

Pilot Hobbies Scout32 Quick Start Guide

Updates

Version 1.9 – Dependencies for Libraries have been updated. Please use ESP32 Boards Version 1.0.6+

Version 2.0 – Added MX1508 motor driver support. Added in-app camera rotation. Now includes proper failsafe! Checks every 4 seconds for active connection to controls, and halts if it doesn't receive a valid signal.

Resources

Instructional Build Video with 1S LiPo Power: <https://youtu.be/9PARUXjCns4>

Programming Video – Do not use old board library: <https://youtu.be/ZkOrh7mL2zU>

Project Overview: https://youtu.be/kd5W6zags_w

Videos for MX1508 Variation and Double Sized ScoutXL Builds coming soon!

Please check my YouTube channel and subscribe for updates: <https://www.youtube.com/pilotgeek>

Parts Selection for Building:

The links below are sponsored promotions through the Amazon Affiliate program. Purchasing parts through these links ensures part compatibility, and greatly helps me to fund further products!

Depending on what you want to build, you may have to pick and choose parts!

Tracks & Chassis

Tracks are printed in TPU. I use a fairly stiff 95a shore hardness. You may be able to use softer TPU with varying levels of success, but most printers should handle the 95A without issue. I use PLA for the chassis, and have had great quality prints with the brands below.

- [SainSmart Black TPU](#)
- [ESUN PLA](#)

Electronics, Motors & Hardware

- [ESP32-CAM - Includes ESP32-CAM and Programmer](#)
- N20 Gear Motor with MX1508 Build
 - [2x N20 Gear Motors](#)
 - [MX1508 Motor Driver](#)
 - [M4x20mm Screws](#) – Axle Screws
 - [3x6mm Grub Screw](#) for Drive Wheels
- Servo Build
 - [2x Continuous Rotation Servos](#)
 - [M4x20mm Screws](#) – Axle Screws
- ScoutXL Build
 - [2x 25GA-370 500rpm Gear Motors](#)
 - [M5 x 40mm Screws](#) For Axles
 - [5x11x4mm Ball Bearings](#) for Idlers

- [3x6mm Grub Screw](#) for Drive Wheels
- [2x Electronic Speed Control without Brake](#)
- [3mm Steel Rod](#) to join Chassis Halves
- [4cm x 6cm PCB](#) for mounting ESP32
- [USB-C C Plug](#) Optional – For Powering Internal Charger Circuit
- [USB 2s Charger](#) Optional – I removed the USB A plug and wired to the USB C port.
- Optional
 - [Wide Angle Lens](#)
 - [5mm White LEDs \(Requires Resistors in Series!\) - 47 Ohm Resistors](#) OR
 - [10mm White LEDs](#) for ScoutXL

Power Options

1s Lipo Power with Charger

Optional parts for 1s LiPo power. Onboard charging is optional, but a convenient feature. Others have also had success powering their Scout via small USB power banks to provide power.

- [1s Micro USB Charger](#)
- [1s to 5v Booster \(Adjustable\)](#) OR
- [1s to 5v Booster \(Fixed\)](#)
- [SPST Switch](#)
- [LIPO Batteries & Charger](#)
- [Battery Leads](#)

9V Battery Power Setup

This method is the newer cheaper and easier method. Instead of using a 1s lipo with a voltage booster, this opts for a 9v battery with a 5v regulator. Modern 9v lipo batteries are now available, which are really 2s lipo packs with integrated chargers.

- [Rechargeable 9v Battery](#)
- [5V Voltage Regulator](#)
- [Battery Clip](#)

9V Battery Power Setup

You'll need a 2S LiPo for power, around 1500mah – 2000mah. The ESCs listed have a 5v regulator built in which can be used to power the ESP32. Connect the 5v power from one ESC, using just signal and ground from the other.

Using the RTR Scout32

- Charge the Scout32 via MicroUSB at the rear. Red light turns on while charging. Blue light turns on when full.
- Turn on Scout32 via switch. Scout32 is ready when lights flash.
- Connect your computer or phone to the wireless network "Scout32".
- Open a web browser, and enter the address 192.168.4.1

- Once loaded, use the on-screen buttons to control the Scout32, or use the arrow keys on a keyboard.

