4x Hardware Manual

Release 1.8

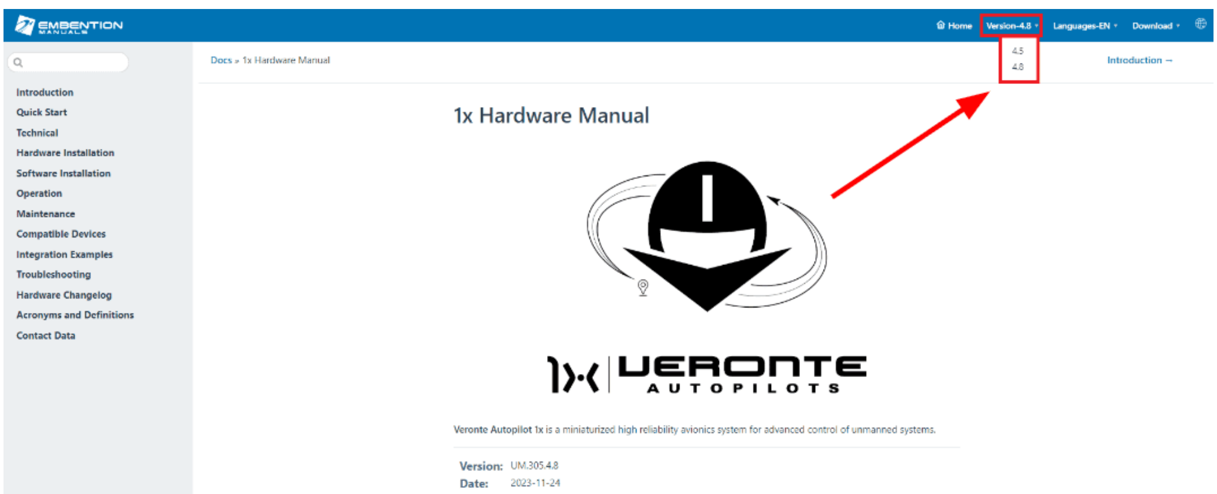
Embention

2025-02-12

CONTENTS

Veronte Autopilot 4x is a miniaturized avionics system for advanced control of unmanned systems, it includes triple redundancy to assure high reliability.

Warning: Select your version before reading any user manual. The following image shows where to select a version from any Embention user manual.

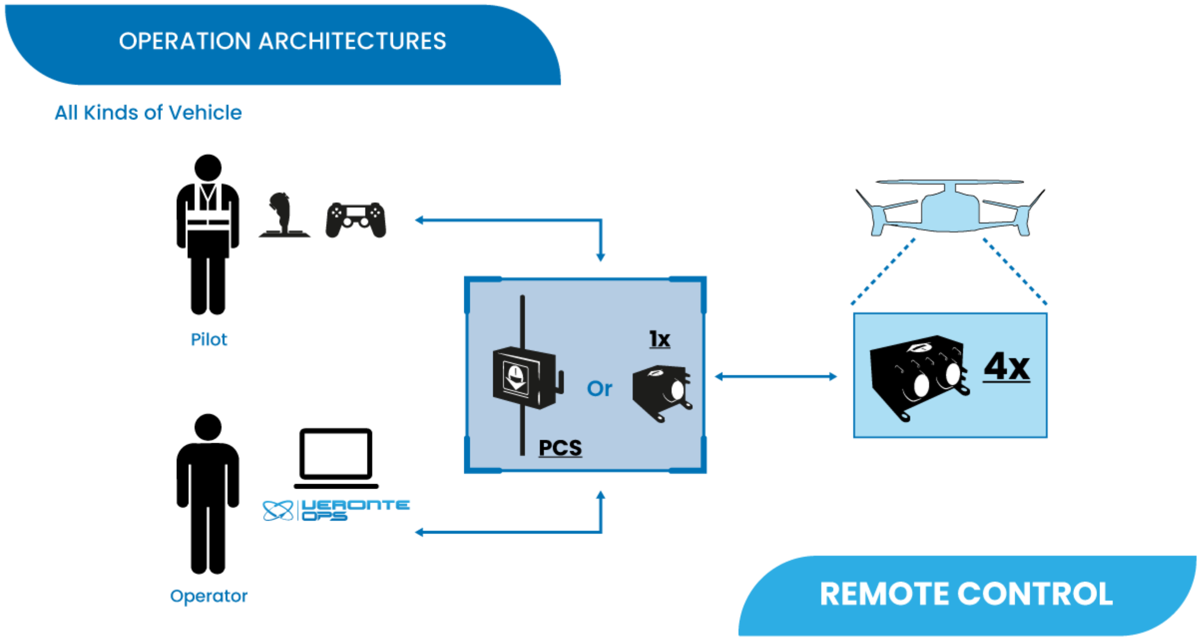


CHAPTER ONE

INTRODUCTION

Fig. 1: **Veronte Autopilot 4x**

Veronte Autopilot 4x is a miniaturized high reliability avionics system for advanced control of **unmanned systems**. This control system embeds a state-of-the-art suite of sensors and processors together with LOS and BLOS M2M datalink radio, all with reduced size and weight.



Veronte Autopilot 1x is designed to control any unmanned vehicle, either aircraft such as: multirotors, helicopters, airplanes, VTOL, blimps… as well as ground vehicles, surface vehicles or many others. Custom flight phases and control channels provide support for any aircraft layout and performance by using the same software and hardware for: UAS, RPAS, Drone, USV / ASV, UGV…

1.1 Applications

Autopilot 4x allows aircrafts to perform sensitive flight missions and transport valuable payloads with advanced safety conditions and high reliabilty. By installing a triple redundant core it is possible to extend the mean time between failures in systems. This control module is also suitable for both, fail-safe and fail-operational missions, extending the operability of the system.

1.2 Control diagram

Veronte Autopilot 4x is a **triple redundant** version of **Veronte Autopilot 1x**. It includes three complete Veronte Autopilot modules fully integrated with dissimilar arbiters to detect system failures and select the module in charge of control. In worst case scenario, if arbiters do not emit any control signal, the **Autopilot 1x** number 1 will take the control of the aircraft.

Each **Autopilot 1x** receives all signals, but only the selected one sends information through multiplexed channels. Arbiters select which **1x** will control the aircraft (and send signals) using watchdog messages. In addition, it is possible to connect an external fourth autopilot.

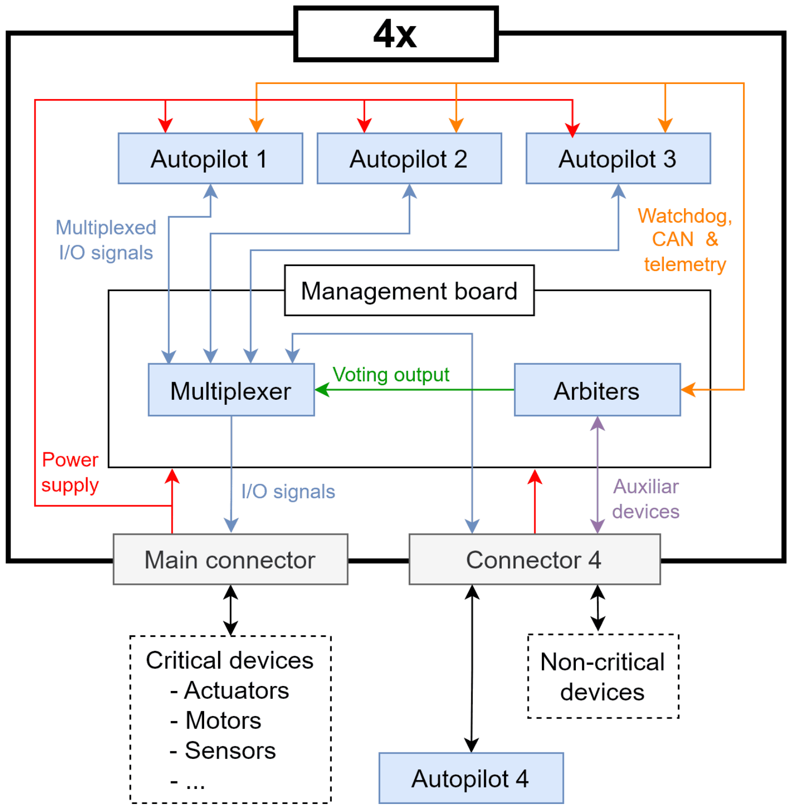


Fig. 2: **General diagram**

CHAPTER

TWO

QUICK START

This user manual covers the *mechanical* and *electrical* assembly. This document includes references to *install and configure software*.

Veronte Autopilot 4x is the main element in our FCS for UAV.

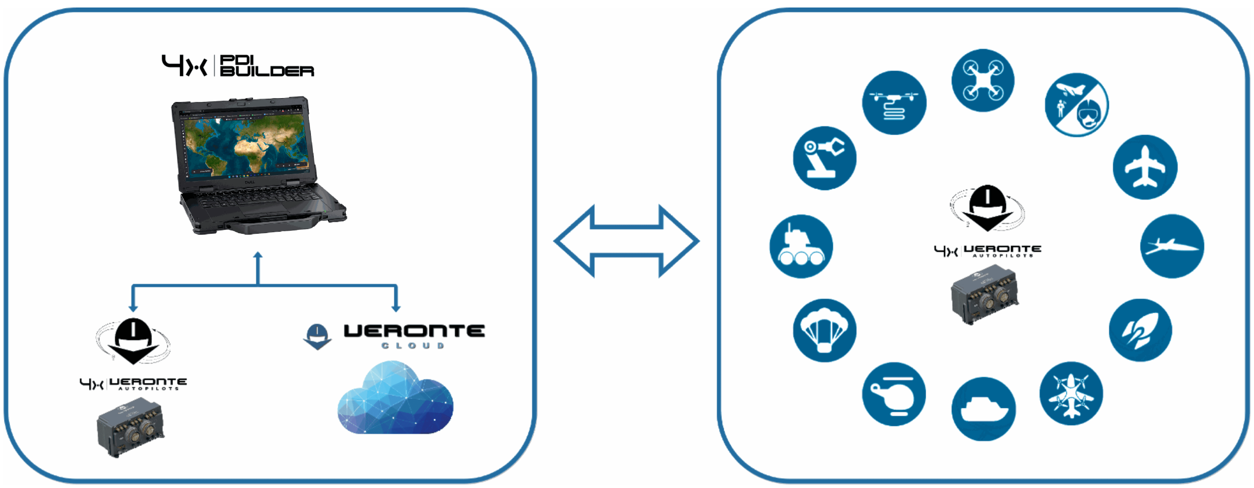
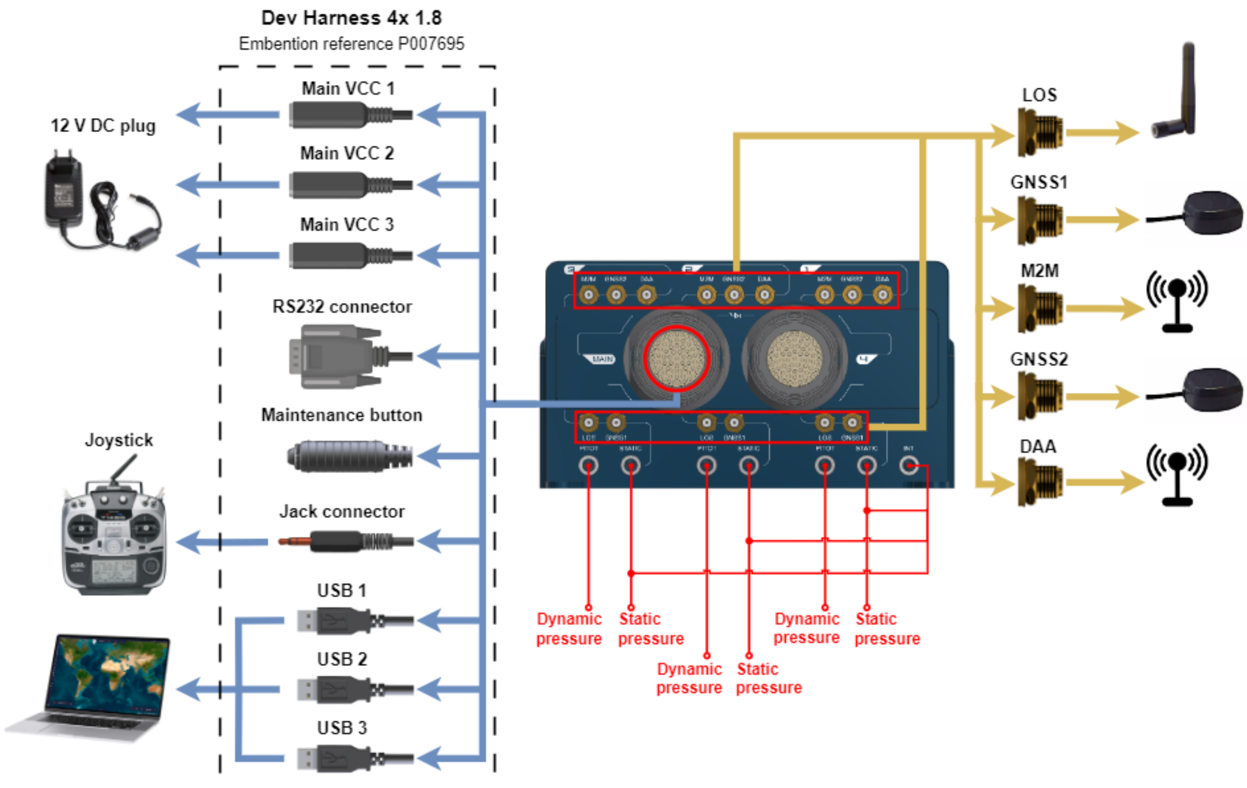


Fig. 1: **System Overview**

Veronte Autopilot 4x contains all electronics and sensors required to properly execute all the UAV functions. A Veronte-based FCS contains the following elements:

- A **Veronte Autopilot 4x** installed in a vehicle to be controlled. This autopilot executes GNC algorithms in real time to accomplish the planned mission and transport the payload.
- **Veronte Ops** - Software dedicated to mission planning, configuration and operation. It allows the user to monitor connected UAS in real time, to interact with them and to replay previous missions for post-flight analysis.
- An Autopilot 1x **GND unit** or **PCS** linked between **Veronte Ops** and **Veronte Autopilot 4x**. They support manual and arcade modes with conventional joysticks.

2.1 Basic Connection Diagram



For further information on the *Dev Harness 4x 1.8* connectors, refer to the *Dev Harness 4x 1.8 - Hardware Installation* section of the present manual.

2.2 Warnings

- This user manual includes references to manuals for software applications. Select your software version to read them.
- Power out of range can cause irreversible damage to the system. Please read carefully the manual before powering the system.
- Users **must not power on a Veronte Autopilot 4x** without **a suitable antenna** or **50 Ω load** connected to the DAA SSMA if the unit has an ADS-B and/or 4G module activated.

> **Danger:** This may damage the Autopilot 4x unit.

CHAPTER

THREE

TECHNICAL

3.1 General description

Veronte Autopilot 4x is a **triple redundant** version of **Veronte Autopilot 1x**. It includes three complete **Autopilot 1x** modules fully integrated with dissimilar arbiters for detecting system failures and selecting the module in charge of the control. The autopilot selected has the master controls actuators and communications. The following diagrams summarize the connections between autopilots and the elements of the flight control system.

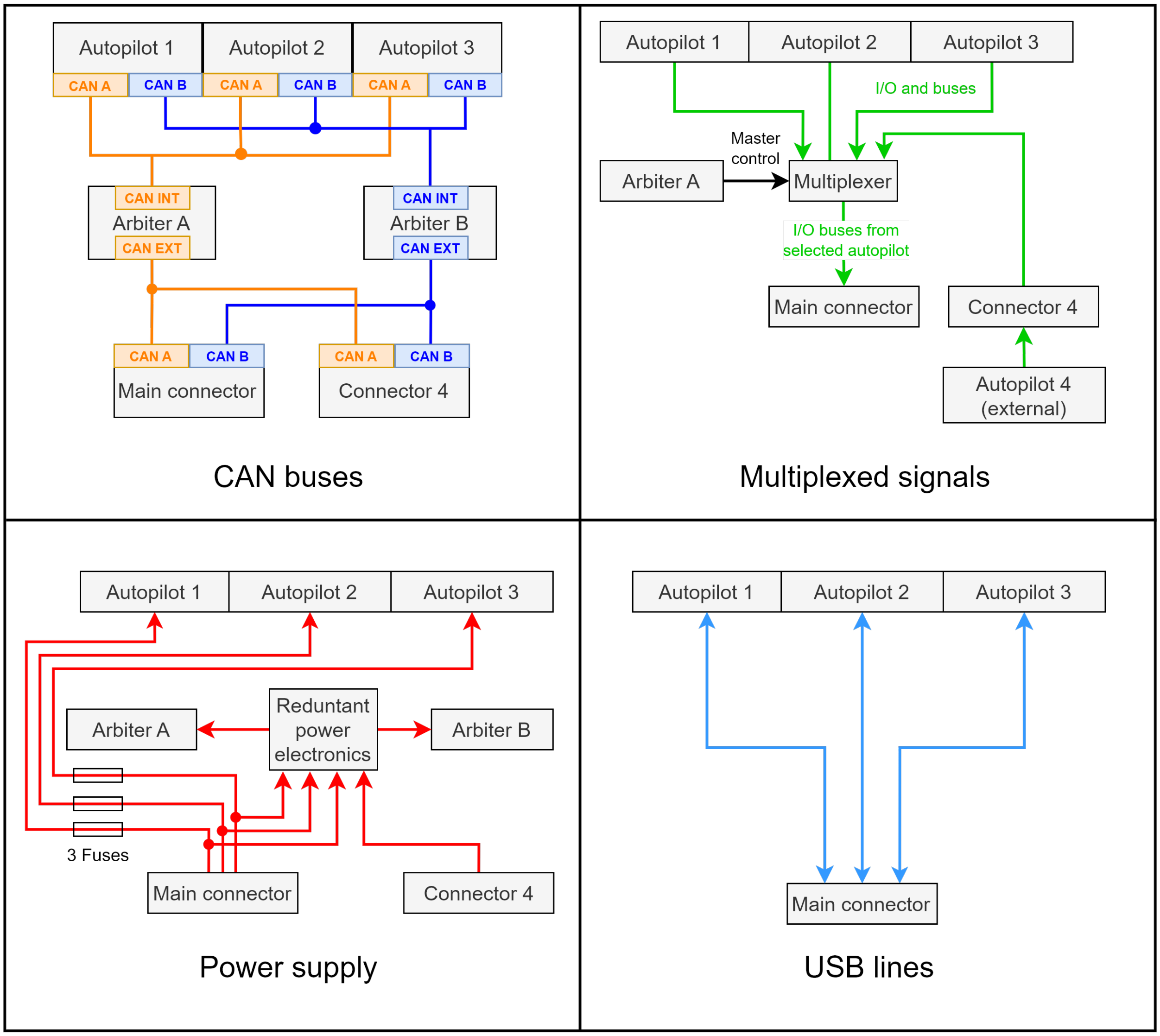


Fig. 1: **Internal diagrams**

Important: Apart from CAN buses, all communications are established only with arbiter A (I2C, RS-232, RS-485 and ARINC).

In addition, only arbiter A controls the autopilots multiplexing.

Each **Veronte Autopilot 4x** contains all the electronics and sensors to properly execute all the functions needed to control the UAV. **Autopilot 4x** executes in real time guidance, navigation and control algorithms for the carrying airframe. It controls propulsion systems and signals processing from different sensors: accelerometers, gyroscopes, magnetometer, static pressure, dynamic pressure, GNSS and externals.

Additional I/O ports are available for connection of an external control system in case it is required (for example another **Autopilot 1x**). In case of using an additional control system, it will be included in the redundant scheme. Veronte systems provide the system full dissimilarity for high demanding environments, as required by civil aviation authorities.

Datalink communications can be also redundant, being possible to install inside the autopilot 3 radios with different frequencies. For example, it allows to have two radios working in the 900 MHz frequency and one in 2.4 GHz, so in case there is any issue in the 900 MHz bandwidth the module connected to the 2.4 GHz bandwith will take the control. In addition, an external radio can be controlled as a critical device using the serial port in the redundant connector.

Veronte Autopilot 4x also includes two separate flight termination voting logics, completely dissimilar and implemented with simple hardware, with the purpose of giving the internal three **Veronte Autopilots 1x** a way to decide by consensus if a flight termination signal should be activated or not. This flight termination signal is employed to activate emergency systems, such as parachutes.

All three modules are managed by a **Management Board**; it includes voting algorithms to manage the module in charge of vehicle control. This device compares data from all modules in real time and processes it for discarding any autopilot module with undesired performance.

The arbitration algorithm in **Veronte Autopilot 4x** is based on a scoring system. Each autopilot must send continuously a set of arbitration variables to the arbiters in order to calculate the score for each unit. Then, based on scores and current arbitration mode, the **4x** will use the autopilot with the highest score.

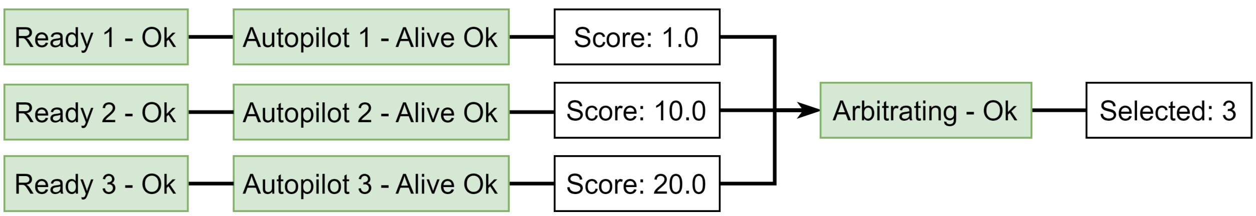


Fig. 2: **Arbitration diagram**

3.2 Variants

Variant name	Reference
W/O DAA	P006984
With remote ID	P006146
With ADS-B	P006147

3.3 Mechanical and Electrical specifications

All sensors are located inside the **Autopilots 1x**. To know their specifications read the Sensor Specifications - Technical section of the **1x Hardware Manual**.

Variable	Value
Weight	W/O DAA: 615 g (±2 g) With Remote ID or ADS-B: 632 g (±2 g) With *Damping System*: + 76 g
Temperature range	-40 ºC to 65 ºC
Protection Rating	IP67
Power input voltage	6.5 to 36 V
Power consumption	Up to 15 W
Maximum acceleration	32 g

3.3.1 Dimensions

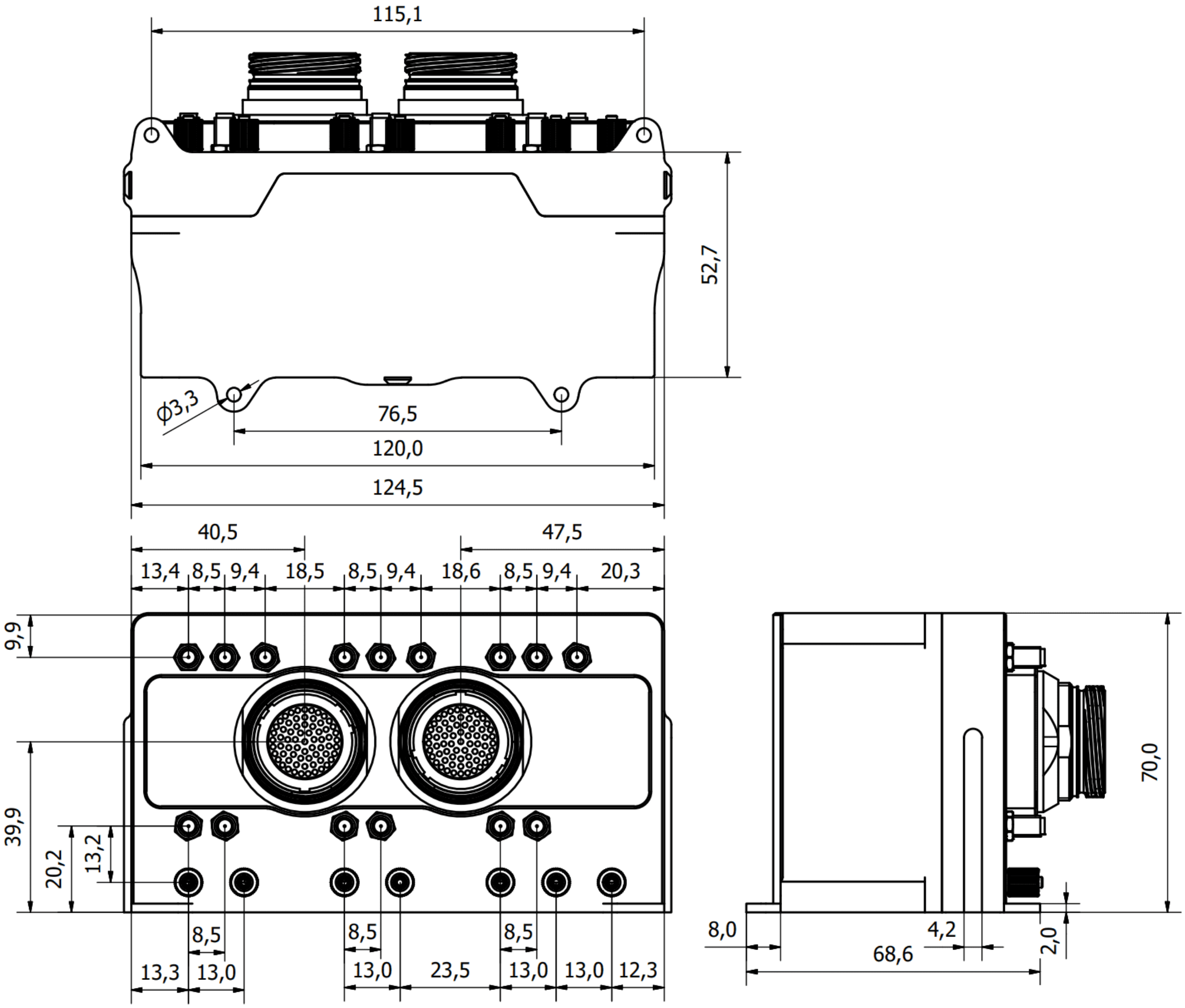


Fig. 3: **Veronte Autopilot 4x dimensions (mm)**

M3 screws are recommended for mounting. In saline environments such as coastal and oceanic, the screw material should be stainless steel.

3.4 Interfaces

3.4.1 Connector layout

The three inner **Autopilots 1x** are connected to the **MAIN Connector** and the **Management Board** is connected to **Connector 4**. In case of using an **external autopilot**, it must be plugged to the **Connector 4** according to the *Pinout - Hardware Installation* section of this manual.

Each inner **Autopilot 1x** has asigned a connector block with its respective number. All blocks have the same connectors with the same functions.

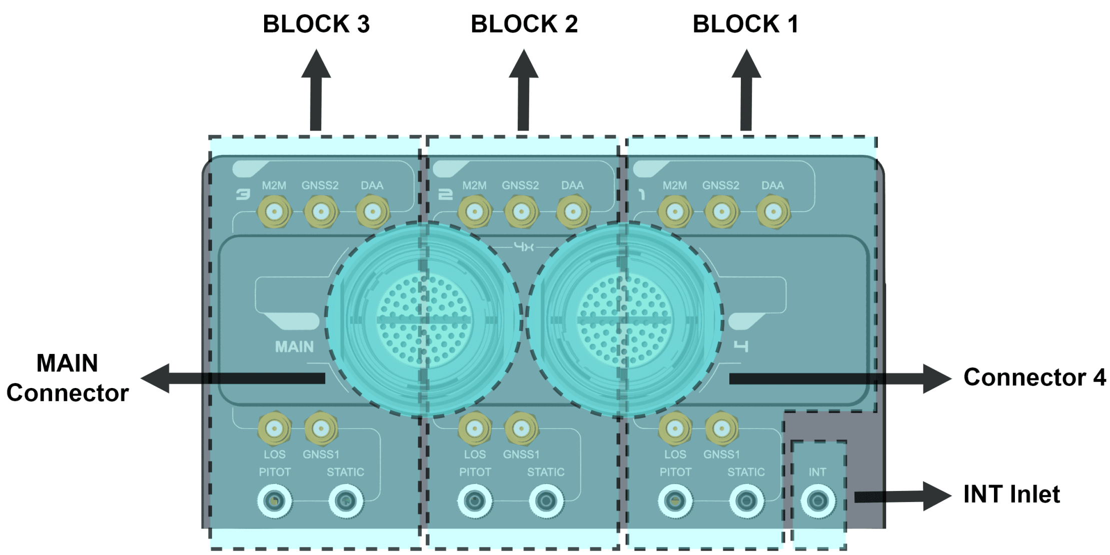


Fig. 4: **Veronte Autopilot 4x connectors**

- **M2M**: SSMA connector for machine to machine communication.

> **Warning:** If the BLOS module is enabled, a suitable antenna must be connected to this SSMA port.
>
> The `4G Antenna` with the Embention reference **P000112** is recommended.

- **GNSS1**: SSMA connector for global navigation satellite system 1.
- **GNSS2**: SSMA connector for global navigation satellite system 2.
- **DAA**: SSMA connector for ADS-B or remote ID.

> **Warning:** When using ADS-B or remote ID, there must be an adequate antenna or load connected to the DAA SSMA.

- **LOS**: SSMA connector for line of sight communications.
- **PITOT**: Dynamic pressure port (Int. D. 2.5 mm x Out. D. 4 mm) of each internal autopilot.
- **STATIC**: Static pressure port (Int. D. 2.5 mm x Out. D. 4 mm) for static pressure sensor 1 of each internal autopilot.
- **INT Inlet**: Static pressure port (Int. D. 2.5 mm x Out. D. 4 mm) for static pressure sensor 2. This port is common for all internal autopilots.

Note: Each autopilot employs both static pressure ports for sensor redundancy, then Y tubing connection is strongly recommended.

> **Warning:** The static pressure port *STATIC* is always used by **Autopilot 4x** to calculate speed (using the difference of pressure between ports *STATIC* and *PITOT*), no matter which sensor is selected in configuration.

3.4.2 Mating connectors

Abbreviation	Autopilot 4x connector	Mating connector
GNSS	SSMA Jack female for GNSS antenna	SSMA male Plug, low-loss cable is recommended. Active Antenna GNSS: • Gain min 15 dB (to compensate signal loss in RF Cable) • Gain max 50 dB • Maximum noise figure 1.5 dB • Power supply 3.3 V • Max current 20 mA
DAA	SSMA Jack female for ADS-B or remote ID	SSMA male Plug, low-loss cable is recommended
M2M	SSMA Jack female for M2M antenna	
LOS	SSMA Jack female for RF antenna	
MAIN	Main Connector HEW.LM.368.XLNP	Mating connector: FGW.LM.368.XLCT (Embention reference **P005550**) Mating harnesses: • Veronte Autopilots: Dev Harness 4x 1.8 (Embention reference **P007695**) • Veronte Harness Blue 68P (Embention reference **P001114**) • Veronte Autopilots: HIL Harness 4x 1.8 (Embention reference **P007739**)
4	Connector 4 HER.LM.368.XLNP	Mating connector: FGR.LM.368.XLCT (Embention reference **P005654**) Mating harness: Veronte Harness Yellow 68P (Embention reference **P001118**)

CHAPTER

FOUR

HARDWARE INSTALLATION

4.1 Mechanical

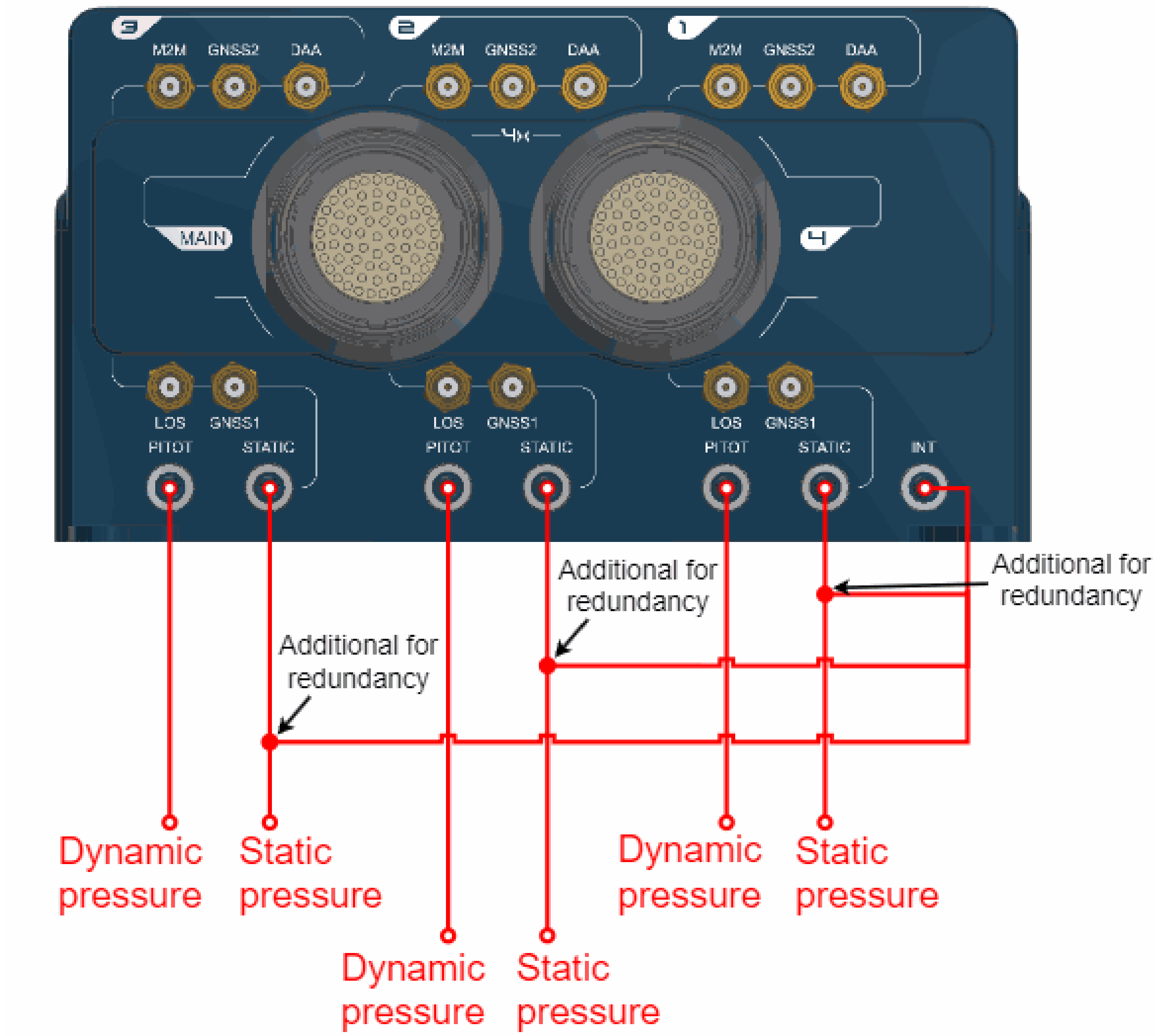
Veronte Autopilot 4x is covered with an aluminium enclosure with enhanced EMI shielding and IP protection, with 750 g as total weight.

4.1.1 Pressure lines

Veronte Autopilot 4x has seven redundant pressure input lines; four for static pressure to determine the absolute pressure and three for pitot in order to determine the dynamic pressure.

For the fittings it is recommended to use a polyurethane tube of 2.5 mm inner diameter and 4 mm outer diameter.

- **Pressure Intake**
  - Pressure intakes must be located in order to prevent clogging.
  - Do not install pressure intakes on the propeller flow.
  - Design pressure tubing path in order to avoid tube constriction.
- **Static Pressure**
  - It is not recommended to use inside fuselage pressure if it is not properly vented.
- **Pitot Tube**
  - Pitot tube must be installed facing the airflow.
  - It is recommended to install it near the aircraft's x axis in order to avoid false measures during manoeuvres.
  - For low-speed aircraft it is recommended at least 6.3 mm tubes to prevent any rain obstruction.



