

Inhaltsverzeichnis

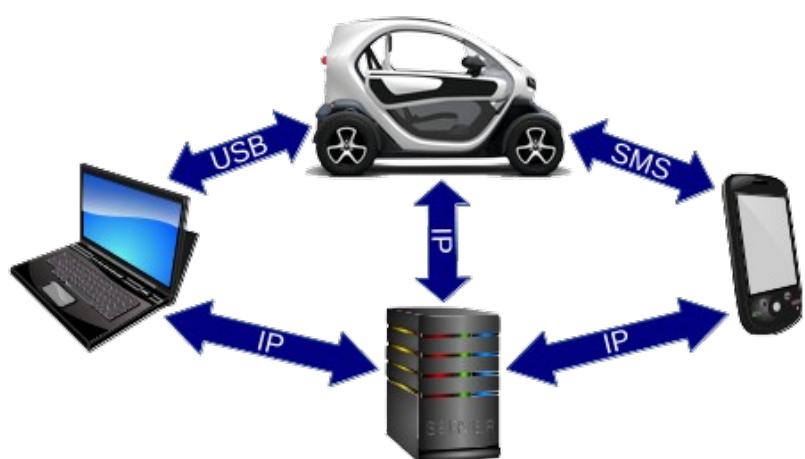
Was ist das OVMS?	1
Was kann das OVMS im Twizy?	2
Ist das erlaubt?	3
Ist das gefährlich?	3
Wie steuert man das OVMS?	4
Kann man damit auch den Ladeprozess fernsteuern?	4
Was kostet es? Woher bekommt man es?	5
Wieviel SMS- und Datenvolumen braucht man?	6
Wo ist das User-Manual?	6
Wie baut man es ein?	7
Muss es angeschlossen bleiben?	7
Kann man es gleichzeitig mit einem Twizplay nutzen?	7
Wie führt man die Grundkonfiguration durch?	7
Wie komme ich an mein erstes Tuning?	10
Kann man ohne Handy Tuning-Profile umschalten?	11
Warum so viele Logins und Passwörter?	11
Wieso muss man die Firmware aktualisieren?	12
Wie aktualisiert man die Firmware?	12
Wie installiert man die aktuellste Android-App?	12
Typische Probleme und deren LED-Blinkcodes	13
Wozu brauche ich CFG PRE und CFG OP?	13
Wie kann ich mich bedanken und das Projekt unterstützen?	15

Was ist das OVMS?

OVMS = Open Vehicles Monitoring System

Das OVMS ist ein kleines Modul, das an den Diagnose-Port angeschlossen wird (OBD2/CAN-Bus), und darüber permanent die Fahrzeugdaten überwacht. Es kann auch auf den CAN-Bus schreiben und auf diesem Weg die Motorsteuerung umkonfigurieren ("Chip-Tuning").

Das OVMS kann (muss aber nicht) mit einer handelsüblichen SIM-Karte mit dem Handynetzt verbunden werden. Es kann dann per SMS, über eine eigene Smartphone-App, sowie programmgesteuert vom PC aus angesprochen werden. Das OVMS kann dabei Messdaten eigenständig



an einen Server im Netz senden, so dass auch längerfristige Aufzeichnungen ohne ständige Verbindung mit dem Modul möglich sind.

Das Modul kann bspw. einen SMS-Alarm senden, wenn der Ladevorgang unterbrochen wird oder die 12V-Batterie aufgeladen werden muss. Außerdem enthält es einen GPS-Empfänger, damit kann man das Fahrzeug lokalisieren und die Fahrtstrecken aufzeichnen.

Alles zum OVMS findet man auf der offiziellen Website: www.openvehicles.com/

Schaltpläne und Software des OVMS sind komplett **Open Source** und frei verwendbar. Jeder kann daran mitarbeiten, jede Hilfe wird gern angenommen. Das OVMS ist ursprünglich ein Projekt für den Tesla Roadster gewesen, enthält inzwischen aber Support für eine Vielzahl von Fahrzeugen.

Was kann das OVMS im Twizy?



