MULTIDRONE –
MULTIple DRONE platform for media production

Project start date: 01.01.2017
Duration: 36 months
Lead contractor: Aristotelio Panepistimio Thessalonikis (AUTH)

Deliverable D5.1: Drone platform implementation report

Date of delivery: 30 June 2018

Contributing Partners: Alerion, Aristotle University of Thessaloniki, Universidad de Sevilla, Instituto Superior Técnico

Version: v4.0

Title:	**D5.1: Drone platform implementation report**	
Project:	**MULTIDRONE** (ICT-26-2016b RIA)	
Nature:	Report	**Dissemination Level:** PU (PUblic)
Authors:	**Nikolaos Nikolaidis (AUTH), Paraskevi Nousi (AUTH), Eustratios Kakaletsis (AUTH), Pantelis Kaplanoglou (AUTH), Arturo Torres (USE), Jesús Capitán (USE), Cédric Le Barz (TS), Tiago Goncalves (TS), Michael FAGNO (TS), Grégoire Guerout (AlR), David Escudeiro (IST), Bruno Gomes (IST), José Tojeira (IST)**	
Lead Beneficiary:	Alerion (AlR)	
WP	5	
Doc ID:	MULTIDRONE_D5.1.pdf	

Document History

Version	Date	Reason of change
1.0	19/06/2018	First complete draft
2.0	26/06/2018	Final draft for internal review
3.0	27/06/2018	Revised version including internal reviewer's comments
4.0	29/06/2018	Final version to be submitted to the EU, including modifications agreed in the 3rd Consortium meeting.

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Table of contents

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Executive Summary

This document reports the work performed by the MULTIDRONE partners on the implementation of the drone platform with inputs from Task T5.1. It gives an overview of the hardware, the platform and the payloads decided by partners for the drone prototype, keeping the detailed description for the deliverable D2.3. This overview aims to help to apprehend better the rest of the document which focuses on the integration and the testing of the drone. The HW system assembly/integration is presented, followed by the description of the HW tests. The document also overviews the software modules that will run on-drone, and the achievements in terms of integration. The deliverable finishes by the initial purchase list of the drone prototype parts as Appendix A and the software development tools and issues as Appendix B.

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

1. Introduction

The deliverable D5.1 is dedicated to the drone platform implementation. The document contains a description of the implemented drone platform and overviews of the software modules that will run on-drone. It will more specifically present the system integration and testing of the drone platform hardware and the embedded software achieved. In this period, the overall Multidrone workflow was as follows: research was performed in WP3 and WP4 and then development followed by integration was performed in WP5. In this reporting period, the work in WP5 in general and in T5.1 in particular was focused on a) HW assembly and integration and b) SW development with some integration, mostly between the SW modules and also, to a lesser extent, between HW and SW. In the future, the balance in (b) between SW development and HW/SW integration is expected to reverse. Furthermore, an overlap between research and SW development and SW integration is expected that will blur their boundaries, as new research results will enter into new SW versions.

In T5.1, a significant effort was devoted by AlR to drone HW system integration and the testing as it plays a major role in the project progress. This deliverable focuses on the drone itself, without the ground station infrastructure that is in development for the MULTIDRONE project. The drone configuration for this deliverable is, as described in the [D2.3], section 4.4.2, that is without LTE. The ground station implementation report will be described in the [D5.2], and the overall and final description of the full system implementation will be described in the [D5.5].

The implementation of the drone platform comprises the assembly of the drone, its integration and its testing on the hardware side. On the embedded software side, it comprises the development of the software modules, its integration and testing on the on-board computers. The document will present an overview of the hardware design of the drone and the software development of the modules. A more detailed description can be found on the [D2.3] document.

Each component was tested first alone, as far as possible, in order to verify whether the component was properly working or defective, and to be sure that the features meet the expected specifications.

Then experiments were conducted by integrating several components into subsystems to check compatibility and communication between these modules.
On the software side, the compatibility of the modules on the on-board computers as well as the load of the module on the processor was checked as it is a vital part of the safety of the subsystem and to the whole system.

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

2. Drone hardware description

2.1 Overview

The drone platform implemented follows the specifications of the drone described in the deliverables [D2.2]/[D2.3] which are themselves based on the requirements from the media partners. These requirements are detailed in deliverable [D2.1]. More generally, the design of the drone was established by collecting specifications and wishes from all the partners and sorted by their significance and taking into consideration national and European regulations. Several proposals have been formulated with different compromises.

This part will describe the hardware of the drone platform to have an overview of it and a better understanding for the system integration and testing. A more detailed description can be found in the deliverable [D2.3].

The functional requirements of the partners were synthesised and translated into technical specifications. Components were found to comply with these specifications, which resulted in having estimated weight and size of the drone as well as its flight time.
After looking to the current technologies, their weight and their prices, it was found that the requirements are diverging: having all the components required in the range of the quality desired is not compatible with the wished price and the size/weight of the drone. The weight and the size of the drone are linked. Generally speaking, the bigger the drone is, the more it can carry and is heavier.

The guidelines of the proposals are the choice of the lightest components, the total cost, a flight duration of at least 15 min in an ideal scenario. Different proposals were made and are mainly about different configurations to have different weights and prices. The wish to have high-end components was restrained by the costs of these devices, as the overall drone budget could be much higher than the estimated cost from the description of actions if all these high-end components were chosen. The choice of the component was also restrained by the wish to have a not too big drone, and not too heavy drone. Several discussions amongst the partners lead to a choice of a final compromise between the possible components.

The first proposal is a drone with the maximum components possible with the limit of the 25 000 € budget. This one has all the features presented in the previous paragraph. It is based on a DJI M600 and weight 15 kg.
The second proposal is considering the current regulations in Germany and Italy, where having parachute is not mandatory. By removing this system, a lighter frame can be chosen. As a result, a drone based on a DJI S1000+ that weight 11 kg with a current estimation of around 23 000 € is the second proposal which goes down to 16 000 € by using another LIDAR.
A last proposal only based on the weight is being considered, in order to have a multirotor copter that weight a maximum of around 5 kg. With this constraint, none of the components chosen for their desired quality can be chosen. As an example, instead of having an

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

audiovisual camera with broadcasting quality, an action camera, such as a GoPro Session is the best option. This drone would weight around 5.5 kg and should cost around 8000 €.

The compromise decided by the consortium corresponds to the second proposal modified. The drone is based on a DJI S1000+ platform. It has an audiovisual camera and a relatively inexpensive LIDAR (compared to the one budgeted in the DoA). If the situation needs a parachute, then smaller batteries will be used, in order to add a parachute. Due to uncertainty of the stage of the DJI S1000+ in its product life cycle, and the real flight time of the drone that needs to be confirmed, it was decided to have the possibility to change the frame to another one, like from Gryphon Dynamics, after tests about fight time.