4.1.2 Location

The location of **Veronte Autopilot 4x** has no restrictions. It is only required to configure its relative position respect to the centre of mass of the aircraft and the GNSS antenna. The configuration of the location of each **Autopilot 1x** can be easily configured using 1x PDI Builder.

4.1.3 Orientation

The orientation of **Veronte Autopilot 4x** has no restrictions either. It is only required to configure axes respect to the aircraft by means of a rotation matrix or a set of correspondences between axes. The configuration of the orientation can be easily configured using **1x PDI Builder** for each **Autopilot 1x**.

Axes are printed on the **Autopilot 4x** box. Aircraft coordinates are defined by the standard aeronautical conventions, shown in the following figure.

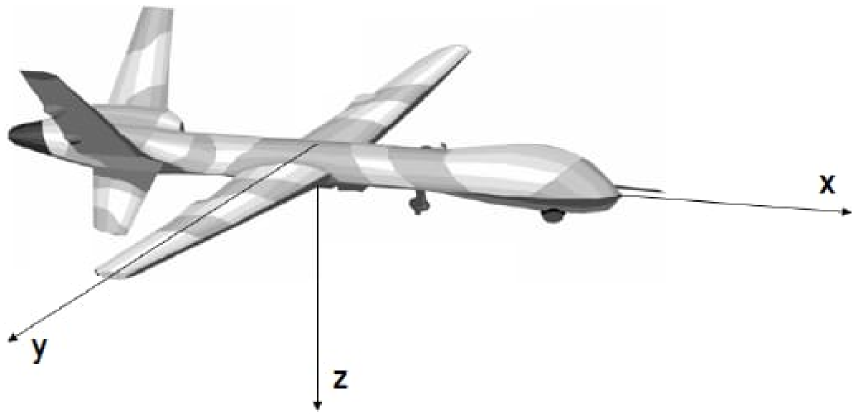


Fig. 1: **Aircraft Coordinates (Standard Aeronautical Convention)**

4.1.4 Vibration Isolation

Although **Veronte Autopilot 4x** rejects noise and high-frequency vibration modes with electronic filters, there may be situations where external isolation is needed.

Autopilot 4x can be mounted in different ways in order to reject the airframe vibration, but it is recommend to use the *Damping System* designed for that porpuse. It covers a wide frequency range of different aircraft types.

Note: The user should take into account that wiring should be loose enough so that vibrations are not transmitted to **Autopilot 4x**.

4.1.4.1 Damping System

Embention offers the **Damping System** as a solution to isolate **Veronte Autopilot 4x** from vibrations.

Important: Only effective with **Autopilot 4x** in horizontal position.

This damping system weighs **76 g**.

Fig. 2: **Damping System**

> **Warning:** The **Damping System** is designed for version 1.8 of **Autopilot 4x**.

4.1.4.1.1 Dimensions

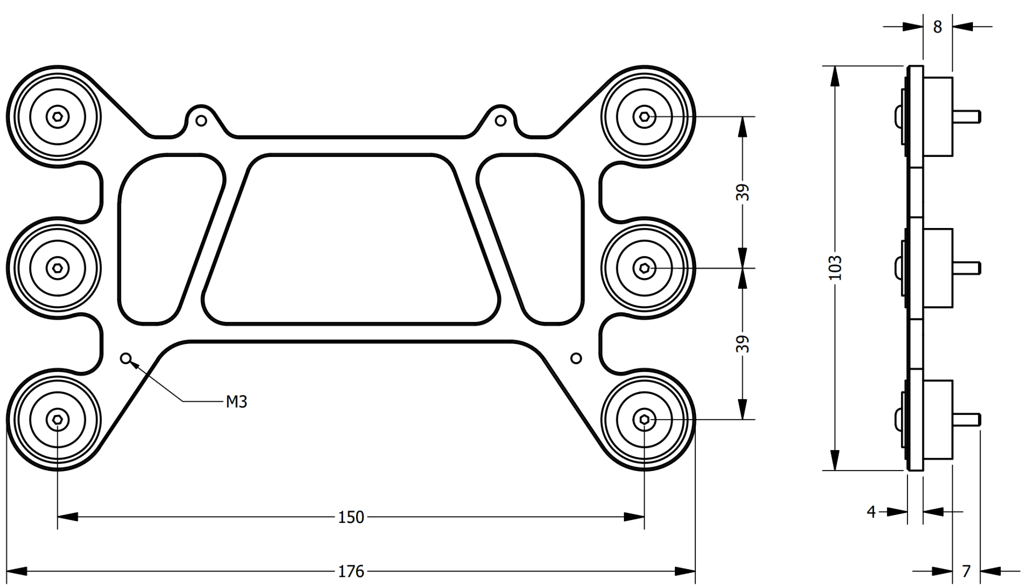


Fig. 3: **Damping system dimensions (mm)**

4.1.4.1.2 Assembly steps

To assembly the Damping System into a vehicle with an **Autopilot 4x**, read the following steps.

1. Remove the six nuts located under the platform.

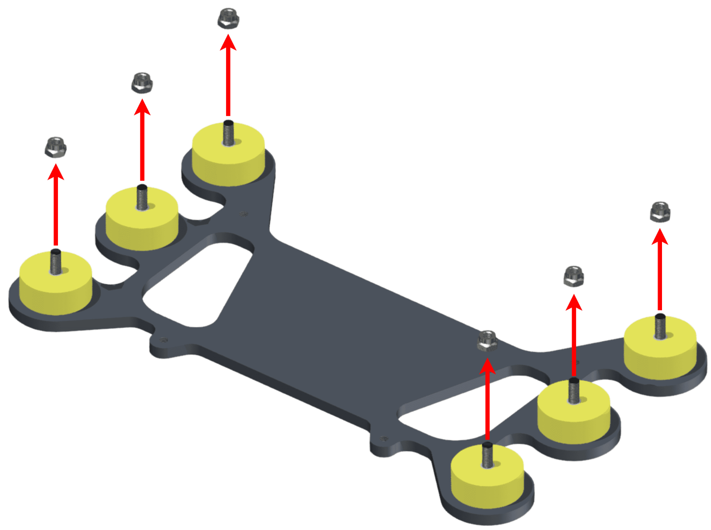


Fig. 4: **Step 1**

2. Screw the platform on the aircraft frame. The included screws have M3.

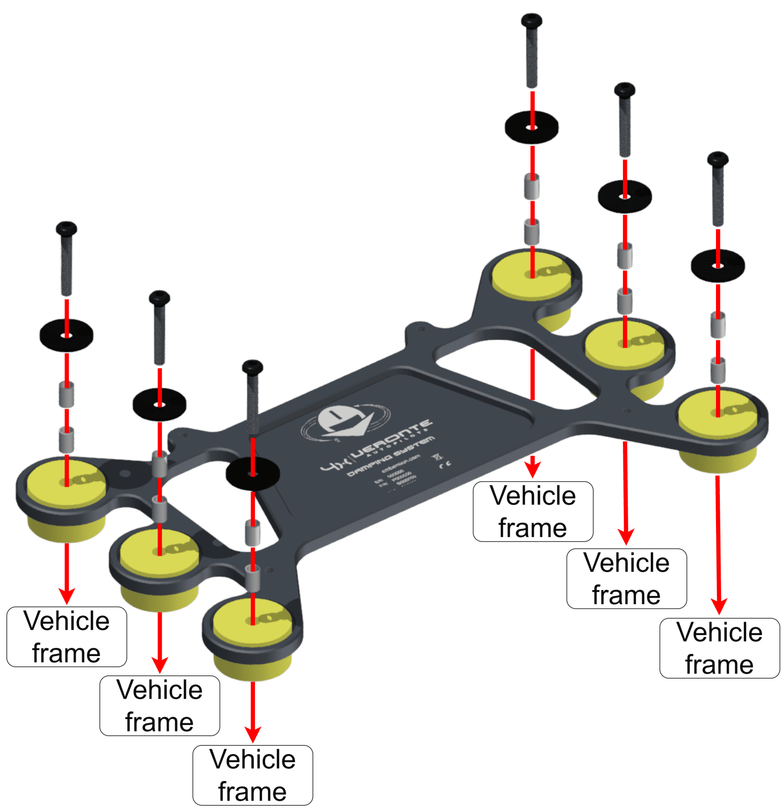


Fig. 5: **Step 2**

3. Screw the **Autopilot 4x** on the **Damping system**.

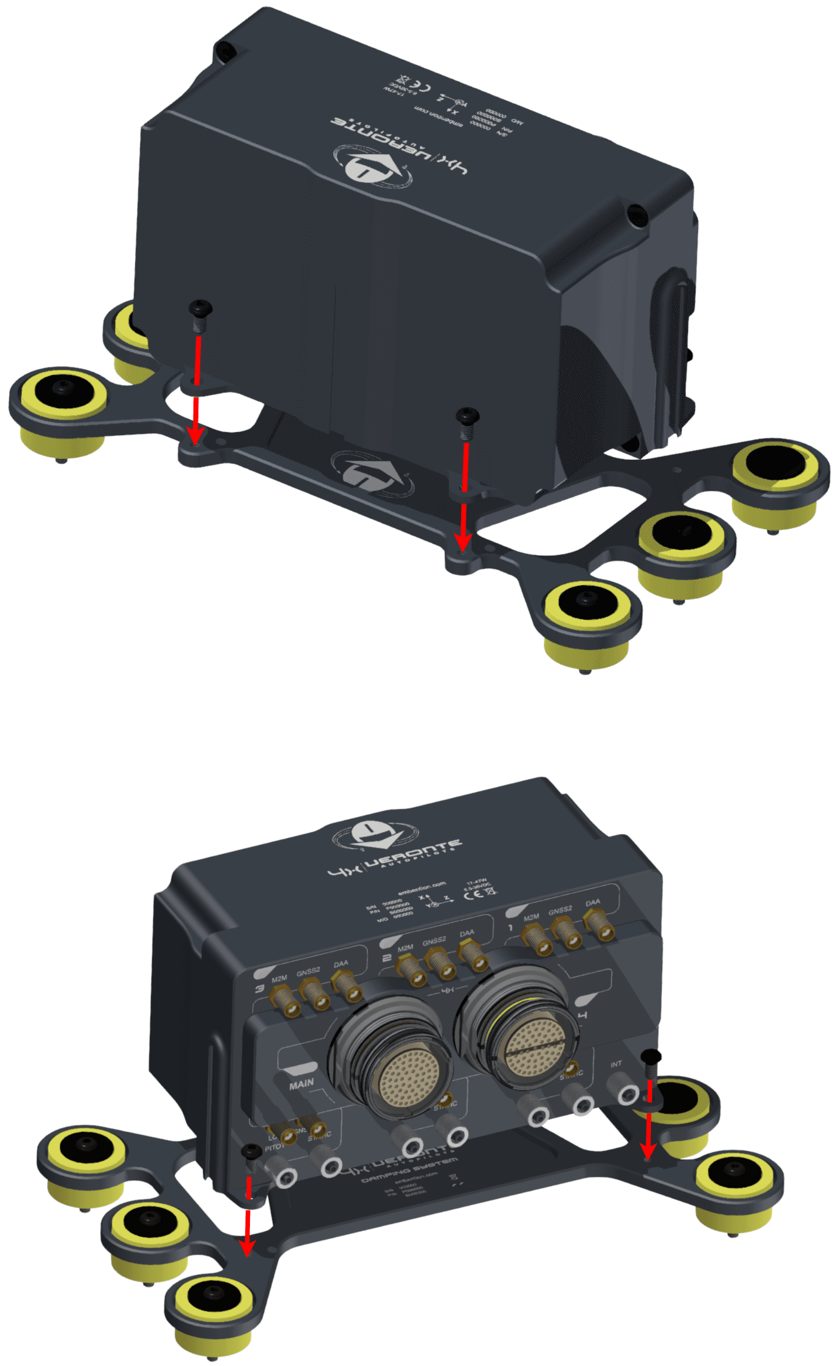


Fig. 6: **Step 3**

Fig. 7: **Result**

4.2 Antenna Integration

The system uses different kinds of antenna to operate, they must be installed on the airframe. Here you can find some advice for obtaining the best performance and for avoiding antenna interferences.

- **Antenna Installation**
  - Maximize separation between antennas as much as possible.
  - Keep antennas far away from alternators or other interference generators.
  - Always isolate the antenna ground panel from the aircraft structure.
  - Make sure antennas are securely mounted.
  - Always use high-quality RF wires minimising the wire length.
  - Always follow the antenna manufacturer manual.
  - SSMA connections shall be tightened applying 1Nm of torque.
  - For all-weather aircraft, insert SSMA lightning protectors.
- **GNSS Antenna**
  - Antenna top side must point to the sky.
  - Install them on a top surface with direct sky view.
  - Never place wires or parts made of memetal or carbon above the antenna.
  - It is recommended to install antennas on a small ground plane.
  - For all-weather aircrafts, insert SSMA lightning protectors.

- **Recommended specifications for GNSS antennas**

Specifications	Range
Antenna frequency L1	1561.098 MHz to 1602 MHz
Antenna frequency L2	1207.14 MHz to 1246 MHz
Amplifier gain	17 dB to 35 dB
Out-of-band rejection	40 dB **Note:** Higher values are preferable. 30dB is considered the minimum acceptable value.
Polarization	RHCP (Right-Hand Circular Polarization)
Minimum supply voltage	2.7 V to 3.3 V
Maximum supply current	50 mA

4.3 Electrical

4.3.1 Power supply

Autopilot 4x can use unregulated DC (6.5 V to 36 V) for the internal Veronte autopilots and also for the **Management Board**. All power supply pins are not common. It is possible to supply them with different voltages since they are internally protected with diodes. Nonetheless, all power supply pins must be connected to a power supply, in order to guarantee that **Autopilot 4x** will work in case of failing one of them. These pins are summarized in the following table:

Connector	PIN	Signal
Main Connector	68	BAT_0
	67	BAT_1
	64	BAT_2
Arbiter Connector	68	VCC_ARBITER

LiPo batteries between 2S and 8S can be used without voltage regulation. Remaining battery can be controlled by the internal voltage sensor and by configuring the voltage warnings on the PC application.

For higher voltage installations, voltage regulators must be used. For dimensioning voltage regulators take into account that a blocked servo can activate regulator thermal protection.

Warning: Power **Veronte Autopilot 4x** out of the given range can cause irreversible damage to the system. Please read carefully the manual before powering the system.

Autopilots and servos can be powered by the same or different batteries. In case there are more than one battery on the system, a single point ground union is needed to ensure a good performance. The ground signal should be isolated from other system ground references (e.g. engines).

It is recommendable to use independent switches for autopilot and motor/actuators. During the system initialization, PWM signal will be fixed to low level (0V), please make sure that actuators/motor connected support this behavior before installing a single switch for the whole system.

4.3.2 Power Domains

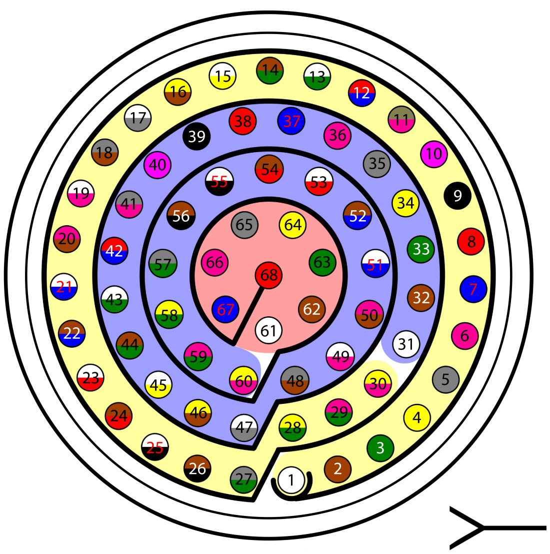
Veronte Autopilot 4x has two internal power domains (A and B). Power domains are isolated each other; hence, if one of them fails, the other one will remain operational. Many of the signals on the *pinout* are powered by one power domain.

Separated power domains allow to manage redundancy against internal power failures. For example, if the aircraft requires to measure a critical temperature of an external device, the user can use two different analog sensors and connect them to analog inputs of different power domains. One option for this example is connecting them to pin 38 (domain A) and pin 22 (domain B). Thus, if there is an internal failure and power domain A fails, the autopilot will still read measurements from pin 22.

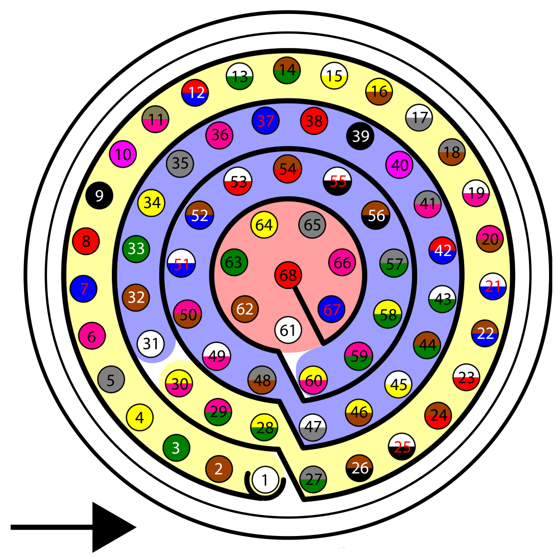
Any power supply pin (pins 64, 67 and 68 from Main Connector and pin 68 from Arbiter Connector) powers both domains. Nonetheles, this is independent of the power supply for each internal **Autopilot 1x**. Since pins 64, 67 and 68 from Main Connector power one single autopilot (appart from power domains).

4.3.3 Pinout

Warning: Pins can transfer 2 A as maximum current.



Pinout of Main and Arbiter connectors



Pinout of Harness plug of both connectors

Warning: Check the pin number before connecting. The color code is repeated 3 times due to the amount of pins. First section (yellow) corresponds to pins 1-30, the second section (blue) to pins 31-60 and the third one (red) to pins 61-68. Pin number increases following the black line of the pictures above: counterclockwise for the connector and clockwise for the plug.

4.3.3.1 Main Connector pinout

PIN	Signal	Type	Internal Power Domain	Description
1	I/O_0_MUXED	I/O	A	MUXED PWM / Digital I/O signal (0-3.3V) **Warning:** Each pin withstands a maximum current of 1.65 mA.
2	I/O_1_MUXED	I/O	B	
3	I/O_2_MUXED	I/O	A	
4	I/O_3_MUXED	I/O	B	
5	I/O_4_MUXED	I/O	A	
6	I/O_5_MUXED	I/O	B	
7	I/O_6_MUXED	I/O	A	
8	I/O_7_MUXED	I/O	B	
9	GND*	GROUND		Ground pin for signals 1-8
10	I/O_8_MUXED	I/O	A	MUXED PWM / Digital I/O signal (0-3.3V) **Warning:** Each pin withstands a maximum current of 1.65 mA.
11	I/O_9_MUXED	I/O	B	
12	I/O_10_MUXED	I/O	A	
13	I/O_11_MUXED	I/O	B	
14	I/O_12_MUXED	I/O	A	
15	I/O_13_MUXED	I/O	B	
16	I/O_14_MUXED	I/O	A	
17	I/O_15_MUXED	I/O	B	
18	GND*	GROUND		Ground pin for signals 10-17
19	MUXED_RS232_TX	OUTPUT	A	MUXED RS-232 output
20	MUXED_RS232_RX	INPUT	A	REDUNDANT RS-232 input
21	V2_USB_DP	I/O		Autopilot 2 USB positive data line
22	ANALOG_3	INPUT	B	REDUNDANT analog input (0-36V)
23	ANALOG_4	INPUT	B	
24	V2_USB_DN	I/O		Autopilot 2 USB negative data line
25	CANA_ARB_P	I/O	A	CAN-bus interface. It supports data rates up to 1 Mbps. A 120 Ohm Zo is required and twisted pair is recommended.
26	CANA_ARB_N	I/O	A	
27	GND*	GROUND		GROUND pin for buses (except USB)
28	CANB_ARB_P	I/O		CAN-bus interface. It supports data rates up to 1 Mbps. A 120 Ohm Zo is required and twisted pair is recommended

continues on next page

Table 1 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
29	CANB_ARB_N	I/O		
30	V2_USB2_GND	GROUND		Autopilot 2 USB ground
31	I2C_CLK	OUTPUT A		MUXED Clock line for I2C bus (0.3V to 3.3V)
32	I2C_DATA	I/O	A	MUXED data line for I2C bus
33	GND*	GROUND		Ground for 3.3V power supply
34	V1_ARB_TX	OUTPUT		Microcontroller UART transmitter for Autopilot 1
35	V1_ARB_RX	INPUT		Microcontroller UART receiver for Autopilot 1
36	V2_ARB_TX	OUTPUT		Microcontroller UART transmitter for Autopilot 2
37	V2_ARB_RX	INPUT		Microcontroller UART receiver for Autopilot 2
38	ANALOG_0	INPUT	A	REDUNDANT analog input (0-36V)
39	ANALOG_1	INPUT	A	
40	ANALOG_2	INPUT	A	
41	GND*	GROUND		Ground signal for buses
42	V3_USB_DP	I/O		Autopilot 3 USB positive data line
43	V3_USB_DN	I/O		Autopilot 3 USB negative data line
44	GND*	GROUND		Ground signal for buses
45	V3_ARB_TX	OUTPUT		Microcontroller UART transmitter for Autopilot 3
46	V3_ARB_RX	INPUT		Microcontroller UART receiver for Autopilot 3
47	GND*	GROUND		Ground signal for buses
48				
49	V3_USB3_GND	GROUND		Autopilot 3 USB ground
50	OUT_RS485_P	OUTPUT	B	MUXED non-inverted output RS-485 bus

continues on next page

Table 1 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
51	OUT_RS485_N	OUTPUT	B	MUXED inverted output RS-485 bus
52	IN_RS485_N	INPUT		REDUNDANT inverted inout RS-485 bus
53	IN_RS485_P	INPUT		REDUNDANT non-inverted input RS-485 bus
54	RS485_GND	GROUND		Ground for RS-485 bus
55	EQEP_A	INPUT	A for autopilots 1 and 2 B for autopilot 3	Encoder quadrature redundant input A (0-5V)
56	EQEP_B	INPUT		Encoder quadrature redundant input B (0-5V)
57	EQEP_S	INPUT		Encoder strobe redundant input (0-5V)
58	EQEP_I	INPUT		Encoder index redundant input (0-5V)
59	GND*	GROUND		Autopilot 3 ground pin
60	V1_USB_DP	I/O		Autopilot 1 USB positive data line
61	V1_USB_DN	I/O		Autopilot 1 USB negative data line
62	V1_USB1_GND	GROUND		Autopilot 1 USB ground
63	GND*	GROUND		Ground signal for buses
64	BAT_2	POWER		Autopilot 3 power supply (6.5 to 36V)
65	GND*	GROUND		Autopilot 2 ground pin
66	GND*	GROUND		Autopilot 1 ground pin
67	BAT_1	POWER		Autopilot 2 power supply (6.5 to 36V)
68	BAT_0	POWER		Autopilot 1 power supply (6.5 to 36V)

Warning: Common grounds are marked with *.

Note: MUXED (multiplexed) signals are connected to the **Autopilot 1x** decided by the **Management Board**, then only the selected autopilot is connected to MUXED pins.

REDUNDANT signals are connected to the three inner autopilots, so all of them receive the same REDUNDANT signals.

To know the differences between version 1.2 and 1.8 (this one), read the *Pinout changes from Autopilot 4x 1.2 - Troubleshooting* section of the present manual.

4.3.3.2 Arbiter Connector pinout

Although being the same component, **Main Connector** and **Arbiter connector** are polarized differently, but they have different mechanical connections to avoid wiring swapping.

PIN	Signal	Type	Internal Power Domain	Description
1	I/O_0_EXTERNAL	I/O	A	External MUXED PWM / Digital I/O signal (0-3.3V). In case of employing an additional external autopilot, its pins **I/XX** must be connected here. Each signal **I/O_XX_EXTERNAL** will be sent to **I/XX** of **Main Connector** if the arbiter commands it. **Warning:** Input current must be limited to 25 mA for each I/O EXTERNAL pin.
2	I/O_1_EXTERNAL	I/O	B	
3	I/O_2_EXTERNAL	I/O	A	
4	I/O_3_EXTERNAL	I/O	B	
5	I/O_4_EXTERNAL	I/O	A	
6	I/O_5_EXTERNAL	I/O	B	
7	I/O_6_EXTERNAL	I/O	A	
8	I/O_7_EXTERNAL	I/O	B	
9	I/O_8_EXTERNAL	I/O	A	
10	I/O_9_EXTERNAL	I/O	B	
11	I/O_10_EXTERNAL	I/O	A	
12	I/O_11_EXTERNAL	I/O	B	
13	ARBITER_ANALOG_I	INPUT		Arbiter analog input (0-36V)
14	EXTERNAL_ANALOG_0	OUTPUT	A	External analog signal (0-3V). This is the analog signal from **ANALOG_0** on **Main Connector**, which is reduced from 0-36V to 0-3V.

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
15	EXTERNAL_ANALOG_1	OUTPUT	A	External analog signal (0-3V). This is the analog signal from **ANALOG_1** on **Main Connector**, which is reduced from 0-36V to 0-3V.
16	EXTERNAL_ANALOG_2	OUTPUT	A	External analog signal (0-3V). This is the analog signal from **ANALOG_2** on **Main Connector**, which is reduced from 0-36V to 0-3V.
17	EXTERNAL_ANALOG_3	OUTPUT	B	External analog signal (0-3V). This is the analog signal from **ANALOG_3** on **Main Connector**, which is reduced from 0-36V to 0-3V.

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
18	FTC_VOTING_B	OUTPUT	B	This pin is an **open drain output** (0 - 48V), which is open or connected to GND depending on the **FTS signals of the Autopilots 1x**. This logic is implemented at the **Voting Stage**, explained in detail in the *Flight Termination System* section of this manual. Use this pin for an emergency device; for example the ground of a relay that activates a parachute. FTC_VOTING_A (pin 53) and FTC_VOTING_B (pin 18) do the same function, but they have dissimilarity.
19	EXT_RS232_TX	INPUT	A	In case of employing an additional external autopilot, its pin **RS 232 TX** must be connected here. If arbiter decides to multiplex this signal, it will be transmitted to **MUXED_RS232_TX** on **Main Connector** with RS232 protocol.

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
20	EXT_RS232_RX	OUTPUT	A	In case of employing an additional external autopilot, its pin **RS 232 RX** must be connected here. If arbiter decides to multiplex this signal, it will be transmitted to **MUXED_RS232_TX** on **Main Connector** with RS232 protocol.
21	IN_RS485_P	OUTPUT		This pin is connected with **IN_RS485_P** from **Main Connector**
22	IN_RS485_N	OUTPUT		This pin is connected with **IN_RS485_N** from **Main Connector**
23	EXT_OUT_RS485_P	INPUT	B	In case of employing an additional external autopilot, its pin **OUT_RS485_P** must be connected here. If arbiter decides to multiplex this signal and **EXT_DETECT** of **Arbiter Connector** is connected to GND, it will be transmitted to **OUT_RS485_P** on **Main Connector** with RS232 protocol

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
24	EXT_OUT_RS485_N	INPUT	B	In case of employing an additional external autopilot, its pin **OUT_RS485_N** must be connected here. If arbiter decides to multiplex this signal and **EXT_DETECT** of **Arbiter Connector** is connected to GND, it will be transmitted to **OUT_RS485_N** on **Main Connector** with RS232 protocol
25	CANA_P_ARB_A	I/O		This pin is connected with **CANA_ARB_P** from **Main Connector**
26	CANA_N_ARB_A	I/O		This pin is connected with **CANA_ARB_N** from **Main Connector**
27	ARBITER_ANALOG_B	INPUT		Arbiter analog input (0-36V)
28	CANB_P_ARB_B	I/O		This pin is connected with **CANB_ARB_P** from **Main Connector**
29	CANB_N_ARB_B	I/O		This pin is connected with **CANB_ARB_N** from **Main Connector**
30	OUT_RS485_ARB_P	OUTPUT	A	Non-inverted output for arbiter A RS-485 bus
31	OUT_RS485_ARB_N	OUTPUT	A	Inverted output for arbiter A RS-485 bus
32	IN_RS485_ARB_N	INPUT	A	Inverted output for arbiter A RS-485 bus

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
33	IN_RS485_ARB_P	INPUT	A	Non-inverted input for arbiter A RS-485 bus
34	TX_OUT_P	OUTPUT	A	Arbiter A ARINC positive output
35	TX_OUT_N	OUTPUT	A	Arbiter A ARINC negative output
36	RIN1_ARINC_P	INPUT	A	Arbiter A ARINC positive input
37	RIN1_ARINC_N	INPUT	A	Arbiter A ARINC negative input
38	GND*	GROUND		Ground pin for buses
39	SCL_A_OUT_ARB	OUTPUT	A	Clock signal for arbiter A I2C bus
40	SDA_A_OUT_ARB	I/O	A	Data signal for arbiter A I2C bus
41	DSP_232_RX_B	INPUT	A	Arbiter A RS-232 input B
42	DSP_232_TX_B	OUTPUT	A	Arbiter A RS-232 output B
43	DSP_232_RX_A	INPUT	A	Arbiter A RS-232 input A
44	DSP_232_TX_A	OUTPUT	A	Arbiter A RS-232 output A
45	GND*	GROUND		Ground pin for analog signals
46	ARBITER_ANALOG_0	INPUT	A	Arbiter A analog input (0-36V)
47	ARBITER_ANALOG_1	INPUT	A	
48	ARBITER_ANALOG_2	INPUT	A	
49	ARBITER_ANALOG_3	INPUT	A	
50	ARBITER_ANALOG_4	INPUT	A	
51	ARBITER_ANALOG_5	INPUT	A	
52	ARBITER_ANALOG_6	INPUT	A	

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
53	FTC_VOTING_A	OUTPUT	A	This pin is an open drain output (0 - 48V), which is open or connected to GND depending on the **FTS signals of the Autopilots 1x**. This logic is implemented at the **Voting Stage**, explained in detail in *the Flight Termination System* section of this manual. Use this pin for an emergency device; for example the ground of a relay that activates a parachute. FTC_VOTING_A (pin 53) and FTC_VOTING_B (pin 18) do the same function, but they have dissimilarity.
54	GPIO_8_ARB	I/O	A	Arbiter A PWM / digital I/O signal (0-3.3V)
55	GPIO_9_ARB	I/O	A	
56	WD_EXT	INPUT	A	Watchdog signal from external autopilot to arbiter A (0-3.3V)
57	EXT_DETECT	INPUT	A	Connect to **GND** if external autopilot is connected, otherwise keep open
58	GND*	GROUND		Ground signal for GPIO
59	GPIO_0_ARB	I/O	A	Arbiter A PWM / digital I/O signal (0-3.3V)
60	GPIO_1_ARB	I/O	A	
61	GPIO_2_ARB	I/O	A	
62	GPIO_3_ARB	I/O	A	
63	GPIO_4_ARB	I/O	A	
64	SEL_AP	OUTPUT	A	CAP signal indicating the AP selected
65	FTS1_OUT	OUTPUT	A	Deadman signal from comicro

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
66	FTS2_OUT	OUTPUT	A	System OK bit
67	GND*	GROUND		Management Board ground
68	VCC_ARBITER	POWER		Power supply for Management Board (6.5 to 36 V)

Warning: Common grounds are marked with *.

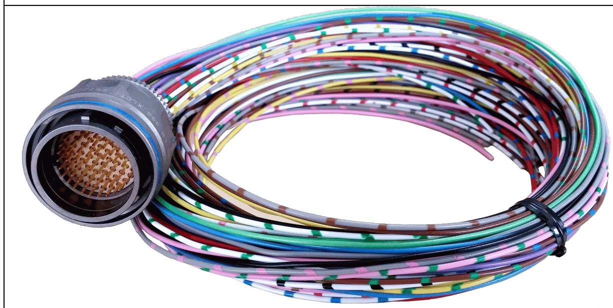
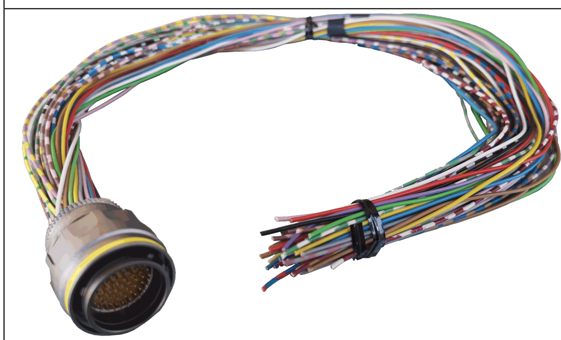
Important: Apart from CAN buses, all communications are established only with arbiter A (I2C, RS-232, RS-485 and ARINC).

To know the differences between version 1.2 and 1.8 (this one), read the *Pinout changes from Autopilot 4x 1.2 - Troubleshooting* section of the present manual.

4.3.4 Harnesses

A wire harness is a structured assembly of cables and connectors used to organize and manage wiring in electrical and electronic systems. It is designed to ensure a tidy and secure installation of cables, preventing tangles, electromagnetic interference, and facilitating maintenance.

Veronte Autopilot 4x 1.8 has the following compatible harnesses:

<table>
<tr><td colspan="2">For Main Connector</td></tr>
<tr><td>Veronte Harness Blue 68P</td><td>Dev Harness 4x 1.8</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Harness available on demand with the Embention reference P001114</td><td>Harness available on demand with the Embention reference P007695</td></tr>
<tr><td colspan="2">For Connector 4: Veronte Harness Yellow 68P</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td colspan="2">Harness available on demand with the Embention reference P001118</td></tr>
</table>

4.3.4.1 Dimensions

- **Harness Blue/Yellow 68P wire gauge:** 22 AWG
- **Cables lenght:** 52 cm
- **Harness plug dimensions:**

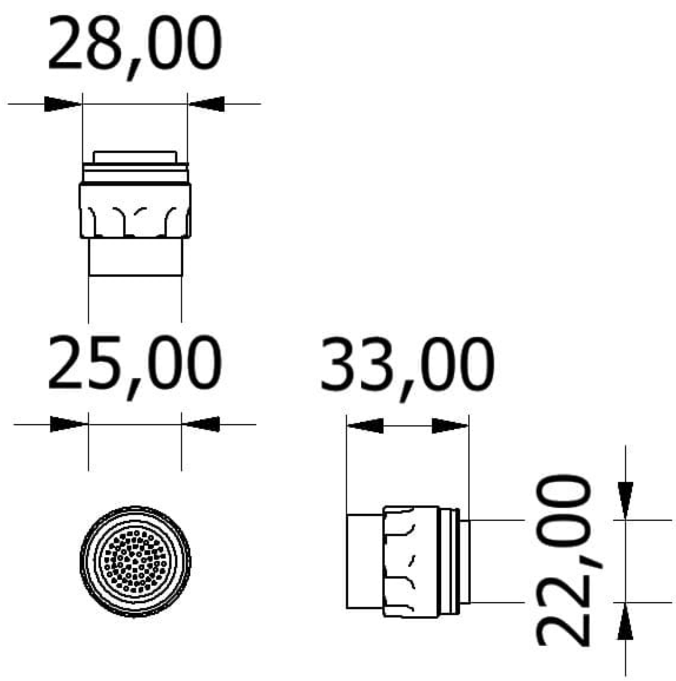


Fig. 8: **Connector FGW.LM.368.XLCT/FGR.LM.368.XLCT dimensions (cm)**

4.3.4.2 Pinout

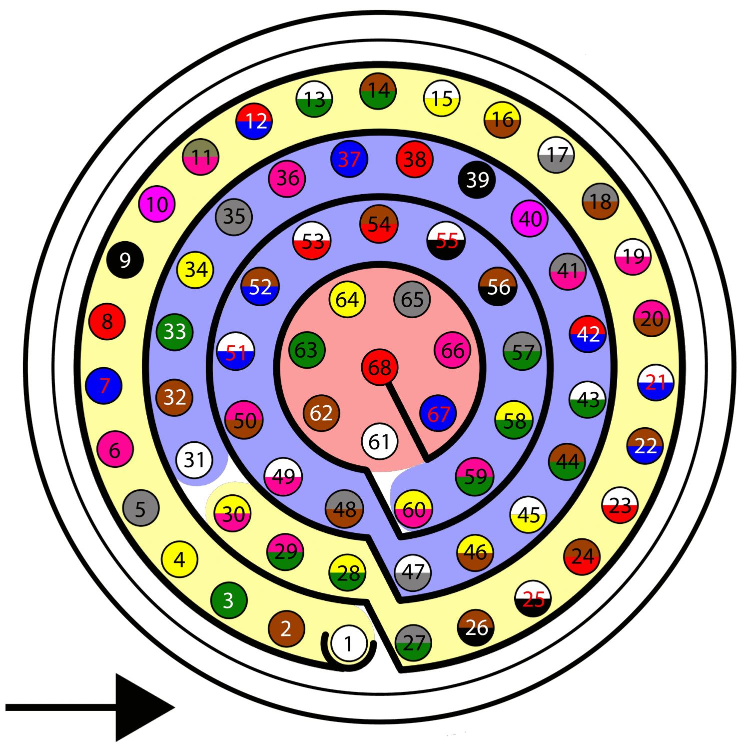


Fig. 9: **Harness plug - FGW.LM.368.XLCT (frontal view)**

4.3.4.2.1 Veronte Harness Blue/Yellow 68P

- The pinout of the Veronte Harness **Blue** 68P is the same as the *Main Connector pinout* above. The **color code** of the harness wires is given below.
- The pinout of the Veronte Harness **Yellow** 68P is the same as the *Arbiter Connector pinout* above. The **color code** of the harness wires is given below.

> **Warning:** Check the pin number before connecting. The color code is repeated 3 times due to the amount of pins. First section (yellow) corresponds to pins 1-30, the second section (blue) to pins 31-60 and the third one (red) to pins 61-68. Pin number increases following the black line of the pictures above: counterclockwise for the connector and clockwise for the plug.

PIN	Color code	PIN	Color code
1	White	35	Gray
2	Brown	36	Pink
3	Green	37	Blue
4	Yellow	38	Red
5	Gray	39	Black
6	Pink	40	Violet
7	Blue	41	Gray - Pink
8	Red	42	Red - Blue
9	Black	43	White - Green
10	Violet	44	Brown - Green
11	Gray - Pink	45	White - Yellow
12	Red - Blue	46	Yellow - Brown
13	White - Green	47	White - Gray
14	Brown - Green	48	Gray - Brown
15	White - Yellow	49	White - Pink
16	Yellow - Brown	50	Pink - Brown
17	White - Gray	51	White - Blue
18	Gray - Brown	52	Brown - Blue
19	White - Pink	53	White - Red
20	Pink - Brown	54	Brown - Red
21	White - Blue	55	White - Black
22	Brown - Blue	56	Brown - Black
23	White - Red	57	Gray - Green
24	Brown - Red	58	Yellow - Green
25	White - Black	59	Pink - Green
26	Brown - Black	60	Yellow - Pink
27	Gray - Green	61	White
28	Yellow - Green	62	Brown
29	Pink - Green	63	Green
30	Yellow - Pink	64	Yellow
31	White	65	Gray
32	Brown	66	Pink
33	Green	67	Blue
34	Yellow	68	Red

4.3.4.2.2 Dev Harness 4x 1.8

The pinout of this harness is the same as the *Main Connector pinout* above. In addition, this harness has some connectors already implemented for easy operation. Below is detailed information on which pins these connectors are connected to:

Connector	PIN	Signal
Main VCC 1	66	GND
	68	BAT_0
Main VCC 2	65	GND
	67	BAT_1
Main VCC 3	59	GND
	64	BAT_2
RS232 connector	18	GND
	19	MUXED_RS232_TX
	20	MUXED_RS232_RX
Maintenance button	31	I2C_CLK
	32	I2C_DATA
Jack connector	47	GND
	55	EQEP_A
USB 1	60	V1_USB_DP
	61	V1_USB_DN
	62	V1_USB1_GND
USB 2	21	V2_USB_DP
	24	V2_USB_DN
	30	V2_USB2_GND
USB 3	42	V3_USB_DP
	43	V3_USB_DN
	49	V3_USB3_GND

4.3.5 Flight Termination System (FTS)

The **Flight Termination System** determines the behavior of **Autopilot 4x** in case of severe failure. There are FTS output signals of 4x for failures of **Autopilots 1x** (*FTC_VOTING_A* and *FTC_VOTING_B*) and for failure of the **Arbitration system** (*FTS1_OUT* and *FTS2_OUT*).

4.3.5.1 Autopilots 1x failure - Voting Stage

Autopilot 4x FTS works based on a **Voting Stage**, a simple hardware circuit made of logic gates, which analyzes the FTS signals of each **Autopilot 1x** in order to determine if terminating the mission or not.

The FTS signals of **Autopilots 1x**, which correspond with the voting signals considered in the Voting Stage, are the following:

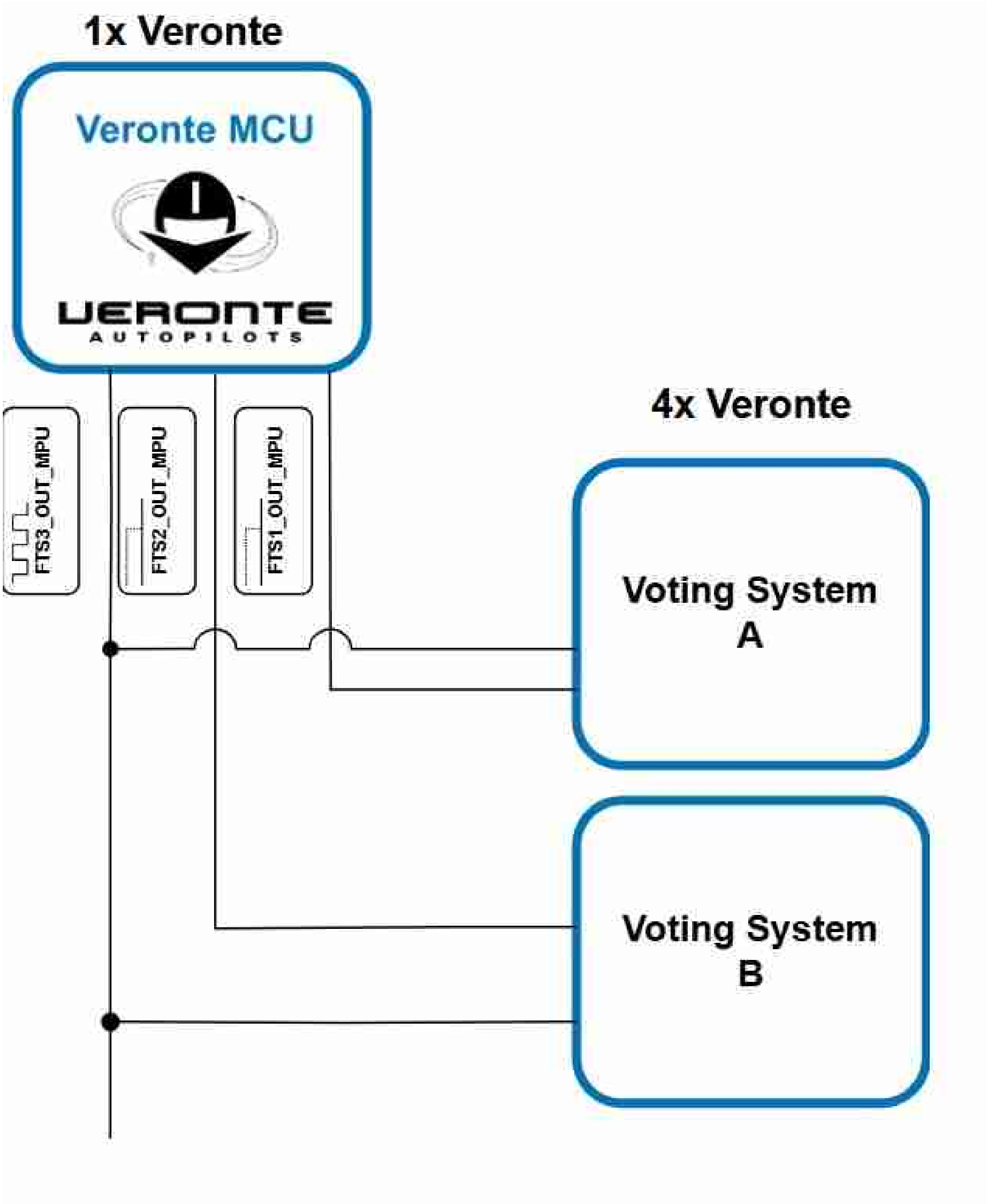


Fig. 10: **Voting System Inputs**

- **Pin 63 - FTS_OUT_MPU:** Its output is 0 V when the system is working as expected and 3.3 V when some error is detected.
- **Pin 64 - FTS2_OUT_MPU:** Its output is 0 V when the system is working as expected and 3.3 V when some

error is detected.

- **Pin 49 - FTS3_OUT_MPU:** MPU alive voting signal. Its output is a square wave at [100,125] Hz.

Note: For further information regarding these pins, please refer to Pinout - Hardware Installation section in **1x Hardware Manual**.

The functionality of the **Voting Stage** is to implement the following logic:

- **Isolate internal Flight Termination Units (FTUs) with failure**. When a deadman signal indicates that an internal Veronte FTU has a failure, this FTU will be excluded from the voting scheme.
- If all **three internal FTUs are OK,** then termination will occur if two of three FTUs detect that the vehicle is out of the restricted area.
- If **two FTUs are ok and one is dead**, then termination will occur if one of the remaining FTUs detect that the vehicle is out of the restricted area.
- If **one FTU is ok and two are dead**, then termination will occur if the remaining FTU detects that the vehicle is out of the restricted area.
- If **all three FTUs are dead**, terminate the mission.

In **Autopilot 4x**, there are **two Voting Systems** available whose output signals are **FTC_VOTING_A** and **FTC_VOTING_B** (pins 53 and 18).

Note: These pins will be **open** in case of **terminating the mission** and **connected to GND** when **continuing the mission**.

4.3.5.2 Arbitration failure

The **Management Board** also includes two FTS pins in the **Arbiter Connector**:

- **Pin 65 - FTS1_OUT:** Deadman signal. Its output is a square wave.
- **Pin 66 - FTS2_OUT:** System OK. Its output will be 3.3 V when an error has been detected and 0V when the arbitration system is working normally.

4.3.6 Electrical diagram of CAN bus

Autopilot 4x requires two termination resistors (120 Ω) to allow multiple CAN Bus devices to be connected to the same line.

Since there is already an internal 120 Ω CAN resistor in the Autopilot 4x (connecting the line to CAN A or CAN B), it is only necessary to place an external 120 Ω resistor at the end of the cable:

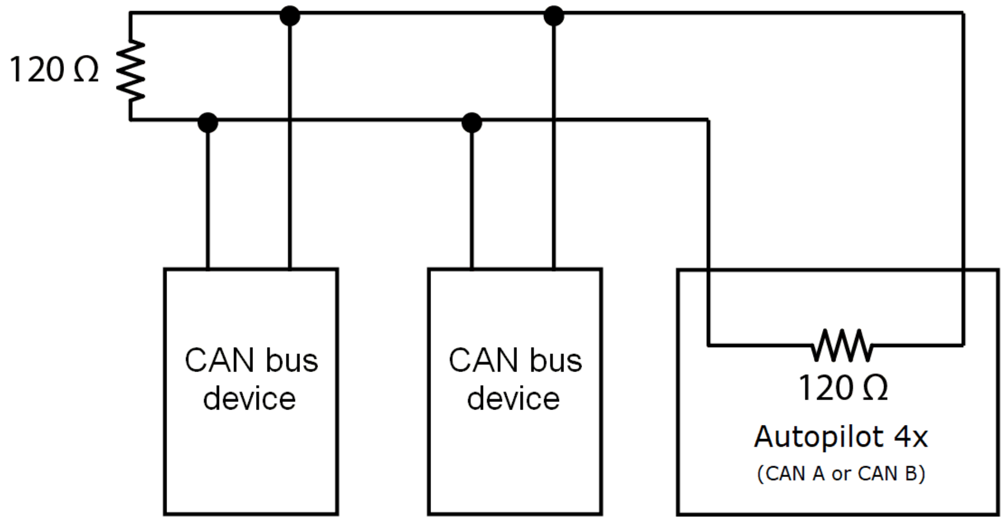


Fig. 11: **CAN resistor**

CHAPTER

FIVE

SOFTWARE INSTALLATION

5.1 Connections to computer

5.1.1 Internal autopilots

Each **Veronte Autopilot 1x** must be connected to a computer individually to configure it. The **Management Board** must also be connected individually.

There are two ways to connect a **Veronte Autopilot 1x** to a computer: USB or serial with RS-232. Both options require power supply for the connected **Autopilot 1x**. Attach the **4x Redundant Harness** to the **Connector 4** to connect a computer and any **Autopilot 1x**.

Note: To use the RS-232 connection with a computer, an USB-RS232 converter is required.

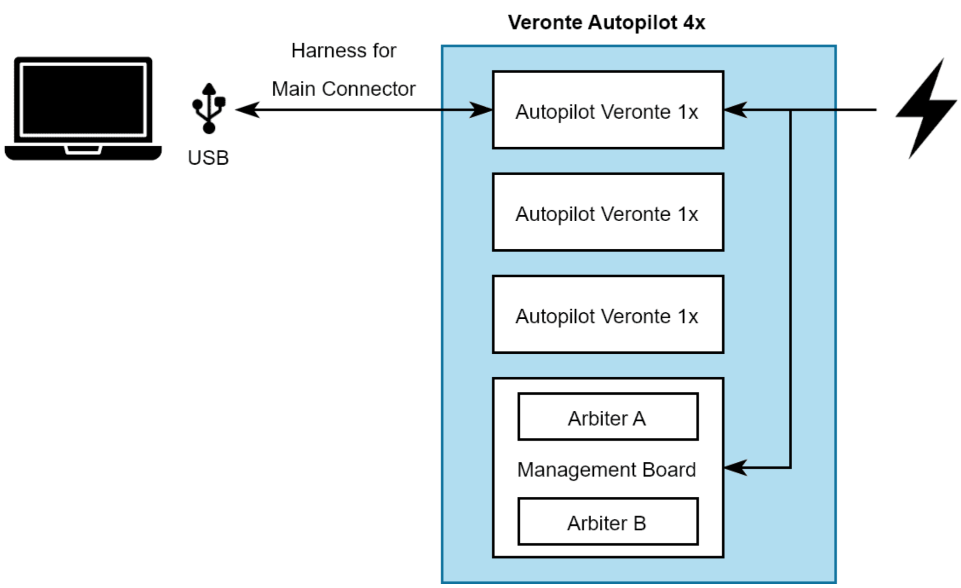


Fig. 1: **USB connection to Autopilot 1x**

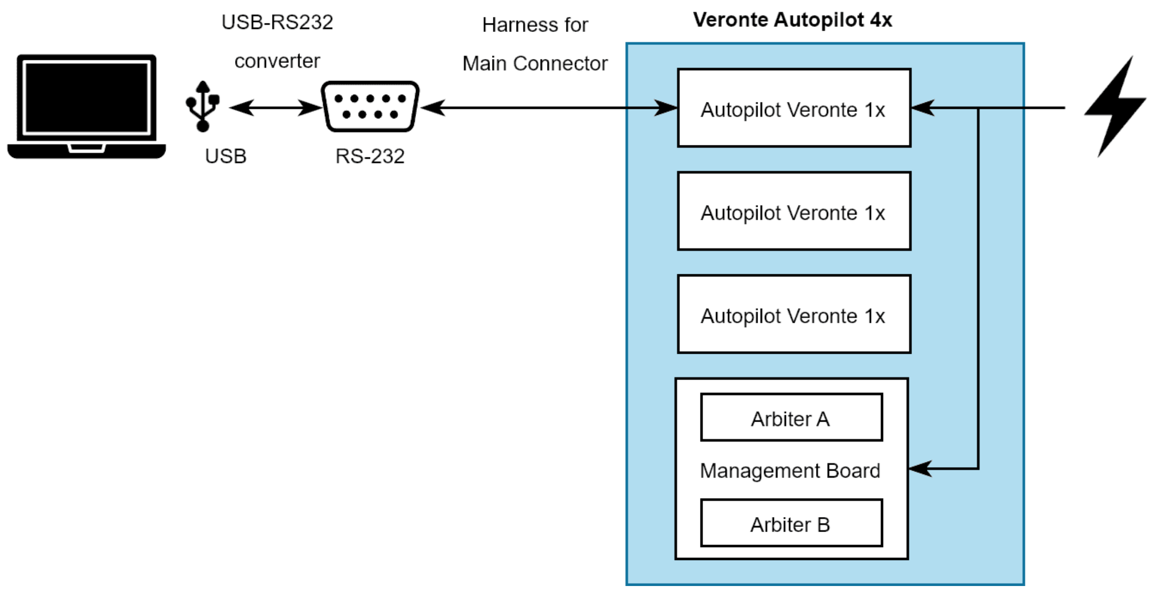


Fig. 2: **Serial connection to Autopilot 1x**

If there is any doubt about electrical connections, check the *Main Connector pinout* and the *Arbiter Connector pinout* in this manual.

5.1.2 Management Board

Autopilot 4x with hardware version 1.8 has a **Management Board**, which is composed by 2 arbiters. Both arbiters have the external communications separated and both require configuration. Hence, **Management Board** requires two differents connections to a computer, each one configuring a different arbiter. Previous hardware versions only had one arbiter, requiring only one connection and configuration.

Arbiter A can be connected to a computer via RS-232 or CAN (through one internal **1x**).

To connect a computer to Arbiter A via RS-232, the connection is as follows:

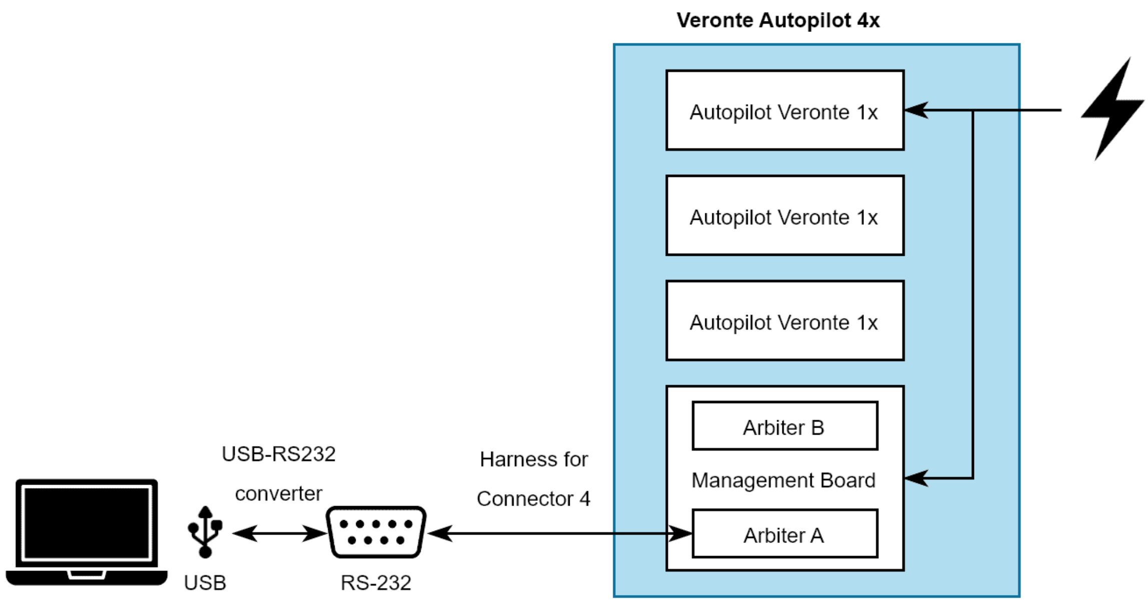


Fig. 3: **Arbiter connection via Serial**

Another way to connect a computer to arbiter A is to use an **Autopilot 1x** as a CAN tunnel, so messages travel through intenal CAN connections. Nonetheless, an **Autopilot 1x** requires a configuration to perform tunnel communications, which is explained in the Arbiters communication - Integration examples section of the **1x PDI Builder** user manual.

Arbiter B does not have RS-232 port, hence the only way to connect B with a computer is through CAN, as explained before.

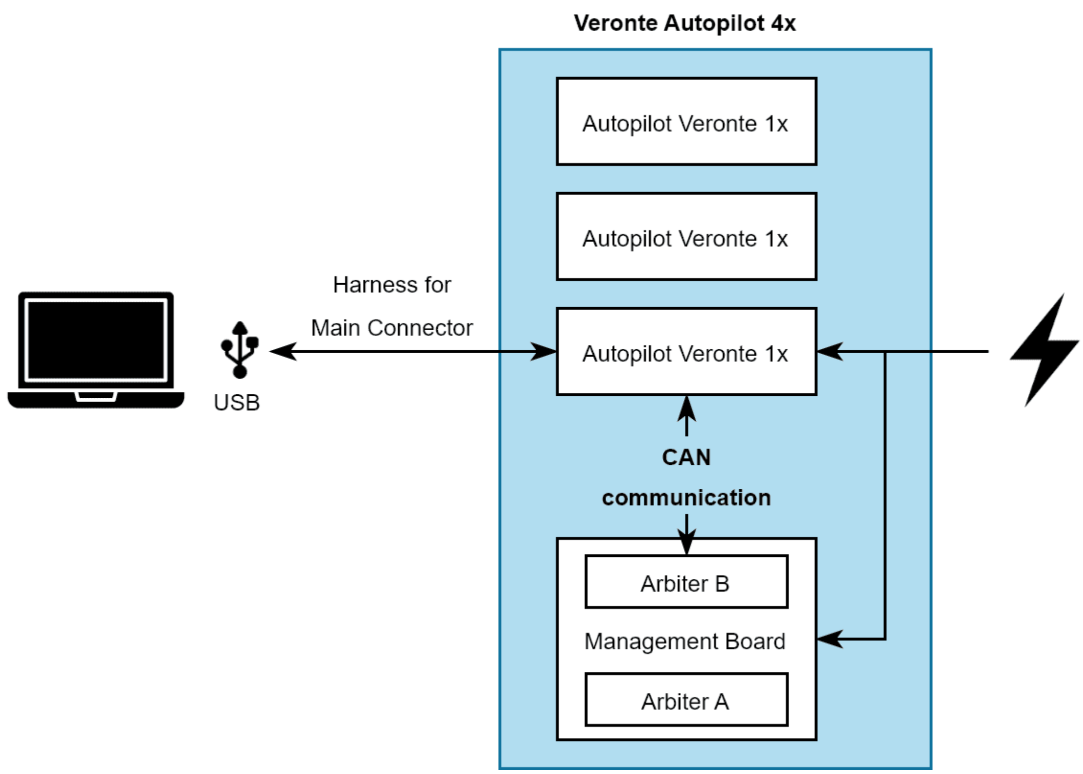


Fig. 4: **Arbiter connection via CAN tunnel**

Important: Each arbiter will be identified by software applications as a different device.

5.2 Software manual

To install the required software and configure each **Veronte Autopilot 1x** and the arbiters, read the 4x Software Manual.

CHAPTER

SIX

OPERATION

6.1 Types of operations

Veronte Autopilot 4x is an advanced system designed to enable the operation of autonomous vehicles, offering three control modes: automatic, assisted, and manual. This versatile autopilot can be used in both uncrewed and manned vehicles, integrating a **FLY-BY-WIRE** system that ensures precise and safe control at all times.

One of the main advantages of the Veronte Autopilot 4x is its configurability, allowing it to be adapted for different operational needs. Depending on the chosen configuration, the system can handle various types of takeoff, such as runway or catapult launches, among others.

Veronte Ops is the Veronte application dedicated to operating the system, providing an intuitive interface for mission management and monitoring. Additionally, for more flexibility, the system can also be operated through VCP (Veronte Communication Protocol), enabling the creation of custom control stations or integration with onboard mission computers for more specific or advanced applications.

In summary, **Autopilot 4x** stands out for its versatility, ease of integration, and customization options, offering a comprehensive solution for a wide range of autonomous vehicle applications.

In addition, for the different types of operations, the user may need to make different connections, configurations and/or integrations with external devices with **Veronte Autopilot 4x**. Therefore:

- Examples of how to integrate **Autopilot 4x** with external devices such as datalinks are detailed in the *Datalinks - Integration examples* section of the present manual.

Please take a look at these sections for further explanations.

This section summarizes a list of possible options to operate an **Autopilot 4x** in different situations.

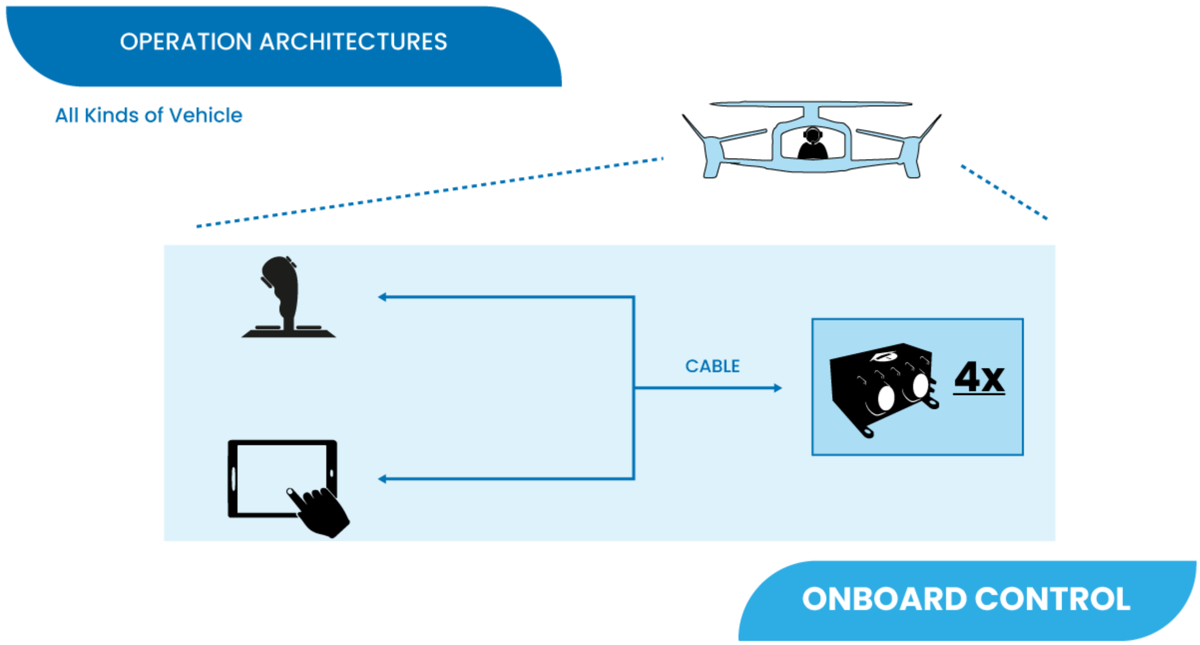
Tip: Most of the following diagrams can be used independently or combinated, to create redundant systems or backup solutions.

6.1.1 Operation Architectures

Veronte Autopilot 4x allows for a wide variety of communication and control solutions to adapt to each mission and platform specifications.

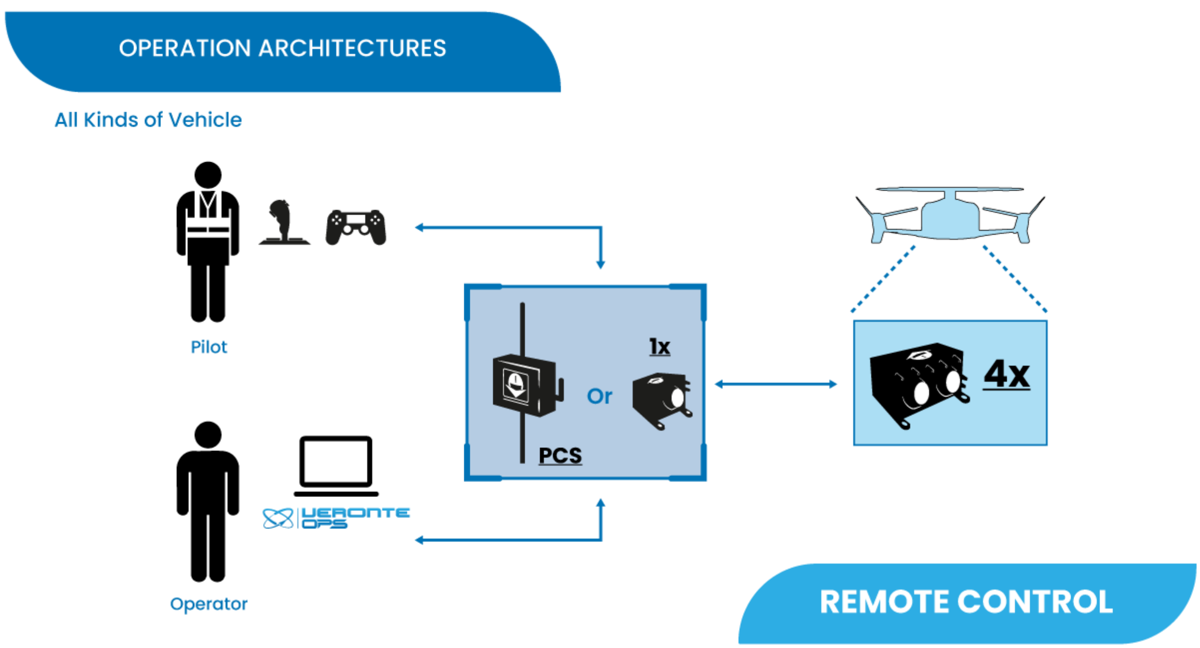
6.1.1.1 Onboard Control Setup

4x allows to control aircrafts (such as eVTOLs) by pilots on board in a flight deck. Pilots can use as controller joysticks, computers, tablets or any device able to communicate through PPM, CAN Bus, RS232 or RS485.



6.1.1.2 Remote Control Setup

The following image shows the standard Veronte System Layout for remote operation.

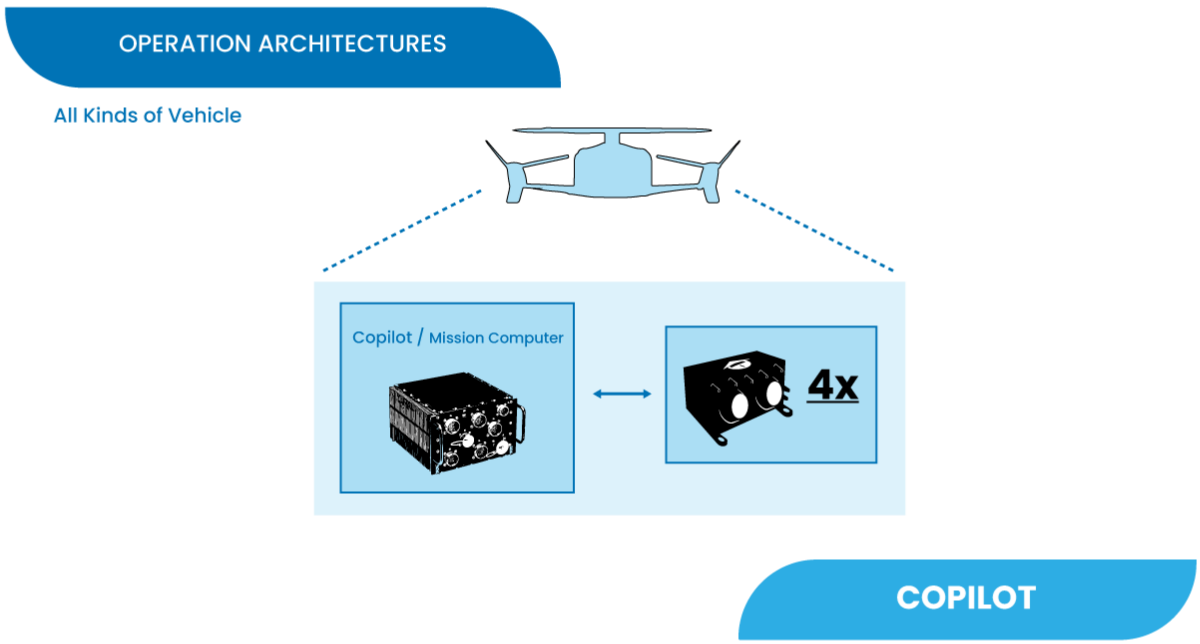


In the standard remote layout, an Operator (Internal Pilot) controls the UAV from the Ground Station using **Veronte Ops**.

Additionally, a Safety Pilot (External Pilot) is connected to the Ground Station using a radio controller. The stick commands are read by the Ground Unit and re-routed to the Air Unit. The Safety Pilot is able to take control of the flight at any point using an automation.

6.1.1.3 Copilot Control Setup

Veronte system allows integration with onboard mission computers for more specific or advanced applications.

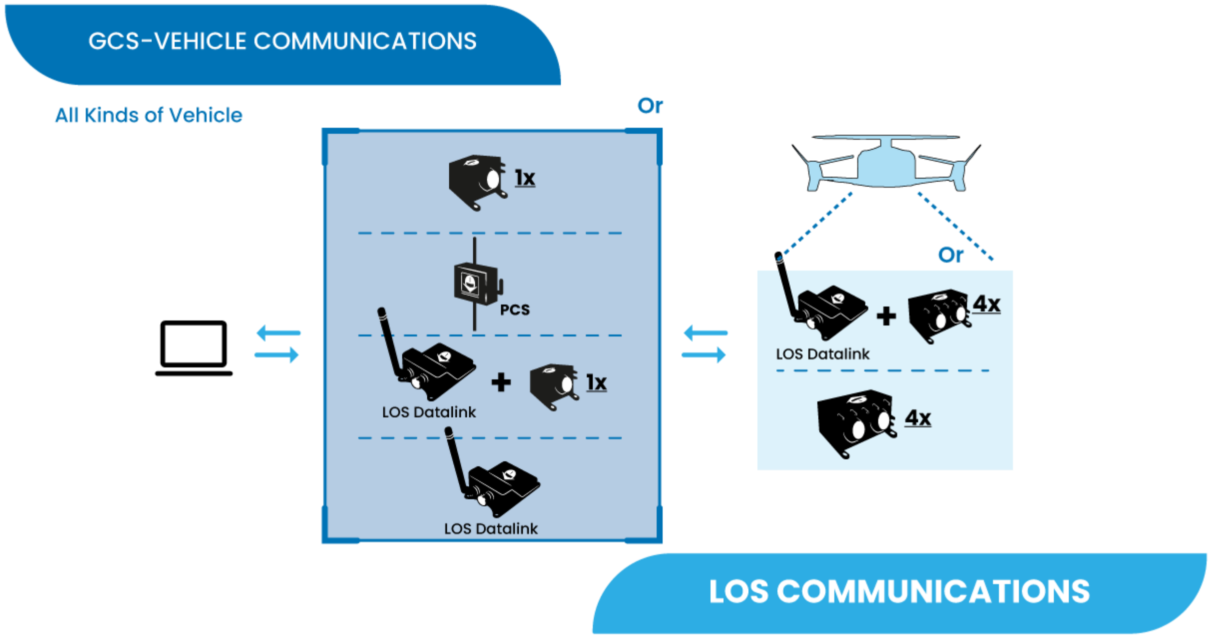


6.1.2 GCS-Vehicle Communications

The following are some examples and possible solutions for establishing communication between the ground control station and the vehicle.

6.1.2.1 LOS Communications

The following diagram shows the different options of GCS and in-vehicle solutions to establish correct Line of Sight (LOS) communications between them.



Depending on the requirements and needs of their mission, users can choose as GCS:

- **Autopilot 1x** with its internal LOS module
- **PCS**
- **Autopilot 1x** with an **external LOS Datalink**
- **LOS Datalink**

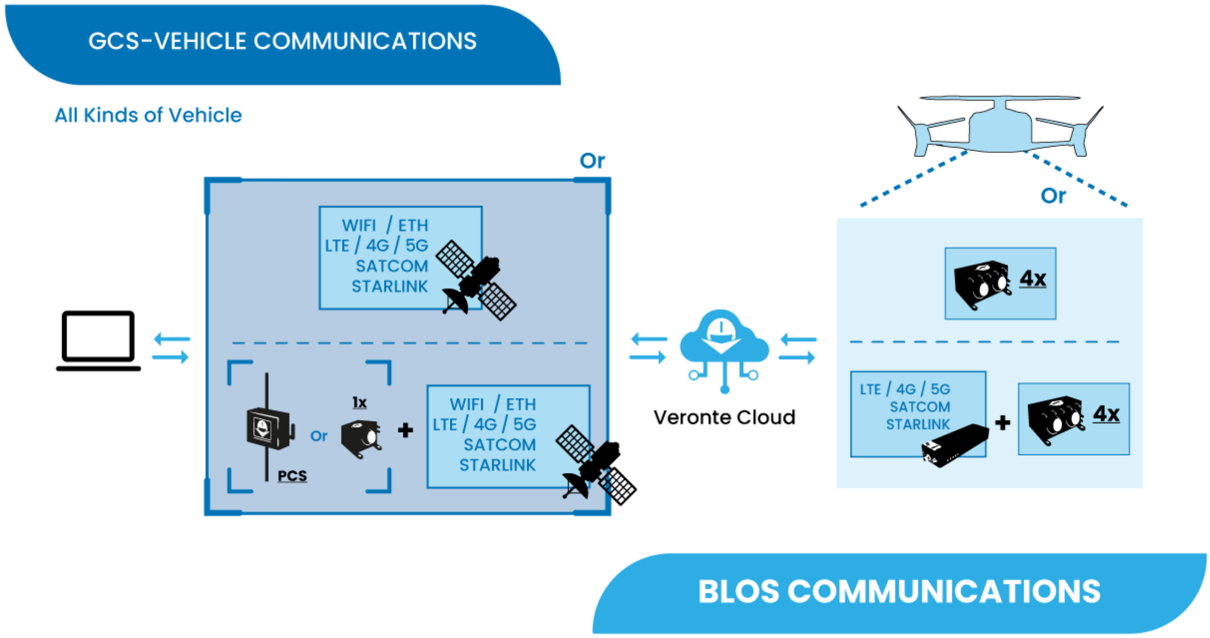
And on the vehicle side:

- **Autopilot 4x** with its internal LOS module
- **Autopilot 4x** with an **external LOS Datalink**

6.1.2.2 BLOS Communications

Veronte Cloud enables secure and efficient Beyond Line of Sight (BLOS) communication between the autopilot onboard a vehicle and the control station. It supports various communication methods, offering a flexible architecture to suit different operational requirements:

- **Autopilot 4x Internal Module**: Embedded **4G** module within **Autopilot 4x**.
- **LTE/4G/5G Module**: External LTE module for wireless communication.
- **Satcom Module**: Satellite communications device for global coverage.
- **Starlink**: High-bandwidth, global communications module.



These communication methods can be used both at the **ground control station** and **onboard**, enabling seamless switching between methods or simultaneous use for redundancy and enhanced reliability. They can also be combined to meet specific project requirements.

Control Station Connectivity Options

The control station connects to **Veronte Cloud** through two primary methods:

- **Option A: Direct PC Internet Connection**

  The **control station PC** connects directly to the Internet for communication with **Veronte Cloud**. This can be achieved using any available means of Internet communication:

  - Ethernet or Wi-Fi
  - LTE/4G/5G
  - Satellite Communication (Satcom)
  - Starlink

- **Option B: Connection via Veronte PCS/4x**

  The **control station PC** connects to the **Veronte PCS/1x module**, which manages the connection to the BLOS datalink module. The **PCS/1x** module supports:

  - Its **internal 4G module** for direct connectivity.
  - **External communication modules** (LTE/4G/5G, Satcom, Starlink, etc.).

  This setup enhances communication reliability by leveraging Veronte's dedicated hardware for connection management and enabling the use of additional sensors integrated within the **PCS/1x** module.

Onboard Connectivity Options

For onboard BLOS communications, **Veronte Autopilot 4x** system offers two main methods:

- **Option C: Internal 4G Module in Veronte Autopilot 4x**

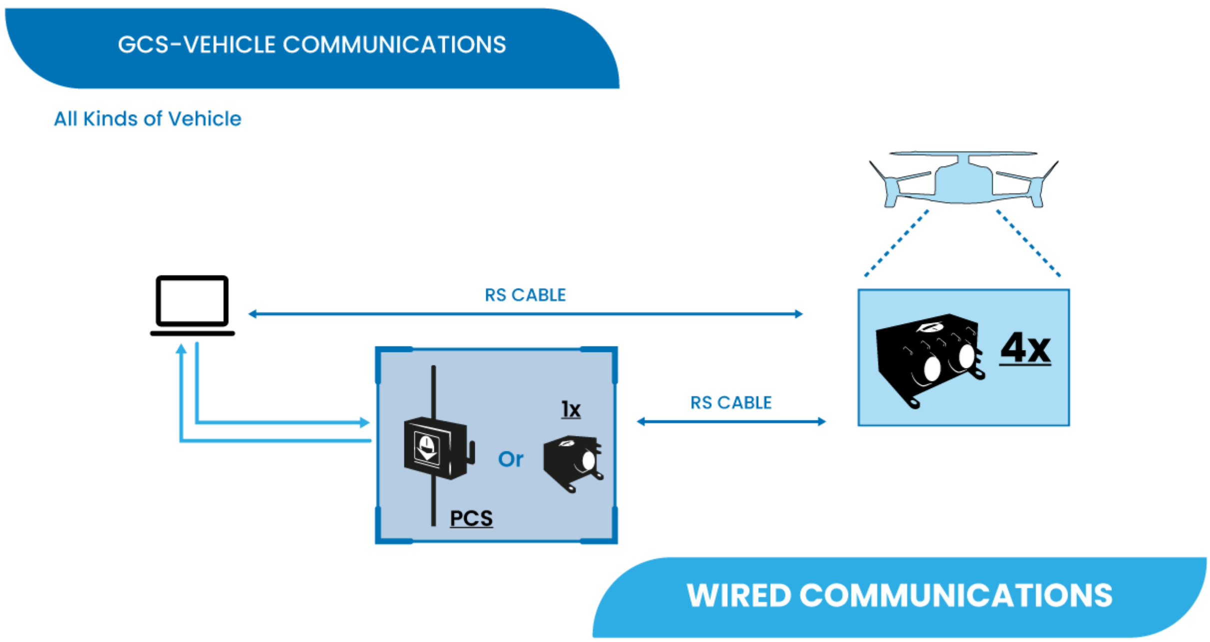
Autopilot 4x comes equipped with an internal 4G module that connects directly to **Veronte Cloud**. This option is compact and does not require additional external hardware.

- **Option D: External Module Connected to Autopilot 4x**

  The autopilot can integrate an **external communication module** (LTE/4G/5G, Satcom, Starlink, etc.) to enable BLOS communication with **Veronte Cloud**. This provides flexibility and allows for customization based on specific mission or environmental needs.

6.1.2.3 Wired Communications

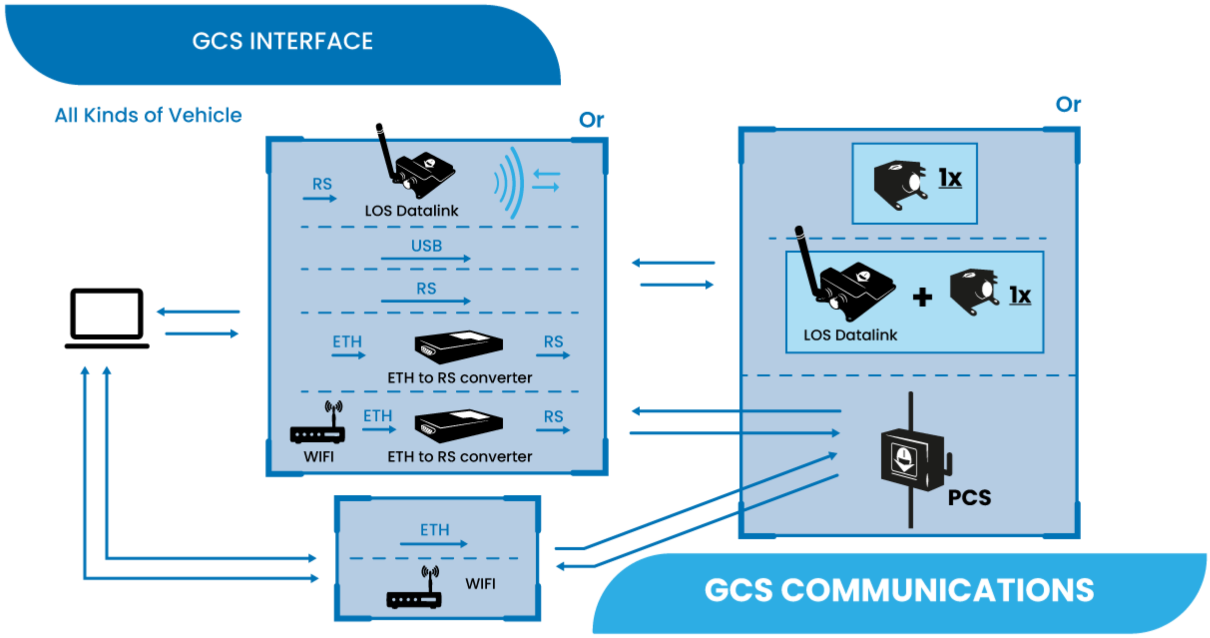
For operations where the control station is directly connected to the onboard autopilot by cable.



6.1.3 GCS Interface

This diagram represents some of the many ways to establish communication between the different parts of a Ground Control Station setup.

Note: In a Ground Control Station setup there is usually a PC on one side and an **Autopilot 1x** with its internal LOS module, an **Autopilot 1x** with an **external LOS Datalink** or a **PCS** on the other side.



Direct connection

- The **PC** can directly connect a **PCS** via **USB**, **RS**, **Ethernet** or **Wifi**.
- The **PC** can directly connect an **Autopilot 1x** with its internal LOS module or **Autopilot 1x** with an **external LOS Datalink** via **USB/RS**.

Combined connections

Below are different connection methods that enable communication between the **PC** and an **Autopilot 1x** with its internal LOS module, an **Autopilot 1x** with an **external LOS Datalink** or a **PCS** via an additional device:

- PC connected via RS to a **LOS Datalink**, establishing a datalink connection to the other side of the GCS setup.
- Connection through an **Ethernet-to-RS converter**, i.e. Ethernet on the PC side and RS on the other side.
- The PC connects via **wifi** and the wifi modem then communicates with the other side of the GCS setup through an **Ethernet-to-RS converter**.

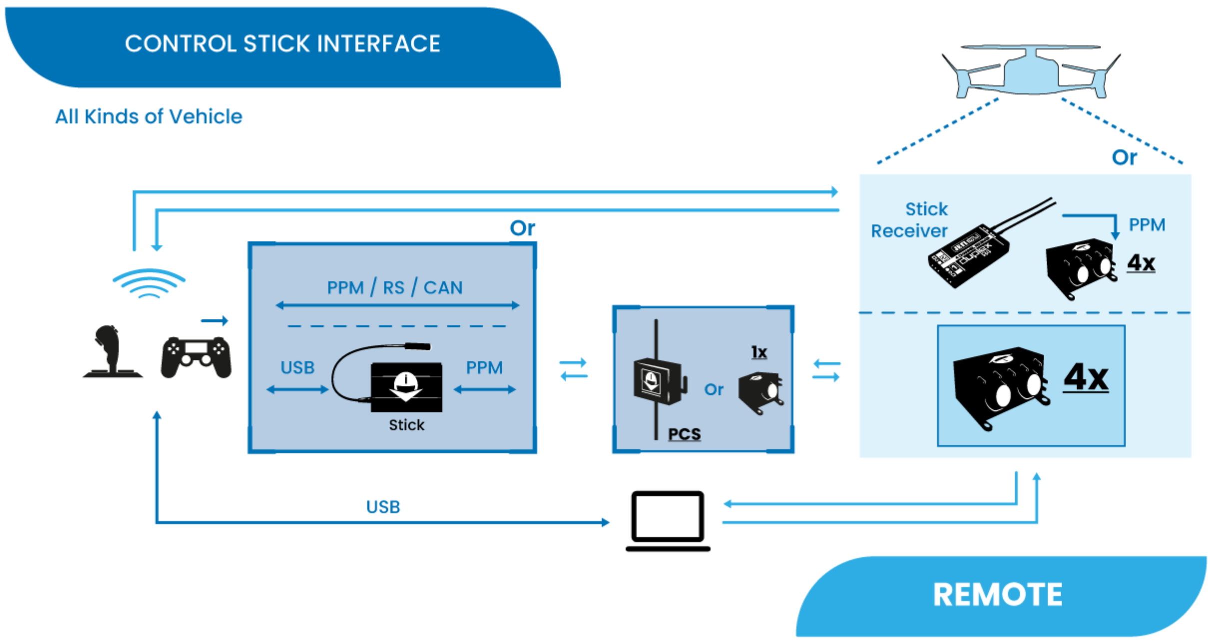
6.1.4 Control Stick Interface

This section presents the different types of manual control from stick to the onboard autopilot.

Veronte allows for a wide variety of pilot interface solutions in order to interact with manual flight modes, assisted flight modes (arcade) or payloads.

6.1.4.1 Remote Control Stick

A wide variety of controllers can be used to pilot manually aircrafts, such as RC transmitters, pedals, sticks or buttons. Veronte software allows the use of any device that is detected as a remote controller by the operative system



Although the most common way of control is to directly connect a **stick** via **PPM**, **RS** or **CAN** to a control station (**PCS/1x**) which then communicates with the onboard autopilot,

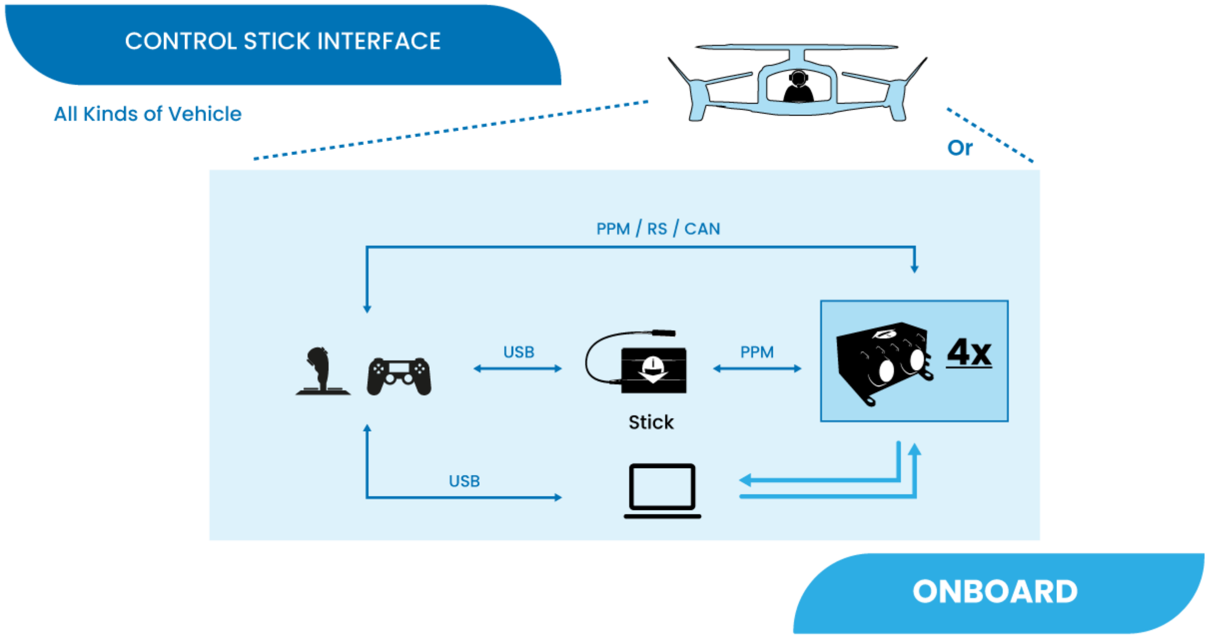
- It is possible to establish a link connection between a stick and a **stick receiver** integrated in the vehicle, which is connected via **PPM** to **Autopilot 4x**.

  This allows for a backup manual channel when there is a main channel loss and an emergency manual landing is needed. Recommended for initial developement stages where automatic landing phases are not defined yet.

- A **Veronte Stick** allows the connection of **USB sticks** to a control station (**PCS/1x**), converting **USB to PPM**. Then, the GCS communicates with the onboard autopilot for control.
- A **USB stick** can be connected directly to the **PC** to establish communication with the onboard autopilot for control.

6.1.4.2 Onboard Control Stick

In operations with pilots onboard in a flight deck (such as eVTOLs), the sticks can directly control the vehicle's **Autopilot 4x**.



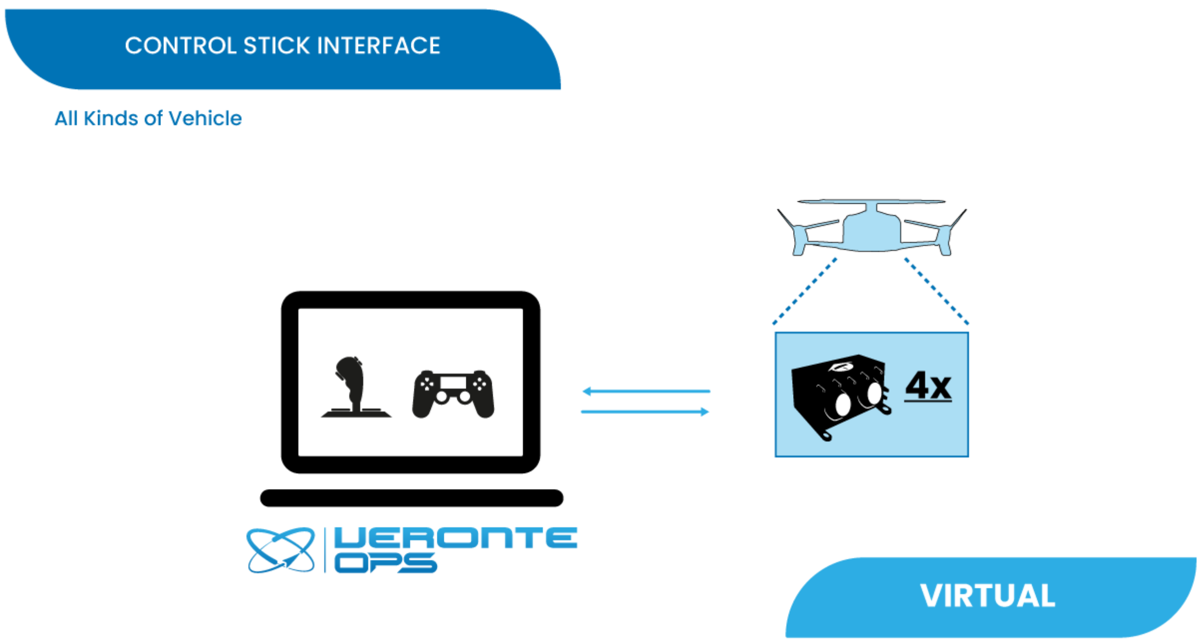
Some examples are:

- To directly connect a stick via **PPM**, **RS** or **CAN** to the autopilot.
- To use a **Veronte Stick** that converts **USB to PPM**, allowing connection between **USB sticks** and the autopilot.
- Connect a **USB stick** to a **PC** which establishes communication with the autopilot.

6.1.4.3 Virtual Stick

The Virtual stick feature allows to integrate as a stick controller any device that can interface with **Autopilot 4x** (RS232, RS485, ADC, CAN…) and can provide control reference values.

While the configuration is slightly more complex, this feature allows using a wide variety of devices as flight control interfaces.



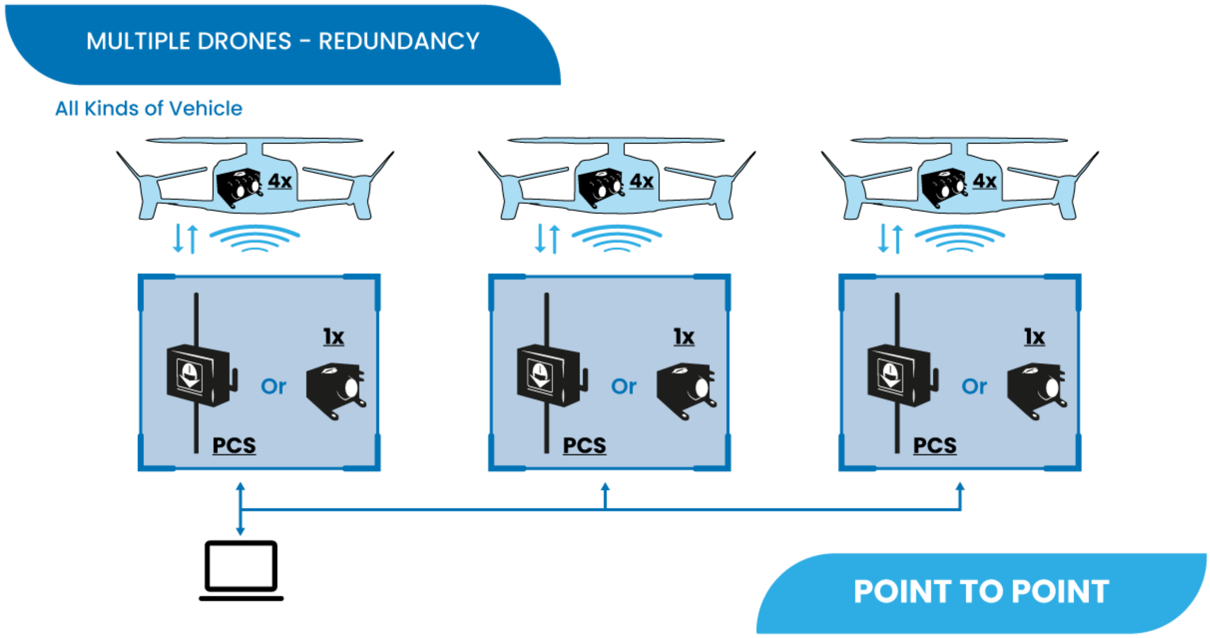
6.1.5 Multiple Drones/GCS - Redundancy

Due to Veronte's modular configuration, it is possible to integrate several onboard and ground units within the same network.

Note: Users are free to combine the different multiple drones solutions with the multiple GCS solutions.

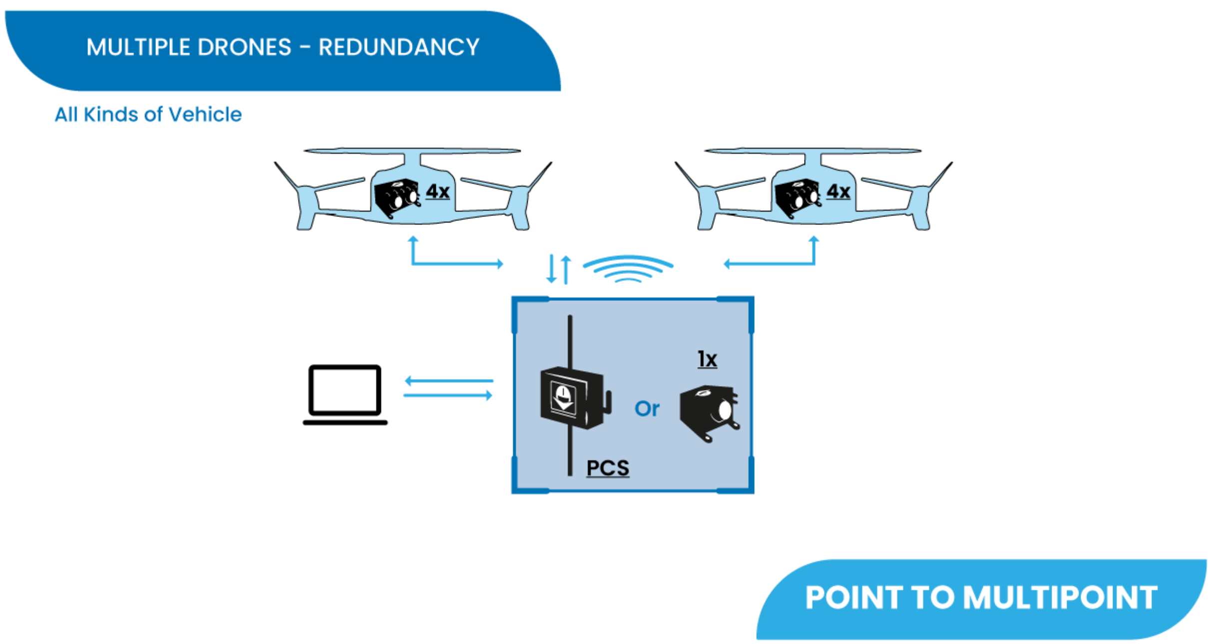
6.1.5.1 Multiple Drones - Point to Point

Standard multiplatorm setup.



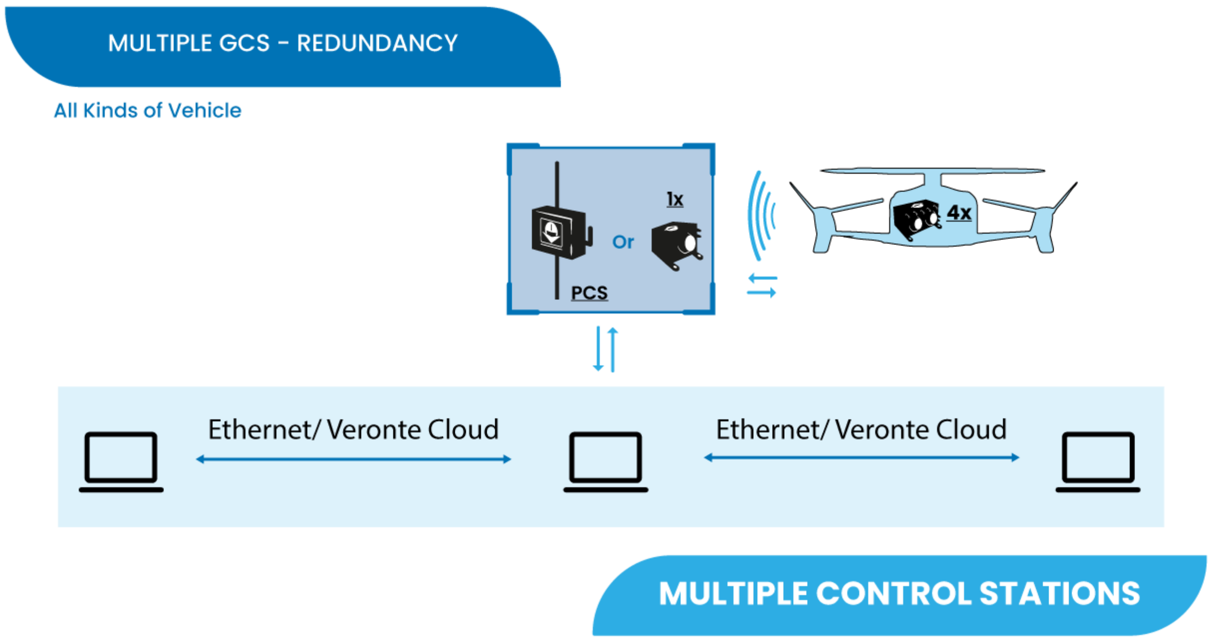
6.1.5.2 Multiple Drones - Point to Multipoint

Managing several platforms with a single radiolink.



6.1.5.3 Multiple GCS

For long range operations with several LOS stations.



For remote solutions with LOS backup operator, **Veronte Cloud** allows the connection between PCs.

To correctly establish communication between the different PCs via **Ethernet** the following steps should be carried out:

1. Make the **physical connection** with ethernet cables, the two different types of ethernet cables can be used:

   - **Straight-Throught**

     Connect each PC to an **ethernet switch** with its Straight-Through ethernet cable (i.e. users will need 2 cables).

   - **Crossover**

     Connect the PCs directly to each other with a crossover ethernet cable.

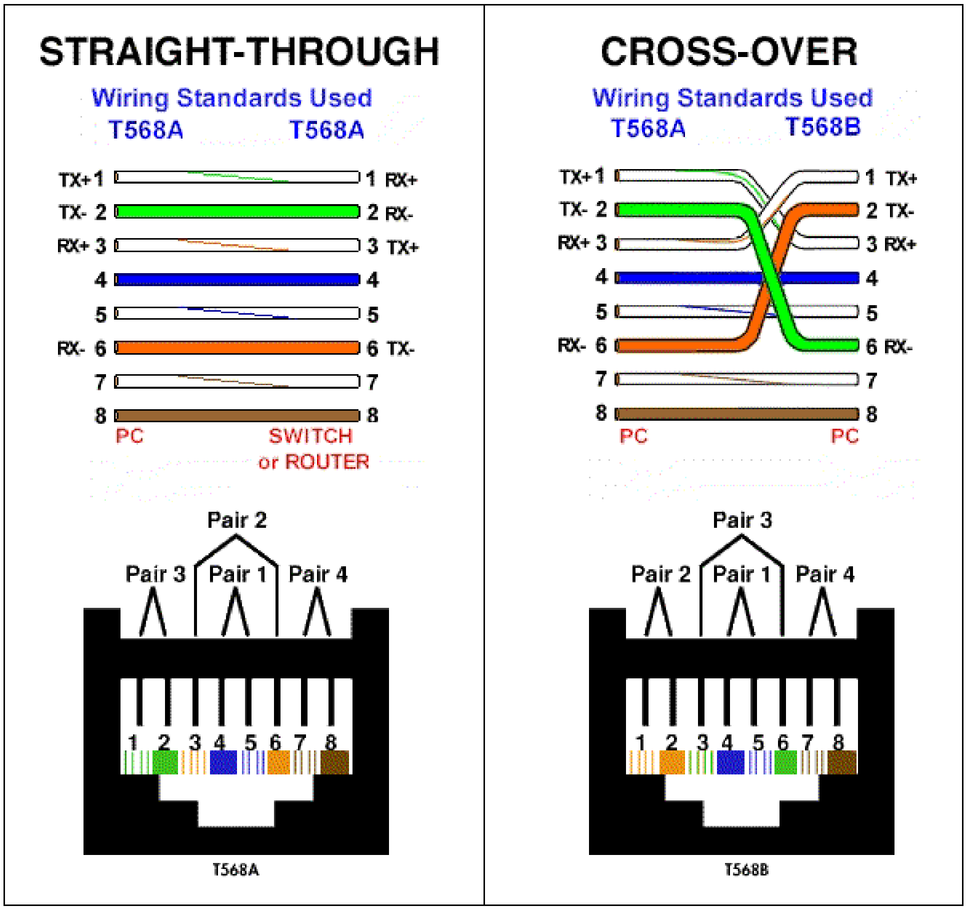


Fig. 1: **Straight-Through vs Crossover cables**

2. On each PC, change the ethernet adapter settings to a static IP so that both are on the same subnet. To do this:
   - In the **Control Panel**, go to **Network and Internet**.
   - Open **Network and Sharing Centre** menu and click **Change adapter settings**.

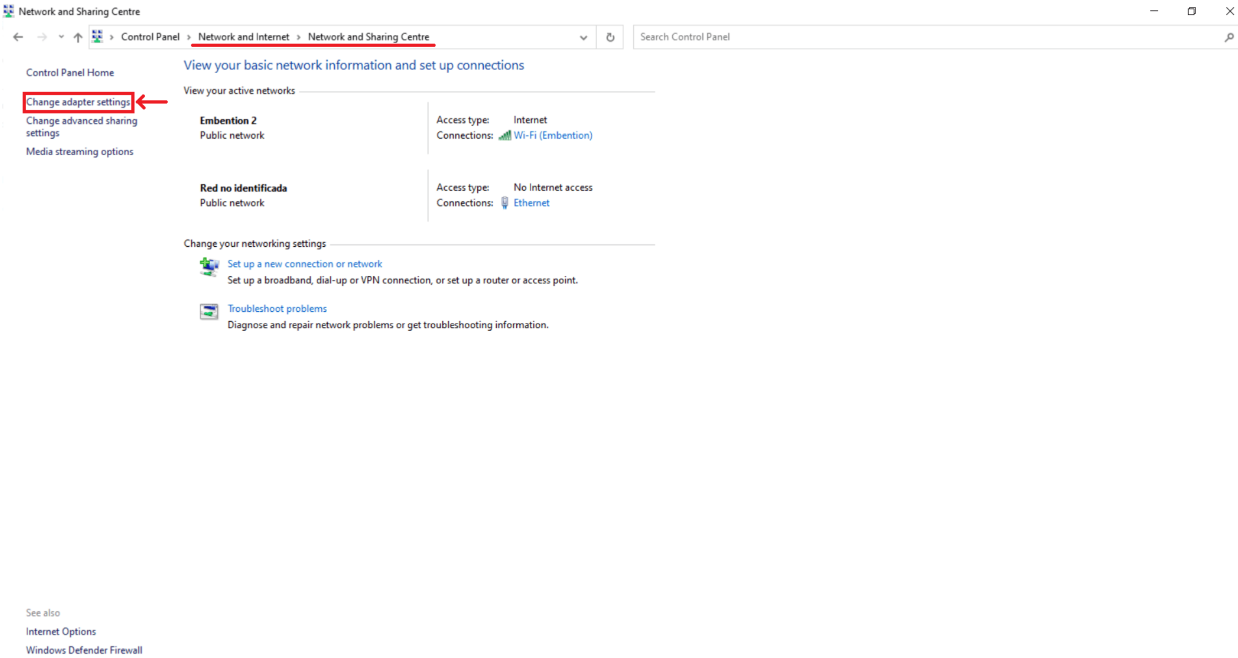


Fig. 2: **Ethernet connection 1**

- Select **Local Area Connection**, right click, and select **Properties**.

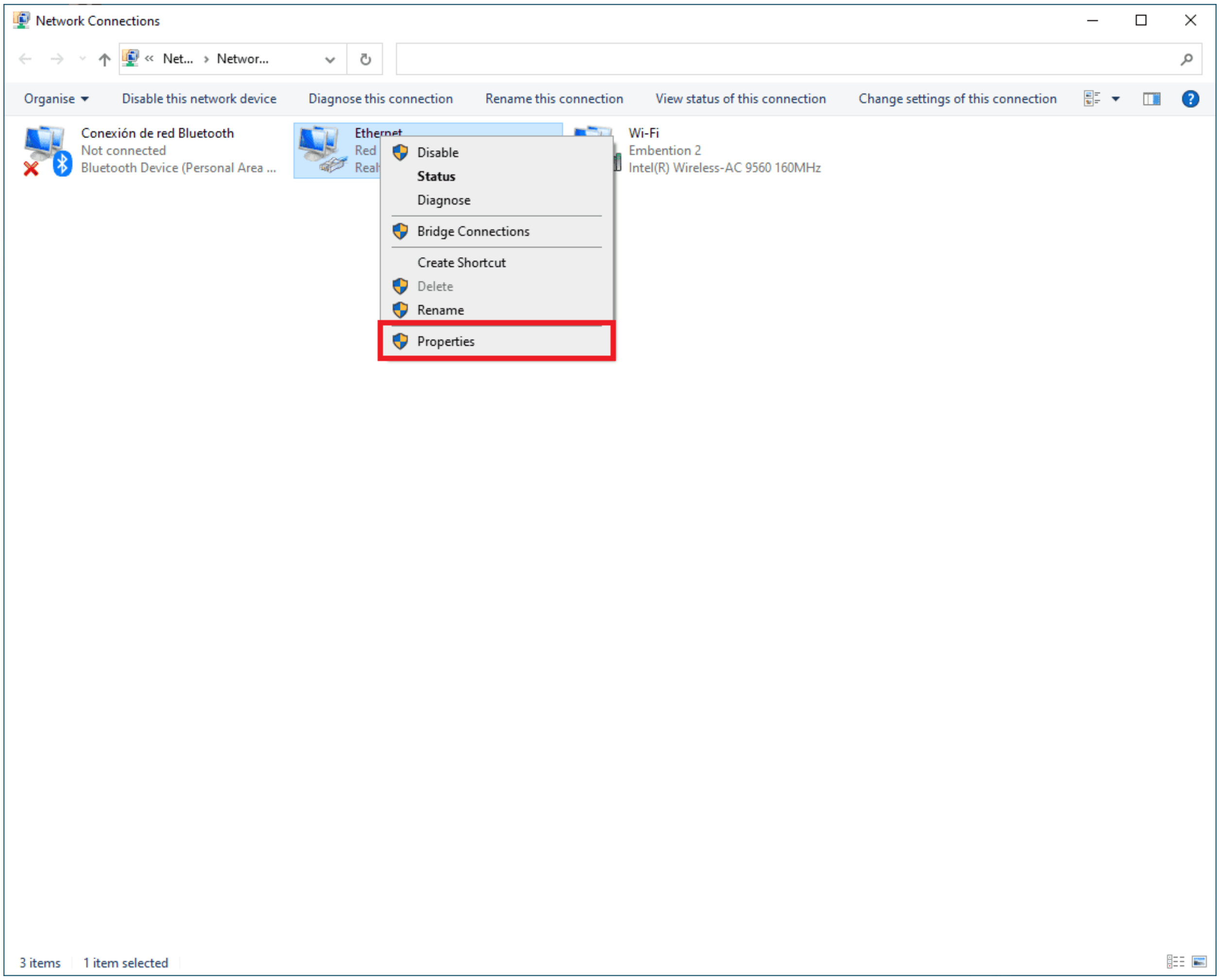


Fig. 3: **Ethernet connection 2**

- Select **IPv4** and click **Properties**.

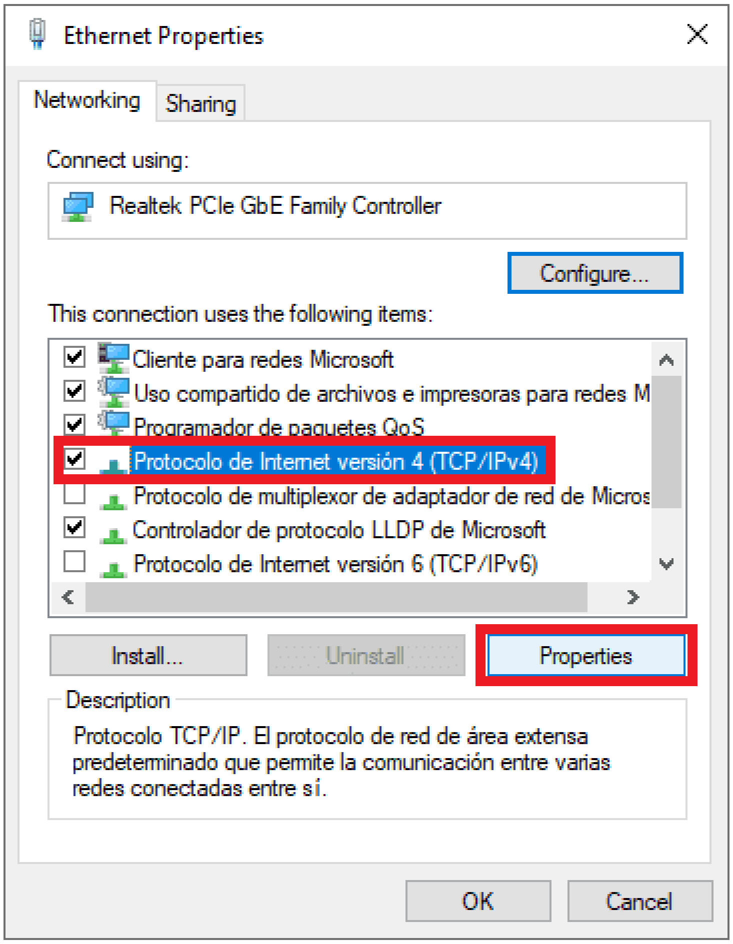


Fig. 4: **Ethernet connection 3**

- Set **IP address** to a **static IP** (e.g. 192.168.0.100) and **Subnet mask** to 255.255.255.0. Click **OK**.

Important: If on this PC the IP address is set to 192.168.0.100, on the other PCs, the IP address must be set to **192.168.0.XXX** (e.g. *192.168.8.234*), so that they are on the same subnet.

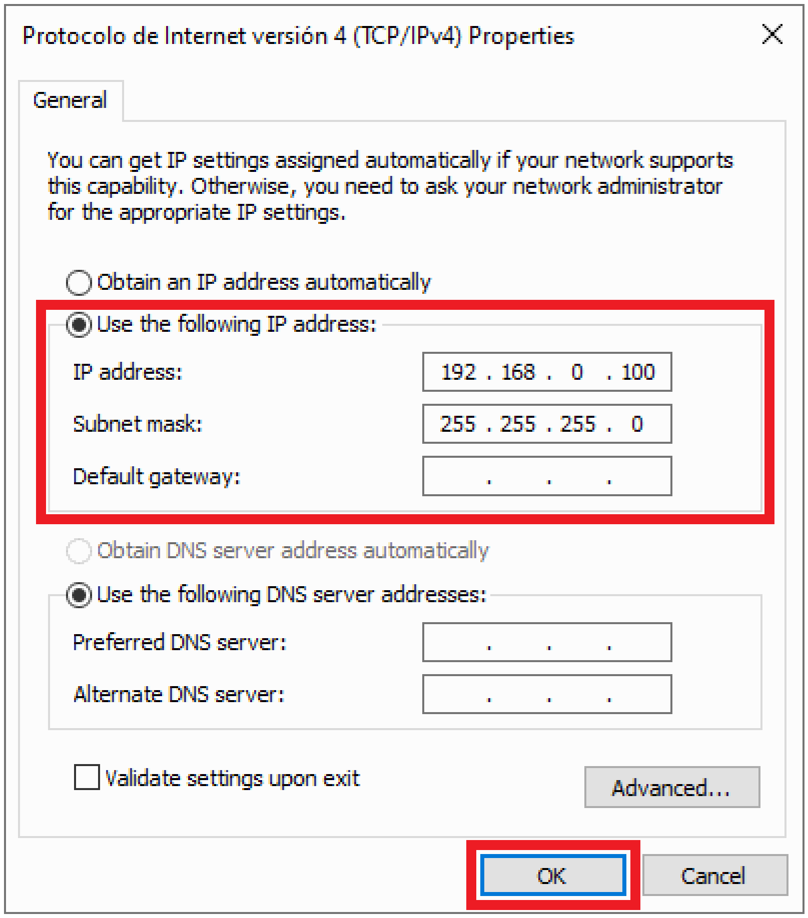


Fig. 5: **Ethernet connection 4**

3. Allow **VeronteLink** to go through the Firewall on the PC that will run it, hereafter *PC primary*. To do so:
   - In the **Control Panel**, go to **System and Security**.

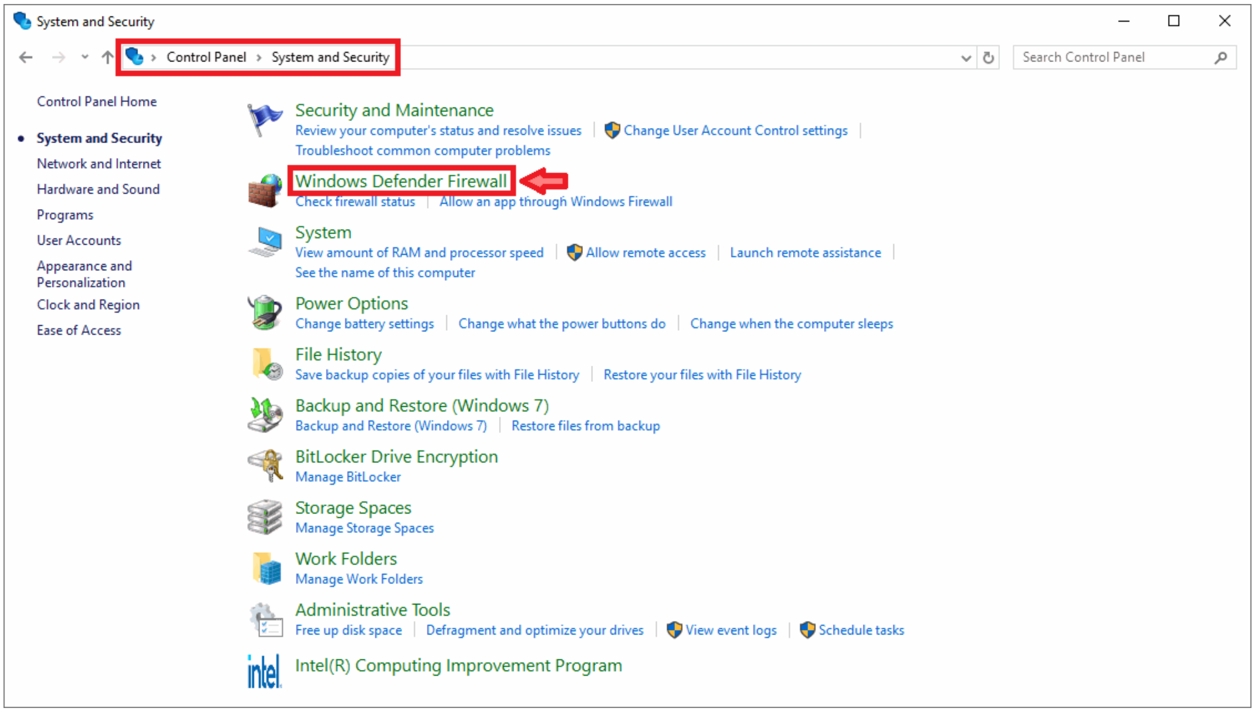


Fig. 6: **Windows Firewall 1**

- Open **Windows Defender Firewall** and click on **Allow an app through Windows Defender Firewall**.

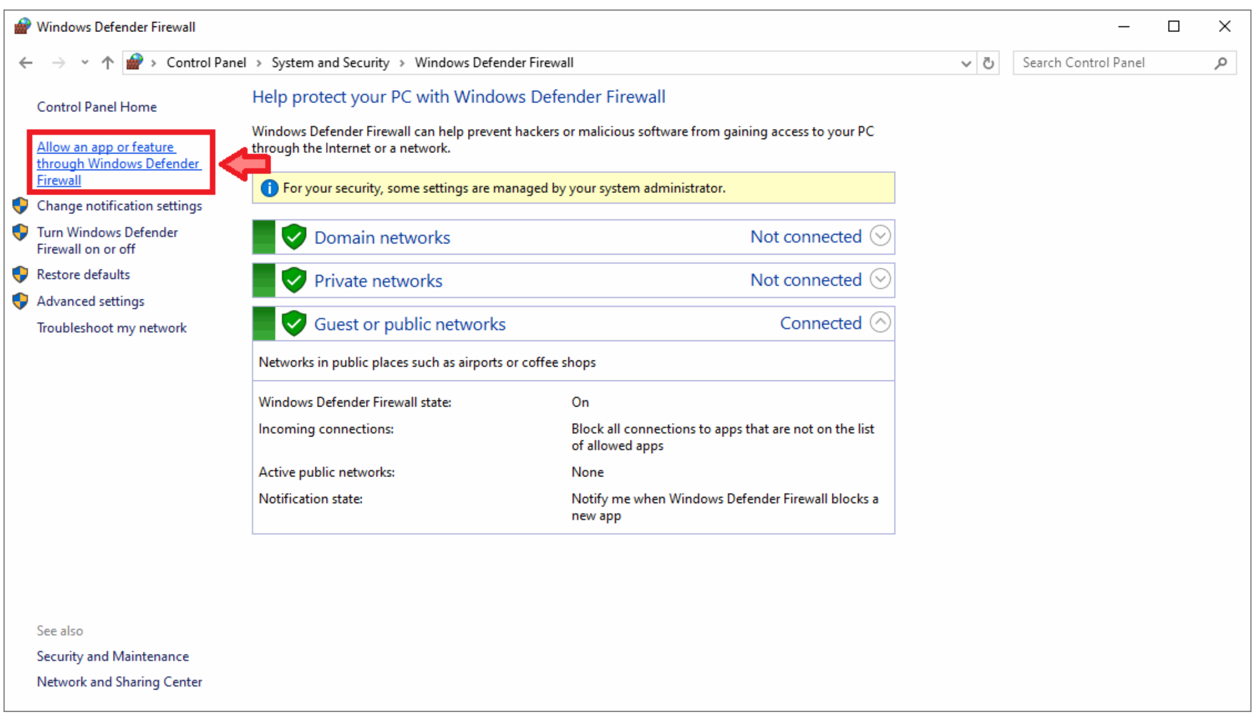


Fig. 7: **Windows Firewall 2**

- Check that **Veronte Link** app is **allowed**.

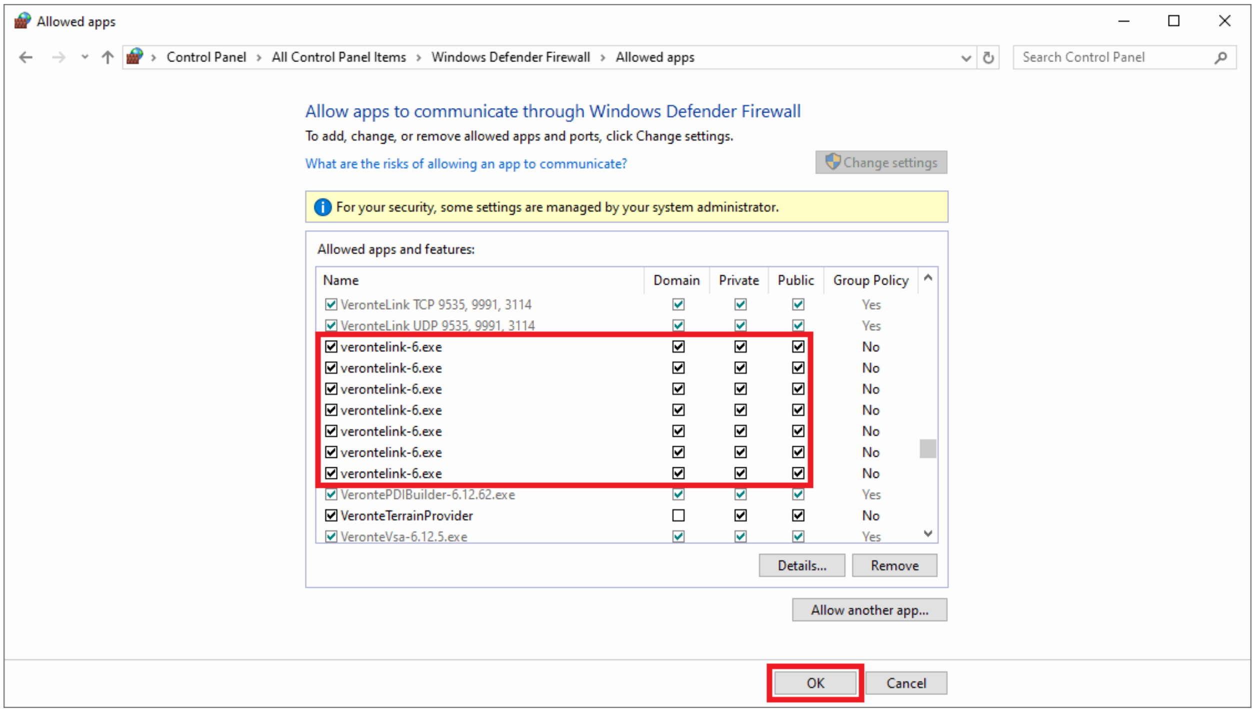


Fig. 8: **Windows Firewall 3**

4. On the *PC secondary*, in **Veronte Ops** change the **Veronte Link Host** option setting to the **IP of the PC primary**. To do this:
   - Open **Veronte Ops**.
   - In the **Status bar**, click the **arrow** on the right of the bar to display a **drop-down menu**.

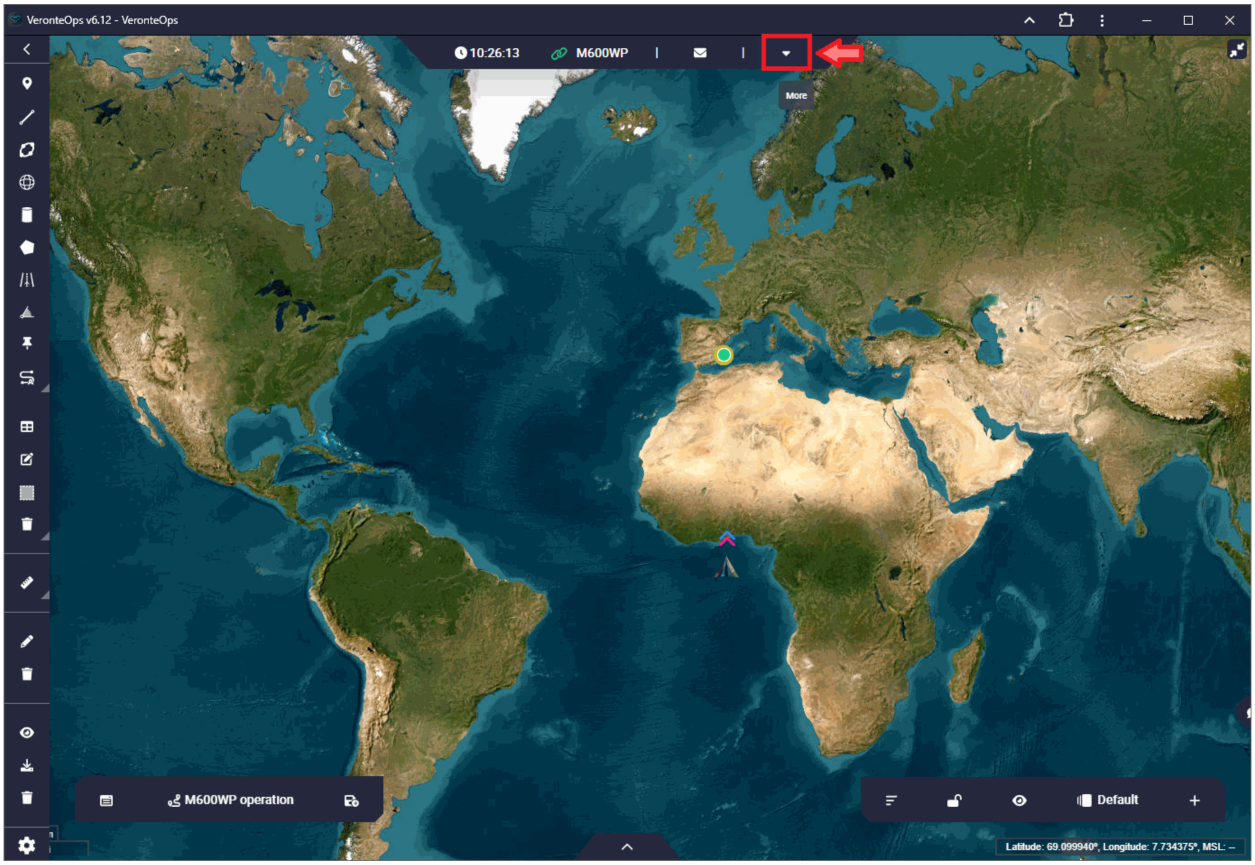


Fig. 9: **Veronte Ops - Status bar**

- In it, open the **Setup** menu.

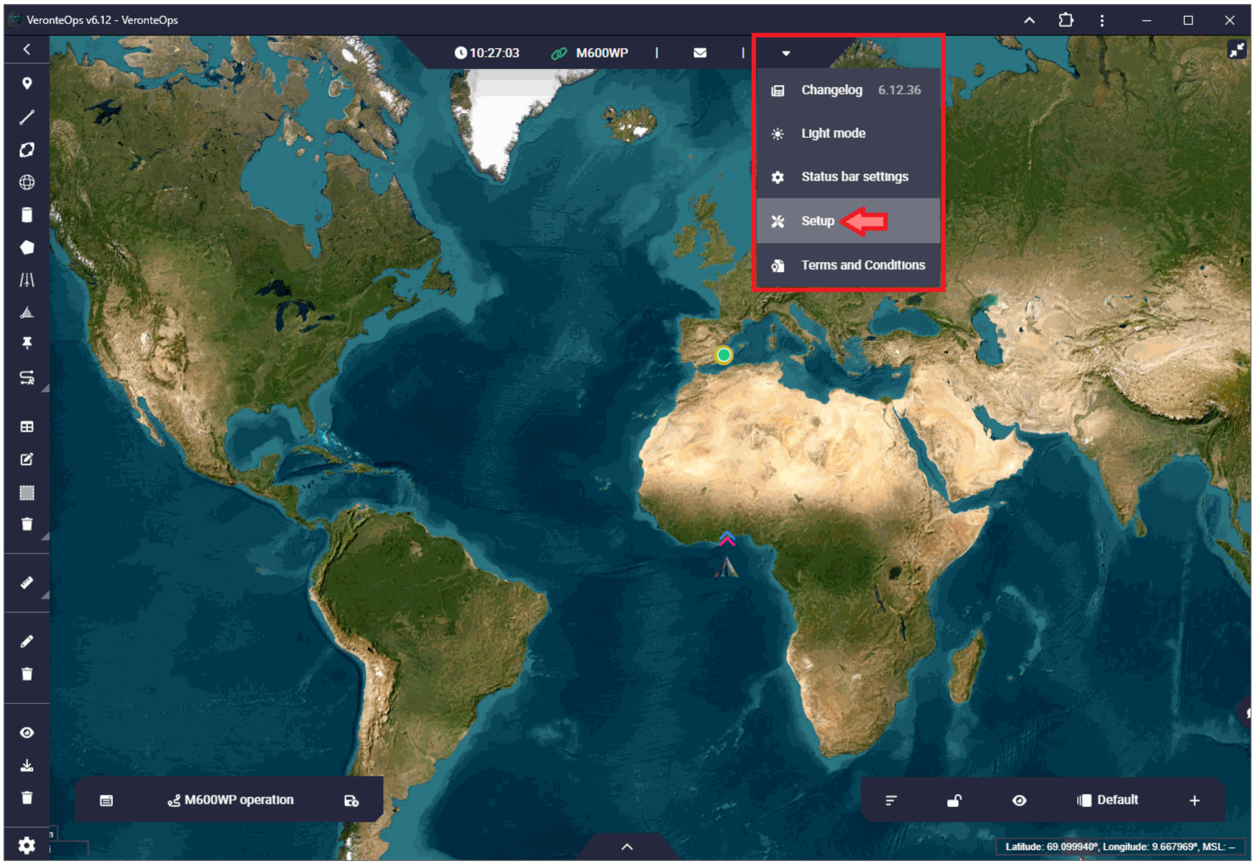


Fig. 10: **Veronte Ops - Setup menu**

- Next, go to the **Veronte Link Host** settings.
- Change the IP *localhost* to the IP of the *PC primary*.

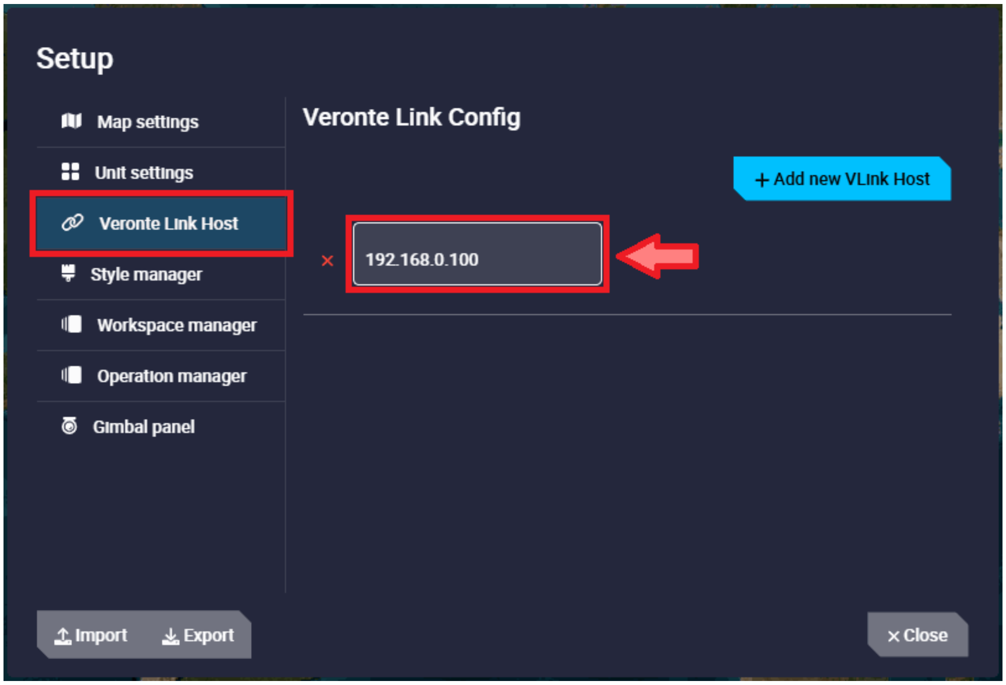


Fig. 11: **Veronte Ops - Veronte Link Host settings**

For more information on this settings, refer to the Setup - Veronte Ops configuration section of the **Veronte Ops** user manual

5. Finally, **Autopilot 1x** connected to the *PC primary* should be seen in the **Veronte Ops** open on this PC, as well as on the *PC secondary*.

   If users have any problems when trying to connect **Veronte Ops** to **Veronte Link**, refer to the Connecting to Veronte Link - Troubleshooting section of the **Veronte Ops** user manual.

If after following the steps described above users are not able to operate in this way, please contact support team by opening a **Ticket** in your Joint Collaboration Framework.

CHAPTER

SEVEN

MAINTENANCE

7.1 Preventive maintenance

Apart from cleaning, no extra maintenance is required to guarantee the correct operation of **Veronte Autopilot 4x**.

In order to clean **Veronte Autopilot 4x** properly follow the next recommendations.

- Turn off the device before cleaning.
- Use a clean, soft, damp cloth to clean the unit.
- Do not immerse the unit in water to clean it.

7.2 Software update

To update the software, an additional app is required: Veronte Updater.

Note: The file with the new software version will be shared with the customer in the **Joint Collaboration Framework** when it is requested.

For more information about the **Joint Collaboration Framework**, read its user manual.

CHAPTER

EIGHT

COMPATIBLE DEVICES

Since **Veronte Autopilot 4x** is based on **Veronte Autopilot 1x**, both products have the same compatible devices. To know more, read the Compatibles Devices section of the **1x Hardware Manual**.

CHAPTER

NINE

INTEGRATION EXAMPLES

The step-by-step instructions for the following external devices will be explained in detail in the following sections:

- *Connection Examples*
- *Air Data Sensors*
- *Datalinks*
- *Stick*

9.1 Connection Examples

9.1.1 Ground Stations

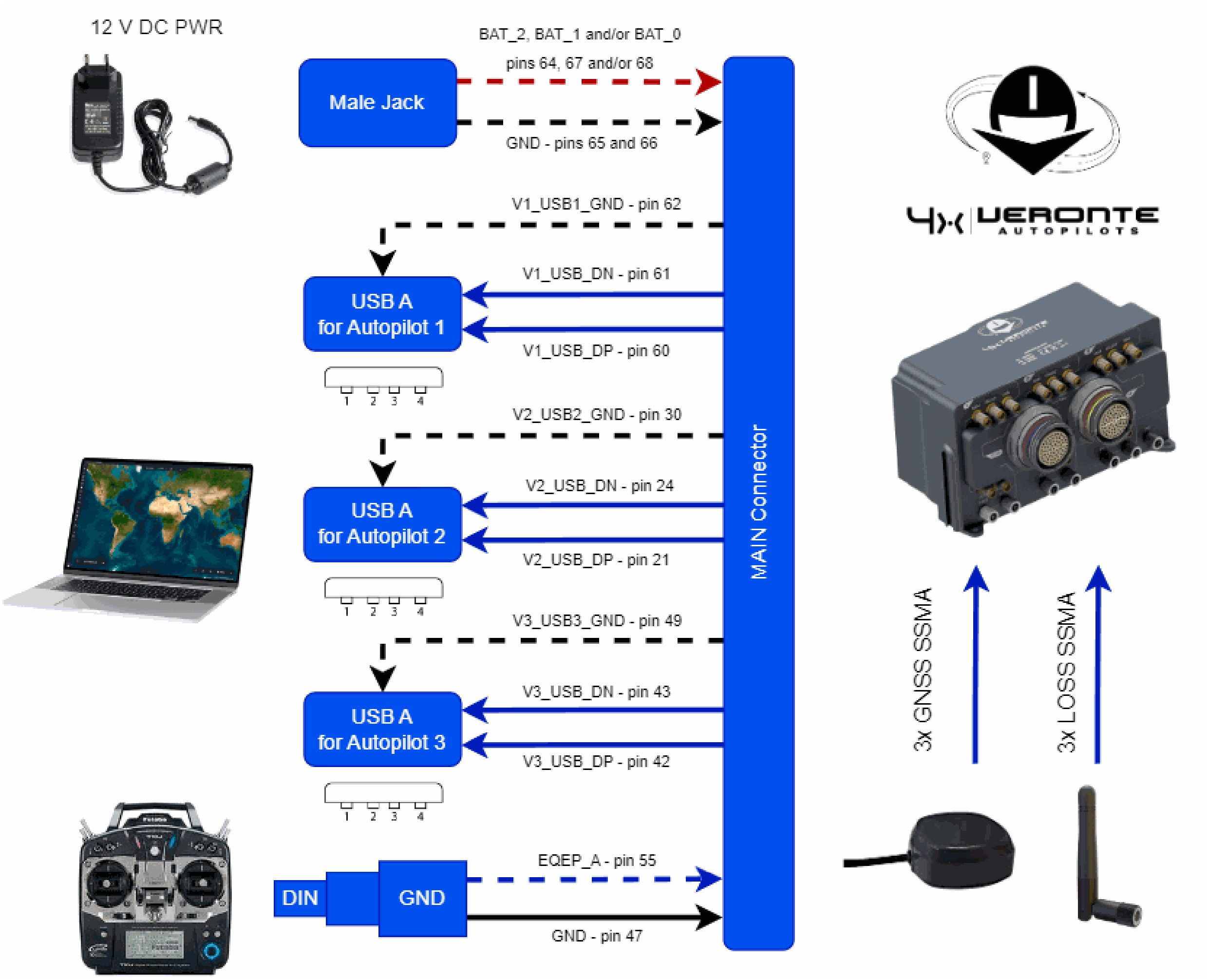


Fig. 1: **Basic Autopilot 4x Ground Station**

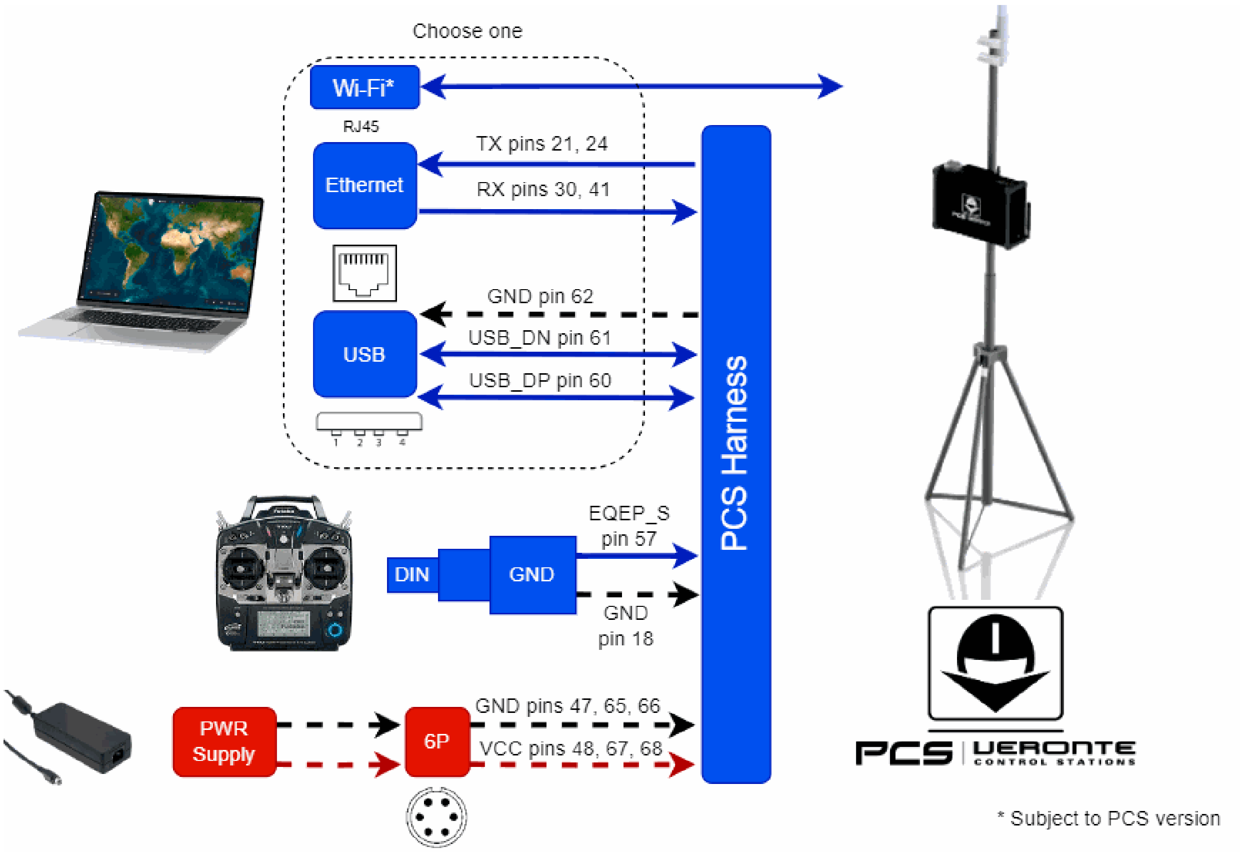


Fig. 2: **Autopilot 4x PCS Ground Station**

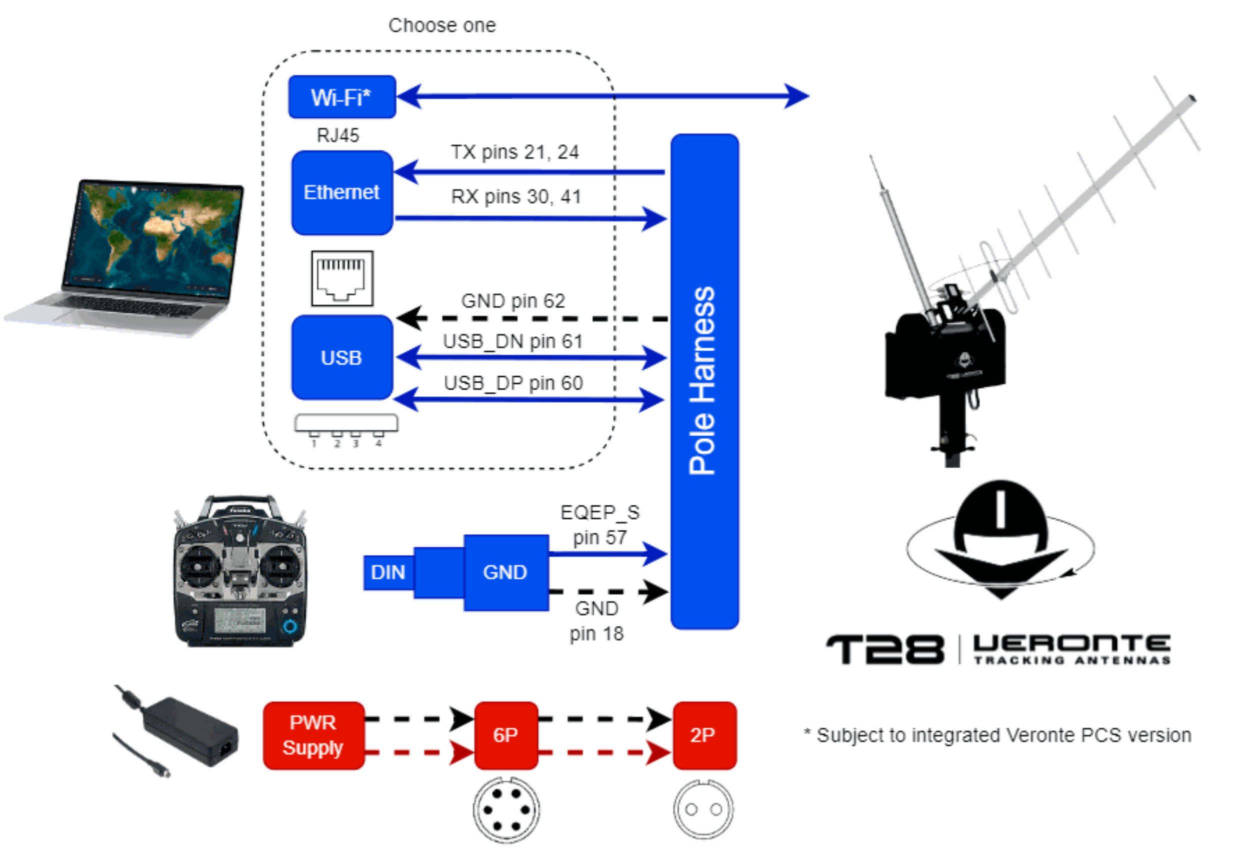


Fig. 3: **Autopilot 4x Tracker Ground Station**

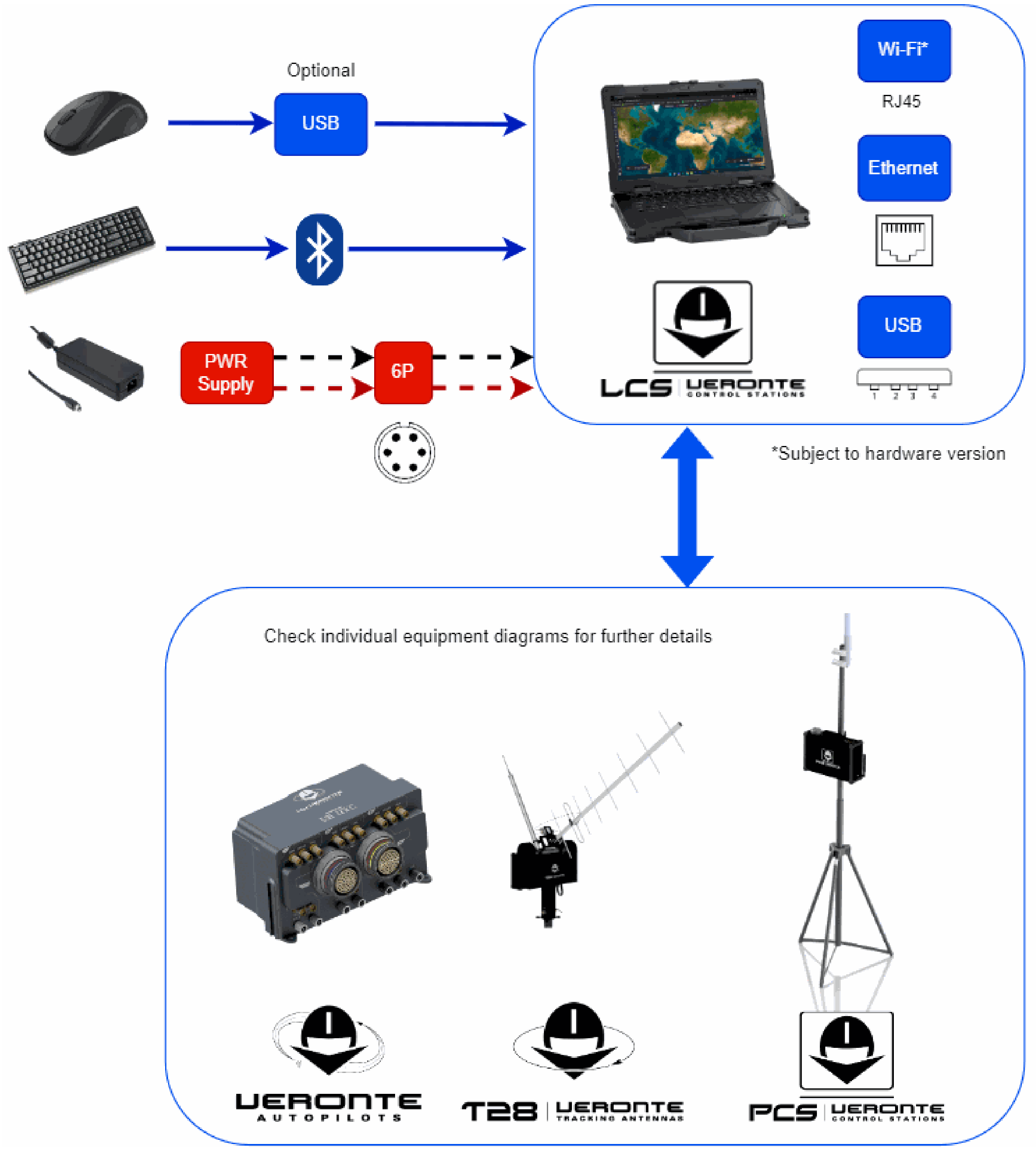


Fig. 4: **Autopilot 4x LCS Ground Station**

Warning: **Veronte Autopilot 4x** equipment harnesses have specific pin layouts. Only use their own matting connectors, do NOT mix harnesses: misuse may lead to destruction.

9.1.2 Aircrafts

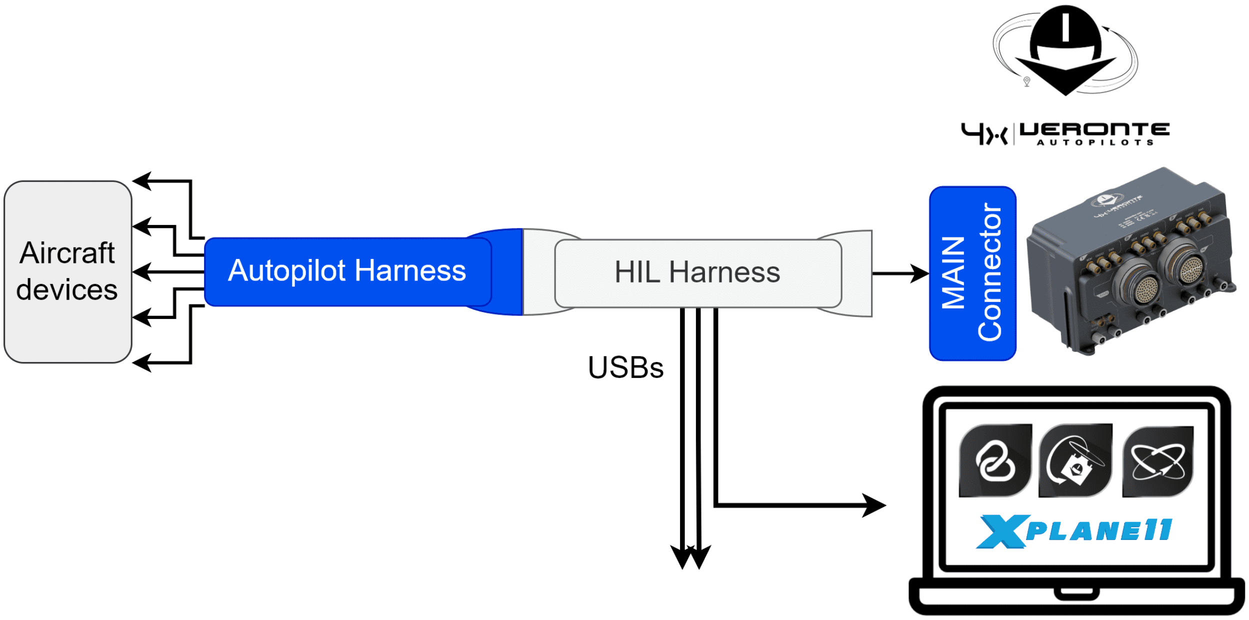


Fig. 5: **HIL Harness**

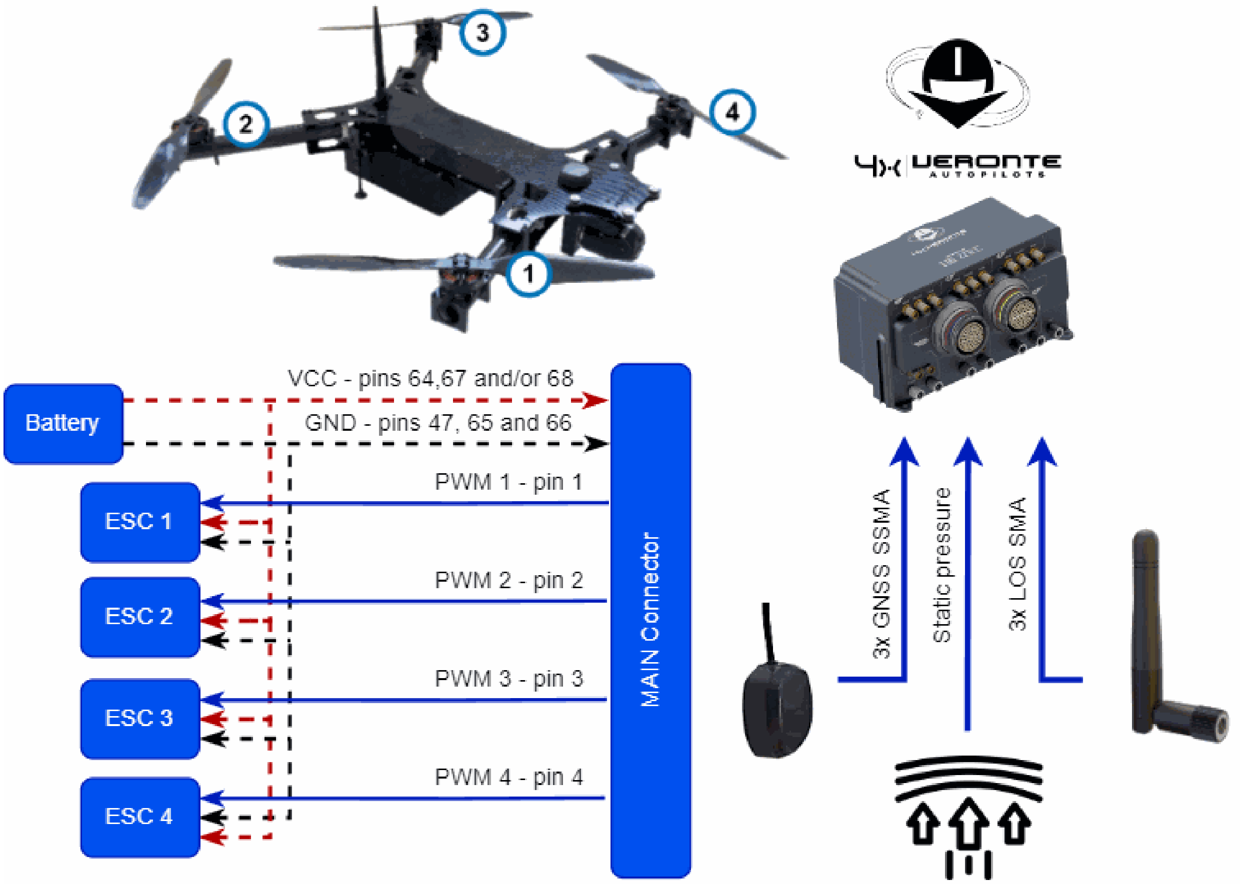


Fig. 6: **Multicopter**

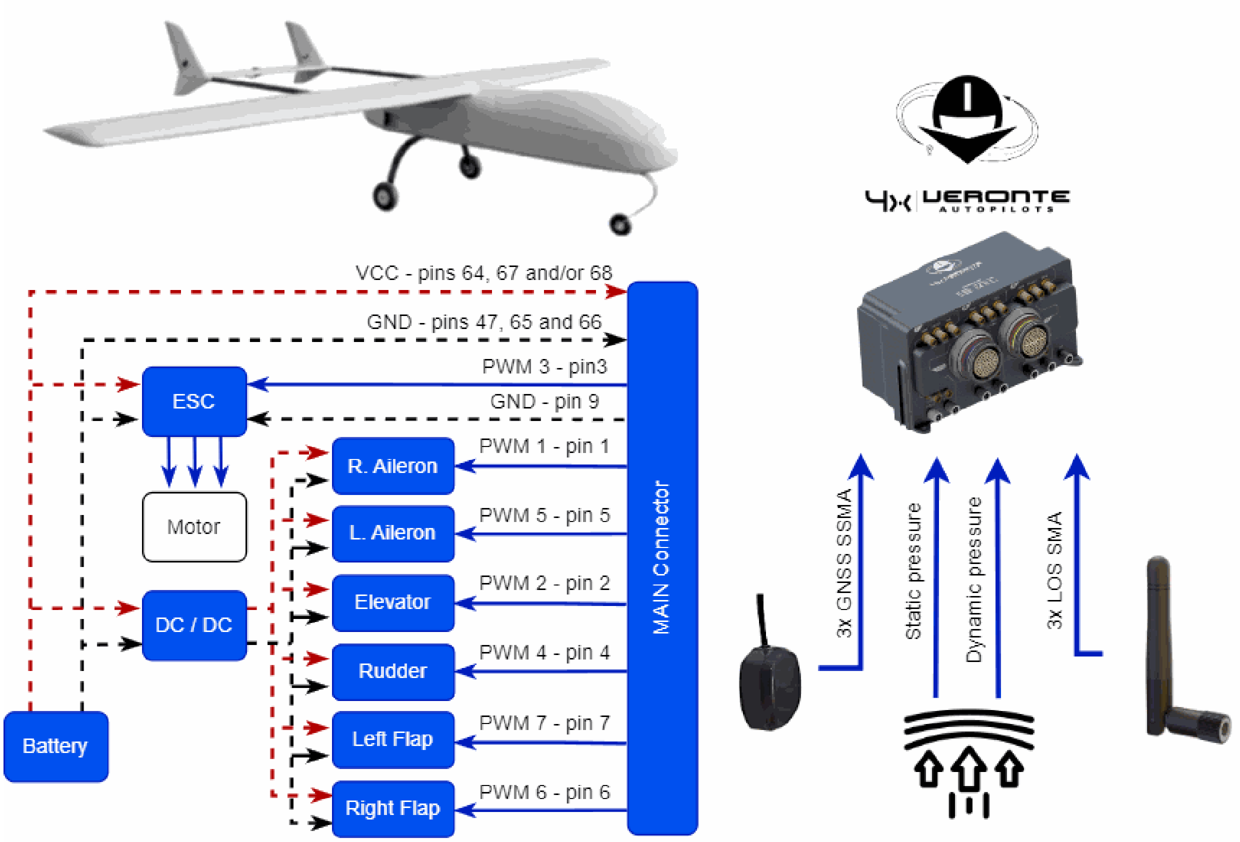


Fig. 7: **Fixed Wing Airplane**

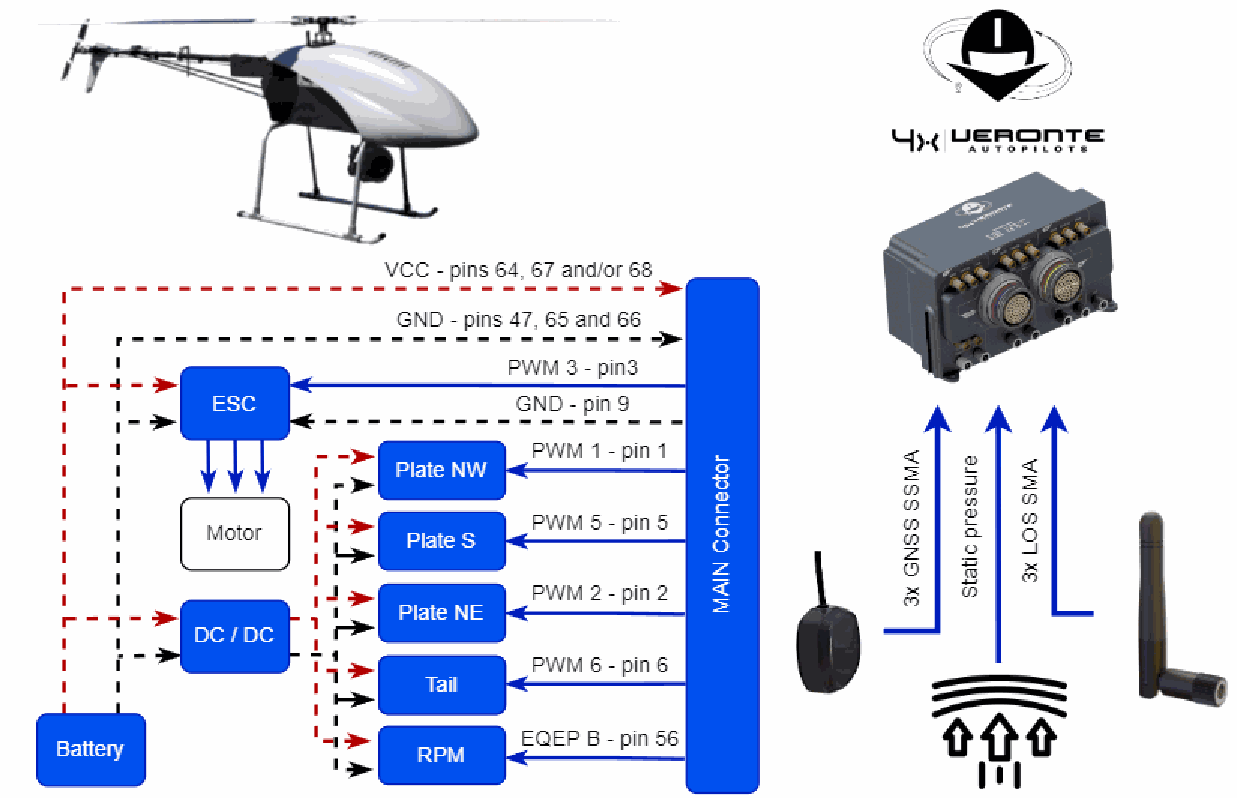


Fig. 8: **Helicopter**

9.2 Air Data Sensors

9.2.1 Temperature sensor LM335

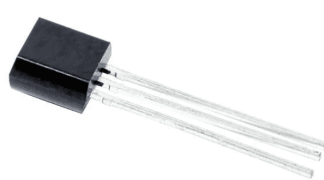


Fig. 9: **LM335 sensor**

The **LM335** is an analogical temperature sensor that measures temperatures from -40°C to 100°C.

It changes the voltage according to the temperature measured and therefore the connection to the autopilot is performed using the ADC pins.

The following wiring is necessary to connect a sensor to the autopilot:

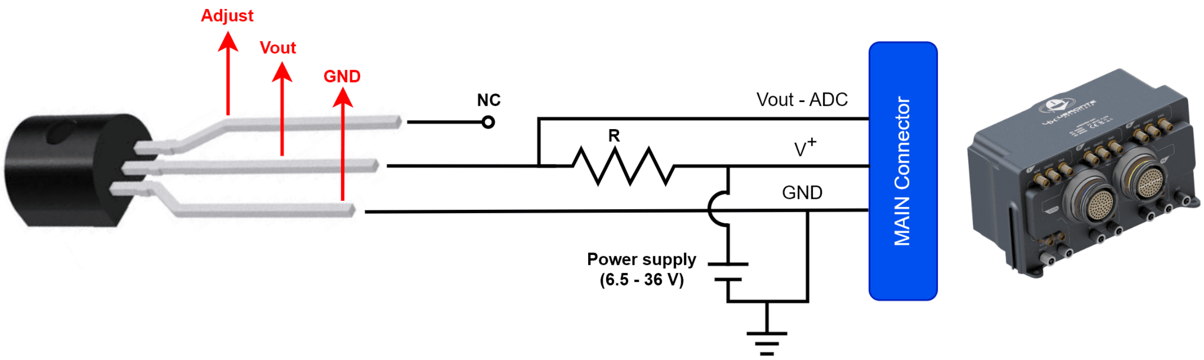


Fig. 10: **LM335 sensor - 4x wiring diagram**

The **LM335** sensor can share voltage supply with the **Autopilot 4x**. However, the impedance of the resistor **R** must vary with the voltage; since the sensor requires a forward current between 0.4 and 5 mA to operate. We recommend **to use a resistor as high as possible** complying with the folllowing equation:

$$\frac{Vin - 2.33}{5 \cdot 10^{-3}} < R < \frac{Vin - 3.73}{5 \cdot 10^{-4}}$$

Where **R** is the value of the resistor (ohms) and **Vin** is the supply voltage (V). The following table shows a list of examples of a tested resistor for each voltage supply. Obtaining an error average of 1.5 °C.

Vin (V)	R (kohms)
6.5	5
12	16
24	37
36	60

The **Vout** pin has to be connected to an analog input of the **Main connector**.

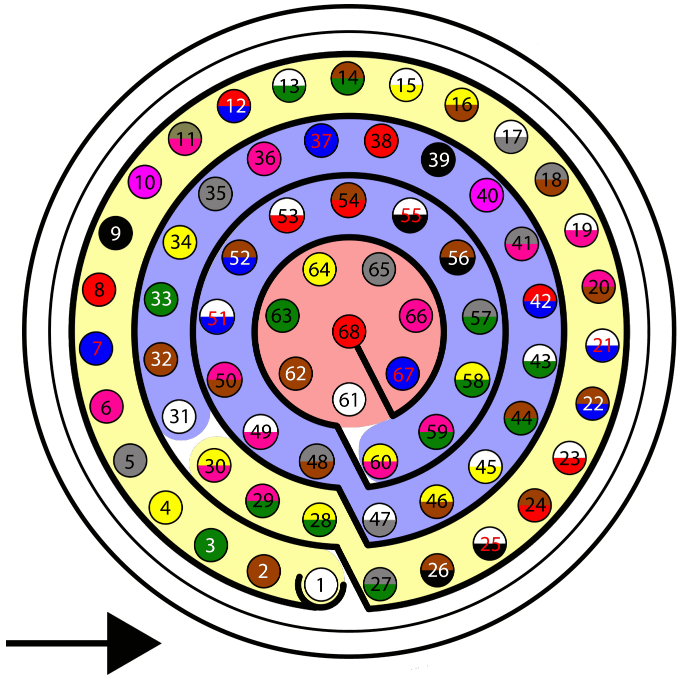


Fig. 11: **Main connector harness pinout**

- Users must choose one **analog** pin to connect:

Main connector harness			LM335 sensor
PIN	Signal	Color code	Signal
22	ANALOG_3	Brown - Blue	Vout
23	ANALOG_4	White - Red	
38	ANALOG_0	Red	
39	ANALOG_1	Black	
40	ANALOG_2	Violet	

- Users must use the **supply voltage** according to the Autopilot 1x being used:

Main connector harness			LM335 sensor
PIN	Signal	Color code	Signal
64	BAT_2	Yellow	V+
67	BAT_1	Blue	
68	BAT_0	Red	

- Users must choose one **ground** pin to use:

Main connector harness			LM335 sensor
PIN	Signal	Color code	Signal
9	GND	Black	GND
18		Gray - Brown	
27		Gray - Green	
33		Green	
41		Gray - Pink	
44		Brown - Green	
47		White - Gray	
48		Gray - Brown	
59		Pink - Green	
63		Green	
65		Gray	
66		Pink	

Once connected, the temperature can be monitored in **1x PDI Builder** by using the variables ADC0 to ADC4. The wiring *explained previously* obtains a relationship of Temperature and **Vout** as follows:

$$T = Vout \cdot 100 - 273$$

Where **T** is the measured temperature (in °C, since 273 is substracted in the formula) and **Vout** the output voltage of the *previous circuit*.

The integration of this device with **Autopilot 4x** is explained in the LM335 with Autopilot 4x - Integration examples section of the **1x PDI Builder** manual.

9.3 Datalinks

9.3.1 LOS

9.3.1.1 Digi radio (as internal radio)

Internal Digi radios can establish communication between Veronte Autopilots.

Each internal **Autopilot 1x** has to be configured one by one with **1x PDI Builder**. The necessary configuration of Digi radios for proper communication between them is described in the Digi internal radio - Integration examples section of the **1x PDI Builder** user manual.

9.3.1.2 Silvus radio (StreamCaster 4200E model)

9.3.1.2.1 System Layout

The following image shows the standard connection between **Silvus** radios and **Autopilot 4x** for operation:

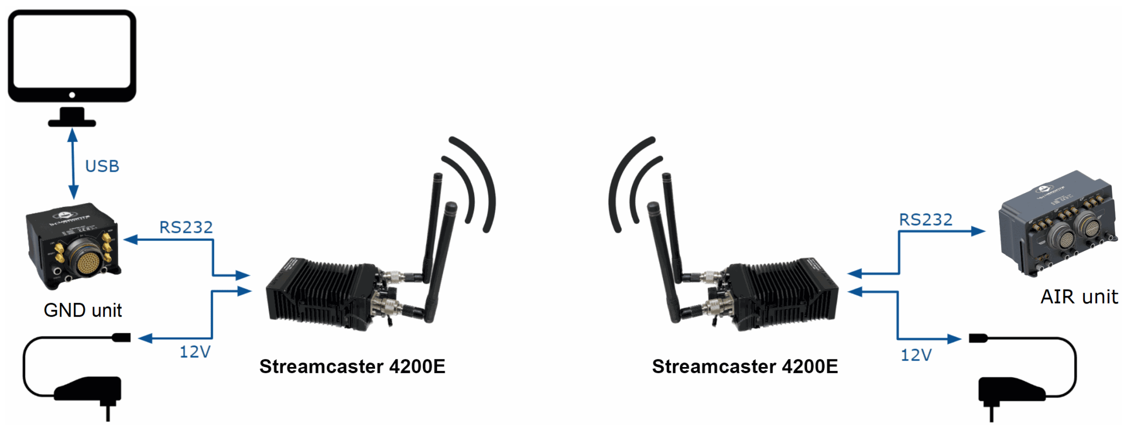


Fig. 12: **Silvus and 4x connection**

9.3.1.2.2 Hardware Installation

A wiring configuration of the PRI cable connected to the PRI port of the radio is required, in order to connect to the power supply, ethernet and RS-232.

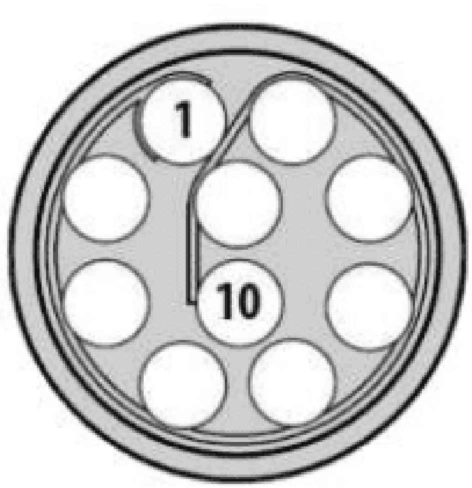


Fig. 13: **PRI port connector (mounted in radio)**

- **Power supply**

Fig. 14: **Female DC Power Jack connector**

PRI port connector - Silvus radio		Power connector
PIN	Signal	Signal
2	GND IN	Power -
3	VCC IN	Power +

- **Ethernet**

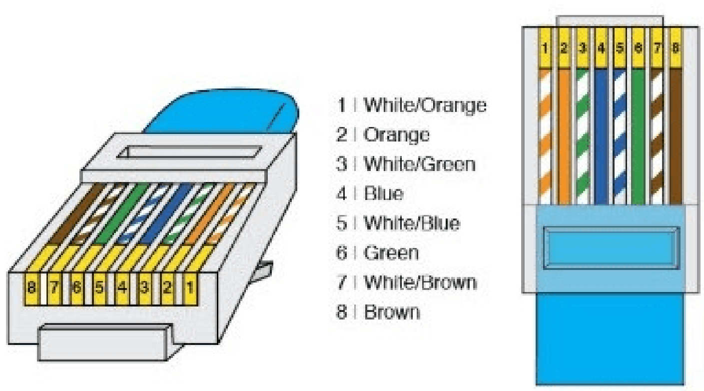


Fig. 15: **RJ45 pinout T-568B**

PRI port connector - Silvus radio		RJ45 Connector (T-568B)		
PIN	Signal	PIN	Signal	Color code
4	ETH0_MX2N (RX-)	6	RX-	Green
5	ETH0_MX2P (RX+)	3	RX+	Green-White
6	ETH0_MX1P (TX+)	1	TX+	Orange-White
10	ETH0_MX1N (TX-)	2	TX-	Orange

- **RS-232**

The RS-232 from the PRI cable should be connected to the RS-232 of **Main Connector** with **Veronte Harness Blue 68P**.

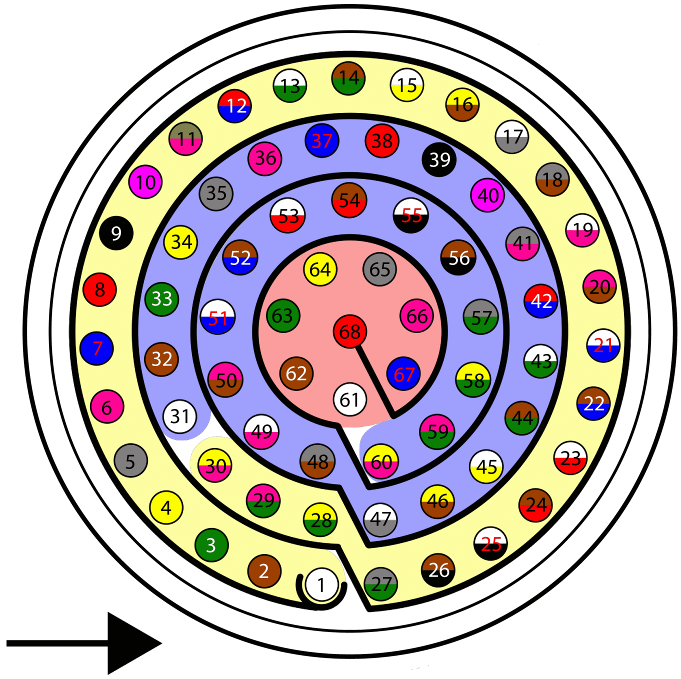


Fig. 16: **Main connector harness pinout**

PRI port connector - Silvus radio		Veronte Harness Blue 68P - Autopilot 4x		
PIN	Signal	PIN	Signal	Color code
7	RS232_RXD	19	MUXED_RS232_TX	White-Pink
8	RS232_TXD	20	MUXED_RS232_RX	Pink-Brown
9	GND	18	GND	Gray-Brown

9.3.1.2.3 Silvus radio configuration

This section shows a basic configuration for **Silvus** radios.

9.3.1.2.3.1 First Steps

1. Connect antennas (or attenuators) with male TNC ends to 2 RF ports.
2. Connect power supply to power port on PRI cable.
3. Connect non-forked female side of PRI cable to radio's PRI port.

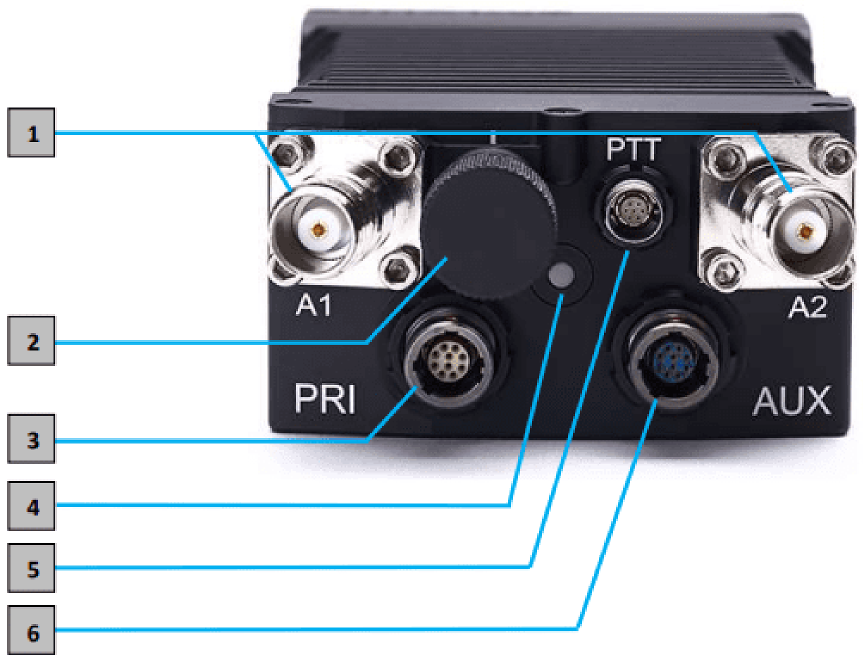


Figure 3 StreamCaster 4200E Ruggedized Enclosure

1 RF Channels 1-2 Connectors [TNC Female]

2 Power Switch [15-Position Rotating]

3 Power (EB Version Only, 9-20V), Ethernet, and Serial Port Connector [ODU GK0YAR-P10UC00-000L]

4 Bi-Color Status LED

- Red – Radio is in the process of booting up
- Flashing Green – Radio is fully booted but not wirelessly connected to any other radio
- Green – Radio is wirelessly connected to at least one other radio

Fig. 17: **Silvus connectors**

4. When looking at the rotary multi position switch from the top, pull the knob towards you while rotating the knob towards the 1 position. This turns radio on. LED indicator will turn to fix red.
5. In order to access the StreamScape graphical user interface (GUI), connect Ethernet (RJ45) connector of PRI cable to Ethernet port of laptop/computer.
6. Make sure computer is set to static IP address on same subnet as radio. The following substeps clarify how to set the IP adress:
   1. Open network and sharing menu and click **Change adapter settings**.