Assembly Guide & Printing

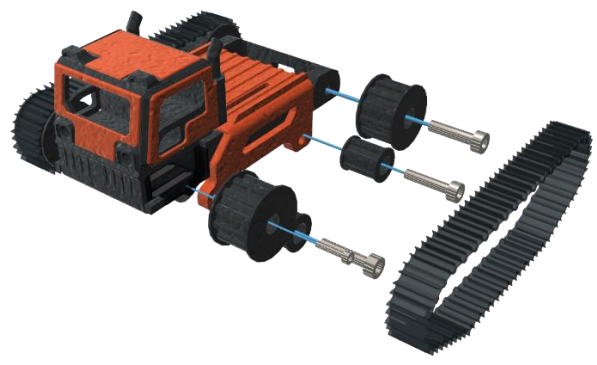
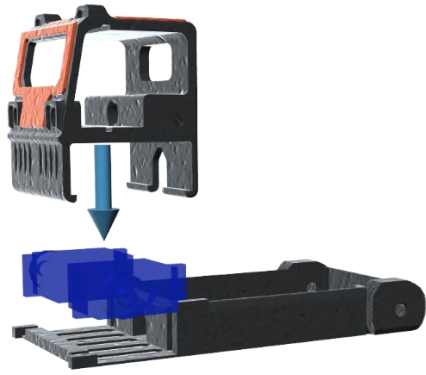
- The decorative plating for a two-tone color scheme may be glued onto the cab and doors.
- Tracks are tested to print with a 0.4mm nozzle in Cura 4.6. I recommend a direct drive hot end.
- All parts should be printed in their default orientations. The “bed” of the truck may be folded down and superglued in place to cover the open area on the rear of the chassis.
- I recommend a spacer or washer between each idler and the chassis.
- N20 Variations may require different spacers for the idlers, because of a different offset from the motor.
- The cab is somewhat difficult to snap on. It’s easiest to hook it under the front of the chassis and rock it backward into the slots for the clips. Axles use M4x20mm bolts. A bit of plastic safe grease helps minimize wear. Do not overtighten.

ScoutXL Additions

- 10mm LEDs are used instead of 5mm.
- The chassis may require printing in two halves. Use 3mm rod through the holes in the face of the seam as reinforcement and use glue to join the two halves permanently.
- Idlers and Bearings
 - Idler wheels use a pair of 5x11x4mm ball bearings, riding on M5x40mm bolts.
 - Do not overtighten! To help prevent binding, BearingSpacers.stl goes in between the ball bearings, though this is optional.
 - Each idler wheel uses a single Idler-Spacer.stl to space it away from the chassis.
 - Each Drive wheel uses a single DriveWheel-Spacer.stl to space it away from the chassis.
 - MySpacer.stl is an extra optional spacer used in front of the wheel to take up space on longer screws.

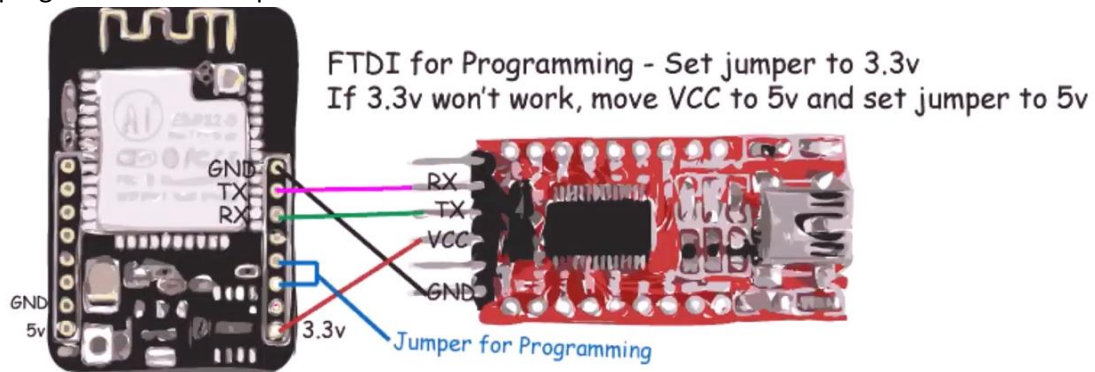


- Drive Wheels use M3 set screws. I recommend longer 6mm screws to prevent stripping the plastic.
- The rear charge port is resized to take a USB-C Breakout board, which I wired to a USB to 2s lipo charger circuit.
- To mount the ESP32, I used a 40x60mm PCB screwed to BoardMount.stl.

