

# ESPcopter SDK 1.0.0 (Beta)

## 1-) General Review

## 1.1-) Internal Features of the ESPcopter:



Full charge in 45 minutes with USB connection





ESP8266-12S 32-bit 160MHz



IEEE 802.11 b/g/n Wi-Fi connection



3- axis Gyro, accelerometer & magnetometer

## 1.2-) ESPcopter Switch and Button

- The button is used to reset ESPcopter MCU
- The switch is used to open and close ESPcopter



## 1.2-) Pinout and Propeller and Motor Directions

When installing in accordance with the letters on the propellers, the motors must be fitted according to the cable colors.

#### **Engine positions:**

Left Front: B - Red, Blue

**Right Front: A - Black, White** 

Left Rear: A – Black, White

**Right Rear: B – Red, Blue** 



#### 1.3-) How to Charge the ESPcopter:

The ESPcopter will charge when connected to the micro-usb. The switch on the ESPcopter must be in the off position to charge.

#### **Red Light: Charging Green Light: Fully Charged**

## 2-) Software:

#### 2.1-) Arduino Installation:

Download and install the latest version from the Arduino web site: https://www.arduino.cc/en/Main/Software

## Download the Arduino IDF



The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other opensource software. This software can be used with any Arduino board. Refer to the Getting Started page for Installation

Windows Installer, for Windows XP and up Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10 Get 📕

Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits Linux 64 bits Linux ARM

**Release Notes** Source Code Checksums (sha512)

## 2.2-) Driver Installation:

The following driver is reqired for the ESPcopter to be recognized by the computer. Download and install the approperate driver version for your OS.

https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

#### 2.3-) ESP8266 Library

To begin, we'll need to update the board manager with a custom URL. Open up Arduino, then go to the Preferences (File> Preferences). Then, towards the bottom of the window, copy this

URL into the "Additional Board Manager URLs" text box:

#### http://arduino.esp8266.com/stable/package\_esp8266com\_index.json

	sketch_aug07a   Arduino 1.6.5			
	File Edit Sketch Tools Help			
references	00 888			23
Sketchbook locati	on:			
C:\Users\user.na	ame \Dropbox \Work \Arduino			Browse
Editor language:	System Default	← (requires restart of Ar	duino)	
Editor font size:	12			
Show verbose ou	tput during: 📝 compliation 📝 upload			
Compiler warning	s: None 👻			
Addie	Panel Manage HDIs		×	
Additional	Boards Manager UKLS			
Enter addition	al URLs, one for each row			
http://a	rduino.esp8266.com/stable/pa	ackage_esp8266com_inde	x.json	
			tfun ir	ndex.tson
			K Canal	
			Card	
(edit only when A	rduine e not running)			
			0	Cancel
				0.05
		SparkFun ESP8266 Thing, 80 MHz	, 115200 on COM210	

Hit OK. Then navigate to the Board Manager by going to Tools > Boards > Boards Manager. There should be a couple new entries in addition to the standard Arduino boards. Look for esp8266. Click on that entry, then select Install. You need to install **2.5.0** version of esp8266 library.

Boards Manager	x
Type Al  Filter your search	
Intel i686 Boards by Intel Boards included in this package: Edison. More info	*
AMEL-Tech Boards by AMEL Technology Boards included in this package: SmartEverything Fox. Online help More info	
esp8266 by ESP8266 Community Boards included in this package: Generic ESP8266 Module, Olimex MOD-WIFI-ESP8266(-DEV), NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Adafruit HUZZAH ESP8266 (ESP-12), SweetPea ESP-210. Online help More info	111
	ose

The download process can take up to 10 minutes depending on your internet speed. After the download is done, select NodeMCU 1.0 from the **Tools tab** and follow the other settings.

espcopterlst - FlightControl.h | Arduino 1.8.8 (Windows Store 1.8.19.0)