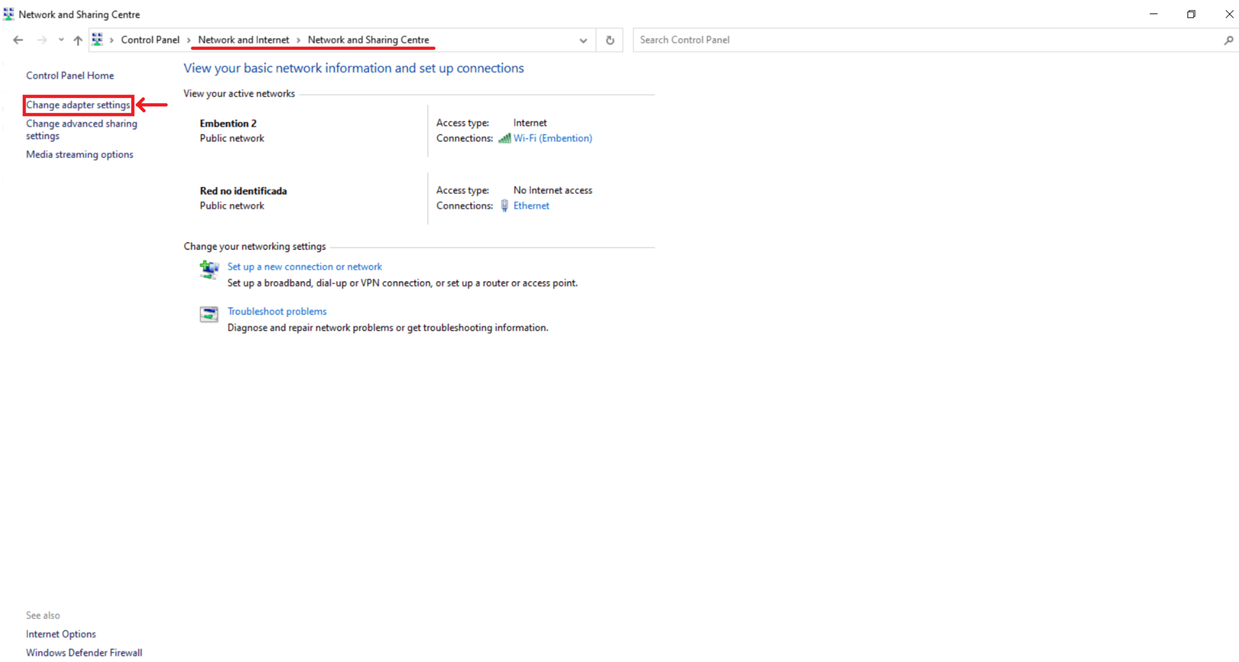


Fig. 18: **Ethernet connection 1**

2. Select **Local Area Connection**, right click, and select **Properties**.

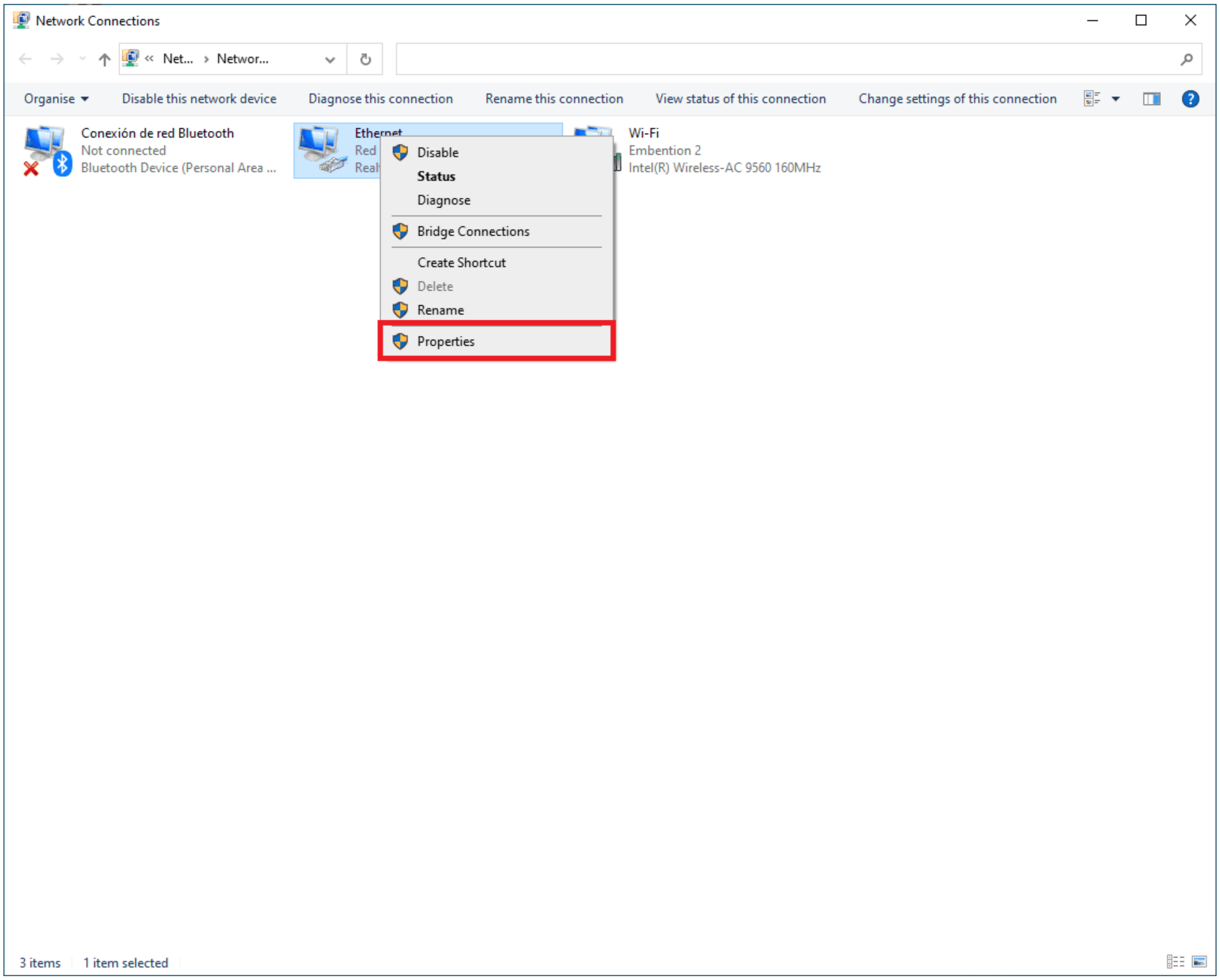


Fig. 19: **Ethernet connection 2**

3. Select **IPv4** and click **Properties**.

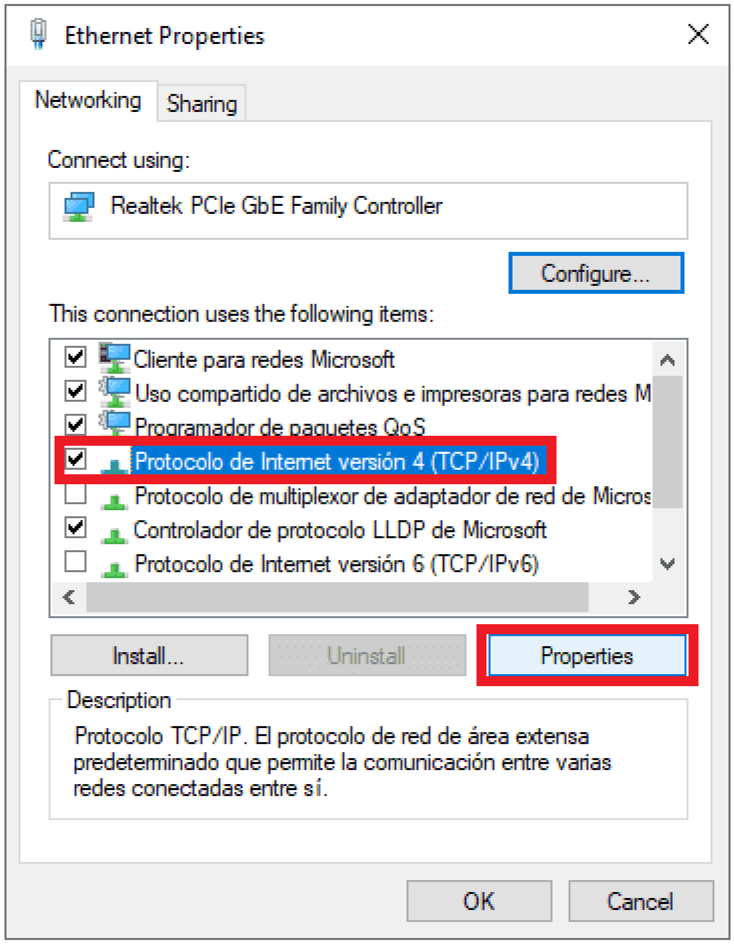


Fig. 20: **Ethernet connection 3**

4. Set **IP address** to 172.20.XX.YY (e.g. if the IP of the radio is 172.20.178.203, set the IP 172.20.178.200) and **Subnet mask** to 255.255.0.0. Click **OK**.

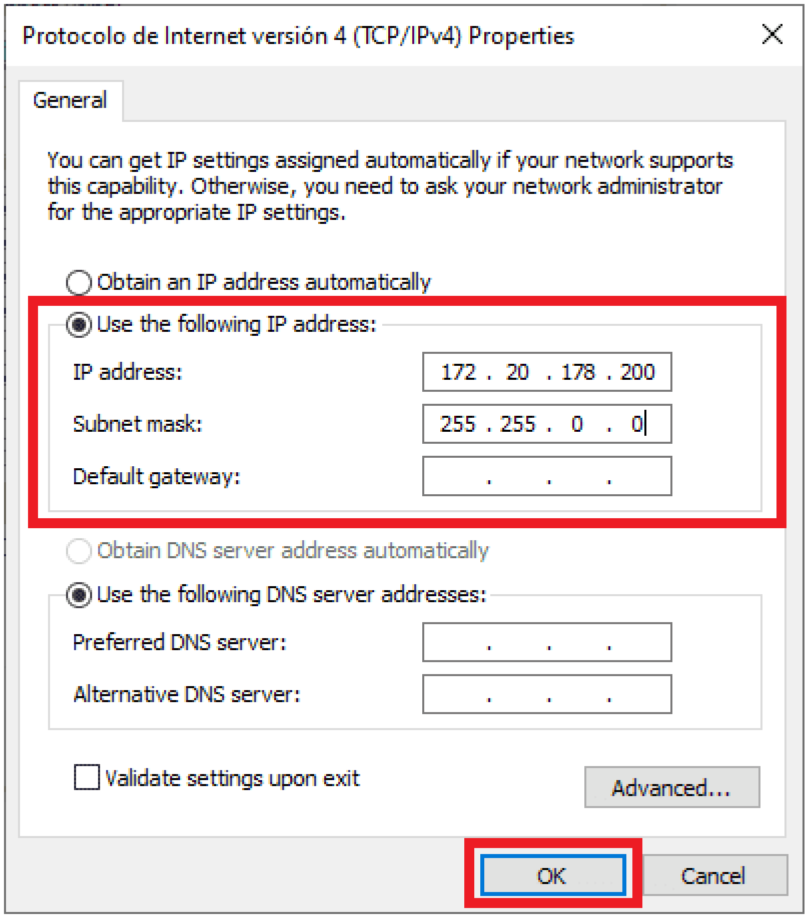


Fig. 21: **Ethernet connection 4**

7. Wait for LED indicator to turn to blinking green.
8. Access **StreamScape** GUI in web browser. To access, enter IP address of radio into web browser and press enter.

> **Note:** Latest version of Firefox or Google Chrome are preferred. Internet Explorer or others are not recommended.

Fig. 22: **Silvus initial menu**

9. User manual can be accessed by clicking the book icon in the GUI (Next to **Basic Configuration** in the previous screenshot).

9.3.1.2.3.2 Basic radio configuration

Once the website has been accessed, follow the steps below which show the parameters that need to be modified for correct operation and pairing of the radios.

Note: This is an example of the radio configuration linked to a 4x air unit.

Note: After making changes to each window, it is important to click on "**Save and apply**".

1. Basic Configuration.

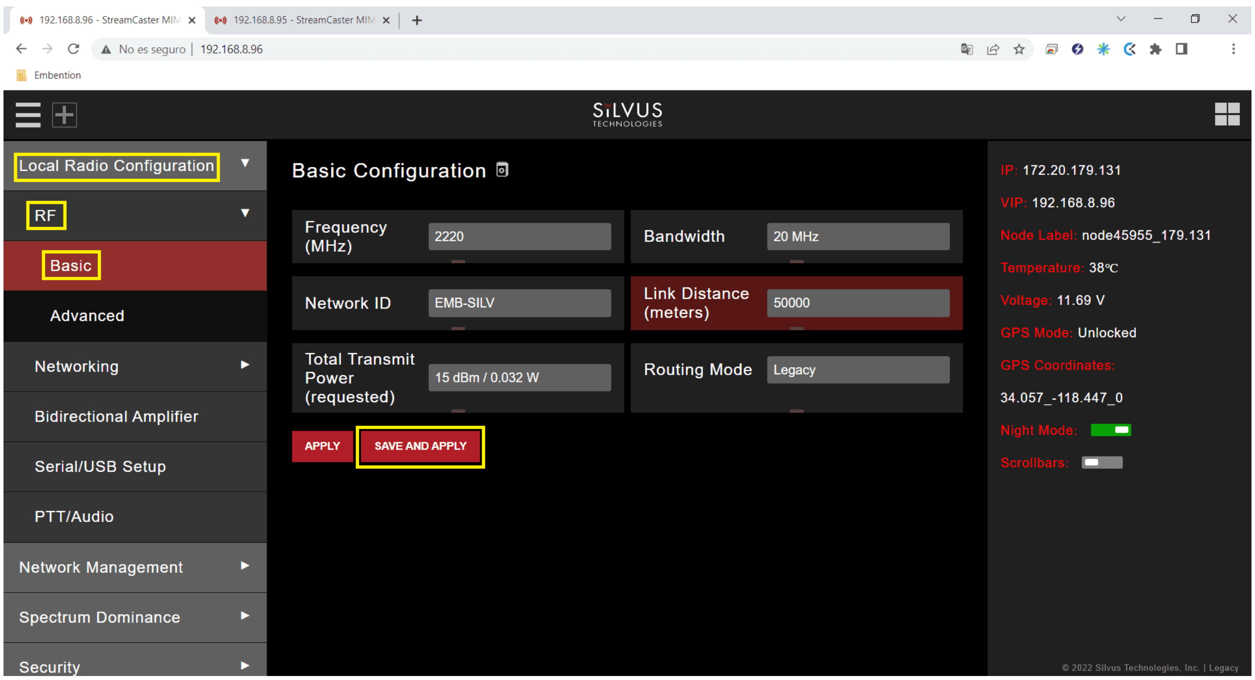


Fig. 23: **Basic configuration panel**

- **Frequency (MHZ)**: This defines the frequency of the signal. There is a drop-down menu for frequency selection. We recommend 2220 MHz.

  Warning: Be careful when choosing the frequency. The user may see interference with the Wifi frequency band, consult the radio spectrum.

- **Bandwith**: This defines the RF bandwidth of the signal. Default value.
- **Network ID**: Network ID allows for clusters of radios to operate in the same channel, but independently.

  A radio with a given Network ID will only communicate with other radios with the same Network ID.
- **Link Distance (meters)**: Set to an approximate maximum distance between any two nodes in meters. It is important to set the link distance to allow enough time for packets to propagate over the air.

  It is recommended to set the link distance 10-15% greater than the actual maximum distance.

- **Total Transmit Power (requested)**: This defines the total power of the signal (power is divided equally between the radio antenna ports). Set the appropriate power for each application. The power that has been set is small, as it is sufficient for our tests.
- **Routing Mode**: As *Large Network* mode requires a license and is not available outside USA, we set *Legacy* mode.

2. Advanced configuration.

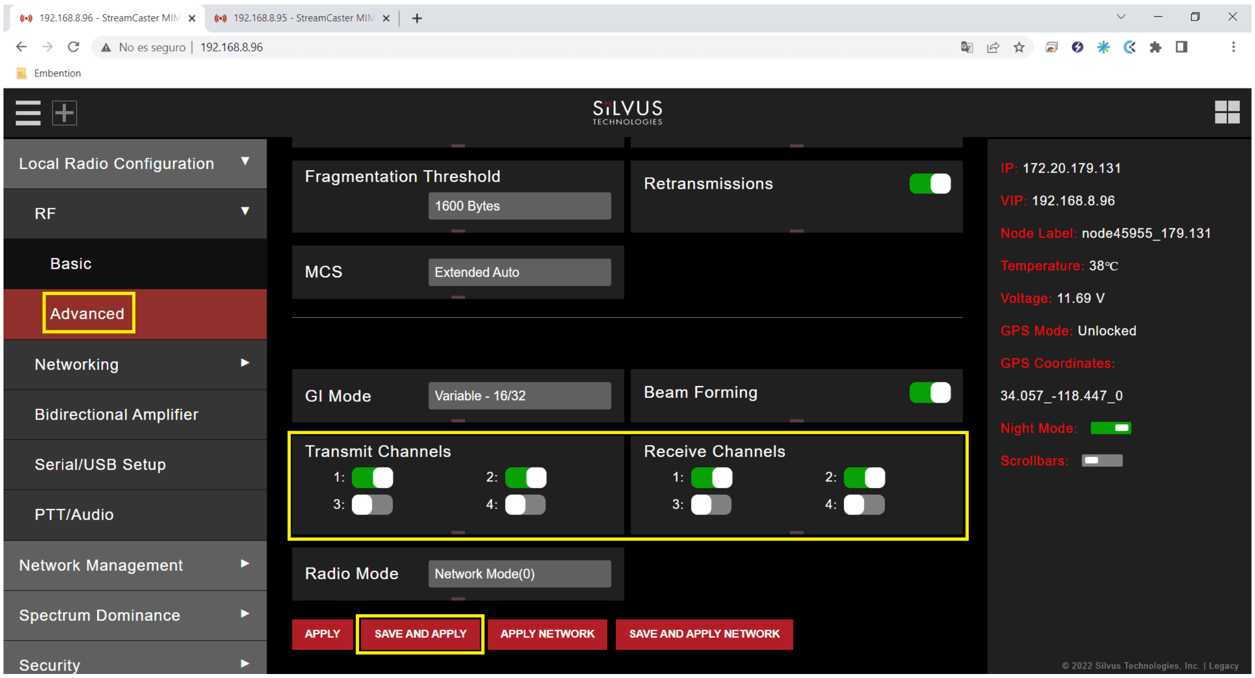


Fig. 24: **Advanced configuration panel**

- **Transmit/Receive Channels**: Allows user to enable or disable each channel on the radio for TX/RX (each RF port is a channel). We have enabled both channels.

3. Networking. Multicast.

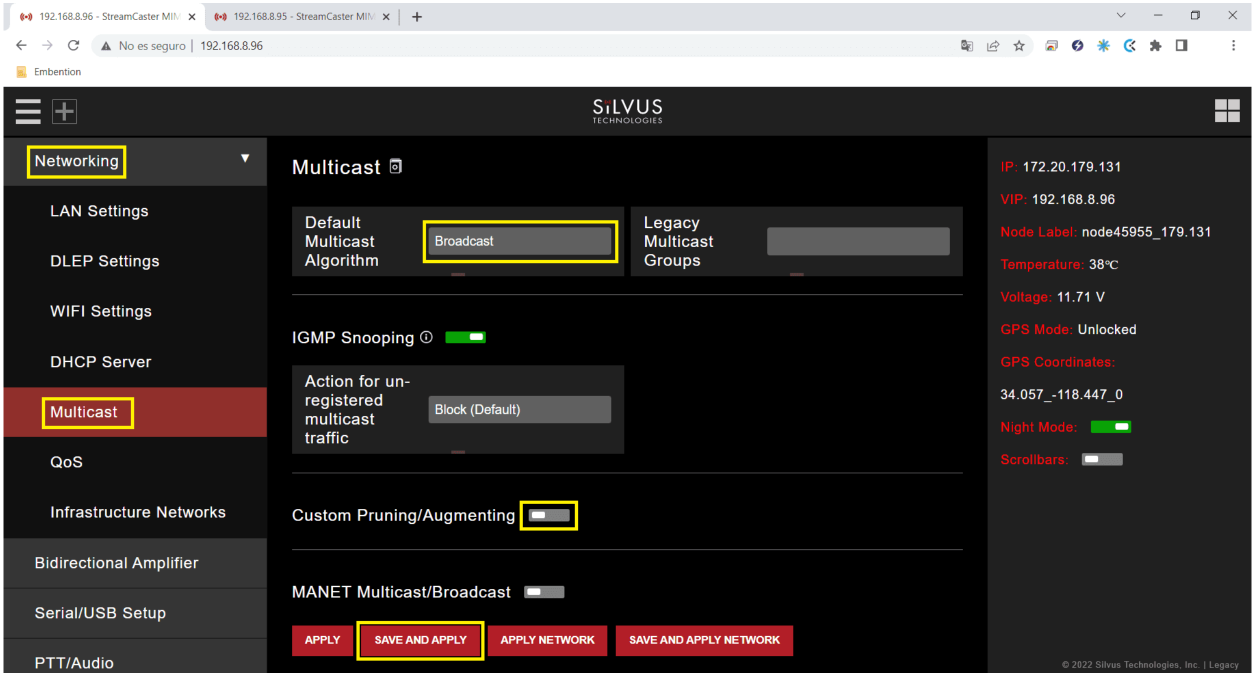


Fig. 25: **Multicast panel**

- **Default Multicast Algorithm**: Broadcast.
- **Custom Pruning/Augmenting**: Disable.

4. Serial/USB Setup

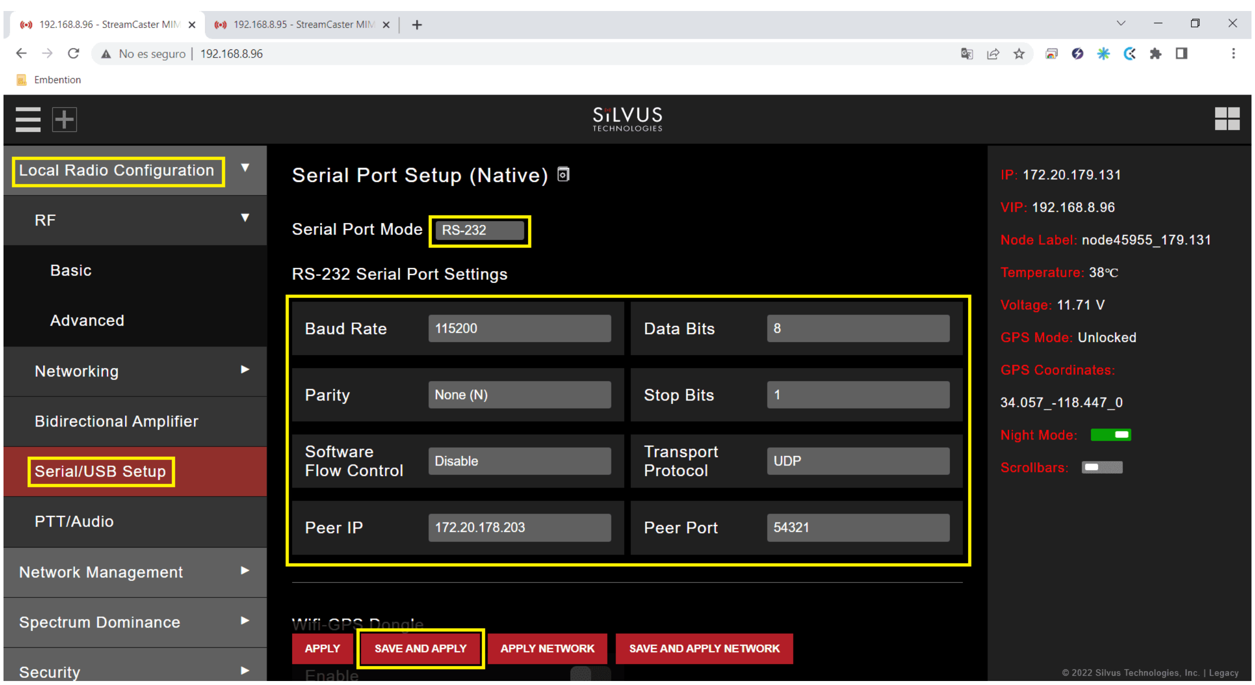


Fig. 26: **RS-232 settings**

- **Serial Port Setup**: RS-232.

- RS-232 Serial Port Settings
  - The value of the **Baudrate**, **Data Bits**, **Parity** and **Stop Bits** parameters must be the same as those configured in 1x PDI Builder software.
  - **Software Flow Control**: Disable.
  - **Transport Protocol**: We recommend **UDP**. If no data loss can be tolerated, change this setting to TCP on the radio corresponding to the 4x **air** unit.
  - **Peer IP**: This should be the IP address of the radio on the other end of the RS-232. In this example, we must set the IP address of the radio linked to the ground unit.

Note: Both radios (the one connected to the GND unit and the one connected to the AIR unit), have the same configuration except for the **Peer IP**.

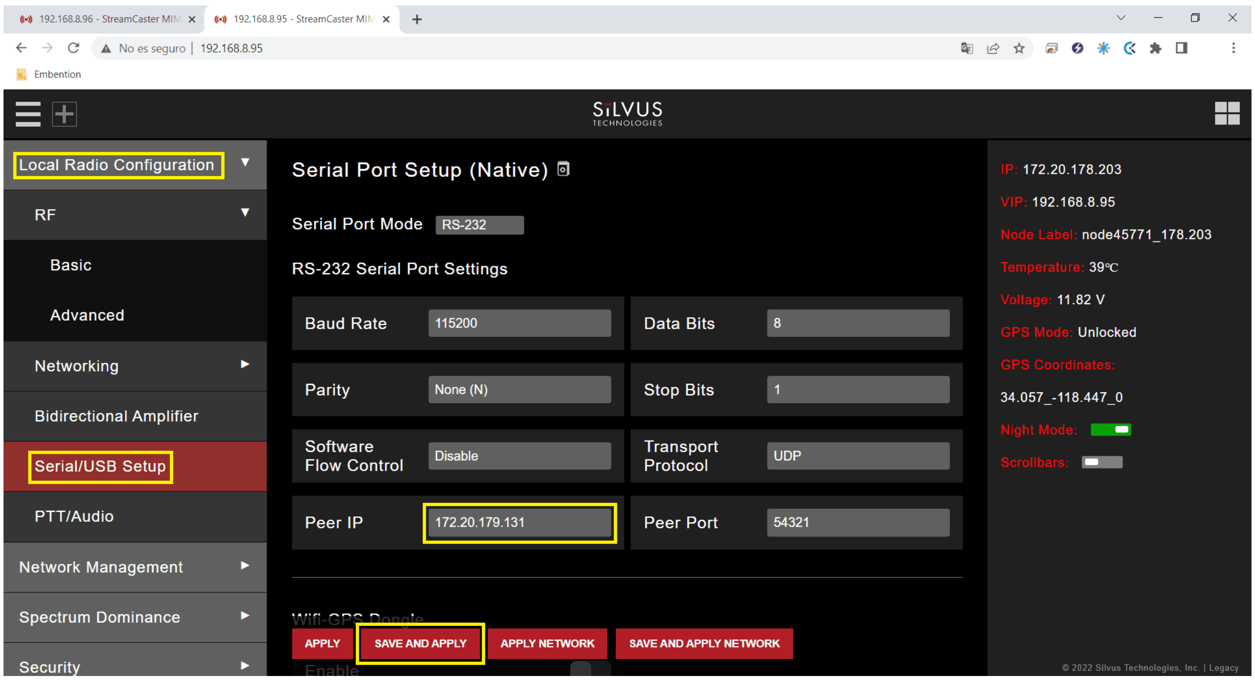


Fig. 27: **Peer IP in radio linked to the GND unit**

In addition to these settings, different configurations can be stored in the same radio, on the **Multi-Position Switch** panel. The user can select the one that will work, with the radio's switch position.

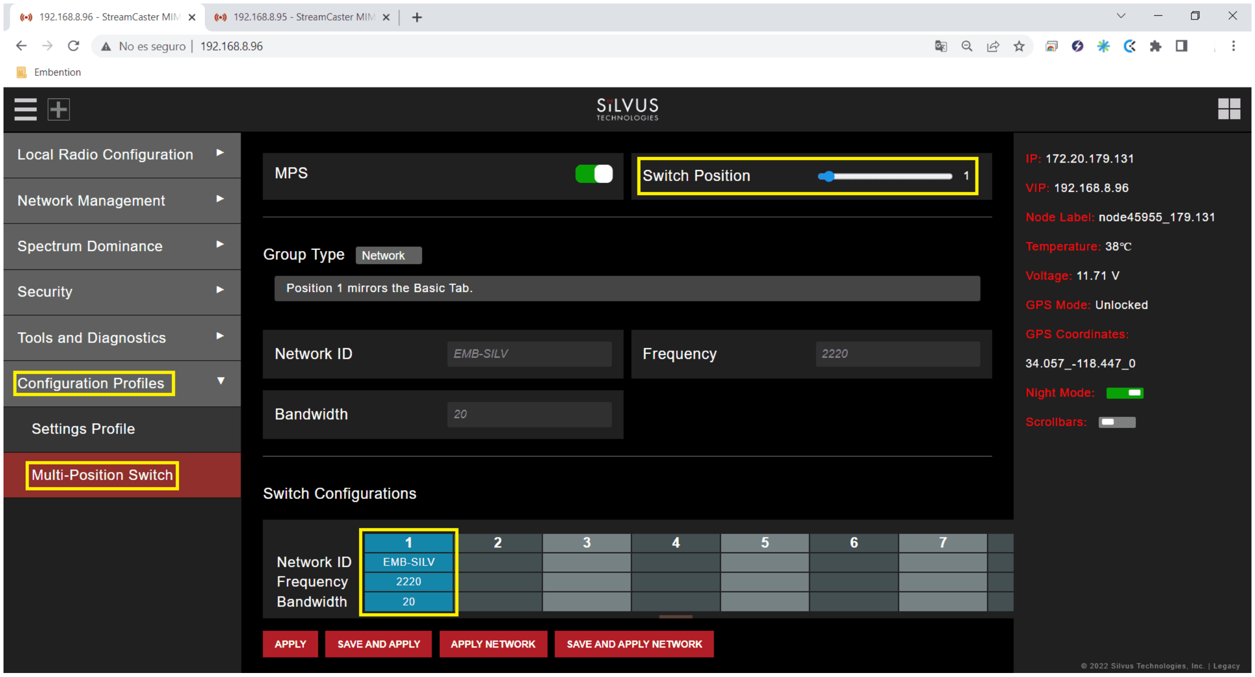


Fig. 28: **Multi-Position Switch panel**

In this example only one configuration has been created.

With the above settings the configuration is finished. Furthermore, this configuration can be saved and downloaded in the **Settings Profile** window of the Configuration Profiles section.

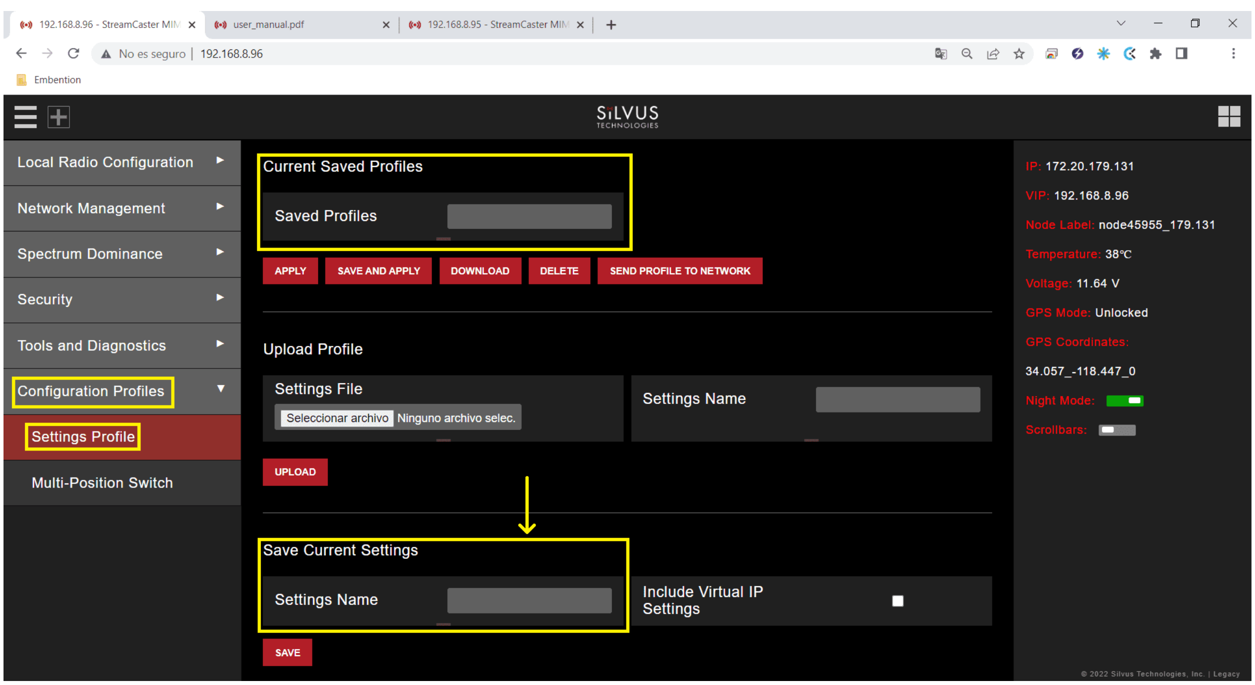


Fig. 29: **Settings Profile panel**

Before downloading the configuration, it is necessary to save it.

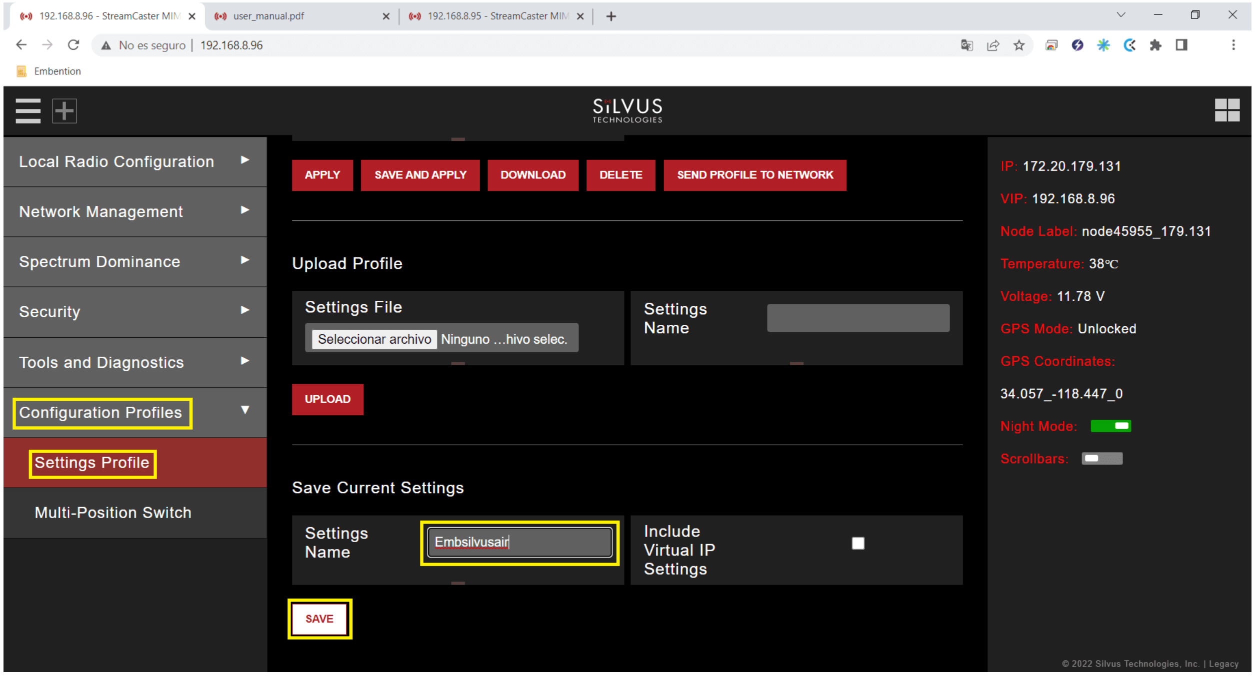


Fig. 30: **Save settings**

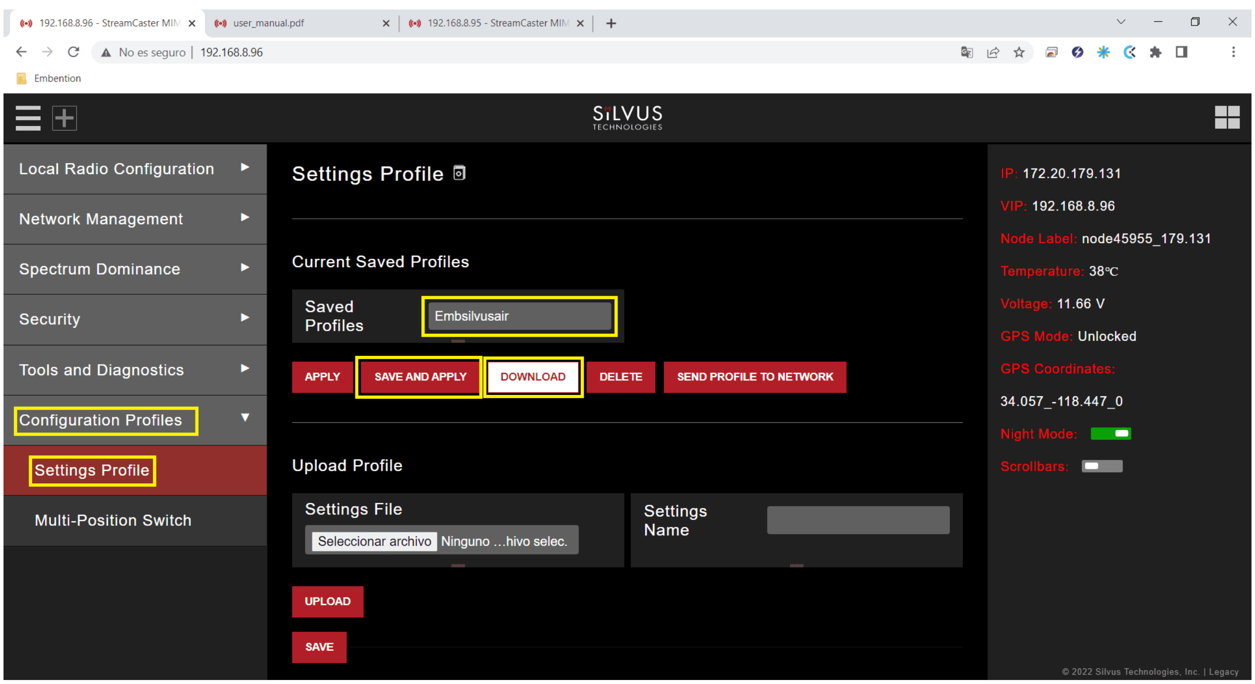


Fig. 31: **Download settings**

After configuring both radios with these settings they should be paired. Therefore, if we connect them to the power supply, when we switch them on, the LED will turn from fix red to fix green, this indicates that it is connected to at least one radio. Also, if we connect only one of them to the computer, we can access the **StreamScape** GUI of both.