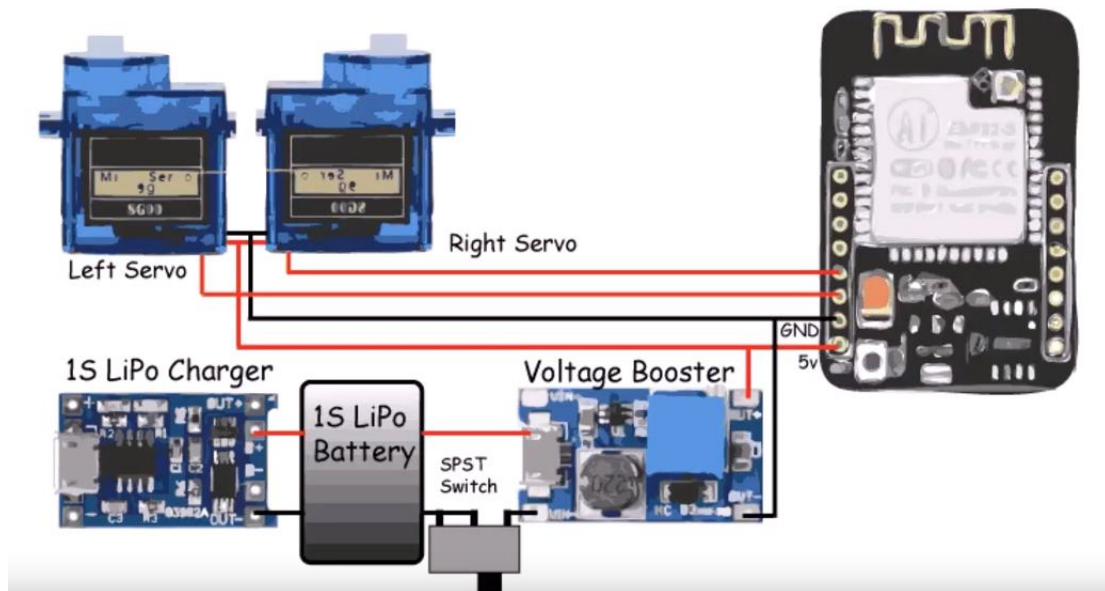


Wiring & Pinout

- Wiring / Pinout should be the same for both ESC and Servo operation.
- The Default LED configuration is to remove the stock LED and solder to those pads for the cab LEDs. If you don't want to sacrifice the stock LED, you can instead use an external LED driver connected to GPIO4. I don't recommend powering LEDs straight off a GPIO pin, but I've been told it does work.
- For easier programming after assembled, I use a 3 pin servo connector attached to an FTDI adapter. This only uses GND TX and RX, and I power the Scout via battery while programming. I found this to be more reliable and it makes updates to the Scout easier, as you can plug the programmer into the pins from the rear window on the cab.



