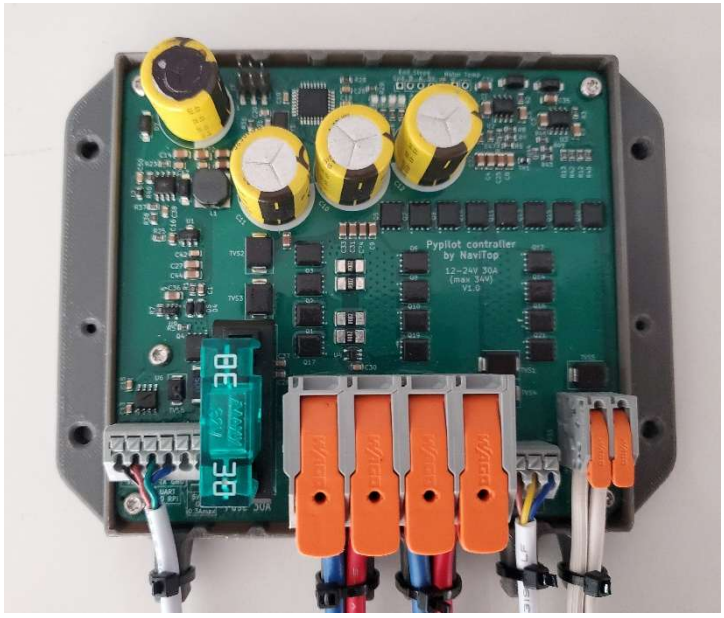
NaviTop

PYPILOT CONTROLLER 30A* WITH CLUTCH OUTPUT BY NAVITOP

NaviTop models: C30 or P30



1 – Presentation and specification

Pypilot was conceived and designed by Sean D'EPAGNIER. Thanks to him for this fantastic boat autopilot.

This motor controller drives autopilot actuators with DC motors and electromagnetic clutches. It must be paired with a suitable autopilot computer such as **Tinypilot**, **OpenPlotter** or BBN running Pypilot.

- Supports 12 or 24 volts (10v to 34v maximum)
- Power input protected against reverse polarity, EMC Surges and a 32V 30A "MAXI" automotive fuse
- All outputs protected by electronic current limitation or polyfuse, including the rudder angle sensor and limit switch power supplies
- With pypilot, once the clutch is engaged, this controller allows to reduce the electrical consumption of the clutch only to what is sufficient to keep it engaged
- With pypilot, this controller allows you to adjust the acceleration/deceleration and speed of the motor
- Software over- current (stall) detection and over-temperature protection
- Voltage, current, temperature and rudder feedback
- Internal electrical resistance < 20mΩ
- Box allowing easy fixing of cables using nylon collars
- Power supply input and motor output with lever connection terminals Wago 66 amps for conductors up to 16mm²
- Clutch output with lever connection terminal Wago 32 amps for conductors up to 4mm² conductors
- Terminal blocks with push button for conductors 0.75 to 1.5mm² (strip length 8.5 to 9.5mm) for connecting:
    - Serial communication to Raspberry Pi (tinypilot or openplotter) with galvanic isolation, prevent ground loops and other electrical problems.
    - output 5V DC 0.3A max to powering tinypilot Pi Zero autopilot computer.
    - rudder feedback sensor

- 4 pins for connection to optional rudder end of stops (limit switches or proximity detectors 5V)
- The controller's 5V DC power supply has been redesigned to be more robust and allow the pypilot computer, with Pi Zero W or Zero 2W, to be powered. All 5V outputs are short-circuit protected
- Intrinsic consumption about 5 mA (60 mW with 12V, 120 mW with 24V)
- PCB marinized with tropicalizing acrylic varnish
- The open-source Arduino software designed by Sean d'Epagnier is flashed. The processor can be reprogrammed using the ISP programming connector on PCB

2 - Dimensions

- Box : 146 x 132 x 50 mm
- PCB : 107 x 117 x 43 mm

3 - Power Supply and Motor Connection - Power Cable Sizing

For the controller's power supply and connection to the motor, use a cable with a sufficient cross-section, whenever possible, so that its electrical resistance does not exceed the internal resistance of the motor controller.

