

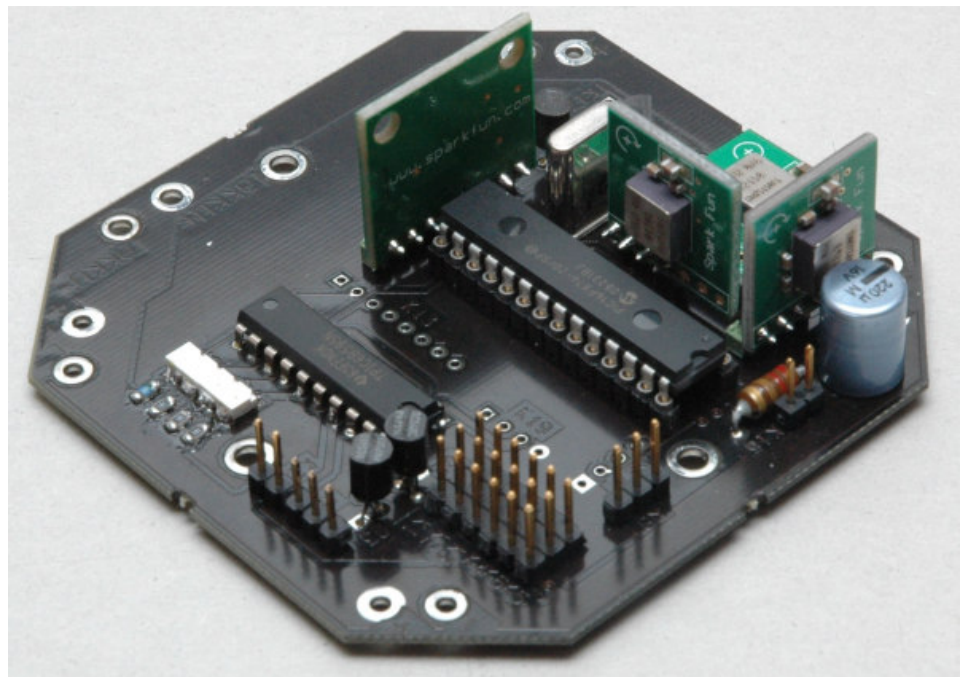
# Operation manual

for

## Test software

designed for

### UFO mainboards 3.0 and 3.1



**U . A . V . P .**  
Universal Aerial Video Platform  
developed by Ing. Wolfgang Mahringer

**Version 1.00 of 4-4-2007**

# What's it all about?

This test software for the UFO mainboard is intended for diagnoses of the PCB. It is encouraged that you use this software to find out if your newly build UFO PCB is functioning correctly.

Additionally, this tools can be of invaluable help for locating problems, especially if after a serious crash something does not work anymore ☺

This test software is **open source** and can be downloaded in the net kann im Internet at <http://www.opensourcequadrocopter.de> !

If you have any problems, first refer to the troubleshooting section at the end of this document. If nothing helps, you can sign up to the forum on OpenSourceQuadrocopter.de to get expert help. You can also mail the author, see first page for email address.

## Features

The design goal was to create a **test software** that allows to test all the hardware components on the mainboard:

- allows use on a newly built mainboard without any external connections and on ready-for-takeoff-UFOs
- updateable by bootloader
- reading of analogue values (gyros, battery voltage, reference voltage)
- checking of receiver impulses
- checking polarity of receiver impulses polarity
- test of the LEDs (onboard and external) and the beeper
- test the servo outputs (ESC and camera servos)
- test the linear sensors
- diagnosis of the sensor bus (I<sup>2</sup>C)
- diagnosis of the compass module

Some of the tests are not offered on the 3.0-mainboards (the green PCB) because its hardware is different and so do not permit some tests.

If you use this test software on a mainboard with ESC and motors connected, **please be very careful!** The servo output test, for instance, drives the ESCs up, so your UFO may take off totally uncontrolled!

**It is good practice to remove the propellers or, at least, fixate your UFO firmly to make a takeoff impossible.**

## What do I need?

- an UFO mainboard ☺
- suitable battery pack or, even better bench power supply, set to about 12 volts.
- RS232 cable to connect the mainboard to your laptop or computer
- the software „UAVPset“
- a terminal software (e.g. Hyperterminal or Minicom)

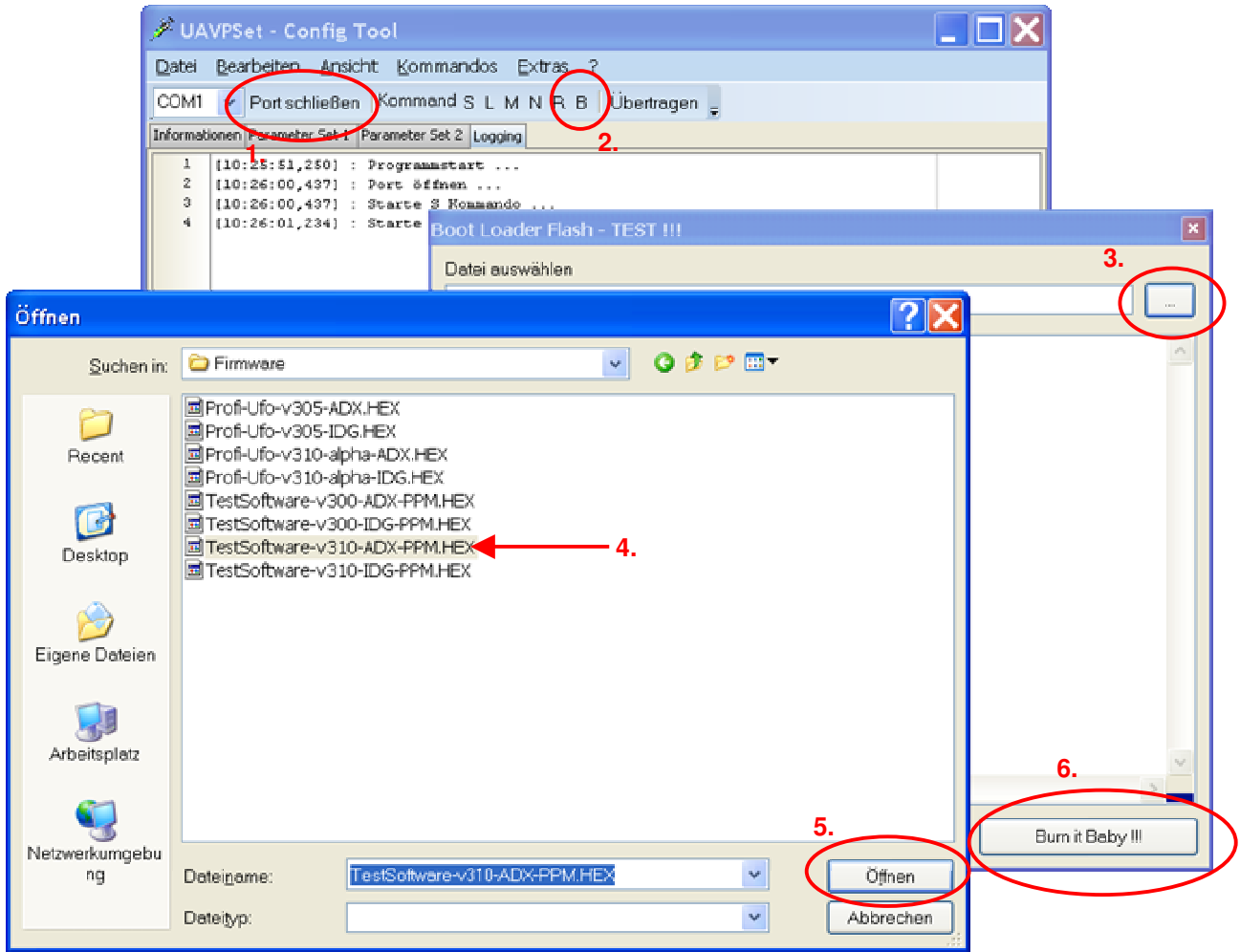
## Programming a PIC

In order to put the software on a PIC processor (e.g. to “burn” it), you’ll need a PIC programmer. There are loads commercial available programmers and even more easy to build do-it-yourself circuits on the web.

You can get the necessary hex files in the download area at <http://www.opensourcequadrocopter.de>

# Flashing a PIC

If the PIC does already contain a UFO software (test or flight software), you can use the UAVPset tool to update processor, eliminating the need of a programming device. Unfortunately this tool is only available in German, so please follow these picture to get it done:



Connect up your mainboard using the RS232 cable and power mainboard up. Then start UAVPset.

1. Port öffnen (open serial port, you may select the proper port before)
2. click on the "B" to select the bootloader
3. select "Browse"
4. select your HEX file, be careful to get the correct one (!)
5. click „Öffnen“ to open it
6. start the update by clicking on „Burn it Baby“

**Important note:** While updating, do not disconnect the serial cable nor disconnect power to the mainboard!

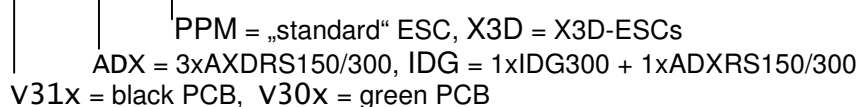
## Even more important:

Please check before updating if you are using the correct HEX file for your board. A change from V3.0 to V3.1 or vice versa is not possible using UAVPset!

If you mock things up, you'll have to burn the PIC with a PIC programmer!

This is how the HEX files are named:

TestSoftware-v310-ADX-PPM.hex



# Test software

At first, you'll need to have your terminal software configured to 38400 baud, 8 data, 1 stop, no parity and no handshake.

Connect the RS232 cable to your mainboard and to your computer.

If you power the board up, you should see a message like this:

```
U.A.V.P. TESTSOFTWARE V3.10 Copyright (c) 2007 Ing. wolfgang Mahringer
U.A.V.P. comes with ABSOLUTELY NO WARRANTY
This is FREE SOFTWARE, see GPL license!

Profi-Ufo TEST V3.10 ready.
Gyro: 3x ADXRS300
Linear sensors ONLINE
Compass sensor ONLINE

# _
```

The blue LED should be lit, also the red or green LED.  
Blue LED on: test software running (that's a good sign ☺)  
Green LED on: a valid receiver signal is detected  
Red LED on: no (or invalid) receiver signal

There is a help function (simply type "?") which lists the available commands:

```
# ?
Commands:
A....analog voltages
B....call Bootloader
C....Compass test
I....I2C sensor bus scan
L....Linear sensor test
N....negated Receiver-PPM
R....Receiver test
S....Show setup
V....Servo test (CAUTION!)
1-8..Power output test

# _
```

by entering the letters (upper or lower case) you can start the test run.

## Note:

This test software does not use or modify the flight parameters that may be stored in the nonvolatile memory in the PIC chip.

You can play around with this software as you like, it does not harm the parameter settings.

Wrongly entered characters are echoed, but simply ignored.

# Analog voltages

This test measures all the analogue signals available to the microprocessor:

```
# a
Analog channels test:
Ub: 5.14 volt
Ur: 2.622 volt
Un: 2.511 volt
Uy: 2.435 volt
Uz: 3.783 volt
# _
```

Ub = Battery voltage. Because of tolerances at the resistor divider and the accuracy of the 5 volts line this measurement may not be very accurate, but the reading should be 0.2 volts near its real value.

Ur, Un, Uy = Voltages of the gyro modules roll, nick and yaw. An ADXRS150/300 module should read about 2.5 volts, an IDG300 module (only at roll and nick!) should read 1,5 volts. Even on a IDG300 equipped mainboard, the yaw gyro is always a ADXRS150/300 module, so the Uy reading must always be in the 2.5 volts area.

If you tilt or turn the mainboard and conduct an "a" command at the same time, so should see a good increase or decrease of the voltage reading.

Uz = Reference voltage of the 3,6 volts line (only available on the black V3.1 PCBs!). The voltage should be in the 3.5 to 4.0 volts range.

# Compass test

This test initializes the compass module and then tries to read the compass heading (this is only possible on the black V3.1 PCBs).

```
# c
Compass device test
299 deg
# _
```

If the compass device works properly, the current heading with respect to "north" is output. If you now turn the PCB a bit, and do the "c" command again, the displayed value should change accordingly, e.g.:

```
# c
Compass device test
316 deg
# _
```

## I<sup>2</sup>C sensor bus scan

This one scans the I<sup>2</sup>C sensor bus for so called I<sup>2</sup>C slave devices, which is only possible on the black V3.1 PCBs (the older, green V3.0 PCBs do not have a sensor bus).

To do this, it tries a dummy write for all possible I<sup>2</sup>C slave addresses (from 0x10 up to 0xF6) and checks if there is a reaction at the acknowledge bit. All positive answers are listed.

This function is very handy for modders who want to connect their own sensors or devices to the board.

```
# 1
I2C bus scan running, listing slave addresses...
0x42
001 device(s) found
# _
```

Address 0x42 in this example is the compass sensor.

## Linear sensor test

This one tests the accelerometer (or linear) sensor (LIS3LV02DQ), which is only possible on the black V3.1 PCBs.

While the older, green V3.0 PCBs do support this linear sensor, it shares some lines with the serial COM port, making it impossible to access the sensor while the COM port is active (which always is the case in this test software).

The test outputs the status byte and acceleration values of all 3 axis.

```
# 1
Linear sensor test:
S:0xFF
X:+ 0.057 G
Y:- 0.103 G
Z:+ 1.050 G
# _
```

S = Status byte, at least the 3 LSBs must be set (0x07)!

X = in roll direction

Y = in nick direction

Z = up/down, notice the earth's gravitation ☺

Try to move or tilt the mainboard a bit and see how readings change!

# Receiver test

This test is the most important one, because 99% of the problems (UFO mainboard does not get "green") have to do with the transmitter or receiver and its programming and wiring.

```
# r
Last RX values received:
1:0x0113
2:0x017A
3:0x0177
4:0x0177
5:0x0111
6:0xFFFF FAILED
7:0xFFFF FAILED
P:15.208 ms
received values are OK!
# _
```

1: to 7: shows the length of the PPM receiver signal pulses for channel 1 to 7. 0x0100 means, the pulse is 1 ms long (the minimum), 0x0200 means 2 ms long (the maximum).