Lade-Überwachung: Akkustand, SOC-Nutzungsfenster, Aufzeichnung von Lade- und Entladekurven, Alarm bei Ladeabbruch, Meldung bei Ladeende und optional bei ausreichender Ladung für die nächste Fahrt, Berechnung der vorr. Ladezeiten, Überwachung des 12V-Akkus

Reichweite: Auslesen der Twizy-Schätzung, Berechnung der individuellen "idealen" Reichweite (einstellbar), Hochrechnung beider Schätzungen während des Ladens

Batterie-Monitoring: Überwachung und Aufzeichnung aller einzelnen Zellenspannungen und Temperaturen, Anzeige von Auffälligkeiten, Alarm bei zu hoher Einzelabweichung und bei zu hoher Gesamtdrift (Standardabweichung), Ermittlung der Akku-Kapazität (SOH)

Power-Monitoring: Aufzeichnung des Energieverbrauchs gesamt sowie separat nach Fahrmodus (konstant/beschleunigend/bremsend) und Höhenänderung (auf/ab, jeweils mit Summe der Höhenmeter), Effizienz (Wh/km) und Rekuperationsraten jeweils, automatischer Report nach Fahrtende, Aufzeichnung der Energiesumme beim Laden

GPS-Tracking: laufende Abfrage der aktuellen Position, Aktualisierung auch wenn Twizy aus ist

(Diebstahl-Verfolgung), Track-Logging auf den Server wahlweise 1x pro Minute / alle 5 Sekunden (Streaming-Modus), Aufzeichnung und Auswertung des Strecken-Energieverbrauchs

Tuning (SEVCON-Konfiguration): Einstellen der Höchstgeschwindigkeit und Warngrenze, Erhöhung des maximalen Motorstroms, Drehmoments und/oder der maximalen Motorleistung, Veränderung der verfügbaren Leistung beim Fahren sowie Rekuperation im Neutral- und Bremsmodus (Maximum + je nach Geschwindigkeit), Veränderung der Drehmoment-Rampen, Veränderung der Drehmoment-Glättung, Aktivierung der Bremsleuchte für neutrale Rekuperation (benötigt Hardware-Modifikation), Realisierung eines Kickdowns, dynamische Akku/BMS-Leistungsnachführung, Low-Level-Zugriff auf alle SEVCON-Register

Ist das erlaubt?

NEIN.

Das OVMS ist ein reines Hobbyprojekt. Jegliche Anwendung des OVMS erfolgt komplett auf eigene Verantwortung und eigenes Risiko.

Das OVMS hat keine offizielle Zulassung und keine allgemeine Betriebserlaubnis (ABE). Die Nutzung ist daher rechtlich gesehen allenfalls auf nicht-öffentlichen Straßen bzw. außerhalb des öffentlichen Straßenverkehrs zulässig. Der Versicherungsschutz kann verfallen. Die Garantie/Gewährleistung seitens Renault kann bereits durch Einbau des Moduls ebenfalls verfallen.

Tuning-Maßnahmen zur Leistungs- und Geschwindigkeitserhöhung können i.d.R. auch nach dem Rückbau dauerhaft nachvollzogen werden. Die Strom- bzw. Leistungsgrenzen des Akkus werden durch das Modul nicht außer Kraft gesetzt, der Akku wird also nicht stärker belastet als technisch vorgesehen. Falls Renault aber bei 75% Kapazität einen Ersatzakku wegen Tuning verweigern, so müsste der Nachweis dafür ggf. auf eigene Kosten und eigenes Risiko geführt werden.

Ist das gefährlich?

Das kommt darauf an, wie man es einsetzt. Bei reiner Fernüberwachung ist eine Gefährdung von Twizy oder Fahrer so gut wie ausgeschlossen, solange man sich die Antennenkabel nicht gerade um Hals oder Lenkrad wickelt.



Mit erhöhter Leistung, Geschwindigkeit und Rekuperation geht erhöhte

Verantwortung einher. Insbesondere im Winter ist bereits die geringe Standard-

Rekuperation problematisch für unerfahrene/assistentzverwöhnte Fahrer, da diese nur die Hinterachse bremst. **Bei erhöhter Rekuperation wird der Twizy auf glatter Fahrbahn schnell auch für gute Fahrer unkontrollierbar.** Allerdings gilt umgekehrt auch, dass er bei **reduzierter Rekuperation** im Winter deutlich besser beherrschbar bleibt als in der Standardkonfiguration.



Auch mit erhöhter Beschleunigung bleibt der Twizy leise und niedlich. D.h. man muss um so mehr auf **Fußgänger und andere Verkehrsteilnehmer** achten, die dem Twizy eine solche Agilität i.d.R. nicht zutrauen.