File Edit Sketch	Tools Help					
	Auto Format Archive Sketch	Ctrl+T				
espcopterist	Fix Encoding & Reload		版版中的	NeoPixel.h	PID.cpp	PID.h
	Manage Libraries	Ctrl+Shift+I	-			
	Serial Monitor	Ctrl+Shift+M				
#include	Serial Plotter	Ctrl+Shift+L				
	WiFi101 / WiFiNINA Firmware Updater					
unsigned	Board: "NodeMCU 1.0 (ESP-12E Module)"	3	>			
	Upload Speed: "115200"	3	>			
unsigned	CPU Frequency: "160 MHz"	3	>			
boolean	Flash Size: "4M (1M SPIFFS)"	3	>			
	Debug port: "Disabled"	3	>			
	Debug Level: "None"		>			
float fa	IwIP Variant: "v2 Lower Memory (no features)"	3	>			
float fa	VTables: "Flash"	3	>			
void Fli	Exceptions: "Disabled"	3	>			
	Erase Flash: "Only Sketch"	3	>			
	Port		>			
if ((micr	Get Board Info		ТА	on		
tsbat =						
//Serial	Programmer: "AVRISP mkll"	3	>			
if long	Burn Bootloader					
II (anal	$OgRead (AU)^{6} < 3000) {$					
batter	<pre>yCount = batteryCount +1;</pre>					

}else{

#### Downloading the ESPcopter library:

Before downloading the code from the website, you must use the contacts page to request source code. See the following site: http://espcopter.com/code-release/

Code

DOWNLOAD ESPCOPTER ARDUINO CODE

Version 1.0.0 Beta - 13.03.2019

After downloading, remove the zip file twice and put the file (ESPcopter "(Files> Arduino> Library) into the file. In the Examples section you will see the sample codes of the ESPcopter.

Open the following example program:

Ν	lew	Ctrl+N				
C	)pen	Ctrl+O				
C	Open Recent	>				
S	ketchbook	>		_		
E	xamples	>	<b>A</b>			
C	lose	Ctrl+W	Temboo	>	e:	
S	lave	Ctrl+S	RETIRED	>		
S	ave As	Ctrl+Shift+S	Examples for NodeMCU 1.0 (ESP-12F Module)			
		0.1.010.0	ArduinoOTA	>		
P	age Setup	Ctrl+Shift+P	DNSServer	>		
P	rint	Ctri+P	EEPROM	>		
F	references	Ctrl+Comma	ESP8266	>	todly.	
	);+	Ctrl I O	ESP8266AVRISP	>	iccury.	
(	zuit	CIII+Q	ESP8266HTTPClient	>		
			ESP8266httpUpdate	>		
			ESP8266HTTPUpdateServer	>		
			ESP8266LLMNR	>		
			ESP8266mDNS	>		
			ESP8266NetBIOS	>		
			ESP8266SSDP	>		
			ESP8266WebServer	>		
			ESP8266WiFi	>		
			ESP8266WiFiMesh	>		
			Ethernet(esp8266)	>		
			Hash	>		
			SD(esp8266)	>		
			Servo(esp8266)	>		
			SoftwareSerial	>		
			SPISIave	>		
			TFT_Touch_Shield_V2	>		
			Ticker	>		
			Wire	>		
			Examples from Custom Libraries			
			ESPcopter	>	Basics >	
			RemoteXY	>	Remotes >	ESPconterRemoteXY
			WILC LL	1		

The ESPcopter will have the flight control software to make the flight stable. This ESPcopterWebApp example is a demonstration of the ESPcopter having the additional ability to act as a self-contained web server. This will get you running your ESPcopter fast because you can control it from a web page on your phone, tablet or laptop. More advanced examples (Although this is very cool) are already available from the Arduino IDE menu.

Let's hope you have had the ESPcopter switched off and the batteries charging via the USB port so you can be ready for the first flight.

## ESPcopter standalone Code(Web APP)

#define REMOTE\_WEB\_APP

#define WEB\_APP\_WIFI\_SSID "ESPcopter" //ESPcopter Wiffi ismi

#define WEB\_APP\_WIFI\_PASSWORD "12345678" //ESPcopter Wiffi şifresi

<mark>#include <espcopter.h></mark>

<mark>void setup() {</mark>

<mark>mainSetup();</mark>

}

void loop() {

mainLoop();

## }

## **Connection:**

After installing the software, turn on the drone. In the wifi tab of your phone, you will see the name of the drone.

WIFI\_SSID "ESPcopter WIFI PASSWORD "12345678"

After making the connection, open any web browser application from your phone. Type 192.168.4.1 in the search line.



