

Flight Controllers

www.dupedup.cz

Petr Dušek, 3/2019 - revision 4/2020, 1/2021, 7/2021

I am interested in RC models/drones and I use **RC Flight Controllers (FC)**. I like some commercial FC boards e.g. MATEK. The FC board consists of MCU (STM32) and sensors. The sensor IMU (Inertial Measurement Unit) acts as 6-axis accelerometer/gyro and a pressure chip sensor acts as the barometer. The IMU sensor is often MPU6000 and barometer BMP280. Current, power and temperature sensors are also embedded. The MCU I/O pins are led out on the board and may be connected via UART to other sensors (mostly a GPS modul, compass or pitot tube module), RC receiver (via Spektrum DSM SRXL2 on 2.4 GHz in my case) or other devices. Some MCU I/O pins are connected to servos via Pulse Width Modulation (PWM) and control flight surfaces for pitch, roll and yaw motion. Some I/O pins are also connected to the regulator (ESC) to control a DC brushless motor of the aircraft. The ESC regulator may be controlled by PWM or digital protocol e.g. DSHOT.

The base part of the **FC firmware** is the software PID regulator and the regulation loop. Inputs are data from sensors (IMU, barometer, GPS modul) and commands from RC transmitter/receiver. The regulated quantity is the signal to the servos and ESC regulator of motor. This way can be reached the knowledge of 3D position and ability to autoregulated motion of the aircraft. As the FC firmware I mostly use open iNAV or I develop own FC firmware using STM32.

The RC models or drones with the FC are capable of auto balanced flight with RTH (Return To Home point as failsafe during loss of the control signal) or Way Points mission capabilities. The great advantage of FC is rich flight information from embedded and connected sensors. **Telemetry data** are on-line transmitted to a ground station at the pilot. The ground station receive, decode and display flight data (speed, altitude, tilt, distance etc.). Can be used suitable protocols where telemetry data are coded, I use serial LTM (Light Telemetry Protocol) and MSP (Multiwii Serial Protocol). Telemetry data are transmitted via radio ISM band. Can be used 433/868 MHz (e.g. LoRa) or 2.4 GHz for short distance (Bluetooth or WiFi). There is also option to mix the flight video from a small board camera and telemetry data from FC sensors as OSD (On-Screen Display) and the final video signal transmit via board video transmitter (VTX) e.g. from ImmersionRC, received on the ground and display on a small monitor. The video signal is transmitted often via 5.8 GHz radio band. This is useful for FPV flights.

Used Commercial Flight Controllers (FC)

MATEK F405 WING + iNAV (MATEKF405SE) - the best for „fixed wing“ models (*update 2021 the best is **Matek F722-WPX** in my opinion now*)

- MCU: STM32 F405 (Cortex™-M4 core, 168 MHz, 192 KB RAM, 1 MB FLASH)
- IMU: MPU6000 accelerometer/gyro (SPI)
- Baro: BMP280 (I2C)
- OSD: iNAV OSD w/ AT7456E chip
- Blackbox: MicroSD card slot (SD/SDHC)
- VCP & 6x UARTs (e.g. for GPS modul , RC receiver, Telemetry modul, etc.)
- 9xPWM (2x Motors, 7x Servos outputs)
- 2x I2C
- 3x LEDs for FC STATUS (Blue, Red) and 3.3V indicator(Red)
- Built in inverter for SBUS input (UART2-RX)
- PPM/UART Shared: UART2-RX
- SoftSerial on TX2 pad
- Battery Voltage Sensor: 1:10 (Scale 1100)
- WS2812 Led Strip : Yes
- Beeper : Yes
- RSSI: Yes
- Mounting: 30.5 x 30.5mm, Ø4mm with Grommets Ø3mm , Dimensions: 56 x 36 x 13 mm Weight: 25g

MATEK F765 WING + iNAV (MATEKF765) - expensive and quite heavy FC board

- MCU: STM32F765VIT6 (Arm® Cortex®-M7 core, 216MHz , 512KB RAM, 2MB Flash)
- IMU: MPU6000 (SPI1) & ICM20602 (SPI3)
- Baro: BMP280 (I2C2)
- OSD: AT7456E (SPI2)
- Blackbox: MicroSD card slot (SDIO)
- 7x Uarts (1,2,3,4,6,7,8) with built-in inversion (e.g. for GPS modul , RC receiver, etc.)
- 1x Softserial1_Tx (iNAV)
- 12x PWM outputs (S1~S10 support Dshot)
- 6x ADC (VBAT, Current, RSSI, Analog AirSpeed, VB2, CU2)
- 3x LEDs for FC STATUS (Blue, Red) and 3.3V indicator(Red)
- 2x I2C
- 1x SPI4 break'out
- Switchable Dual Camera Inputs
- Switchable 5V/9V(12V) for Camera/VTX
- High-precision Current Sense
- ADC VB2 voltage divider: 1K:10K
- ADC AirSpeed voltage divider: 10K:10K
- TR/SA VTX control: Yes
- WS2812 Led Strip : Yes
- Beeper : Yes
- RSSI: Yes
- Analog Airspeed sensor: Yes, Digital Airspeed sensor: Yes