Für den Twizy bzw. die Komponenten ergibt sich aus stärkerer Beanspruchung selbstverständlich **ein höherer Verschleiß**. Schon ab 77 km/h wird das Twizy-Getriebe außerhalb seiner Spezifikation betrieben, bei dauerhaft hoher Geschwindigkeit kann sich das Getriebeöl überhitzen und die Simmerringe schädigen.

Der Twizy-Controller schützt sich selbst, den Akku und den Motor vor Überlastung und Übertemperatur. Aber höhere Leistung geht immer mit mehr Erwärmung und entsprechend mechanischer Ausdehnung einher, d.h. die Komponenten altern schneller.

Umgekehrt altert der Twizy natürlich langsamer, wenn man die Leistung reduziert.

Wie steuert man das OVMS?

Der einfache Weg: man sendet dem OVMS Textkommandos per SMS und erhält die Antwort ebenfalls per SMS. Dies funktioniert mit jedem handelsüblichen Handy.

Der Smartphone-Weg: für iOS und Android existieren OVMS-Apps in den jeweiligen App-Stores, Links siehe OVMS-Website bzw. aktuellste Android-Version unter dexters-web.de/downloads. Die App ist sehr komfortabel, deckt aber derzeit noch nicht alle Spezialfunktionen des Twizy ab. Nutzt man die App, so kann man auf SMS i.d.R. verzichten, da die Alarmer und Meldungen auch über die App empfangen werden können (solange OVMS und Handy Internetverbindung haben).

Der Server-Weg: die vom OVMS über den Server aufgezeichneten Daten kann man als CSV-Dateien abrufen und dann bspw. mit OpenOffice Calc / MS Excel o.ä. verarbeiten und analysieren. Die GPS-Tracklogs kann man sehr einfach in GPX-Dateien umwandeln und mit Google Maps oder OSM visualisieren. Ein Web-Interface zu diesen Daten existiert ebenfalls.

Der Offline-Weg: das OVMS kann auch mit einem seriellen Kabel direkt angesteuert werden. Alle SMS- und Smartphone-Befehle können so bspw. per Terminalprogramm aufgerufen werden, auch ohne Netzverbindung und ohne SIM-Karte. Eine Anleitung dazu sowie eine Benutzeroberfläche für MS-Windows findet man auf dexters-web.de/links.

Kann man damit auch den Ladeprozess fernsteuern?

Ja, das geht mit einem Heim-Automatisierungssystem oder auch schon einer einfachen fernsteuerbaren Steckdose und einem PC, Raspberry Pi o.ä..

Möchte man bspw. die Ladung starten wenn genügend Leistung auf der Solaranlage anliegt, dann

fragt der Home-Automation-Server, PC oder RasPi die Solarleistung regelmäßig ab und schaltet entsprechend die Steckdose für den Twizy ein.

Soll die Ladung dann bspw. bei 80% automatisch stoppen, dann fragt das System dazu über einen der Kommandozeilen-Clients regelmäßig den Ladestand des Twizy am OVMS-Server ab und schaltet die Steckdose aus, sobald die 80% erreicht sind.

Fertige Automatisierungs-Skripte findet man im Projektbereich auf Github bislang für [Edimax-Steckdosen](#) und das [Homematic-System](#).

Auch für andere Systeme sind die Skripte leicht anzupassen: [Anwenderbericht und Script für das Conrad/ELV FS20 Funksteckdosensystem](#)

Natürlich sind auch weitere Spielereien möglich, bspw. per Abfrage der GPS-Koordinaten automatisch die Garage öffnen wenn der Twizy auf den Hof fährt...

Was kostet es? Woher bekommt man es?

Die Gesamtkosten (ohne SIM-Karte) für ein komplettes Kit mit Flash-Programmierer liegen bei etwa 160 Euro zzgl. Zollgebühren (wenn man Glück hat geht's auch mal am Zoll vorbei).

Standard-Paket für SIM-Nutzung:

- [OVMS Car Module v2](#)
- [OBD-II to DB9 Data Cable for OVMS \(left\)](#)
- [OVMS GSM Antenna](#)
- [Universal GPS Antenna \(SMA Connector\)](#)
- SIM-Karte

Paket für Nutzung ohne SIM-Karte:

- [OVMS Car Module v2](#)
- [OBD-II to DB9 Data Cable for OVMS \(left\)](#)
- [RS232 Serial to USB Cable](#)

Für Firmware-Updates:

- [PICKIT 2 Compatible Programmer/Debugger](#)
- oder [PICKIT 3.5 Compatible Programmer/Debugger](#)

Weitere Infos zum Bezug des OVMS sowie Tipps zur Auswahl einer passenden SIM-Karte findet man im User-Manual.

