A black and white logo

Description automatically generated

UplinkRobotics Mobile Platform Educational Kit Documentation and User Guide

Contents

[Introducing the UplinkRobotics Mobile Platform Educational Kit 3](#_Toc177479461)

[Kit Contents 4](#_Toc177479462)

[Assembled Kit Dimensions: 6](#_Toc177479463)

[Top View (96mm Gecko Wheels): 6](#_Toc177479464)

[Front View (96mm Gecko Wheels): 7](#_Toc177479465)

[Internal Views (96mm Gecko Wheels): 7](#_Toc177479466)

[COR Specifications: 8](#_Toc177479467)

[Connectors: 8](#_Toc177479468)

[Battery Connector: 8](#_Toc177479469)

[Left Motors 1 and 2: 8](#_Toc177479470)

[Right Motors 1 and 2: 9](#_Toc177479471)

[LED Array: 9](#_Toc177479472)

[Receiver Header: 9](#_Toc177479473)

[Daughter Board Header: 9](#_Toc177479474)

[Servo Jumpers: 9](#_Toc177479475)

[USB Jumpers: 9](#_Toc177479476)

[USB Header: 10](#_Toc177479477)

[Servo Headers: 10](#_Toc177479478)

[VTX Header: 10](#_Toc177479479)

[Camera Header: 10](#_Toc177479480)

[VTX Power Jumper: 10](#_Toc177479481)

[Headlight CH2: 10](#_Toc177479482)

[Headlight CH1: 10](#_Toc177479483)

[Chassis Specs and Mounting Holes 12](#_Toc177479484)

[Front: 12](#_Toc177479485)

[Rear: 12](#_Toc177479486)

[Side Motor Pod Dimensions: 13](#_Toc177479487)

[13](#_Toc177479488)

[Main Top: 14](#_Toc177479489)

[Mounting Hole Top: 15](#_Toc177479490)

[Wheel Choice Specifications: 16](#_Toc177479491)

[Base Options: 16](#_Toc177479492)

[135mm Series Leopard Tires 16](#_Toc177479493)

[144mm Wasteland Wheels 16](#_Toc177479494)

[Gecko Wheels 16](#_Toc177479495)

[Rhino Wheels 17](#_Toc177479496)

[Programming the COR: 18](#_Toc177479497)

[Used Programming Libraries 21](#_Toc177479498)

# Introducing the UplinkRobotics Mobile Platform Educational Kit

UplinkRobotics has spent the last few years revolutionizing the ground-based remote inspection industry with our inspection crawlers. These crawlers are ahead of the game in reliability, performance, and cost. We have done this through revolutionary mechanical design and custom-built electronics. We are now proud to release an educational kit that utilizes the same custom-built electronics making it extremely easy to build a variety of ground-based robotic platforms.

This kit is powered by the COR. An ESP32 based microcontroller board that is focus built for ground based robotic platforms. Powered by a 3S LiPo or Li-Ion battery, it has a 5V output, a 3.3V output, as well as motor controls, servo controls, DAC, ADC, I2C, UART, I2S, Wi-Fi, Bluetooth, and analog video connections. This wide range of capabilities makes it extremely capable as a base platform to attach a variety of sensors to. These sensors can range from simple sonar sensors to microcomputers with lidar, stereo vision, or SLAM.

This kit prioritizes flexibility and reliability. No more buying new kits every year after the last batch fell apart due to poor construction. These kits are custom designed, and 3D printed to fit reliable, high power planetary gear motors as well as a purpose-built controller. The COR has been deployed in inspection crawlers demonstrating their reliability and capability.

Programming the COR is as simple as firing up the Arduino IDE, installing the ESP32 board library, and loading up the examples developed by us. These examples show how to control motors and servos, connect to various RC receivers, interface sensors, and drive basic rovers. Since the COR uses the popular ESP32 microcontroller, there are hundreds of examples online for IOT, sensors, manufacturing, etc. The ESP32 is a powerful dual core 240MHz microcontroller with many built-in features that is quite popular.

The chassis of the kit has many mounting points with standardized spacing. These mounting points can easily be used to mount anything the user desires, especially with access to a 3D printer. There are options already available for mounting things like a Raspberry Pi, sonar sensors, and servo gimbals that can be purchased from the UplinkRobotics store.

The kit also has many different wheel options. With a motor on each wheel, the kit has great torque given its size. There are wheel options ranging from hard rubber wheels for indoor use, to airless all-terrain tires for outdoor use. The different tire options are explained in detail in this document.

# Kit Contents

1 set: Chassis, 4 motor pods, top, and battery tray lid

* PETG 3D printed
* Painted with bed-liner for scratch resistance (optional)
* Temperature resistance up to 140F long-term, 180F for 2 minutes, bed liner adds additional short-term resistance

4 pcs: TT Motors 97 RPM

* Planetary gearbox
* 6mm D-shaft

Wheels and Couplers

* These vary based on the kit options

COR microcontroller board:

* 9.6V-12.6V input voltage (3S LiPo or Li-Ion)
* 5V @ 3.5A output
* 3.3V @ 2A output
* 2 dedicated 5V servo outputs
* I2C, 2xUART, 3.3V GPIO, 5V GPIO, ADC, DAC, I2S
* ESP32-WROOM-DA module
  + 16MB Flash
  + 2 cores @ 240 MHz
  + Wi-Fi and Bluetooth capability
  + OTA updates
* 4 PWM bi-directional brushed motor drivers
  + 2 separately controllable banks of two drivers
  + 4.5A sustainable
  + 9.6V – 12.6V output (direct from input)
* 2 PWM LED channel drivers
  + 12V @ 1A
  + 3.3V with 1 ohm resistor in-line @ 1A current limit
* Analog camera and video transmitter dedicated ports
  + 12V or 5V switchable power
  + Video pass-through
  + Up to 2 switchable camera inputs with optional camera switcher board
* 8 LED array

Battery and Charger

* 2200mAh 3S Li-Po
* 25W balance charger that charges through the balance lead

All Needed Hardware and Tools

* All the screws and tools needed to assemble the kit will be present

# Assembled Kit Dimensions:

## A drawing of a machine Description automatically generatedTop View (96mm Gecko Wheels):

## A screenshot of a computer Description automatically generatedFront View (96mm Gecko Wheels):

## A black and white drawing of a machine Description automatically generatedA black and white drawing of a machine Description automatically generatedInternal Views (96mm Gecko Wheels):

# COR Specifications:

## Connectors:

### Battery Connector:

* Screw terminal that accepts 12-30AWG Ferrules
* Input voltage 9.6V – 12.6V – equivalent to 3S LiPo and Li-Ion batteries
* Input should be switched elsewhere and 15A fuse in-line is recommended
* NO REVERSE POLARITY PROTECTION, NO OVER-VOLTAGE PROTECTION

### Left Motors 1 and 2:

* Brushed motor drivers – both of these channels are tied together and given the same input signals
* 9.6V – 12.6V @ 4.5A on each channel
* Controlled with pins:
  + DIR: IO14
    - High/Low determines output polarity/direction of the PWM signal
  + PWM: IO12
    - Must be a PWM signal higher than 20KHz to not be audible
* Need a specific startup sequence to initialize that can be found in the motor driver example code
* JST-XH connectors

### Right Motors 1 and 2:

* Identical to left motor drivers except:
* Controlled with pins:
  + DIR: IO32
    - High/Low determines output polarity/direction of the PWM signal
  + PWM: IO33
    - Must be a PWM signal higher than 20KHz to not be audible

### LED Array:

* LEDs are controlled by a shift-register controlled by pins:
  + CLK: IO13
  + IN: IO2
* LEDs activated with an input-high in the register
* Follow LED array example code for specifics

### Receiver Header:

* Designed to be connected to an RC receiver that outputs one of the following protocols:
  + i.BUS (FlySky), S.BUS, ELRS
  + REC = IO16
* Use JST-EH connectors

### Daughter Board Header:

* All unused connectors IO are broken out here as well as 12V, 5V, and 3.3V supplies
* Standard 2.5mm pitch “dupont” connector

### Servo Jumpers:

* These jumpers determine whether the Servo headers are connected to the GPIO from the ESP32 or the REC input from the receiver header
* This is useful if you would like to use S.BUS servos or i.BUS servos that need their data input line connected directly to the output from the receiver.
* Standard 2.5mm pitch “dupont” connector

### USB Jumpers:

* These jumpers determine whether the TX0 and RX0 pins are connected to the USB header or the daughter boards header
* Standard 2.5mm pitch “dupont” connector

### USB Header:

* Used to program the COR
* This header is also a UART port that can be used for anything
* U5v is not connected, board must always be powered through the 9.6V – 12.6v input
* RX and TX must be routed here with the USB Jumpers in the right positions.