In order to get the test software (and the flight software too) to "green", it must comply with some requirements:

1. Channel 1 to 5 must be correctly read and be in range of 0x0100 to 0x01FF.
2. The so called „synchronisation gap“ must be no less than 5 msec and not more than 18 msec.
3. The channels 6 and 7 may miss, the camera servo outputs are not used in this case.

If any channel does not comply with the timing requirement, it is marked with „FAILED“.

The green and red LED on the mainboard do immediately show if the received signal is valid or not. You don't need to conduct the test over and over.

The example shows that channels 1 and 5 are near the minimum, while channels 2, 3 and 4 are midways (ca. 0x0180).

Please try all extreme stick positions on your transmitter, also do not forget channel 5 to 7! The red LED should NEVER light up! If it does, check servo limits on the affected channel, set it to not more than +/- 100%, but this is highly dependant of the transmitter model.

If there is no receiver connected at all, or your transmitter is not turned on, the software can't read any transmitter pulses. In that case, there is only a message printed:

```
# r
(no new values)
# _
```

# Negated receiver PPM

Some quite exotic receiver models output their pulses inverted, especially if imported from the U.S. In that case, the software is of course not able to read a good signal.

You can modify your board to suit this type of receivers:

- turn diodes D1 to D4 180° degrees
- remove resistor R9 and connect one side of that resistor to the diodes common cathodes and the other side of the resistor to +5 volts
- solder a 4k7 resistor on transistor Q1 from pin 1 (emitter) to pin 2 (base)

Then switch the polarity of the input signal conditioner:

```
# n
Setting negative receiver PPM
# _
```

Now you can try a “r” command if you get a good signal now.

Every start of the “n” command reverses the polarity which the software uses to detect the pulses. The selected setting is not stored anywhere. If you boot up the test software, “positive PPM pulses” are used by default.

## Show setup

This command show the actual configuration of the board:

```
# n
Profi-Ufo TEST v3.10 ready.
Gyro: 3x ADXRS300
Linear sensors ONLINE
Compass sensor ONLINE
# _
```

The output depends on the hardware options you choosed.

## Servo test

This command allows testing of all the servo outputs.

```
# v
Servo test running...
```

The ESCs connected on K1 to K4 are powered up slowly

The camera servo outputs go at first to their maximum position, then gradually move to the opposite extreme.

At the end of the test run, the ESCs are put to their minimums (throttle cut off), and the camera servo outputs are put to their middle positions.

**Important Note:** If you have real ESCs and motors connected, be careful! The motors are accelerated up to full power! You can unmount the propellers or fix your UFO firmly to avoid an uncontrolled takeoff!

The test can be interrupted any time by pressing any key. The motors will then stop at once.

# Power output test

This one lets you test the LEDs and the beeper. The selected power output line is activated 5 times. This allows to check if you mounted the LEDs correctly.

```
# 2
Power output test
2:Blue LED

# _
```

Output 2: blue LED  
Output 3: red LED  
Output 4: green LED  
Output 6: yellow LED  
Output 7: beeper (on mainboard V3.0, green PCB)  
Output 8: beeper (on mainboard V3.1, black PCB)

Unlisted outputs may or may not be connected to anything, so it depends what you see ☺

## Call Bootloader

This command is not a test ☺

It is used by the UAVPset tool to call the bootloader in order to perform a software update.

You should never enter the bootloader manually.

If you did, you need to cycle power to get the board working again.

As soon as the bootloader is ready to receive the update, the yellow LED lights up.

# Troubleshooting

## **Blue LED does not light up, no output at the terminal software**

May be there is no 5 volts power. This happens if you disconnect all the ESCs.

You should at least have the ESC on K1 connected, for safety, unmount the propeller, or use an ESC without any motor connected.

## **Blue LED is on, but no output at the terminal software**

Check if you have connected the RS232 cable wrong way round (connector K7). Don't worry, a wrongly connected RS232 cable does not harm anything.

## **Blue LED does not light up, but output at the terminal software seems OK**

On a V3.0 mainboard (green PCB) you must connect battery to connector K16 and jumper J1 must be jumpered correctly to allow power to the LEDs.

On a V3.1 mainboard (black PCB): check polarity of the mounted LEDs. Are the LED current limiting resistor soldered properly? The large copper areas on the PCB makes good soldering joints a bit difficult!

## **Not the blue, but some other LEDs light up, while output at the terminal software seems OK**

It seems that you have mounted the coloured LEDs wrongly.

Use the power output test to find out which ones are wrong.

# **Version history:**

V1.00 04-04-2007 First version of the document