



Hyperion Pixhack Setup and Installation guide



Pixhack is a flight controller, based on Pixhawk hardware architecture platform, designed and produced by CUAV. As the main architectural hardware the same with pix, so fully compatible with the 3DR APM firmware and pix native firmware.

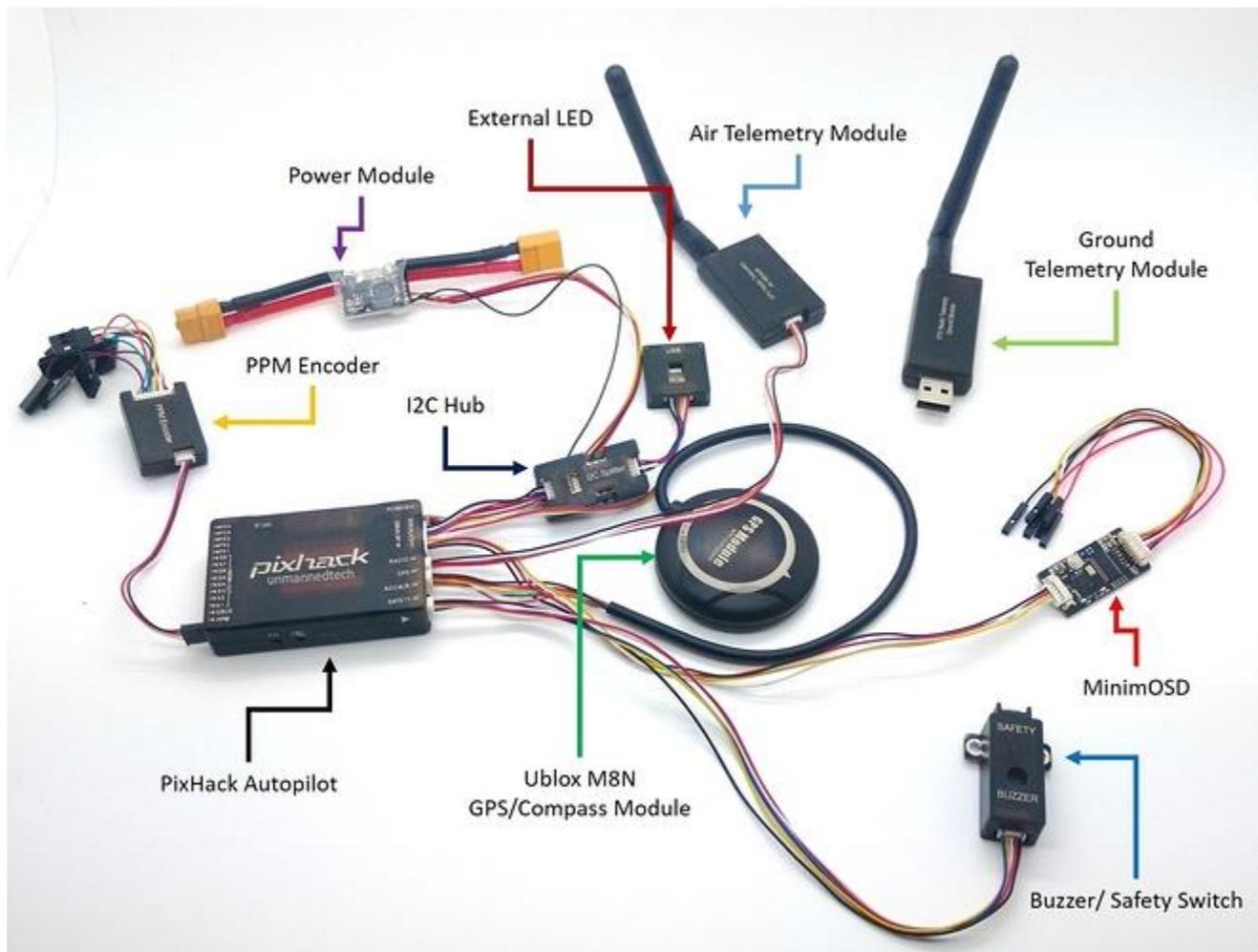
Based on original Pix, Pixhack optimized the power supply chip, reduced unnecessary interface, adjusted and optimized the interface and changed the plug wire direction.

The highlights of Pixhack is the IMU (**inertial measurement unit**) separation, an innovative design feature, built-in small general damping structure adopting the CNC milling technique. Resulting in qualitative improvement of anti-interference and stability.

After a year's design and countless versions of optimization and testing, Pixhack has reached its goal of giving a stable and ideal platform.