# **Control review:**



#### **Buttons:**

••••• \	vodafone TR 🗢	20:46	%66 💶 )•
←	192.168.4.1		1 :



First Button: ARM – DISARM Second Button: AUTO YAW CONTROL Third Button: AUTO ALTITUDE CONTROL Forth Button: AUTO MISSION CONTROL

#### ESPcopter standalone Code (RemoteXY)

#define REMOTEXY\_WIFI\_SSID "RemoteXY"

#define REMOTEXY WIFI PASSWORD "12345678"

#define REMOTE\_XY\_REMOTE

#include <espcopter.h> // library

void setup() {

mainSetup(); // main flying setup

void loop() {

mainLoop (); // main flying loop

}

#### Connection:

After installing the software, turn on the drone. In the wifi tab of your phone, you will see the name of the drone.

WIFI\_SSID "RemoteXY WIFI\_PASSWORD "12345678"

After you make the connection, open RemoteXY from your phone.



After making the connection, the screen below will open automatically.

After the first connection. There will be ESPcopte box in RemoteXY app. You can connect ESPcopter by cliking this box.



# **Controller Review:**





# **Computer Control (Processing)**

```
#define PROCESSING_REMOTE
#include <espcopter.h>
void setup() {
  mainSetup();
  setTrimRoll(0);
  setTrimPitch(0);
  setTrimYaw(0);
}
void loop() {
  mainLoop ();
}
```

#### **Connection:**

After the uploading the Processing control code to ESPcopter. Open the wiffi screen and then connect the ESPcopter



# **Control Screen**



# **Controller Review:** 1-) Control Switches:



If ESPcopter did not connect, reset it and then click the reconnect button two times.

# 2-) Control Switches:



Button: ARM – DISARM Fly Mode 1: YAW CONTROL Fly Mode 2 AUTO ALTITUDE CONTROL Fly Mode 3: AUTO ALTITUDE AND FLIGHT MISSION CONTROL

# 3-) Control RGB LED:



It is not working current code. We will solve the problem next version.

# 4-) Graph:



It will show something like connection speed. Current code is not working.

# 5-) Control Joystick 1:



6-) Control Joystick 2:



# AUTOHOMOUS

# 7-) Auto Flight Route Control Part

# 8-) Auto Flight Route Simulation:



## **Calibration:**

## 1- Start Calibration

The ESPcopter has at a minimum a Gyro, acceleration and Earth magnetic field sensors. They must be calibrated when you first turn on the ESPcopter. Calibration mode starts with the red light flashing. Do not turn off the ESPcopter until the calibration algorithm has finished. If you turn it off, the calibration will run again at the next boot.



## 2- Magnetic Field Calibration (Compass):

Now place it on a flat, non-metal surface. The blue light indicates that the magnetic field calibration has started. At this stage you have to turn the ESPcopter around twice.



## 3- Gyro Calibration:

After you have calibrated the compass the ESPcopter's light will be purple. Do not touch the ESPcopter after this step. It will run its engines for a few seconds without taking off from the ground.



Calibration is completed!!!

#### **Custom commands:**

#### **ESPcopter functions:**

The LEDs and motors on the ESPcopter can be controlled using the functions found in this list.

Function	Acceptable Values	Description
esp.redLed_Digital();	0 - 1 or FALSE - TRUE	Controls Red LED on/off
esp.blueLed_Digital();	0 - 1 or FALSE - TRUE	Controls Blue LED on/off
esp.greenLed_Digital();	0 - 1 or FALSE - TRUE	Controls Green LED on/off
esp.redLed_Analog();	0 - 255	Controls Red LED brightness
esp.blueLed_Analog();	0 - 255	Controls blue LED
		brightness
esp.greenLed_Analog();	0 - 255	Controls green LED
		brightness
esp.motorFL_Analog();	0 - 255	Operates front left engine at
		desired power.

esp.motorFR_Analog();	0 - 255	The front right operates the engine at the desired power.
esp.motorRL_Analog();	0 - 255	Operates the rear left engine at the desired power.
esp.motorRR_Analog();	0 - 255	The rear right operates the engine at the desired power.

## **ESPcopter Control Table:**

The control method of the ESPcopter can be changed using the definitions in this list. Only one definition should be activated from this list.