2.2 Presentation of parts

The drone platform includes the frame, the arms, the landing gear, the propulsion systems (motors + propellers) and the Electronic Speed Controllers (ESC). The DJI S1000+ is the chosen platform for the first version of the prototype as a compromise between the weight (payload) it can carry, the price and the size, despite its unknown stage in its product life cycle, and limited payload it can carry with respect to the desired equipment.

The drone core is composed of the Flight Control Unit (FCU), which includes the Inertial Measurement Unit (IMU) and is linked to an external RTK GNSS module. The Pixhawk 2.1 will be used as the FCU due to, amongst other, its triple redundant IMU. The flight logs may be recorded on the SD card of the Pixhawk and the batteries power will be monitored by current sensors linked to the Pixhawk. An RTK GNSS will be used to obtain a centimetre position accuracy using a sensitive antenna, while the main communication system is the Thales Module, which uses LTE technology to communicate with the Ground Station and has a Wifi mesh system for inter-drone communication.

The flight payload is composed of systems that are not vital to have a drone flying but are needed to implement autonomy, safety, and security features on the drone. This definition corresponds to the use of a LIDAR. It will be used for mapping, localisation, and obstacle avoidance. The navigational (FPV) camera will be used by the Supervisor to see where the drone is going. These sensors will be connected to one of the two on-board computers, a Nvidia TX2 with its carrier board and an Intel NUC, both of which have a more powerful CPU and GPU than the flight control unit. The previous algorithms as well as the tracking algorithm and some emergency systems will run on these on-board computers. Depending on the needs during the tests and experimentations, a parachute safety system may be added to the drone.

The audiovisual payload contains all the components needed to acquire images and video for media production. It includes the audiovisual camera, which should provide picture quality and resolution comparable to those of other video sources used in media production. Discussions with DW and RAI have led to the selection of the Blackmagic Micro Cinema Camera with a motorised Panasonic x3 lens. This camera can record in a RAW format on a SD card while outputting the video flow with an HDMI connector. The video flow

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

is then transferred to the Nvidia TX2 using an adaptor. The stabilisation and orientation of the camera are managed by a 3-axis gimbal and controlled by a BaseCam (AlexMos) controller. The commands to control the gimbal and the camera may come from the ground station (through LTE) or a transmitter (through RF radio link).

Like for aircraft, the energy source is not included in the drone core, nor the payload. As it is an essential part of the system, it helps calculate the maximum payload the drone can have. The current drone has two main batteries (10 000 mAh, 6S) to power the drone and the on-board payload while flying, and a second power source is being considered as back-up batteries for the FCU. In order to have a 1-hour mission, at least 6 other main batteries will be bought and will replace the drained ones during a scenario.

Other components have been considered: a case for storage and transport; spare parts as it is usual to have faulty delivered parts, and failures/faults during the prototyping; chargers and LiPo safe bags for recharging the batteries; miscellaneous tools and components like wires, heat shrink, stain, glue, etc.; specific components for integration like a parachute training actuator. Finally, it is usual in hardware design to have to adapt by adjusting some components during the prototyping, potentially adding the purchase of other components.
The price of this drone platform is around 15 000 €, without the spare parts. Its weight is around 11 kg. The rationale of the consortium for choosing a less expensive platform than the budgeted one (25000 €) is to have room for insurance and logistics costs (for running the experiments in WP5 and WP6) and other costs (e.g., ground equipment) that were not originally foreseen.

Main parts

Category	**Part**	**Brand & Model**
Frame	Frame	DJI S1000+
Drone core	FCU	Pixhawk 2.1, power module, SD card
	GPS RTK	Here+
	On-board Communication Module	Thales LTE/Wi-Fi Communication Module (motherboard + modems + antennas)
	Back-up radio	Futaba T14SG

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Drone payload	On board computer	Nvidia TX2 + Intel NUC + hat/shield + heat sink + power module
	LIDAR	Leddartech M16 + interface
	Laser Altimeter	LIDAR SF11/c
	Parachute	If necessary - System based on the Galaxy GRS 10/350
Audiovisual Payload	AV camera	BMMCC +SD card + Lens Panasonic Lumix G X Vario PZ 14-42mm
	Gimbal	iFlight G40 + Basecam controller
	Back-up radio	Futaba T14SG

Main specifications

	Selected drone proposal
Frame	DJI S1000+ 

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Camera +gimbal	BMMC + iFlight G40 
Parachute	Optional (~1 kg - with smaller batteries)
LIDAR	Leddartech M16 (0,2 kg; 45° HFOV, 7° VFOV, 50m) 
Weight batteries for 15 mins (kg - Ideal case)	2,4
Usable payload (kg) (Other than batteries)	3,8
Size (prop to prop - m)	1,45
Weight (kg - Estimation)	11
Price (€ - Gross) with accessories	16 000
Advantages	AV quality camera with x3 zoom

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Drawbacks	Heavy Limited sense & avoid, localisation Risk of spare parts shortage (approaching End of Life)

Drone configurations

Due to the maximum takeoff weight of the drone, the drone will have 2 configurations: one with an audiovisual camera for the shooting mode and the other one with a high-end LIDAR for the mapping mode.
The global architecture of the drone hardware does not change between these two configurations.

3. Hardware integration report

3.1 Overview

The hardware implementation of the drone platform includes the assembly of the prototype and testing this prototype. The integration and the testing are carried out at the same time and are part of a same time planning. The components of the drone are tested first to check whether they are working and not defective. This is done before mounting the component on the drone (if the component does not need to be on the drone to work).

The process for hardware implementation started by creating the drone part purchase list from the specifications of all partners. This list can be found in Appendix A. To purchase the parts for the drone prototype, it was decided to request three offers from three different European companies and choose the offer following the best value for money rules. This procedure was to prepare and to simplify the purchase of the 6 following drones by the MULTIDRONE partners. The components were then received and tested to be sure than the items received are working and not defective. If it is possible, first functional tests can be performed at this stage. Depending on the situation, subsystems can be assembled and tested. These subsystems tests are performed step by step, in order to have a higher chance of finding the reason of an issue if one occurs. The tests and the assembly are realised in the same period of time. During this phase, components may be changed, if it appears that they do not fit the project, amongst others, either because of its real performances or its assembly in a sub-system is compromised by specifications that were wrong on the data sheet or it appears that its position is less usable than though in the design phase. The assembly and integration phases are phases with significant work of building, trying, testing following the

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

initial plan, and regularly changes. These changes often require buying new pieces, and so extra delays.
At the end of the assembly and integration, final tests are performed to check if all planned features are working.

The aim of the integration process is to verify and validate the conception, and the theoretical specifications of the drone platform.

3.2 System Integration

Once the specifications for the drone prototype and the design were finalised, a purchase list was established. Due to the complexity of the system, the list includes more than 100 items, despite the choice of COTS systems for some components. The list of components proposed for the call for tenders for the prototype drone can be found in Appendix A.

