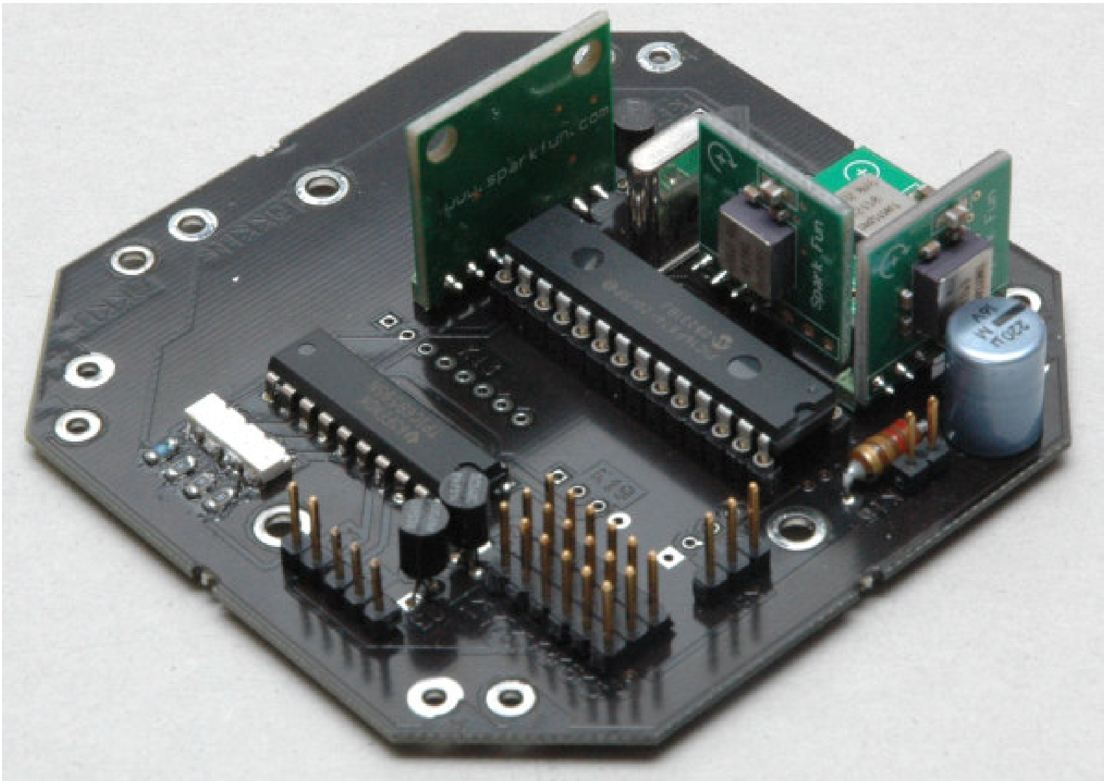
Manual

For building and operating

A professional Hovering Platform



U.A.V.P.

Universal Aerial Video Platform

developed by Ing. Wolfgang Mahringer

Version 3.14, December 30, 2007

English translation: Thomas Bögel (bogomir@opensourcequadrocopter.de)

Features

The target for the development was to build a **universal hovering platform**, including the following features:

- single-board-electronics, additional components are motors, motor controllers, R/C system and a fuselage/frame
- easy power cabling
- interference-proof design, no special intricate skills needed.
- programmable signal update (up to 250Hz)
- custom settings for different motor/controller/propeller combinations
- 2 additional R/C channels for controlling servos (camera angles)
- 7 Hi-Power LED-drivers (each up to 500mA)
- acoustical signal for low-battery-warning, signal failure and locator beacon
- gyros: 3 Gyro modules ADXRS300/150 (recommended) or 1x ADXRS300/150 and 1xIDG300
- parameters programmable via RS232 or R/C-Tx
- optional digital acceleration sensor LIS3LV02DQ
- optional digital compass sensor HMC6352
- optional digital barometric sensor SMD500
- optional GPS-support
- automatic parachute recovery system in case of loss of control.

Options

To allow some degree of customizing, a number of options can be selected or omitted. In all cases it is possible to add or remove these options later. Some of the options do however have influence on others.

Parts needed for any option are mentioned separately in each building step.

- **Base Components**, must always be installed
- Gyro **ADXRS300/150** (not with Gyro IDG300)
- Gyro **IDG300** (not with Gyro ADXRS300/150)
- **Magnet sensor**
- **Linear sensor**
- **compass sensor**
- **barometric sensor**
- **Parachute trigger** (needs linear sensor)
- Servo-connectors for **camera control**
- **GPS-Module** (no programming via RS232)

Note: In contrary to the earlier Version 3.0x (green PCB) this version can only be operated with speed controller which have their negative supply connected to servo GND!
If unsure, please check with an ohmmeter to avoid short circuits that can damage controllers and electronics!

List of Parts

Basic R/C equipment:

4x Brushless-Motors
2x clockwise turning propellers (i.e. EPP1045 or other, matching the used motors)
2x counter-clockwise turning propellers (i.e. EPP1045 or other, matching the used motors)
4x brushless-controllers, matching the power requirements of the used motors i.e. Arkai R-RBR-4 (18 amps), the YGE I-series or Holger's BL-Controller. These ESCs give great performance, but other ESC models can also be used.
1x. 5-channel R/C receiver (7-channel recommended) with quartz
Alternatively a receiver with a PPM composite output may be used.

Electronics Parts:

To save both space and weight, most of the electronic components are SMD packages. However, the parts used here are easy to work with a standard soldering iron using a pencil-type tip, and steady hands. Where possible part numbers from Conrad.com have been stated in brackets, as well as the package Type or sizes. Since they are all standard parts and packages, any other source can be used.

Option „Base Components"

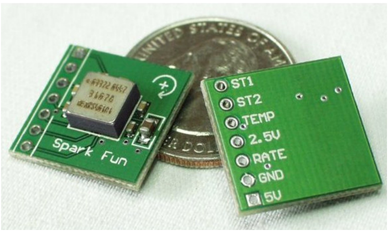
1 x	Printed circuit board, to be purchased at www.lipoly.de	
1 x	U1	PIC16F876 or 876A (DIL28, 165117) programmed with the current HEX file. A pre programmed µC can be obtained from the author or www.lipoly.de
1 x	U2	TPIC6B595N (DIL20)
1 x	U3	TL431 voltage regulator (TO92)
1 x	X1	Quartz 16MHz (HC49U)
4 x	R1,R2,R3,R23	SMD resistor 10kΩ 5% (0805)
11 x	R6…R8,R24…R26, R21,R10…R13	SMD resistor 4,7kΩ 5% (0805)
2 x	R4,R9	SMD resistor 1kΩ 5% (0805)
1 x	R5	SMD resistor 2,2kΩ (0805)
1 x	R22,R28	SMD resistor 68Ω 5% (0805)
3 x	R29,R30,R31	SMD resistor 180Ω 5% (0805)
1 x	R16	SMD resistor 100Ω 5% (0805)
4 x	D1,D2,D3,D4	SMD-Diode LL4148 (SOD80) note orientation on PCB!
1 x	D5	SMD LED blue 30mA, PLCC2
1 x	D6	SMD LED red 30mA, PLCC2
1 x	D7	SMD LED green 30mA, PLCC2
1 x	D8	SMD LED yellow 30mA, PLCC2
3 x	Q1,Q2,Q3	NPN transistor BC548B or similar (TO92)
4 x	C1,C2,C6,C7	SMD capacitor 100nF (0805)
2 x	C4,C5	SMD capacitor 22pF (0805)
2 x	C8,C9	SMD tantalum capacitor 1µF
1 x	C3	polarized capacitor 100µF ≥10V, radial package, note orientation on PCB!
1 x	L1	inductive coil 100 to 220 µH, 0,1A, axial package
1x	all K	36pin male connector RM2,54 straight, gold galvanized, detachable
2x	all K	12pin female connector RM2,54 straight, detachable
2x	for U1	IC socket DIL14 (2 mounted in a row to make DIL28)

Option „LEDs“

Everybody wants something different ☺
Any type of LED will do. The 4 necessary LEDs are already mounted on the PCB (contained in option “base components”).
If you want to use additional LEDs, you can use connector K13, where all 7 power LED outputs are available. You do have to calculate the value for the current-limiting resistor for each LED (check LED nominal current and battery voltage). The positive leads (anodes) of all LEDs are common to Pin 8 of K13. The negative leads (cathodes) of each LED are connected to pins 1 to 7 respectively, the signal running through the individual LED’s current limiter resistors.

In addition, any type of LED can be used to denote the „front“ of the quadrocopter. Connect this LED to K15. A current limiting resistor (R16) is already installed (100kohms). This LED is fed through the BEC in all of the options!
Tipp: Use a red 8mm „Jumbo-LED“ for this.

Option „Gyro ADXRS300/150“



A stable hovering control requires 3 gyro sensors: Nick, Roll, and Yaw. Sensors by Analog Devices are great quality, but not usable without special soldering equipment. Therefore we are using sensors already mounted to break-out-PCBs.

The ADXRS150 is cheaper, and also usable.

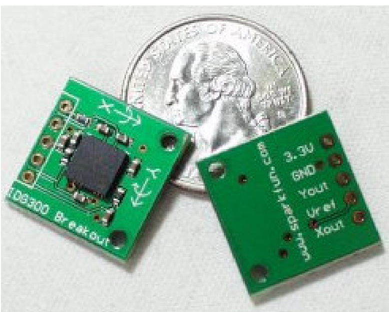
Picture: Gyro module ADXRS300 from Sparkfun/USA

Roll (K9) and Nick (K10) are connected to the PCB via 90°-type 1x4 pin outs, the yaw sensor uses a 1x4 straight pin out (a sufficient number if pins is supplied in the base package). The pins also provide ample structural stiffness, so no further soldering or glue is required. But you can use some hot-glue to improve stiffness even more.

1x	K9, K10	1x8 90° pin out 1/10”

Important Notice:
The ADXRS300 provides far stronger signal levels as compared to the IDG300. This provides significantly more stable controlling, especially in control code with high integral values (providing AoA rather than acceleration). Therefore, we recommend choosing the ADXRS over the IDG300, although the higher price!

Option „Gyro IDG300“



A stable hovering control requires 3 gyro sensors: Nick, Roll, and Yaw. Sensors by IDG are great quality, but not usable without special soldering equipment. Therefore we a re using sensors already mounted to break-out-PCBs.

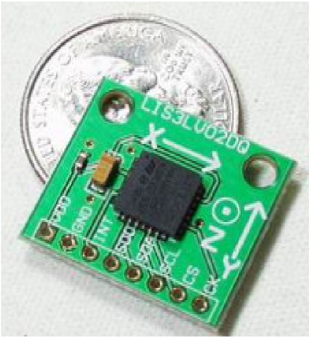
However, we recommend choosing the ADXRS300 sensor module.

Picture: Gyro module IDG300 from Sparkfun/USA

The IDG300 break-out PCB is connected to the PCB (K18) via straight 1x5 pin outs; the yaw sensor uses a 1x4 straight pin out (a sufficient number of pins is supplied in the base package). The pins also provide ample structural stiffness, so no further soldering or glue is required

Important Notice:
The ADXRS300 provides far stronger signal levels as compared to the IDG300. This provides significantly more stable controlling, especially in control code with high integral values (providing AoA rather than acceleration). Therefore, we recommend choosing the ADXRS over the IDG300, although the higher price!

Option „linear acceleration sensor“



For even more hovering stability, a 3-DOF-acceleration sensor LIS3LV02DQ is required, in addition to the gyro sensors.

This sensor module alone is not usable without special soldering equipment. Therefore we a re using sensors already mounted to break-out-PCBs.

Picture: acceleration sensor LIS3LV02DQ from Sparkfun/USA

The LIS3LV02DQ break-out PCB is connected to the PCB (K14) via 90° 1x8 pin outs (a sufficient number of pins is supplied in the base package). The pins also provide ample structural stiffness, so no further soldering or glue is required

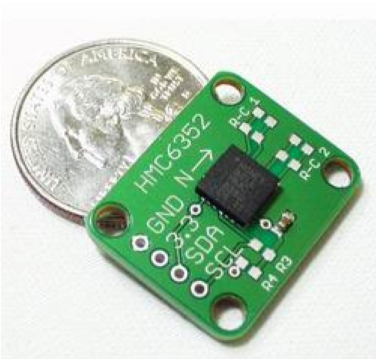
3 x	R14,R15,R17	SMD resistor 4,7kΩ 5% (0805)

Option „parachute trigger“

One of the special features of the acceleration sensor is that it produces a signal whenever a certain acceleration along the Z-axis is detected. Even in case of an unwanted loop the trigger signal is provided at the right moment – when the UAVP is in a horizontal position, thus the UAVP will not fall into it’s own parachute.

1 x	R20	SMD resistor 1kΩ 5% (0805)
2 x	R18,R19	SMD resistor 10kΩ 5% (0805)
1 x	Q4	NPN transistor BC548B or similar (TO92)

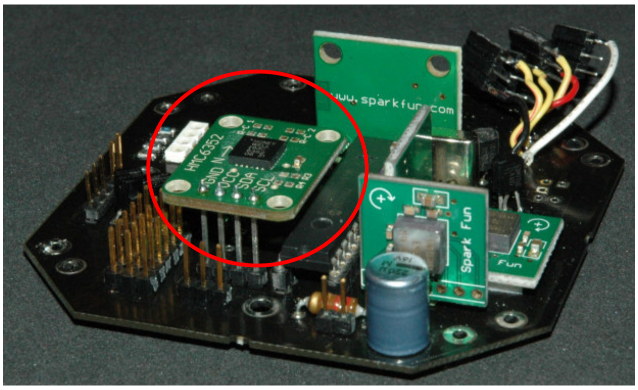
Option „compass sensor“



A compass sensor allows absolute flight direction measurement and a perfectly long term stability of the yaw axis

This sensor module alone is not usable without special soldering equipment. Therefore we a re using sensors already mounted to break-out-PCBs.