Function	Description	Control Device
#define REMOTE_XY_REMOTE	Control with RemoteXY	Phone - Tablet
#define BLYNK	Control with BLYNK	Phone - Tablet
#define PROCESSING_REMOTE	Control with processing	Computer
#define MQTT	Control with MQTT	Computer
#define REMOTE_WEB_APP	Web application control	Phone - Tablet
#define PPM_REMOTE	Control with PPM receiver	Standard RC Remote

## **ESPcopter Global Variable Definitions:**

Function	Description	Value Range
setTrimRoll();	Trim on the X-axis.	-500 - 500
setTrimPitch();	Trim on the Y-axis.	-500 - 500
setTrimYaw();	Trim on the Yaw	-500 - 500
setArmControl();	Motor Enable	false - true
setFlyMode_1();	Z-axis stabilization on-off	false - true
setFlyMode_2();	Height fixing on-off	false - true
setFlyMode_3();	Optical flow module with	false - true
	motion stabilization	
landing();	Landing	false - true
setMotorMax();	Set maximum motor power	600-900
getRX_throttle();	Motor power rating	0 – (motorMax)
getRX_roll();	The value in the X-axis	-100 : + 100
getRX_pitch();	The value in the Y-axis	-100 : + 100

getRX_yaw(); The value in the Z-axis -100 : + 100
---

## Autonomous Flight Commands:

Function	Description	Value Range
takeOff(Y, T);	When the command line	Y: 200 - 1000 Height
	runs, the drone	T: Flight time
	automatically takes off.	
goforward(T);	The drone moves forward	T: Flight time
	during the duration.	
goBack(T);	During the T Time the drone	T: Flight time
	moves back.	
goLeft(T);	The drone moves to the left	T: Flight time
	during the duration.	
goRight(T);	The drone moves right	T: Flight time
	through the time.	
turnRight(D);	D rotates right up to its own	D: Rotation angle
	angle in angle.	
turnLeft(D);	The angle of D turns to the	D: Rotation angle
	left in its own frame	
delay_(T);	It allows you to wait before	T: Standby time
	executing the next	
	command	
Land();	In autonomous flight mode,	
	this must be at the end of	
	the commands.	

## Altitude Hold Module

Function	Description	Value Range
setVI5310xControl ();	vl5310x module on-off	False - true
setTargetOto();	Height stabilizer with vl5310x module	250 - 1000
getOtoMeasure();	Drone's elevation data	0- 1000

## **Buzzer Module:**

Function	Description	Value range
esp.buzzer();	0 - 1 or FALSE - TRUE	On- Off buzzer

## **Neopixel Module:**

Function Description Value range		Function	Description	Value range
----------------------------------	--	----------	-------------	-------------

#define NeoPixel	Turns the Neopixel module	Include in the program
	on and off	
ESPrainbow();	Makes an automatic	
	rainbow effect	
ESPsetPixel (x,r,g,b);	Set each led separately.	X= 1 - 12
	After setting pixels call	R(Red)= 0 - 255
		G(Green)= 0 - 255
		B(Blue)= 0 - 255
ESPpixelShow();	Applies the changes made	
	with	

## **Optical Flow Module:**

Function	Description	Value range
SetPointOpt[0]	Sets the speed of Drone	-15 - +15
	using the optic flow sensor.	
	If this value is equal to zero,	
	the drone remains	
	stationary in the x-axis.	
	Positive moves right,	
	Negative moves left	
SetPointOpt[1]	Sets the speed of Drone	-15 - +15
	using the optic flow sensor.	
	If this value is equal to zero,	
	the drone stops at the y	
	axis. Positive goes forward,	
	Negative- goes back	
deltaCalX	X-axis flow data from the	Relative to the drone
	optical flow extender.	current speed
deltaCalY	Y-axis flow data from the	Relative to the drone
	optical flow damper.	current speed

## Multi-Distance Module:

Function	Description	Value range
#define HandControl	Manual control or collision	
#define AntiCollision	prevention system	
	Must choose one or the	
	other	
Distance_Y_1();	Y (+) axis distance data	50-1000
Distance_Y_0();	Y (-) axis distance data	50-1000
Distance_X_1();	X (+) axis distance data	50-1000

	Distance_X_0();	X (-) axis distance data	50-1000
--	-----------------	--------------------------	---------

## **Other Commands**

All other common Arduino and ESP8266 commands can be used in the library except for the following which will interfere with drone operation.

delay();	
analogWrite();	
Tone();	