



Construction and installation instructions for the FreeVario

Table of Contents

I.	General.....	2
II.	Assembling the boards and assembling the vario gauge.....	2
III.	Programming the two ESP32.....	7
IV.	Backing up and restoring your settings.....	9
V.	Install the new OV image including XCSoar with FreeVario driver.....	11
VI.	Install XCSoar incl. FreeVario driver without reinstalling the OV image.....	12
VII.	Switch off the vario sound from OpenVario.....	12
VIII.	Settings for a doubleseater.....	13
IX.	Blocking the standard profile for club use.....	13
X.	Settings in XCSoar.....	13
XI.	Settings for the stick remote control.....	13
XII.	Installation of the new OpenVario menu from Kedder.....	15
XIII.	Update maps, airspaces etc. with the new menu from Kedder.....	16
XIV.	Installation.....	16
XV.	Disclaimer.....	17

Rev. 1.1 (01.2022)

I work on a Mac or Linux system. Little things can differ on a Windows computer!

I. General

The FreeVario is operated with 12V. If e.g. the micro USB plug is only connected to 5V, it basically works, but the audio amplifier will not work properly. He will makes noises. So this is not a mistake, it is due to the wrong voltage.

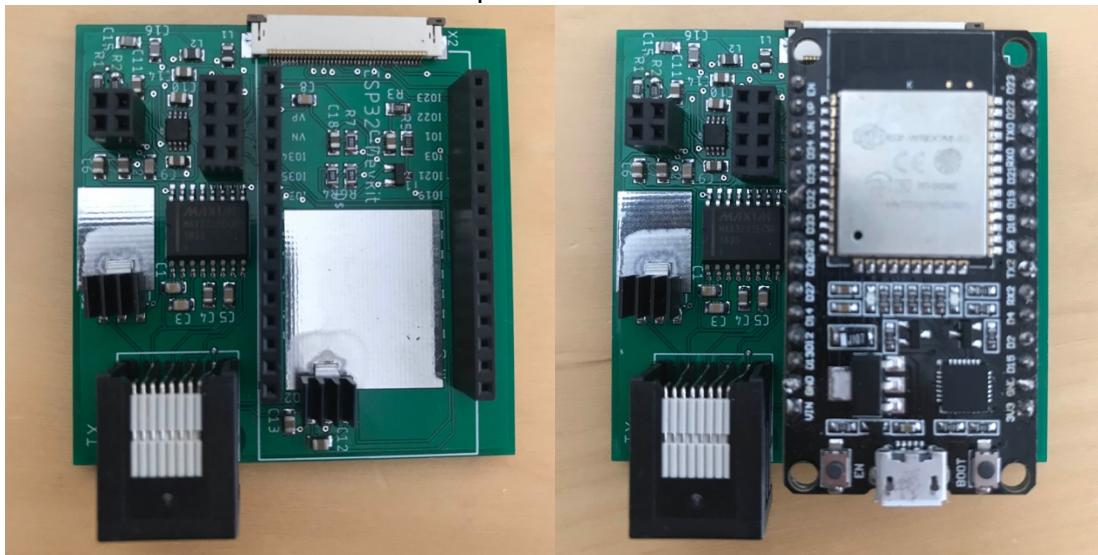
The dimensional accuracy of the 3D printed parts is extremely dependent on the settings of your printer and the material. So it can happen that the parts are too small. Then you have to print the parts a few percent larger.

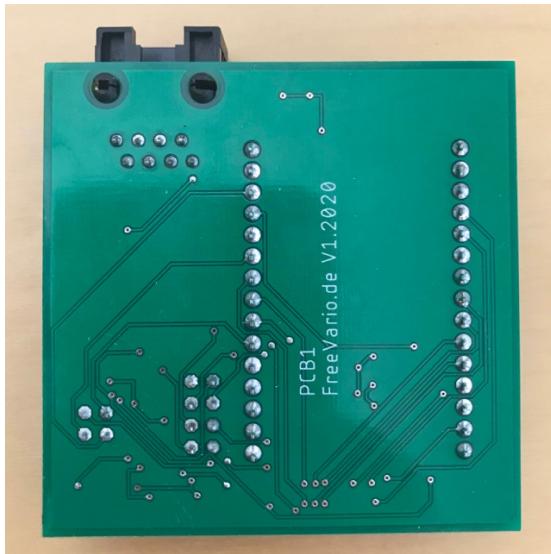
There seems to be different board sizes for the ESP32. The result is that the housing does not close properly. For this reason I always cut off one corner of the board.