Wieviel SMS- und Datenvolumen braucht man?

Bis die Datenverbindung steht (und falls sie mal nicht verfügbar ist oder man sie nicht nutzen möchte) benötigt man SMS. Für die Grundeinrichtung sind i.d.R. nur eine handvoll SMS-Nachrichten nötig.

Im laufenden Betrieb sendet das OVMS Alarme und Benachrichtigungen wie den Ladestatus und Tripreport wahlweise per SMS und/oder IP. Nutzt man die App, dann erhält man diese im IP-Modus als Systembenachrichtigungen und kann sie im Tab "Nachrichten" aufrufen -- SMS sind hierfür also überflüssig.

Die automatischen Benachrichtigungen kann man wahlweise auch jederzeit auf die Alarme einschränken oder ganz deaktivieren. Die Steuerung der Kanäle und Filter erfolgt über Parameter #3 und Feature #14.

Mit der iOS-App oder einer älteren Firmware benötigt man SMS noch für die Twizy-Spezialfunktionen wie den Batteriemonitor und die SEVCON-Konfiguration. Mit der aktuellen Firmware und Android-App werden SMS auch dafür nicht mehr benötigt, da alle SMS-Befehle damit auch über den IP-Kanal genutzt werden können.

Das IP-Datenvolumen hängt natürlich von der Nutzung ab, je mehr man den Twizy nutzt, desto mehr Datenvolumen fällt an. Bei durchschnittlicher Nutzung genügen i.d.R. 10 MB pro Monat inkl. Nutzung der normalen GPS-Tracking-Auflösung. Möchte man öfter den hochauflösenden GPS-Stream verwenden so kann ein höheres Volumen anfallen.

Wo ist das User-Manual?

Hier sind alle Dokumente in der jeweils aktuellsten Version:

github.com/openvehicles/.../Renault-Twizy

Direkt-Link zum User-Manual (PDF):

github.com/openvehicles/.../OVMS-UserGuide-RenaultTwizy.pdf

Allgemeine Infos findet man derzeit noch im Tesla-Roadster-Manual:

github.com/openvehicles/.../OVMS_UserGuide_TeslaRoadster.pdf

Wie baut man es ein?

Im einfachsten Fall packt man das Modul inkl. der beiden Antennen einfach in das linke Handschuhfach, wo auch der Diagnose-Port ist.

Schöner geht natürlich auch: [Einbau unter dem Handschuhfach \(Wido Vogel\)](#)

Muss es angeschlossen bleiben?

Nein, das Modul muss nicht angeschlossen bleiben, wenn man lediglich Tuning und Diagnose damit durchführen möchte, und man kann ein Modul verwenden um beliebig viele Twizy zu tunen.

Der Nachteil ist dabei, dass man dann nur ein fest eingestelltes Tuningprofil verwenden kann. Die dynamischen Anpassungsmöglichkeiten umfassen den Kickdown und insbesondere die Leistungsnachführung des Tunings. Letztere braucht man, da der Akku je nach Ladestand und Temperatur unterschiedlich viel Leistung freigibt. Braucht das Tuning mehr Strom als gerade verfügbar, dann entstehen abrupte Leistungseinbrüche beim Beschleunigen und Bremsen, und der Akku wird nicht optimal ausgenutzt.

Auch kann man dann nicht unterwegs die Leistung und Rekuperation an die aktuelle Witterung anpassen, bspw. um im Winter auf Blitzeis zu reagieren.

Kann man es gleichzeitig mit einem Twizplay nutzen?

Ja, das funktioniert mit jedem handelsüblichen OBD2 Y-Kabel (Verteiler/Splitter), das auch den CAN-Bus auf beide Anschlüsse verteilt. Alternativ kann man sich ein Y-Kabel selbst löten oder von einem halbwegs erfahrenen Bastler löten lassen.

Wie führt man die Grundkonfiguration durch?

Tipp: die Installation und Grundeinrichtung wird optimal durch den Workshop von Peter Saß erklärt: [WORKSHOP.pdf](#)

- **Vorbereitung 1:** SIM-Karte besorgen, freischalten und PIN-Abfrage deaktivieren (in einem normalen Handy), Guthaben aufladen.
- **Vorbereitung 2:** Auf [dexters-web.de](#) oder [www.openvehicles.com](#) Benutzerkonto einrichten und Fahrzeug darin registrieren.
- **Vorbereitung 3:** Modul-Firmware aktualisieren (s.u.).

