Checking the Flight Stability of a Rotary UAV in Navigation Modes for Different Firmware

Serhii Lienkov [1], Alexander Myasischev [2], Olexander Sieliukov [3], Oleksandr Pashkov [4], Genadiy Zhyrov [5], Andrii Zinchyk [6],

[1] *Military Institute of Taras Shevchenko National University of Kyiv, Lomonosova street, 81, Kyiv, 03189, Ukraine*
[2] *Khmelnytskyi National University, Instytuts'ka str. 11, Khmelnytskyi, 29016, Ukraine*
[3] *State enterprise "Scientific center of accurate machine building", Boryspylska street, 9, Kyiv, 02099, Ukraine,*
[4] *Ministry of Defense of Ukraine, Povitroflostky Ave, 6, Kyiv, 03168, Ukraine*
[5] *Department of the Radio Engineering and Radioelectronic Systems of Taras Shevchenko National University of Kyiv, Volodymyrska str., 64/13, Kyiv, 01601, Ukraine*
[6] *Military Institute of Taras Shevchenko National University of Kyiv, Lomonosova street, 81, Kyiv, 03189, Ukraine*

Abstract

In this work, the stability of a rotor-type UAV based on the STM32F405 microcontroller was studied for flying in a fully automatic mode along a given trajectory during a gusty wind with an average speed of 6-7 m/s. The INAV software with firmware ver.2.6 and the Ardupilot software with arducopter firmware ver.4.0.5 were used. The possibility of correct formation of the flight trajectory in the semi-automatic mode for taking photographs, studying the radiation situation in the area, spraying fields, etc. has been established for Arducopter firmware. The development of flight mission for INAV is more primitive, it is performed only in manual mode, using preliminary calculations. The obtained results show that the INAV firmware with the UAV allows only telemetry data to be transmitted to the ground station. It is impossible to change the UAV flight trajectory from the ground station via telemetry. For the Ardupilot software, the possibility of changing the route is shown by dragging the flight point to which the UAV is approaching with the mouse pointing device. For the INAV firmware, it was experimentally not possible to connect flying over a given waypoint with the execution of an action, for example, turning on and off the sprayer nozzle, and dropping the load at a given point.

Keywords

OMNIBUSF4V3, INAV 2.6, GPS receiver, STM32F405, UAV, OSD, ESC regulator, FlySky FS-i6, Failsafe, Arducopter, Ardupilot, Pixhawk

1. Introduction

Currently, topical issues are the study of the radiation situation in the area, performing topographic and geodetic surveys for drawing up a plan of the area, spraying agricultural fields, carrying out rescue operations [1-4]. To solve these issues, unmanned aerial vehicles (UAVs) of both rotary type (quadcopters, hexacopters, orthocopters) [1] and with a fixed wing (aircraft, flying wings) [3] can be used.

To solve the problems under consideration, the UAV should be able to operate in a fully automatic mode when flying around an area along a given route and perform actions set before the flight, for example, turn on and off the nozzles when flying over the specified waypoints. The

ISIT 2021: II International Scientific and Practical Conference «Intellectual Systems and Information Technologies», September 13–19, 2021, Odesa, Ukraine
EMAIL: lenkov_s@ukr.net (A.1); alex56ma@gmail.com (A.2); selukov@3g.ua (A.3); torwer007@gmail.com (A.4); genna-g@ukr.net (A.5); andrik71@meta.ua (A.6).
ORCID: 0000-0001-7689-239X (A.1); 0000-0003-1269-425X (A.2); 0000-0001-7979-3434 (A.3); - (A.4); 0000-0001-7648-7992 (A.5); 0000-0003-4029-2282 (A.6)

software of the ground station should be able to automatically generate flight trajectories and actions performed during the flight over specified sites [5]. It is important that during its flight the vehicle should be sufficiently resistant to external influences, for example, to gusts of wind. It is necessary that the UAV flight is displayed on the screen of the ground station, superimposed on the geographical map of the area [6], as well as there will be a possibility of changing the route by dragging the flight point with the mouse pointing device, to which the UAV is approaching. It is advisable to use the FPV flight capability [7] with displaying telemetry data on the helmet screen for maximum control over the UAV flight. In [6, 8], other ways of using UAVs are presented, for which their stability in flight and positioning at a given point are also important.

2. Problem Statement

To solve the problems described above, it is very important that the UAV is maximally resistant to changing external impacts. For example, with a constant wind of 5-6 m/s, simplified mathematical models are used to ensure flight stability. An example would be the use of complementary filters for the joint operation of a gyroscope and an accelerometer and simplified models of PID regulators. However, with a dynamically changing wind of up to 15-20 m/s, it is required to use a more complex mathematical apparatus and develop software on its basis for the flight controller. In this regard, the work considers the construction of a quadcopter according to a well-known model (X-copter) and its adjustment for modern firmware INAV and Arducopter [8, 9]. The adjustment is carried out experimentally with enumeration of a large number of parameters (including adjustment of the PID regulators), on which the flight stability of the copter with the given geometric parameters as well as the parameters of the propeller group depends.

In this study, the most common firmware, INAV and Arducopter. These firmwares are free to use, open source and user-configurable. INAV supports a large number of flight controllers based on 32-bit microcontrollers of the STM32F4 and STM32F7 sets, positioning sensors (gyroscopes, accelerometers, magnetometers, etc.), and OSD. The Arducopter firmware was developed for flight controllers APM 2.6 and Pixhawk [10]. In this work, we have studied how the ported Arducopter ver.4.0.5 firmware practically functions on the OMNIBUSF4V3 controller [11] and how stable flight will be in navigation mode for such flight controllers based on STM32F4. The OMNIBUSF4V3 controller is also used for the INAV ver.2.6 firmware. Therefore, the testing conditions are the same. The use of free firmware and common cheap flight controllers makes it possible to design low-cost UAVs. For example, a copter assembled for experimental purposes is at a cost 2-3 times lower than commercial ones, which have similar flight modes. Pixhawk flight controller is about 5 times more expensive than OMNIBUSF4V3.

3. Research Design

For Arducopter versions, up to ver. 3.2.1, which was used by the autopilot based on atmega2560 microcontroller – for example, APM 2.6 and 2.8 – the Discrete Cosine Matrix (DCM) mathematical apparatus was used to estimate the position/orientation of the aircraft [12]. The DCM was also called the first generation attitude and position estimation system.

The current stable version of ArduPilot (4.0.5) uses EKF2 as the main source of orientation in space, with the DCM running in the background. If the autopilot has two (or more) available IMUs (the Inertial Measurement Unit contains a gyroscope and accelerometer), two EKF "cores" will work in parallel, each of which will use its own IMU. At any given time, only the output signal of one EKF core is used, and it is this core that reports the best state, which is determined by the consistency of the data of its sensors.

Figure 1 shows the evolution of the Attitude and Heading Reference System (AHRS) [13]

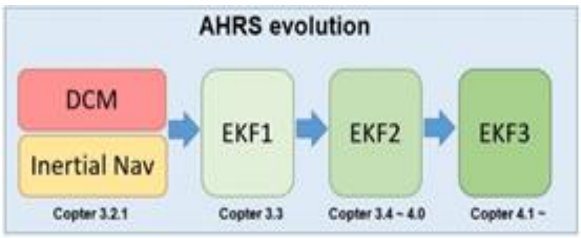

Figure 1: Evolution of AHRS for different versions of ArduPilot

INAV firmware for OMNIBUSF4V3 flight controller uses Alpha-Beta filter (complementary filter) for position estimation. It allows using the accelerometer and gyroscope to obtain fairly accurate pitch and roll values. However, with large vibrations of the motors, its accuracy becomes insufficient and this filter is not suitable for using another sensor, a magnetometer. That is why aircraft behave unstable when switching to fully automatic control with INAV firmware. Therefore, in this work, along with INAV firmware, the behaviour of the craft in automatic mode, using the Kalman filter, is investigated.

Another alternative is the Madgwick filter [14], which gives a better result than the Kalman filter in the results of position accuracy and calculation performance.

The Advanced Kalman Filter (EKF2) algorithm used in ArduPilot provides a way to combine or consolidate data from an IMU, GPS, compass, airspeed sensor, barometer, and other sensors to calculate a more accurate and reliable estimate of the UAV position, speed and angular orientation. This algorithm evaluates a total of 22 states from different sensors.

The issues of constructing mathematical models to ensure the stability of the UAV were also discussed in [15-17].

4. Basic Material and Working Results

For testing the UAV, a quadcopter with a 250-mm frame was chosen (Figure 2). On the quadcopter, the following equipment was installed: motor 2204/2300 KV with ESC regulator 30A; propeller 5×4.5 inches; GPS receiver: u-blox NEO-6M; control equipment: FlySky FS-i6 firmware upgraded from 6 to 10 channels with a communication range of up to 1.0 km; flight controller OMNIBUSF4V3 based on STM32F405 LQFP64 microcontroller (168Mhz, 1M Flash, 192kB SRAM) with built-in gyroscope, MPU6000 accelerometer and BMP280 barometer; battery 1500 mAh/hour, 11.1 V; video camera and video transmitter TS832 with a power of 0.6 W. The flight weight of the quadcopter was about 550 g.

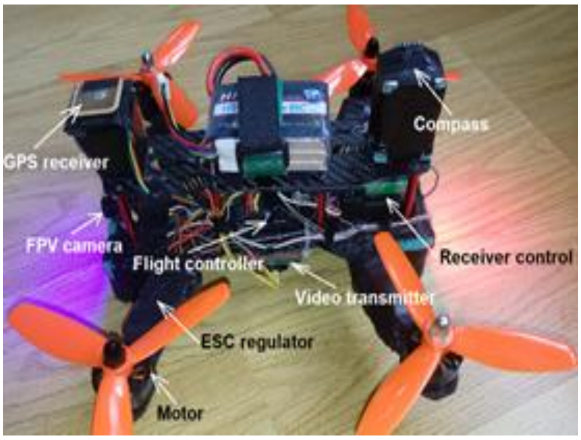

Figure 2: Photo of the tested quadcopter

The interconnecting network of the copter electronic components is given in Figure 3. The video camera is connected to a separate power source. The copter sends telemetry data, which are superimposed on the image of the terrain obtained from the video camera, through the video transmitter.

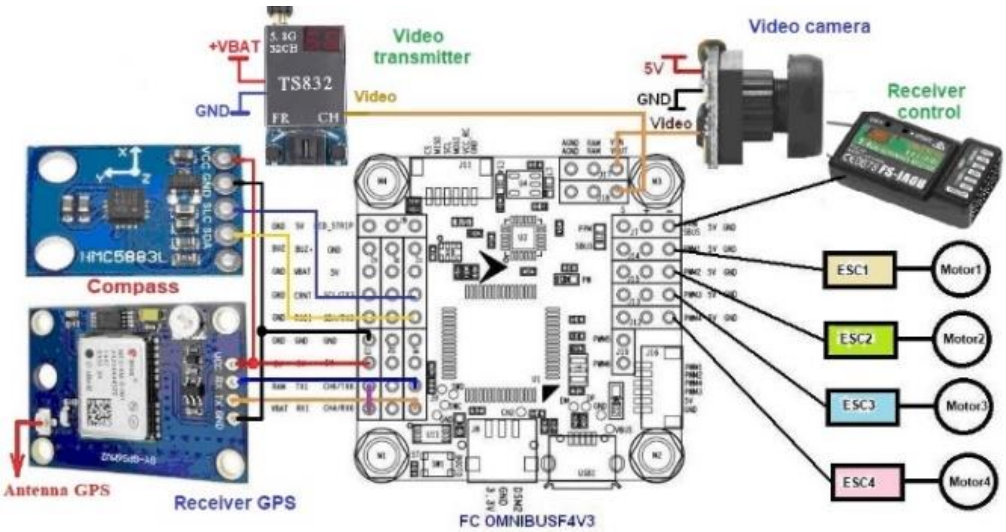

Figure 3: The interconnecting network of the copter electronic components

To configure the flight controller of the copter, the firmware file for the microcontrollers and the ground station software are used. The ground station uses Mission Planners, which is located at https://firmware.ardupilot.org/Tools/MissionPlanner/. The firmware for OMNIBUSF4V3 is copied from https://firmware.ardupilot.org/Copter/stable-4.0.5/omnibusf4/ under the name arducopter_with_bl.hex.

When sold OMNIBUSF4V3 controller is usually pre-flashed with Betaflight firmware. After Betaflight, INAV firmware is installed quite easy [9]. Setting up this firmware using the INAV navigator was considered in [18-20]. After its installation, test flights were performed with INAV ver.2.6. After completing the testing, arducopter firmware was installed. However, it is not possible to directly install arducopter firmware from Mission Planners, the ArduPilot ground station. To use ArduPilot, you need an ArduPilot compatible bootloader on the microcontroller. Therefore, _bl.hex file is used, which contains the firmware and bootloader compatible with ArduPilot. _bl.hex file can be flashed using the INAV configurator, which uses a convenient graphical interface for this purpose. Before flashing, the controller should be switched to DFU mode (Figure 4).

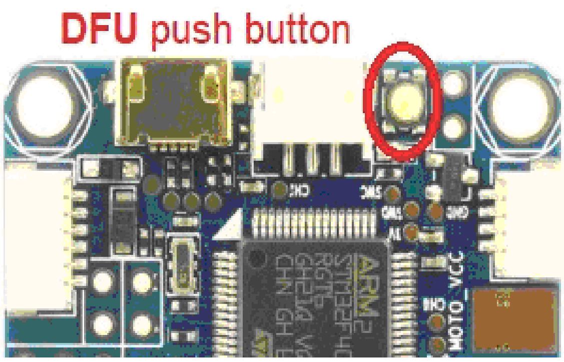

Figure 4: Switching OMNIBUSF4V3 controller to DFU mode

If OMNIBUSF4V3 controller is not recognized as a com port under Windows, zadig program is downloaded from http://zadig.akeo.ie/ and installed.

It should be noted that the peculiarities of the firmware for OMNIBUSF4V3 controller compared to the typical firmware developed for Pixhawk 2.4.6 flight controller based on STM32F427 Cortex M4 processor with FPU 168 MHz/256 Kb RAM and 2 Mb flash memory. The microcontroller on OMNIBUSF4V3 has 1 Mb flash memory, so its capabilities are limited. For example, some sensors and telemetry formats may not work; some devices cannot be controlled, etc. Lua scripting is not possible due to insufficient flash memory. These scripts provide a safe sandbox environment for adding new behaviour to the autopilot without changing the main flight code.

Despite this, the flight performance with OMNIBUSF4V3 is higher than with APM 2.6 based on ATmega2560 processor due to a more advanced mathematical model of flight control when using the same type of sensors.

The quadcopter firmware is configured using MissionPlanner ver. 1.3.74 application.

A computer is connected to the flight controller using a USB cable. Then the Mission Planner application is launched, which connects to the firmware of OMNIBUSF4V3 controller by clicking on CONNECT (1) in the upper right corner. The frame type is selected step by step (Figure 5).

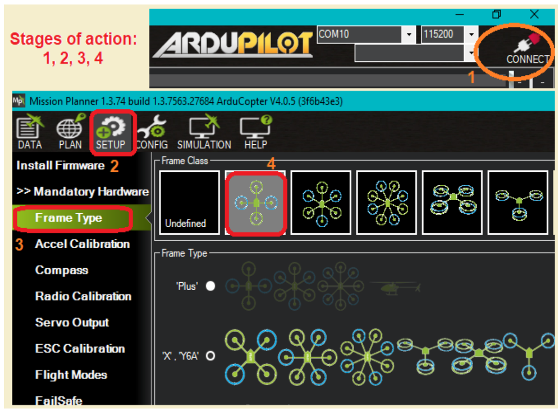

Figure 5: Selection of a frame type

The calibration of the accelerometer, compass, control equipment and ESC regulators is performed. Calibrations are performed according to the step-by-step instructions on the corresponding tab. Figure 6 shows a combination of setting tabs. To calibrate the accelerometer, the copter is sequentially installed to a fixed position along 6 axes, followed by fixing each of the positions. It is best to calibrate the magnetometer in the launch field by rotating the quadcopter in six axes until the Mission Planner reports the end of the calibration. The location of the magnetometer relative to the flight controller is determined automatically.

ESC controllers are configured as described in the ESC Calibration tab.

Before setting flight modes (Flight Modes) it is necessary to configure FlySky FS-i6 control equipment. In order to do this, it is first upgraded to a 10-channel operating mode [18,19]. Then, on the FlySky FS-i6 operation panel, go to the End points and Aux channels tabs, in which the parameters shown in Figure 7 are set for the 5th channel. The Reverse tab sets the reverse for the 2nd channel.

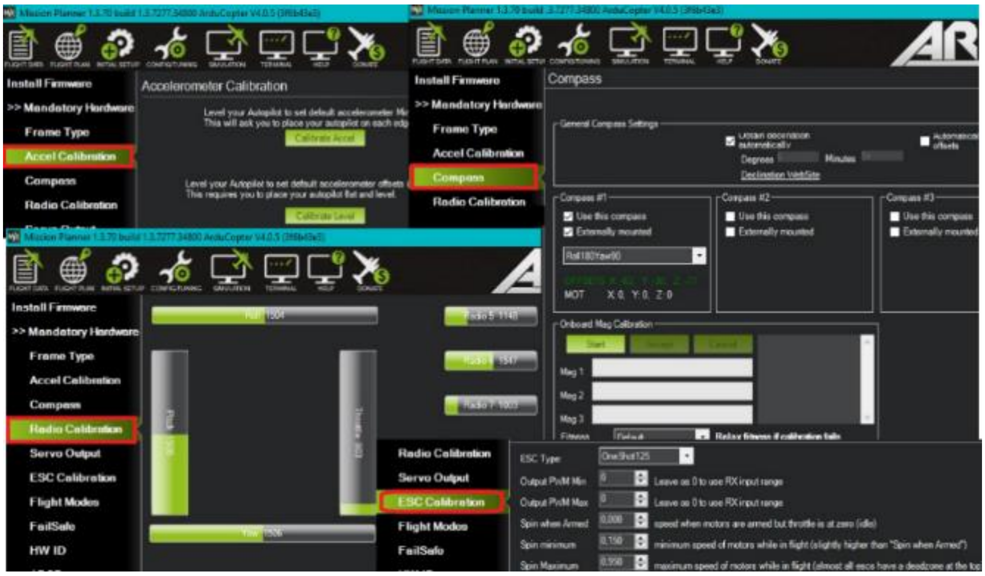

Figure 6: The combination of tabs for parameter settings

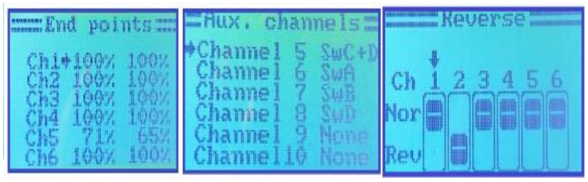

Figure 7: Setting parameters on the FlySky FS-i6 operation panel for the formation of flight modes

Such an installation will allow using the three-position SwC and two-position SwD switches to work with 6 different flight modes. Figure 8 shows the flight mode and Failsafe settings tabs. To implement Failsafe, the control hardware and receiver are preconfigured as presented here [18, 19]. FailSafe mode is set to minimum throttle. That is throttle value is equal to 1003 pulses when control equipment is on. If communication with the operation panel is broken, this value will be equal to 900 pulses. Fig. 8 shows that it is set to 950, below which FailSafe will be triggered. This will turn on the Enabled Continue mode with Mission and Auto modes, so the flight mission is continued in automatic mode.

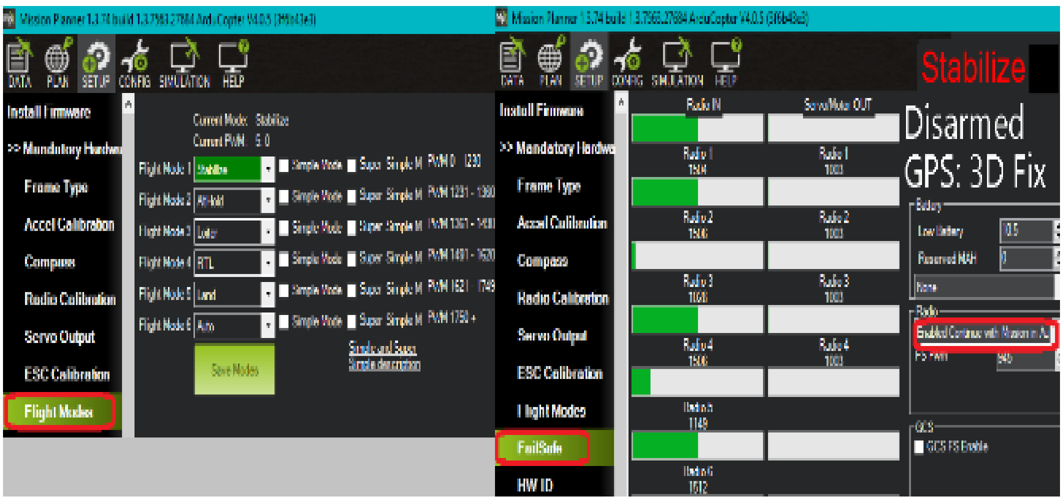

Figure 8: Setting flight modes and FailSafe

To control the battery charge, in order to notify about its charge via telemetry, to give an audible signal informing about a low battery charge, and execute a command to return the copter to the takeoff point when the battery is low, the battery should be calibrated. It is performed in the tab shown in Figure 9. For OMNIBUSF4V3, the values are set inked with green rectangles.

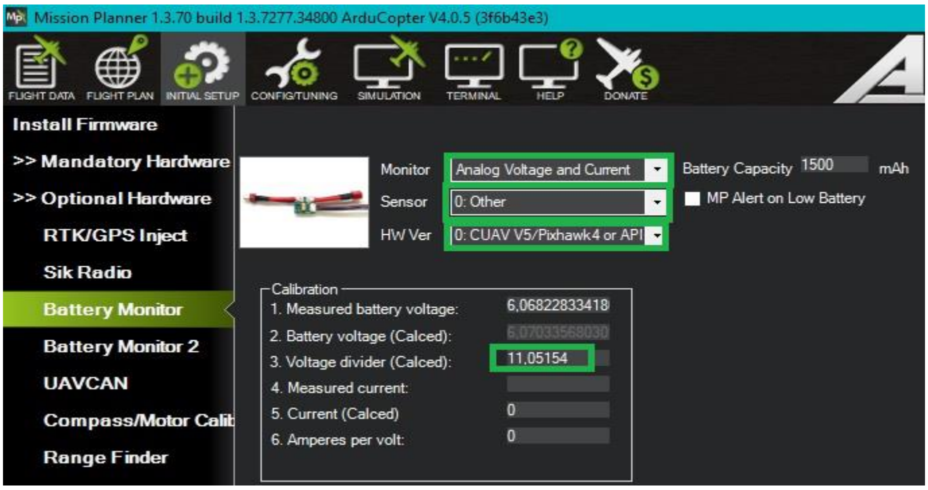

Figure 9: Battery calibration

The Extended Turning tab is used to configure the parameters of the PID controller and some navigation flight modes (inked in a red rectangular).

For copters, the PID controllers are adjusted by manual selection based on visual control over the stability of the aircraft behaviour. At present there are no reliable mathematical models for the automatic determination of the parameters of PID controllers. Therefore, for the convenience of adjusting the parameters during the flight, the sixth channel of the control equipment – the "spinner" – is used to set specific PID parameters and designate the range of these parameters. In Figure 10, the above said is inked in green rectangles

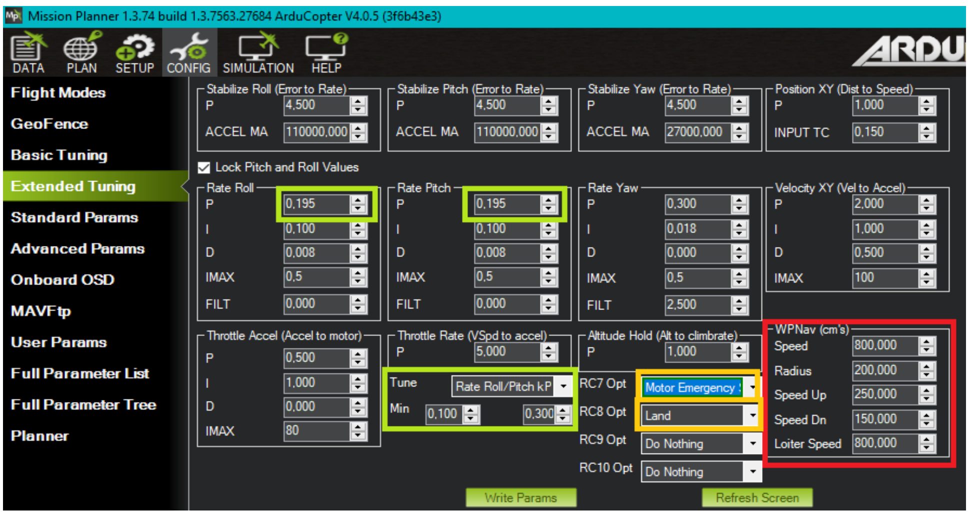

Figure 10: Setting up the PID regulators

The OMNIBUSF4V3 flight controllers are equipped with an OSD chip, with the help of which it is possible to overlay telemetry parameters (flight altitude, speed, distance travelled, number of received satellites, battery charge, etc.) on the video image of the FPV camera. The Onboard OSD tab is used to configure the OSD (Figure 11).

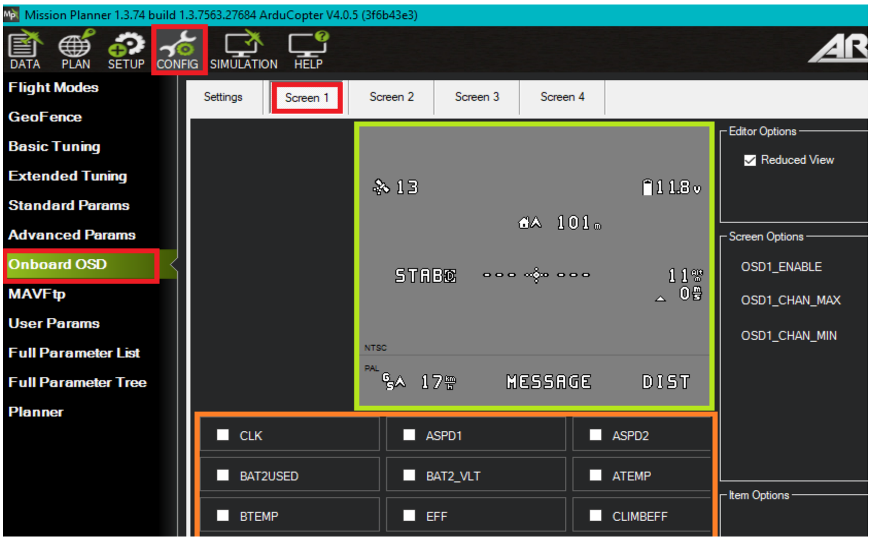

Figure 11: Setting up the OSD

The FPV camera screen field with the selected parameters is ensquared in green, and the settable parameters are selected at the bottom, ensquared in orange.

More fine adjustment of aircraft flight modes is performed in the Full Parameter List tab (Figure 12).

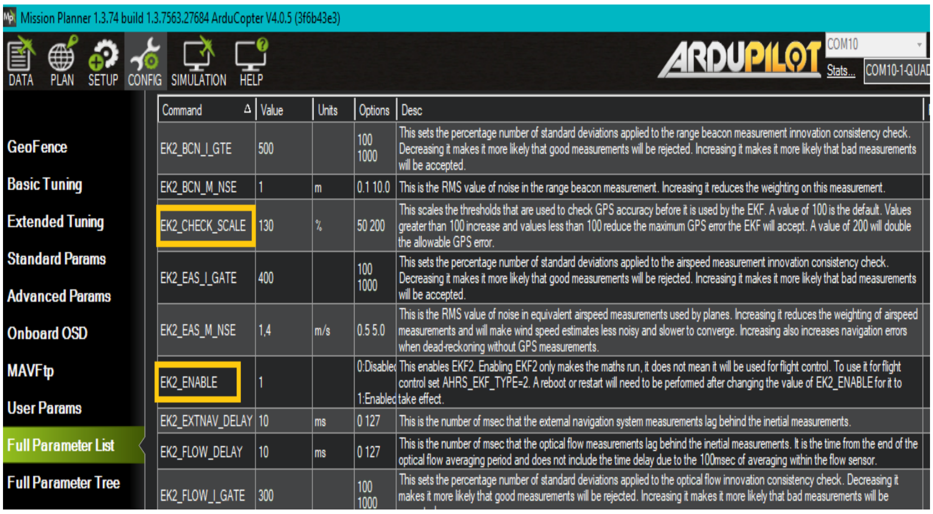

Figure 12: The Full Parameter List tab for fine adjustment of flight parameters

When configuring the firmware of the ArduPilot copter, one should pay attention to setting the following parameters:

AHRS_EKF_TYPE = 2 This parameter determines which version of the Kalman filter is used to estimate orientation and position;

EK2_ENABLE = 1 This enables EKF2. Turning on EKF2 only triggers mathematical calculations, but that does not mean that it will be used for flight control. To use it, you should set AHRS_EKF_TYPE = 2. After changing the value of EK2_ENABLE, you must perform a restart procedure for the parameters to take effect;

EK2_IMU_MASK = 1. One-byte IMU bitmap for use in EKF2. A separate item of EKF2 will be launched for each selected IMU. Value = 1 only uses the first IMU (default), if value = 2 then only the second IMU is used, value 3 allows the first and second IMU to be used. Up to 6 additional IMUs can be used if memory resources and processing speed permit. There may be insufficient memory and processing resources to run multiple items. If this happens, EKF2 will not start;

EK2_ALT_SOURCE = 0 defines which sensor is used as the main one for determining the altitude: 0: barometer is used (default); 1: Rangefinder is in use.

2: GPS is being used. Useful when GPS quality is very good and barometer drift can be a problem. For example, if the copter will perform long-distance missions with elevation differences> 100m;

EK2_GPS_TYPE = 0: controls the use of GPS.

0: use 3D speed and 2D position from GPS; 1: use 2D speed and 2D position (GPS speed does not affect the altitude estimate); 2: use 2D position; 3: no GPS (it will use the optical flow from the sensor only if available);

EK2_CHECK_SCALE scales thresholds that are used to check GPS accuracy before the EKF is used. The default is 100. Values greater than 100 increase, and values less than 100 decrease the maximum GPS error that the EKF accepts. A value of 200 will double the acceptable GPS error. For EK2_CHECK_SCALE of the tested aircraft the value of 130 is set. In this case, it establishes communication with satellites faster, but also flies stably;

AHRS_GPS_GAIN = 0 parameter controls how much intensive GPS should be used to correct the position. The parameter for the airplane should not be equal to 0, as this will lead to loss of control of the airplane when turning. For an airplane, the value is 1.0. For a copter is 0. The consequence of this parameter turned on is the twitching of the horizon line when the craft is stationary if the GPS does not perfectly capture the position and drifts. With strong jumps in GPS position, the roll can reach critical values, which will lead to instability of the copter. Therefore, in copters this parameter is set to 0.

GPS_AUTO_SWITCH = 0. Setting up automatic switching between multiple GPS. For one GPS, the parameter is set to 0.