II. Assembling the boards and assembling the vario gauge

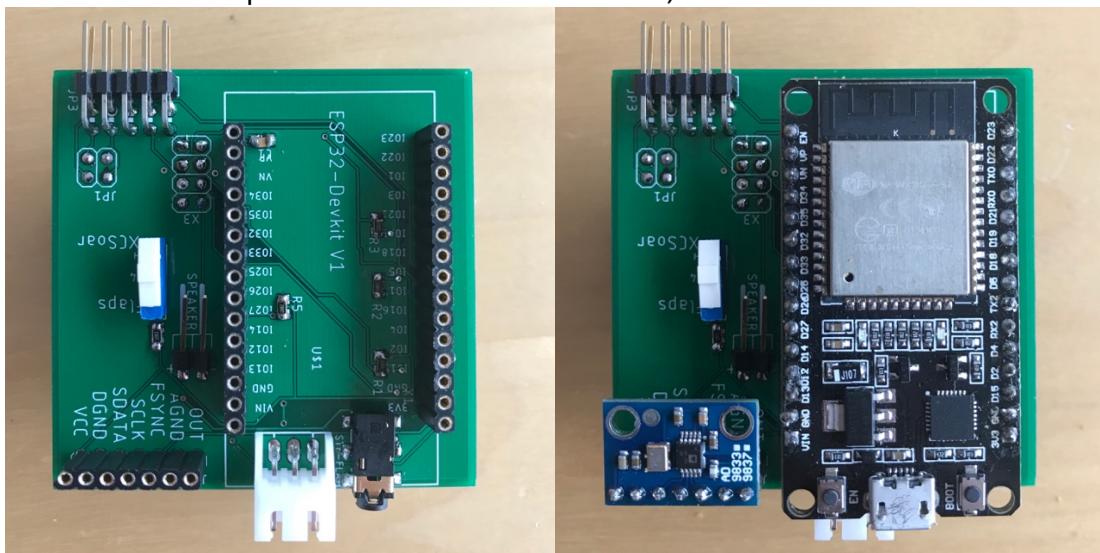
1. Solder PCB1 as well as shown in the pictures.

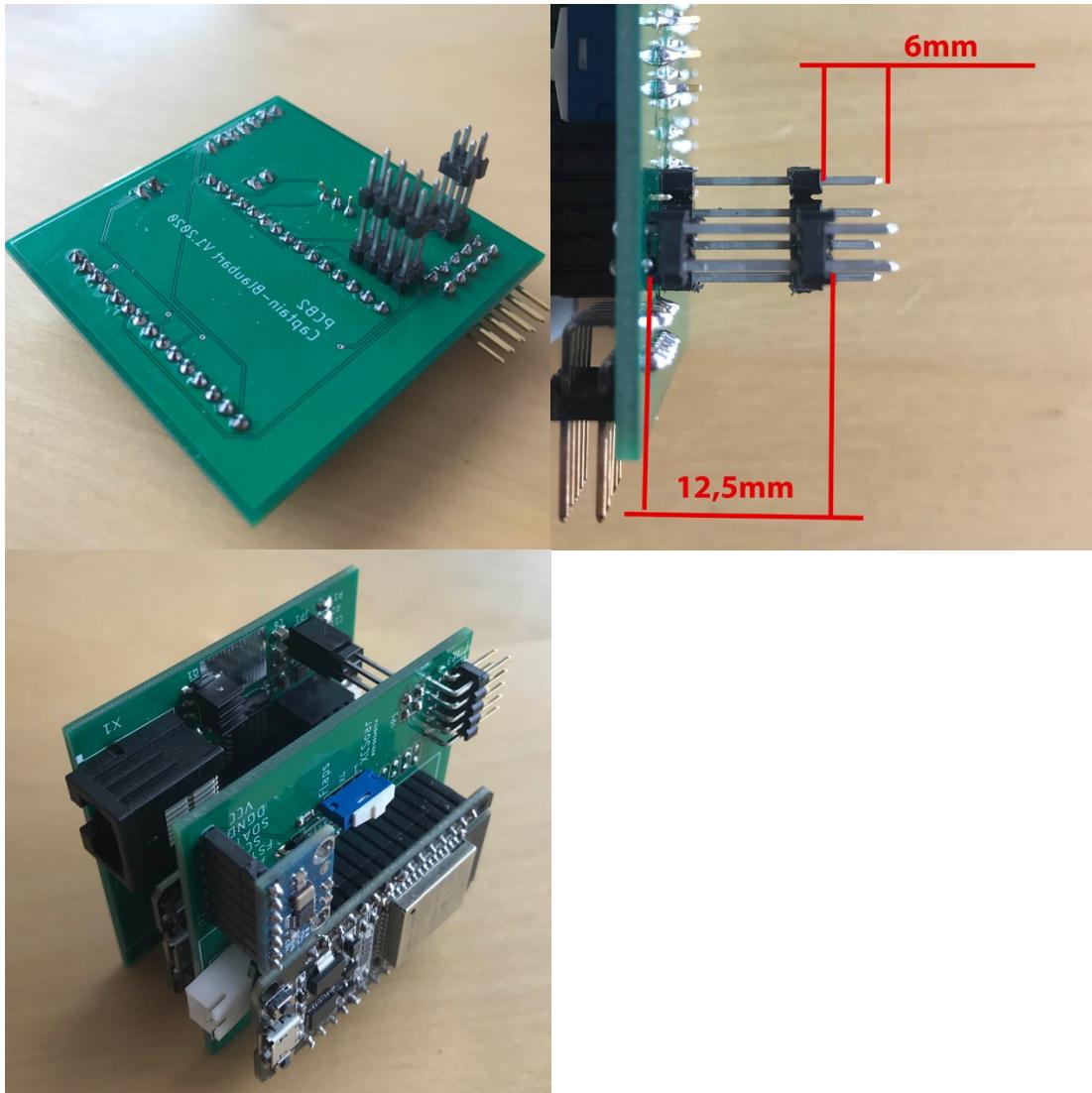




2. Solder PCB2 as well as shown in the pictures. The distance between the two plastic holders on the stacking strip must be set to 8mm. Shorten the long pins with a side cutter to 22 mm (see 3rd picture).