Most of the companies contacted across Europe declined to send a quote, due to the number of components, including some non-common ones and, especially, because of the many suppliers or manufacturers that would be involved for having these components. These delays the drone platform assembly, and especially its integration phase.
Alerion (AIR) was the first partner to buy the drone parts to build the first drone prototype, to be used as a guide for building 6 more drones belonging to the partners.
The reception of the pieces took more time than expected, as 3 months after the order to the company, some pieces are still missing, mostly due to restocking from the manufacturer. This resulted in delays that change the system integration and testing phase, as tests and assembly cannot be completed until these pieces are received.
During the reception and the assembly, some pieces were found to be defective, or the alternative pieces received to the quote was not meeting all the requirements. New pieces needed to be purchased, in order to fix these issues.
All the pieces received were weighed and compared to the estimated weight from data sheets or other estimations. Before mounting the drone and integrating all the parts, the weight was 400 g higher than the expected one. After analysis, it appears that the main reason is that the weight displayed on data sheets is the weight of the component only, excluding its cables and, sometimes, its supports.
This extra weight, if not properly managed, will reduce the flight time and reduce the manoeuvrability of the drone. Due to this reason, drone frame (DJI S1000+) is under investigation by AIR, regarding its suitability, particularly with respect to the maximal flight time. If it is found unsuitable, the consortium will switch to a more powerful (but also more expensive) frame, e.g., from Gryphon Dynamics. This will entail a delay of about 1 month in the drone purchase and build-up plan.
The AlR drone will be the basis for purchasing and building up all other MULTIDRONE drones by AlR. Drone payload may differ from partner to partner (e.g., USE and AUTH will certainly have more expensive LIDARs to meet the needs for WP4).

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Fig. 3.1: Drone platform with the drone core components (without flight nor audiovisual payload), ready for basic flight tests.

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Fig. 3.2: Remote Control with Screen.

Fig 3.3: Camera and Gimbal on a temporary support.

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

3.3 HW testing

3.3.1 Introduction

This section describes the tests that will be run on each component, subsystem and on the full system, in order to identify whether each part is working properly or not following a test scenario.

At this phase, each subsystem is tested to check if it is working properly, in a normal situation up to the scenario situation. Furthermore, the real performances of the system and subsystems are tested. If a subsystem does not match the expected results, a decision needs to be made between keeping the chosen parts and adjust the expectations to match the chosen components, or to change the component as one specification of the drone which rely on it, is a priority. The expectations/specifications from the media partners (DW and RAI) are in document D2.1.

The drone configuration presented in this document is the drone “without LTE” as mentioned and describe in the D2.3 section 4.4.2.
The systems should be tested in an organised manner, and ideally, prioritising the most critical parts first and the tests than require the longest time. However, due to the delay in the reception of drone parts, this order could not be followed. The parts were tested following the components delivery schedule and the delay due to complications on previous tests.
The most important items to check are the ones for the drone to be able to fly. It includes the components of the drone core, that are, the frame, the Flight Control Unit, and the RC controller.

In order to test the full system, tests of components and subsystems have to be done first. If an issue occurs while running tests on the full system, the cause of this issue can be very hard to find. That’s why subsystems are tested first to eliminate most of the issues than can occur with the full system.

Different tests are performed at the several levels of sets/subsets. The testing phase starts by the components, then the subsystems and to finish by the full system in the configuration without LTE.

- Components: Test of the components received. In order to work, some components need to be used with other(s) component(s), creating a small subsystem, but only the performance of the tested component is taken into account.
- Subsystems: A subsystem is a group of components. Components are added one by one or by group following the need until reaching the subsystem. The components of the subsystem are tested at the same time. A subsystem can be the camera and the gimbal, then we add the RC transmitter then we add the video streaming system.
- Full system: It is the final system, it includes all the parts and they are all working together.

This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No *731667*.

The HW testing phase can be divided into two sections :

- the tests to check whether the received item is defective or not, if the component, the subsystem or the system is working.
- the tests to check the functionalities of the component, the subsystem or the system.

3.3.2 Tests Description

Each test has a description and the expected results, if not obvious, and the test results. If the results are not reached, a gravity estimate of the issue is added as well as proposed ways to fix the issue, which will be tested in a following test.

ID: 1
Component(s): Pixhawk
Test Presentation: Internal power management on the Pixhawk - The Pixhawk is turned on.
Expected results: The Pixhawk turns on, the power is stable.
Test passed/Results: Ok

ID: 2
Component(s): Pixhawk+QGroundControl
Test Presentation: Telemetry and commands by cable-Check if data is sent correctly between Pixhawk and the computer with QGroundControl
Expected results: Synchronisation without any issue - indicated on QGroundControl
Test passed/Results: OK

ID: 3
Component(s): Pixhawk + other components such as the GNSS module + QGroundControl
Test Presentation: Test of the Pixhawk ports-All the ports that will be used during the project will be tested to check if a signal is sent. The ports will be tested by putting the respective component such as the GNSS module and check in a console that the message is correct.
Expected results: When connected to a device, the respective ports are sending a correct message with a correct structure.
Test passed/Results:

- 1st test) Not OK: GNSS module powered by the Pixhawk but not receiving signal
- 2nd test) OK with a new Pixhawk

Gravity: High
Reason if problem & modification: 1st test) Defective Pixhawk - To test on a new one

ID: 4
Component(s): Pixhawk + QGroundControl
Test Presentation: Test of the internal sensors (IMU)-The internal sensors values are read on the Nutshell console to check if all sensors are working and within the same range of values.
Expected results: All the sensors are sending a value, which vary in time (due to the noise) and when the Pixhawk is moved. All the values from redundant sensors are within the same

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

range
Test passed/Results: OK (need calibration)

ID: **5**
Component(s): Pixhawk + RC transmitter + QGroundControl (+ oscilloscope)
Test Presentation: Test of the PWM outputs-The Pixhawk is only linked to the computer and a RC receiver. The Pixhawk is armed and a throttle command is sent using the RC transmitter. All the PWM output commands are checked using a console, and if necessary, they are directly checked on the physical pins using an oscilloscope
Expected results: All the PWM commands for the 8 arms should vary following the RC commands. All the physical connectors should send the same PWM signals as the commands
Test passed/Results: OK

ID: **6**
Component(s): Pixhawk + RC transmitter + QGroundControl (+ oscilloscope)
Test Presentation: Test of the basic control-command-The aim of this test is to check the autopilot, its response to different commands and different sensors input, mostly IMU.
Expected results: While the Pixhawk is flat, the output should be stable, if the pixhawk is moved, the PWM outputs should move accordingly, if a command or roll, yaw or pitch, the PWM signals should change accordingly
Test passed/Results:

- 1st test) Not OK : Regular Offset with the accelerometers
- 2nd test) OK

Gravity: High
Reason if problem & modification: 1st test) Calibration before each flight

ID: **7**
Component(s): TX2
Test Presentation: Powering the Onboard computer
Test passed/Results: OK

ID: **8 - Test to be done**
Component(s): TX2+Auvidea Carrier Board
Test Presentation: Powering the system
Test passed/Results: (Carrier board not received yet)