Beim Einsetzen der SIM-Karte in das Modul prüfen ob der Schiebeschalter in der Position "nach innen" steht (= Datenverbindung an).

Modul anschliessen. **Warten bis nur die grüne LED noch 2x blinkt.** Hinweise: oft braucht eine SIM-Karte beim ersten Einsatz etwas länger für das Einbuchen in das Handy-Netz, manchmal mehrere Stunden. Auch kann ggf. der Empfang am Standort des Fahrzeugs schlecht sein. Die LED-Blinkcodes zeigen solche Probleme an, die Erläuterung steht [hier](#) und im Tesla-Manual auf Seite 14.

Hinweise zu den SMS-Kommandos: **Groß-/Kleinschreibung** ist bei den Befehlen egal (`PASS` = `Pass` = `pass`), aber bei den Parameterwerten (bspw. Passwort) i.d.R. wichtig. Wenn eine Eingabe nicht funktioniert also zunächst die Groß-/Kleinschreibung überprüfen.

Die **Anführungszeichen "..."** stehen im Text ggf. für Anfang und Ende eines Befehls, diese werden nicht mit eingegeben. **Klammern <...> oder [...]** stehen für benutzerdefinierte Eingaben (statt fester Werte) und werden ebenfalls nicht mit eingegeben.

Leerzeichen müssen bei allen Textbefehlen genau so eingegeben werden wie gezeigt, es dürfen keine zugefügt oder weggelassen werden. Insbesondere dürfen bspw. bei Abfragekommandos wie `MODULE?` kein Leerzeichen vor dem "?" stehen, und zwischen zwei Parametern muss immer genau ein Leerzeichen stehen.

Nach jedem Befehl erst die **Bestätigung abwarten**, bevor man den nächsten sendet. Hat das Modul den Befehl empfangen und verarbeitet, dann schickt es eine Bestätigungs-SMS. Diese kommt i.d.R. innerhalb von maximal 3 Minuten und bleibt nur aus, wenn man den Befehl falsch geschrieben hat, oder die SMS verloren gegangen ist -- dann die Schreibweise prüfen und den Befehl neu senden.

Hinweis zur Fernkonfiguration / "SMS Provisioning": diese Komfortfunktion der OpenVehicles-Website (der grüne Button beim Fahrzeugeintrag) stellt fast alle Konfigurationsparameter (Schritt 1-6) über eine Sammel-SMS ein. Die Funktion arbeitet leider aktuell unvollständig: vor oder nach der Anwendung muss der Fahrzeugtyp "RT" von Hand eingestellt werden (per SMS-Befehl `MODULE ...` oder Parameter #14). Bitte bei Verwendung der Funktion eine Spende an das Projekt schicken um die SMS-Kosten auszugleichen.

- **Schritt 1:** Als SMS `REGISTER OVMS` vom Handy an das OVMS senden. "OVMS" ist das Standard-Modulpasswort, möchte man später ein anderes Handy registrieren, dann muss man dies durch das aktuelle Modulpasswort ersetzen (siehe nächster Schritt).
- **Schritt 2:** Per SMS `PASS <passwort>` das gewünschte neue Modulpasswort setzen (das ersetzt das Standard-Passwort "OVMS"). Das Modulpasswort ist das SMS-Passwort für nicht registrierte Handys. **Beispiel:** `PASS sesam`
- **Schritt 3:** SMS `MODULE <fahrzeugkennung> K SMSIP RT` -- die Fahrzeugkennung ist der auf dem verwendeten OVMS-Server eingerichtete **Fahrzeug-Login**, dieser besteht aus Großbuchstaben und Ziffern. "K" steht für "Kilometer", "SMSIP" schaltet Benachrichtigungen per SMS und App ein, und "RT" = "Renault Twizy". **Beispiel:** `MODULE TWIZY007 K SMSIP`

RT

- **Schritt 4:** Prüfen ob soweit alles klappt: Twizy einschalten. Per SMS **STAT?** sollte man nun den Ladestand abfragen können, und per **GPS?** sollte man die Koordinaten abrufen können.

Nun ist das OVMS komplett per SMS nutzbar. Möchte man ausserdem Server und App nutzen, dann geht es weiter mit...