With the help of the slide switch you can set how the vario will later work in automatic mode. If you put it on XCSoar, XCSoar sends commands to switch between STF and vario. If you set it to flaps, the automatic mode switches between the STF and vario using a switch connected to the white, three-pin connector. This switch can e.g. be attached to the flaps. If the vario is to be set to STF, this switch should be closed.



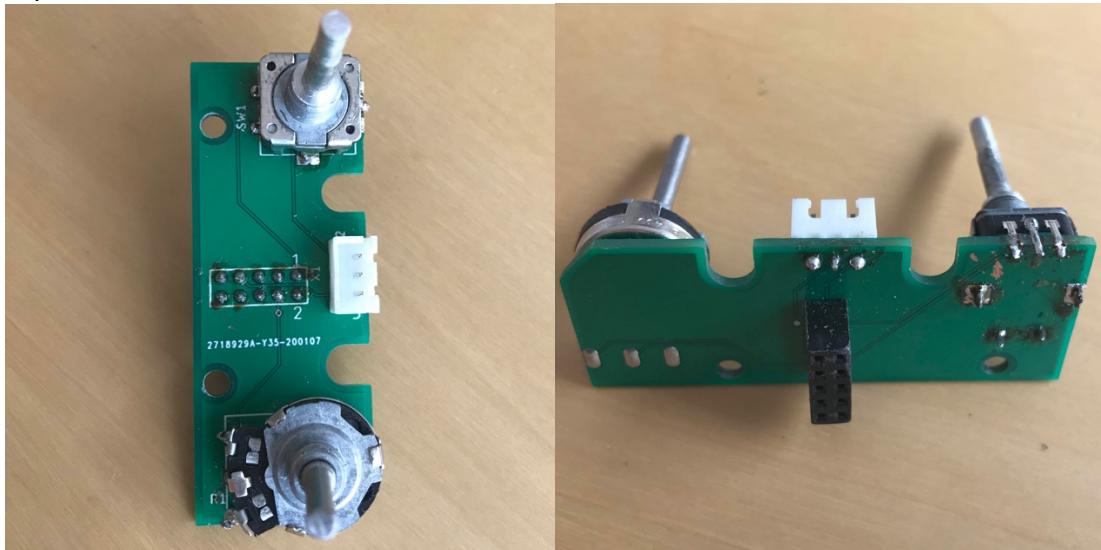


The 2.5mm jack socket is used to connect an external speaker. If the external speaker is connected, the internal one is automatically switched off. You have to be very careful when soldering the jack socket. If too much solder is used, the switch in the jack socket will not work properly because the solder penetrates and blocks. I recommend measuring the socket with and without the plug after soldering and, if necessary, unsoldering it and trying again.

3. Carefully disassemble the rotary encoder by bending the small metal brackets. The axis of the rotary encoder is secured with a locking ring. You have to go carefully with a side cutter into the gap between the axis and the upper part of the housing and press lightly. If the axis is slightly damaged by the side cutter, file away the spot. Clamp the axes in the lathe, turn by 4mm. Then carefully reassemble the rotary encoder and bend the metal brackets back. Secure the axis of the rotary encoder again with the ring. For safety's sake, buy the encoder twice, the first time it may go wrong. Drills the banjo bolts to 4.3mm.

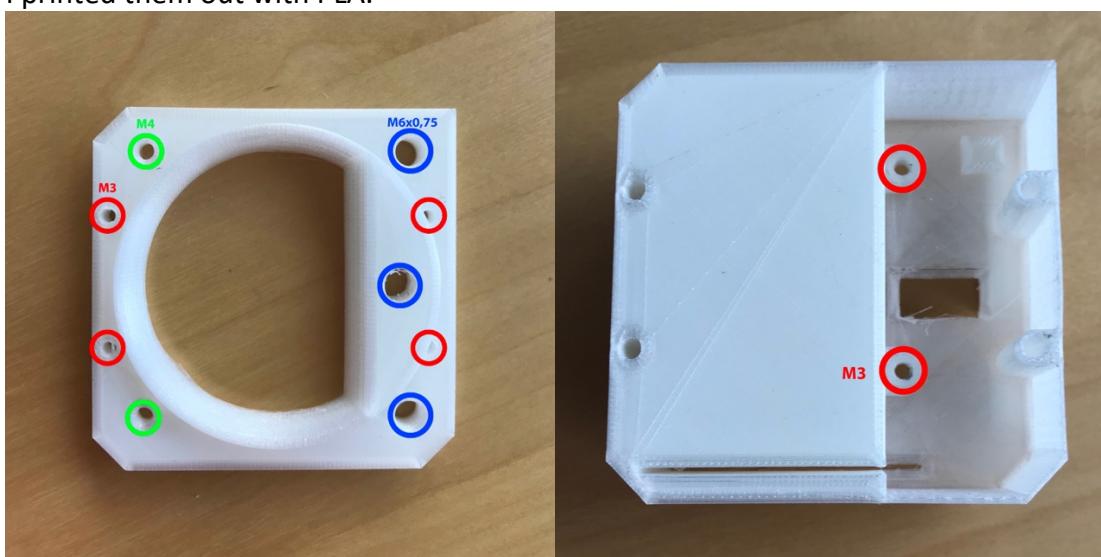
4. On the PCB3 at the position where the encoder is located, you have to file away the protruding part of the board on the side with the 3 PINs and above, otherwise the board will not fit into the housing without tension. It may well be that you have to file