- 3m length between the electrical panel and the motor = 2 x 5mm² (AWG10)
- 4m length between the electrical panel and the motor = 2 x 7mm² (AWG9)
- 6m length between the electrical panel and the motor = 2 x 10mm² (AWG7)

Strip 18 to 20 mm of insulation from the power and motor wires, then:

- Connect the + and – power wires to the WAGO + and – terminals marked "Power".

- Connect the motor wires to the Wago A and B terminals marked "Motor" on the circuit board. The two motor wires must be reversed if the driver corrections are made in the wrong direction.

4 - Clutch output

There are two terminals for powering a solenoid valve for hydraulic actuators, or a mechanically controlled solenoid clutch for other applications.

The output voltage for the clutch is the same as the power supply voltage of the motor controller.

Strip the wires between 9 and 11mm and connect them to the + and – terminals of the Wago terminal block marked “Clutch”.

To reduce power consumption, pypilot energizes the clutch coil at full power for 200-300 ms, then the controller outputs pulse width modulated (PWM) pulses to keep the clutch engaged while reducing power consumption. This can be adjusted with pypilot parameter "servo.clutch_pwm" key (0-100%) accessible in the additional "pypilot client" settings. For a cylinder solenoid valve, a setting of 16% decreases the current by a ratio of 6.25 while keeping the cylinder fully engaged, which reduces consumption from 30Ah per day to 5Ah.

5 - Serial Data Connection

Connection to Raspberry Pi (tinypilot, openplotter or BBN) via 4 Wago push-button terminal block contacts for conductors up to 1.5 mm² (strip length 8.5 to 9.5mm).

Wago terminal	Function	To raspberry pin
VCC	+3.3v	1
RX	Rx (to Tx)	8
TX	Tx (to Rx)	10
GND	0v, GND	6, 9

It is recommended to use telephone or Ethernet cable directly connected to the WAGO push-in terminals. This cable can be extended by more than 30 meters.

The controller's serial input and output are galvanically isolated to prevent ground loops and other electrical problems. The + wire powers the controller's galvanic decoupling circuit, which can be powered from +3V to +5.5V. The current consumption, in the order of mA, allows the use of a fine-wire cable. To facilitate connection to the autopilot computer, a 4-pin waterproof connector is provided and connected to the controller.

The serial cable can be extended by more than 30 meters with telephone or Ethernet cable. If you are particularly concerned about interference, you could use a shielded cable, but this is rarely, if ever, a concern in practice.

If you connect the motor controller directly to a Pi, we recommend installing TVS diodes on the Pi side to limit potential surges that could be induced by a nearby lightning strike, especially if this cable is more than a meter long (see diagram of a simple tinypilot in the documentation tab of www.navitop.fr).

6 - 5V DC output to power the Pi Zero autopilot

5V DC output (0.3A max) to power computer equipped with a Pi Zero W or Pi Zero 2W.

All Navitop computers (simple or with remote control) are equipped with a 5V DC power cable. With this 30A controller, you must strip the red and black wires of the cable by 9mm and insert them into the two push-in terminals + and – 5V DC terminals, near the four UART communication terminals with the Raspberry Pi.

If the autopilot computer has'nt power cable 5V, it is recommended to use a cable equipped with a microUSB connector that connects directly to the Pi Zero.

7 - Rudder travel limiter switches

With a powerful actuator, installing limit switches to stop the motor before it forces the mechanical rudder stops is the most reliable solution, even though these switches are optional with pypilot. They will be more reliable than a poorly calibrated rudder angle sensor whose rod can bend or detach. They are essential with certain hydraulic actuators for which current limiting with the servo.max_current setting does not effectively limit the force.

GND is the common wire for both switches. Connecting wire A or wire B to GND prevents any motor movement in the corresponding direction.

The connection on the PCB is labeled "End Stops." The square pad is GND, the second is End B, the third is End A, and the last pad is the +5V power supply, used only if you are installing proximity switches or Hall effect logic sensors requiring a 5V power supply. In all cases, limit switches must be installed so that the electrical contact remains closed beyond their engagement threshold, preventing the motor from starting with the rudder beyond the adjustment threshold.

After connection, test to ensure that when the motor rotates in one direction, the limit switch corresponding to that direction stops the motor. If this is not the case, the End A and End B wires must be reversed.

Upon request, we can supply and solder a 3 or 4 pin waterproof connector for the optional limit switches (+5V = brown, End A = green, End B = blue, GND = black).

