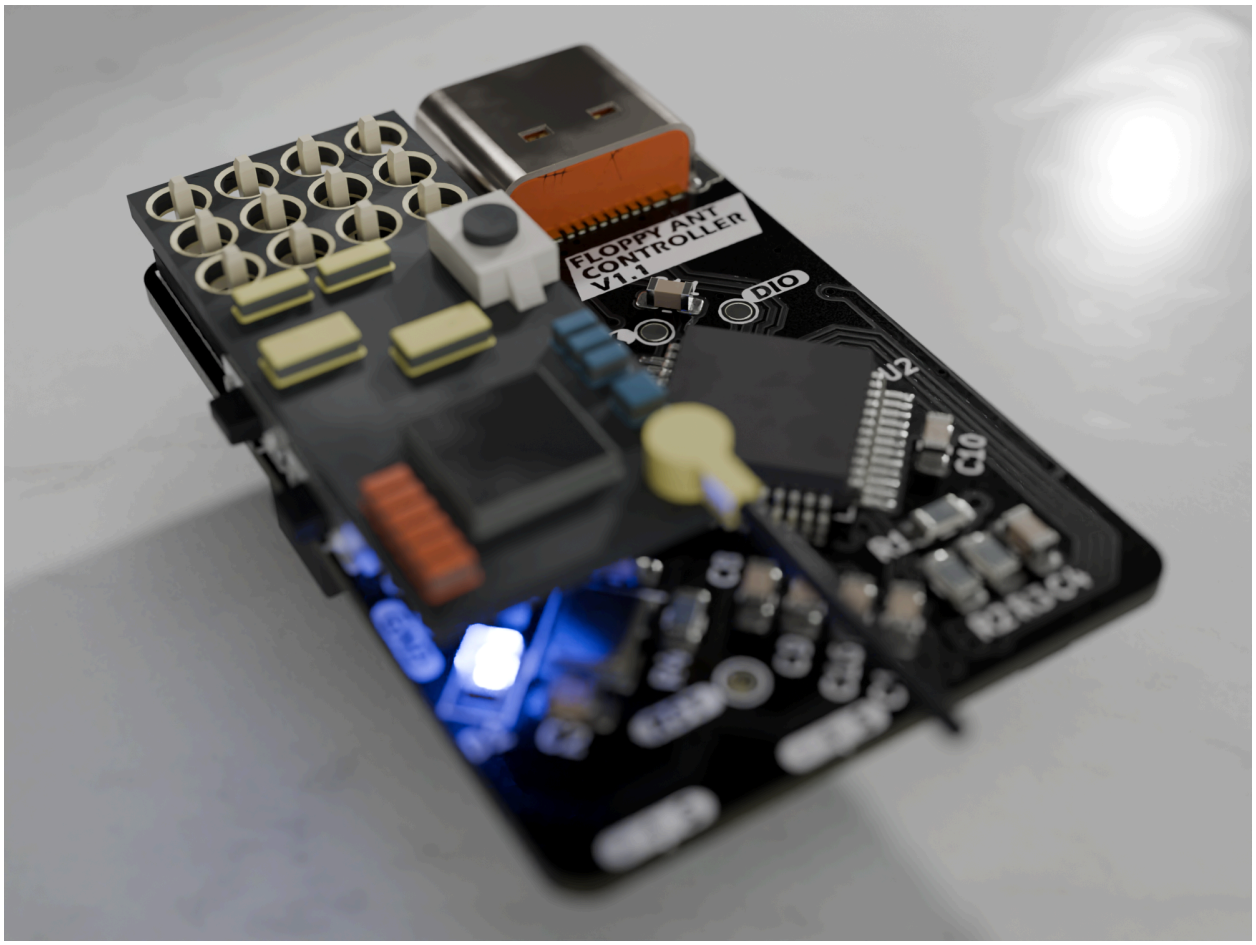


Floppy Ant Controller V1.3

last update 08/2024

—

This is the documentation in which the functions and methods of use of the Floppy Ant Controller (version 1.3) will be explained.

thefloppylab@gmail.com

Floppy Lab

Definitions

1. FAC: Short for Floppy Ant Controller.
2. Tank mix: mixing of the channels to move the wheels with the same logic as the tracks of a tank.
3. BEC: alternative power source, in fact it is a buck converter that lowers the voltage from the battery (2S) down to 5V (3A peak maximum).
4. Pad: where wires can be soldered to make the various electrical connections.