ID: **9 - Test to be done**
Component(s): TX2+Auvidea Carrier Board
Test Presentation: Test of all necessary ports for the project

ID: **10**
Component(s): NUC
Test Presentation: Powering the Onboard computer
Test passed/Results: OK

ID: **11**
Component(s): NUC

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Test Presentation: Test of all necessary ports for the project
Test passed/Results: Not all tested yet. OK up to now

ID: **12 - Test to be done**
Component(s): NUC
Test Presentation: Checking if the computer is working after removing all the not necessary parts (wifi, etc.)-For the weight optimisation process
Test passed/Results: OK up to now

ID: **13**
Component(s): Wireless Telemetry+Pixhawk+QGroundControl
Test Presentation: Powering the system and checking on QGroundControl if the link is established
Expected results: QGroundControl should show that the link is established with the Pixhawk, and the current status of the Pixhawk should be displayed
Test passed/Results: OK

ID: **14**
Component(s): Camera
Test Presentation: Powering the camera
Test passed/Results: OK

ID: **15**
Component(s): Camera
Test Presentation: Checking if all ports used during the project are working
Test passed/Results: OK

ID: **16**
Component(s): Camera
Test Presentation: Checking if all the commands using the buttons on the camera are working-(commands that will be remote during the project)
Test passed/Results: OK

ID: **17 - Test to be done**
Component(s): Camera+RC
Test Presentation: Checking if the camera can be commanded via the extension port

ID: **18**
Component(s): Camera+Lens
Test Presentation: Checking if the physical buttons on the lens are working (zoom and focus)
Test passed/Results: OK

ID: **19**
Component(s): Camera+Lens
Test Presentation: Recording 1h in Full HD in CinemaDNG Raw format
Test passed/Results: OK

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

ID: **20**
Component(s): Camera+Lens
Test Presentation: Check the clarity of the image (check if defective lenses)
Test passed/Results: OK

ID: **21**
Component(s): Gimbal
Test Presentation: Powering the gimbal
Test passed/Results: Waiting for power cable

ID: **22 - Test to be done**
Component(s): Gimbal + Camera
Test Presentation: Balancing the system and checking if the movements are smooth

ID: **23**
Component(s): Landing gear
Test Presentation: Check if working
Test passed/Results: OK

ID: **24 - Test to be done**
Component(s): Parachute system
Test Presentation: Powering the system

ID: **25 - Test to be done**
Component(s): Parachute system + RC
Test Presentation: Test the system with the training pyro-actuator

ID: **26**
Component(s): RC transmitter
Test Presentation: Powering the RC transmitter
Test passed/Results: OK

ID: **27**
Component(s): RC transmitter
Test Presentation: Link with receiver
Test passed/Results: OK

ID: **28**
Component(s): RC+Pixhawk+QGroundControl
Test Presentation: Check if the transmitter emits in all its channels
Test passed/Results: OK

ID: **29**
Component(s): Screen
Test Presentation: Powering the screen
Test passed/Results: OK

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

ID: **30**
Component(s): Amimon Connex + Screen
Test Presentation: Powering the system
Test passed/Results: OK

ID: **31**
Component(s): Amimon Connex + Screen
Test Presentation: Check if communication link is established
Test passed/Results: OK

ID: **32**
Component(s): Pixhawk + GNSS module
Test Presentation: Checking if receiving satellite signals
Test passed/Results: OK

ID: **33**
Component(s): Pixhawk + GNSS module + RTK + QGroundControl
Test Presentation: Checking if the base is receiving satellites and sending corrections to the Pixhawk
Test passed/Results: OK according to QGroundControl indications

ID: **34 - Test to be done**
Component(s): LIDAR
Test Presentation: Powering the component

ID: **35 - Test to be done**
Component(s): LIDAR+NUC
Test Presentation: Checking if it sends signals to the onboard computer

ID: **36**
Component(s): Onboard power system+component
Test Presentation: Check if the on-board power module is compatible with the respective component
Test passed/Results: In progress - The one delivered for NUC not sufficient

ID: **37**
Component(s): Motors+Pixhawk+RC
Test Presentation: Test of the motors-The Pixhawk is connected to the ESC which are connected to the motors. First a calibration of the ESCs has to be performed. The blades of the motors are removed. Commands are sent through RC. The type of commands tested in the previous test is reproduced to check the motors and the ESCs
Expected results: When the throttle increase, and the system is flat, all the motor should start spinning at the same or close to the same throttle input. The other RC inputs should produce coherent motor spinning.
Test passed/Results: Problem linked to Test 6 - Ok with a calibration before each "flight"

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

ID: **38 - Test to be done**
Component(s): Camera+Gimbal+RC
Test Presentation: Test of the controls of the Gimbal and the camera using the RC

ID: **39 - Test to be done**
Component(s): Camera+Gimbal+RC+Connex+Screen
Test Presentation: Test of the full AV system in "without LTE" mode

ID: **40 - Test to be done**
Component(s): Pixhawk+Altimeter
Test Presentation: Test if the altimeter is working

ID: **41 - Test to be done**
Component(s): Pixhawk+NUC
Test Presentation: Communication test between the Pixhawk and the NUC

ID: **42**
Component(s): Nuc+TX2
Test Presentation: Communication test between the NUC and the TX2

ID: **43 - Test to be done**
Component(s): Frame+Pixhawk+RC
Test Presentation: Powering the drone

ID: **44**
Component(s): Frame+Pixhawk+RC
Test Presentation: Arm/Disarm the drone
Test passed/Results: OK

ID: **45**
Component(s): Frame+Pixhawk+RC
Test Presentation: The drone takes off
Test passed/Results: OK

ID: **46**
Component(s): Frame+Pixhawk+RC
Test Presentation: The drone hovers
Test passed/Results: Not OK: uncontrollable yaw
Gravity: High
Reason if problem & modification: Checking the potential reasons of that problem : yaw command (autopilot error), defective sensor, Electromagnetic perturbation, incompatibilities

ID: **47 - Test to be done**
Component(s): Frame+Pixhawk+RC
Test Presentation: The drone does basic movements

ID: **48 - Test to be done**

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Component(s): Frame+Pixhawk+RC
Test Presentation: The drone makes scenario like movements

ID: **49 - Test to be done**
Component(s): Frame+Pixhawk+RC
Test Presentation: Test the RC system at a distance

ID: **50 - Test to be done**
Component(s): Frame+Pixhawk+RC
Test Presentation: Flight time

ID: **51 - Test to be done**
Component(s): Camera+Gimbal+RC+Connex+Screen
Test Presentation: Test the RC system at a distance

ID: **52 - Test to be done**
Component(s): Full system
Test Presentation: Confrontation of the HW design with the reality

ID: **53**
Component(s): Full system
Test Presentation: Estimation weight with the real weight
Test passed/Results: In progress, up to now higher than expected weight

ID: **54 - Test to be done**
Component(s): Full system
Test Presentation: Each part is well powered on the drone

ID: **55 - Test to be done**
Component(s): Full system subsystem by subsystem
Test Presentation: Each part is well placed on the drone following the initial design