AIRBOT OMNIBUS F4V6 + iNAV (OMNIBUSF4V6) – suitable for telemetry/FPV

- MCU: STM32 F405
- BARO: BMP280
- IMU: MPU6000 SPI
- 6PWM output
- 5x UART (e.g. for GPS modul , RC receiver, Telemetry modul, etc.)
- 5V1A BEC on board(3-6s) 8V BEC Camera
- Flash, Only 36x36mm, holes 30.5x30.5mm

AIRBOT OMNIBUS F4Pro(V3) + iNAV (OMNIBUSF4PRO) /Betaflight (OMNIBUSF4SD) - few UART ports, suitable for telemetry/FPV

- MCU: STM32 F405
- BARO: BMP280
- IMU: MPU6000 SPI
- 6PWM output
- 3x UART (e.g. for GPS modul , RC receiver and Telemetry modul)
- On-Board OSD (FC over SPI bus)
- MicroSD Blackbox
- 5v3A SBEC
- Built-in Current Sensor
- On-Board Video Filter 5V to VTX and Camera

SucceX-E F4 V2.1 + iNAV/Betaflight - suitable for FPV

- MCU : STM32F405
- Gyro : MPU6000
- Baro: BMP280(I2C)
- Betaflight OSD : IFLIGHT OSD (AT7456E)
- Blackbox : 16MB(W25Q128)
- 4X Uarts
- 4 x Dshot/Proshot/Oneshot output
- 4x0201 LEDs for FC STATUS (Blue, Red, Orange) and 3.3V indicator(Red) BAT/+5V/3.3V/start
- 1 x Group of Gnd/Vbat/M1/M2/M3/M4/Cur/TX
- VTX telemetry: YES
- WS2812 led Strip : YES
- Beeper : YES
- Input : 26S (Max 8S)
- BEC : 5V / 3A
- Firmware target : IFLIGHT_SUCCEXE F4
- Mounting : 30.5*30.5/φ4, Dimensions : 36*36mm Weight : 6.9g

Holybro Kakute F7 mini V3 + Betaflight - suitable for telemetry

- MCU: STM32F722RET6 32-bit processor,216MHz, 256Kbytes RAM, 512Kbytes Flash
- IMU: MPU6000 (SPI)
- Barometer: BMP280
- USB VCP Driver (all UARTs usable simultaneously; USB does not take up a UART)
- 5hardware UARTS (UART1,2,3,4,6)
- Supports serial receivers (SBUS, iBus, Spektrum, Crossfire) only. PPM and PWM receivers are not supported.
- Onbord 16Mbytes for Blackbox logging
- 5V Power Out: 1.0A max
- 3.3V Power Out: 0.2A max
- Dimensions: 30x30mm
- Mounting Holes: Standard 20mm square to center of holes(M4)
- Weight: 4g

Others Components

- **RC Receiver:** SPEKTRUM Serial Receiver 2,4 GHz SRXL (4649T) or SRXL2 (SPM4650)
- **GPS Modul:** u-blox NEO 6M or Beitian BN280, BN220, BN880
- **Video FPV Transmitter** 5,8 GHz (VTX): Immersion RC TRAMP HV
- **Camera FPV** : RunCam Robin, FOXEER Arrow Mini Pro
- **Video FPV Receiver** 5.8 GHz: RC700D (diversity)
- **Telemetry:** HC-06 Bluetooth, HC-12 433 MHz, E32-TTL-100 LoRa 433 MHz, TTGO ESP32 LoRa/LoRaWAN 866 MHz
- Digital ESC DYS Aria 35A 32bit + firmware BL_Heli32 (v.32_64) + SBEC (switched BEC) FOX Y UBEC 5V/3A
- RC Transmitter SPEKTRUM 2,4 GHz 8G2

Nejdříve jsem použil desku FC **AIRBOT OMNIBUS F4V6**. Desky (35x35) FC typu OMNIBUS jsou zejména určeny pro řízení koptér než křídlových (FIXED WING) modelů se servy. Deska je napájena z 3-6S LiPo a obsahuje MCU STM32, BEC pro napájení periférií 5V, čip barometru a čip gyra + akcelerometru (IMU), čip OSD a senzory napětí a proudu. Z MCU je vyvedena šestice seriových UART (Rx/Tx) a šestice PWM výstupů pro řízení serv a regulátoru (ESC) motorů.