Pixhack Overview

The great thing about the Pixhack autopilot kit that we sell, is that it includes all the accessories that you could possibly need. Each of these include a pixhack compatible connector. You can still use other accessories designed for the original pixhawk, but you will most likely need to change the connector as the pixhack does not use DF13 connectors common with pixhawk modules.



The connectors are all clearly labeled and can only go in one way so you don't need to worry about connecting something upside down. In terms of installing the firmware such as arducopter, you will just need to follow the same steps as you would for the Pixhawk/PX4 autopilot. Simply install mission planner and install the firmware onto the board, for full details please refer to the pixhawk wiki.

Introduction to the Pixhack Hardware:

The processor:

1. 32-bit 2M flash STM32F427 Cortex M4, With hardware floating-point processing unit. Frequency: 168MHZ, 256K RAM
2. 32-bit STM32F103 backup co-processor.

Built-in sensors:

Pixhack use an innovative design of separation. Built-in damping feature allows for installation in general aircraft frames and don't need to do shock treatment (if the vibration is too large and with high frequency then only it is recommended to do damping)

- 1, L3GD20 three-axis digital 16bit gyroscope
- 2, LSM303D three-axis 14bit accelerometer/ magnetometer
- 3, MPU6000 6 axis accelerometer/magnetometer
- 4, MS5611 High precision barometer

Working environment and voltage:

Temperature range: - 5 ~ 50 degrees

PM sensor, operating voltage: 2-6S

PM sensor output voltage: 5.4V 3A

PWM OUT input power supply voltage: 9V Highest (support high pressure actuator, and the original Pixhawk does not support the power supply above 5.5V) Two-way power supply automatic redundancy (PM interface and PWM OUT interface), PM interface preferred power supply, and malfunction automatically switch to the PWM OUT interface power supply.

Physical Dimensions:

Main flight controller: 68mmX44mmX15MM

Weight: 68g

Data interface:

- 1, Five UART, one compatible with high voltage, two with the hardware flow control.
- 2, One CAN bus (Controller Area Network).
- 3, Spektrum DSM/DSM2/DSM-X Satellite receiver compatible with the input
- 4, Futaba SBUS compatible with the input and output
- 5, PPM signal input
6. RSSI (PWM or voltage) input
- 7,I2C Protocol device extension
- 8, Reserve SPI interface
- 9, 3.3 and 6.6 VADC input
- 10, External MICRO USB interface

11,13 PWM/Actuator output

12, Multi tone buzzer and unlocking button state, LED interface

Please read notes below to understand the types of interfaces and terms used on Pixhack.

Note:

- Pronounced **u-art**, and short for **universal asynchronous receiver-transmitter**, the UART is a computer component that handles asynchronous serial communication. Every computer contains a UART to manage the serial ports, and some internal modems have their own UART.
- A Controller Area Network (CAN bus) is a vehicle bus standard designed to allow microcontrollers and devices to communicate with each other in applications without a host. It is a message-based protocol, designed originally for multiplex electrical wiring within automobiles, but is also used in many other contexts.
- The innovative Futaba S.Bus system lets you unleash your flight system's full potential and cut down on cable clutter at the same time. It uses digital serial data communication technology to transmit control signals between your receiver and servos.
- **Spektrum** is a brand of radio control systems designed for use with hobby [radio-controlled cars](#) and [aircraft](#). Spektrum is a division of [Horizon Hobby](#). Spektrum systems are distinguished in that they use [direct-sequence spread spectrum](#) (DSSS) technology on the 2.4 GHz [ISM band](#). Spektrum refers to their technology as "Digital Spectrum Modulation." Each transmitter has a [globally unique identifier](#) (GUID), to which receivers can be bound, ensuring that no transmitter will interfere with other nearby Spektrum DSMx systems.
- PWM and PPM are two common words used in the R/C industry. PWM stands for Pulse Width Modulation and PPM stands for Pulse Position Modulation. Some devices that use PWM for control are ESC's (electronic speed controls) and servos. PWM is a technique used to relay data in the form of a varying pulse width. For more details please see <http://www.endurance-rc.com/ppmtut.php>.
- The Received Signal Strength Indicator (RSSI) is a measure of the RF power input to the transceiver. The RSSI value is based on the gain setting in the RX chain and the measured signal level in the channel.
- **I²C (Inter-Integrated Circuit)**, pronounced *I-squared-C*, is a slave, single, [serial computer bus](#) invented by [Philips Semiconductor](#) (now [NXP Semiconductors](#)). It is typically used for attaching lower-speed peripheral [ICs](#) to processors and [microcontrollers](#) in short-distance, intra-board communication. Alternatively I²C is spelled *I2C* (pronounced *I-two-C*) or *IIC* (pronounced *I-I-C*).
- The **Serial Peripheral Interface (SPI) bus** is a [synchronous serial communication](#) interface specification used for short distance communication, primarily in [embedded systems](#).