### Servo Headers:

* Can be used to power and control 5V servos. Can output s.BUS, i.BUS from the receiver or RC-PWM from the ESP32
* Use JST-EH connectors
* Designed to connect directly to Robotis Dynamixel XL330 and XC330 servos

### VTX Header:

* Switchable between 5V and 12V (input power)
* IO23 connected for GPIO
* VTX pin connects directly to Camera Header for analog video passthrough
* Use JST-EH connectors

### Camera Header:

* 12V (input power)
* VTX pin connects directly to VTX Header for analog video passthrough
* IO19 @ 5V for camera control
* Use JST-EH connectors

\*\* The VTX and Camera Headers can be used interchangeably if the power voltages or output GPIO levels need to be reversed\*\*

### VTX Power Jumper:

* Switches whether 12V or 5V goes to the VTX Header
* Standard 2.5mm pitch “dupont” connector

### Headlight CH2:

* Screw terminal that accepts 12-30AWG ferrules
* 12V @ 1A output that is switched on and off by IO0
  + This channel is also connected to the ESP32 Debug mode button

### Headlight CH1:

* Screw terminal that accepts 12-30AWG ferrules
* 3.3V A 1A output with a 1ohm 1.5W resistor in series to current limit a very powerful LED diode
* Controlled by IO4

# Chassis Specs and Mounting Holes

### Front:

M3 mounting holes w/ 5mm depth

A drawing of a line

Description automatically generated with medium confidence

## Rear:

M3 mounting holes w/ 5mm depth

A drawing of a camera

Description automatically generated

## Side Motor Pod Dimensions:

M3 holes w/ 7mm max depth

## A screenshot of a computer Description automatically generated

## Main Top:

A blueprint of a machine

Description automatically generatedM3 mounting holes w/ 5mm depth

## Mounting Hole Top:

A blueprint of a machine

Description automatically generatedMounting holes w/ 5mm depth

# Wheel Choice Specifications:

This educational kit can support many wheels up to 150mm in diameter. Many available options are based off the goBILDA ecosystem of wheels and couplers. These use a 14mm bore supported by a 16mm square hole pattern with M4 hardware. The hubs used must be able to adapt to a 6mm D-shaft that is 15mm long.

## Base Options:

These are the base options that can be selected to be included with the educational kit. These will be paired with goBILDAs Series Sonic Hub (6mm D-Bore) to adapt to the motor. Note that the chassis has no suspension, therefore any drops must be absorbed by the tires/wheels. For applications where the chassis may experience drops, the Rhino series wheels are not recommended.

### 135mm Series Leopard Tires

goBILDA tires that should be paired with their Series Hurricane Rims. These all terrain tires utilize an airless design with a foam core for padding. **These are a good fit for general all-terrain use.** These will likely need the tire to be CA glued to the rim so that it does not come off.

### 144mm Wasteland Wheels

goBILDA tires are a futuristic airless design made of pliable rubber. The tire is molded straight to the rim so there is no need for CA glue**. These are good for dry all-terrain conditions.** However, mud will get stuck in the tires and be hard to clean and remove.

## Gecko Wheels

These goBILDA wheels come in two sizes: 72mm and 96mm. They utilize a futuristic airless design and are very soft and pliable. **These are best for hard un-even surfaces. Use these for indoor designs that will require good traction and grip.**

## Rhino Wheels

These goBILDA wheels have a hard inner plastic rim with a relatively thin rubber outer gripping surface. They come in many different sizes varying from 72mm to 120mm. They also have two variants; high-traction and high-durability. The difference is in the hardness of the outer rubber shell. There is also a thinner version; 16mm vs. 32mm. The high-traction 96mm or 120mm tires are recommended for this application**. Rhino wheels should only be used if the kit will always be on flat and even surfaces.**