Overview

In this document you will find all the information necessary to use the Floppy Ant Controller version 1.3.

The methods of use are almost the same for all versions, there are only some variations in the settings and the position of the connection points of the external components.

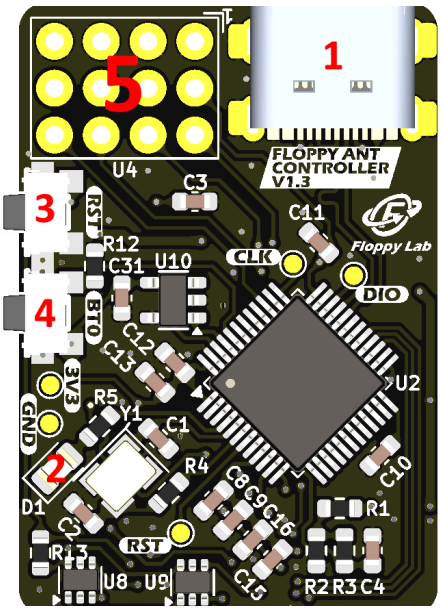
The FAC has been designed to combine perfectly with the FS2A receivers, excellent for being associated with the economical FlySky-i6 and FS-i4 radio controls, so as to also be accessible for those who do not already have more advanced radio systems.
Obviously no one forbids connecting other types of receivers, the important thing is that they have a PPM or PWM output, this limit is currently dictated by the firmware and part of the hardware.

To make it easy to use, a PC program called FAC Settings Tool which allows you to easily modify, export and import all the settings of the Floppy Ant Controller.

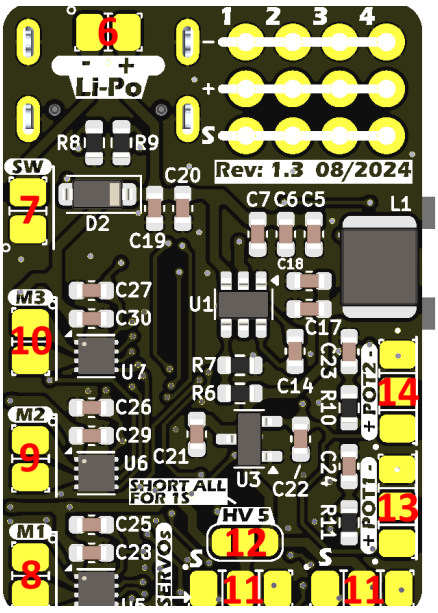
Specifications

- 3x DC motor drivers (200 speed steps).
- 2x servo outputs (one of which can be selected with a solder jumper whether 5V or HV from battery).
- Servo output signal frequency 200Hz.
- BEC 5V, 3A maximum (great for a servo motor).
- 2x expansion ports with possible digital or analog readout (e.g. potentiometers, switches, etc.).
- Support for 1s and 2s batteries.

- Battery voltage reading, with weapon speed limitation threshold and full shutdown threshold (independent).
- Settings can be set via USB with dedicated software or serial terminal.
- Compatibility with 4 PWM channels and 8 PPM channels.
- Dimensions of 23mm*33mm for a weight of 2.9g (without receiver).