Picture: Compass sensor HMC6352 from Sparkfun/USA



Mounting the compass sensor:

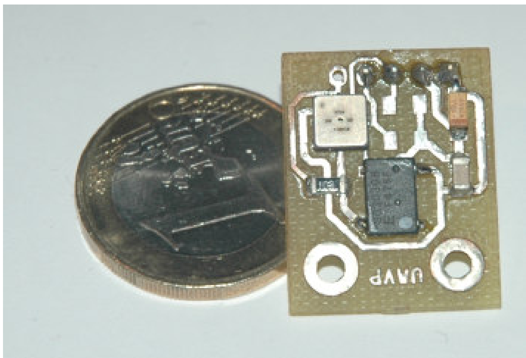
Mounting is done best using special long-pin header („wire-wrap“), 4 pin.

Be careful when mounting the sensor. It must still be possible to plug in the servo cables easily. However, mount the sensor high enough to allow changing the PIC if necessary.

This sensor is supported from version 3.12 on. There is no special activation necessary

Be careful that no magnetized object are located near the magnet sensor (steel rod antenna etc)

Option „barometric sensor“



A barometric sensor allows automatic flight level control (the UAVP will maintain the current altitude without pilots help)

This sensor module alone is not usable without special soldering equipment. Therefore we a re using sensors already mounted to break-out-PCBs.

Picture: Barometric sensor SMD500 from lipoly.de

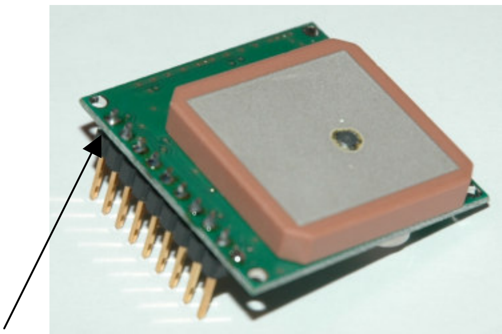
Mounting the barometric sensor:
The barometric sensor can be mounted like the compass sensor, using a long-pin header („wire-wrap“), 4 pin. If you have compass and barometric sensor, you can mount them one on top of the other. All 4 pins are to be wired in parallel.
Be careful when mounting the sensor. It must still be possible to plug in the servo cables easily. However, mount the sensor high enough to allow changing the PIC if necessary.
This sensor is supported from version 3.14 on. There is no special activation necessary

Option „camera tilt servos“

Two standard servo connectors are available (K5 and K6) supplying a positive PPM signal. To utilize this feature, a minimum of 7 channels on your R/C Rx is required. These servos can be used to pan and tilt your camera.
Channels 5 and 6 define the wanted position for the camera. Movement of the platform will be corrected automatically.
Power supply for those servos comes from the aft controller only and is therefore separate from the other BEC clients.
Use 2 1x3 straight pin outs 1/10” supplied in the base package.

Option „GPS-Module“

You can connect a cheap GPS-module (conrad.com No. 989777) to the main board. The manual denotes the module as “USB-only”, but inside the cover some other connectors are available.
After Setting up the module using the PC-software supplied with the module, an RS232 interface can be added. To do so, remove the USB cable and replace it with a pin header as shown in the pictures. Connect the pins to K19 on the main board in the same sequence as on the module.



Pin 1

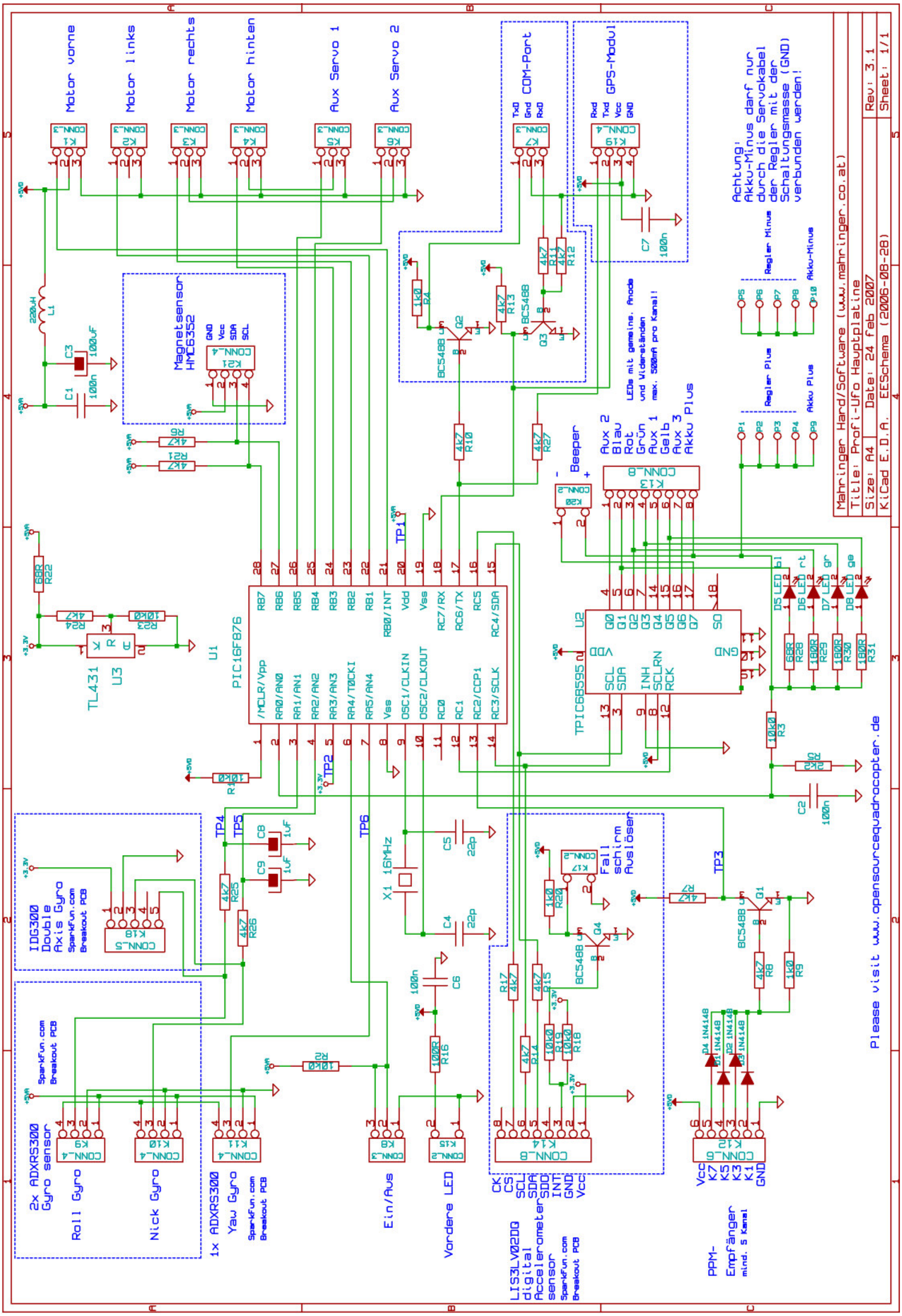
Pin 1 = RxD
Pin 2 = TxD
Pin 3 = +5V
Pin 4 = GND

1 x R27 SMD resistor 4.7kΩ 5% (0805)

GPS module setup:
(to be done)

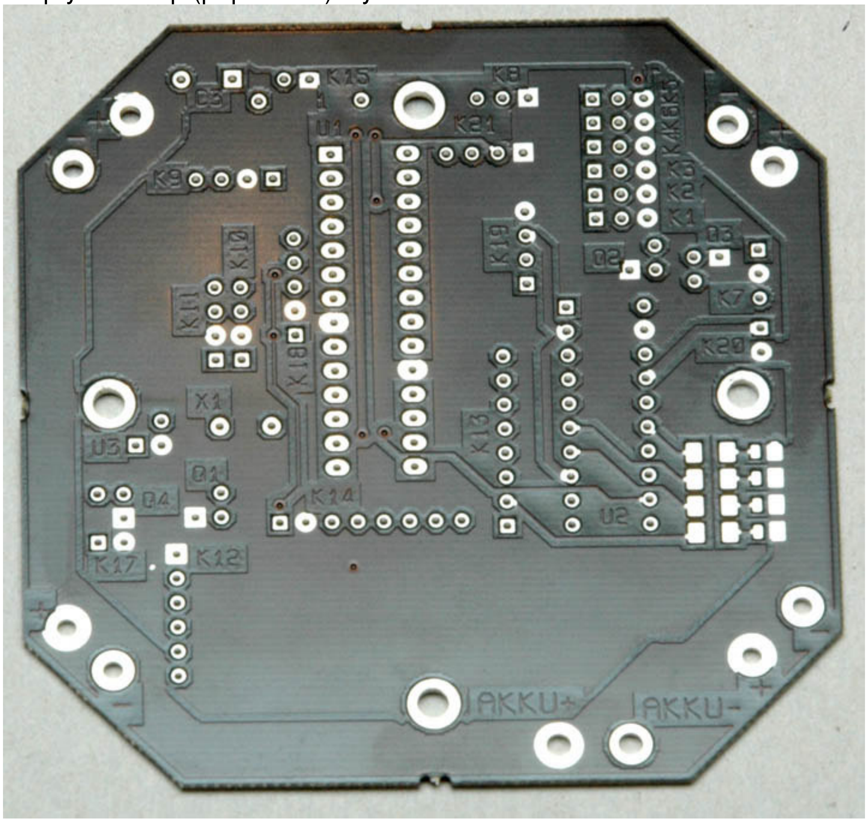
  wolfgang.mahringer@sbg.at

Schematic

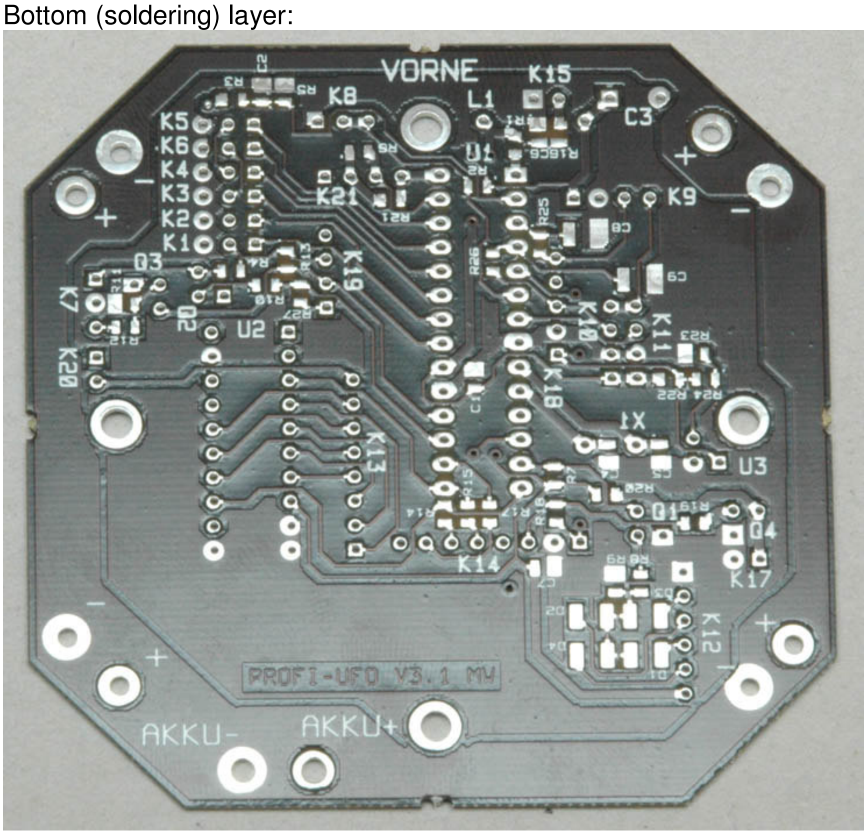

Motor vorne
Motor links
Motor rechts
Motor hinten
Aux Servo 1
Aux Servo 2
COM-Port
GPS-Modul
Achtung! Akku-Minus darf nur durch die Servokabel der Regler mit der Schaltungsmasse (GND) verbunden werden!
Magnetsensor HMC6352
PIC16F876
TPIC6B595
TL431
LEDs mit gemeinsam. Anode und Widerständen max. 500mA pro Kanal!
Beeper
Fallschirm Auslöser
IDG300 Double Axis Gyro SparkFun.com Breakout PCB
2x ADXRS300 Gyro sensor
Roll Gyro
Nick Gyro
1x ADXRS300 Yaw Gyro SparkFun.com Breakout PCB
Ein/Aus
Vordere LED
LIS3LV02DQ digital Accelerometer sensor SparkFun.com Breakout PCB
PPM-Empfänger mind. 5 Kanal
Please visit www.opensourcequadrocopter.de
Mahringer Hard/Software (www.mahringer.co.at)
Title: Profi-Ufo Hauptplatine
Size: A4 Date: 24 feb 2007 Rev: 3.1
KiCad E.D.A. EESchema (2006-08-28) Sheet: 1/1

Layout/Population

Empty PCB top (population) layer:

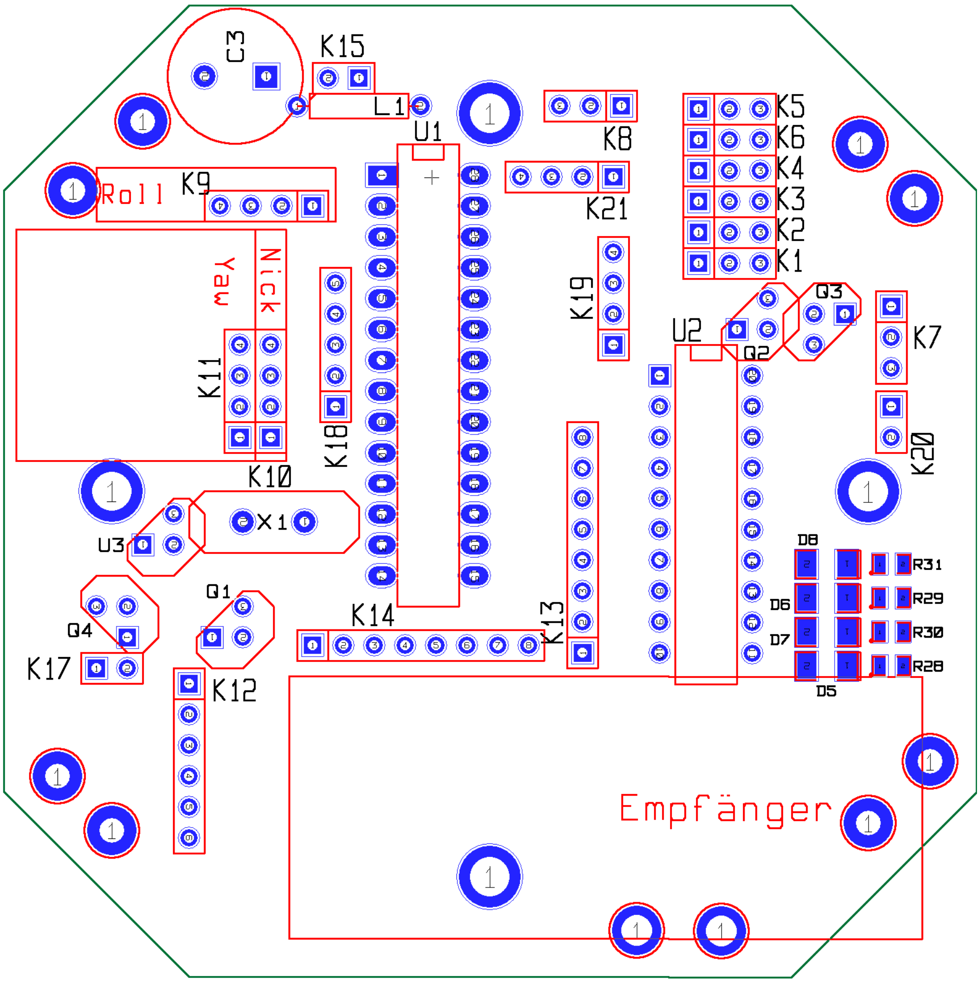


Bottom (soldering) layer:

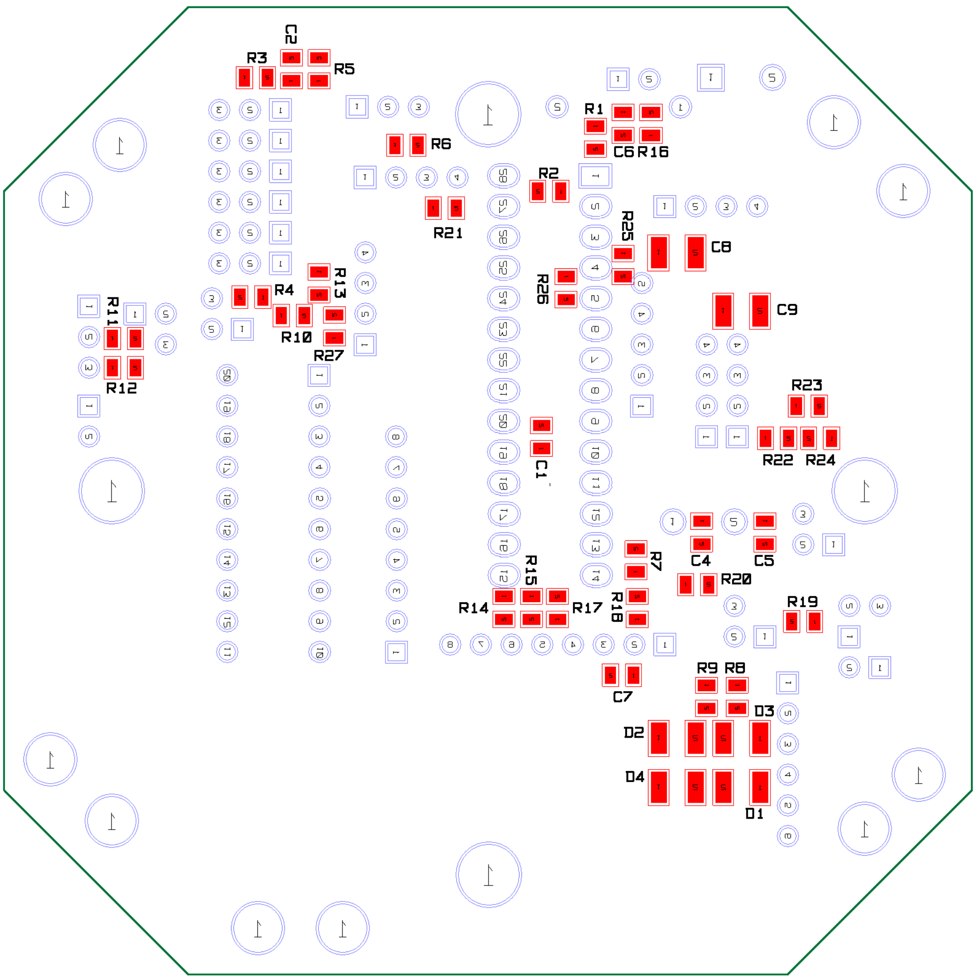


  wolfgang.mahringer@sbg.at

Top layer:



Bottom layer:



Notice:

These parts are polarized and have to be installed accordingly:

D1, D3
(cathode left facing inside)

D2, D4
(cathode right, facing to the edge of the PCB)

C3 (negative is pin 2)

ICs (U1,U2) feature a notch or dot to mark pin 1

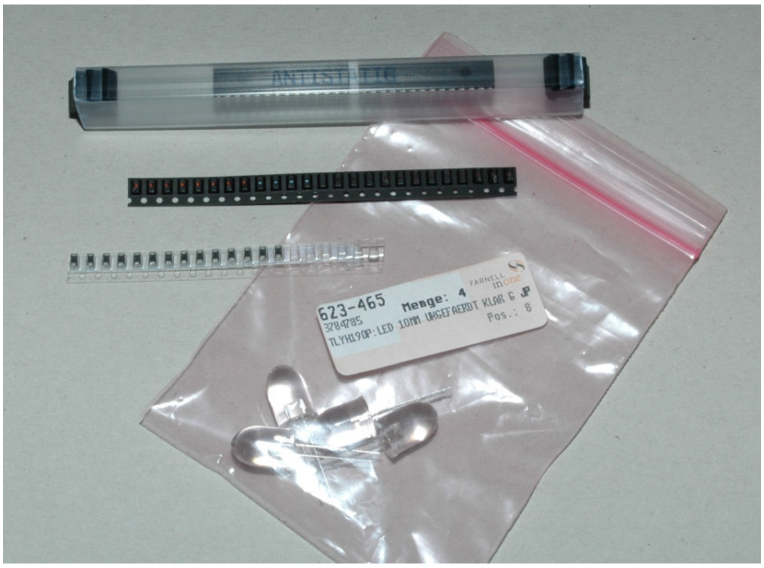
  wolfgang.mahringer@sbg.at

Important: Protect against static discharge!

Most components, like ICs, PIC and the sensors are sensitive for static discharge (ESD). Walking over a carpet alone may cause charges of several 1000 volts! A charge like this will not harm you, but it may destroy the parts mentioned above. Sometimes a discharge will not fry the complete part, but only cause damage. The part appears to function in the desired manner, but behaves erratically from time to time. It may draw more current, develop more heat, stop functioning altogether without any signs of failure before etc.

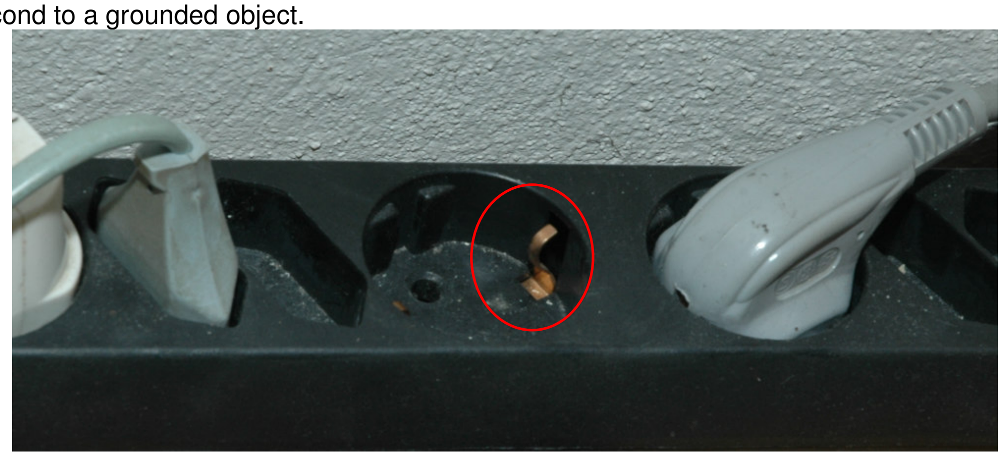
To save the expensive components from ESD damage, please follow these few simple rules. No special equipment is needed at all:

1. All parts are shipped in a <u>conducting plastic bag or tube</u>! The bags may not look like it, but they actually do eliminate ESD perfectly ☺
   Leave the parts in the bag until you need them.



2. <u>Keep your workplace neat</u>

3. <u>Before you begin working and after you leave the workplace:</u> Make sure that your body as well as any tools, such as wires, solder, tweezers etc. are discharged. Do that by grounding them. Touch each tool and your body to a grounded object, such as a heating radiator (not a painted part!), or the ground lead on an electric wall outlet, a faucet, etc. The human body is a good electrical conductor, due to its high content in water and salts. It is quite sufficient to touch any part and tool for about one second to a grounded object.



  wolfgang.mahringer@sbg.at

4. If you should feel a mild electric shock while touching any of the parts or tools, something is wrong! Touch a grounded object more often! Think about wearing other clothes and shoes (non-plastic)

5. Make sure your soldering iron has either an insulated or grounded tip. You can test this by checking with a test lamp. Test the tip of your soldering iron just after it was switched on, before it gets hot. The lamp must not light up – not even dim!

6. Sit still during work, do not wiggle around on your chair. From time to time, repeat step 3

7. Passive components such as diodes, resistors and capacitors are not sensitive to static electricity – the ICs and sensors are! Keep that in mind.

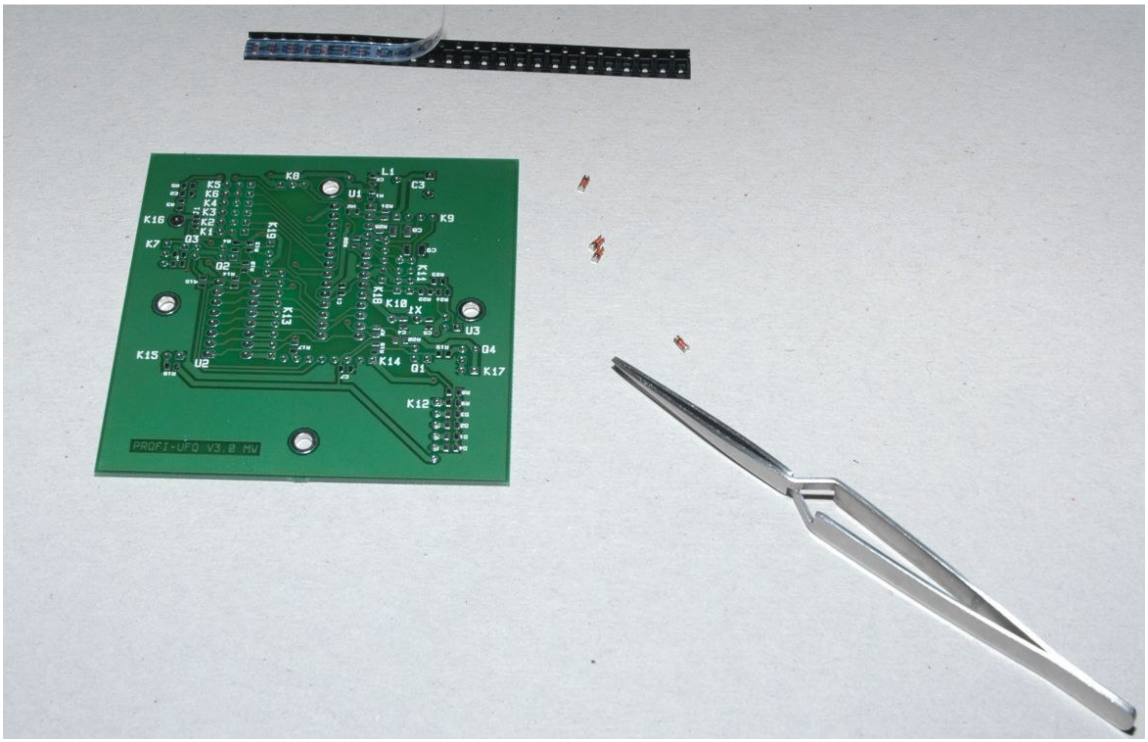
The finished PCB is not that sensitive to static electricity, due to the cross connections on the PCB. Voltage cannot build up, but will be distributed off quickly. However, try to touch the PCB only at the edges.

During installation the finished PCB into your frame, continue to obey all of the above steps.

To store the finished PCB prior to installation, keep it inside the provided plastic bag.

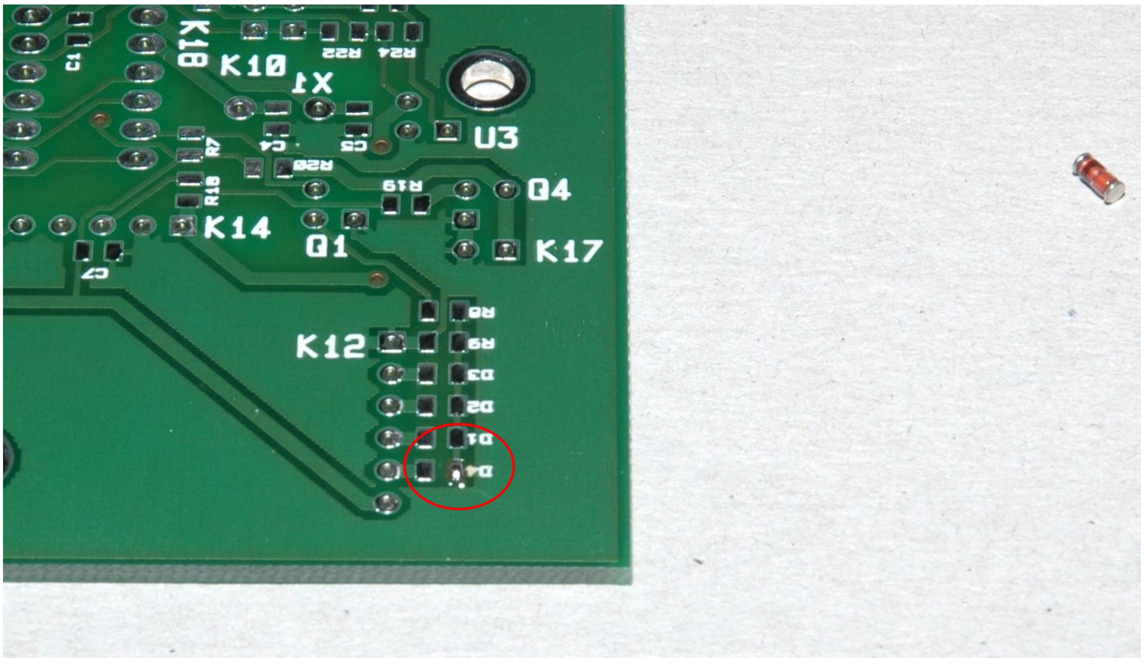
Some tips for soldering small SMD parts:

1. Prepare your Workplace. A pair of pincers is required to hold the parts while applying the solder. Inverted pincers are recommended (see picture below). Use a thin strand of solder, about 1mm in diameter. The PCB comes completely RoHS compliant (meaning you can continue working with lead-free solder). For people who are new to small SMD parts we recommend using standard solder (Sn60PBCu, or similar). It is much easier to work with and the joints are more durable. Set your soldering iron to. 300° - 350℃ (570-66℉) when using standard solder. 400℃ (750℉) with RoHS solder.