ID: **56 - Test to be done**
Component(s): Full system subsystem by subsystem
Test Presentation: Each part is well placed following the updated design

ID: **57 - Test to be done**
Component(s): Full system
Test Presentation: Powering up

ID: **58 - Test to be done**
Component(s): Full system
Test Presentation: Check the calibration of sensors

ID: **59 - Test to be done**
Component(s): Full system
Test Presentation: Check the status of the Pixhawk

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

ID: **60 - Test to be done**
Component(s): Full system
Test Presentation: Check the communications with the RC

ID: **61 - Test to be done**
Component(s): Full system
Test Presentation: Check the communication with the GCS

ID: **62 - Test to be done**
Component(s): Full system
Test Presentation: Hovering

ID: **63 - Test to be done**
Component(s): Full system
Test Presentation: Basic movements

ID: **64 - Test to be done**
Component(s): Full system
Test Presentation: Scenario movements

Summary

Most of the tests are passed up to now. The main tests that did not pass concern or are linked to the Flight Control Unit (Pixhawk). An unsteady offset of the accelerometers was found. A working solution is to calibrate the sensors before each flight which is, however, not very convenient. Another issue is the uncontrollable yaw of the drone while flying. This issue avoids flying safely the drone, so its gravity is high. Several reasons may cause this issue from a defective sensor to compatibility issues with the ESCs. Some causes can be fixed by parameter changes, other needs new hardware. The next steps are to run new tests by changing a possible cause and check the results. If it is still not working, as a plan B, the hardware can be changed to configurations that are known to work.

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

4. Drone software description & software integration overview

In this section, the SW modules running on board the drones will be briefly described. Details and specifications are provided in D2.3, to avoid duplicate reporting. Furthermore, an indicative status report of the SW implementation stage is presented here for each module. This status evaluation is subject to change (possibly significant one), according to future developments in WP3 and WP4 that will be integrated in WP5. Regarding AUTH work on visual analysis algorithm testing and speedup, it is reported in D4.1 and is overviewed D1.3, rather than here for presentation integrity. Ongoing work on these topics will be reported in D4.2.

4.1 Onboard Scheduler

The Onboard Scheduler receives the list of actions corresponding to the drone from the Mission Controller. Anytime the Mission Controller decides that re-planification is needed, it will compute a new plan and send new lists of actions to the drones involved. Then, the Onboard Scheduler is in charge of executing them sequentially, via the Action Executer module, and monitoring the action status.

It will be executed as a ROS node called *onboard_scheduler*. It is currently operational at a level of approximately 70%.

Integration overview

Tests performed: Planning architecture

Results: Tests with the modules involved in the planning architecture running together integrated: Onboard Scheduler, Action Executer, UAL, High-level Planner, Mission Controller, Event Manager. The Software-In-The-Loop simulator based on Gazebo was used for that. Computational load was low for these tests.

Hardware used: Desktop computer

Progress: 70%

4.2 Action Executer

Once the Onboard Scheduler receives the list of actions for the drone, it sends them sequentially to the Action Executer, which is responsible for the execution of these actions. For that, it will command the drone by means of the interface called UAL, and the Gimbal and Camera by means of the Gimbal and Camera interfaces, respectively.

The final output of the Action Executer to command the drone movement will be a *Velocity Tracking* command to be issued by means of the UAL. A *Drone Controller* will compute those velocity commands depending on the shooting action parameters, the target position

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

and velocity, and the drone position. For drone actions that involve a formation of drones, computation of the velocity commands will also depend on the position of the other drones, whose ID is provided in the drone action description, so that collision-free action execution is achieved.

In parallel, the *Gimbal Controller* computes the desired gimbal orientation such that the desired optical axis direction points towards the target, which requires knowledge of the target position and drone pose. Alternatively, the desired gimbal orientation can be computed based on the visual control errors provided by the Visual Shot Analysis module that encodes the error between desired and current 2D positions of the target in the image frame. Based on the orientation error, angular velocity commands are computed and provided to the Gimbal Interface module. The *Camera Controller* is used to control some parameters of the camera, such as focus.

This module will be executed as a ROS node called *action_executer*. This ROS node groups the Drone Action Handler, together with the Emergency Manoeuvers Executer, as well as the Drone, Gimbal and Camera Controllers. It is currently operational at a level of approximately 65%.

Integration overview

Tests performed: Planning architecture

Results: Tests with the modules involved in the planning architecture running together integrated: Onboard Scheduler, Action Executer, UAL, High-level Planner, Mission Controller, Event Manager. The Software-In-The-Loop simulator based on Gazebo was used for that. Computational load was low for these tests.

Hardware used: Desktop computer

Tests performed: Drone-2-drone avoidance

Results: Real flights with up to three drones executing the algorithm of obstacle avoidance in real time. They communicated their positions and avoided each other. Computational load on NUC was low.
Hardware used: Pixhawk and NUC

Progress: 65%

Tests performed: Simulated shooting actions

Results: Several Software-In-The-Loop (SITL) and Hardware-In-The-Loop (HITL) simulations were performed to test the execution of different shooting actions involving one, two, and three drones tracking a moving object of interest. The simulations were conducted using the STIL simulator based on Gazebo, the Action Executer and UAL modules running together, and a real gimbal as HITL. The latter was controlled based on the simulated

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

vehicles position. The orientation measurements provided by the gimbal controller (Basecam) were used to define the orientation of the simulated gimbal and show a synthetic image that is consistent with the real gimbal orientation, commanded to keep the target centred in the image plane.

Hardware used: Desktop computer, 3-axis gimbal with a Basecam Controller.

Progress: 60%

4.3 Gimbal Interface

This module is responsible for the interface between the physical gimbal with the BaseCam SimpleBGC 32-bit Controller and the ROS middleware. It converts the messages from/to the gimbal in the BaseCam Protocol to/from the ROS middleware.

It will be executed as a ROS node called *gimbal_interface*. It is currently operational at a level of approximately 80%.

Integration overview

Tests performed: Simulated shooting actions

Results: Several Software-In-The-Loop (SITL) and Hardware-In-The-Loop (HITL) simulations were performed to test the execution of different shooting actions involving one, two, and three drones tracking a moving object of interest. The simulations were conducted using the STIL simulator based on Gazebo, the Action Executer and UAL modules running together, and a real gimbal as HITL. The latter was controlled based on the simulated vehicles position. The orientation measurements provided by the gimbal controller (Basecam) were used to define the orientation of the simulated gimbal and show a synthetic image that is consistent with the real gimbal orientation, commanded to keep the target centred in the image plane.

Hardware used: Desktop computer, 3-axis gimbal with a Basecam Controller.

Progress: 60%

Tests performed: Gimbal control with a Motion Capture System

Results: The gimbal controller was tested, for a real gimbal equipped with a camera to point to a target (small RC car). A Motion Capture System was used to provide the 3D positions of the target and the gimbal base. With a static gimbal base, the gimbal was able to follow the target while it was moving, keeping it inside the image plane at all times.

Hardware used: Desktop computer, 3-axis gimbal with a Basecam Controller, small RC car, Motion Capture System, GoPro camera

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Progress: 80%

4.4 Camera Interface

This module is responsible for the interface between the Camera Controller in Action Executer and the BMMC camera.

It will be executed as a ROS node called *camera_control*.

Integration overview

Tests performed: Change camera parameters using the S.Bus protocol.

Results: A ROS node generates a S.Bus data stream to control camera settings from a computer. Camera parameters that can be controlled in absolute mode include Focus, Audio, Frame Rate, Codec. Camera parameters that can be controlled in speed mode include zoom, autofocus, ISO, Shutter Angle, and White Balance.

Hardware used: BMMC camera, laptop computer, simple transistor+resistors inverter circuit, USB to TTL RS232 converter.

Progress: 70%

4.5 UAL

The UAL (UAV abstraction layer) is the interface between the controller in the Action Executer and the autopilot. It receives velocity commands from the Action Executer and sends them to the autopilot. It also provides the pose and velocity of the drone in the global metric frame.

It will be executed as a ROS node called *ual*. It is currently operational at a level of approximately 90%.

Integration overview

Tests performed: Planning architecture

Results: Tests with the modules involved in the planning architecture running together integrated: Onboard Scheduler, Action Executer, UAL, High-level Planner, Mission Controller, Event Manager. The Software-In-The-Loop simulator based on Gazebo was used for that. Computational load was low for these tests.

Hardware used: Desktop computer

Tests performed: Drone-2-drone avoidance

Results: Real flights with up to three drones executing the algorithm of obstacle avoidance in real time. They communicated their positions and avoided each other. Computational load on

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

NUC was low. UAL was used to control the drones.

Hardware used: Pixhawk and NUC

Progress: 90%

4.6 Drone Localisation

This module is in charge of estimating the drone pose based on the on-board sensors available, namely GNSS positioning, LIDAR data, video streams from navigation and shooting cameras and the geometric map.

It will be executed as a ROS node called *drone_localization*. It is currently operational at a level of approximately 60%.

Integration overview

Tests performed: Geometric mapping of an outdoor site field and localisation

Results: Real flights with one drone performing a manual exploratory mission and building an accurate map. Then, random flights localising the drone in real time.

Hardware used: Velodyne HDL-32E, fixed ZED stereo camera (used as monocular), NUC, Pixhawk 1.2

Progress: 60%

4.7 Onboard 3D Target Tracker

This module estimates the 3D position of the target detected by the 2D tracking module. Basically, it will project 2D measurements on the image plane onto a 3D global system, by using camera pose. The module could exchange information with other instances on other drones to triangulate and get better 3D estimations.

It will be executed as a ROS node called *onboard_3d_target_tracker*. It is currently operational at a level of approximately 60%.

Integration overview

Tests performed: 3D target tracking from 2D estimations

Results: The Onboard 3D Target Tracker has been tested in simulations with a simple 2D image estimation from the state of the art. The Software-In-The-Loop simulator based on Gazebo was used for that.

Hardware used: Desktop computer

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Progress: 60%

4.8 2D Visual Information Analysis

The 2D Visual Information Analysis module consists of a visual object detector and visual object tracker of the main actors (targets) of each scenario. It receives an uncompressed video frame from the shooting camera in real-time and generates 2D positions of the tracked targets as bounding boxes.

This module will be executed as a ROS node called *master_visual_analysis*. It is currently operational at a level of approximately 70%.

Integration overview

Tests performed: Visual target (bicycle, football player, boat, human face) detection and tracking

Results: Tests with 2D Visual Information Analysis and Visual Shot Analysis running together integrated, getting video input from video file (when running on a desktop computer) or from the TX2 on-board camera. Computational load was low for these tests, but TX2 memory load was significant.

Hardware used: Desktop computer, TX2

Progress: 70%

4.9 Visual Shot Analysis

The Visual Shot Analysis module is initialised by the Set framing type service, which sets cinematographic shot specifications (desired target position on frame, desired framing shot type). The module constantly receives the target 2D position from the 2D tracker and calculates the current visual control error, according to the desired shot specifications.

This module will be executed as a ROS node called *VSA*. It is currently operational at a level of approximately 80%.

Integration overview

Tests performed: Visual control error computation

Results: Tests with 2D Visual Information Analysis and Visual Shot Analysis running together integrated, getting video input from video file (when running on a desktop computer) or from the TX2 on-board camera. Computational load was low for these tests, but TX2 memory load was significant.

Hardware used: Desktop computer, TX2

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Progress: 80%

4.10 Video streaming

Video streaming is about required software to acquire, compress, stream images to the Dashboard and Supervision Station that will decode and display shooting and navigation camera images. Developed solution is described in D2.3. We remind that two specific requirements have to be taken into account :

- **Requirement 1**: All frames have to be time stamped and it should be possible to synchronise them with other frames from other drones. The tolerance for drone A/V cameras synchronisation is about one frame, i.e. 40ms tolerance.
- **Requirement 2**: The end-to-end latency, i.e. the average time difference between the time at which a frame is decoded by a client on the ground station and the time at which the same frame has been captured by the A/V or navigation camera, has to be minimised.

Integration overview

Tests performed: Video streaming solution was tested in June 2018. Objective was to test “on table” hardware and software that will be then integrated within the MULTIDRONE system. For this test, the shooting camera stream (Full HD @ 30fps) was compressed @ 4Mbit/s and the navigation camera stream (VGA @30fps) was compressed 1.5Mbit/s.

Hardware used: Black magic shooting camera, uEye UI-1221LE-C navigation camera, Tegra X1 with the carrier board J130 from Auvidea company, the LTE onboard module with two antennas (WiFi is not used here), the LTE base station with a High power Remote Radio Head connected to a laptop that receives 2 video streams, decode them and display them.

Results:

- Streaming was OK, image quality was good and fluid
- Synchronisation thanks to RTCP packets combined with NTP was OK (less than 20us).
- Measured latencies were the following:
  - 50ms from shooting camera output to on board ROS publication
  - 75ms from shooting camera output to on ground ROS publication
  - Processing power: <2% for GPU ; ~130% (over 400%) for CPU

Hardware used: Black magic shooting camera, uEye UI-1221LE-C navigation camera, Tegra X1 with the carrier board J130 from Auvidea company, the LTE onboard module with two antennas (WiFi is not used here), the LTE base station with a High power Remote Radio Head connected to a laptop that receives video streams, decode them and display them.

This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Figure 4.10.1: Video streaming of onboard Cameras through LTE infrastructure.

Progress: 80%. The next step is to switch from Tegra X1 to Tegra X2. To this aim, an acquisition driver problem has to be solved (CSI driver for Tegra X2).

This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Appendix A: Initial drone prototype purchase list

Category	Hardware Definition	Product
Accessories	LiPo Battery charger	SKYRC Ultimate Duo 400W
Accessories	Battery charger for BMMCC	
Accessories	Additional battery for BMMCC	CANON battery LP-E6N
Accessories	Cleaning Kit for Lens	PHOTOGRAPHIC SOLUTIONS kit Pro Type 3
Accessories	Case for accessories	Multistar 250 Case
Accessories	Foam for case	Pluck et Pull
Accessories	Case for the drone	Multistar Transport Case for DJI-S1000
Accessories	LiPo safe bags	Turnigy® ignifuge Sac Batterie LiPoly
Accessories	Adapters to charge Transmitter batteries	Futaba radio charger
A-V Payload	Gimbal	iFlight G40
A-V Payload	Gimbal power connectors	JST-SH 2Pin Female Plug with 200mm Wire Pigtail
A-V Payload	Gimbal mount on DJI S1000+	Wire-Rope Isolator Mounting Plates Set
A-V Payload	Damping of the Gimbal mount	STO S15
A-V Payload	Camera	Blackmagic Micro Cinema Camera
A-V Payload	Motorised Lens	Lumix G X Vario PZ 14-42 mm f/3,5-5,6 ASPH
A-V Payload	SD card for camera	SANDISK carte mémoire SDXC Extreme Pro 95MB/s 512 Go
A-V Payload	µController	Teensy 3.5
A-V Payload	F/F connectors for µController	Fil de 300mm F/F à 20 Pins

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

A-V Payload	F/F connectors for µController	Kit de 65 Fils Connecteurs Divers
A-V Payload	Cable usb - µController to TX2	25cm USB 2.0 Type de type A mâle vers Mini B 5 broches mâle Câble de caméra 90 degrés Angle
A-V Payload	F/F Servo Cables type Futaba	20cm Female to Female Servo Lead
A-V Payload	M/M Servo Cables type Futaba	30CM mâle à mâle Servo Lead (JR) 26AWG (10pcs / set)
A-V Payload	Heatshrink for power security	Turnigy Heat Shrink Tube 2mm Black (1m)
A-V Payload	HDMI cable for camera to Splitter	30 cm type A/A
A-V Payload	HDMI cable for Splitter to TX2	30cm A/B
A-V Payload	HDMI Splitter	Splitter HDMI, SOWTECH(TM)
A-V Payload	DC/DC power adaptor for splitter	Adjustable Mini DC-DC step-down Module (LM2596 S; 3v-40v to 1.2-35v; 3A Max)
A-V Payload	Power connector with wire	2.1mm DC Plug Power avec 15cm Lead (5pcs)
A-V Payload	Power connector with wire	2.5mm DC Plug Power avec 15cm Lead (5pcs)
A-V Payload	Back-up radio emitter for A-V video	HD Amimon Connex Mini
A-V Payload	Cable Hdmi A D For splitter to Connex	ATOMOS câble micro HDMI / full HDMI
A-V Payload	PPM to Sbus converter	RMILEC haute précision PWM / PPM / SBus Signal Converter V2
Batteries	Batteries	Tattu Plus 10000mAh 22.2V 25C 6S1P Lipo Smart Battery Pack with AS150+XT150 plug
Core	Velcro for the batteries	400mm graphène Battery Strap
Core	Heat shrink for power cable 10 awg	Turnigy 6mm thermorétractable Tube 1M (Rouge)
Core	Heat shrink for power cable 10 awg	Turnigy 6mm thermorétractable Tube 1M (Noir)
Core	Power Cable batteries	Turnigy High Quality 10AWG Silicone Wire 2m (Red)
Core	Power Cable batteries	Turnigy High Quality 10AWG Silicone Wire 2m (Black)
Core	Power Connectors for PDB	Connecteurs XT150
Core	Power Connectors for PDB	Connecteurs XT90
Core	GNSS	HERE+ RTK GNSS - KIT POUR PIXHAWK 2.1

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Core	FCU	Pixhawk 2.1
Core	Auvidea uart connector	PICOBLADE 6 CIRCUIT 150MM
Core	Servo cable type Futaba	Twisted 15CM mâle à mâle Servo Lead (JR) 22AWG (10pcs / set)
Core	LTE/WiFi communication	Thales
Core	DC/DC power adaptor for Thales module	Adjustable Mini DC-DC step-down Module (LM2596 S; 3v-40v to 1.2-35v; 3A Max)
Core	USB between Pixhawk and Thales Module	UUUSBOTG8IN
Core	Micro SD Card for FCU	Kingston microSD 8 Go High Capacity + adaptateur SD
Core	Support for GPS antenna	MÂT GPS PLIABLE
Core	Power & current sensor FCU	POWER BRICK MINI (SONDE DE COURANT POUR PIXHAWK 2.1)
Core	Power connectors XT60	Nylon XT60 Connecteurs Mâle / Femelle (5 paires) AUTHENTIQUE
Core	Heat shrink for power cable 12 awg	Turnigy 5mm thermorétractable Tube 1M (Rouge)
Core	Heat shrink for power cable 12 awg	Turnigy 5mm thermorétractable Tube 1M (Noir)
Core	Heatshrink for power security	Turnigy High Quality 12AWG Silicone Wire 2m (Red)
Core	Heatshrink for power security	Turnigy High Quality 12AWG Silicone Wire 2m (Black)
Core	Cables pour Pixhawk 2.1	SET DE CÂBLES POUR PIXHAWK 2
Core	Telemetry back-up	KIT TÉLÉMÉTRIE POUR PIXHAWK - 433MHZ
Core	Mount	Items mounts - custom made if needed
Core	Servo cable extension M/F type Futaba	20cm JR 22AWG Twisted Extension Lead M to F 5pcs
Drone Payload	Navigational camera	Caméra FatShark 700TVL CMOS FPV
Drone Payload	RCA Splitter	Cinch Diviseur coupleur adaptateur
Drone Payload	Rca to rca cable M/M	Eltax Interconnect Vidéo (1,5 m)
Drone Payload	RCA to HDMI	Neoteck RCA vers HDMI Convertisseur
Drone Payload	USB - DC	Câble USB A mâle / mini USB B mâle - 0.30 m

