automatisch nach 2000 Sekunden oder durch manuelle Betätigung des „Stop“ Buttons (rechts neben „Run“). Sobald das Programm beendet wird sollte ein Plot in der Form wie der unten dargestellte erscheinen:

