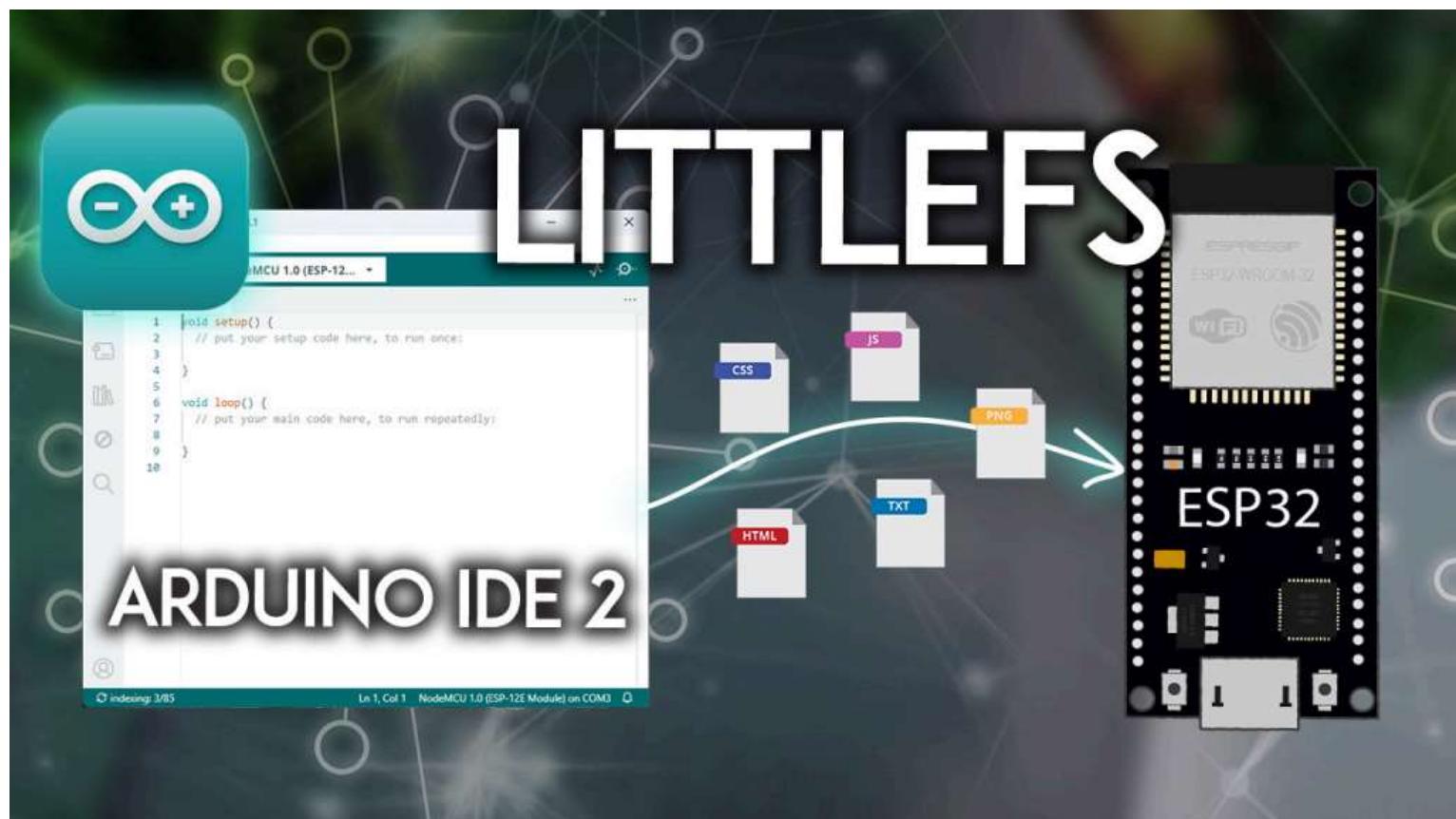


Arduino IDE 2: Install ESP32 LittleFS Uploader (Upload Files to the Filesystem)

In this guide, you'll learn how to upload files to the ESP32 LittleFS Filesystem using Arduino IDE 2 (2.2.1 or a higher version). We'll install an uploader plugin that will add a new menu to the Arduino IDE. This plugin is also compatible with the ESP8266 and Raspberry Pi Pico boards.



Using an ESP8266? Follow this tutorial instead: Arduino IDE 2: Install ESP8266 NodeMCU LittleFS Uploader (Upload Files to the Filesystem).

Table of Contents

- [Introducing LittleFS](#)
- [Installing LittleFS Filesystem Uploader Plugin](#)
 - [Windows Instructions](#)
 - [Mac OS X Instructions](#)
- [Uploading Files to ESP32 using the Filesystem Uploader](#)

- [Testing the ESP32 LittleFS Uploader](#)

If you're still using **Arduino 1.8**, you can follow this tutorial instead: [ESP32: Upload Files to LittleFS using Arduino IDE \(legacy\)](#).

Introducing LittleFS

LittleFS is a lightweight filesystem created for microcontrollers that lets you access the flash memory as you do in a standard file system on your computer, but it's simpler and more limited. You can read, write, close, and delete files. Using LittleFS with the ESP32 boards is useful to:

- Create configuration files with settings;
- Save data permanently;
- Create files to save small amounts of data instead of using a microSD card;
- Save HTML, CSS, and JavaScript files to build a web server;
- [Save images, figures, and icons](#);
- And much more.

Installing the LittleFS Uploader Plugin on Arduino IDE 2

To upload files to the ESP32 on [Arduino IDE 2](#), we'll use [this LittleFS Uploader plugin](#) that is compatible with Arduino 2.2.1 or higher and can be used with the ESP32, [ESP8266](#), and [Raspberry Pi Pico boards](#).

Windows Instructions

Follow the next steps to install the filesystem uploader if you're using Windows ([click here for MacOS instructions](#)):

- 1) Go to the [releases page](#) and click the .vsix file to download.

Release 1.1.5 - ESP32, ESP8266, and Pico Support

Latest

New in this release: ESP32 family board name fixes

▼ Assets

3

[arduino-littlefs-upload-1.1.5.vsix](#)

1.03 MB

last week

[Source code \(zip\)](#)

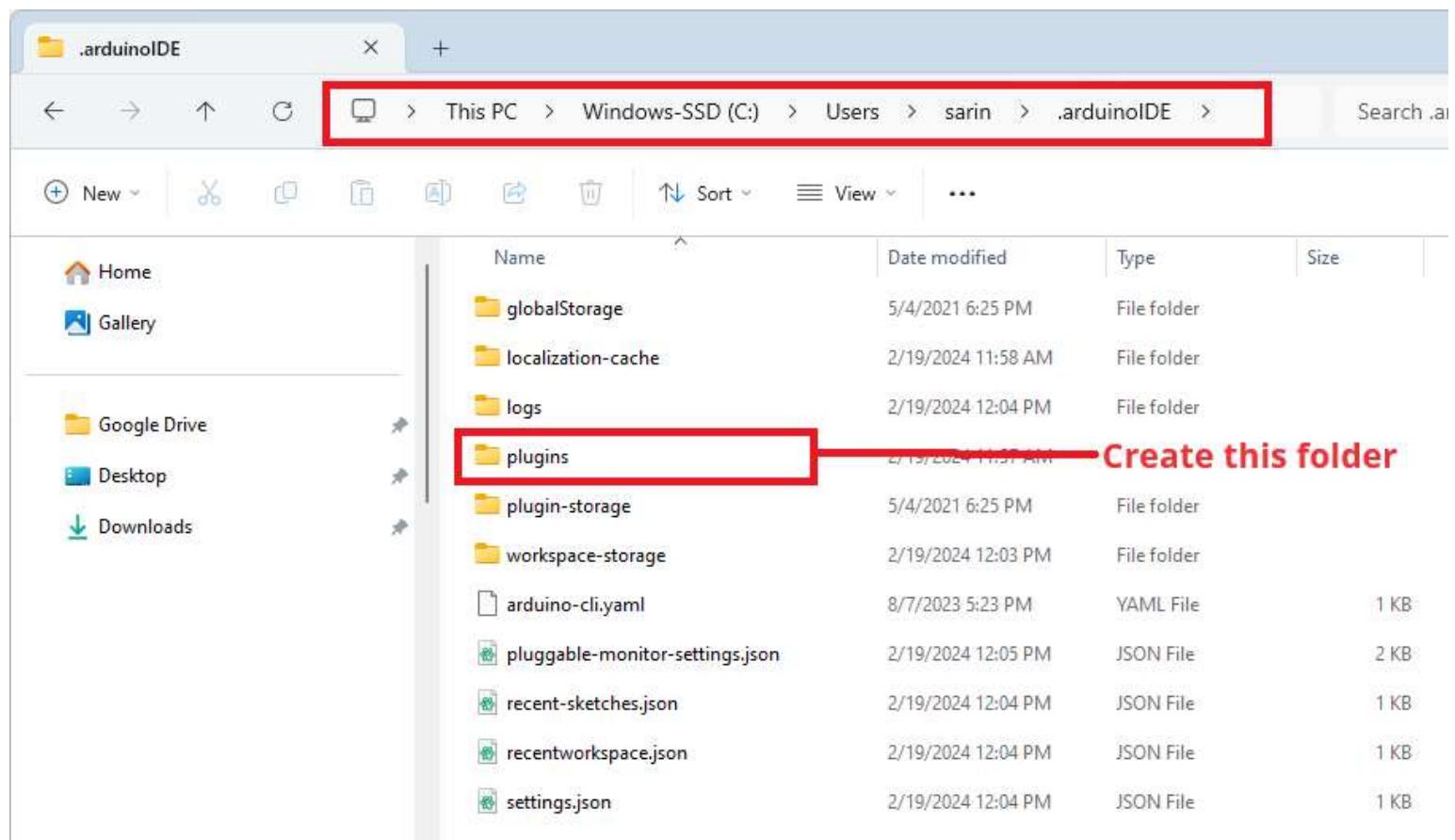
last week

[Source code \(tar.gz\)](#)

last week

4 people reacted

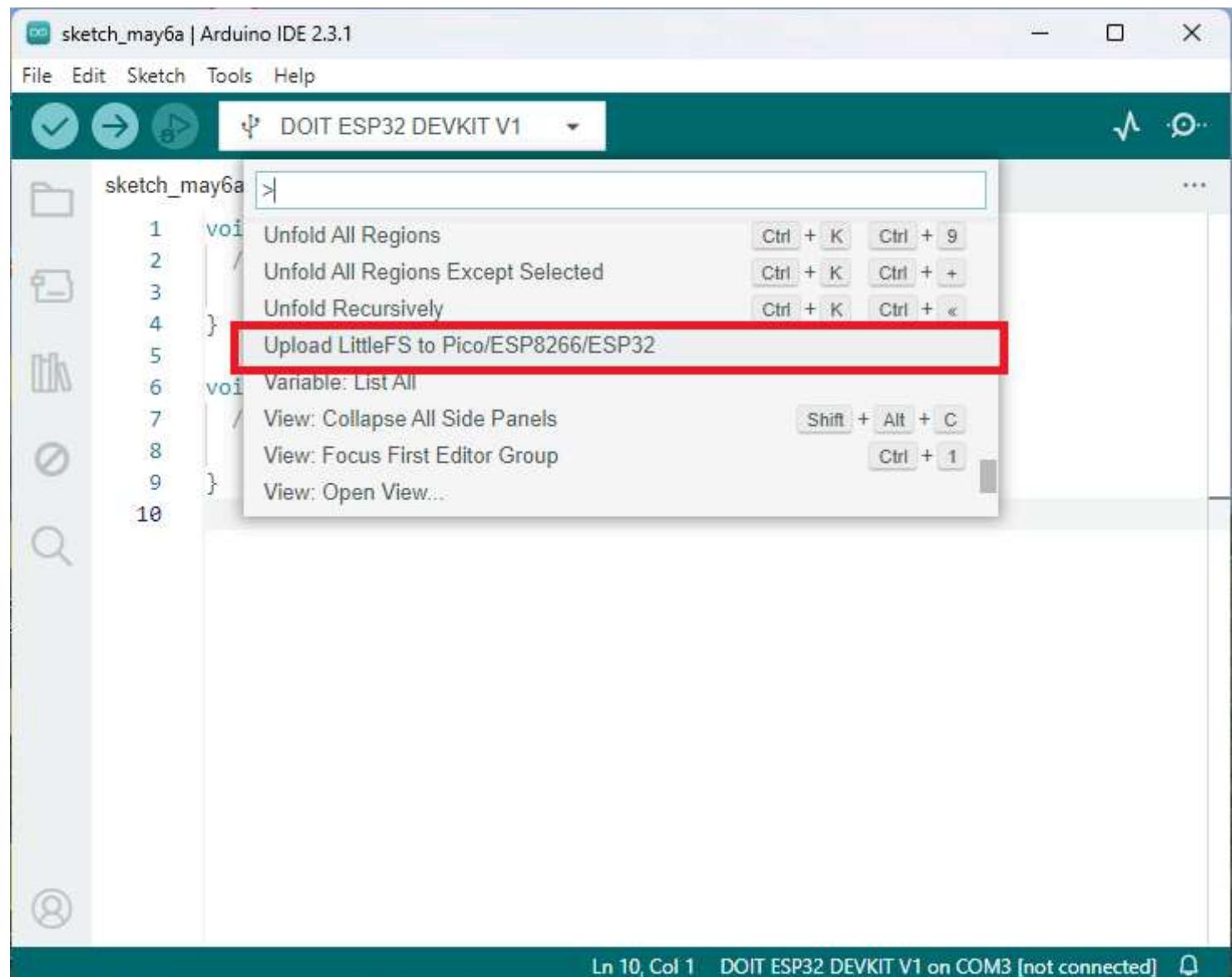
- 2) On your computer, go to the following path: C:\Users\<username>\.arduinoIDE\. Create a new folder called *plugins* if you haven't already.



- 3) Move the .vsix file you downloaded previously to the *plugins* folder (remove any other previous versions of the same plugin if that's the case).



4) Restart or open the Arduino IDE 2. To check if the plugin was successfully installed, press [Ctrl] + [Shift] + [P] to open the command palette. An instruction called '**Upload Little FS to Pico/ESP8266/ESP32**' should be there (just scroll down or search for the name of the instruction).



That means the plugin was installed successfully. Proceed to [this section](#) to test the filesystem uploader plugin.

Mac OS X Instructions

Follow the next steps to install the filesystem uploader if you're using Mac OS X:

- 1) Go to the [releases page](#) and click the .vsix file to download.

Release 1.1.5 - ESP32, ESP8266, and Pico Support

New in this release: ESP32 family board name fixes

▼ Assets 3

File	Size	Last Updated
arduino-littlefs-upload-1.1.5.vsix	1.03 MB	last week
Source code (zip)		last week
Source code (tar.gz)		last week

4 people reacted

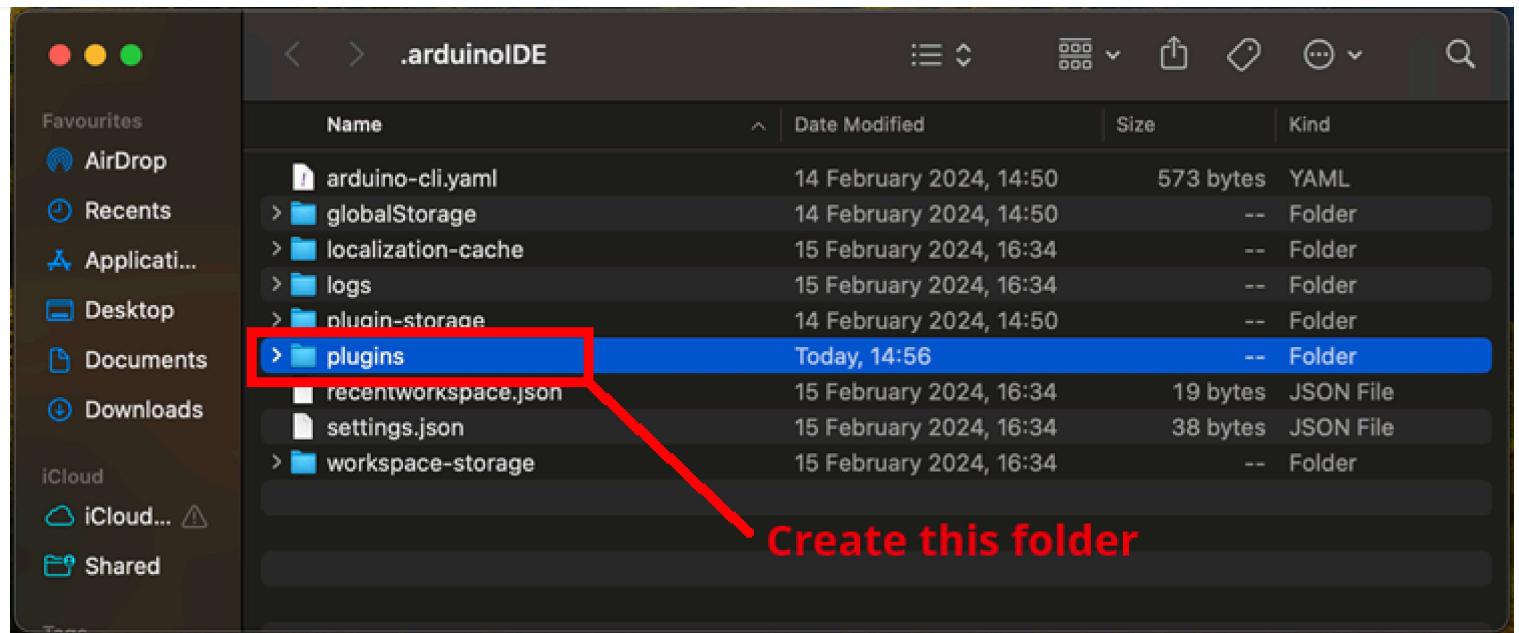
- 2) In Finder, type `~/.arduinoIDE/` and open that directory.