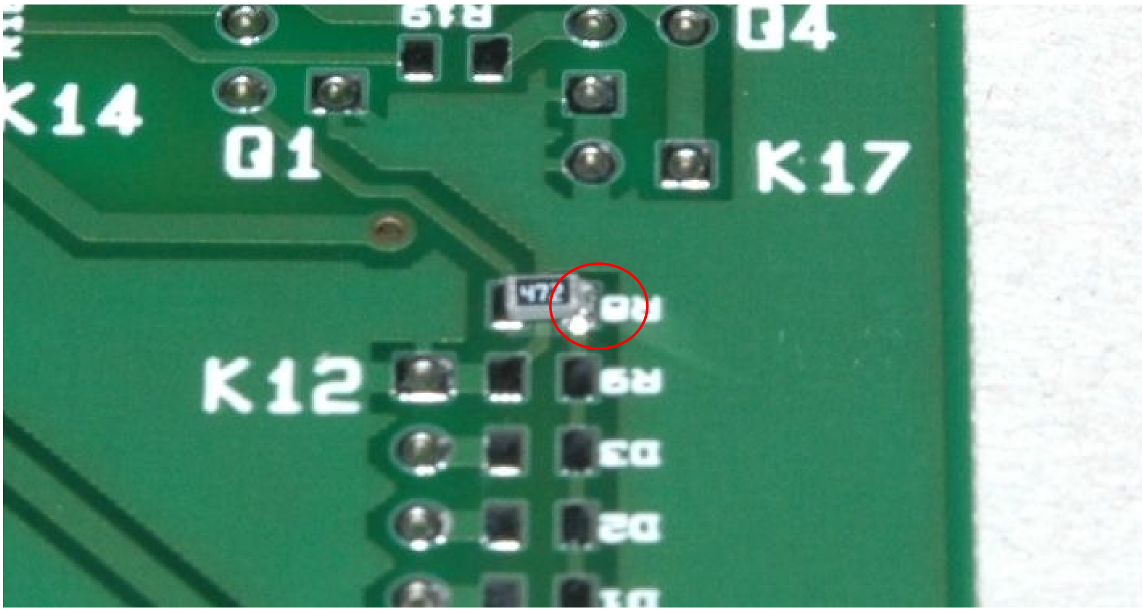


2. Avoid touching the parts with your fingers! Leave the parts in the bag or container until you need them. Some parts are not marked, and mixing them with other parts on the desk renders them useless.

3. Start applying a small amount of solder to one of the pads, crating a small mound.



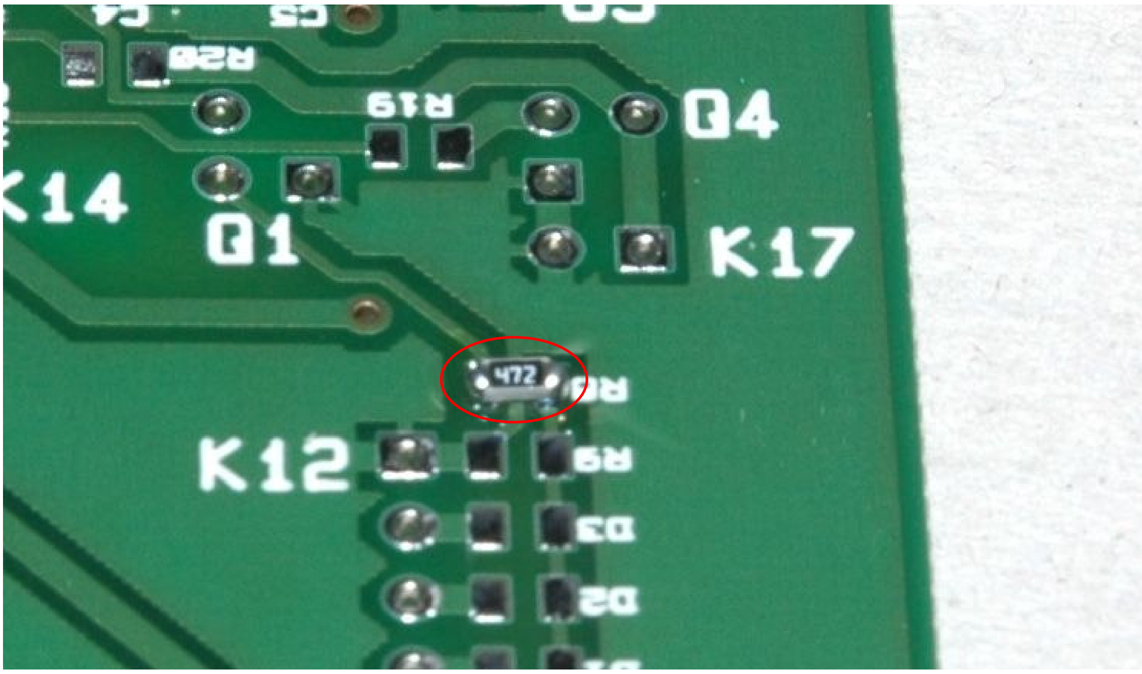
4. Pick up the part with your pincers, heat the solder at the pad and slide the part into place. Remove the tip of the soldering iron when the solder attaches itself to the part. Wait for the solder to solidify. Then let go of the part.



5. Sometimes a part will lift up one side (this is called the “tombstone effect”), sitting on the pad at an angle. In this case apply some light pressure to the lifted side (use your fingernail), and shortly heat the pad on the other side of the part. Let go of the part as soon as it lies flat on the PCB. Watch your fingers!

6. Then work on the other pad.

7. If needed, apply some more solder to the first pad.



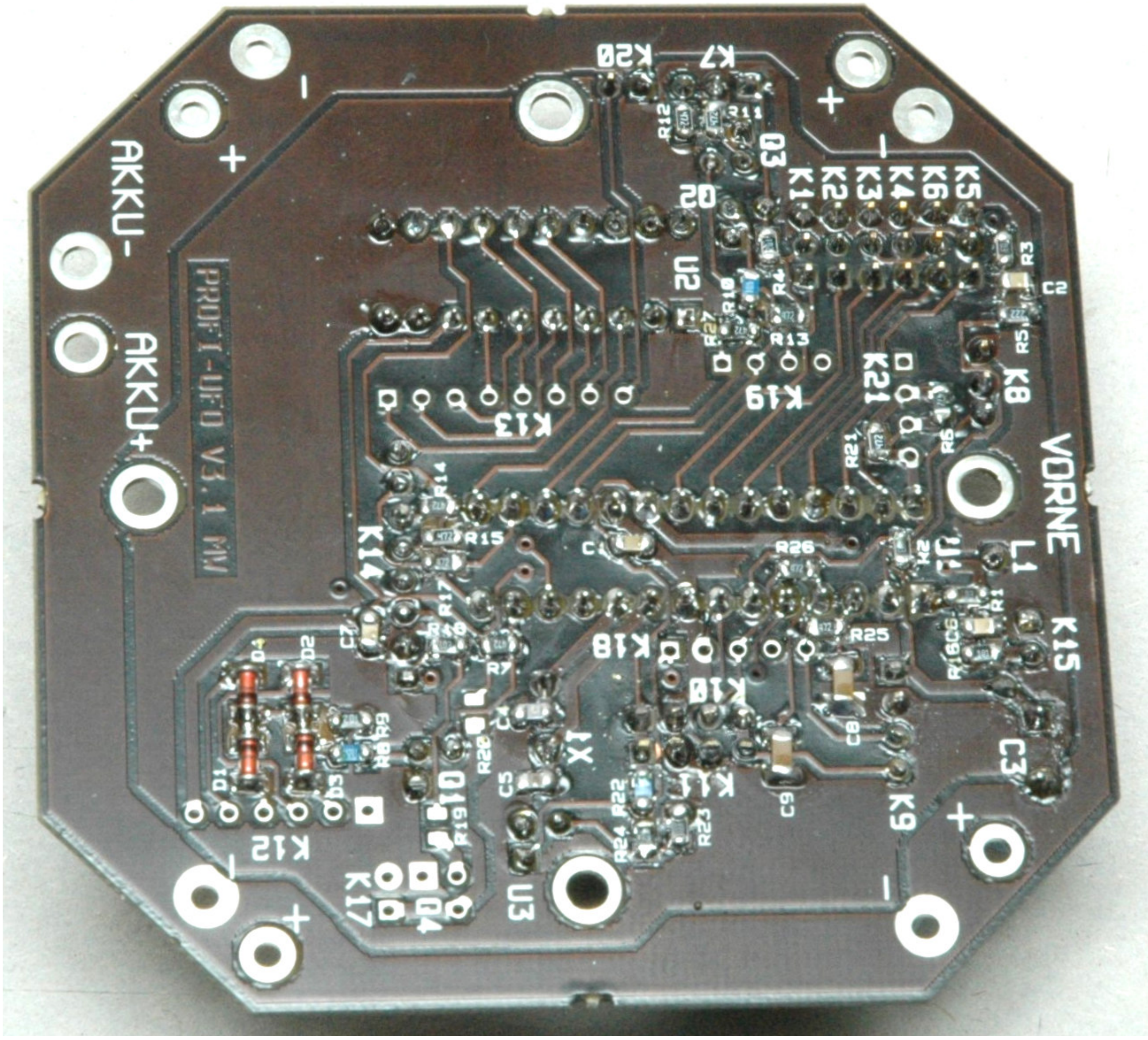
8. Apply heat only as long as necessary. Extended exposure to heat will damage the parts. Try not to stay longer than about 5 seconds on the part with your soldering iron. It is always better to work with more heat than applying less heat for a longer time.

9. Do not short out any pads with dabs of solder!

The surface of the joints should be shiny rather than dull. Check each joint with a magnifying glass!

  wolfgang.mahringer@sbg.at

This is what the finished PCB should look like:

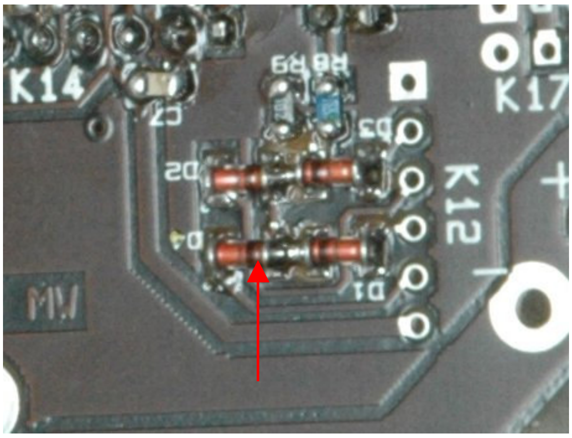


Individual parts may vary, depending on the set of options you chose to install.

Orientation of the polarized parts:

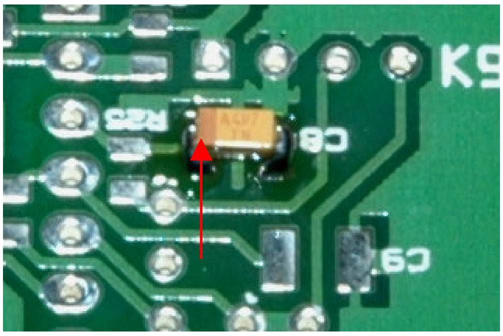
Diodes:

The ring (black or white) marks the cathode.



Tantalum-Capacitors:

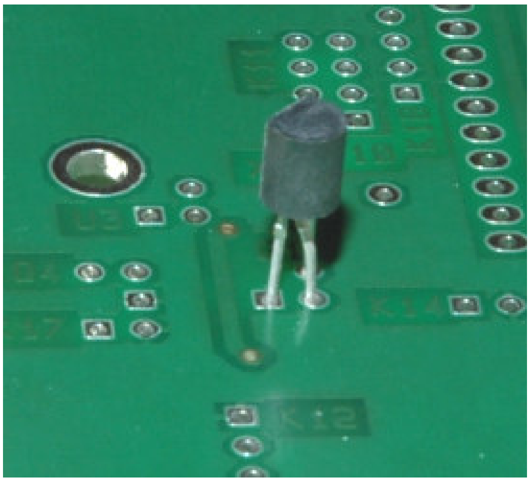
The printed bar marks the anode!



Transistors Q1/Q4 and IC U3:

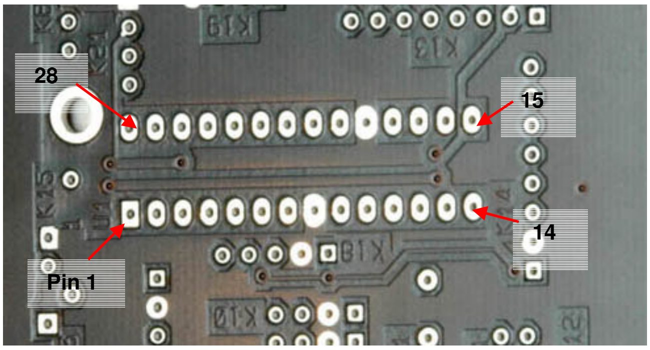
These parts are either shipped for mounting, or the middle pin needs to be bent somewhat.