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Drone Payload	OSD	OSD Minim
Drone Payload	Cable Pixhawk - OSD	Câble Silicone JST-GH vers JWT 6broches 28AWG
Drone Payload	Pin for OSD	Pin Header 1x30Pin 2.54mm
Drone Payload	Altimetre laser	LightWare SF11/C
Drone Payload	ADC connector on pixhawk for altimetre	CONNECTEUR MÂLE JST-GH À 3 CONTACTS
Drone Payload	DC/DC power adapter for LIDAR	Adjustable Mini DC-DC step-down Module (LM2596 S; 3v-40v to 1.2-35v; 3A Max)
Drone Payload	LIDAR	Leddartech M16
Drone Payload	Cable usb - LIDAR to computer	25cm USB 2.0 Type de type A mâle vers Mini B 5 broches mâle Câble de caméra 90 degrés Angle
Drone Payload	Onboard Computer	Nvidia TX2
Drone Payload	TX2 carrier board	Auvidea J140
Drone Payload	HDMI to M.2 (PCIe)	Magewell Eco Capture Dual HDMI M.2
Drone Payload	M.2 type M extender	R44SF - 30cm
Drone Payload	HDMI to CSI-2	B102 HDMI to CSI-2 Bridge (22 pin FPC)
Drone Payload	HeatSink for TX2	Heatsink and fan
Drone Payload	Computer	Intel NUC NUC7i5BNK
Drone Payload	Computer SSD	Kingston SSDNow M.2 SATA G2 120 Go
Drone Payload	Computer RAM	Corsair Value Select SO-DIMM DDR4 8 Go (2 x 4 Go) 2133 MHz CL15
Drone Payload	DC/DC for onboard computer	DC-DC Buck 12A 1.25-30V
Drone Payload	Communication cable between the 2 computers	Câble RJ45 catégorie 5e F/UTP 0,15 m
Drone Payload	USB - RJ45 adapters	Linksys USB3GIG-EJ
Drone Payload	Radio for FPV video flow	Emetteur vidéo 5.8GHz modulable FX800T-A

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

GCS	Back-up radio for pilot & Cameraman	Radio Futaba 14SG 2.4GHz + 1 Récepteur R-7008SB - Mode 2
GCS	Screen for backup pilot	7" Screen with 5,8GHz receiver and HDMI input
GCS	Support of the screen	FPV Moniteur support de montage / Argent (avec CG Adjustment)
GCS	Antenna for FPV	Antenne Foxeer RHCP 5,8 GHz
Materials for assembly	Tin coil	
Materials for assembly	Black Chatterton	
Materials for assembly	Red Chatterton	
Materials for assembly	Tape	
Materials for assembly	Double sided tape	
Materials for assembly	Double sided with foam with tape	
Materials for assembly	Epoxy glue	
Materials for assembly	Glue	
Materials for assembly	Self adhesive velcro	
Materials for assembly	Loctite	Threadlocker
Materials for assembly	Loctite	Threadlocker
Materials for assembly	Cable ties	

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

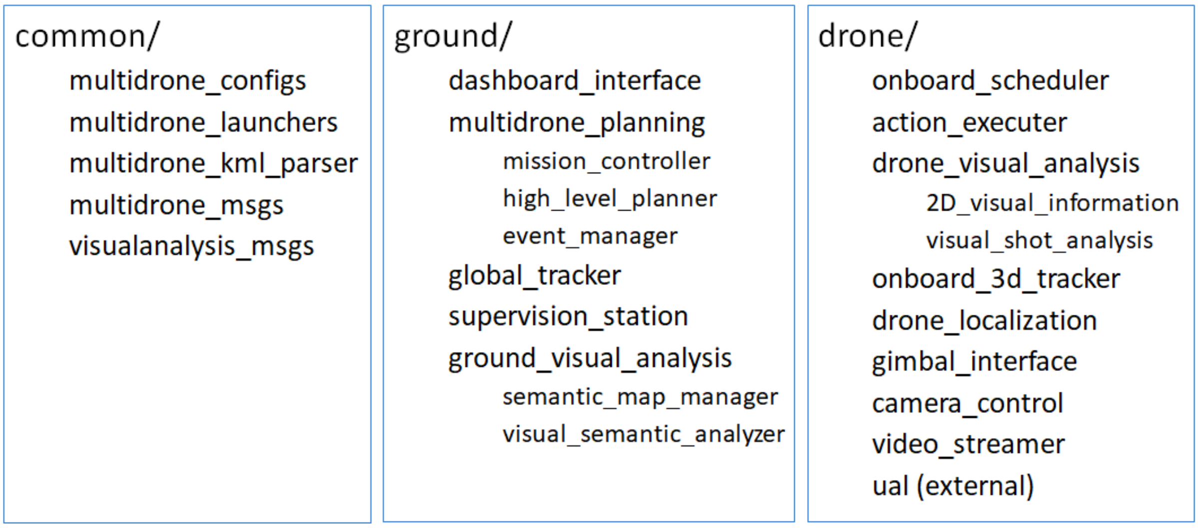
Materials for assembly	Cable ties	
Materials for assembly	Aluminium	
Materials for assembly	Wood plate	
Parachute	Parachute	GBS 10/350
Parachute	Emergency buzzer	
Parachute	Training pyro-actuator	
Parachute	Engine cutter	
Platform	Platform	DJI S1000+
Platform	Battery tray pour DJI s1000+	Part 2 S1000 battery tray
Platform	Soft Landing foam	
Spare parts	Propellers	Pack 8 propellers
Spare parts	Motors and arm	1 CW, & 1 CCW

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

Appendix B: SW development tools and issues

This Appendix refers to the entire SW development, rather than focusing on the on-drone SW only. The software design described in this document is being developed by all the involved partners using a Git repository in Bitbucket (https://bitbucket.org/multidrone_eu/multidrone_full.git). This repository allows to integrate from the beginning of the development all the modules programmed by different partners. It also allows versioning and communication tools between partners. USE has taken the lead on the software integration, and it is responsible of maintaining the repository in good health.

The repository already contains almost all the modules described in the document. It is organized in three folders: common, ground, drone. The following figure shows the organization in folders, ROS packages and modules if there is more than one per package.



MULTIDRONE architecture is based on the ROS middleware, which has several general purpose modules available that are extensively used and tested by the robotics community. Working with thoroughly tested software tools, prevents potential software defects that are common when developing from scratch.

Besides that, ROS implements a software bus through TCP/IP networking, similar to an enterprise service bus (ESB) in a service oriented architecture (SOA). Several heterogeneous modules communicate with a predefined set of messages that are passed by ROS between running processes in the same or different systems. The fixed set of message types enforces interoperability between modules at build time. The separation of functionality in different process spaces ensures the modularity of design and facilitates the benchmarking of each individual module for resource consumption, i.e. memory, CPU and GPU usage.

Additionally the ROS-based architecture ensures software isolation for each module. The interfaces for each module can be specified by means of ROS messages and services allowing the independent development of the modules.

The second aspect of software isolation is the services that are implemented by the MULTIDRONE modules for specific needs. Invocation and proper transmission of

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

information through parameters/return values are managed by ROS. This is preferable for increasing quality than direct calls between modules that exchange of information through the stack, potentially leading to memory-related faults.

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.

References

[D2.1] MULTIDRONE consortium. "*Deliverable D2.1: Multidrone media production requirements*".
[D2.2] MULTIDRONE consortium. "*Deliverable D2.2: Modular multi-actor system architecture, communication and functionality specification and design*".
[D2.3] MULTIDRONE consortium. "*Deliverable D2.3: Experimental dataset, revised set of specifications and design*".
[D5.2] MULTIDRONE consortium. "*Deliverable D5.2: Ground station implementation report*"
[D5.5] MULTIDRONE consortium. "*Deliverable D5.5: Integrated MULTIDRONE system report*"

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No *731667*.