Connections, buttons and LEDs



- The USB-C connector[1] allows the connection of the board to the PC to: change the settings, or update the firmware.
- The status LED[2] (D1) indicates what state the FAC is in, therefore whether it is in normal, cutoff or limit operation.
- The RESET buttons[3] e BOOT[4] they are used to start the bootloader and update the FAC firmware.
- The holes[5] which form a 4*3 matrix, are the ones where the receiver must then be soldered.
- In the pad[6] with the Li-Po label, is where the battery cables must be connected.
- In the SW pad[7] the main switch must be connected, it completely disconnect the battery from the rest of the circuit.
- The pads M1[8], M2[9], M3[10] are connected to the DC motor drivers, the outputs have the maximum voltage of the battery, and a supported current of 2A (DRV8838 driver).
- At the bottom are the servo pads[11], the left one allows you to choose the output voltage of the power supply, between 5V (from the BEC) and VBAT (directly from the battery), by soldering the jumper[12] a little higher. The other servo output is only 5V. Both have a 0-5V logic signal.
- Pads POT1[13] and POT2[14] allow you to connect external potentiometers or switches to implement additional functions such as controlling the position of an arm.

Settings and FAC Settings Tool 2.0

The board's functionality can be changed by connecting the FAC via USB-C to your PC or smartphone and changing its settings.

The FAC settings are saved in an EEPROM memory, so once the power is disconnected they remain stored, so it is not necessary every time change them when turned on the FAC.

Settings

Below are all the commands (i.e. settings) and their descriptions in a table.

COMMAND (with default values)	Valid Values	DESCRIPTION
TH2CH>2	1 - 8	Channel assignment to the throttle
ST2CH>4	1 - 8	Channel assignment to steering
WP2CH>3	1 - 8	Channel assignment to the weapon
S12CH>6	1 - 8	Channel assignment to servo output 1
S22CH>1	1 - 8	Channel assignment to servo output 2
ARM2CH>5	1 - 8	Arming channel assignment
M1REV>1	0, 1 [1]	Reverses motor 1
M2REV>1	0, 1 [1]	Reverses motor 2
M3REV>0	0, 1 [1]	Reverses motor 3
S1REV>0	0, 1 [1]	Reverses servo 1
S2REV>1	0, 1 [1]	Reverses servo 2
M1ASS>ML	ML, MR, WP [2]	Assign motor 1 the left motor (DC motor)
M2ASS>MR	ML, MR, WP [2]	Assign motor 2 the right motor (DC motor)
M3ASS>WP	ML, MR, WP [2]	Assign motor 3 to the weapon motor (DC motor)
WPDD>0	0, 1 [1]	The weapon has two directions not just one (DC motor)
TNCON>1	0, 1 [1]	Tank mix activated
NODISARM>0	0, 1 [1]	No disarm channels (automatic arming active)
VOLIM>28 (= 2.8V)	0 - 42	Weapon speed limiting voltage (voltage per cell)
VCUTOFF>35 (= 3.5V)	0 - 42	Total robot deactivation voltage (voltage per cell)
RXPWM>0	0, 1 [1]	Receiver signal is PWM type (otherwise PPM)

1. In commands with valid values 0 and 1, **one** is equivalent to "active", **zero** equals "deactivated".
2. With motor assignment commands (MxASS>xx), you cannot assign two motors (M1, M2, M3) with the same type (ML, MR, WP).

READ	Reads the settings from EEPROM and prints them on the serial monitor (if it is sent it overwrites the changes made)
SAVE	Save all settings in EEPROM
CON	It does nothing and is only used for the FAC Settings Tool

Connection to PC/smartphone with Serial Monitor

Commands can be sent via serial after connecting to the FAC with a serial monitor. This method allows you to set all the values to your liking, though **non** no check is made that the entered value is valid, therefore use this method with caution when changing settings. You can connect via PC with any serial monitor software (such as Termite, or that of Arduino IDE), you just need to connect the USB, select the correct port and a baud rate to your liking (the FAC adapts accordingly). Once connected you can send commands.

