

## OpenVario Upgrade

Dokumenten-version	Datum	Beschreibung
1.0	10.11.2023	1.Beschreibung des ‚neuen‘ Upgrades
1.1	17.12.2023	Link-Korrektur
1.2	18.12.2023	Aktuelle Version 3.2.20.4 (Patch 4 von 3.2.20)

Das OpenVario hat bisher leider keine so richtige Upgrade-Funktion, da ist bereits sehr viel begonnen worden – aber letzten Endes nicht komplett zu Ende geführt. Das Yocto-System bietet zwar eigene Packages an, mit denen man das machen könnte (mit ziemlich umfassender Funktionalität), aber das habe ich erst zu spät entdeckt - da ich aber mit der Weiterentwicklung des bestehenden Systems außerdem ziemlich weit fortgeschritten war, habe ich dann da weiter gemacht – und hoffe jetzt, dass ich die meisten Features nun schon fertiggestellt habe... Im bisherigen System gab es eine Datei namens ‚**ov-recovery.itb**‘ von 2015, die es bereits ermöglicht hat, vom USB-Stick zu booten – und das OpenVario mit einem neuen bzw. anderen Image zu versehen. Damit gab es allerdings 3 Probleme: Nicht jede Firmware erkennt in ihrem Boot-Sector den USB-Stick bzw. die Upgrade-Datei zuverlässig, die Upgrade-Datei erkennt nur Firmware-Images nach dem (alten) Namensstandard (ein neueres Namens-System wie **OV-3.1.19-CB2-CH57.img.gz** werden nicht gefunden!), das Upgrade überschreibt die SD-Card völlig neu – genauso wie es die Imager-Programme tun, so dass man erst nach ein paar Vorarbeiten (Sichern des bisherigen Daten- und Settings-Standes) und Nacharbeiten (Rückspielen dieser Daten) wieder in alter Form arbeitsfähig ist. Ob man dafür das Kedder-Menü nimmt, sich mit dem ‚alten‘ Shell-Menü ‚**ovmenu-ng**‘ herumquält (z.B. keine richtige Fehlermeldung wenn der USB-Stick nicht dran ist, Verzeichnisse fehlen, und und und) - oder ob man über SSH (z.B. FTP) die Daten herunterholt und danach wieder aufspielt, spielt dabei erst einmal keine Rolle...

Außerdem gab es noch ein zusätzliches Problem: Die letzten Images sind nur noch 1GB groß, ursprünglich z.B. bei der 17119 waren es noch knapp 4GB. Das kann zwar jeder beim Build für sich ändern – aber das machen nur die wenigsten, die meisten Nutzer nehmen die fertigen Image mit 1.1GB! Je größer die Images, umso länger dauert allerdings dann immer das Aufspielen, bei kleineren Images ist aber die Datenreserve für die Log-Dateien, XCM-Maps, Flugdateien usw. so schnell aufgebraucht, dass es vereinzelt auch zu Abstürzen kommt, wenn der Speicher voll ist (wurde mir jedenfalls schon mehrfach berichtet). Deshalb war auch das Ziel des ‚neuen‘ Upgrades, eine extra Datenpartition einzurichten. Das hat mehrere Vorteile: Die Upgrades gehen viel schneller als über die zum Teil recht langsamen USB-Sticks, die XCSoarDaten (bzw. OpenSoarDaten) brauchen NICHT vor und nach dem Upgrade hin und her kopiert werden: Sie bleiben einfach auf der SD-Karte und werden nach dem Upgrade wieder eingebunden (Partition wiederhergestellt und gemounted), die SD-Karte hat keinen ungenutzten Speicher mehr (bei einer heute normalen 16GB Karte stehen 15GB für Daten zur Verfügung – aber auch die häufig verwendeten 4GB-Karten haben dafür immerhin 3GB Platz - ganz zu schweigen von den jetzt üblichen noch größeren Karten >= 32GB 😊

Die Root-Partition beinhaltet nunmehr gar keine Daten, nur ein paar Settings – und kann problemlos auf 512MB verkleinert werden! Wichtig ist es, dass sie später nicht mehr größer als maximal 1GB gemacht werden darf, sonst wird die Daten-Partition dann doch wieder überschrieben ;-(

Mit diesem großen Speicherplatz ist es dann aber auch möglich, auch als ‚Systemwiederherstellungspunkt‘ alte Systemstände abzuspeichern, so dass man auch nach einem

fehlerhaften Update – oder weil man aus irgendeinem Grund mit dem neuen Stand nicht klarkommt, wieder auf den vorherigen Stand zurückspringen kann 😞 Das ist aber momentan noch etwas Zukunftsmusik – das muss noch alles zum ‚Leben erweckt‘ werden, aber der entscheidende Anfang ist gemacht.