Do FLASH paměti desek lze po přepnutí MCU do DFU modu (stisk tlačítka BOOT při zapnutí napájení) „nahrát“ přes USB pomocí Betaflight či iNAV konfigurátoru firmware Betaflight <https://betaflight.com> nebo iNAV <https://github.com/iNavFlight>. Desky mají USB bootloader (do WIN je nutné doplnit driver) a výborný parametrizovaný firmware Betaflight či iNAV lze nastavovat přes přehledné grafické konfigurátory, které obsahují i příkazovou řádku - CLI. Tento popis se přímo konfigurací FC přes iNAV/Betaflight nezabývá - oba firmware mají výborně zpracovaný popis a manuál. Pro přenos konfiguračních dat mezi těmito firmware a jejich konfigurátory se používá seriový protokol MSP.

K jednomu UART jsem připojil **GPS modul (ublox NEO-6M)** a k video vstupům/výstupům OSD **FPV kameru RunCam a VTX ImmersionRC Tramp HV (5.8 GHz)**. Vše příslušně v Betaflight konfigurátoru (UART, GPS a OSD) nastavil. V Betaflight/iNAV lze jednoduše editovat umístění telemetrických hodnot OSD vložených do videesignálu a fonty znaků.

Na RC model Dynamics 1500 jsem **zatím jen ve funkci OSD pro FPV** připevnil desku OMNIBUS F4V6 s GPS modulem, FPV kamerkou RUNCAM a video vysílačem VTX Tramp Immersion. Zkouška letu s touto FPV sestavou se vyvíjela velice pěkně. Obraz z letové kamery na pozemním FPV monitoru s přijímačem (5.8 GHz) s vloženými OSD textovými údaji telemetrických hodnot z FC je super. Oproti jednoúčelovému OSD (např. Tarot) je více letových údajů, grafické vario a umělý horizont. Vše lze jednoduše konfigurátorem rozmístit a upravit font.

Seriový **RC přijímač** Spektrum 4649T (pro až RC 15 kanálů) jsem připojil k jednomu z UART na FC. Používá seriový protokol SRXL a firmware ho umí na většinu UART automaticky rozpoznat. Přijímač má navíc možnost přenášet zpět do vysílače telemetrická data z FC pomocí bi-directional SRXL protokolu. *Aktualizce 2021 – Spektrum již používá pouze SRXL2 4650/4650T - iNAV i Betaflight SRXL2 již podporuje.*

Na **PWM výstupy** FC jsem dále napájel piny a na jednotlivé PWM výstupy připojil ESC motoru a serva. V Betaflight jsou defaultně konfigurací výstupy mapovány pro mix čtyř ESC motorů coptery a lze je konfiguračně „přemapovat“ (příkaz "resources") pro mix čtyř serv a jednoho či dvou ESC motorů (příkazy "smix" a "mmix"). Ještě doplním minimálně tři letové mody MANUAL, ACRO/AIR, HORIZON přepínatelné dalšími RC kanály (AUX).

Desku kontroleru OMNIBUS F4V6 jsem přeflášoval na firmware **iNAV**. iNAV spolupracuje lépe s GPS. Nicméně bez úpravy zdrojáku a překladu neumí konfiguračně přemapovat typy výstupů (motor/serva) firmware určeného pro konkrétní desku FC. Deska OMNIBUS F4V6 není v iNAV zatím přímo uzpůsobena pro modely se servy a tak mi tu lze „namapovat“ PWM výstupy jen pro dvě serva a jeden motor. To na let postačí - mám ovládané a regulované PWM výstupy na **motor/throttle, křídélka/roll a výškovku/pitch a směrovku/yaw** zatím oželím. Použiji tři letové mody dle úrovně RC kanálů AUX a k nim přiřazeným poloh přepínačů - MANUAL- ovládání serv/motoru přímo z rc kanálů, ACRO/RATE - stabilizované výstupy (defaultní mod), HORIZON pro autostabilizaci. Doplnil jsem i mody využívající GPS - NAV CRUISE a NAV ALTHOLD (stabilizace kurzu a výšky) a NAV RTH (Return to Home). RTH (bez AutoLanding) jsem přiřadil také jako failsafe při ztrátě RC signálu. Později dojde i na test NAV WP (Way Points).

Regulace výstupů je pomocí **PID**. Regulační veličinou je defaultně střída PWM signálu (1000-2000 us při 50 Hz) pro ESC motoru a serva řídících ploch s cílem stabilní polohy v prostoru. Velikost regulační odchylky je ovlivněna vstupy z RC kanálů z přijímače a hodnotami z senzorů - barometru, gyra, akcelerometru a případně GPS. Zpětná vazba je počítána proporcionálně (P), integračně z hodnot předchozích údajů (I) a derivací (D) okamžitých změn. Jednotlivé úrovně PID regulace je možno ovlivnit - násobit koeficientem. Koeficienty PID a další hodnoty regulace (např. rychlost otáčení kolem jednotlivých os – rates deg/sec) se nastavují konfigurátorem. Správné nastavení PID je poměrně věda, velké P má za následek rozkmitání výstupů, malé I stálou regulační odchylku, atd. Vše je komplikováno vibracemi z motorů, šumem, atd.

Desku Omnibus F4V6 + iNAV **již ve funkci stabilizace** jsem zkusil na Dynamicsu 1500. Nad očekávání úspěšně i bez směrovky. Po zapnutí ARM modu ("odjištění" motoru) a autonastavení Home Position přes GPS jsem v MANUAL módu model odhodil a přešel na mod ACRO/RATE (defaultní stabilizace). Stabilizace byla účinná, větrné porывy model nerozhodily. Pak jsem zapnul mod HORIZON a model se krom stabilizace snažil držet pozici. Byla tendence klesat - chce to lépe usadit FC a doladit hodnoty PID a rates. Bonusem FC je bohatá telemetrie, OSD pro FPV, záznam BlackBox dat, umělý horizont a statistika letu na závěr.

Mírný nárazový vítr je ideální čas na test dalších letových módů s Dynamicsem a FC Omnibus F4V6. Pomocí funkce „blackbox“ lze zaznamenávat letové údaje a údaje gyra na interní 16MB flash v FC, po přistání data stáhnout a dekodovat si je např. toolom iNAV Blackbox Explorer. Lze pak zkoumat údaje telemetrie a stavy gyra. Na black box záznamu z dnešního letu názorně vidím chování modelu s FC v různých letových modech. Při manuálním módu **MANUAL**, s minimem ručních zásahů do řízení, je let díky větru dost rozháraný. Jakmile se zapne **RATE** a začne fungovat stabilizace vychylek, let se uklidní. S modelem **ANGLE/HORIZON** se model se snaží držet pozici. Po zapnutí modů **NAV CRUISE + NAV ALTHOLD** je jednak model stabilizován a navíc se snaží držet kurs i výšku dle GPS. Pak zapínám mod ReturnToHome (**NAV RTH**) a model se sám poslušně otáčí do směru k startovní GPS pozici o ARM resp. GPS pozice zaznamenané volbou ResetHomePosition. V modu **NAV POSHOLD** krouží na místě.

Jakmile bych vypnul RC vysílač, model by přešel do failsafe a na RTH, případně i na přistání (autoland). Ten poslední bod si nechám na někdy později.

Pořídil jsem (cca 500 Kč) ještě letový kontroler **AIRBOT Omnibus F4ProV3**. Tato deska je o něco starší než F4V6, ale má přímou podporu v současné verzi iNAV (2.1) i pro mix 4 serv a 2 motor. Mohu na ní zapojit všechny výstupy, tedy všechny ovládací plochy a dva motory. Tento FC obsahuje i integrovaný proudový snímač dimenzovaný na vysoký odběr. Proud pro ESC jde přes tento snímač a měří se tedy celkový proud z AKU - zobrazují jej v OSD.

Nakonec jsem pořídil (cca 1000 Kč) i desku **FC Matek F405 WING** (tedy přímo uzpůsobenou pro modely letadel), nahrál na ni poslední iNAV a po nastavení provozuji na dvoumotorovém Tinstaru a jednomotorovém EasyStar. Po pevném ustavení desky na trupu ve správném směru jsou stabilizační i navigační módy bez chyby. Je nutné doladit parametry PID (P=10-15, I=10-20, D=0) a rates (kolem 100 deg/sec). Modely po vyrovnaných letech v nastavené výšce i kurzu až do vzdálenosti více jak 600 m se vždy s pomocí NAV RTH poslušně vrací nad místo vzletu. Pak krouží v modu NAV POSHOLD nad spokojeným "autopilotem".

Doplňuji k FC **pásek s 10 x RGB LED**. Tím dostávám pomocí barev jednotlivých LED signalizaci stavu AKU, GPS, ARM, letových modů, orientace v prostoru či systémových warnings. Lze doplnit bzučák na zvukovou signalizaci.

K deskám FC dále připojuji přes další UART radiový modul HC-12 nebo LORA na 433/866 Mhz pro **zasílání telemetrie** protokolem LTM (viz dále) nebo pro online upload Way Points.

Deska s FC musí být v trupu správně orientovaná a pevně usazená. RC přijímač, GPS modul, VTX s heli anténou pro 5.8 GHz nebo případně vysílač 433MHz (např. LoRa – viz dále) pro telemetrii s anténkou mám upevněny většinou na trupu, FPV či HD kamera je vpředu na přídi. Vše většinou stačí upevnit sucháči. Přívody od serv, ESC, napájení a radio modulů jsou protažené k FC.

Důležité je též zkaliбrovat akcelerometr a srovnat výchylky serv ve správném směru dle vyrovnávaného pohybu z gyro snímače - pohyb ocasu nahoru -> výškovka vyrovnává pohybem nahoru, trhnutí ocasem dolů -> výškovka vyrovnává pohybem dolů, náklon vpravo dolů -> pravé křídélko vyrovnává pohybem dolů a levé nahoru, otočení příďe vpravo -> směrovka vyrovnává otočením vlevo.

Při tomto testu je vhodné dočasně zvětšit P a I zisky, aby pohyby ploch byly větší a trvaly déle - nezapomenout vrátit se k původním hodnotám PID. Směry výchylek lze srovnat kombinací případného reverzu na RC vysílači a reverzu v nastavení serv v iNAV. Na pravý náklon, pravé točení a klopení nahoru (přitažení) na vysílači, se musí RC signál v FC vždy zvyšovat.

PID nastavuji cca P-10, I-12, D-0. Rotace roll 150, pitch 100, yaw 90 deg/sec. V případě kmitů je nutno zmenšit P, v případě stálé odchylky směrů pomalu zvětšovat I. iNAV obsahuje autotunnig, ale je lepší si s nastavením zisků PID a hodnot rates (rychlostí rotace kolem os) pohrát při pokusných letech.

Desky **Omnibus F4V6 a F4Pro** a podobné FC určené zejména pro koptery (rozměry 35x35 nebo 20x20 mm), používám s firmware Betaflight či iNAV pro zasílání telemetrie nebo pro FPV a nikoliv pro stabilizaci/navigaci modelu s křídly.

FC MATEK (F405, kompaktní F722 WPX atd.) s firmware iNAV (v 7/2021 již ve verzi 3.0) jsou výborné pro křídlové modely a jejich stabilizaci/navigaci zejména. Na pokusy i „vážnější projekty“ si vyvožím s vlastními řešeními HW (STM32) a firmware.

Digitální ESC protokol DSHOT

Pro řízení motoru přes ESC na křídlových modelech stačí klasická analogová PWM modulace, ale pro řízení motoru pomocí FC je už PWM "zastaralé", zejména pro koptery. Nové ESC protokoly - digitální DSHOT - jsou zesynchronizovány s regulační smyčkou PID a motory rychleji reagují na změny. **DSHOT** je digitální protokol mezi FC a ESC. Délka jednotlivých pulsů v cyklu definuje bit, tedy 0 nebo 1. V jednom cyklu má 11 bitů (2000 úrovní) na řízení throttle, jeden bit jako požadavek na seriové slovo telemetrie přes separátní seriový kabel zpět do FC a 4 bitové CRC. U DSHOT150 je délka cyklu cca 0,1 ms, je tedy 200x rychlejší reakce pro řízení throttle než 20ms u běžného PWM. Navíc digitální ESC s podporou DSHOT umí posílat telemetrii (otáčky).

U kopter jsou již digi ESC skoro nezbytnost. Ale i FIXED WING modeláři mohou přeci ve větší míře využívat výdobytků kolegů kopterářů. Pořizují si digitální **ESC DYS Aria 35A**. Ten obsahuje 32b CPU ARM s [firmware BLHeli_32](#).

Nové verze [iNAV](#) by měly podporovat **ESC Telemetrii**, tedy posílání telemetrie z datového vývodu ESC (např. můj nový regulátor DYS ARIA s 32b MCU s firmware BLHeli_32 ho má označen TE) na volný UART na FC. Zkouším DSHOT zatím jen s Betaflight na F4V6 Omnibus. Nastavuji Betaflight na ESC DSHOT a propojuji UART4 s datovým výstupem "TE" na desce ESC. Daří se pak zobrazit v OSD Betaflightu otáčky motoru (RPM), teplotu desky ESC a napětí. Hodnoty proudu ale ne. Laboruji ještě s výstupem na destičce DYS s označením "CU" (analogový CURRENT ??) a jeho připojením na analogový vstup pro proudový snímač na desce F4V6. Popis padů pro ESC telemetrii v manuálu DYS ARIA chybí. Snímat proud se mi nedaří. Nyní ESC telemetrii zkouším s iNAV, ale tam se mi vůbec nedaří udržet v nastavení protokol DSHOT, který je pro ESC telemetrii samozřejmě nutný. Po nastavení DSHOT a restartu FC je v iNAV opět analogový protokol (PWM nebo OneShot). Podpora DSHOT u Omnibus s iNAV zatím nefunguje. Poslal jsem issue na iNAV do GitHub, zatím bez odpovědi. *Update – vše řeší nová verze iNAV*. Zkouším i jiné hodnoty nastavení BLHeli pomocí jeho konfiguratoru BLHeliSuite_32. Nastavení firmware v ESC lze provést tzv. passthrough připojením digitálního ESC s firmware - ESC se připojit k motoru i k desce FC s iNAV/Betaflight a v konfigurator BLHeliSuite se nastaví v menu „bootloader Betaflight“. Konfigurace firmware v ESC pak probíhá přes USB na FC.