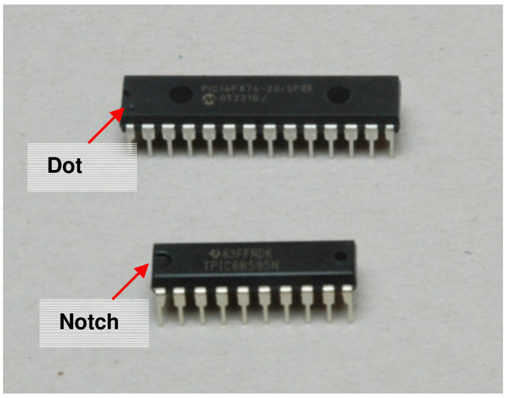
See picture for correct position:



IC's U1 and U2:

Note the notch or dot near pin1 on the IC's package.
On the PCB pin1 is a square pad rather than round or oval.



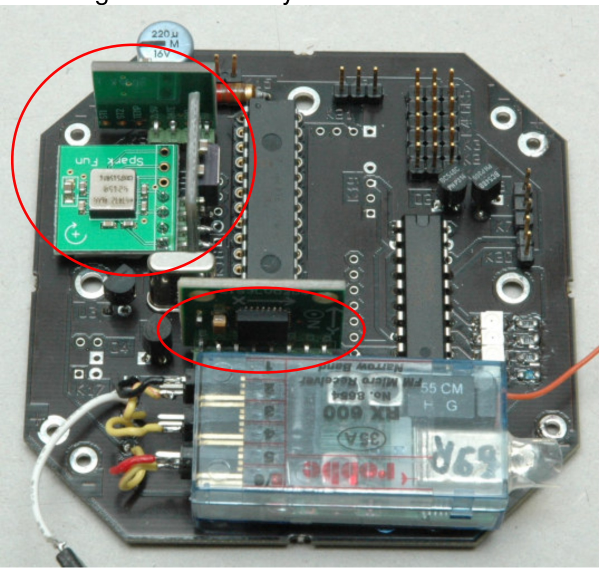


  wolfgang.mahringer@sbg.at

Mounting the gyros:

This is the last step. Make sure that the modules are always placed at right angles. To do this, solder only one pin at first, and then check alignment. If you need to correct the placement, you'll have to unsolder only one pin. Do not apply any force directly to the modules!
Place the modules on the pins without soldering them at first. Then check if they fit. Make sure that the modules do not touch or obstruct any other parts before you start soldering.

Mounting ADXRS300 Gyros and ACC



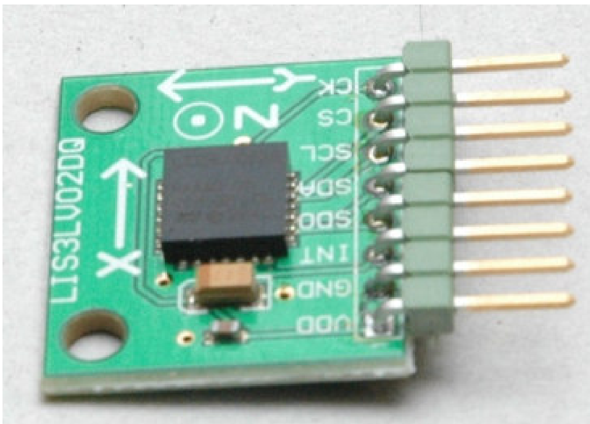
If you like, you can apply hot glue to apply more structural strength, but this is not required.

Mounting the ACC:

To allow as much room as possible for the Rx you should mount the sensor as shown.

Place the 90° pin header into the front side of the sensor's PCB!

For this sensor it is especially important to place it as upright as possible!



Connecting your Rx:

To save I/O ports and to avoid having to open the Rx in search of a serial signal, the Rx signals are connected to the board individually and are then mixed into one signal. To do so, just mount the signals in the shown way (you can also make them removable, as shown in the pictures top and right).

K12: Pin 1 = GND (black or green wire) to any servo signal i.e. No. 1
Pin 2 = servo 1 signal (white or yellow wire)
Pin 3 = servo 3 signal (white or yellow wire l)
Pin 4 = servo 5 signal (white or yellow wire)
Pin 5 = servo 7 signal (white or yellow wire) or leave open (when using 5 channel Rx)
Pin 6 = +5V (red wire) to any servo signal i.e. No. 1

Even channels 2, 4 and eventually 6 are not connected, the software calculates them from some signal gaps.
The servo signals from channel 8 onwards on your Rx (if any) are not needed here and can be used at your discretion!

Mainboard-Tests

Now it is time for some basic testing of the PCB and it's functions. A normal electronics meter is sufficient (set to 10VDC or 20VDC).
Connect the negative probe of the meter to GND, either at the negative battery lead, or pin 3 from K1 through K6.
Connect one brushless controller to K1 an, do not jet connect a motor to it. Then connect the controller to the battery. Do *not* connect anything else, like a receiver, etc.

Tipp:
To be on the save side, use a lab-type DC power supply set to about 12V. Limit the current to 500mA.

Now probe these points with your voltage meter:

Test point 1:
Probe pin 20 of the PIC (U1). You should find a stable voltage between 4.7 and 5.3V.

Test point 2: (only when ACC and/or IDG300 is installed)
Probing pin 3 of the TL431 (U3, round pad located next to K11) should show between 3.4 and 4.0V.

Test point 4:
Probe pin 3 of the PIC (U1).
Option 3x ADXRS300: You should find about 2,5V. Moving the PCB around the roll-axis will show changes in the voltage
Option 1x ADXRS300, 1x IDG300: You should find about 1.5V. Moving the PCB around the nick-axis will show changes in the voltage.

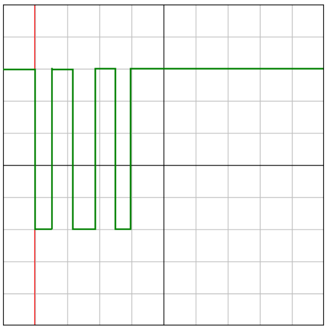
Test point 5:
Probe pin 4 of the PIC (U1).
Option 3x ADXRS300: You should find about 2.5V. Moving the PCB around the nick-axis will show some minor changes in voltage.
Option 1x ADXRS300, 1x IDG300: About 1.5V is nominal. Moving the PCB around the roll-axis will show some minor changes in voltage.

Test point 6:
Probing pin 7 of the PIC (U1) will show around 2.5V. Yawing the PCB should show some minor changes.

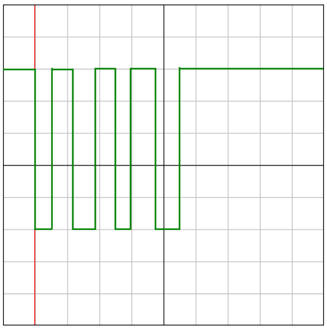
Test point 3:
For probing this signal (pin 13 of U1) you'll need an oscilloscope (DIY PC-soundcard-scope and scope-software is sufficient) as well as a working receiver and working transmitter.
Set the scope to Y-axis 1V/div and X to 2ms/div.
Trigger on DC (level) about 2V, set to negative slope.

The correct signal should look like this:

5 channels connected:



7 channels connected:



Amplitude: 5V
The pulse and pause widths should change by moving the sticks on the Tx!
Note that the slopes should be really steep (below 2 µs from low to high).

wolfgang.mahringer@sbg.at

Connections

Here is a list what should be connected to what connectors:

K1	Speed controller fore motor (flight mode “plus”) or fore -left motor (flight mode “cross”)
K2	Speed controller left motor (flight mode “plus”) or fore -right motor (flight mode “cross”)
K3	Speed controller right motor (flight mode “plus”) or aft-left motor (flight mode “cross”)
K4	Speed controller aft Motor (flight mode “plus”) or aft -right motor (flight mode “cross”)
K5	camera stabilizing servo nick direction
K6	camera stabilizing servo roll direction
K7	RS232 ComPort (for UAVPSet)
K8	On / Off switch
K9	Roll Gyro (if using ADXRS300)
K10	Nick Gyro (if using ADXRS300)
K11	Yaw Gyro
K12	Rx connector
K13	connector for LEDs and beeper
K14	linear acceleration sensor LIS3LV02DQ
K15	front LED connector
K17	parachute trigger
K18	combined Roll/Nick Gyro (if using IDG300)
K19	GPS Module
K20	Undervoltage / Model recovery beeper

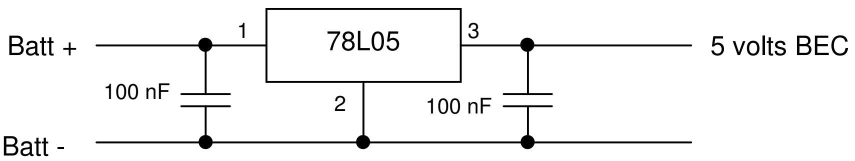
Own BEC supply (5 volts)

Should the 5 volts supplied from the speed controller ever be too weak, too unstable, or even not available (e.g. using Holgers BL-speed controllers), then we need an extra power supply for the board. Normally, when using standard PPM speed controllers, this is not necessary and should be skipped!

You need these components (not in the parts kit!):

1x 78L05	voltage regulator, case TO92
2x 100nF	SMT capacitor, case 0805

The circuit:

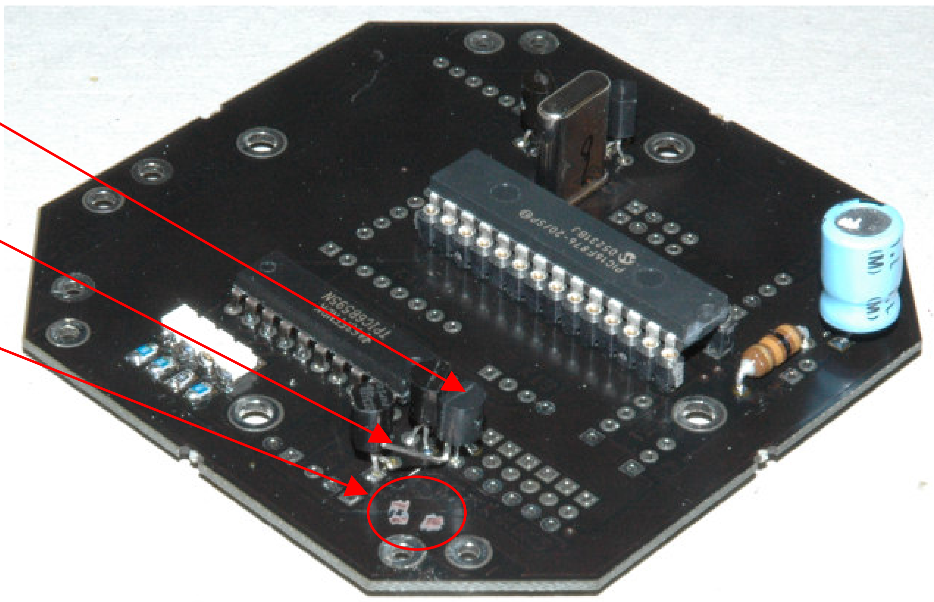


The 78L05 can be easily mounted on connector K1. The flat side of the IC’s case must point towards U2!

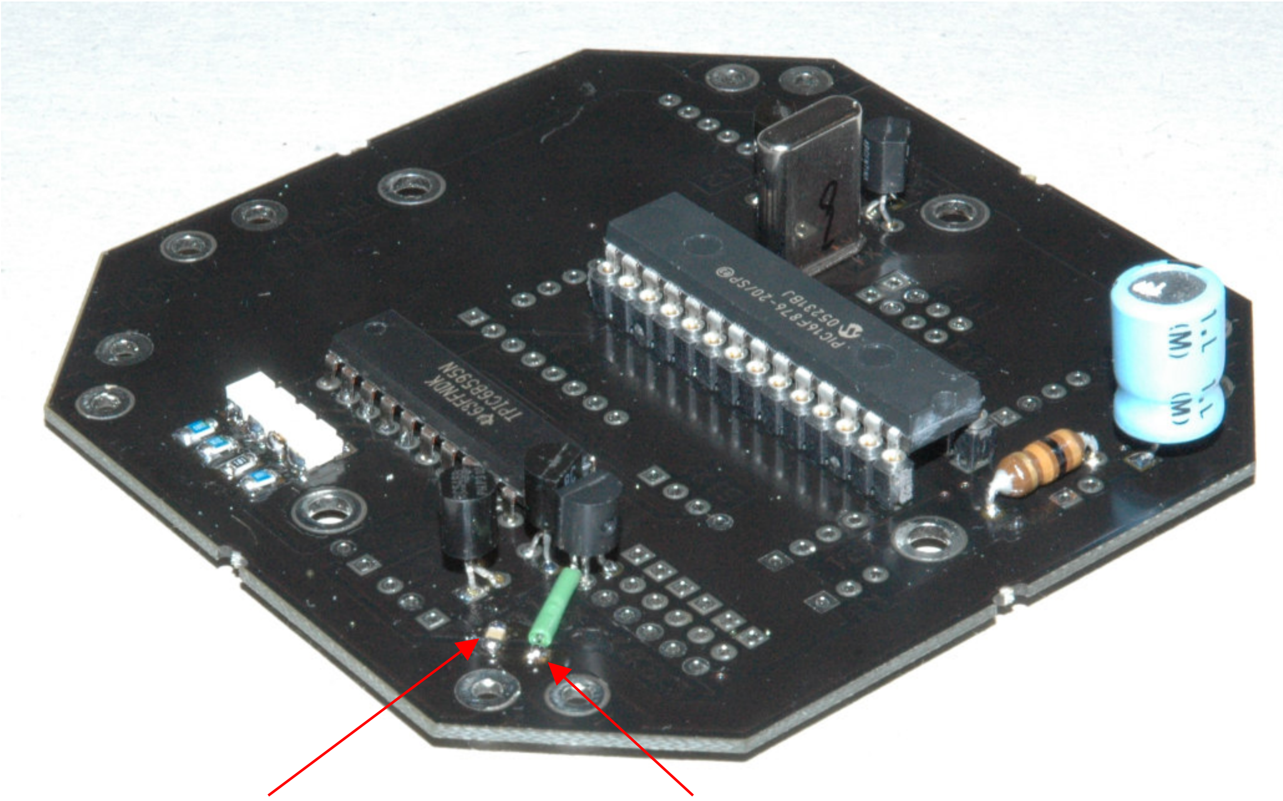
Bend one of the outer pins as shown.