The same thing applies to the smartphone, you can download an application that manages serial communication (such as Serial USB Terminal, excellent because it allows you to save predefined commands) and connect to the FAC as described before. If it doesn't recognize your board, it's most likely because the cable/adapter you used is not OTG.

Composition of a command

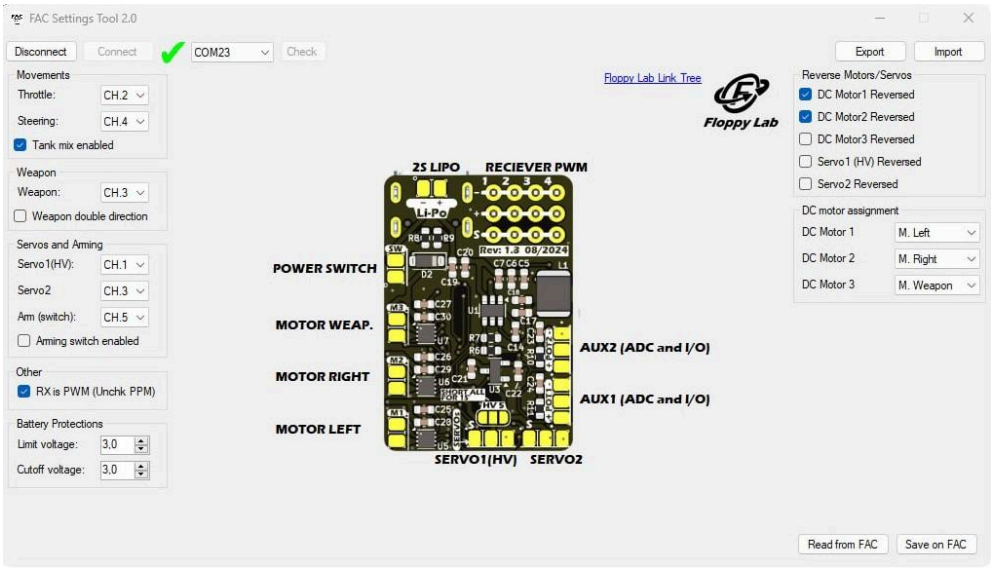
A command is made up of two parts: the first is the type, the second the value. They are separated by the '>' character. For example, if we want the tank mix throttle to be connected to channel 3 of our radio control, we must send: TH2CH>3.

If the FAC understands the command you just sent it responds with “OK”, otherwise with “WHAT??”

To view the settings saved in the FAC EEPROM, you can send the READ command. **Please note that when this command is sent, the settings you have previously changed are overwritten with those read from the EEPROM.**

Once you have changed the settings, to save them simply send the SAVE command and make sure that the OK response appears, which indicates correct receipt of the command.

Changing Settings with “FAC Settings Tool 2.0”



The FAC Settings Tool 2.0 makes changing FAC settings more intuitive and faster.

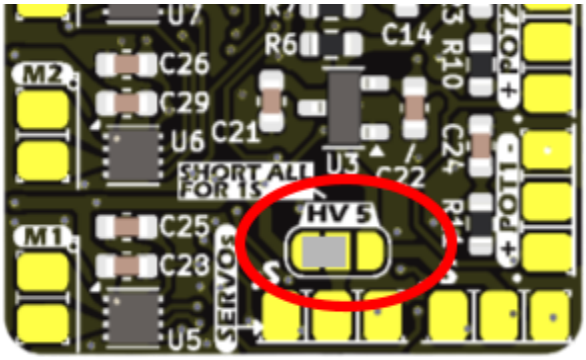
On the sides of the application (currently available only for Windows) there are all the settings previously listed in the table. Furthermore, it allows you to export and import settings in a .csv file, so you can save configurations you create, and if necessary they can be easily shared.