Nové verze [BETAFLIGHT](#) podporují nyní i **Bidirect Telemetry s DSHOT**, zkouším i to. Pro funkční „bidirect“ je nutné flashnout ESC DYS BLHeli_32 na poslední verzi (nyní 32.7) a desku FC F4V6 na nový BetaFlight (nyní 4.1). Datové pakety telemetrie jsou pak obsaženy přímo v odpovědích DSHOT regulačního protokolu a není nutný zvláštní seriový datový propoj z ESC na UART desky FC. BF umí data z paketů DSHOT načíst a online nasnímané hodnoty otáček resp. frekvence napětí a proudu zobrazit, např. v OSD. Navíc nasnímané hodnoty používá firmware jako vstupní hodnoty pro dynamické nastavení softwarových frekvenčních filtrů (RPM filtrů). Dynamické filtry analyzují gyrosignál a ořezávají šumy. Tedy vyšší frekvence signálu (LPF-LowPassFilter, např. od 90Hz) nebo vyřezávají některé (NotchFilter) frekvence signálu (200-300Hz). Tím umožňují zvýšení odstupu užitečného signálu gyra od šumu doplněného z vibrací motorů/konstrukce trupu a tím zpřesnění PID regulace. Filtr RPM umožní dynamicky stanovit filtrovanou frekvenci dle aktuálních otáček motoru a tím pravděpodobných vibrací přenášovaných na gyro.

Telemetrie

Chci na pozemní stanici (**GCS - ground control station**) on-line přijímat telemetrická data z modelu - tedy alespoň GPS pozici, rychlost, výšku, kurs, palubní napětí/proud a náklony. Zároveň chci mít možnost online uploadovat GPS souřadnice z pozemní stanice do FC na modelu (Way Points) pro lety „Flight Mission“. Jako telemetrický protokol lze u FC s firmware Betaflight/iNAV použít **MultiWii Serial Protocol (MSP)**, **LTM (Lightweight Telemetry)** nebo **MAVLink**. Seriový telemetry protokol MSP je tahací a vyžaduje vyšší datovou rychlost (od 9600 bps). Aplikace v pozemní stanici si musí o paket dat nejdříve říci. LTM je tlačíci, automaticky v přednastavené rychlosti data posílá a tím je vhodnější pro nižší přenosové rychlosti (500 – 9600 bps). LTM v posílaných paketech poskytuje méně telemetrických dat, ale mně to dostačuje. V iNAV lze donastavit četnost zasílání LTM dat od 105 B/sec do 302 B/sec tak, aby korespondovala s používanou přenosovou rychlostí (pro 9600 mám 303 B/s). Kvůli radio duty cycle lze jít i níže.

Pro příjem telemetrie na pozemní stanici použiji např. free **mwptools** na LINUXu nebo aplikaci **Mission Planner** na mobilu s Android. Ve Windows lze použít iNAV Configurator. Lze si ale vyrobit vlastní HW i SW projekt.

Pro radiový přenos dat z modelu na zem použiji nejdříve bluetooth moduly (HC-06), pak zkusím radio moduly pro radiový přenos v bezlicenčním pásmu 433 MHz (např. HC-12) a cílově použiji protokol LoRa (LongRange) např. s moduly E45-TTL-100. LoRa používá efektivnější modulaci CSS (Chirp Spread Spectrum) než HC-12 (modulace PSK) a zvládne tak větší dosah. Moduly lze získat v „Arduino shopech“ za cenu v řádu desítek/stovek Kč.

Nejdříve tedy jen BT moduly HC-06. Ty je vhodné nastavit na max. UART rychlost 115200 bps (defaultně má HC-06 nastaveno 9600 bps). K tomu stačí modul USB/UART převodníku (např. CP2102). Stačí ho seriově propojit s HC-06 (Vcc->5V, GND->GND, Rx->Tx, Tx->Rx), strčit do USB PC či notebooku, nainstalovat tam driver pro virtuální COM (VCP driver) a z nějakého seriového terminálu (např. Termit) přes COM port (9600/8N1) do modulu poslat konfigurační AT příkazy. A je tu první zádrhel – můj HC-06 má novou firmware (3.0) a všude dokumentované AT komandy vrací chyby. Nová verze má AT příkazy pro změnu rychlosti "AT+UART=115200,0,0", pro změnu jména "AT+NAME:xxx" a nastavení pinu "AT+PSWD:"PIN"".

V konfiguraci iNAV jsem na volném UART (v mém případě UART2) nastavil telemetry **protokol MSP** a 115200 bps a BT modul HC-06 připojil k UART2 na FC (5V, GND, Tx->Rx, Rx->Tx). UART TX výstup z FC na RX vstup modulu by se měly připojit přes **převodník logických úrovní 5/3.3V** nebo odporový dělič z 1k/2k - logika modulu používá 3.3V. To platí i u všech dalších radio modulů.

Ted' přijde instalace **mwptools** na notebook. Na WIN instaluji VMWARE Workstation 15 a do ní virtuální Ubuntu 18.04. S VMWARE je u mě další zádrhel – současné Win10 mají v jádře ochranu "Device/Credential Guard" a ta si s VMWARE Workstation nerozumí. Zabrala kombinace úpravy registrů, příkazů v powershellu a úpravy bootu pomocí bcdedit. Nakonec ale po instalaci Ubuntu do virtuálu, doplnění potřebných balíčků, zkopírování mwptools z Githubu a "make && sudo make install" je mwptools na světě.

Po bluetooth spárování (ve starší verzi) Windows resp. virtuálního Ubuntu s BT modulem HC-06 se zasílání telemetrie z FC na pozemní stanici rozběhlo. V grafice mwptools jsou pěkně zobrazená telemetrická data ze snímačů připojených k FC na modelu, je tu umělý horizont, trasa GPS na mapě a hlasový výstup. Je možné online do FC poslat až 60 WP bodů pro Flight Mission. Aplikaci mwptools si lze přizpůsobit pomocí klíčů v Gsettings (schema je org.mwptools.planner) v Linuxu.

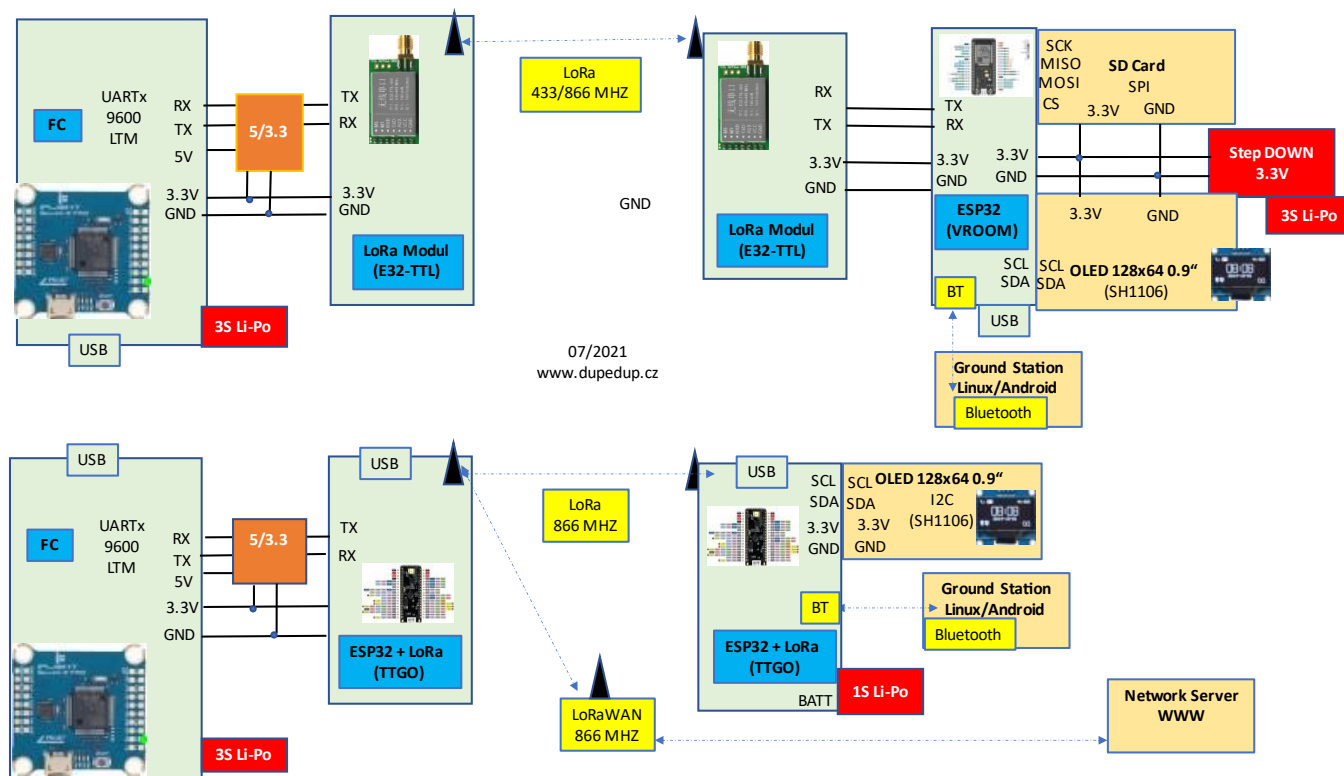
Bluetooth má dosah jen do 10m, proto přistupuji k propojení přes 433 Mhz pomocí **dvou radio modulů HC-12** s malou anténkou pro toto pásmo. Při dobré konstelaci a nízké datové rychlosti je dosah až 500 m. Parametry lze u těchto modulů nastavit také pomocí AT příkazů. Výkon lze nastavit až na 100 mW (v ČR je povolených 15 mW v pásmu 433 MHz a duty cycle 1%, což asi nedodržíme). Rychlost UART na modulu použiji defaultních 9600 bps (lze přenastavit pomocí AT+Bxxx), což je dle manuálu rychlost 15000 bps „in air“.