Using a small grinding disc or a small milling cutter, remove the black coating.
Be careful to not damage the copper underneath too much!

The capacitor should fit nicely on the pad pair to span the gap between the two copper areas.



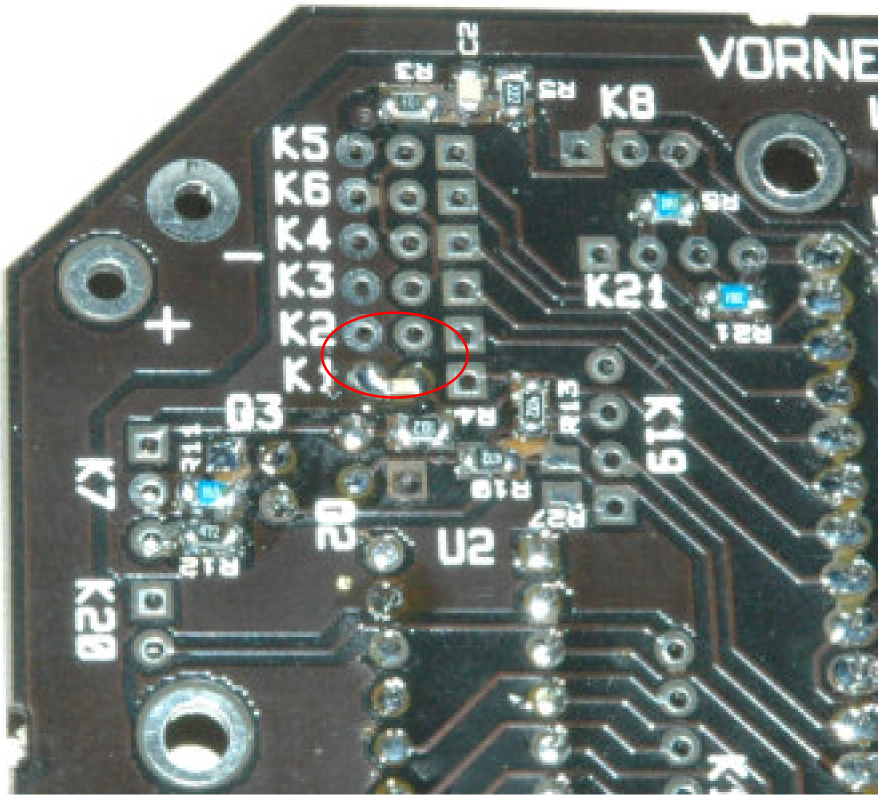
Position the third, single pad in a way to allow the bent pin of the voltage regulator to be soldered to the pad.



Now place and solder a capacitor here, then solder the regulators pin.

The remaining capacitor is to be soldered between pins 2 and 3 of connector K1 on the solder side of the board. It works best to place the capacitor on the long edge up.

This picture shows the capacitor soldered:



  wolfgang.mahringer@sbg.at

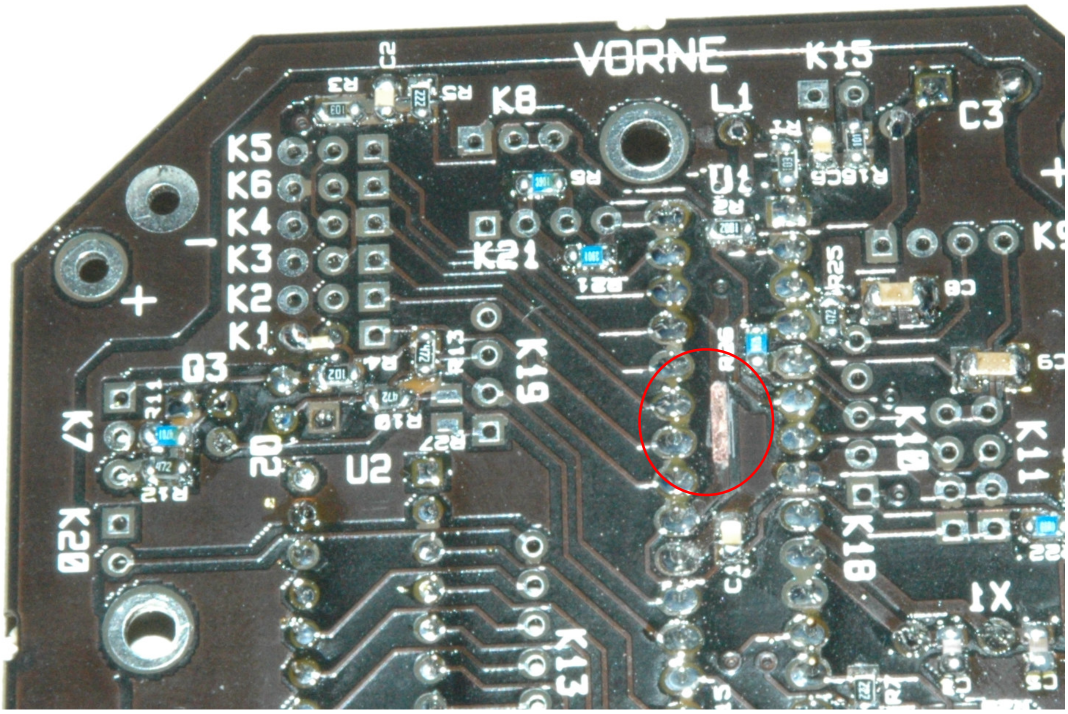
YGE i-series and Holger's BL-ESCs

This chapter is only important if you plan to use Holger's speed controllers (http://mikrocontroller.cco-ev.de/de/KopterBL.php) or YGE's i-series ESCs (http://www.yge.de/produkte.html).
These have some special features:

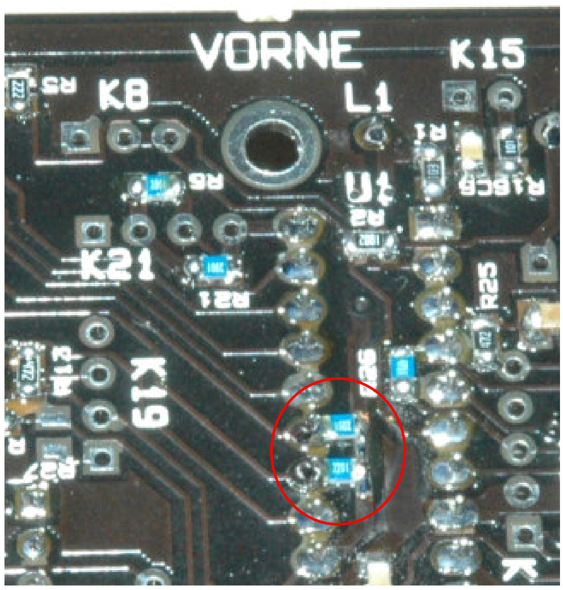
- The controllers do not work with a PPM pulse length, but with a I²C data interface. All four controller must be connected to one bus, which is comprised of two servo outputs (K2 and K3). Using a digital interface ensures exact transmission of values.
- Holgers controllers do not offer a 5 volts BEC source. So an extra 5 volts BEC circuit must be made, see chapter „Own BEC supply".
- YGE i-series controllers have a good, sturdy BEC. Make sure to connect one of the 3-pole servo cables to K1 to supply 5 volts to the mainboard.

To enable communication on the I²C bus, a clock and a data line is needed. The data line is pin 1 of connector K2, the clock line is pin1 of K3. To make it work properly, two additional pull-up resistors 4k7 SMT case 0805 must be mounted.

To allow for mounting these resistors, the black coating must be removed off a short piece of copper trace. Use a small grinding disc or a small milling cutter to do that. Take care to not break the copper trace!



Now you can solder the two resistors on the left side directly to U1 pins 22 and 23, then on the right side on the copper trace



The data and clock lines coming from the Holger controllers must then be connected in parallel to K2 (data) and K3 (clock) respectively.

First time operation

Before you can operate the QC, here are a few things to do first.

- **Programming the brushless controllers:**
  The brushless controllers have to be set up before use. First, connect the motors to the controllers, and the controllers to your Rx. Then set up the controllers as described in the manufacturers manual. To avoid injuries, **do not mount the propellers** for this step! Make sure that the motors turn in the right direction. Some controllers can change the orientation of motor spin via their setup – to find out if you controllers provide that feature please consult the manual. If this is not possible, just switch 2 of the 3 motor leads. Looking at your frame from above, the fore and aft motor should turn clockwise, the left and right motor turn counter clockwise. The setting we are looking for is described in some manuals as setting for „high pole count motors". This setting usually fits all outrunner motors.
  When using YGE's i-series controllers, these must be prepared using UAVP test software (especially about I²C slave addresses).
  It is also important to disable the low-power-cutoff, or alternatively set it to "2-cell-lipo! Otherwise, a controller my suddenly switch off the motor during flight, which will cause an immediate crash. If possible, switch the controller to a higher switching frequency, this eliminates the high-pitched noise from the controller. The drawback with this setting a slightly lower efficiency of the motor and subsequently more heat from the controller.

- **Programming the transmitter:**
  Set your transmitter to "airplane-mode"
  The software expects to find the signals in this order (**Graupner**), in **PPM-mode**:
  Stick 1: throttle
  Stick 2: roll
  Stick 3: nick
  Stick 4: yaw
  Stick 5: In-flight-programming: The software provides two separate sets of parameters which can be switched during flight (switch +/-100%)!
  Stick 6: Camera tilt
  Stick 7: not used yet

  Do not activate any mixing of channels (i.e. swashplate-program, etc.).
  Choose your personal expo and dual-rate settings for roll and nick.

  The software expects positive signals, which is standard for most receivers.

  Robbe-Systems are set up in the same way, except these settings:
  Stick 1: roll
  Stick 2: nick
  Stick 3: throttle

- **Power up the UAVP:**
  Mount the programmed PIC in its socket.
  Set the power switch to „OFF" (closed) position.
  Switch on your transmitter, throttle set to minimum.
  Connect the battery.
  The controllers will signal their readiness by sound and/or light, please consult the controller's manual.
  Red LED (and eventually yellow LED) is lit. If not, something is wrong! Switch off immediately and check your wiring!
  Set the power switch to „ON" position
  Yellow LED is lit for about 5s (Gyro-Init)
  If the red LED is flashing at this point, check the position of the throttle stick on your Tx.
  If the red LED continues to flash, check if the correct stick/channel is used for throttle (Graupner/Futaba-Robbe). Just set register 16 to the right value.
  The green LED should light up after 5s, you are now ready for take off. Well, almost:
  The software controller has to be set up. Before you continue, set the power switch to off and disconnect the battery.

CPU-controller-parameter:

The CPU performs 3 independent PID controls (Proportional/Integral/Differential) in realtime, one for each axis of movement. These controllers have to be parameterized to ensure a fast control response, but also prevent catastrophic oscillations. These parameters depend mostly on the physics of the QC, such as the moment of inertia, reaction time, rate of acceleration of brushless controllers and motors.

Proportional: This part of the commanded action that comes directly from the stick and the gyros. If the UAVP does not react well/fast to stick commands, the value is too low. If the UAVP shows signs of catastrophic oscillations (the oscillations increase to a point where control is not possible) the value is too high.

Differential: Used to control reaction time. Usually uses the opposite sign (positive or negative) as the proportional constant. If the UAVP shows signs of catastrophic assimilations (the oscillations increase to a point where control is not possible) the value is too high.

Integral: This is the summation of the gyro values over time. Very important for the Heading-Lock feature of the yaw axis. A value too high will cause oscillations, too low will cause the UAVP not to turn back to the initial angle or attitude.

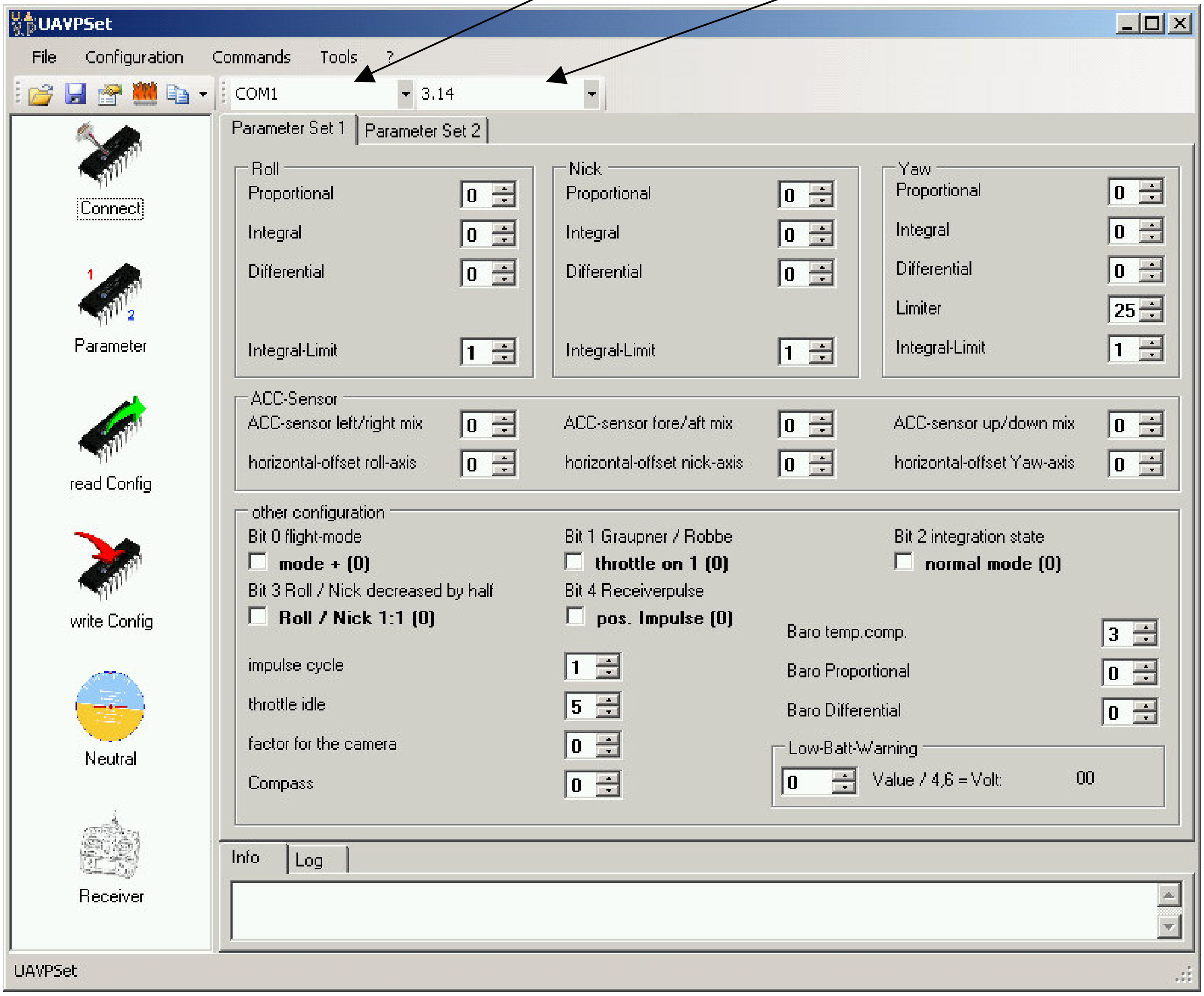
These parameters are stored in the non-volatile memory of the EEPROM. Once set, they are stored permanently.