Servo Pinout

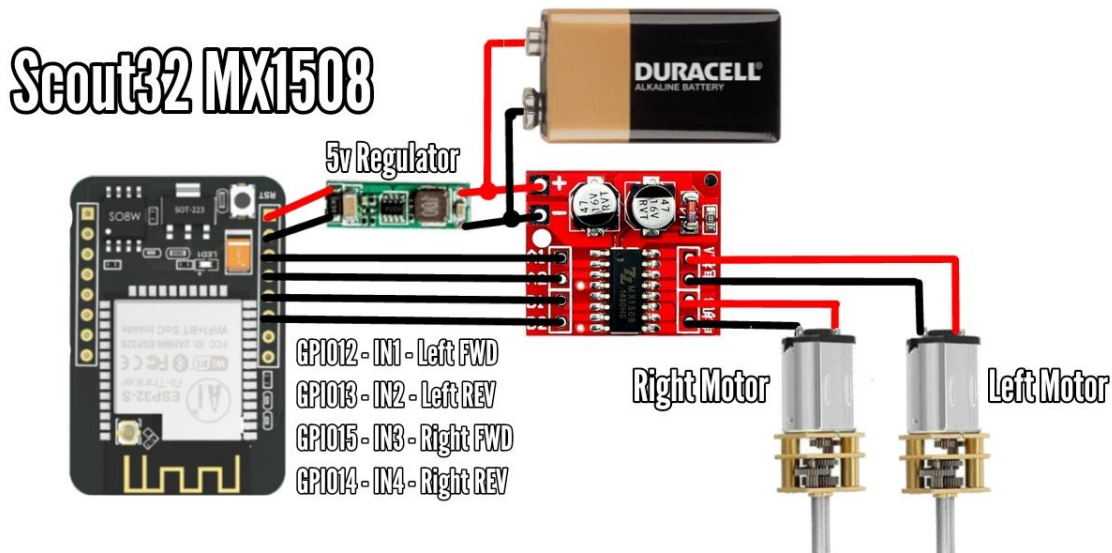


If using a 9v battery or 2s lipo

Omit LiPo charger from circuit shown above, and replace 1s lipo battery with 9v battery. Voltage Booster is replaced with Voltage regulator. If using ESCs with BEC (such as on ScoutXL), use power from one ESC to power ESP32, connecting only ground and signal from the other.

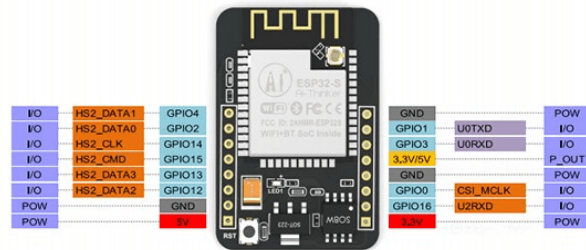
If using an MX1508 or similar motor driver:

Use alternate MX1508 Sketch! An additional two pins are assigned for control, and the PWM code is altered to directly drive motors. No endpoint or neutral tweaking is necessary.



Positive and Negative go to MX1508, and pin assignments from ESP32 are as follows:

- GPIO12 - IN1 (Left Forward)
- GPIO13 - IN2 (Left Reverse)
- GPIO15 - IN3 (Right Forward)
- GPIO14 - IN4 (Right Reverse)



Programming

All files must be unzipped. Both Scout32.ino and app_server.cpp must load.

****Arduino Setup****

- Required Board URLs (File, Preferences):
 - https://dl.espressif.com/dl/package_esp32_index.json
- Under Boards Manager, search ESP32 and install ESP32 by Espressif version 1.0.6
- Select ESP32 Wrover Module as board and set partition to 3mb Huge App.

Code Tweaks, Variables, and Troubleshooting

Servo Tuning

- **Help, my servos are too slow!** If using TowerPro 360 Servos or ESCs, set int endpoint = 180; in app_server.cpp at top of file.
- If using modded servos, start with a value of 30 and go up by increments of 10 until you get full speed. If you set it too high, trim won't work correctly.
- The variables for the servo control code are not immediately obvious. Please refer to Scout32 Servo Control Code.PDF for more info.

Camera Tuning

Sideways Image Fix – Some cameras are oriented differently

Update! The control interface now includes camera rotation, so you shouldn't need to do this anymore!

- Find line: ``
- Replace with: ``
- OR: ``

Range / Reception

- Scout32 may be affected by interference. Please use away from other wireless devices if you encounter trouble. By default, the Scout uses WiFi channel 1. This can be changed in Scout32.ino in the variables at the top.
- If the Scout32 turns off randomly, the battery may be low. Fully charge the Scout32. Try adding a capacitor in parallel with the power input to even out power dips.
- Your phone may detect the wireless connection as “unable to connect to the internet”. In this case, you will have to manually accept to stay connected to enable network traffic.

Thanks for downloading!

If you have any issues, please contact Pilotgeek45@gmail.com . I greatly appreciate any feedback. This project started back in 2019 and has come a long way thanks to the feedback and improvements suggested by users. Revenue from Amazon Affiliate Links has allowed me to further improve new prototypes, so each build helps contribute to further improvements.