- **Schritt 5:** Datenverbindung einrichten per **GPRS <apn> <username> <passwort>** mit den APN-Daten des SIM-Providers (findet man per Google oder in den Infoseiten des Providers). Für leeren Usernamen/Passwort "-" verwenden. **Beispiel für blau.de:** **GPRS internet.eplus.de blau blau** -- andere einfach per Google-Suche nach Provider + "APN" ermitteln.
- **Schritt 6:** Serververbindung einrichten bspw. per **SERVER ovms.dexters-web.de <fahrzeugpasswort> -**. Das Passwort ist hier das zum beim MODULE-Befehl verwendeten **Fahrzeug-Login**, im Beispiel also das für den Login "TWIZY007". Sobald die grüne LED nur noch 1x blinkt ist die Serververbindung aktiv.
- **Schritt 7:** App einrichten: im Tab "Einstellungen" das Fahrzeug zufügen. Fahrzeug-Login und -Passwort eintragen. Die Beschreibung ist frei wählbar. Das Modulpasswort (SMS-Passwort) kann hier auch eingegeben werden, hat aber für die App keine Bedeutung. Dann prüfen ob soweit alles klappt: die App sollte nun ebenfalls den Ladestand und die Position des Fahrzeugs anzeigen.
- **Schritt 8:** Falls gewünscht für die SEVCON-Konfiguration den CAN-Schreibzugriff aktivieren; entweder per SMS **FEATURE 15 1** oder per App (Fahrzeug-Konfiguration → Features).

Damit ist das OVMS vollständig einsatzbereit. Weitere Optionen stehen in den Manuals. Mit der aktuellen Firmware und App kann man nun auf die SMS verzichten und die Befehle über den Tab "Nachrichten" senden. Auch die Modul-Benachrichtigungen müssen nicht mehr per SMS kommen, dazu einfach unter Fahrzeug-Konfiguration → Parameter die Kanäle von "SMSIP" auf "IP" ändern.

Falls es irgendwo klemmt: bitte zuerst die **Support-FAQ (englisch)** durchlesen und die genannten Punkte sorgfältig prüfen!

Wichtiger Hinweis: Damit die nun eingestellte Konfiguration beim Flashen nicht gelöscht wird muss in der Flash-Software die Option "Preserve device EEPROM" aktiviert sein!

Falls mal die Konfiguration komplett klemmt einfach einmal ohne "Preserve EEPROM" flashen. Dadurch wird die gesamte Konfiguration neu initialisiert. Danach wieder mit **REGISTER**

OVMS anfangen.

Wie komme ich an mein erstes Tuning?

Nochmal der Hinweis: immer erst die Bestätigung eines Befehls abwarten, auch wenn man diese per App sendet. Falls eine Fehlermeldung kommt, Schreibweise prüfen und erneut senden. Die App zeigt ggf. auch Befehls-Fehler an wenn die Verbindung schlecht ist.

Vorbereitung: falls noch nicht geschehen zuerst per `FEATURE 15 1` den Schreibmodus aktivieren.

Die drei Hauptwünsche sind **mehr Geschwindigkeit, mehr Leistung und mehr Rekuperation**. Dafür ist ein guter Startpunkt:

- **Schritt 1:** Twizy einschalten, aber nicht zünden (d.h. nicht auf "GO")
- **Schritt 2:** Höchstgeschwindigkeit auf 100 km/h, Warnung bei 95 km/h: `CFG SPEED 100 95`
- **Schritt 3:** Drehmoment auf 70 Nm (Maximum), Leistung auf 120%: `CFG POWER 130 120`
- **Schritt 4:** Rekuperation auf 20% beim Rollen und 40% beim Bremsen: `CFG RECUP 20 40`
- **Schritt 5:** Profil speichern: `CFG SAVE 1`
- **Schritt 6:** Twizy ausschalten, normal starten und Proberunde drehen.

Nun kann man immer nach dem Einschalten (vor "GO") über die App zwischen dem Standard-Profil und Profil 1 wechseln. Ohne App einfach per `CFG LOAD 0` bzw. `CFG LOAD 1`, oder mit der SimpleConsole durch Drücken der Taster. Zwei weitere Profile kann man unter Nr. 2 und 3 abspeichern.

Für die nächsten Experimente bietet sich das Rampen-Tuning an (`CFG RAMPS`, `CFG RAMPL` und `CFG SMOOTH`), und für die Fortgeschrittenen dann die Erstellung eigener Geschwindigkeits-Kennlinien (`CFG TSMAP`).

