FCM developers manual

(c) 2013-2015 Fabian Huslik

This work is licensed under the
Creative Commons Attribution-ShareAlike 4.0 International License.
To view a copy of this license, visit
http://creativecommons.org/licenses/by-sa/4.0/.

1 Introduction

This is a rudimentary attempt to write down, what is going on in FCM.

The most accurate documentation is the source code itself and the schematic with layout.

The only thing, you really can rely on in this document, are the axis definitions in "5 Definitions & Copter Layout".

2 License

If not otherwise stated in the source files, most of the code is copyrighted as follows:
Copyright (c) 2008-2015 by Fabian Huslik

Some other but not less important parts:
MadgwickAHRS / MahonyAHRS: Copyright (c) 2011 by SOH Madgwick
menu: Copyright (c) 2010-2014 by Jörg Schmidt & Fabian Huslik
hottv4: Copyright (c) 08/2012 by Adam Majerczyk
BMP085 driver: Copyright (C) 2009 Bosch Sensortec GmbH
printf: Copyright (c) 2004,2012 Kustaa Nyholm / SpareTimeLabs
ASF Framework: Copyright (c) 2009-2014 Atmel Corporation. All rights reserved.
Mavlink: Copyright Lorenz Meier and contributors

FCM Android: Copyright (c) 2014 Joachim Weishaupt

License details for each file are specified inside the source files.

For the documentation and hardware design the following applies:

This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by-sa/4.0/.

3 Bootloader

To activate bootloader, plug in the FCM via USB and THEN power it up. Hardware version 2.1 is not USB powered, so external 5V is needed.
During bootup, the bootloader looks for USB voltage (5V). If USB is found, the bootloader mode is activated and the main application will not start.

- Blue on and green blinking with 5 Hz: waiting for USB
- Blue blinking with 3 Hz: USB serial port connected.
- Red lit: flash write access.
- 3-colour cycling: Application has started. See FAQ

The USB-waiting-mode will be repeated, if no application is loaded.

To load a hex file, go to FCM_Manager and click on "connect". If the bootloader responds , the "load HEX file" button gets activated.
select "load HEX file".
Do not forget to compile the hex file for YOUR configuration beforehand.
Adjust settings in config.h

Parameters will be preserved only, if nothing in the parameter config has changed.
Save your parameters using the FCM_Manager BEFORE flashing.

The FCM_Manager lets you know, when it's finishes with success.
It does only take a few seconds to load the hex file.
When finished, disconnect USB and cycle power supply.

3.1 USB Drivers

The USB drivers are located in the "Bootloader/Bootloader" directory. Just point windows to this directory, if the board is connected for the first time.