Peripheral device:

1, Support Platform types: Fixed wing, Multicopter, Helicopter, Car and Boat.

2, Supported receiver types: S-Bus, DSM2, PPM. (For common PWM receiver, you need a PPM encoder).

3, Recommend the battery type: 2S ~ 6S lithium polymer battery

4, System configuration requirements: Windows XP SP3/7/8 (32 or 64 bit)

5, Support of external CUAV products: MINIM OSD, CUAV 3DR Radio, DIGI XTend, CUAV RTB BOX, MAAT tracking antenna, Optical flow sensor.

Interface introduction:



Autopilot Connectors

On the left:

- PPM- for SBUS pin out
- SBUS- for SBUS output
- MAIN OUT- for main output ports S1-S8
- FS1-FS5- for extension ports

The main outputs S1-S8 correspond to pixhawk RC1-RC8 pins. The extra outputs FS1-FS5 correspond to pixhawk RC 9-RC13 outputs.

On the right:

- POWER- PM Voltage current module port
- DBUS OP - data extension
- SERIAL4/5- Serial port 4 and 5 extension
- RADIO-data transmission or OSD extension telem1
- GPS - GPS extensions contain electronic compass
- ADC&JR- ADC 3V3 Output and JR, the remote control input extension
- SAFETY- unlock switch and safety button

Warning: Do not connect the PM sensor to other port!

Pixhack: Interface detailed definitions.

SAFETY: Buzzer and unlock switch, interface

ADC&JR: Used to extend JR receiver and previous analog sensors, such as the airspeed and ultrasound

GPS: Used to extend the GPS, built-in I2C definition, all the GPS and compass only need to insert one cable

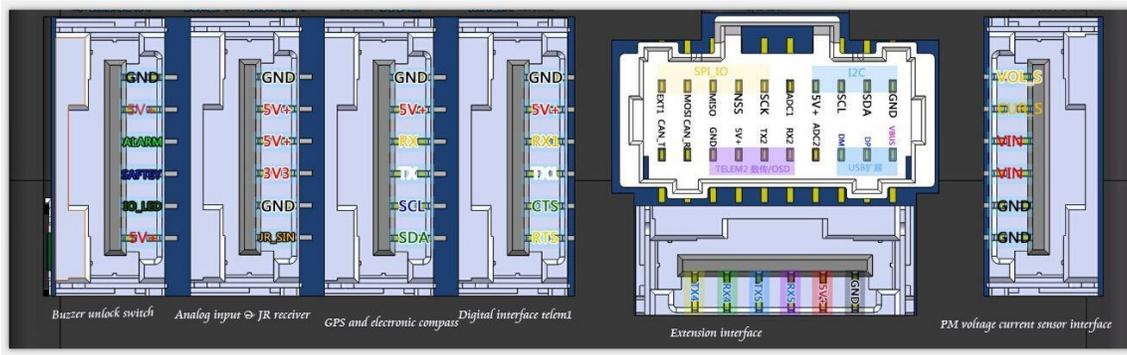
RADIO: 57600 baud rate, Mavlink protocol, Can pick up data transmission or mavlink OSD

DBUS OP: The data bus extension, Specific definition see below, For example you can extend: USB, I2C, SPI, CAN, SERIAL2 etc.

SERIAL4/5: An interface containing two serial ports, SERIAL5 can external the second GPS, SERIAL4 can DEBUG output (according to the software version, some firmware has no DEBUG output)

POWER: Voltage current sensor PM interface, please note that don't connect the PM sensor to other interface, or it will cause short circuit leading to in operation of Pixhack.

Here is the exact pinout of the interface connectors



LED and buzzer meaning:

- The red and blue lights flash: initialization. Please wait a moment.
- Yellow light double flash: errors. System refused to unlock.
- Blue lights flash: already locked, GPS searching the star. Automatic navigation, hover, and return to the starting point mode need GPS lock.

- Green light flash: Has been locked, GPS lock have been obtained. Ready to unlock. From the locked state to unlock, there will be a quick reminder with two sounds.
- Green light long light with single long sound: GPS lock and unlock. Ready to take off!
- Yellow light flash: remote fault protection is activated.
- Yellow light flash with fast repeated sound: battery fault protection is activated
- Yellow light blue lights flash with high - high - high - low sound: GPS data anomalies or GPS fault protection is activated.