And, in the **Network Topology** window of the Network Management section, we can see the link between them.

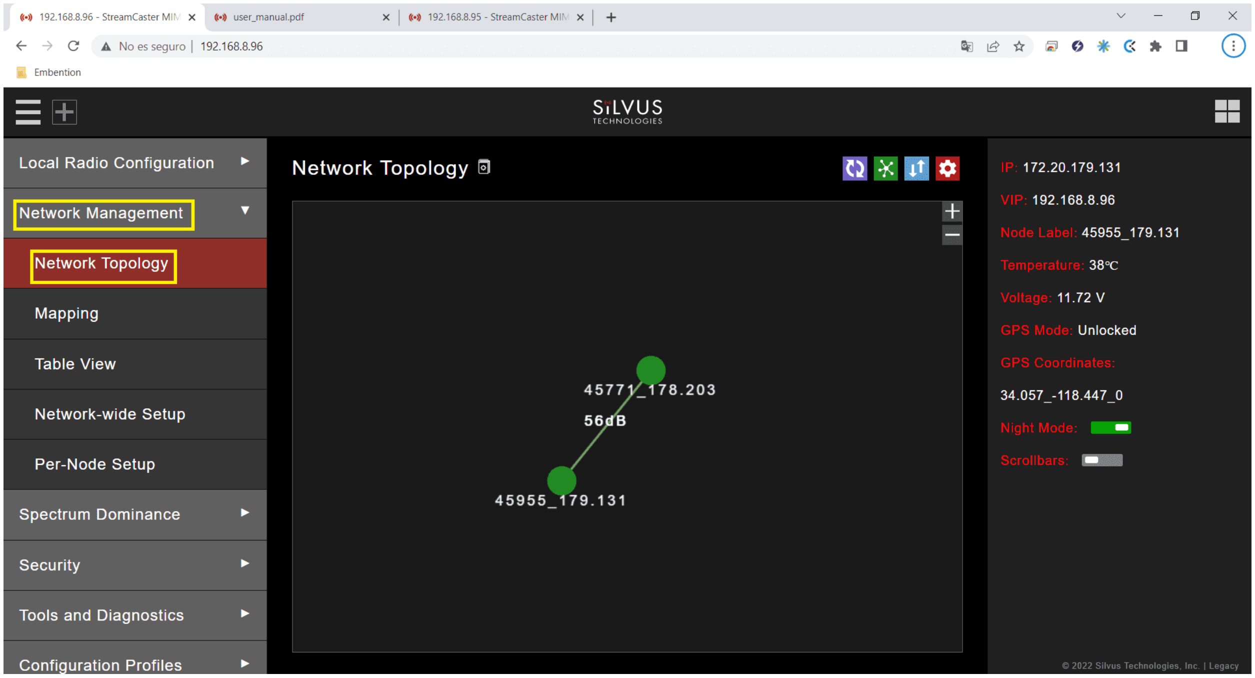


Fig. 32: **Connection between radios**

9.3.1.2.4 Silvus radio configuration in 1x PDI Builder

The necessary configuration of **Silvus** radio in **1x PDI Builder** is described in the External radios - Integration examples section of the **1x PDI Builder** user manual.

9.4 Stick

Veronte Autopilot 4x is compatible with joysticks that use PPM, CAN bus, USB, Serial, etc.

If the PPM level is 3.3V, the following **Autopilot 4x** pins can be used:

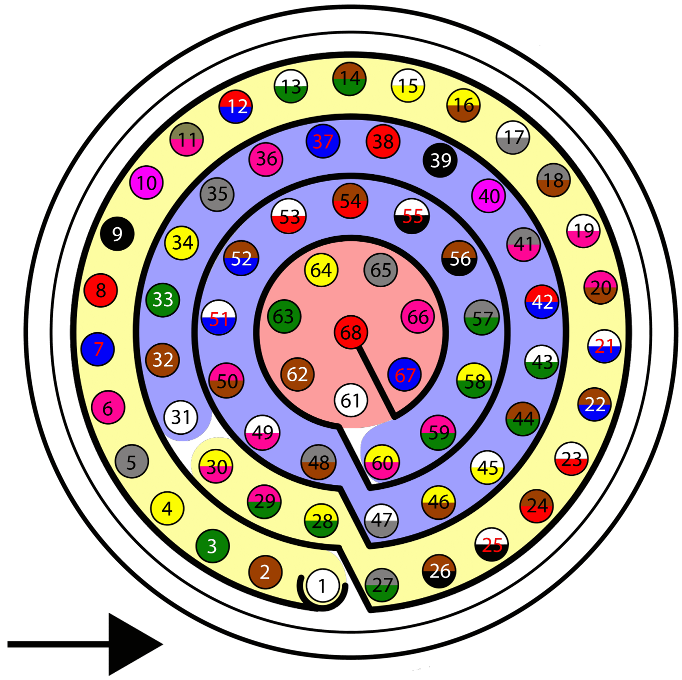


Fig. 33: **Main connector harness pinout**

PIN	Signal	INTERNAL POWER DOMAIN	Color code
1	I/O_0_MUXED	A	White
2	I/O_1_MUXED	B	Brown
3	I/O_2_MUXED	A	Green
4	I/O_3_MUXED	B	Yellow
5	I/O_4_MUXED	A	Gray
6	I/O_5_MUXED	B	Pink
7	I/O_6_MUXED	A	Blue
8	I/O_7_MUXED	B	Red
9	GND*		Black
10	I/O_8_MUXED	A	Violet
11	I/O_9_MUXED	B	Gray - Pink
12	I/O_10_MUXED	A	Red - Blue
13	I/O_11_MUXED	B	White - Green
14	I/O_12_MUXED	A	Brown - Green
15	I/O_13_MUXED	B	White - Yellow
16	I/O_14_MUXED	A	Yellow - Brown
17	I/O_15_MUXED	B	White - Gray
18	GND*		Gray - Brown
55	EQEP_A	A for autopilots 1 and 2 B for autopilot 3	White - Black
56	EQEP_B		Brown - Black
57	EQEP_S		Gray - Green
58	EQEP_I		Yellow - Green
59	GND*		Pink - Green

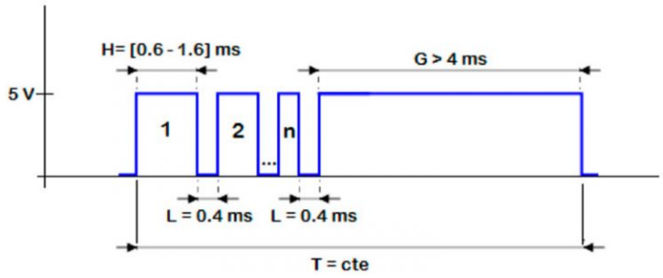


Fig. 34: **PPM signal**

> **Caution:** PPM signal must be into the **Veronte Autopilot 4x** voltage ranges. Some joysticks may need an adaptation board, please ask our team to check compatibility.

Connector for harness is provided with 3.5 mm stereo plug connector as follows:

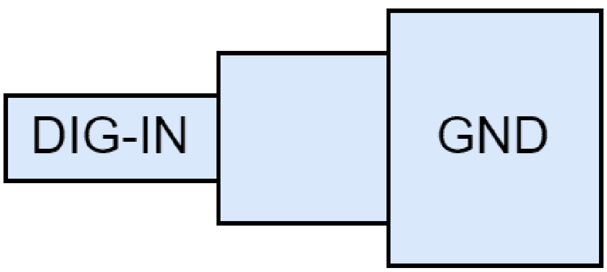


Fig. 35: **PPM pinout**

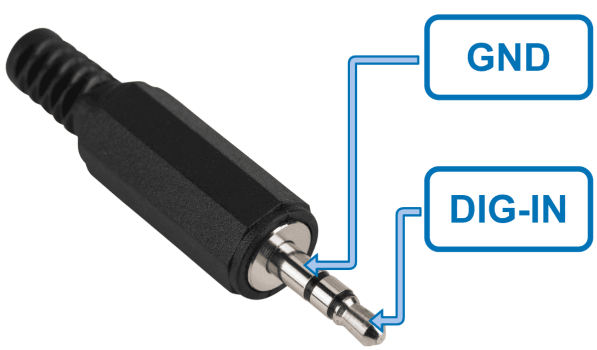


Fig. 36: **PPM connector**

- To use the joystick with **PPM** in the system, connect the PPMout of the trainer port to a digital input of **Veronte Autopilot 4x** and configure that digital input according to the PPM Stick - Integration examples section of the **1x PDI Builder** user manual.
- When using a **USB joystick**, the software installation with **Autopilot 4x** is detailed in the USB joystick - Integration examples section of the **1x PDI Builder** user manual.
- For joysticks with signals **different from PPM or USB**, read the Virtual Stick - Integration examples section of the **1x PDI Builder** user manual.

CHAPTER TEN

TROUBLESHOOTING

In case of any issue with the software of an **Autopilot 1x** located in **Autopilot 4x**, read the Troubleshooting section of the **1x PDI Builder** user manual. Otherwise, in case of any issue with the software of the **Management Board**, read the Troubleshooting section of the **4x PDI Builder** user manual.

10.1 Maintenance mode

Maintenance mode is the main troubleshooting tool that Veronte devices have at the user disposal. While in **maintenance mode**, all communication channels are enabled by default, so it is possible to connect with them through any of its configuration interfaces, no matter its current configuration.

The main use of **maintanance mode** is to solve issues related to the current configuration, mainly related with communication or memory writting issues.

The **maintenance mode** allows to perform actions such as force the load of a new configuration file or to format the SD card.

If at some point the communication with an inner **autopilot 1x** or the **Management Board** is lost, it is possible to use the **maintenance mode** to restore a previous state of the configuration (as long as it was exported previously), format the SD card to start over or update the unit's firmware.

Tip: It is strongly recommended to always use **maintenance mode** to load a new configuration that is very different from the current one.

10.2 How to enter in maintenance mode

There are two ways to enter in **maintenance mode**: using software, or forcing it.

10.2.1 Using software to enter in maintenance mode

To establish the state of an inner **Autopilot 1x** as **maintenance mode** with software, read the Maintenance Mode - Troubleshooting section of the **1x PDI Builder** user manual.

To use the **maintenance mode** for the **Management Board** with software, read the Maintenance Mode - Troubleshooting section of the **4x PDI Builder** user manual.

10.2.2 Forcing maintenance mode

There are two ways to force the maintenance mode: using power supply or using the I2C pins.

> **Caution:** **Arbiter B** cannot enter maintenance mode by forcing it.

10.2.2.1 Power supply

When communication with the **Autopilot 1x** or the **Management Board** is lost, it is possible to active **maintenance mode** using the power supply input.

In order to active **maintenance mode**, power cycle the **Veronte Autopilot 1x** or the **Management Board** repetively with periods of 700 ms (with a margin range between 380 and 965 ms). After 30 cycles, the device will enter in **maintenance mode**.

Autopilot 1x and **Management Board** might enter in **maintenance mode** if a problem with the power supply is detected upon boot up (voltage or current is out of range).

Fig. 1: **How to power cycle an autopilot**

10.2.2.2 I2C pins

10.2.2.2.1 Arbiter A

To enter in **maintenance mode** with I2C, unplug Veronte Autopilot 4x (the **3 inner Autopilots 1x** and the **Management Board**), **connect both I2C pins each other**, then power up **Autopilot 4x**, finally disconnect both pins. Both pins are SCL_A_OUT_ARB (number 39) and SDA_A_OUT_ARB (number 40) according to the *Arbiter Connector pinout*.

10.2.2.2.2 Internal Autopilots 1x

It is possible to enter in **maintenance mode** using I2C pins with an inner autopilot in the same way as the Arbiter A. However, it requires **first selecting the desired Autopilot 1x** with the **Management Board** via the 4x PDI Builder software. To do this:

1. Go to Arbitration menu → **Config panel**.
2. Then, select as **Method** of arbitration the **Fixed** of the corresponding autopilot to be forced in maintenace mode.

   For example, if Autopilot 1 is to be forced into maintenace mode, select *Fixed 0* as the arbitration method.
3. Finally, save the changes.

Note: **Veronte Autopilots: Dev Harness 4x 1.8** (Embention reference `P007695`) has already included a button with this 2 pins to easily enter maintenance mode.

The procedure is the same as for the pins, but instead of connecting and disconnecting the pins, press and release the button.

CHAPTER

ELEVEN

HARDWARE CHANGELOG

Hereby are described the main differences between the latest release of the **Veronte Autopilot 4x** hardware (v **1.8**) and the previous commercial version (v **1.2**).

Note: Note that all the technical differences related to **Autopilot 1x** are detailed in the Hardware Changelog section of the **1x Hardware Manual**.

New funcionalities

- Addition of second arbitrer microcontroller
- Enhancement hardware detection mechanism for arbiters A and B
- Implementation of independent power domains for each microcontroller and different peripherals
- USB ID pin has a different functionality, now is the USB shielding connection
- 5V_BUS and 3.3V_BUS removed from Main connector
- Three UARTs (one from each Autopilot 1x) added to the redundant connector

Improvements

- Optimized power supply circuit

11.1 Pinout changes from Autopilot 4x 1.2

The pinout for `1.2` and `1.8` versions are very similar, but they have several differences. To prevent any confusion, the following table shows the pinout for both versions. The different pins are marked with , all the rest have the same function.

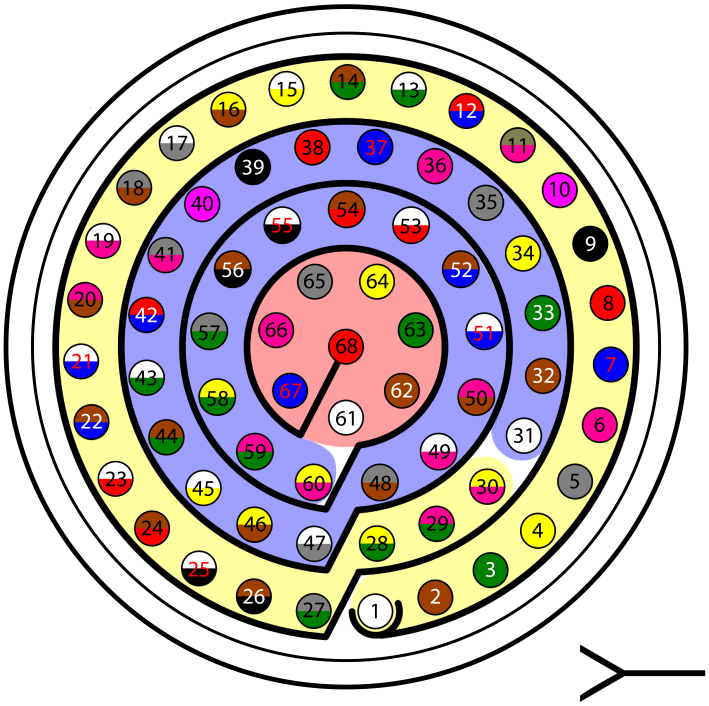


Fig. 1: **Pinout of Main and Arbiter connectors for both versions**

11.1.1 Main Connector pinout

PIN	Signal	Type	Internal Power Domain	Description
1	I/O_0_MUXED	I/O	A	MUXED PWM / Digital I/O signal (0-3.3V) **Warning:** Each pin withstands a maximum current of 1.65 mA.
2	I/O_1_MUXED	I/O	B	
3	I/O_2_MUXED	I/O	A	
4	I/O_3_MUXED	I/O	B	
5	I/O_4_MUXED	I/O	A	
6	I/O_5_MUXED	I/O	B	
7	I/O_6_MUXED	I/O	A	
8	I/O_7_MUXED	I/O	B	
9	GND*	GROUND		Ground pin for signals 1-8
10	I/O_8_MUXED	I/O	A	MUXED PWM / Digital I/O signal (0-3.3V) **Warning:** Each pin withstands a maximum current of 1.65 mA.
11	I/O_9_MUXED	I/O	B	
12	I/O_10_MUXED	I/O	A	
13	I/O_11_MUXED	I/O	B	
14	I/O_12_MUXED	I/O	A	
15	I/O_13_MUXED	I/O	B	

continues on next page

Table 1 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
16	I/O_14_MUXED	I/O	A	
17	I/O_15_MUXED	I/O	B	
18	GND*	GROUND		Ground pin for signals 10-17
19	MUXED_RS232_TX	OUTPUT	A	MUXED RS-232 output
20	MUXED_RS232_RX	INPUT	A	REDUNDANT RS-232 input
21	V2_USB_DP	I/O		Autopilot 2 USB positive data line
22	ANALOG_3	INPUT	B	REDUNDANT analog input (0-36V)
23	ANALOG_4	INPUT	B	
24	V2_USB_DN	I/O		Autopilot 2 USB negative data line
25	CANA_ARB_P	I/O	A	CAN-bus interface. It supports data rates up to 1 Mbps. A 120 Ohm Zo is required and twisted pair is recommended.
26	CANA_ARB_N	I/O	A	
27	GND*	GROUND		GROUND pin for buses (except USB)
28	CANB_ARB_P	I/O		CAN-bus interface. It supports data rates up to 1 Mbps. A 120 Ohm Zo is required and twisted pair is recommended
29	CANB_ARB_N	I/O		
30	`1.2:` V2_USB_ID	`1.2:` I/O		`1.2:` Veronte 2 USB ID Line
	`1.8:` V2_USB2_GND	`1.8:` GROUND		`1.8:` Autopilot 2 USB ground
31	I2C_CLK	OUTPUT A		MUXED Clock line for I2C bus (0.3V to 3.3V)
32	I2C_DATA	I/O	A	MUXED data line for I2C bus
33	GND*	GROUND		Ground for 3.3V power supply
34	`1.2:` 3.3V	`1.2:` POWER	B	`1.2:` 3.3V-100mA power supply
	`1.8:` V1_ARB_TX	`1.8:` OUTPUT		`1.8:` Microcontroller UART transmitter for Autopilot 1
35	`1.2:` GND	`1.2:` GROUND		`1.2:` Ground for 5V power supply

continues on next page

Table 1 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
	1.8: V1_ARB_RX	1.8: INPUT		1.8: Microcontroller UART receiver for Autopilot 1
36	1.2: 5V	1.2: POWER	B	1.2: 5V-100mA power supply
	1.8: V2_ARB_TX	1.8: OUTPUT		1.8: Microcontroller UART transmitter for Autopilot 2
37	1.2: GND	1.2: GROUND		1.2: Ground for analog signals
	1.8: V2_ARB_RX	1.8: INPUT		1.8: Microcontroller UART receiver for Autopilot 2
38	ANALOG_0	INPUT	A	REDUNDANT analog input (0-36V)
39	ANALOG_1	INPUT	A	
40	ANALOG_2	INPUT	A	
41	GND*	GROUND		Ground signal for buses
42	V3_USB_DP	I/O		Autopilot 3 USB positive data line
43	V3_USB_DN	I/O		Autopilot 3 USB negative data line
44	GND*	GROUND		Ground signal for buses
45	1.2: UART_TX	1.2: OUTPUT	B	1.2: Muxed UART output
	1.8: V3_ARB_TX	1.8: OUTPUT		1.8: Microcontroller UART transmitter for Autopilot 3
46	1.2: UART_RX	1.2: INPUT	B	1.2: Redundant UART input
	1.8: V3_ARB_RX	1.8: INPUT		1.8: Microcontroller UART receiver for Autopilot 3
47	GND*	GROUND		Ground signal for buses
48				
49	1.2: V3_USB_ID	1.2: I/O		1.2: Veronte 3 USB ID Line
	1.8: V3_USB3_GND	1.8: GROUND		1.8: Autopilot 3 USB ground
50	OUT_RS485_P	OUTPUT	B	MUXED non-inverted output RS-485 bus

continues on next page

Table 1 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
51	OUT_RS485_N	OUTPUT	B	MUXED inverted output RS-485 bus
52	IN_RS485_N	INPUT		REDUNDANT inverted inout RS-485 bus
53	IN_RS485_P	INPUT		REDUNDANT non-inverted input RS-485 bus
54	RS485_GND	GROUND		Ground for RS-485 bus
55	EQEP_A	INPUT	A for autopilots 1 and 2 B for autopilot 3	Encoder quadrature redundant input A (0-5V)
56	EQEP_B	INPUT		Encoder quadrature redundant input B (0-5V)
57	EQEP_S	INPUT		Encoder strobe redundant input (0-5V)
58	EQEP_I	INPUT		Encoder index redundant input (0-5V)
59	GND*	GROUND		Autopilot 3 ground pin
60	V1_USB_DP	I/O		Autopilot 1 USB positive data line
61	V1_USB_DN	I/O		Autopilot 1 USB negative data line
62	`1.2:` V1_USB_ID	`1.2:` I/O		`1.2:` Veronte 1 USB ID Line
	`1.8:` V1_USB1_GND	`1.8:` GROUND		`1.8:` Autopilot 1 USB ground
63	GND*	GROUND		Ground signal for buses
64	BAT_2	POWER		Autopilot 3 power supply (6.5 to 36V)
65	GND*	GROUND		Autopilot 2 ground pin
66	GND*	GROUND		Autopilot 1 ground pin
67	BAT_1	POWER		Autopilot 2 power supply (6.5 to 36V)
68	BAT_0	POWER		Autopilot 1 power supply (6.5 to 36V)

Warning: Common grounds are marked with *.

11.1.2 Arbiter Connector pinout

Although being the same component, **Main Connector** and **Arbiter connector** are polarized differently, but they have different mechanical connections to avoid wiring swapping.

PIN	Signal	Type	Internal Power Domain	Description
1	I/O_0_EXTERNAL	I/O	A	External MUXED PWM / Digital I/O signal (0-3.3V). In case of employing an additional external autopilot, its pins **I/XX** must be connected here. Each signal **I/O_XX_EXTERNAL** will be sent to **I/XX** of **Main Connector** if the arbiter commands it. **Warning:** Input current must be limited to 25 mA for each I/O EXTERNAL pin.
2	I/O_1_EXTERNAL	I/O	B	
3	I/O_2_EXTERNAL	I/O	A	
4	I/O_3_EXTERNAL	I/O	B	
5	I/O_4_EXTERNAL	I/O	A	
6	I/O_5_EXTERNAL	I/O	B	
7	I/O_6_EXTERNAL	I/O	A	
8	I/O_7_EXTERNAL	I/O	B	
9	I/O_8_EXTERNAL	I/O	A	
10	I/O_9_EXTERNAL	I/O	B	
11	I/O_10_EXTERNAL	I/O	A	
12	I/O_11_EXTERNAL	I/O	B	
13	`1.2:` VCC2	`1.2:` POWER		`1.2:` Veronte 2 power supply (6.5 to 36 V)
	`1.8:` ARBITER_ANALOG_1	`1.8:` INPUT		`1.8:` Arbiter analog input (0-36V)
14	EXTERNAL_ANALOG_0	OUTPUT	A	External analog signal (0-3V). This is the analog signal from **ANALOG_0** on **Main Connector**, which is reduced from 0-36V to 0-3V.

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
15	EXTERNAL_ANALOG_1	OUTPUT	A	External analog signal (0-3V). This is the analog signal from **ANALOG_1** on **Main Connector**, which is reduced from 0-36V to 0-3V.
16	EXTERNAL_ANALOG_2	OUTPUT	A	External analog signal (0-3V). This is the analog signal from **ANALOG_2** on **Main Connector**, which is reduced from 0-36V to 0-3V.
17	EXTERNAL_ANALOG_3	OUTPUT	B	External analog signal (0-3V). This is the analog signal from **ANALOG_3** on **Main Connector**, which is reduced from 0-36V to 0-3V.

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
18	FTC_VOTING_B	OUTPUT	B	This pin is an **open drain output** (0 - 48V), which is open or connected to GND depending on the **FTS signals of the Autopilots 1x**. This logic is implemented at the **Voting Stage**, explained in detail in the *Flight Termination System* section of this manual. Use this pin for an emergency device; for example the ground of a relay that activates a parachute. FTC_VOTING_A (pin 53) and FTC_VOTING_B (pin 18) do the same function, but they have dissimilarity.
19	EXT_RS232_TX	INPUT	A	In case of employing an additional external autopilot, its pin **RS 232 TX** must be connected here. If arbiter decides to multiplex this signal, it will be transmitted to **MUXED_RS232_TX** on **Main Connector** with RS232 protocol.

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
20	EXT_RS232_RX	OUTPUT	A	In case of employing an additional external autopilot, its pin **RS 232 RX** must be connected here. If arbiter decides to multiplex this signal, it will be transmitted to **MUXED_RS232_TX** on **Main Connector** with RS232 protocol.
21	IN_RS485_P	OUTPUT		This pin is connected with **IN_RS485_P** from **Main Connector**
22	IN_RS485_N	OUTPUT		This pin is connected with **IN_RS485_N** from **Main Connector**
23	EXT_OUT_RS485_P	INPUT	B	In case of employing an additional external autopilot, its pin **OUT_RS485_P** must be connected here. If arbiter decides to multiplex this signal and **EXT_DETECT** of **Arbiter Connector** is connected to GND, it will be transmitted to **OUT_RS485_P** on **Main Connector** with RS232 protocol

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
24	EXT_OUT_RS485_N	INPUT	B	In case of employing an additional external autopilot, its pin **OUT_RS485_N** must be connected here. If arbiter decides to multiplex this signal and **EXT_DETECT** of **Arbiter Connector** is connected to GND, it will be transmitted to **OUT_RS485_N** on **Main Connector** with RS232 protocol
25	CANA_P_ARB_A	I/O		This pin is connected with **CANA_ARB_P** from **Main Connector**
26	CANA_N_ARB_A	I/O		This pin is connected with **CANA_ARB_N** from **Main Connector**
27	`1.2:` VCC1	`1.2:` POWER		`1.2:` Veronte 1 power supply (6.5 to 36 V)
	`1.8:` ARBITER_ANALOG_1	`1.8:` INPUT		`1.8:` Arbiter analog input (0-36V)
28	CANB_P_ARB_B	I/O		This pin is connected with **CANB_ARB_P** from **Main Connector**
29	CANB_N_ARB_B	I/O		This pin is connected with **CANB_ARB_N** from **Main Connector**
30	OUT_RS485_ARB_P	OUTPUT	A	Non-inverted output for arbiter A RS-485 bus
31	OUT_RS485_ARB_N	OUTPUT	A	Inverted output for arbiter A RS-485 bus

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
32	IN_RS485_ARB_N	INPUT	A	Inverted output for arbiter A RS-485 bus
33	IN_RS485_ARB_P	INPUT	A	Non-inverted input for arbiter A RS-485 bus
34	TX_OUT_P	OUTPUT	A	Arbiter A ARINC positive output
35	TX_OUT_N	OUTPUT	A	Arbiter A ARINC negative output
36	RIN1_ARINC_P	INPUT	A	Arbiter A ARINC positive input
37	RIN1_ARINC_N	INPUT	A	Arbiter A ARINC negative input
38	GND*	GROUND		Ground pin for buses
39	SCL_A_OUT_ARB	OUTPUT	A	Clock signal for arbiter A I2C bus
40	SDA_A_OUT_ARB	I/O	A	Data signal for arbiter A I2C bus
41	DSP_232_RX_B	INPUT	A	Arbiter A RS-232 input B
42	DSP_232_TX_B	OUTPUT	A	Arbiter A RS-232 output B
43	DSP_232_RX_A	INPUT	A	Arbiter A RS-232 input A
44	DSP_232_TX_A	OUTPUT	A	Arbiter A RS-232 output A
45	GND*	GROUND		Ground pin for analog signals
46	ARBITER_ANALOG_0	INPUT	A	Arbiter A analog input (0-36V)
47	ARBITER_ANALOG_1	INPUT	A	
48	ARBITER_ANALOG_2	INPUT	A	
49	ARBITER_ANALOG_3	INPUT	A	
50	ARBITER_ANALOG_4	INPUT	A	
51	ARBITER_ANALOG_5	INPUT	A	
52	ARBITER_ANALOG_6	INPUT	A	

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
53	FTC_VOTING_A	OUTPUT	A	This pin is an open drain output (0 - 48V), which is open or connected to GND depending on the **FTS signals of the Autopilots 1x**. This logic is implemented at the **Voting Stage**, explained in detail in *the Flight Termination System* section of this manual. Use this pin for an emergency device; for example the ground of a relay that activates a parachute. FTC_VOTING_A (pin 53) and FTC_VOTING_B (pin 18) do the same function, but they have dissimilarity.
54	GPIO_8_ARB	I/O	A	Arbiter A PWM / digital I/O signal (0-3.3V)
55	GPIO_9_ARB	I/O	A	
56	WD_EXT	INPUT	A	Watchdog signal from external autopilot to arbiter A (0-3.3V)
57	EXT_DETECT	INPUT	A	Connect to **GND** if external autopilot is connected, otherwise keep open
58	GND*	GROUND		Ground signal for GPIO
59	GPIO_0_ARB	I/O	A	Arbiter A PWM / digital I/O signal (0-3.3V)
60	GPIO_1_ARB	I/O	A	
61	GPIO_2_ARB	I/O	A	
62	GPIO_3_ARB	I/O	A	
63	GPIO_4_ARB	I/O	A	
64	`1.2:` ARB_GPIO5	`1.2:` I/O	A	`1.2:` Arbiter´s PWM/Digital Output/Digital Input signal (0-3.3V)

continues on next page

Table 2 – continued from previous page

PIN	Signal	Type	Internal Power Domain	Description
	1.8: SEL_AP	1.8: OUTPUT	A	1.8: CAP signal indicating the AP selected
65	1.2: ARB_GPIO6	1.2: I/O	A	1.2: Arbiter´s PWM/Digital Output/Digital Input signal (0-3.3V)
	1.8: FTS1_OUT	1.8: OUTPUT	A	1.8: Deadman signal from comicro
66	1.2: ARB_GPIO7	1.2: I/O	A	1.2: Arbiter´s PWM/Digital Output/Digital Input signal (0-3.3V)
	1.8: FTS2_OUT	1.8: OUTPUT	A	1.8: System OK bit
67	GND*	GROUND		Management Board ground
68	VCC_ARBITER	POWER		Power supply for Management Board (6.5 to 36 V)

Warning: Common grounds are marked with *.

CHAPTER

TWELVE

ACRONYMS AND DEFINITIONS

12.1 Acronyms

16 VAR	16 Bits variables (Integers)
32 VAR	32 Bits variables (Reals)
ADC	Analog to Digital Converter
AGL	Above Ground Level
AoA	Angle of Attack
ARC	Arcade Mode
AUTO	Automatic Mode
BIT	Bit Variables
CAN	Controller Area Network
CAP	Capture Module
CMB	CliMb Phase
CRU	CRuise Phase
DAA	Detect And Avoid
DC	Direct Current
DGPS	Differential GPS
ECAP	Enhanced CAP
ECEF	Earth Centered – Earth Fixed
EGNOS	European Geostationary Navigation Overlay Service
EKF	Extended Kalman Filter
FCS	Flight Control System
FHSS	Frequency Hopping Spread Spectrum
FLR	Flare Phase
FTS	Flight Termination System
GIS	Geographical Information System
GND	Ground
GNSS	Global Navigation Satellite Systems
GPIO	General Purpose Input Output
GPS	Global Positioning System
GS	Ground Speed
GS	Ground Segment
HLD	Hold Phase
HUM	Hardware User Manual
I2C	Inter-Integrated Circuit
IAS	Indicated Air Speed
ID	Identification

continues on next page

Table 1 – continued from previous page

ISM	Industrial Scientific and Medical
LED	Light-Emitting Diode
LND	Landing Phase
M2M	Machine To Machine
MSL	Mean Sea Level
NC	No Connect
OAT	Outside Air Temperature
PFD	Primary Flight Display
PID	Proportional Integral Derivative
PPM	Pulse Position Modulation
PWM	Pulse Width Modulation
QNH	Barometric atmospheric pressure adjusted to sea level
RC	Radio Control Mode
RF	Radio Frequency
RPAS	Remotely Piloted Aircraft System
RPM	Revolutions Per Minute
RS 232	Recommended Standard 232
RS 485	Recommended Standard 485
RX	Reception
SMA	SubMiniature Version A Connector
SSMA	Miniature-Sized Connector
STB	Standby Phase
SU	Servo-Output matrix
SUM	Software User Manual
TAS	True Air Speed
TKO	TakeOff Phase
TPDR	TransPonDeR
TX	Transmission
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
US	Output-Servo matrix
VTOL	Vertical TakeOff and Landing
WGS 84	World Geodetic System 84
WP	Waypoint

12.2 Definitions

- **Control Phase:** The operation is divided into phases in which the UAV has a specific performance. Each of this phases is called a control phase.
- **Control Channel:** It is each of the signals used to control a behavior or action.
- **Control Mode:** It is possible to make a manual control of the UAV by stick, assisted control and fully automatic control.
- **Actuator:** It is a mechanic device to provide force to move or "act" another mechanical device.

CHAPTER

THIRTEEN

CONTACT DATA

For support-related inquiries, customers have access to a dedicated portal through the Joint Collaboration Framework. This platform facilitates communication and ensures traceability of all support requests, helping us to address your needs efficiently.

For other questions or general inquiries, you can reach us via email at sales@embention.com or by phone at (+34) 965 115 421