Vysílací modul HC-12 připojuji k seriovému portu na desce FC (UART2). V iNAV na tomto UART nyní nastavuji telemetry protokol **LTM** a 9600 bps. Lze použít současně i MSP na 9600 bps. iNAV použije nejdříve MSP a po odjištění motorů (ARM) se změní na LTM. Druhý „pozemní“ radio HC-12 modul propojuji přes UART 9600 bps s osvědčeným Bluetooth modulem HC-06. Ten musím zpět pomocí AT přenastavit z 115200 na 9600 bps. Přes takto vytvořenou „433MHz/bluetooth gateway“ zasílám data telemetrie na zmiňovaný notebook s mwptools v Linux distribuci Ubuntu. Gateway HC-12/HC-6 napájím step down modulem 3,3V z 2-3 článkového LiPo akumulátorku.

Po bluetooth spárování se mwptools chytlo protokolem LTM (nebo nejdříve MSP a následně po ARM na modelu na LTM) a zobrazuje telemetrické údaje.

Jako poslední jsem vyzkoušel a nyní používám radiové propojení přes radiové **LoRa moduly** na 443/886 MHz s anténkou pro tato pásma. Ideálně **E32-TTL-100**, které mají UART interface. Lze použít i ESP32 s LoRa (TTGO) nebo podobné. Vysílací radiový LoRa modul je připojen k FC přes UART na LTM a max. 9600 bps. Přijímací LoRa modul je přes UART 9600 bps připojen buď opět pouze k bluetooth modulu HC-06 nebo pomocí k UART k MCU ESP32 (VROOM) nebo TTGO. V MCU lze pomocí vlastního firmware vstupní data upravovat, připojit OLED display na přímé zobrazení dat nebo SD kartu V testu zatím en telemetrická data „přesměrovávám“ přes bluetooth na pozemní stanici. Pro tu používám zmiňovaný mwptools na Linuxu v NB nebo aplikaci EZ-GUI v Android na mobilu.

Jak jsem zmínil, vhodným „arduino compatible“ MCU a vlastním firmware si lze vytvořit vlastní projekty pozemní stanice bez nutnosti použít mobil nebo notebook. Jako příklad takového řešení jsem si vytvořil malou přijímací stanici, která je tvořena **MCU ESP32 VROOM32D**, LoRa modulem k ESP32 připojeném přes UART, OLED displayem (I2C interface) a SD čtečkou (SPI interface). Vše je napájeno 3.3V z napěťově stabilizačního modulu step-down připojeného k 3S LiPo aku. Do ESP32 jsem naprogramoval firmware, který parsuje přijmutý LTM datastream na hodnoty telemetrie, které zobrazuje na display, zapisuje do CSV souboru na SD a i posílá LTM stream do bluetooth. Tak je stále možné data přijímat a zobrazovat pomocí nějaké grafické appky na mobilu či na NB. Pokud je v místě signál LoRaWAN, je možné radio LoRa modulem dat posílat data i na bránu LoRaWAN. Samozřejmě ve větším intervalu (menším datovém toku) a v orientačním množství - např. občasná GPS data. Z řídicího serveru LoRaWAN lze pak data přes callback a http POST dále přeposílat na REST aplikaci na vzdáleném síťovém serveru a zde zaznamenávat. Takový "remote on-line blackbox".



Na závěr mávám čínským výrobcům modulů, nemávám ČTÚ a smekán před vývojáři iNAV, mwptools, LTM a dalších GNU knihoven a projektů na GitHub.