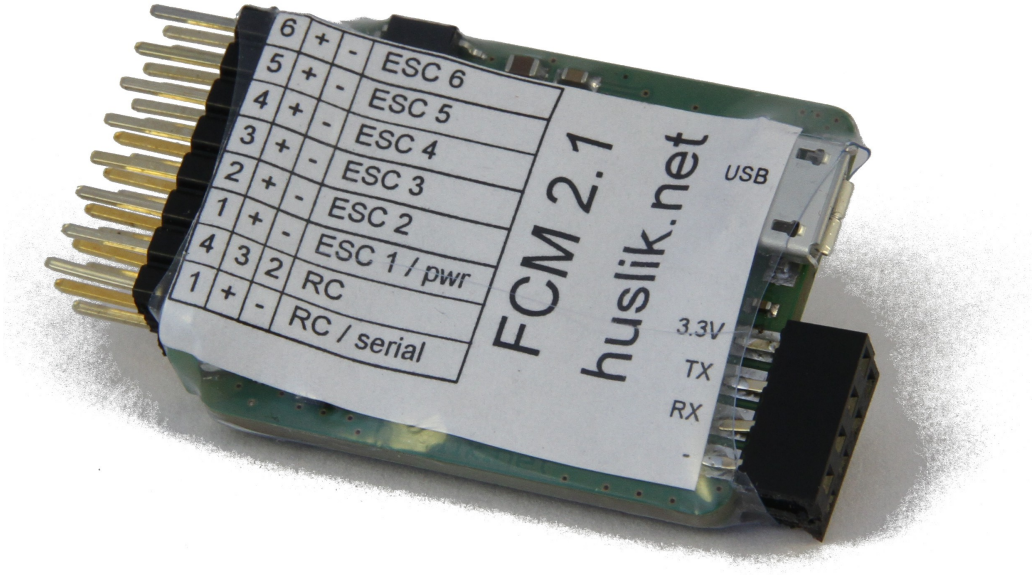
3.2 Installing USB Bootloader on Board

Do this in order:

1. Erase the controller
2. Flash the bootloader hex file (in bootloader directory, there is only one)
3. set fuse bootloader flash size to 16kByte.
4. Do not set the BOD fuse, this will be activated in SW!

4 Building FCM hardware

The

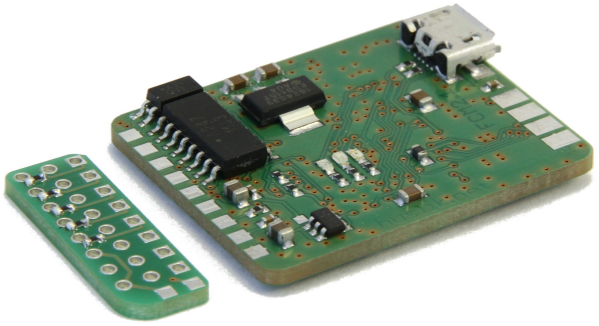


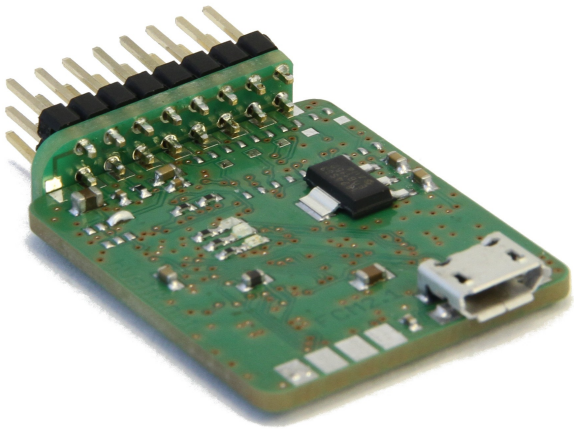
schematics of the latest hardware can be found at:

versions\hardware V2.0 08-2012\hexa2wide.pdf

The "trunk" already contains hardware adaptations for the next version.

FCM consists of two PCBs. The small PCB carries the connectors and is fixed to the big one at the front. Solder the big PCB between the two lower connector rows. Solder the two extra pads at the left (botton and top)





4.1 Placement variant Receiver supply

For SPEKTRUM sattelite operation a P-channel MOSFET "T1" is placed, so that the 3.3V supply become switchable by FCM.

To enable 5V supply for the receiver, connect TP24 to the adjacent MOSFET pad. Usually a solder blob will do the job.

4.2 Placement variant with optocouplers

For opto version use the small PCB with resistor pads, Place 5x 2k2 resistors before soldering the pin headers. Use only little solder for the pin headers before fixing it to the main PCB.

4.3 Placement variant without optocouplers

If the optocouplers are not required, do not place them, place instead 5x **2k2** resistors from the TP19..23 test points to the LED Anode pad. The "other" small PCB without resistor pads has to be used, where all supplies are connected.

4.4 Final assembly

Use a 1 mm piece of ABS to cover the lower part of the PCB. The components are selected for equal height.

Fix the printed paper cover to the small pcb with a little (hot) glue

Shrinkwrap the whole thing.

Fix it to your copter with double-sided foam adhesive as explained in 7 Installation on Copter.

