

OpenMoverPlatform

Projektstatus-Dokumentation

Marcel Kitzbichler

22. Januar 2026

1 Einleitung

Die OpenMoverPlatform ist ein mobiler Roboter, der selbstständig fahren oder ferngesteuert werden kann. Dieses Dokument beschreibt den aktuellen Stand des Projekts. Es ist eine Zusammenfassung der Mechanik, der Elektronik und der Software. Alle Designs und der Quellcode können im Projekt-Repository gefunden werden: <https://github.com/Marcel-Kitzbichler/OpenMoverPlatform>

2 Aufbau des Systems

2.1 Mechanik

Das Gehäuse und die mechanischen Teile wurden mit SolidWorks konstruiert. Das Design ist modular aufgebaut. Der Modulanschluss befindet sich oben auf der Plattform, dieser Anschluss ist sowohl für die Mechanische Befestigung als auch für die elektrische Verbindung des Moduls mit der Plattform zuständig.

2.1.1 Der Antrieb

Der Antrieb wurde bisher in zwei Versionen entwickelt:

- **Version 1:** Ein erster Prototyp, um testen zu können ob ein komplett auf dem 3D-Drucker gefertigtes Getriebe überhaupt machbar ist.
- **Version 2 (Aktuell):** Diese Version ist fortgeschrittener. Sie besitzt ein Getriebe mit mehreren Gängen und einen Kettenantrieb (**Panzerketten**). Damit soll der Roboter auch in schwierigem Gelände gut fahren können.



Abbildung 1: CAD-Modell des Antriebs V2

2.2 Elektronik

Das Herzstück der Elektronik ist eine eigene Platine (PCB) die in der Schule gefertigt wurde.

- **Steuerung:** Verwendet wird ein ESP32-S3 Mikrocontroller. Er ist leistungsstark und hat bereits ein USB und ein Bluetooth Interface integriert.
- **Sensoren:**
 - **Kompass:** Ein HMC5883L Sensor, damit der Roboter weiß, in welche Richtung er schaut.
 - **GPS:** Damit der Roboter seine Position bestimmen kann.
 - **Stromversorgung:** Die Batteriespannung wird überwacht.



Abbildung 2: Design der Roboter-Platine

3 Software

Die Software besteht aus zwei Teilen: Dem Programm auf dem Roboter (Firmware) und dem Programm auf dem Computer zur Steuerung.

3.1 Firmware (Auf dem Roboter)

Die Firmware steuert die Hardware direkt. Wichtige Funktionen sind:

- `main.cpp`: Startet das System und lädt Einstellungen.
- `serialManager`: K kümmert sich um die Kommunikation mit dem Computer.
- `wpManager`: Speichert die Wegpunkte, die der Roboter abfahren soll.
- `compass`: Liest die Kompass-Daten aus.

3.2 PlatformHelper

Um die Platform zu steuern und Daten auszutauschen gibt es sowohl eine Handy App als auch ein PC Programm, welches im Projekt Repo gefunden werden kann. Diese Anwendungen verbinden sich über die unten beschriebenen Schnittstellen mit der Steuerung.

4 Kommunikation

Die Platform und jegliche Peripherie unterhalten sich über ein einfaches Textprotokoll (JSON). Es gibt verschiedene Befehle, die in der Zukunft noch erweitert und dann vollständig dokumentiert werden wird. Dieses JSON Interface kann auf gleiche Art und Weise über USB , eine Serielle Schnittstelle und über Bluetooth angesprochen werden. Aktuell sind diese Befehle implementiert:

- **Intent 1**: Gespeicherte Wegpunkte abrufen.
- **Intent 2**: Automatische Fahrt starten.
- **Intent 4**: Motoren direkt steuern.
- **Intent 5**: Neue Wegpunkte hochladen.
- **Intent 6**: Status-Informationen abfragen.

5 Aktueller Stand und Zukunft

Was schon fertig ist:

- Das Design für den neuen Kettenantrieb (V2).
- Der Entwurf der Platine.
- Die grundlegende Software zum Fahren und Steuern.
- Das PC-Programm zur Bedienung.

Was noch fehlt:

- Der finale Zusammenbau des neuen Antriebs.
- Tests im Freien, um zu sehen, wie gut der Roboter alleine fährt.
- Eine vollständige Dokumentation der Verkabelung.