Ab der heutigen Version >= 3.2.20 gehört das Upgrade gewissermaßen zur Grundausstattung der Firmware und kann auch völlig ohne USB-Stick durchgeführt werden - das bedeutet aber auch, dass bei allen älteren FW-Versionen noch zusätzliche Schritte für ein Upgrade notwendig sind (und ein USB-Stick immer erforderlich ist):

- Ein USB-Stick ist so zu bespielen, wie die Datei-Struktur in der ZIP-Datei OV-Upgrade-CH70.zip gespeichert wurde, d.h. man muss diese ZIP-Datei einfach auf das Wurzelverzeichnis des USB-Sticks auspacken, so dass folgender Inhalt auf dem USB-Stick zu finden ist
  - \
  - **fw-upgrade.sh**
  - **[openvario]**
    - **update-system-config.sh**
    - **[images]**
      - **OV-3.2.20.4-CB2-CH70.img.gz**
      - + weitere FW-Images...
      - **[CH70]**
        - **ov-upgrade.itb**
- Wenn sich mehr oder andere Dateien auf dem USB-Stick befinden, sollten diese aber nicht stören
- Nach dem Einstecken des USB-Sticks im OpenVario muss die fw-update.sh-Datei in das root Verzeichnis des OV kopiert werden
  - **cp /usb/usbstick/fw-upgrade.sh ./**
- Das kann man in der OV-Shell mit einem angeschlossenen Keyboard machen, per SSH-Kommando (falls SSH enabled und das Netzwerk verbunden) oder wie auch immer. Gegebenenfalls muss erst der USB-Stick gemounted werden (m.E. zum Beispiel bei der FW 17119), aber die meisten älteren Firmware binden den Stick per Automount selbständig ein (bei den FW-Ständen der letzten 3 Jahre praktisch immer!)
  - Zusätzliches mounten bei älterer Firmware (wie z.B die weit verbreitete **17119**):
    - **mount /dev/sda1 /usb**
    - Der Unterordner **./usb** sollte eigentlich vorhanden sein, muss deshalb nicht auch noch angelegt werden. Ob nun der USB-Stick in **./usb** oder in **./usb/usbstick** eingehängt wird, spielt für den weiteren Ablauf keine Rolle, nur der Copy-Befehl lautet etwas anders
    - **cp /usb/fw-upgrade.sh ./**
- Danach ist das Shell-Programm **fw-upgrade.sh** in der Shell aufzurufen:
  - **./fw-upgrade.sh**
- Der nächste Schritt ist die Auswahl der korrekten Firmware, im Zip-File ist nur ein Image enthalten – dieses sollte natürlich auch ausgewählt werden! Wenn noch ältere Images sich in dem Ordner **(USB)/openvario/images** befinden, kann man diese auch auswählen!
- Wichtig ist auch noch, dass man keinen ‚Systemwechsel‘ durchführt: Also von **CH57** auf **CH70** oder ähnliches. Auch das soll in Zukunft funktionieren, aber da gibt es im Moment noch ein paar Stolperstellen... Das geht auch heute schon, man darf sich nur nicht darüber wundern, dass danach nicht alles wie gewünscht funktioniert!

- Apropos ‚System‘ oder ‚Machines‘: Ich habe eine Umbenennung der Hardware-Konfiguration durchgeführt: In Yocto/BitBake meckerte es immer über die Großbuchstaben, die Machine-Namen sind auch ‚gewachsen‘ – also wurden immer mal wieder neu angepasst, außerdem sind die Namen ziemlich lang geworden, was sich dann auch in der Länge der Dateinamen niederschlägt... In der folgenden Tabelle habe ich mal die Änderungen für die verschiedenen Systeme zusammengefasst:

Alter Name	MACHINE	System	
openvario-7-CH070	ov-ch70	CH70	
openvario-7-PQ070	ov-pq70	PQ70	
openvario-57-lvds	ov-ch57	CH57	
openvario-43-rgb	ov-am43	AM43	
openvario-7-CH070-DS2	ov-ch70s	CH70s	
openvario-7-AM070-DS2	ov-am70s	AM70s	
openvario-57-lvds-DS2	ov-ch57s	CH57s	

- Damit ist m.E. mit einem sehr kurzen Indexnamen klargestellt, welches Hardwaresystem gemeint ist: die ‚**MACHINE**‘ bildet sich aus ‚ov-‘ und dem (klein geschriebenen) System, die ersten beiden Buchstaben beschreiben den Display-Hersteller, die 2 folgenden Ziffern beschreiben die Größe des Displays – und das eventuell angehängte ‚s‘ die Hardware-Konfiguration von Daniel Smolka (bei anderen Konstellationen müsste dann ein anderer Buchstabe her) .