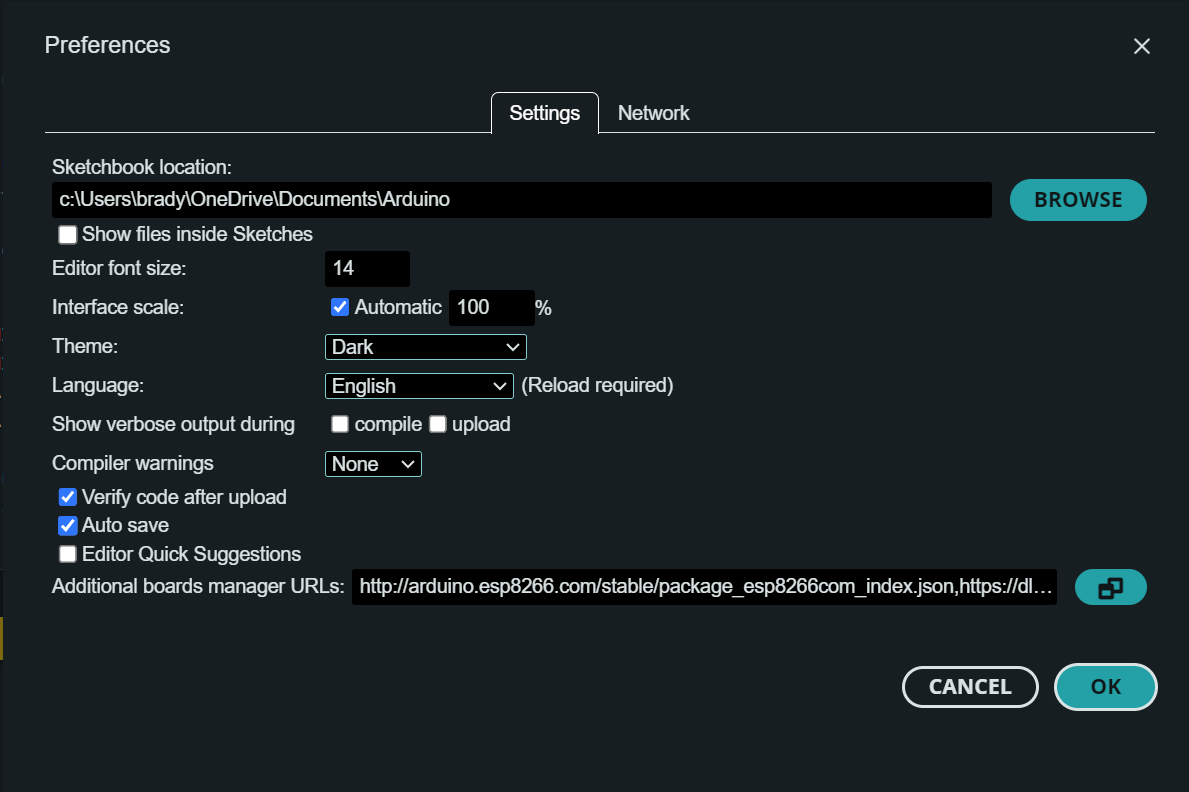
# Programming the COR:

The COR uses a standard ESP32-WROOM-DA module with 16MB of flash. However, the board does not have a built-in USB to UART adapter so an external one must be used. This should be included in the kit and will be needed to program the COR. It is recommended to use the Arduino IDE for all programming and all examples will be .ino files intended for use with the Arduino IDE.

In order to program the COR the following steps must be taken:

1. Download and install the Arduino IDE on your computer.
2. Add the following board manager to the IDE
   1. Go to File -> Preferences
   2. Paste the following into the “Additional boards manager URL” section:

http://arduino.esp8266.com/stable/package\_esp8266com\_index.json, https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\_esp32\_index.jsonz



* 1. Close the Preferences Tab

1. Install the ESP32 boards library version 2.0.17
   1. A screenshot of a computer

      Description automatically generatedGo to the Boards Manager on the left tab and search for ESP32. Select the esp32 library by Espressif Systems. **Make sure to select the dropdown and choose version 2.0.17.** Newer versions are not compatible with some of the libraries used in the examples for this kit. **Make sure not to update this library when the IDE opens.**
2. Next you need to install the driver for the USB programming dongle that is included with the kit. This dongle uses the Silicon Labs CP210X chip for USB to UART conversion.
   1. The driver can be found here:

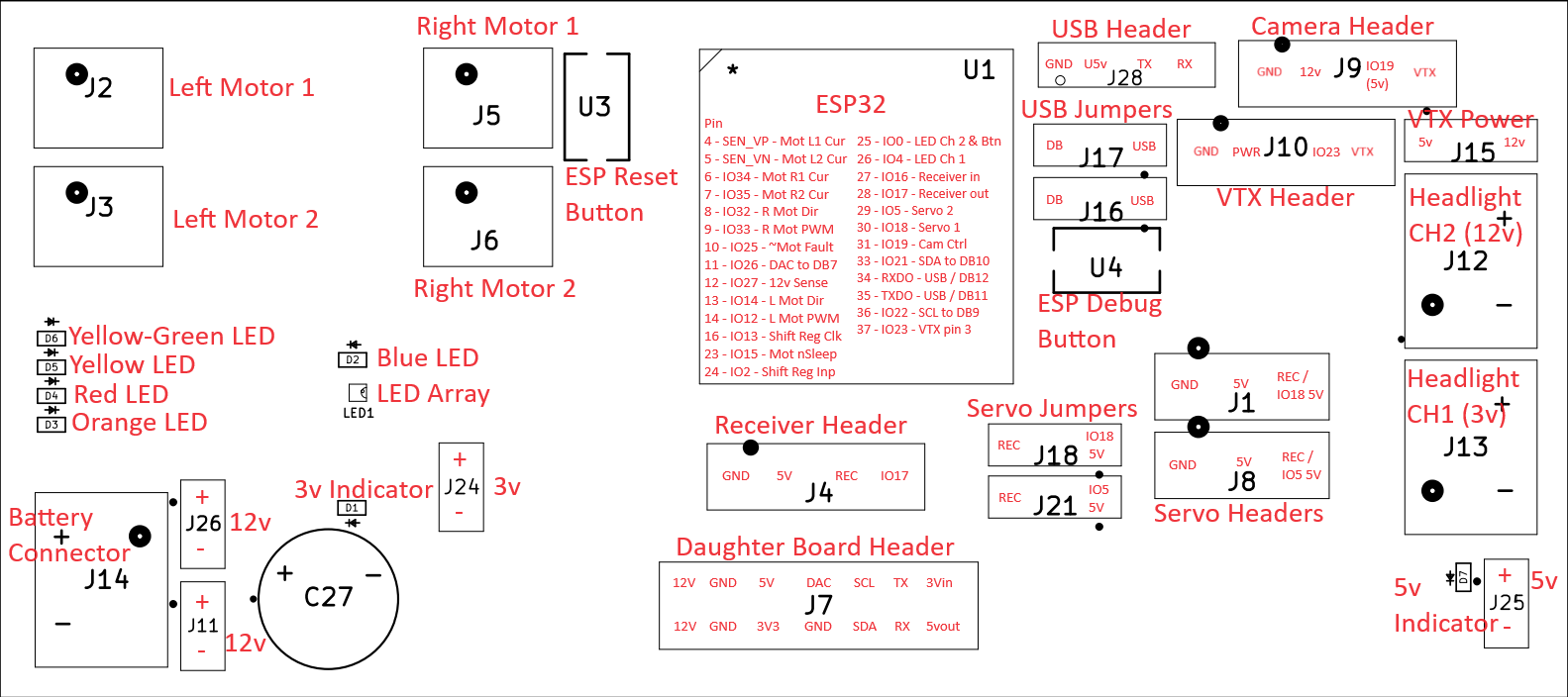
<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers?tab=downloads>

Or by searching for Silicon Labs CP210X

* 1. You can tell the driver is installed correctly by opening the IDE and you should have a COM# port available with the dongle plugged in.

1. Next you should open the “led\_array.ino” example file to test uploading.
2. Choose the Board and Port
   1. A screenshot of a computer

      Description automatically generatedThe board you need to choose is the “ESP32-WROOM-DA module” and the Port will show up as COM#. The # will depend on your system.
3. To upload, you need to press and hold the ESP Debug Button, while holding, press the ESP Reset Button once. This reboots the ESP32 into programming mode. You can now hit the Upload in the IDE and it should program the ESP32.



**1st HOLD**

**2nd PRESS**

Used Programming Libraries:

The following libraries are used in the example code provided.

ESP32Servo

This library makes it extremely simple to output RC-PWM signals from core to control any 5V servo with the on-board connectors.

IBusBM

This library is used to connect to a FlySky receiver to read signals from a transmitter.

TheDIYGuy999\_SBUS

This library is used to communicate with receivers that use the SBUS protocol. This protocol is very popular and used by companies like FrSky, Herelink, and SIYI.

esp32-elrs-crsf

This library is used to communicate with open source ExpressLRS receivers which use a modified version of the CRSF protocol that TBS Crossfire receivers use. ExpressLRS is an extremely fast and long range option but due to its open source nature, compatibility problems can arise.

These libraries can all be found and downloaded