Im User-Manual findet man ein paar Beispielprofile -- bitte dadurch nicht vom eigenen Experimentieren abhalten lassen!

Kann man ohne Handy Tuning-Profile umschalten?

Ja. Über die Gangwahl-Taster D/N/R kann man einen Notfall-Reset auf die Standardkonfiguration durchführen. Dazu einfach direkt nach dem Einschalten des Twizy (vor "GO") dreimal D oder R drücken.

Die **optimale Lösung** ist eine "SimpleConsole" mit vier Tastern und LEDs, die an das Modul angeschlossen wird. Die LED zeigen an, welches der vier Fahrprofile (Standard oder Tuning 1/2/3) aktiviert ist, mit den Tastern kann das Profil umgeschaltet werden.

Auch **während der Fahrt** können der Leistungs- und Rekuperationslevel sowie die Rampen und die Glättung jederzeit umkonfiguriert werden. So kann man bspw. im **Winter bei Blitzeis** einfach durch Druck auf den entsprechenden Taster die Rekuperation während der Fahrt ausschalten.



Leider gibt es die SimpleConsole bislang nicht fertig zu kaufen. Sie ist aber sehr einfach und kann von jedem versierten Bastler innerhalb etwa einer Stunde nachgebaut werden. Die Kosten für die Bauteile liegen bei etwa 20 Euro. [Bauanleitung SimpleConsole \(PDF\)](#)

Warum soviele Logins und Passwörter?

Tipp: für den Anfang genügt es auch, gleiche Logins und Passwörter für alles zu verwenden.

Das Modul bekommt ein SMS-Passwort, damit man es auch noch ansprechen kann wenn man sein registriertes Handy verloren hat.

Der Server braucht zwei Logins weil man eine ganze Flotte mit einem Account verwalten kann. D.h. der Hauptlogin ist für den Server-Account, und darin kann man beliebig viele OVMS-Fahrzeuge registrieren, die jeweils einen Fahrzeuglogin erhalten.

Der Hauptlogin wird nur für die Verwaltung der Fahrzeuge verwendet, das Modul und die App kennen immer nur die Fahrzeug-Logins.

Übrigens kann man durch die "Social Groups" dann auch die aktuellen Positionen der Fahrzeuge untereinander austauschen lassen.

Wieso muss man die Firmware aktualisieren?

Die **aktuellste und vollständigste direkt flashbare Version** ist die aus dem aktuellsten ZIP auf der Seite dexters-web.de/downloads.

Die Standard-Firmware im Modul ab Werk hinkt üblicherweise hinter der aktuellen Entwicklung einige Monate hinterher, und unterstützt daher die neuesten Funktionen noch nicht.

Die vorkompilierte Twizy-Firmware aus dem Github ist zwar aktuell, enthält aber ggf. nicht alle Funktionen für den Twizy (Entscheidung des Projekt-Maintainers). Das betrifft derzeit vor allem das Batterie-Monitoring (**BATT**) und die SEVCON-Konfiguration (**CFG**).

Alternativ kann man die Firmware natürlich auch selbst kompilieren, dann hat man die volle Kontrolle über den Funktionsumfang. Infos dazu gibt es in der [OVMS Developer-Dokumentation](#).

Wie aktualisiert man die Firmware?

Das hängt davon ab, welchen Programmer man gekauft hat:

- **PICkit 2**: idleloop.com
- **PICkit 3**: tff-forum.de

Die MPLAB Entwicklungsumgebung ("IDE") enthält das Programmer-Tool "IPE". Die Software gibt es kostenlos für Linux, Mac-OS und Windows:

www.microchip.com/pagehandler/en-us/family/mplabx/

Spannungsproblem bei PICkit 3: manche USB-Ports liefern keine 5 V, bspw. liefern viele Macs nur 4,7 V. Normalerweise kann man damit trotzdem flashen, man muss dazu nur im Programmer einstellen, dass das OVMS mit 4,7 V Nennspannung betrieben werden kann. Ist die Spannung noch niedriger, dann muss man das Modul extern mit Strom versorgen (siehe Manual).

Wie installiert man die aktuellste Android-App?

Die aktuellste Version der Android-App findet man nicht im Play Store sondern auf dexters-web.de/downloads.