Q ~/.arduinoIDE/ – ~/.arduinoIDE

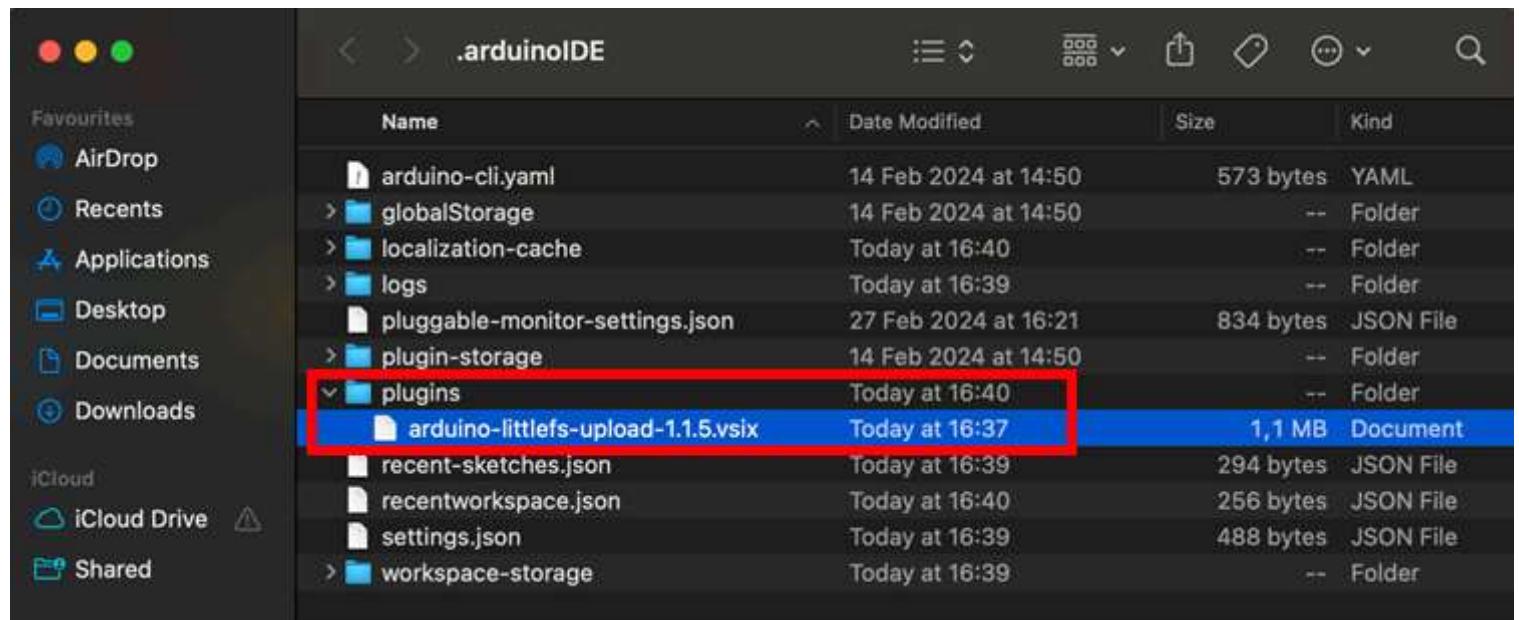
Folders Show More ▾

.arduinoIDE folder · Modified 14/02/2024, 14:50
--

- 3) Create a new folder called *plugins*.



4) Move the .vsix file to the *plugins* folder (remove any other previous versions of the same plugin if that's the case).



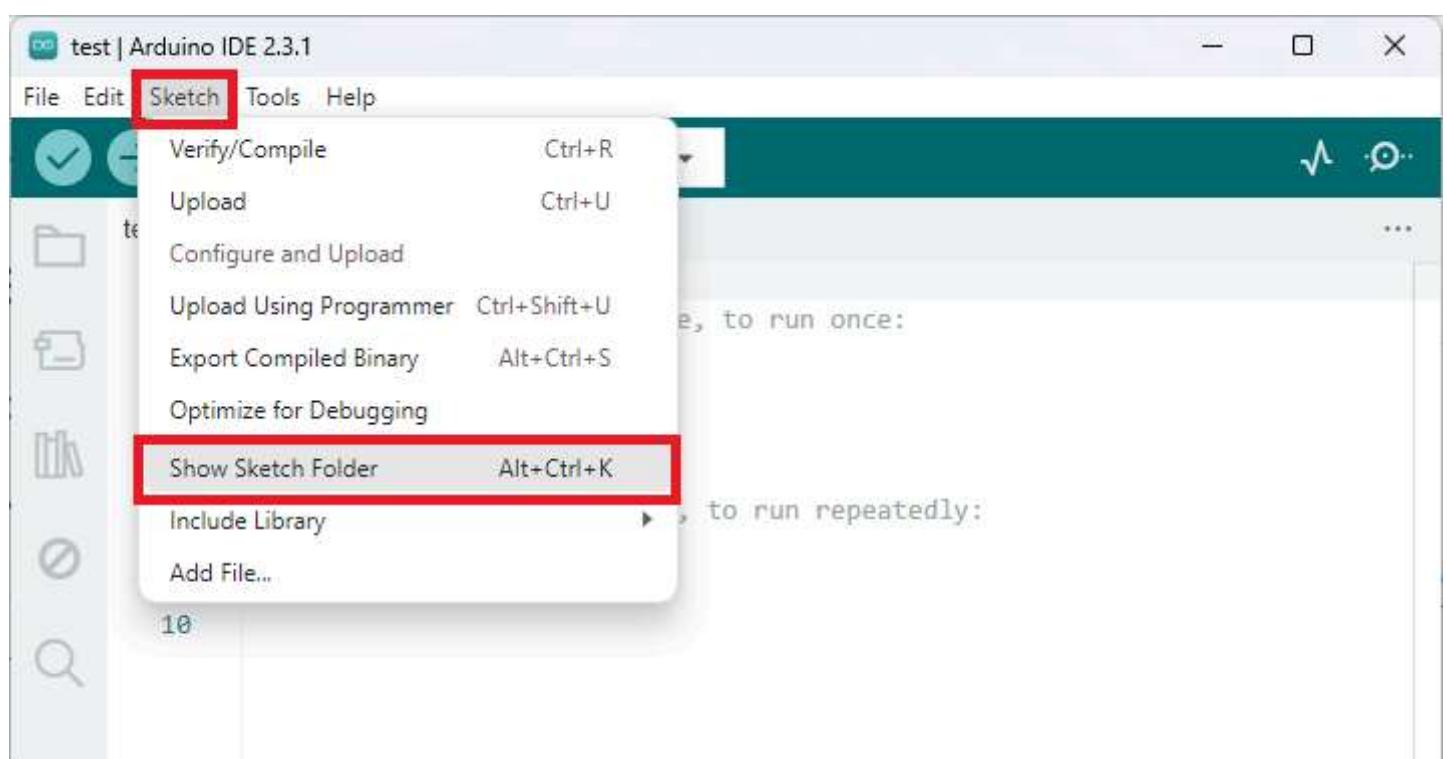
5) Restart or open the Arduino IDE 2. To check if the plugin was successfully installed, press [⌘] + [Shift] + [P] to open the command palette. An instruction called '**Upload LittleFS to Pico/ESP8266/ESP32**' should be there (just scroll down or search for the name of the instruction).