5 Definitions & Copter Layout

For a printout of axis definition, see later in this manual.

5.1 Motor layout - top view:

```
(1)   (2)       ^
    X           |
(4)   (3)  dir of flight

 (1) (2)        ^
(6) X (3)       |
 (5) (4)  dir of flight
```

5.2 Motor turning directions:

```
(+)   (-)   + = right
    X       - = left
(-)   (+)

 (+) (-)    + = right
(-) X (+)   - = left
 (+) (-)
```

5.3 Axes positive directions for Vehicle:

Axis	Acceleration	Rotation
X:	left to right	around x backward
Y:	back to front	around Y right
Z:	bottom to top	around Z CCW (looking from above)

Rotation directions follow the right hand rule from each axis direction!

5.4 Axes for Board:

Connectors for RX and ESCs face into flight direction.

Flat side with CPU faces down.

5.5 Servo input scaling

throttle = 1000 = idle - 9000 full
roll: -4000 = right +4000 = left
elev : +4000 nose down -4000 nose up
yaw: -4000 = right +4000 = left

6 External electrical connection

GND is up!

Connect power supply to the "ESC1" connector. Connecting to the Receiver input may fry your board (3.3V-supply in SPEKTRUM version)

From left (seen in flight direction)

(1 + -) RC / serial : Receiver with power supply and Serial / PPM or "classic" channel 1

(4 3 2) "classic" channels 2,3,4 or special function

(1 + -) The first ESC with BEC, which also supplies the 5V power for the FCM

(2 + -) ESC with BEC (supplies only opto)

(3 + -) ESC with BEC (supplies only opto)

(4 + -) ESC with BEC (supplies only opto)

(5 + -) ESC with BEC (supplies only opto)

(6 + -) ESC with BEC (supplies only opto)

At the no-opto variant there is obviously no galvanic separation.

The Comm ports are left of the USB connector.

USART0 is up (Telemetry)

USART2 is bottom side (GPS)

Pins LTR: GND, rx, tx, +3.3V

Telemetry (no matter which type) goes to the upper connector.

GPS goes to the LOWER connector.

Absolute max total load at all 3.3V outputs is 200 mA at 5V supply.

7 Installation on Copter

7.1 Mechanical alignent

FCM must be installed flat with the RX and motor connectors facing straight forward.

Up to now, there is no rotation offset possible.

Align the FCM perpendicular to vehicle axises. (naah the multiple of axis..)

7.2 Magnetometer considerations

- Absolutely NO current loops on motor / battery / ESC wires
- Twist all wires and ensure equal length
- The bigger the distance of the FCM to the high current wires, the better.
- The influence is reduced with the squared distance.
- Even batteries can emit a significant magnetic field at high currents.

Try in "direct mode" with the fixed copter and watch the magnetometer values using telemtery. They should not move by more than a few percent of the max-min value during motor running. (To be honest, if the fluctuations are less than one third of the signal span, flight could be possible).

7.3 Vibration considerations

Vibration has a big effect on sensor performance. Especially the accelerometer will see all vibration.

Signs of a vibration issue could be poor levelling performance.

Therefore some tips to prevent vibration related issues:

- Balance your props as good as possible
- Balance your props in all dimensions (some props have a significant imbalance perpendicualar to the blades – the short way, which you wouldnt expect)
- Center your props (some hubs require 0.1mm additional sheet or some tape to center perfectly)
- Mount the FCM on the 2mm white foam (if no problems exist)
- Mount the FCM on 3mm black foam rubber (if there is residual vibration)
- If there is still unresolvable vibration, reduce the area of foam (cut out two stripes or leave a hole at center)
- Do not mount FCM too softly, as it will not directly follow the airframe movement, which leads to oscillation
- A loosened FCM will lead to a uncontrollable vehicle.
- Make sure, that there are no resonances on the airframe. (GPS-Antenna on long pole etc...)