### Vorteile der neuen Firmware und des Upgrade-Vorgangs

- Es gibt nun eine 3.Partition (‚**data**‘), in der die OpenSoar- und XCSoar-Daten gespeichert sind. Diese brauchen während eines Upgrades nun auch nicht mehr gespeichert und wieder hergestellt werden, weil dieser Bereich nicht überschrieben wird!
- Viele Systemdaten werden automatisch aus dem ursprünglichen Zustand gespeichert und nach dem Upgrade unmittelbar wieder hergestellt! Dazu gehören:
  - Display-Ausrichtung (‚Rotation‘)
  - Display-Helligkeit (‚Brightness‘)
  - SSH-Einstellung (erlaubt oder verboten)
  - WLAN-Settings (‚**connman**‘)
- Folgende Daten werden dabei bewusst nicht berücksichtigt
  - XCSoar-Sprache (Language): Das wird direkt in OpenSoar/XCSoar ausgewählt, wobei da alle vorhandenen Sprachen zur Verfügung stehen (und nicht jede Sprache noch einmal extra im Menü freigegeben werden muss) Apropos: Das Setting der ‚XCSoar-Sprache‘ ist deshalb auch aus dem Menü entfernt worden!
- Folgende Einstellungen werden zurzeit noch nicht verwaltet und werden gegebenenfalls in Bälde nachgebaut (d.h. zur Zeit müssen sie wieder bei Wunsch extra nachinstalliert werden):
  - **varioid** status
  - **sensord** status
  - **opkg**
  - Touchscreen-Einstellungen (‚**pointercall**‘, Calibration)

In diesen Fällen sind nach dem Upgrade die Standard-Voreinstellungen gültig!

- In OpenSoar ist im System-Menü eine neuer Punkt ‚OpenVario‘ vorbereitet, wo die verschiedensten OpenVario-System-Einstellungen direkt vorgenommen werden können –

ohne Shell-, Kedder- oder sonstigen Zusatzmenüs („x-menu‘,...). Momentan ist das leider nur ein Platzhalter, der später unter anderem folgende Punkte beinhalten soll:

- SSH-Enabling
- WLAN-Einstellungen
- **sensord**- und **varioid**-Einstellungen
- Touchscreen-Einstellungen (falls Touchscreen vorhanden)
- Zusätzlicher Speicherpfad für IGC-Downloads (z.B. auf dem USB, falls einer eingesteckt ist)
- Display-Helligkeit
- OpenVario-System-Info

Sprache und Display-Ausrichtung wird bereits in anderen Menü-Punkten bearbeitet, eventuell kann das hier ein 2.Mal einstellbar sein...

- Neue Events wurden zum OpenSoar hinzugefügt, die eine bessere Steuerung des Beenden/Herunterfahrens erlauben:
  - Quit („Beenden“) beendet OpenSoar wie bisher, man landet im Shell-Menü
  - Reboot („Neustart“) beendet OpenSoar – und startet OpenVario noch einmal neu
  - ShutDown („ShutDown“) beendet OpenSoar – und fährt OpenVario herunter und schaltet es aus!

Vor dem ShutDown und dem Reboot ist noch ein (Shell-Menü-)Timer eingebaut worden! Außerdem ist noch ein zusätzlicher Menüpunkt ‚Restart‘ denkbar, der OpenSoar unmittelbar nach dem Beenden wieder startet, ohne das ein Reboot von OpenVario durchzuführen ist! Das ist z.B. nach einer Änderung der Display-Ausrichtung, Änderung der Sprache – oder auch der Änderung des Users möglich...

### **Kurzanleitung bei einem Upgrade von einer älteren Firmware als 3.2.20 auf einen neuen Stand**

- Download der richtigen Zip-Datei (Beachte HW-Version!)
- Entpacken in der Wurzel eines USB-Sticks (leer oder benutzt spielt erst einmal keine Rolle)
- USB-Stick in das (laufende) OpenVario einstecken (möglichst direkt ohne Hub, bitte auch keinen anderen USB-Stick vorher einstecken, sonst verweist evtl. das AutoMount auf einen falschen Stick!) Falls das OV noch aus ist, kann er auch dann erst eingeschalten werden!
- Die Linux-Shell aufrufen, z.B. im Standard-OV-Menü durch den Menüpunkt ‚Exit to shell‘ und dort die Befehle
  - `cp /usb/usbstick/fw-upgrade.sh ./` und
  - `./fw-upgrade.sh`

Eintippen, die richtige Firmware aussuchen (>= ‚3.2.20.4‘ – in der Regel gibt es nur diese eine FW)

- Alle Zwischenabfragen bestätigen oder auf den Ablauf der Timer abwarten...
- Nach verschiedenen Boot- und Kopiervorgängen startet wieder OpenSoar – nun allerdings mit der neuen Firmware!