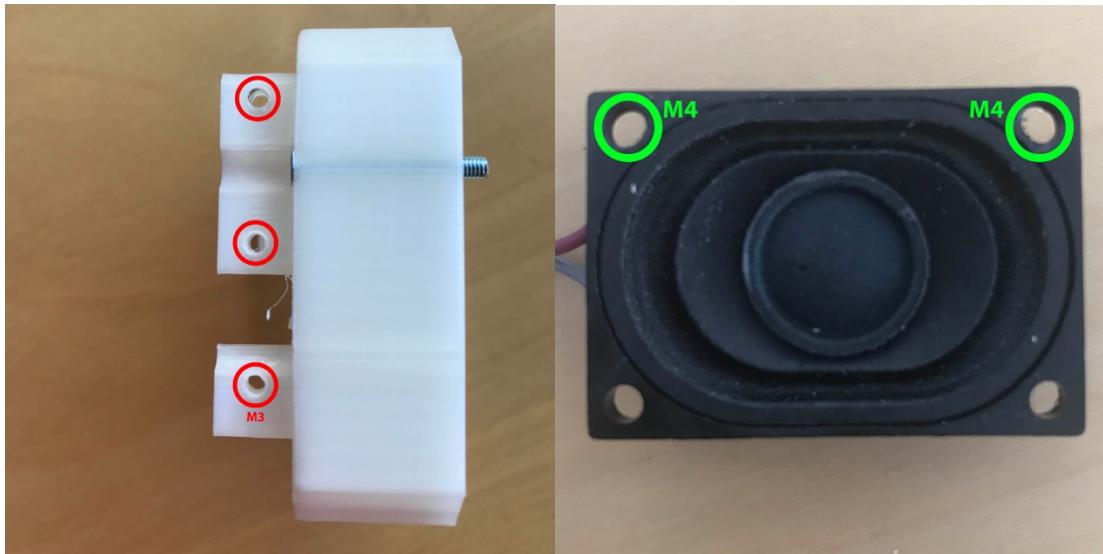
a little more after soldering. Carefully bend the contacts of the potentiometer downwards by 90° and solder them on. Test whether the axes of the encoder and the potentiometer fit well through the banjo bolts and are easy to operate after installing the front plate. If necessary, loosen again and move something. Solder PCB3 as well as shown in the pictures. Make sure that you solder the white connector so that the PINs are on the right side. Otherwise the board will not fit into the housing later. Shorten the axes to a length of 34mm + thickness of the instrument board, measured from the top of the board!



5. Print out the three housing parts and cut the threads into the plastic as marked in the pictures. Red = M3, green = M4, blue = M6x0.75.

I printed them out with PLA.





6. Cut two M4 threads in the speaker and fasten the speaker in the housing with 2 countersunk screws M4x8. Solder about 10cm long cables to connect the speaker beforehand. **The speaker is only required if the FreeVario is to output the vario sound. If you want to continue using the vario sound from OpenVario, you can omit the speaker.**
7. Crimp the connector to an approximately 5cm long, 3-pin cable and solder it to the toggle switch. Solder as space-saving as possible by soldering at an angle. Otherwise there will be a short circuit later between the solder joints and the contacts behind them on the PCB3. Make shrink wise the solder contacts!
The middle cable comes to the middle solder lug. Depending on how you solder the two outer cables, you will later have STF or Vario on top. If you solder the upper cable of the plug to the upper soldering tab of the toggle switch, and the lower cable of the plug to the lower soldering tab of the toggle switch, STF will be in the lower position of the toggle switch, and Vario in the upper. Automatic is in the middle.
8. Shorten the threads of the two M4 fastening screws to 3mm + instrument panel thickness. Work very precisely here, because if the screws are too long, press the display later and destroy it. It is best to screw the front panel into the cockpit on your own and check that the screws do not protrude into the area of the display.
9. Install the toggle switch and the display in the front panel.
10. Insert PCB3 in the middle part and screw tight with 2 screws M3x6. Tighten screws carefully, the threads are plastic!
11. Screw the front panel and middle section together using 4 countersunk screws M3x25. Connect the toggle switch beforehand and lead the FFC cable back through the slot. Tighten screws carefully, the threads are in plastic!
12. Now first test whether the two circuit boards can be easily pushed into the pot individually. Otherwise damage can occur later. If in doubt, rework the groove a little.