The following is how braking is performed in LOITER flight mode (holding position and altitude):

LOIT_ACC_MAX = 500 Maximum acceleration of position correction in Loiter mode in cm/s/s. Higher values cause the aircraft to correct position errors more aggressively (for example, in gusts of wind);

LOIT_BRK_ACCEL = 100 Acceleration of braking when the copter jams on the brakes, in cm/s/s. Higher values stop the copter faster when the left-to-left and front-to-back stick of the controlling equipment is moved to the centre. Large values of this parameter lead to sharp braking, which can lead to a rollover of the copter, especially in gusty winds;

LOIT_BRK_JERK Braking jerk at Loiter in cm/sec/sec/sec. Higher values will eliminate braking more if the pilot moves the sticks during a braking manoeuvre. The default setting is used.

Ardupilot software allows performing automatic formation of the flight path, for example, for taking photographs, studying the radiation situation in the area, spraying fields, etc. Figure 13 shows an example of the formation of a flight mission for fertilizing a field. The processing contour is preselected (highlighted in red) and the flight path is automatically formed to cover the area that can be treated by the spraying system (the path is represented by yellow lines) [21]. Usually, the flight altitude above the ground is set equal to 1.5-3 m, which should be provided by the ultrasonic sensor. Its use is especially important if the field surface is not flat, i.e. there are depressions and hills.

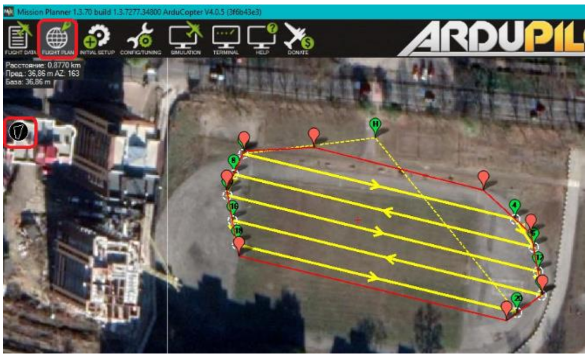

Figure 13: Formation of a flight mission in semi-automatic mode

In this work, an experimental test of the flight stability of the quadcopter presented in Fig. 2 with INAV firmware, ver. 2.6 and Arducopter firmware, ver. 4.0.5, was carried out. The test was carried out during winds of different intensity (table 1) in navigation modes. In fig. 14 (right) shows the flight path for two firmwares, which are programmatically set in the Mission Planner ground station for Arducopter and similarly in INAV - Configurator for INAV ver.2.6 firmware. Figure 14 on the left shows the real flight path for Arducopter ver.4.0.5, which was built using the Google Earth software package based on log files from the flight controller. A similar trajectory was obtained for INAV. As can be seen from Table1, the flight of the copter in navigation mode on the firmware INAV ver.2.6 was completely unsuccessful in a wind of 6-7m / s with gusts. Loss of stability of the quadcopter was recorded with its fall at the moment of passing an arbitrary waypoint, when the stability of the copter was minimal due to a sharp change in the direction of flight and a sharp gust of wind. Tests for high wind speeds were not carried out due to the obvious advantage of the Arducopter ver.4.0.5 firmware.

Table 1
Experimental parameters and results

№	Wind speed, m/s	Number of trajectory flights	The number of crashes of copter for firmware Ardupilot	INAV
1	1-2	5	0	0
2	3-4	5	0	0
3	4-5	5	0	1
4	6-7	5	0	5

Thus, the expediency of using the mathematical model of the extended Kalman filter was experimentally established to ensure the stability of the flight of a rotor-type UAV.

Figure 14: Copter flight trajectories according to the program (right) and obtained experimentally (left)

5. Conclusions

1. Experience has shown that the operation of Arducopter firmware for OMNIBUSF4V3 flight controller, which was developed for the Pixhawk family flight controllers is correct.

2. A test of the stability of the flight of the quadcopter on a frame of 250 mm during a gusty wind of 6-7 m/s in navigation modes Arducopter firmware was carried out. The stability of the flight in the mode of automatic flight by points and in the mode of automatic return to the starting point was noted.

3. Likewise for INAV firmware, ver. 2.6, flight stability of the same quadcopter was tested during a gusty wind of 6-7 m/s in navigation modes. Significant flight instability was found in the auto return and point-to-point flight modes. Loss of stability of the quadcopter with its fall was recorded.

4. The expediency of using the mathematical model of the extended Kalman filter to ensure the stability of the flight of a rotor-type UAV, which is used in the arducopter 4.0.5 firmware, has been experimentally established. It provides better flight results in navigation modes than the complimentary filter based firmware (INAV 2.6 firmware). When flying in acro (gyroscope) and stabilize (gyroscope + accelerometer) modes, the behavior of the copter for these two firmwares did not differ noticeably.

5. It is deemed that the mathematical model of the extended Kalman filter is expedient to use on high-performance microcontrollers such as STM32F4 and STM32F7 for flight in navigation modes, where many sensors are simultaneously used to provide information about the position of the UAV.

6. The possibility of correct formation of the flight trajectory in the semi-automatic mode for taking photographs, studying the radiation situation of the area, spraying fields, etc. has been experimentally established for Arducopter firmware. Formation of a flight mission for INAV is more primitive, it is performed only in manual mode, using preliminary calculations.

7. It has been established that INAV firmware with the UAV only allows transmitting telemetry data to the ground station. From the ground station, changing the UAV flight trajectory through telemetry is impossible, for example, when "dragging" the flight point with the mouse manipulator [22-24]. Arducopter firmware supports the two-way MAVLink protocol for receiving and transmitting telemetry data between the flight controller and the ground station.