8 Source code management

We use github at the location

https://github.com/fabianhu/FCM/

If you are allowed to commit to the repo, please feel free to do so.

PLEASE write in commit message, what you did.
Most important stuff in first line, add a '+' at the end of the first line, if you provide a multi-line commit message, which is highly appreciated.

Or use the github facilities to open an issue or a pull request.

If you need an idea, what to do, look at the todo.txt in the "doc" directory .

9 Parameter Description

Default values for small (800g) quad in "()".

The following text is auto-generated out of the Treebuilder.exe !!! The single source for this is the *.jmdef file.

GYRO PID	Gyroscope based Governor settings
| <back>	
| Max Integr	Max Integrator windup (4000)
| Gov at Max	Governor intensity at max power [%] (20)
| Q Gain	Quaternion turn rate (4000)
| *Swash as Poti	Use Potis as PID settings for Swash (xy)
| Swash P	(60)
| Swash I	(37)
| Swash D	(37)
| *Yaw as Poti	Use Potis as PID settings for Yaw (z)
| Yaw P	(58)
| Yaw I	(21)
| Yaw D	(16)
NAV	Navigation Governor Parameters
| <back>	
| NAV Limits	Navigation Governor Parameters
| | <back>	
| | Max Angle	The max lean angle in degrees (30)
| | Max Accel	The max acceleration out of nav in 1/10 m/s^2 (2)
| | Max Acc int	The max acceleration integrator windup out of nav in 1/10 m/s^2 (2)
| | Radius NAV	The radius in meters, at which the cruise speed starts to decelerate (20)
| | Radius h	The radius in meters, at which the height change speed starts to decelerate (20)
| NAV Filters	Navigation Governor Parameters
| | <back>	
| | Acc Flt glob	global acceleration filter value (1000) 1000 is unfiltered
| | alpha pos	complementary filter for position (???) 0 is prio for GPS
| | alpha spd	complementary filter for position (???) 1000 is accel prio
| NAV Compensation	Navigation Compensation
| | <back>	
| | Comp On	Activate the compensation algorithm
| | Min Speed	Minimum speed, at which the vectoring calculation starts (1)
| | Min Time	Minimum time en route, after which the vectoring calculation starts (5)
| | Filter	100 is unfiltered
| | Min Distance	Minimum distance from target to activate the compensation
| *NAV as Poti	Use Potis as PID settings
| NAV P	(25)
| NAV I	(0)
| NAV D	(30)
| *H as Poti	Use potis as PID settings for height governor
| H P	(300)
| H I	(50)
| H D	(1000)
Settings	General settings
| <back>	
| Max Power	Max throttle (6500)
| Idle Power	Min throttle, use above 1500 for idle running props

| Agility fact | How aggressive the copter reacts to RC (1-2)

| wait GPS Fix | 1 = Flight is only released if GPS is fixed

| No BT222 cfg | 1 = skip BTM222 initialisation sequence at start

| Direct Mode | 1 = NO Flight control, throttle directly without unlock to all ESC! Use this to calibrate all ESCs at once. Negative numbers = single motor control, others off

| SPEKTRUM DSMX | 1= try bind in DSMX mode

| *Bnd SPEKTRUM | Do bind Spectrum Sattelite at RC-Port

| *Start Bootloader | Go into bootloader mode

Calibration | Sensor calibration

| <back>

| Filters | Sensor calibration

| | <back>

| | Gyro Filter | Gyroscope filter: 1= unfiltered (1)

| | Accel Filter | Accelerometer filter: 1= unfiltered (4)

| | Magneto Flt | Magnetometer filter: 1= unfiltered (4)

| | Baro Filter | Barometer filter: 1000= unfiltered (990)

| | GPS Filter | GPS filter: 1000= unfiltered (990)

| *Calibrate R+A | Calibrate Gyro and Accelometer ? put on flat surface!

| Gyro X | Gyro calibration value

| Gyro Y | Gyro calibration value

| Gyro Z | Gyro calibration value

| Acc X | Accelometer calibration value

| Acc Y | Accelometer calibration value

| Acc Z | Accelometer calibration value

| MagCal flght | Magnetometer calibration during engines on allowed ? (0)

| Mag X | Magnetometer calibration value (via online calibration)

| Mag Y | Magnetometer calibration value (via online calibration)

| Mag Z | Magnetometer calibration value (via online calibration)

RC Channels | RC channel assignment. Go to diag mode to see results.

| <back>

| Sw PosHold | Which channel is used for Pos hold (ch #)

| SwP PosHold | At which pos is the function active (0=lo 1=mid 2=hi) Pos Hold wins over RTH!

| Sw H Gov | Which channel is used for height governor on (ch #)

| SwP H Gov | At which pos is the function active (0=lo 1=mid 2=hi)

| Sw RTH | Which channel is used for Return To Home (ch #)

| SwP RTH | At which pos is the function active (0=lo 1=mid 2=hi)

| Sw Aux | Which channel is used for Auxiliary function (ch #)

| SwP Aux | At which pos is the function active (0=lo 1=mid 2=hi)

| Poti P | Which channel is used for P potmeter

| Poti I | Which channel is used for I potmeter

| Poti D | Which channel is used for D potmeter

*Save to Flash | Saves all settings values to flash

*Diag mode | Display diag page (press enter to leave)

Test | This is magic stuff... will disappear later.

| <back>

| Madgwick

| alpha Madg | x 0.01 --> 100 = 1.0 (10)

| prop Mah | x 0.01 --> 100 = 1.0 (50)

| integr Mah | x 0.01 --> 100 = 1.0 (10)

| test P

| test I

| test D

| wind

```
|  | wind_freq
|  | wind_ampl
|  | <back>
| mag_mis
|  | mag_mis_freq
|  | mag_mis_ampl
|  | <back>
| gyro
|  | gyro_freq
|  | gyro_ampl
|  | <back>
| accel
|  | accel_freq
|  | accel_ampl
|  | <back>
| magneto
|  | magneto_freq
|  | magneto_ampl
|  | <back>
```

10 Firmware internals

The "secret" keywords are:
"fixme" for things, that should be fixed soon or require some more analysis.
"todo" for things, that may be unfinished or prepared new features or just have much room for improvements.
All variables starting with "debug" should have no functional meaning and should be removed in the future.

10.1 Folder structure

The FCM flight-Software is in "software".

All sources are in "src".

In the long run, all .h and .c files should move into modular sub-folders.

in "src" the following important folders exist:

menu : here is all menue related

modules: here the "beautiful" code is put, that means, it has somehow clear interfaces.

10.2 Menue editing

For editing the menue, use ONLY the "Treebuilder.exe" from the software folder.
The menue content is stored in "quadcopter.jmdef".
If changing a node, do not forget to press "change"
If finished with editing, click on generate and safe the "quadcopter.jmdef" file over the old one.
After generating, click into the left output window, this will put the content into the windows clipboard. Paste it into the menu_variant.h file lower part. Some editors (as the Atmel Studio) insert line breaks. You may have to remove these after inserting. You will see on build...

The menue description for each node is also edited in TreeBuilder. It comes out again as comments with the generated code and also as text to be inserted into this document in 9 Parameter Description.

10.3 Adaptations for your copter

In the file config.h are some important switches. These are explained there.

The miximg matrix is defined in /src/modules/servo_out.c. Look at the existing variants.

10.4 Building a hex file

Use Atmel Studio 6.2.1153 with

Atmel AVR (32 bit) GNU Toolchain – 3.4.2.1057 Version:
AVR32_Toolchain_Version:3.4.2.435 GCC_VERSION:4.4.7

Atmel Software Framework - 3.19.0.1334

You need to have TortoiseSVN installed for the pre-build script to work.

Do not change any linker or compiler settings, it is working beautifully as adjusted.

The first instruction goes to 0x80004000, The bootloader will jump there for start.

You may switch off the '-pedantic' option, if you get too many warnings. Ignore all warnings, which are telling "... is a GCC extension".

There are some adaptions inside the framework files regarding the trampoline and exception table (scall) – so be careful with updating the framework not to kill the stuff.

10.5 Communication protocol

Please see FCMCP.ods.
The parameter part of MAVlink is implemented and working.

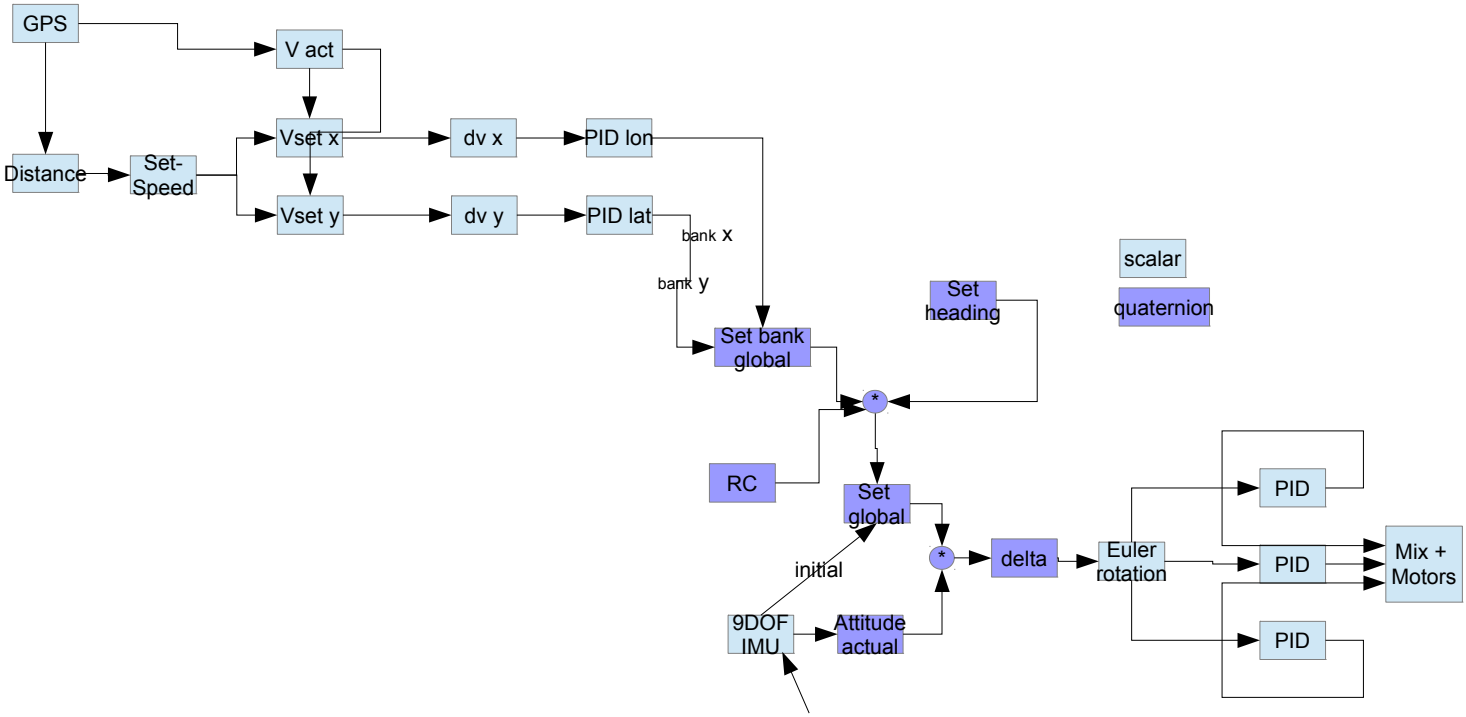
11 Math – some hints

Excellent source for all math in 3D: http://www.euclideanspace.com/maths/index.htm

another "primer" for quaternions:
http://www.cprogramming.com/tutorial/3d/quaternions.html

11.1 The master plan

(copy of New Governor Math.odg)



An efficient orientation filter for inertial and inertial/magnetic sensor arrays
Sebastian O.H. Madgwick
April 30, 2010

The most promising source: (LGPL – but not only because of this)
madgwick_algorithm_c.zip with
madgwick_internal_report.pdf

11.2 Quaternion math in code

Look at /modules/quaternion.c

11.3 How to get the "difference" of two Quaternions

Just "multiply" them.

Which means, take the first rotation and rotate it further by the second.

11.4 How to filter Gyro, Acc and Mag into one "My actual attitude" quaternion:

Madgwick or Mahony it. There is some sensor fusion involved.

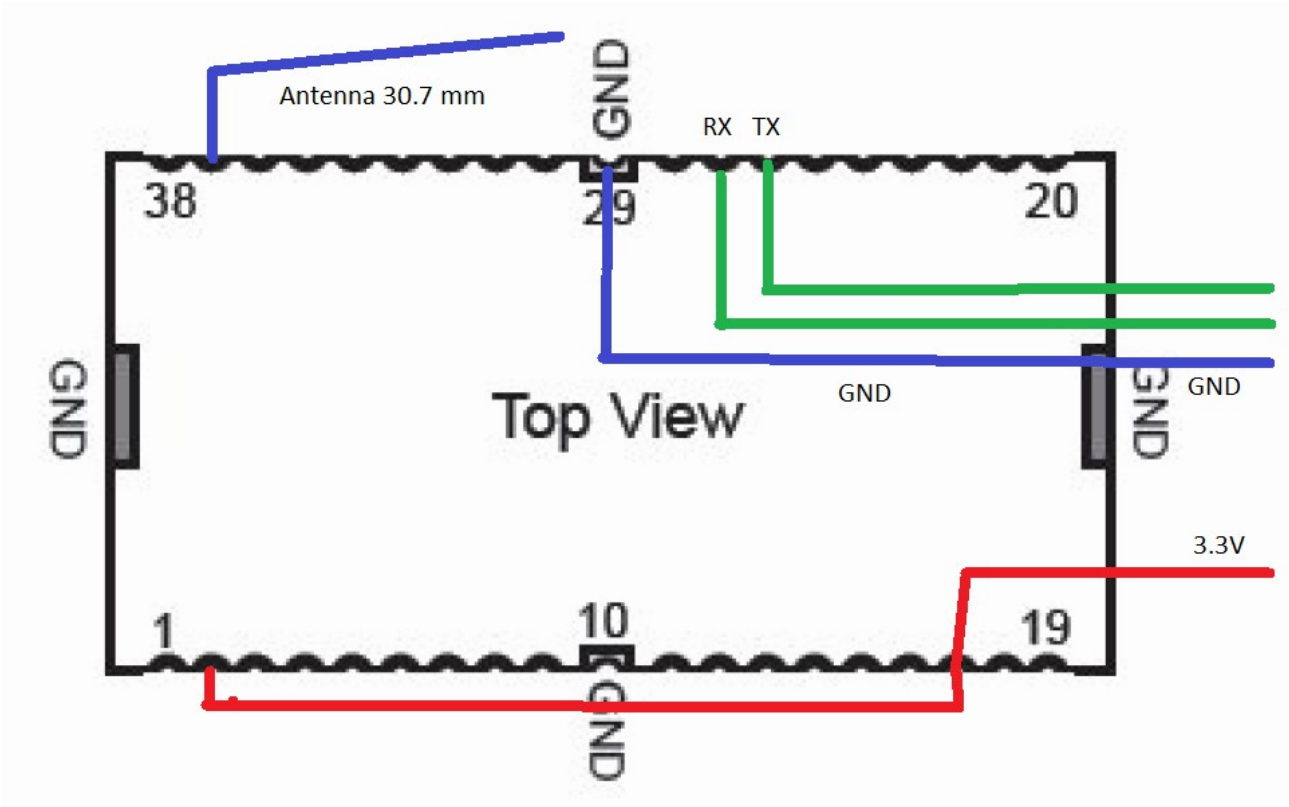
11.5 Some other IMU source – just for reference:

GPLV3:

https://code.google.com/p/gentlenav/downloads/list

12 BTM222 Hints

Referring to the BTM222 datasheet for the pinout, solder the following:



This is a TOP view !!!