Upgrading from flight software 3.11 or older to 3.12 or newer

With flight software version 3.12 the meaning of roll and nick proportional and differential parameters has been changed to allow easier parameter finding:
Proportional-roll-new = Proportional-roll-old + Differential-roll-old
Proportional-nick-new = Proportional-nick-old + Differential-nick-old
Set „Differential“ (means control stability) as you prefer, start with 0.

Programming via UAVPSet

This Windows application makes parameters setup easy.
Connect UAVP to the RS232 port, start UAVPSet, choose correct COM port and Version 3.14.
Apply power to the UAVP.



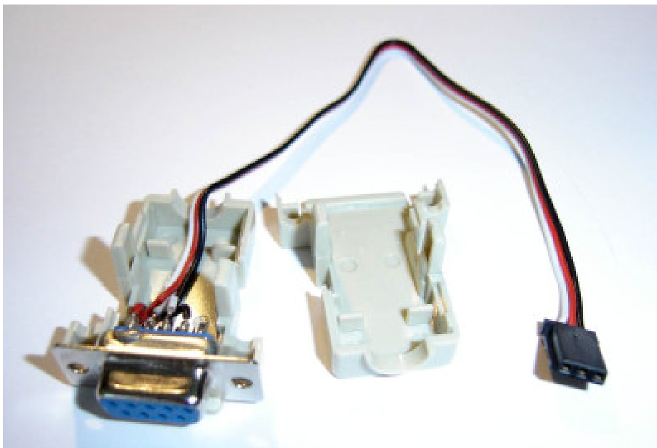
It is not necessary to close the application or even the serial port after every setup.
Just click „write Config“, wait for the OK, then you can unplug the UAVP and try out your settings. The application does actively check if the write was successful.
For the next modification, just reconnect the UAVP, adjust values to your desire.

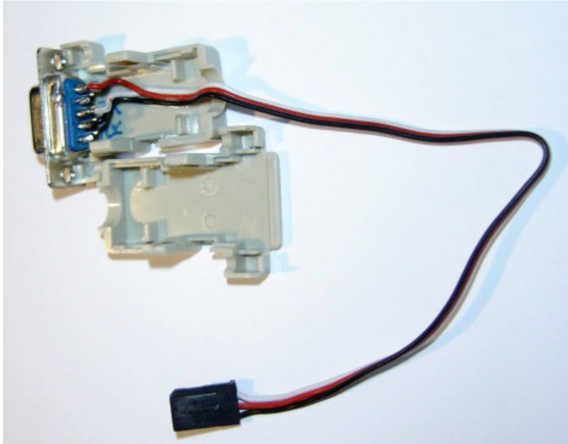
  wolfgang.mahringer@sbg.at

Programming parameters via RS232

To connect the PCB to a PC, you will need to make a cable. Utilizing a serial modem cable is a good way of doing this.

K7 Pin 1	2	
2	5	Sub-D 9pol female
3	3	





Start a terminal software (such as Hyperterm or Minicom, etc.) on your PC and set the COM port settings to 38400 baud, 8 data bits, 1 stop bit, no parity and no handshake.

Connecting the battery should cause this display:

```
Profi-Ufo V3.12 ready
Gyro modules: 3x ADXRS300
Linear sensors ONLINE
Compass sensor not available
Channel mode: Throttle ch 1
Selected parameter set: 1
>_
```

Entering a "?" shows a list of valid commands:

```
> ?
Commands:
L...List param
M...Modify param
S...Show setup
N...Neutral values
R...Show receiver channels
>_
```


 wolfgang.mahringer@sbg.at

Entering “L” or “l” (upper or lower case) will list current settings:

```
> L
Parameter list for set #1
Register 01 = -01
Register 02 = -01
Register 03 = -01
Register 04 = -01
Register 05 = -01
Register 06 = -01
Register 07 = -01
Register 08 = -01
Register 09 = -01
Register 10 = -01
Register 11 = -01
Register 12 = -01
Register 13 = -01
Register 14 = -01
Register 15 = -01
Register 16 = -01
Register 17 = -01
Register 18 = -01
Register 19 = -01
Register 20 = -01
Register 21 = -01
Register 22 = -01
Register 23 = -01
Register 24 = -01
Register 25 = -01
Register 26 = -01
>_
```

Depending on the current setting of stick 5 on your transmitter you can use 2 different sets of settings that can be switched even in mid-flight. Make sure to select the right set of parameters before you change any values!

Modifying a register’s value:

```
> M
Register _
```

Now the software expects a two-digit number for the register.

```
> M
Register 03 = _
```

Now enter a two-digit value for this register. Three digits are accepted if the first one is a sign:

```
> M
Register 03 = -06
>_
```

Values entered in this way are stored immediately in non-volatile memory.
After entering the values you can check them by listing them using the “L”-command.

Applying throttle on your transmitter will close the RS232-port. It can be reactivated using the switch connected to K8), or just remove the UAVP’s power supply for a few seconds.

  wolfgang.mahringer@sbg.at

List of parameters:

1. Roll Proportional
2. Roll Integral
3. Roll Differential
4. Baromatric sensor temperature coefficient
5. Roll Integral-Limit (always positive!)
6. Nick Proportional
7. Nick Integral
8. Nick Differential
9. Barometric sensor proportional value
10. Nick Integral-Limit (always positive!)
11. Yaw Proportional
12. Yaw Integral
13. Yaw Differential
14. Yaw Limiter (always positive!)
15. Yaw Integral-Limit (always positive!)
16. Configuration:
    Bit 4: 1=inverted RX signal is applied (only use in very special cases!)
    0=non-inverted TX signal
    Bit 3: 1=Roll- and Nick-stick commands (values) are decreased by half (Specky-bit)
    You may want to consider this with proportional values above 20!
    0= Roll- and nick-stick commands normal (100% in is 100% out)
    Bit 2: 1=programming-LEDs show state of integration (see Reg. 24 and 25)
    0=normal flight-mode
    Bit 1: 0=Graupner-mode (K1=throttle, K2=roll, K3=nick, K4=yaw, K5=select)
    1=Futaba/Robbe-mode (K1=roll, K2=nick, K3=throttle, K4=yaw, K5=select)
    Bit 0: 0=flight-mode: “Plus” (1 motor front and 1 motor back)
    1= flight-mode: “X” (2 motors front, 2 motors back)
17. impulse cycle (in milliseconds, 1 < n < 20)
18. Low-Batt-Warning (LEDs blinking)
    value = voltage [V] * 4.6 (for 3S LiPolys meaning about +46)
19. ACC-sensor left/right mix
20. ACC-sensor fore/aft mix
21. ACC-sensor up/down mix
22. ACC-sensor horizontal-offset Yaw-axis
23. throttle idle setting for brushless controllers
24. ACC-sensor horizontal-offset roll-axis (only valid if Reg. 2 not eq. 0)
25. ACC-sensor horizontal-offset nick-axis (only valid if Reg. 7 not eq. 0)
26. Mixing factor for the camera roll and nick stabilizing servos
27. Compass sensor factor
28. Barometric sensor differential value

<u>Caution:</u> Maximum value for the Integral-Limit-Register is 127 / |<Integral-Register>|!
Example: You want Reg. #2 to show a value of –6. In this case the maximum value for the corresponding Integral-Limit-Register #5 is 127 / 6 = 21 (rounded). If the value is too high, you may experience flips during wide banks!

There are 2 sets of parameters to choose from by toggling stick #5. If the impulse width on channel #5 on your Rx is below 1.5ms, parameter set #1 is active.

<u>Recommended values</u> for the large Hammer-Frame, 3xADXRS-gyros:
Roll: 1: -26, 2:-5, 3:0, 4:+13, 5:+15
Nick: 6: -26, 7:-5, 8:0, 9:+1, 10:+15
Yaw: 11: +16, 12:+30, 13:0, 14:+30, 15:+4
16:0, 17:+6, 18:+50, 19:0, 20:0, 21:0, 22:+5, 23:+22, 24:0, 25:0, 26:16, 27:4, 28:4

Meaning settings for Graupner, „PLUS“-flight-mode, 160Hz update-rate (6ms), using Roxxy 2824/26 motors on YGE-30i controllers and EPP1045 props.

  wolfgang.mahringer@sbg.at

Explanation of PID flight controllers

PID-controllers are common throughout the technical world (the acronym means Proportional, Integral, Differential). They continuously compare sensory inputs (in our case the gyros) and the values commanded by the pilot (i.e. horizontal attitude) and then calculate the delta. This delta is then used as the deviation value.
The deviation value is in turn used to calculate a value for the actors (here: motor rpm) to achieve the wanted effect as fast as possible, without oscillations.

The gyros are built to measure an angle velocity rather than an absolute angle.

The proportional portion discerns the amount of influence of the gyro's value on the rpms. A high value will cause a hard counter reaction to a change in attitude. Above a certain value the system will start to oscillate, however. The ideal spot is just below this point. You cannot fly a UAVP with a proportional value of 0.

The Integral portion calculates the current absolute angle by adding up the gyro values. The parameter rules the amount of influence from this to the propeller rpms. The UAVP can fly fine with a integral value of 0. It is then controlled like an RC helicopter, the pilot commands angle rates rather than absolute angles.

The Differential portion calculates how fast a change in angle occurs. In a quadrocopter these values are usually used as a damping factor, meaning negative signs in the formula. The damping mostly eliminates vibration effects from the motors and props.

There are 2 areas, or points in the graph (= combinations of parameters) that will allow a stable operation: the positive feedback and the negative feedback point.

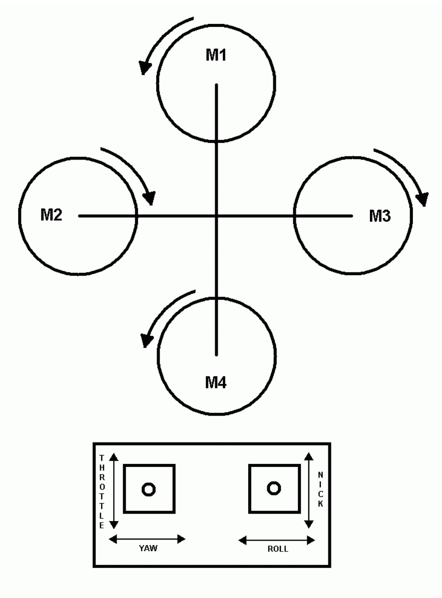
The positive feedback point features a small proportional portion and an even lower differential proportion **equally signed**. This is a rather shaky way of control, since even the slightest deviations from the set values will cause large effects, including oscillations, caused by the differential value actually amplifying the oscillations.
Typical values (3xADXRS300 version): Proportional roll and nick = -11, Differential roll and nick = +2
It is not recommend using the positive feedback setup!

The negative feedback point features relatively large proportional- and differential portions, with **opposite signs**. Valid combinations of values are many, so stable flying is relatively easy, even when adding weight or payload. A differential value with a reversed sign helps damping vibrations and causes the UAVP to be less twitchy.
Typical values (3xADXRS300 Version): Proportional roll and nick = -24, Differential roll and nick = +10

Builder's trials

It makes sense to set all parameters to "0" for the initial flight, except:

- Register yaw-Limiter (Reg. 14), set to +40.
- Register yaw-proportional (Reg. 11), set to +40.
- Flight configuration (Reg. 16) accordingly
- Timing-Register (Reg. 17), set to +4
- Low-Batt Register (Reg. 18) accordingly
- Throttle-Idle-Register (Reg. 23) set to +50



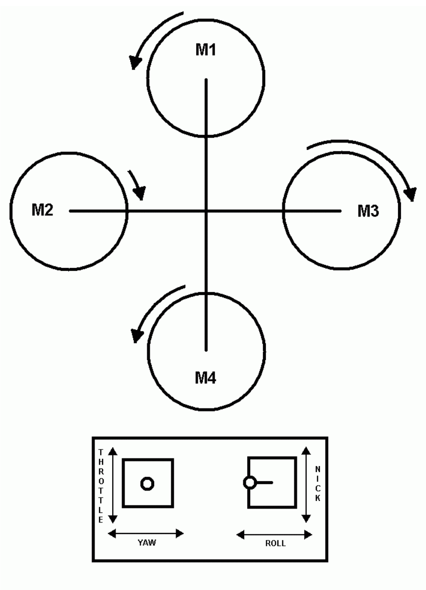
Grab your UAVP from underneath so that the rear motor (M4) shows towards you, and apply throttle – real softly – up to half throttle! All 4 motors should start up.

The motors should turn steadily without causing any vibration.

Check each prop's direction of turn: Front and back counterclockwise the others clockwise. You can switch this to the opposite configuration; you only have to invert the signs on the yaw-parameters.

You will not see any reaction to roll, nick or yaw movements – that's normal since the parameters are still set to zero.