8 - Motor Current Measurement and Limiting

All pypilot controllers are equipped with a motor current measurement function that allows you to determine the current drawn by the motor in amps using the servo.current variable, visible in the pypilot client (configuration page) or with the pypilot_scope script.

Important: The pypilot servo.max_current parameter must be set appropriately for the motor and the controller installed on the boat. Pypilot detects the end of travel by stopping the motor in one direction when the measured current (servo.current) exceeds the pypilot servo.max_current setting. If this current limiting setting is set too low, "OVERCURRENT_FAULT" messages will constantly appear and block the actuator from moving. If the value is too high, the actuator could continue to output power even when the actuator is at its limit.

For tiller actuators, a setting of 3.5 to 5 amps is generally correct. For powerful hydraulic drives, higher values such as 12 to 20 amps should be used.

Using the pypilot_scope script can be useful to know the actual consumption curve of the motor during operation and at full throttle.

9 - Rudder Feedback sensor

First, with pypilot, the rudder feedback is optional. It can be disconnected while underway and pypilot will continue to steer. It is generally used to report the rudder angle on a display and avoid relying on end-of-travel stop by intensity limitation. It also may be used by certain pilot algorithms to enhance steering, but the basic pilot algorithm does not require it. To be clear, corrections needed in moderate conditions are 10 or more times that of the errors due to integration from not knowing the rudder position, so the potential improvement in steering performance from rudder feedback is not huge.

A potentiometer with 3 wires can be connected to the controller. The potentiometer should range from 1k ohms to 100k ohms. Recommend 10k. You can also connect a Hall effect sensor 5v with analog output to these 3 wires.

The rudder angle sensor is connected via 3 push-button terminal block contacts for 1.5 mm² conductors labelled “rudder”. The first pin is +5V for the sensor power supply, the middle V pin is for measurement and the last pin is GND.

It is not critical that the voltage increases or decrease with rudder angle as the rudder feedback calibration takes care of the direction.

If connecting an existing rudder sensor with only two wires, then a 1kΩ resistor must be added between the 5V pin and the measurement pin.

When the rudder feedback sensor is installed, you can check the rudder calibration page to read the value and make sure it is working.

The rudder feedback must be calibrated. You must manually turn the rudder to port range, starboard range, and center and press each button for each position. The order is not important, but once all 3 operations are complete, the scale, offset, and nonlinearity must be calculated. The "Rudder Range" field must be manually set to indicate the actual angle at each range position and to limit the autopilot movement beyond that position. It is possible to set the "rudder range" to say 35 degrees and calibrate the rudder by moving it to 35 degrees in each direction and later set it to 30 degrees to further constrain the range the autopilot can move the rudder. So, to be clear, the "rudder range" is for calibration and whatever the value is when the button is pressed, but in operation it specifies the maximum angle the motor controller can move the rudder to.

Note: It should also always be remembered that using the rudder angle sensor to limit the actuator travel may disable the autopilot if the rod connecting the rudder is accidentally bent or disconnected. Properly installed limit switches or a good adjustment of the current limiter can often limit rudder loads more reliably.

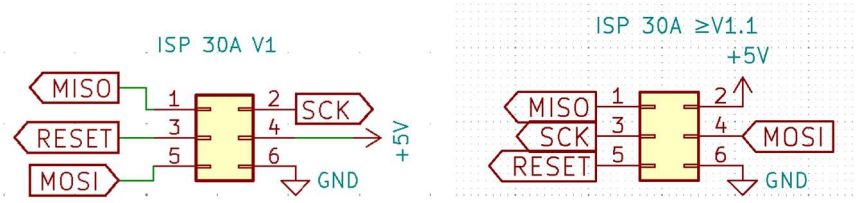
10 - Motor Temperature Sensor

Optional 10k NTC (2 wires) for temperature of the electric motor.

This is generally not needed because most motors will not overheat unless stalled for prolonged periods. It can be used to prevent the motor from overheating and burning out.

11 - ISP connector

Important: The ISP programming connector in version V1 is special. The ISP connector is normal from version V1.1



12 – Navitop 12-24V 30A Motor Controller Designations

- C30 : Pypilot motor controller 12-24V 30A by Navitop with case
- P30 : Pypilot motor controller 12-24V 30A by Navitop without case