Put the wires to the bottom! (the 2 – 17 jumper can stay at top)

Pin 2 + Pin17 = 3.3V supply

Pin 29 = GND (all other required GNDS are already connected to the tin shield)

Pin 26 = connect to FCM TX

Pin 27= connect to FCM RX

Pin 37 = connect a wire of more or less exactly 30.7mm measured from the BTM222 edge (cut after soldering)

To clean superfluous com ports:

Create a batch file and execute as admin with:

```
set devmgr_show_nonpresent_devices=1
set devmgr_show_details=1
start DEVMGMT.MSC
```

Then select "show unused devices" and remove the stuff.

13 Thanks

Many thanks go to my great beta testers:

Uli

Joachim

Jörg

Armin

Christian

14 MbAQ Might be Asked Questions or FAQ

14.1 What are the blink codes:

- red on, blue&green blinking alternating: No RC signal
- running light: not ready (throttle zero, no GPS lock or FCM not level)
- red on and blue&green blinking together: fatal error state: count number of blinks and search for "emstop(n)", where n is the number of blinks.
- Green blinking: ready to be armed with yaw right and throttle 0
- blue steady on: flight; may be disarmed with yaw left and throttle 0

14.2 LSM303DLHC magnetometer not working

If the LSM303DLHC delivers irregular values (too low, only one axis working or so...):

check the external capacitor CA2 or CA3. One of these IS failed.
Thats easier than changing the chip, as I did first, which of course didn't solve the problem.

14.3 Whats this nice diagram on the last page?

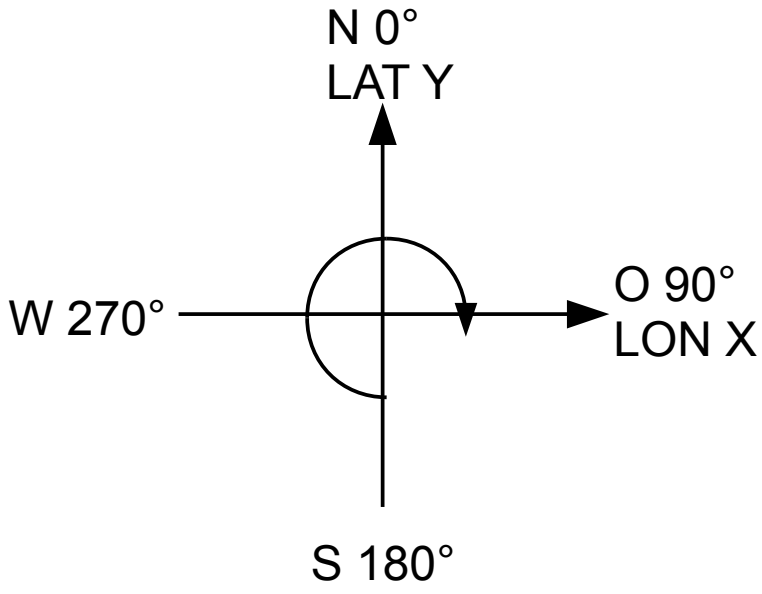
That's the truth. Believe it.

We use a ENU orientation for all 3d stuff. Right handed!

The navigation is done in a x-y = E-N order, where the rotation is right positive.

All this might be changed to the NED standard later. But anyway, there is no standard.

2D Navigation



3D Orientation