13. Carefully assemble PCB1 and PCB2, connect display, loudspeaker (do not interchange plus and minus) and connect PBC3. Close the clamping mechanism of the display cable very carefully with a small screwdriver or something similar. If you slip with your fingers or the wrong tool, it is easy for you to tear the ribbon cable.
14. The best thing to do now is to jump to the programming of the two ESP32 and then continue here.
15. Now push the housing pot carefully over the circuit boards. Make sure that the boards are in the rails and carefully guide the Western Digital socket through the rear wall.
16. Screw the housing parts together with 6 countersunk screws M3x6. Tighten screws carefully, the threads are in plastic!
17. Using the two fastening screws and the two banjo screws, install the Vario in the I-board, mount the knobs and connect the Vario. Tighten screws carefully, the threads are in plastic!
18. The connection cable has the same assignment on both sides, i.e. same color on the same PIN, like a normal LAN cable.

III. Programming the two ESP32

1. Put the Binaries folder e.g. to the desktop.
2. Open a terminal or the command prompt. There you can see if you can find the path to Binaries. For example, it is /Users/PC1/Desktop/Binaries/
3. In Arduino IDE click on Arduino -> settings.
4. Add the URL under "Additional Board Administrator URLs". If URLs have already been entered, add further URLs with commas and spaces.
https://dl.espressif.com/dl/package_esp32_index.json
5. Also click here under "Verbose output during" "Upload".
6. Click Tools -> Board -> Board Administrator and search for ESP32.
7. Select „ESP32 by Espressif Systems“ and install.
8. You may have to install the driver for the serial adapter of the ESP.
<https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>
9. Open and save a new sketch. Remember the name. To upload, set the following under Tools:

- Board: DOIT ESP32 DEVKIT V1
- Upload Speed: 115200
- Flash Frequency: 80MHz
- Core Debug Level: "none"
- Port: Check out which is added after connecting the ESP32 with the USB cable

10. Then upload the empty sketch. There is a very long command in white letters above the error message in the Arduino IDE, which is roughly as follows:

```
/Users/PC1/Documents/Arduino/hardware/espressif/esp32/tools/esptool/esptool --chip esp32 --port /dev/cu.SLAB_USBtoUART --baud 115200 --before default_reset --after hard_reset write_flash -z --flash_mode dio --flash_freq 80m --flash_size detect 0xe000
/Users/PC1/Documents/Arduino/hardware/espressif/esp32/tools/partitions/boot_ap_p0.bin 0x1000
/Users/PC1/Documents/Arduino/hardware/espressif/esp32/tools/sdk/bin/bootloader_dio_80m.bin 0x10000
/var/folders/6t/qjgfw2413f7ddnjzk_08btwx0000gp/T/arduino_build_661147/sketch_dec30d.ino.bin 0x8000
/var/folders/6t/qjgfw2413f7ddnjzk_08btwx0000gp/T/arduino_build_661147/sketch_dec30d.ino.partitions.bin
```

11. Copy this command, you need it to load the file onto the ESP32.

12. The first part of the command shows you where Arduino IDE creates its working directory. Remember the place. In this example it would be:

/Users/PC1/Documents/Arduino

13. Change the command with your location where the Binaries folder is located. Also change the last two file names (see bold).

Display-ESP:

```
/Users/PC1/Documents/Arduino/hardware/espressif/esp32/tools/esptool/esptool --chip esp32 --port /dev/cu.SLAB_USBtoUART --baud 115200 --before default_reset --after hard_reset write_flash -z --flash_mode dio --flash_freq 80m --flash_size detect 0xe000 /Users/PC1/Desktop/Binaries/boot_app0.bin 0x1000
/Users/PC1/Desktop/Binaries/bootloader_dio_80m.bin 0x10000
/Users/PC1/Desktop/Binaries/FreeVarioGauge.ino.bin 0x8000
/Users/PC1/Desktop/Binaries/FreeVarioGauge.ino.partitions.bin
```

Sound-ESP:

```
/Users/PC1/Documents/Arduino/hardware/espressif/esp32/tools/esptool/esptool --chip esp32 --port /dev/cu.SLAB_USBtoUART --baud 115200 --before default_reset --after hard_reset write_flash -z --flash_mode dio --flash_freq 80m --flash_size detect 0xe000 /Users/PC1/Desktop/Binaries/boot_app0.bin 0x1000
/Users/PC1/Desktop/Binaries/bootloader_dio_80m.bin 0x10000
/Users/PC1/Desktop/Binaries/VarioSound.ino.bin 0x8000
/Users/PC1/Desktop/Binaries/VarioSound.ino.partitions.bin
```