Pixhack safety switch meaning (green LED):

- Fast, continuous flash: execution system self-check. Please wait a moment.
- Intermittent flash: System is ready. Please press the safety switch button to activate the system.
- Often bright: ready to unlock. Can perform the unlock procedure.

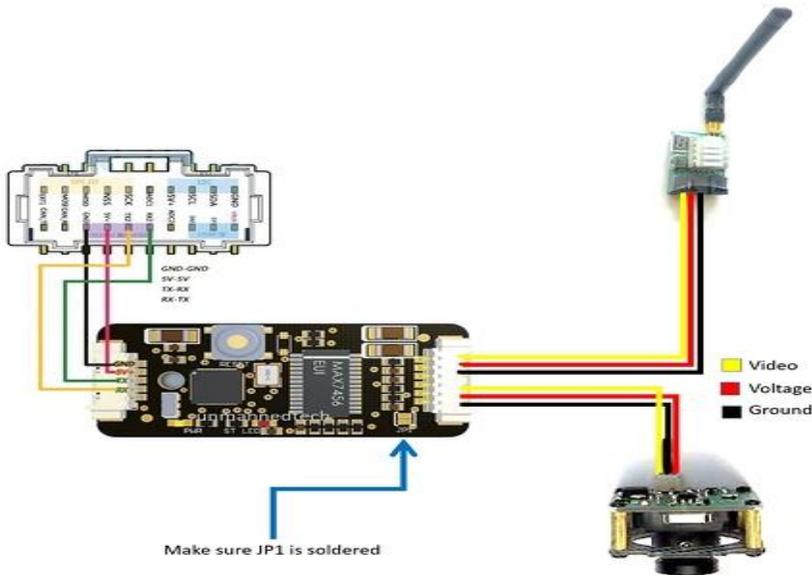
Connecting your receiver

Just like the original Pixhawk, you need to use a RC receiver that is able to output PPM/SBUS signals for the autopilot to read. Either way the Pixhack reads the PPM input from your receiver via the PPM port. Many users get confused between the SBUS and PPM ports on the board, the SBUS is purely used for output so you should always connect your receiver to the PPM port.

If you want to find out more about PPM, PWM check out: www.Hyperion-world.com/help.

However if you are using a R/C receiver that only supports PWM, the Pixhack kit we sell also includes a PPM encoder that will convert each of the PWM channels to a single PPM signal that will be read by the autopilot as shown below. The PPM encoder supports a total of 8 PWM channels.

minimod board to your FPV transmitter/receiver according to the pinout as shown below. The MinimOSD then connect to the Telem 2 port (DBUS connector) on the Pixhack.



You will also need to change some settings within mission planner to enable the MinimOSD board, to get here you will need to open mission planner, and go to the `config>tuning` tab, and view the full parameter list. You will need to change all the `SR2_EXT` parameters to 2.

SR2_EXTRA1	2
SR2_EXTRA2	2
SR2_EXTRA3	2
SR2_PARAMS	0
SR2_POSITION	2
SR2_RAW_CTRL	2
SR2_RAW_SENS	2
SR2_RC_CHAN	2

Powering the output rail

The output rail on the autopilot is powered separately from the main pixhawk (which is powered via the power module). This does not really matter for most multirotor applications since ESCs are powered via the PDB so the pixhawk does not need to provide output power to the ESC's. However if you want to use a servo you will need to power the output rail with a 5V BEC. In most fixed wing aircraft the ESC includes a BEC so you can

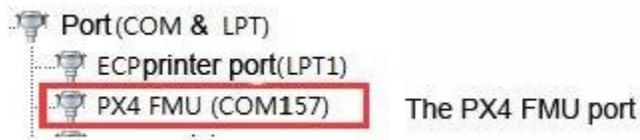
use your ESC to power the entire output rail. Just be careful to ensure your BEC can provide enough current to power all of your servos.

Mission Planner the classic APM ground station software.

APM Planner 2.0 cross-platform ground station, choose the Windows version from <http://ardupilot.com/downloads/?did=82>