Um diese Version zu installieren muss man folgende Schritte ausführen:

1. ggf. die aus dem Play Store bezogene Version deinstallieren
2. unter Einstellungen → Sicherheit die Option "Unbekannte Herkunft" aktivieren
3. das APK-Paket herunterladen und installieren

4. falls Probleme beim Start oder in der App auftreten, einmal die App-Daten löschen

Typische Probleme und deren LED-Blinkcodes

- **Rot 1:** Empfangsstörung, i.d.R. örtlich bedingt schlechter Netzempfang, evtl. auch Antennenproblem
- **Rot 6:** OVMS bekommt kein Netz, Ursache i.d.R. schlechter Empfang oder SIM nicht freigeschaltet
- **Rot 7/8:** OVMS bekommt keine Datenverbindung (GPRS), bspw. weil APN-Daten falsch, Guthaben alle oder Provider Datenverbindungen sperrt
- **Rot 9:** Server nicht erreichbar (evtl. gerade down) oder Login schlägt fehl wegen Fehler in Serveradresse / Fahrzeugkennung / -passwort

Falls das Modul per SMS erreichbar ist:

1. LED- und Netzstatus abfragen: **DIAG**
2. APN abfragen: **GPRS?**
3. Fahrzeugkennung abfragen: **MODULE?**
4. Serveradresse und -Passwort abfragen: **SERVER?**

...und die Ergebnisse im Twizy-Forum posten.

Wozu brauche ich **CFG PRE** und **CFG OP**?

Benutzt man nur die Makro-Kommandos wie **CFG SPEED** und **CFG POWER**, dann braucht man diese Befehle nicht.

Möchte man weiter in die Tiefen der SEVCON-Konfiguration eintauchen, dann braucht man diese, um bestimmte Änderungen durchführen zu können.

Um den SEVCON zu konfigurieren werden Speicher-Register verändert, sogenannte "SDO" = "Service Data Object". Manche davon können jederzeit live verändert werden, andere nur wenn der SEVCON im "PRE"-Modus ist. Der "PRE"-Modus ist sozusagen der BIOS-Screen des SEVCON, und analog dazu auch nur direkt nach dem Einschalten erreichbar.

Die Low-Level-Kommandos **CFG READ** / **CFG WRITE** etc. ermöglichen den Zugriff auf alle SEVCON-Register. Welche es gibt und was diese bedeuten steht in der SEVCON-Doku und im Master-Dictionary aus dem DVT-Softwarepaket.

Ob ein Register den "PRE"-Modus braucht sagt uns der SEVCON netterweise auch selbst wenn man versucht es zu verändern.

Beispiel: nachdem man Seite 13 des User-Guides gelesen hat möchte man, dass der Twizy statt bei 15% erst bei 10% SOC anfängt, die Leistung zu reduzieren. Das SDO dafür hat die Adresse 3813.13 und der Wert wird mit 50 skaliert. Zuerst probiert man also:

```
CFG WRITE 3813 13 500
```

...und bekommt als Antwort:

```
CFG WRITE: OLD: SDO 0x3813.13: 0x02EE = 750
```

```
ERROR 0008 SDO ABORT 0x3813.13: 0x08000004
```

Das besagt zweierlei: 1. der alte Wert ist 750 ($750/50=15\%$), und 2. der WRITE hat nicht geklappt. Der Code 0x0800... ist ein CANopen/SEVCON-Standardfehlercode. Die kann man im SEVCON-Manual in Kapitel 7 nachlesen: "4" bedeutet "Not in pre-operational" -- d.h. um dieses Register zu beschreiben braucht man den "PRE"-Modus.

Also macht man nun:

```
CFG PRE
```

...und dann nochmal...

```
CFG WRITE 3813 13 500
```

...was jetzt klappt. Um nun den Twizy wieder fahrbereit zu bekommen macht man abschliessend noch

```
CFG OP
```

...oder schaltet den Twizy 1x aus + wieder ein.

Nun reduziert der Twizy erst bei 10% SOC die Leistung.

Es gibt noch jede Menge Einstellungen im SEVCON, die ggf. auch für den Twizy sinnvoll änderbar sind.

Wie kann ich mich bedanken und das Projekt unterstützen?

OpenVehicles ist ein nicht-kommerzielles Hobbyprojekt von Enthusiasten für Enthusiasten. Es gibt keine regulären Erlöse oder Sponsoren.

Als Entwickler der Twizy-Firmware und Betreiber des deutschen OVMS-Servers dexters-web.de freue ich mich daher über Spenden jeder Höhe. Einfach mit Stichwort "OVMS" per Paypal an dexter@dexters-web.de - Danke!