1. Once the tool has been opened and the FAC connected to the PC, click on the "Check" button at the top left of the application, and select the correct port with the drop-down menu.
2. Once selected, press the “Connect” button and wait for the green check to appear next to the button you just pressed.
    a. If a window appears with an error message, it means that the selected port does not correspond to the FAC, but to some other device. In this case select a different port and press “Connect” again.
3. As soon as the FAC connects all the settings are loaded and the graphics are updated.
4. Now you can change the settings to your liking.
5. Once you are satisfied with the setup, at the bottom right you can click the “Save on FAC” button and wait for the saving to complete.
6. If you want to export or import your settings, simply press the “Export” or “Import” buttons and save the file with the name you prefer.

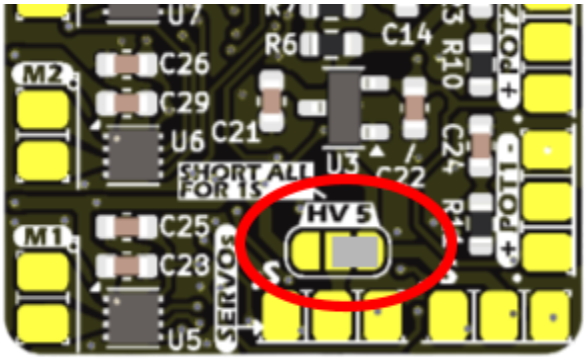
FAC capabilities and limitations

The FAC is designed to be used with 150g robots, so the performance of the DC drivers and BEC is limited.

1. The three drivers for the DC motors are designed to operate up to a voltage of 12V, and a peak current of 1.9A (the driver used is the DRV8838, the same as Malenki nano) so they are great for running N20 motors.
2. The status LED together with the sounds played by the DC motors indicate the status of the board. When the board is working normally, the LED flashes with a period of 1 second. If the board is not armed one of the motors emits a "beep", otherwise no sound is emitted.
3. The card allows you to set arming, i.e. if the robot is not armed all its functions are deactivated (wheels and weapon stopped, servo position 0). If arming via remote control channel is not active (can be activated from settings), the FAC does not arm until the steering and throttle controls are in the center, and the weapon is set to minimum. With this mode, once armed the FAC cannot be disarmed.
   If a dedicated channel is used for arming, the FAC can be disarmed whenever you want.
4. The FAC integrates the battery voltage reading. The reading may be inaccurate due to component tolerances.
   Two (independent) thresholds can be set which are activated 10 seconds after the battery voltage drops below them:
   a. The first (Limit) limits the speed of the weapon (only if DC motor) to 50% of the speed, so as to extend the battery life during the fight.
   b. The second (Cut-Off) completely disables the robot, this threshold is usually very low, and serves to preserve the health of the batteries. When the robot is in Cut-Off the status LED flashes quickly.
5. The BEC is effectively a buck converter, which is designed to withstand a maximum current of 3A (better to stick to a maximum of 2.5A). Therefore, when servo motor outputs are used, I strongly recommend not using two servos powered by the BEC and not connecting ESCs with BLDC motors, as this may result in malfunctions and breakage of the BEC and the rest of the board, making it unusable.
   To overcome this problem, you can use a 5V servo (i.e. from the BEC) and an HV servo by soldering the jumper in the marked part.



HV (from battery)



5V (from BEC)

6. The FAC can be powered at a maximum voltage of 8.4V, therefore a 2S battery. This limitation is due to the maximum voltage of the DC motor drivers, and the battery voltage reading.
7. To use a FAC with a 1S battery, you can short (as marked on the board) the three servo motor voltage selection pads, so the input and output of the BEC are shorted, so the BEC stays off. **Be very careful not to power more than 5V when using this configuration, otherwise the board will burn out.**
8. **Be careful, when the FAC is connected to the PC all systems are also active, including the DC drivers and servo motors.**

More information

The entire project is open source, and you can find all the files on this GitHub page. For any questions or requests do not hesitate to send an email to: thefloppylab@gmail.com.