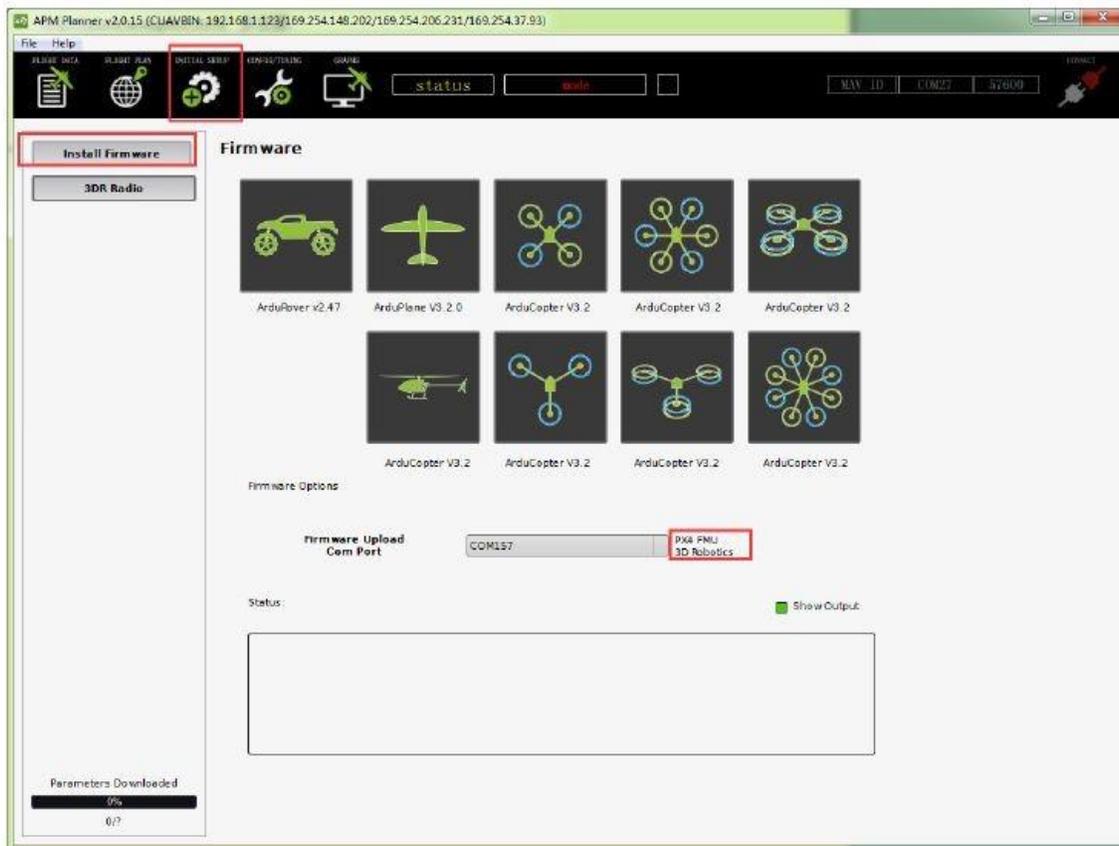
Download the firmware to the flight control:

APM Planner 2.0, for example, install the ground station, insert pixhack, should appear



Open MP 2.0 ground station

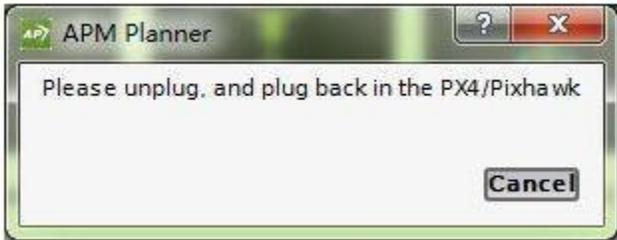
Ground station will automatically detect flight control ports, and display the version of the hardware



Then click on the corresponding icon, type of platform you intend to use.

Note: cross model and X model, are the same firmware, after download, you can set the corresponding flight mode (the default is X mode)

After clicking firmware will be automatically downloaded from the Internet.



You can Click on the FMU reset or reinsert pixhawk to burn firmware to the flight control and then automatically Erasing the board flight control firmware, finally prompting PX4 flashing is complete, burn out!

Tip: Because the firmware is on server and may fail to download. You can try to use QGC ground station to burn the firmware locally. For methods refer to another document: "Burn local firmware"

Flight Components:

A basic flight system must contain the following parts:

- 1, Main controller: pixhawk flight control.
- 2, The remote control: with 7 or more channels like futaba or JR with PPM receiver.
You will to connect the PWM switch to the PPM encoder. (JR compatibility to be tested)
- 3, GPS and electronic compass
- 4, Wireless digital, recommend the XBEE or 3DR Digital, Best choose XTEND
- 5, PM voltage current sensor built-in BEC
- 6, The complete aircraft contains the aerial carrier, motor, ESC, propeller, etc. . .

Optional OSD video overlay system (AV camera + OSD + figure digital), MAAT tracking antenna, etc.

For detail installation process, Please refer to the corresponding versions on website.

Main website <http://ardupilot.org>
Fixed wing <http://ardupilot.org/plane/index.html>
Multicopter <http://ardupilot.org/copter/index.html>

<http://px4.io/user-guide>