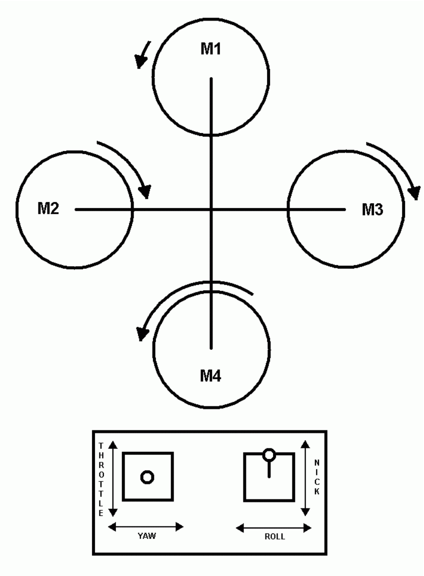
Now you should check if the sticks move the UAVP to the correct directions.



Again, grab your UAVP from underneath and apply half throttle.

Move the roll stick softly to the left. If your UAVP wants to tilt to the left, that's fine. If not, use the servo reverse feature on the roll channel of you transmitter. Try again, to make sure it is now correct.

When rolling, one motors runs slower, while the opposite motor accelerates.



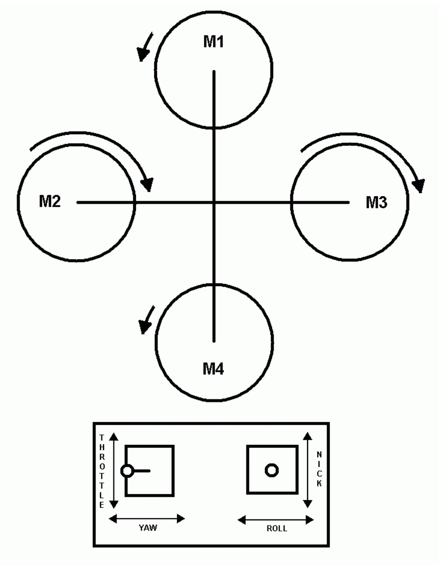
The same procedure is used to check the nick axis.

Grab your UAVP from underneath and apply half throttle.

Move the nick stick softly up. If your UAVP wants to tilt forward, that's fine. If not, use the servo reverse feature on the nick channel of you transmitter. Try again, to make sure it is now correct.

When nicking, one motors runs slower, while the opposite motor accelerates.

Now, only the yaw axis is left to check.



Grab your UAVP from underneath and apply half throttle.

Move the yaw stick to the left. If your UAVP wants to rotate counter-clockwise, that's fine. If not, use the servo reverse feature on the yaw channel of you transmitter. Try again, to make sure it is now correct.

When rolling, two opposite motors are running slower, while the others accelerate.

At this point you should set the idle-throttle register. Apply some throttle, to have the motors run at idle. Then set the throttle stick back to minimum immediately. The motors should continue to run at the lowest possible speed for a couple of seconds and then stop completely and simultaneously.
In case of the motors stopping right away increase the value of register #23 (use steps of 5). This may cause motors to stop completely in those situations. Since the motors need a few moments to start up again, this may cause a crash!.

Now you should check which timing the speed controllers will accept. Generally speaking - smaller and lighter UFOs need faster timing. Decrease the value of register #17 in small steps (the minimum is +3) and see if the motors will still start up without problems. It may happen that the motors go to full speed right after startup! To prevent accidents, perform this step without propellers!
To be on the save side, increment the lowest possible value that you found by +1. Since all other parameters depend on this setting, try not to alter the setting of register #17 later on!

Usually, timing setting between +3 and +6 are fine.

Remount the propellers. Continue setting roll and nick proportional values (starting at -18). Testing roll and nick stick commands will show the correct direction of control. Test the gyro reaction by tilting the UAVP along the nick and roll axis'. You will feel a decisive counter force.
If the UAVP should try to flip over in your hand, the direction of control for that axis is reversed! It will therefore amplify your motions rather than countermand them. Invert the signs on roll and nick proportional settings, and try again. If your stick commands appear to be opposed, just invert the channel on your transmitter.

Now move on to set the yaw proportional value. This value is not that critical, just begin with +16. Try to yaw the UAVP with the throttle set to just above idle and note if the direction of yaw matches your stick's movement. With the stick set to neutral, yaw the UAVP with your hand and check if the UAVP tries to countermand that action.

Actually, your UAVP is now ready for a first flight! Find a big room (no obstacles please) since this eliminates wind problems. Careful on the throttle! Do a short take-off and trim your UAVP so it won't drift in any direction, and will not yaw.
Your UAVP will probably handle like a wet sponge. To get better performance, you can increment the roll and nick proportional registers step by step, until the UAVP tends to oscillate. Decrement the value a bit to be on the safe side.

For perfect control, you can now add the differential register values (roll and nick). Just let somebody tip the UAVP from underneath the aft motor and see how the UAVP corrects it. Choosing the values too high will cause the UAVP to wobble in hovering flight.
Usually, the differential registers can be kept at 0.

The next step is to set the heading-lock for the yaw axis. Start with a value of 20 for the yaw-integral register, same sign as the proportional register's value. Set the Yaw-Integral-Limit-Register to +4. Slowly increment the value, in steps of 5. Check if the UAVP will yaw left and right on it's own while hovering. If so, decrease the value. Now calculate the correct value from Integral-Limit-Register and set this value.

The UAVP should now hover perfectly. Only slight corrections may still be needed.

Integral flight

Normal Flight Mode means controlling the rate of nick and roll by stick command, eventually controlling the angle velocity per axis. This is comparable to controlling a model helicopter.

Additional stability is achieved by mounting the ACC option.

First, you have to set the horizontal levels for each axis. Place the UAVP on a level surface, like a table top. For best results, check the horizontal orientation using a spirit scale. Also make sure that the UAVP sits flat and without wobbling.

Connect the battery, hook up the RS232 connection and enter “N”:

```
> N
Neutral Roll:+019, Nick:-052, Yaw:-040
>_
```

Copy the roll value (here: +19) to register 24 and the value for nick to register 25. The yaw value is not used here.
These values are quite twitchy – you can check the values again by reconnecting the battery.

The neutral values cannot be collected by the UAVP by itself, because you will not always start from a horizontal surface.

After this, you also have to set the integral values for the flight controller.

To do this, increment the integral registers (both) by steps of 1, starting from 0. The sign has to be same as the proportional registers’ values. Then set both Integral-Limit-Registers to +4 an leave them at that for now. A test flight will show that you are controlling angles rather than angle rates now. Let go of the sticks and the UAVP will try to roll and nick back to horizontal. The effect is increased by incrementing the integral-parameters.
You may experience erratic behaviour, the UAVP starts to wobble. If you still want to increment the integral value further you’ll have to decrement the proportional values at the same time.
When you are done setting the integral values, continue setting the value for the Integral-Limit-Registers. The maximum value is 128/<Integral-Reg>.
Example value is –6: 128 / |-6| = 21.333, rounded down to 21.

Caution:
Do not set too high values in the limiter register – in extreme flight manoeuvres this may cause unexpected flips and therefore a crash!

Integral flight mode without ACC-sensor

You can fly in integral mode without an ACC sensor. However, since the “zero” setting is not resetted by the sensor, it will “drift away”. In this case, you will need to apply stick in that axis just to hold the UAVP level. To reset, just land, switch off the motors, and start again.

Notice:
Make sure to start from a horizontal place!

Registers 24 and 25 in this mode are used as trim registers. If your UAVP always drifts away in the same direction, you can trim it with these parameters. Try with steps of 8.

As for the flight controller-integral-parameters – the above mentioned rules apply!

Setup of the compass sensor

If you have put a compass sensor on the mainboard, it serves 2 purposes:

1. Correction of transmitter-caused slow yaw drift
2. Detection of the flight position/direction (important for GPS controlled flight)

Holding the yaw position (heading hold) is always done by the yaw gyro. But often one can see a very slow yaw rotation when hovering. This drift can't even be stopped by using the trimming facility on the transmitter. But the compass sensor can take care of this. As soon as the yaw stick is not moved for about 3 seconds, the UAVP locks the current compass direction reading and tries to hold it constant.

There is a compass factor (on UAVPset) which controls how strong this locking mechanism works. This factor should never be set to more than 8, because then the effect of the compass is stronger then the yaw gyro control. Since the HMC6253 is only a 2-axis-compass, it will misread when the UAVP is not hovering perfectly even. When flying maneuvers, the UAVP will inadvertently yaw.

Values of 5 or 6 have been proven to work fine.

Setup of the barometric sensor

The barometric sensor allows the UAVP maintain a flight altitude automatically. The achievable accuracy is about 1,5 Meter (5 ft).

Air pressure is very dependable on the air temperature, so the UAVP sports a temperature compensation. Measurements showed a value of +13 for baro-Temp (UAVPset) to be optimal.

Nevertheless, the UAVP should not be flown directly after seeing large temperature differences, e.g. from the warm car trunk to the cold winter air. It is best to allow 5 minutes to accommodate the electronics.

The altitude controlling is implemented as an PD algorithm. There are 2 values to setup: baro proportional and baro differential. For baro proportional, a value of +1, for baro differential, a value of +4 is a good start.

The altitude control does also work indoors.

The UAVP locks in the current altitude as soon as the throttle stick is not moved for about 3 seconds.

Note:
To bring an UAVP down for landing, it is a good idea to do that stepwise, reducing and increasing throttle. If you just reduce throttle, the UAVP will come down, but after about 3 seconds it will lock the new, lower altitude.

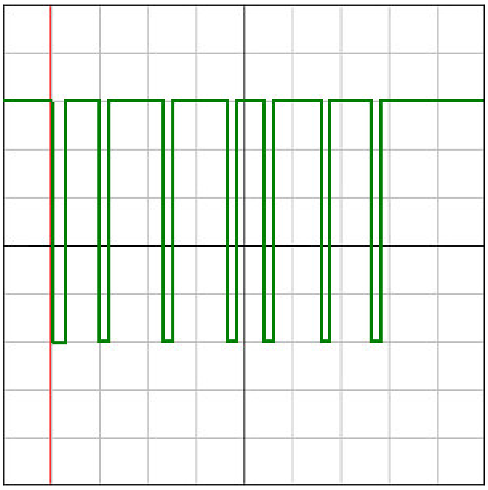
Using RX composite signals

If you plan to use small, lightweight and reasonably prices receivers, one often faces the problem that these receivers only sport 4 channels. These cannot be used for conventional connection to the UAVP mainboard, since it needs at least 5 channels.

Often receivers can be modded to output the PPM composite signal. This way, the number of usable channels only depends on your transmitter.

The PPM composite signal carries (after demodulation of the RF) the information of the sticks positions.

The composite signal can only be found in PPM receivers, not in PCM models!



The leading edge (at the red marker) ist the start pulse. The time to the next falling edge is the servo time of channel 1. All the other channels are following consecutively, in this case 6 channels.

The composite signal can directly connected to K12 on the Mainboard, no matter if it looks inverted or not. The signal should at least show an amplitude of 1.5 volts.

To use this feature, you need to flash a HEX file with „RX_PPM“ in its name onto the board!

Misc advice

A red flashing LED shortly after power-up means that your throttle stick is not set to minimum (-100%). It also can denote an offset throttle trim. To clear that, just set the throttle to minimum and the trim accordingly.

Some brushless controllers will cut off power to the motor when a low battery voltage is detected.
You should consider setting a timer on your transmitter to know when to land. If possible, switch off the low-batt setting on all of your controllers, and activate it on the main board.

The low battery detection implemented in the main board will activate a beeper when battery levels are low. This may prevent crashes from dead batteries an save LiPo batteries from being drained below safe voltage levels.

Write down all of your register settings! In case the EEPROM needs to be reprogrammed, this will save you a lot of time!

Last but not least:
It is strongly recommended that you get yourself a liability insurance which covers harm caused by models or model pilots. A signal failure, a technical problem, or a pilot error may cause serious crashes!

Troubleshooting

If you get no reaction from the main board, an oscilloscope comes in handy. A simple, analogue device will do, even a DIY PC-soundcard-mod serves the purpose.
However, here is list of known problems to check before you start excessive testing.

You should consider removing the props before any testing. Leave them on and your fingers will tell you what I mean… ☺

Notice:
Do not apply full throttle to motors without a prop attached! Due to the missing load the controller may take damage.

Nothing happens (no LEDs, not even the “ready”-sound from the BL-controllers):

- Can you find the +5VDC from the BEC to the PIC?
- Do the BL-controllers get a signal?
- Processor not programmed/wrong program/wrong version?
- Configuration fuses set wrong?
- Crystal is not oscillating, check soldering joints. C4, C5: correct values?
- Processor set in the socket the wrong way?
- Electrical short on the 5V supply?

Motors beep, but no LEDs:

- LEDs mounted the wrong way?
- +5VDC or GND at the processor’s pins not connected
  (check with a line probe)

Motors beep, red LED (maybe also yellow LED) OK, switch set to ON, no green LED:

- Power switch connected wrongly?
- Bad signal from receiver?
- No +5VDC or GND at the receiver?
- Rx crystal installed?
- Transmitter switched on? Tx crystal installed??

Motors beep, red LED (maybe also yellow LED) OK, switch set to ON, red LED flashing:

- Throttle stick not set to minimum
- Maybe the throttle channel on the Tx needs reversing
- Register 16 set to the wrong Tx make.

Stick action wrong, gyros correct:

- Reverse the appropriate channel on your Tx.

Stick action correct, gyros wrong:

- Register signs wrong, reverse them.

UAVP wobbles after takeoff:

- Roll/Nick-Proportional Parameter values too high
- Roll/Nick-Integral Parameter values to high
- UAVP too light (use bigger battery)
- Antenna not connected

UAVP flies OK, but sometimes motor or motors stop suddenly:

- Radio problems
- BEC overload (check temperature of the BEC supplying controller), some controllers will supply only .1A BEC. These controllers are not usable!

UAVP shows yaw oscillations or refuses to yaw in one direction:

- One or more motors are not aligned vertically (motor axis)
- Yaw-Parameter value too high, decrease Integral-Parameter

Version history:

Version numbers of this document correlate with the firmware version numbers on the PIC!

V3.10 29.3.2007	First version of document, for HEX version 3.10
V3.11 20.4.2007	Compass sensor added Added hints for mounting tantalum caps
V3.12 13.5.2007	Section about changed meaning of roll and nick P and D params Text about support of magnet sensor added Advice for a liability insurance added Chapter “Own 5 volts supply” added Chapter “Holger’s brushless speed controllers” added added pictures RS232 cable
V3.14 31.12.2007	General rewrite, totally new “Builder’s trials” part new pictures for UAVPset new section for compass- and baro sensors added RX composite signal section