The screenshot shows the Arduino IDE 2.3.2 interface. The title bar reads "sketch_may6a | Arduino IDE 2.3.2". In the top menu, "Sketch" is selected. A dropdown menu is open under "Sketch", showing options like "Verify/Compile", "Upload", "Configure and Upload", "Upload Using Programmer", "Export Compiled Binary", "Optimize for Debugging", and "Show Sketch Folder". The "Show Sketch Folder" option is highlighted with a red box. The main code editor window displays a simple sketch named "sketch_may6a.ino". The code contains two functions: "void setup()" and "void loop()". The "setup" function has a single line of code: "// put". The "loop" function also has a single line of code: "// put your main code here, to run repeatedly.". To the right of the code editor, there is a toolbar with icons for file operations like Open, Save, and Print.

Uploading Files to ESP32 using the Filesystem Uploader in Arduino IDE 2

To upload files to the ESP32 LittleFS filesystem, follow the next instructions.

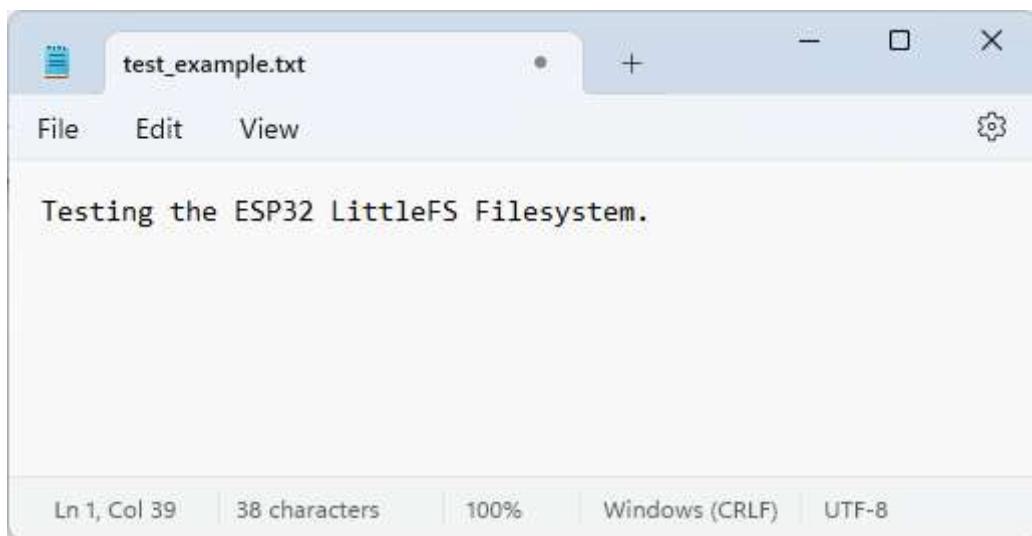
- 1) Create an Arduino sketch and save it. For demonstration purposes, you can save an empty sketch.
- 2) Then, open the sketch folder. You can go to **Sketch > Show Sketch Folder**. The folder where your sketch is saved should open.



- 3) Inside that folder, create a new folder called **data**.