14. Connect the ESP32 to the PC using the USB cable and open a terminal or the command prompt. Issue the **appropriate** command there.
15. After the upload is complete, the ESP32 restarts. The Sound-ESP32 is done. If the display is already connected, the display ESP32 ends with a black screen. Files still have to be uploaded to the memory of this ESP32.
16. Install ESP32 Sketch Data Upload. Download ESP32FS-1.0.zip from <https://github.com/me-no-dev/arduino-esp32fs-plugin/releases/>
17. On Mac, copy the extracted folder ESP32FS to /Programme/Arduino (Show package contents)/Contents/Java/tools.
18. Under Windows, copy the extracted folder ESP32FS to /Programs/Arduino/tools.
19. Unzip the data.zip file. Go to the working directory of Arduino IDE (e.g. /Users/PC1/Documents/Arduino) and there into the folder of the sketch you just created. Copy the extracted data folder here.
20. Restart the Arduino IDE and open your newly created sketch.
21. IMPORTANT!! Make sure the serial monitor is closed.
22. Click on “ESP32 Sketch Data Upload” under Tools.
23. After the upload is complete, the ESP32 restarts and, if the display is already connected, you can now see the display of the vario gauge.
24. Now go back to point 13 of the assembly part of the instructions.

IV. Backing up and restoring your settings

1. Save the settings:

Variant 1: the original menu is used:

- Back up the .xcsoar folder from the old system
- Boot OpenVario and go to the menu. Select "Exit to the shell" and confirm with Yes. Connect your USB stick to manage the OpenVario and a USB keyboard. Enter the following commands.
- mkdir /tmp/USB
- mount /dev/„USB-Stick“ /tmp/USB („USB-Stick“ usually is /dev/sda1)
- cp -r .xcsoar /tmp/USB (takes a bit of time!)
- cd /tmp/USB
- mv .xcsoar xcsoar
- rm -r openvario/upload/xcsoar
- mv xcsoar openvario/upload/

- Save the value for calibrating the voltage display (if you have calibrated it at all, skip step otherwise):
- Boot OpenVario and go to the menu. Select "Exit to the shell" and confirm with Yes. Connect a USB keyboard and enter the following commands.
- nano /opt/conf/sensord.conf
- write down the value for the "voltage_config" variable. For me it was „voltage_config 1592.4 4.54545“.
- 1592.4 is the scaling factor, 4.54545 is the offset. The formula to convert the value of the AD converter into the voltage value is:
- $U = (\text{value AD converter} / \text{scaling factor}) + \text{offset}$
- Save the WiFi settings:
- Copy the folder /var/lib/connman to the USB stick, which is already mounted.
- cp -r /var/lib/connman /tmp/USB

Variant 2: the new menu from Kedder is used:

- Insert an empty FAT32 formatted USB stick into the OpenVario
- Start the OpenVario and select “Applications” -> “Backup” -> “Backup” in the menu
- Everything important is now automatically backed up

2. Upload your saved data to the OpenVario:

Variant 1: the original menu is used:

- Insert the new SD card into the OpenVario and start. Then go back to the OpenVario menu.
- Click "Copy file to and from OpenVario" -> "Upload files from USB to XCSOar". This will restore all of your settings. Wait until “done !!” is shown.
- Then set the desired orientation of the display under "Update, Settings, ..." -> "System Settings" with "Set rotation of the display".
- If necessary, select “Calibrate Touch” under “Update, Settings,...”.
- Select “Calibrate Sensors” under “Update, Settings,...”.
- Finally, set the language under "Update, Settings, ..." -> "System Settings" -> "Set language used for XCSOar".
- Back to the main menu with ESC and select "Restart" and confirm with Yes.
- Restore the voltage display calibration:
- nano /opt/conf/sensord.conf
- „voltage_config” reset to the noted values
- Restore backup of old WiFi settings:
- Copy the connman folder saved on the USB stick back into the /var/lib directory
- mkdir /tmp/USB
- mount /dev/„USB-Stick“ /tmp/USB („USB-Stick“ usually is /dev/sda1)
- cp -r /tmp/USB/connman /var/lib
- Set up WiFi again if there is no backup of the old settings:
- Boot OpenVario and go to the menu. Select "Exit to the shell" and confirm with Yes. Connect a USB keyboard and enter the following commands.

- connmanctl
- enable wifi
- scan wifi
- services
- agent on
- connect wifi_...
- Enter password

Variant 2: the new menu from Kedder is used:

- Insert the USB stick with your backup into the OpenVario
- Start the OpenVario and select "Applications" -> "Backup" -> "Restore" in the menu
- Everything important is now automatically restored

V. Install the new OV image including XCSoar with FreeVario driver

1. From the official Image 21033 of OpenVario, the FreeVario driver is integrated in the image. So you are no longer dependent on using images from FreeVario.de. If you have already installed an image higher the 21032, you can connect the FreeVario directly to your OpenVario without making any changes at the software.
2. Install the new image:

Variant 1: Use a new SD card (secure method)

- Use a Linux PC to write the new image to another SD card. So you can be sure that on the old map there is still a working system in case something goes wrong.
- unzip the image first:
- gunzip "Path_to_Image"
- e.g. : gunzip /home/USER/Schreibtisch/OpenVario-linux-openvario-image-testing-glibc-ipk-20149-openvario-7-PQ070.rootfs.img.gz
- dd if=" Path_to_image" of=/dev/, new_SD card "
- z.B.: dd if=OpenVario_new.img of=/dev/sda

Variant 2: Overwrite old SD card (insecure method)

- In the main directory of the USB stick "openvario" you put the recovery file "ov-recovery.itb", which you can download on the FTP server (<ftp://ftp.openvario.org/recovery>).
- IMPORTANT!! If you do not want to perform a recovery, the file "ov-recovery.itb" must be renamed, e.g. in "ov-recovery.xxx".
- Copie the image to be installed as a .gz file to the USB stick in the images subfolder
- Insert the USB stick into the OpenVario and boot.
- The recovery menu is shown in red. Remember that all files in OpenVario will be deleted when you restore!
- Select "Write image to SD Card" and then "Update complete SD Card".