### **Kurzanleitung bei einem Upgrade von einer neueren Firmware**

Prinzipiell kann die gleiche Zip-Datei (und der gleiche Ablauf) wie bei einer älteren Firmware genommen werden – es wird aber hier nur noch die (neue) Firmware-Version(en) im Ordner ‚**openvario/images**‘ benötigt, es sind keine weiteren Dateien erforderlich!

- Entweder:
  - Download der richtigen Zip-Datei (Beachte HW-Version!)
  - Entpacken in der Wurzel eines USB-Sticks
  - USB-Stick in das OpenVario einstecken
- Oder:
  - In einem USB-Stick in Ordner **,openvario/images'** die Firmware-Datei ablegen
- Im Shell-Menü den Menüpunkt **,Update, Settings...** -> **Upgrade Firmware'** auswählen
- Weiter wie oben: Firmware auswählen -> und dann den Upgrade-Prozess abwarten...
- Übrigens: Falls kein Image im images-Ordner der Data-Partition bzw. des USB-Sticks gefunden wird, springt das Menü wieder in das Hauptmenü.

Bei der Erarbeitung eines neuen Update-Konzeptes gibt es mit der OV-Version 3.2.20 eine Zäsur: Ältere Firmware (Damit meine ich auch neue Firmware von OpenVario nach dem alten Standard) benötigt ein paar zusätzliche Dateien auf einem USB-Stick, damit man das Upgrade und die Rekonstruktion der Daten ‚on the fly‘ auf dem OpenVario durchführen kann – die neuere Firmware benötigt das nicht mehr. Weil im 2.Fall außerdem viel weniger Daten umzukopieren sind, geht es auch am Ende viel schneller – aber erst muss diese neuere Firmware installiert werden... Das geht also mit einem präparierten USB-Stick – oder in der alten Form, dass man die SD-Karte außerhalb vorbereitet / präpariert – in das OpenVario einsetzt, die vorher gesicherten Einstellungen und Programm-Dateien (XCSoar oder OpenSoar) wieder aufspielt – und damit später die Vorteile des schnellen Upgrades nutzt. Aber, wie gesagt, das Upgrade funktioniert auch mit der alten Firmware ... (Mit dem USB-Stick kann man auch ältere Firmware bzw. Firmware, die noch keine interne Upgrade-Funktionalität besitzt, ohne weiteres auch verwenden).

Das Upgrade unterscheidet sich demnach in 4 Fälle:

- [1] – Neu auf Neu: der eigentliche Fall: auf dem OV ist bereits eine neue FW – und die ist auf einen neueren Stand zu bringen
- [2] - Alt auf Neu: Das OV hat eine FW die noch nicht die Upgrade-Option hat (17119, 22304 oder auch 3.0.1-18). In diesem Fall wird die ältere FW auf einen neueren Stand gebracht – kann aber auch wieder über den nächsten Fall [3] wieder zurück auf ‚bekannte Pfade‘ reaktiviert werden.
- [3] Neu auf Alt: Reaktivierung eines älteren – bekannten - Standes ohne interne Upgrade-Funktion
- [4] Alt auf Alt: Upgrade von Firmware-Images ohne die neueren Feature wie die Upgrade-Funktionalität, aber auch ohne den großen Daten-Speicher als 3.Partition... (und natürlich möglicherweise auch ohne OpenSoar 😊 )

Momentan scheinen alle 4 Fälle zu funktionieren, natürlich hoffe ich sehr, dass Euch die Vorteile der Variante 1 überzeugen – und Ihr auf diese Variante umschaltet!

Viel Spass beim Probieren!

Mit ‚neuerer Firmware‘ meine ich konkret Images, die nach dem (meinen) neueren Standard gebaut wurde, also mit 512MB Firmware (Partition 1 und 2) und einer großen Data-Partition (ext4), die sich

der Größe der SD-Karte anpasst! Alle anderen Images – auch die neu Veröffentlichten auf [GitHub.com/Openvario/meta-openvario](https://github.com/Openvario/meta-openvario) fallen unter die Kategorie ‚Alte Firmware‘