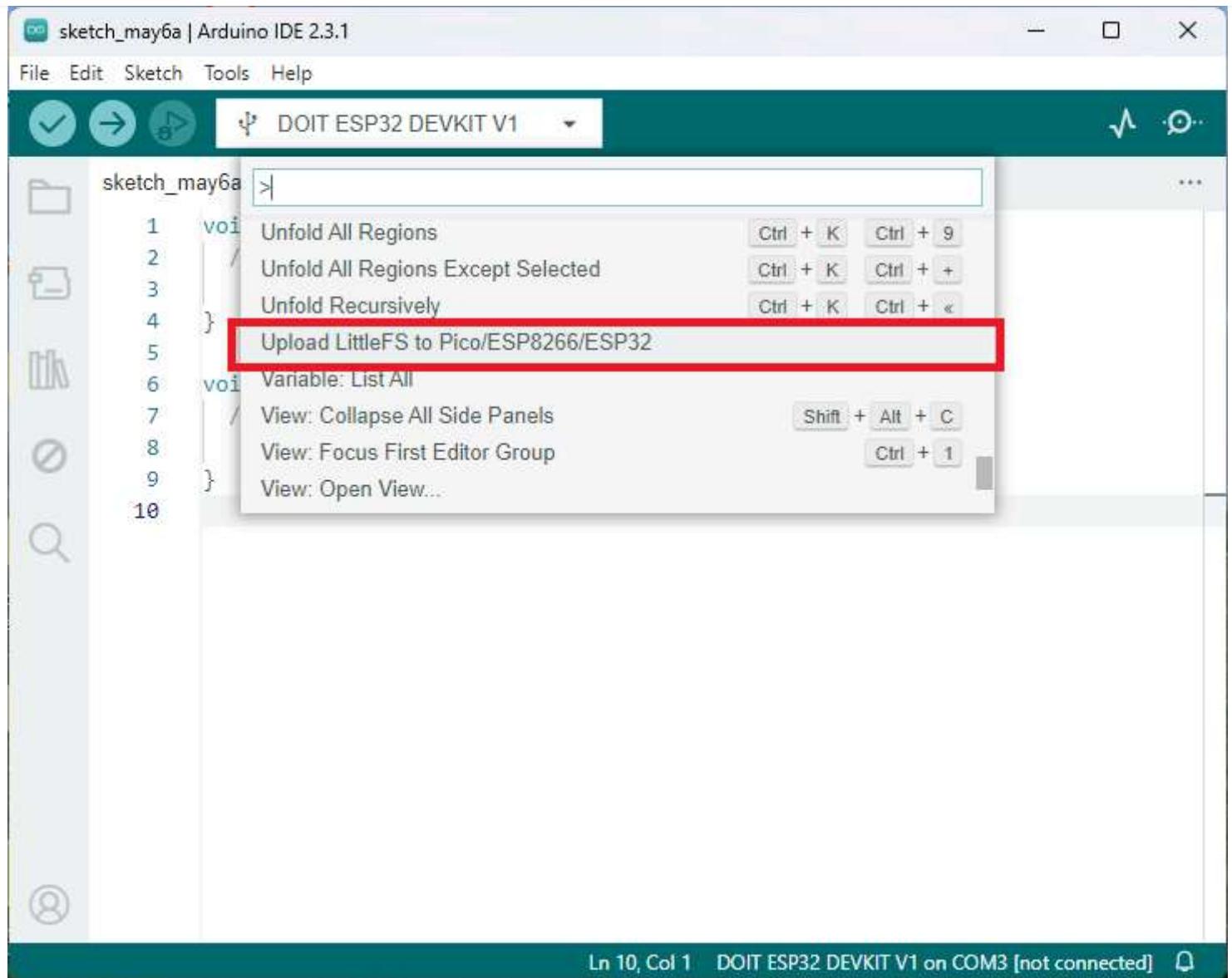
4) Inside the **data** folder is where you should put the files you want to upload to the ESP32 filesystem. As an example, create a **.txt** file with some text called **test_example.txt** (and save it inside the **data** folder).



5) Make sure you have the right board (**Tools > Board**) and COM port selected (**Tools > Port**).

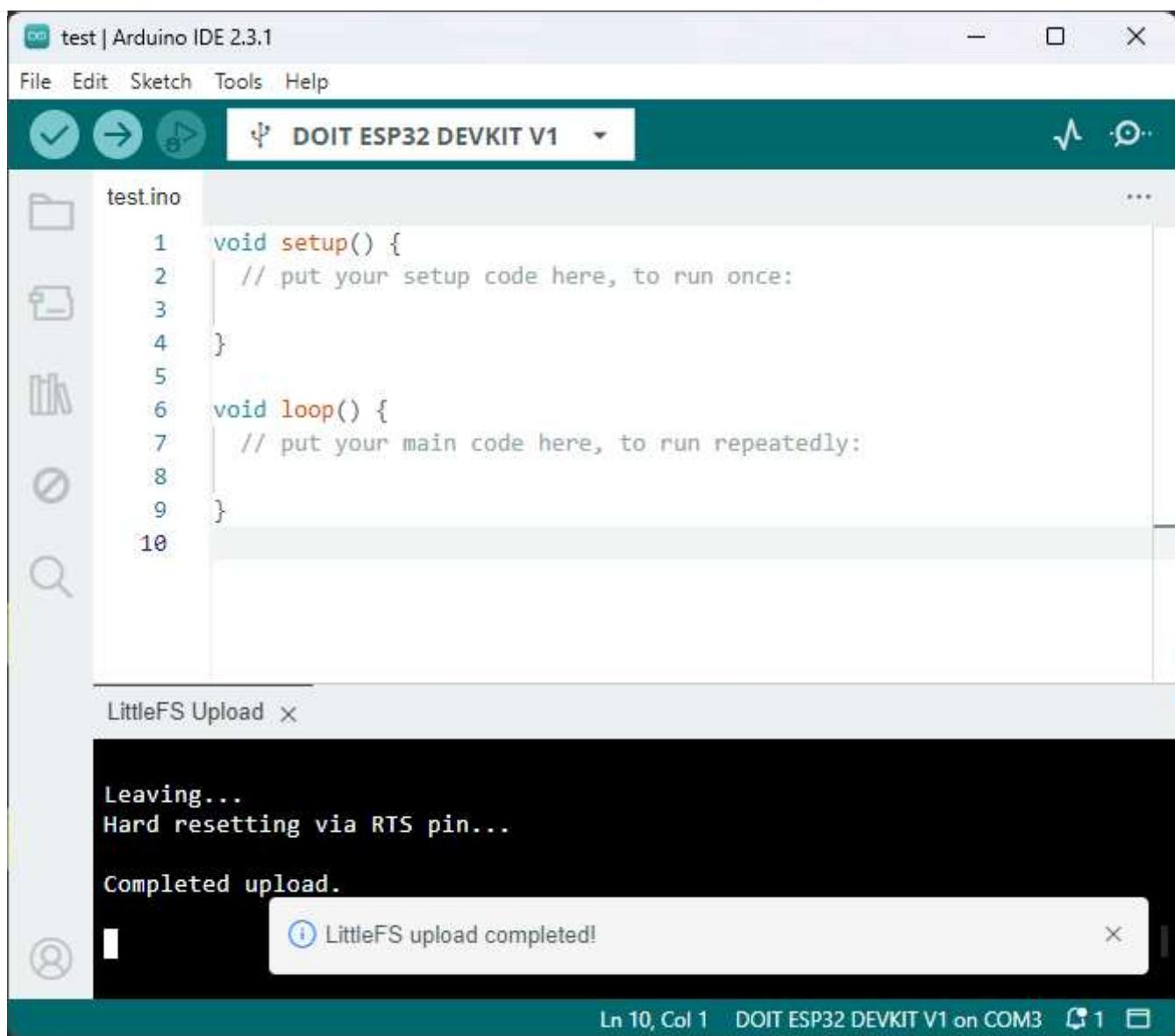
6) Depending on the ESP32 board selected, you may need to select the desired flash size (some boards don't have that option, don't worry). In the Arduino IDE, in **Tools > Flash size**, select the desired flash size (this will depend on the size of your files).