VI. Install XCSoar incl. FreeVario driver without reinstalling the OV image

1. If you don't want to reinstall the complete image of OV, but only want to replace the version of XCSoar, you can do that as described below. Your settings in XCSoar remain completely intact. I would always create a backup file of the SD card beforehand. Something always can go wrong!
2. Copy the new version of the installation file from XCSoar to a USB stick, select "Exit to the shell" in the OpenVario menu. Find the path of the USB stick with fdisk -l (e.g. /dev/sda1).
3. mkdir /tmp/USB
4. mount /dev/sda1 /tmp/USB
5. cd /tmp/USB
6. opkg remove xcsoar
7. opkg install xcsoar*.ipk
8. Set the language in the OpenVario menu again and reboot.

VII. Switch off the vario sound from OpenVario

1. **Skip this section completely if you want to continue using the vario sound from OpenVario. If you wanted the FreeVario to make the sound, you have to work through this section.**
2. Choose "Exit to the shell" from the OpenVario menu and enter the command "systemctl disable variod" to switch off the vario sound from OpenVario.
3. Reboot OpenVario.
4. Changes the port of NMEA port A from 4352 to 4353.
5. If vario sound can still be heard in XCSoar, it will be generated by XCSoar.
6. At System -> Display -> Audio Vario set Audio Vario to ON and set volume at 0%. Confirm with Close.

VIII. Settings for a doubleseater

1. To connect the front and rear OpenVario to each other, you have to use a crossed cable. That means you have to swap PINs 5 and 6 for RX and TX on one connector.
2. The front device is set to the RJ45 socket to be used, e.g. ttyS3, NMEA output is set as driver, the baud rate e.g. to 38400 (it must be the same in front and in the back!)
3. The rear device is also set to the RJ45 socket to be used, e.g. ttyS3, OpenVario is set as the driver, and the baud rate must be set in the same way as the front device.

IX. Blocking the standard profile for club use

1. Go to the console in the main menu of the OpenVario, in which you select "Exit to Shell".
2. Go to the directory in which the profiles are saved with "cd .xcsoar".
3. By default, the profile is called openvario.prf. I renamed it to clubprofile.prf (mv openvario.prf clubprofile.prf)
4. Lock the profile with the following command:
`chattr +i clubprofile.prf`
5. Unlock the profile with the following command:
`chattr -i clubprofile.prf`

X. Settings in XCSoar

1. Click Config. -> Devices, select a free device from A, B, C or D and select Edit.
2. You should assign the connections so that the Flarm is above the FreeVario.
3. Select the port, where the electronical vario is connected with (e.g. ttyS1).
4. Baud rate is 115200, Driver is FreeVario, confirm with OK and Close.

XI. Settings for the stick remote control

In order to be able to use the STF switch to control the FreeVario in automatic mode, a few settings must be made. The changeover from Vario to STF, or vice versa, then works as follows:

1. priority has the toggle switch of the FreeVario. If STF or Vario is selected with it, the FreeVario is set to the respective mode. At the moment, the FreeVario does not send any information to the OpenVario to change the tone. To do this, we have to make a few more attempts. If the toggle switch of the FreeVario is on automatic, the 2nd priority comes into play.
2. The stick remote control has priority. A short click activates Vario, a long click STF and a double click the automatic mode. The mode set with the stick remote control is sent to the FreeVario and to the OpenVario to change the tone. If the stick remote control is also in automatic mode, the third priority is active.
3. Depending on how the switch in the FreeVario is set on the circuit board, XCSoar or a switch connected to the flaps has priority. Here, too, only the FreeVario is being switched at the moment. A command to the OpenVario to switch the sound does not work at the moment.

Important!! Sets in XCSoar OpenVario on device A and FreeVario on device B. Otherwise it won't work !!