8. For INAV firmware, it was experimentally impossible to connect the flight over a given waypoint with the execution of an action, for example, turning on and off the sprayer nozzle, dropping the load, turning on/off the video camera. Arducopter firmware allows doing these.

6. References

[1] A. Boyko, AV applications, 2017. URL:http://robotrends.ru/robopedia/oblasti-primeneniya-bespilotnikov.

[2] The modernized Spectator drone from OJSC "Meridian" named after SP Korolyova, 2019.

URL:https://www.youtube.com/watch?time_continue=6&v=HvLErmgBRX4&feature=emb_logo.
[3] Spectator (UAV), 2020. URL:https://ru.wikipedia.org/wiki/Spectator_(БПЛА).
[4] A. Boyko, Spraying plants from drones, 2019. URL:http://robotrends.ru/robopedia/opryskivanie-rasteniy-s-bespilotnikov.
[5] iNavFlight Missions, 2020. URL: https://github.com/iNavFlight/inav/wiki/iNavFlight-Missions.
[6] Copter Mission Command List, 2020. URL: https://ardupilot.org/copter/docs/mission-command-list.html.
[7] FPV piloting theory, 2021, URL:https://github.com/mlutskiy/pioneer-doc/blob/master/database/pilot-module/pilot-3part.rst.
[8] ArduPilot Firmware builds, 2020, URL:https://firmware.ardupilot.org.
[9] INAV 2.6.0, 2020, URL:https://github.com/iNavFlight/inav/releases/tag/2.6.0.
[10] Pixhawk Overview, 2020, URL:https://ardupilot.org/copter/docs/common-pixhawk-overview.html.
[11] Omnibus F4 Pro (on-board current sensor) and Omnibus F4 AIO (no sensor onboard), 2020, URL: https://ardupilot.org/copter/docs/common-omnibusf4pro.html?highlight=inav.
[12] Starlino, DCM Tutorial-An Introduction to Orientation Kinematics, 2011, URL: http://www.starlino.com/dcm_tutorial.html.
[13] Extended Kalman Filter (EKF), 2020, URL: https://ardupilot.org/copter/docs/common-apm-navigation-extended-kalman-filter-overview.html.
[14] Filtr Madzhvika, 2015, URL: https://habr.com/ru/post/255661.
[15] Ibrahim K. Mohammed, Abdulla I. Abdulla, Elevation, pitch and travel axis stabilization of 3DOF helicopter with hybrid control system by GA-LQR based PID controller, International Journal of Electrical and Computer Engineering, vol. 10, no. 2, pp. 2020, 1868-1884. doi: 10.11591/ijece.v10i2.pp1868-1884.
[16] Agung Prayitno, Veronica Indrawati, Ivan Immanuel Trusulaw, Fuzzy Gain Scheduling PID Control for Position ofthe AR.Drone, International Journal of Electrical and Computer Engineering, vol. 8, no. 4, 2018, pp. 1939-1946. doi: 10.11591/ijece.v8i4.pp1939-1946.
[17] Almido H Ginting, Oyas Wahyunggoro, Adha Imam Cahyadi, "Attitude Control of Quadrotor Using PD Plus Feedforward Controller on SO(3)," International Journal of Electrical and Computer Engineering, vol. 8, no. 1, 2018, pp. 566-575. doi: 10.11591/ijece.v8i1.
[18] S.Lienkov, A.Myasischev, O.Banzak, L.Komarova, N.Lytvynenko, O.Miroshnichenko, Construction of an Aircraft-Type UAV for Flight Along a Given Trajectory in the Automatic Mode, International Journal of Emerging Trends in Engineering Research, vol. 8, no.9, 2020, pp. 6145-6150. doi:10.30534/ijeter/2020/200892020.
[19] S. Lienkov, A. Myasischev, L. Komarova, N. Lytvynenko, V. Shvab, O. Lytvynenko, Creation of a Rotor-Type UAV with Flight Controllers, Based On a ATmega 2560 and STM32f405 Microprocessors, International Journal of Emerging Trends in Engineering Research, vol. 8, no. 8, 2020, pp. 4703-4710. doi:10.30534/ijeter/2020/104882020.
[20] S. Lienkov, A. Myasischev, O. Banzak, Y. Husak, I. Starynski, Use of rescue mode for UAV on the basis of STM32 microcontrollers, International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 3, 2020, pp. 3506-3513. doi:10.30534/ijatcse/2020/156932020.
[21] Mission Planning, 2021. URL: https://ardupilot.org/copter/docs/common-mission-planning.html.
[22] Flight by plane using INAV telemetry, 2021. URL:https://sites.google.com/site/webstm32/inav_25_wp_sam/telemetriy-inav.
[23] Chi-Bao Le, Dinh-Thuan Do, Employing non-orthogonal multiple access scheme in UAV-based wireless networks, Bulletin of Electrical Engineering and Informatics, vol. 10, no. 1, 2021, pp. 241-248. doi: 10.11591/eei.v10i1.2102.
[24] Deha Agus Umarhadi, Projo Danoedoro, Comparing canopy density measurement from UAV and hemispherical photography: an evaluation for medium resolution of remote sensing-based mapping," International Journal of Electrical and Computer Engineering, vol. 11, no.1, 2021, pp. 356-364. doi: 10.11591/ijece.v11i1.pp356-364.