1. The image of the FreeVario page must be newer than 22005 on the OpenVario. How to do this is given in Chapter V.
2. On the FreeVario, the ESPs of the display board and the soundboard must be updated to at least version 1.1.3. How to do this is in Chapter III.
3. If you are using the original openvario.xci file from the image and have not changed it to sort the menu or something similar, you do not need to do anything here. Be careful, if you import a backup, this file will be overwritten in the old version. Then you should change the file in your backup. If you use your own file, you have to change your file as well. To do this, the following lines:

```

mode=default
type=key
data=V
event=SendNMEAPort1 POV,C,VAR
event=StatusMessage Vario Mode

mode=default
type=key
data=S
event=SendNMEAPort1 POV,C,STF
event=StatusMessage Speed to Fly Mode

```

Change to:

```

mode=default
type=key
data=V
event=SendNMEAPort1 POV,C,VAR

```

```

event=SendNMEAPort2 PFV,REM,C
event=StatusMessage Vario Mode

mode=default
type=key
data=S
event=SendNMEAPort1 POV,C,STF
event=SendNMEAPort2 PFV,REM,S
event=StatusMessage Speed to Fly Mode

mode=default
type=key
data=A
event=SendNMEAPort2 PFV,REM,A
event=StatusMessage Automatic Mode

```

4. If you use a variant of the stick remote control with a toggle switch, you can skip the remaining menu items. If you have a version with a button, you have to update your stick remote control.
5. How it works, Stefan Langer has published here in a video:
<https://youtu.be/vwJqPAwFBIU>
6. To do this, install the Arduino IDE on your PC (<https://www.arduino.cc/en/software>)
7. Then download the file bounce.zip in our download area, unzip it and save the unzipped folder in the library directory of the Arduino IDE. Where you can find the library directory depends on the operating system on your computer. It's best to google it.
8. Then download the file Stick_Remote_Control_FV.zip from us and unzip it
9. Opens the Stick_Remote_Control_FV.ino file
10. Click on Tools in the Arduino IDE and select the Arduino Leonardo at Board
11. As the port you choose the one that came last when you connected the stick remote control to the computer via the USB cable
12. Click on “Upload” and that's it

XII. Installation of the new OpenVario menu from Kedder

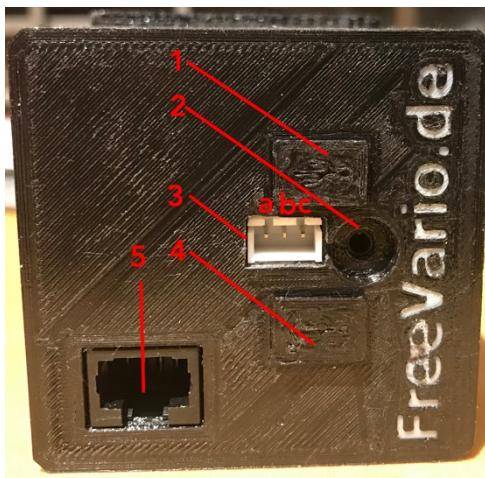
13. The OpenVario must be connected to the Internet and a keyboard must be connected.

14. Start OpenVario and go to the OV menu. Select "Exit to the shell" and confirm with Yes.
15. Enter the following command there:
 - wget <https://raw.githubusercontent.com/kedder/openvario-shell/master/scripts/openvario-shell-install.sh> -O - | sh
16. Use ESC to return to the main menu and select "Restart" and confirm with Yes. The new menu is now used.
17. If the menu is to be uninstalled again, use the command:
 - wget <https://raw.githubusercontent.com/kedder/openvario-shell/master/scripts/openvario-shell-uninstall.sh> -O - | sh
18. The menu can be updated via the menu itself or using the shell. The following commands are required in the shell:
 - opkg update
 - opkg upgrade

XIII. Update maps, airspaces etc. with the new menu from Kedder

1. Create a USB stick with your backup as described in Chapter IV, Variant 2
2. Put the stick in your PC and go to the directory
/openvario/backup/home/root/.xcsoar
3. Put the files you want to update here
4. Insert the stick into the OpenVario
5. Start the OpenVario and select “Applications” -> “Backup” -> “Restore” from the menu
6. All files that are newer than those on the OpenVario are updated

XIV. Installation



1. Micro-USB port for updating the soundboard
2. 2.5mm jack socket for external speaker
3. Plug for speed-to-fly switch/flap switch and PTT switch
 - a. PTT
 - b. STF switch
 - c. STF switch
4. Micro-USB connection for updating the display board
5. IGC plug for OpenVario

1. Slide the two hollow screws over the axis of the encoder or potentiometer and screw them to the FreeVario
2. It is essential to shorten (3mm + instrument panel thickness) the two M4 screws before assembly and check for correct length, otherwise the display will be destroyed!
3. Fasten the FreeVario with the M4 screws
4. Tighten the two rotary knobs. Make sure that the encoder button has enough space to the rear for the push button to work
5. Attach the cover caps to the rotary knobs
6. Connect FreeVario to OpenVario with an IGC-compliant cable
7. If necessary, connect the flap switch (closed, if the vario is to be set to STF) and PTT button to the three-pin plug
8. To avoid reception problems, the antennas of other devices should be installed as far away as possible from the FreeVario

XV. Disclaimer

The vario gauge is not EASA or FAA certified. Note that this is exactly the same for all the fancy commercial electronic varios and flight computers you are probably used to.

If would you like to know, if you are allowed to use it legally at your glider, ask this question to your airworthiness inspector. Can you legally install any other big brand (as well, non certified) vario gauges? If yes, you should also be able to legally install this vario gauge. If you are under EASA rules see CS-STAN, standard change CS-SC402a.

This instrument is a craft project and not an aviation-approved instrument. I explicitly point out that the use is made under exclusion of any warranty at your own risk!