7) Then, upload the files to the ESP32 board. Press **[Ctrl] + [Shift] + [P]** on Windows or **[⌘] + [Shift] + [P]** on MacOS to open the command palette. Search for the **Upload LittleFS to Pico/ESP8266/ESP32** command and click on it.



Important: ensure the Serial Monitor is closed. Otherwise, the upload will fail.

After a few seconds, you should get the message “**Completed upload.**”. The files were successfully uploaded to the ESP32 filesystem.



Troubleshooting

If you get the following error message “[ERROR: No port specified, check IDE menus](#)”, restart the Arduino IDE, and try again.

Testing the ESP32 LittleFS Uploader

Now, let's check if the file was saved into the ESP32 filesystem. Upload the following code to your ESP32 board. This code will read the contents of the `.txt` file you saved previously on LittleFS.

```
#include "LittleFS.h"  
  
void setup() {  
    Serial.begin(115200);
```

```

if(!LittleFS.begin()){
    Serial.println("An Error has occurred while mounting LittleFS");
    return;
}

File file = LittleFS.open("/test_example.txt", "r");
if(!file){
    Serial.println("Failed to open file for reading");
    return;
}

Serial.println("File Content:");
while(file.available()){
    Serial.write(file.read());
}
file.close();
}

void loop() {
}

```

[View raw code](#)

After uploading, open the Serial Monitor at a baud rate of 115200.



Press the ESP32 on-board “RST” button. It should print the content of your *.txt* file in the Serial Monitor.

The screenshot shows the Arduino IDE 2.3.1 interface with the DOIT ESP32 DEVKIT V1 plugin installed. The code editor displays a sketch named 'test.ino' containing C++ code for interacting with the LittleFS library. The Serial Monitor window shows the output of the uploaded code, including the file content 'Testing the ESP32 LittleFS Filesystem.' which is highlighted with a red box.

```
#include "LittleFS.h"

void setup() {
    Serial.begin(115200);

    if(!LittleFS.begin()){
        Serial.println("An Error has occurred while mounting LittleFS");
        return;
    }

    File file = LittleFS.open("/test_example.txt", "r");
    if(!file){
        Serial.println("Failed to open file for reading");
        return;
    }
}
```

Message (Enter to send message to 'DOIT ESP32 DEVKIT V1')
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:1184
load:0x40078000,len:13260
load:0x40080400,len:3028
entry 0x400805e4
File Content:
Testing the ESP32 LittleFS Filesystem.

You've successfully uploaded files to the ESP32 filesystem using the plugin.

Wrapping Up

In this tutorial, we've shown you how to upload files to the ESP32 LittleFS filesystem on [Arduino IDE 2](#) using an uploader plugin.

We've shown you how to upload a *.txt* file, but you can upload other file formats like HTML, CSS, and Javascript files to build a web server, images, or small icons, save configuration files, etc.

To learn more about the ESP32, check our resources:

- [Learn ESP32 with Arduino IDE \(eBook\)](#)
- [SMART HOME with Raspberry Pi, ESP32, ESP8266](#)
- [Build Web Servers with ESP32 and ESP8266](#)
- [Firebase Web App with ESP32 and ESP8266](#)
- [More ESP32 Tutorials and Guides](#)

Thanks for reading.

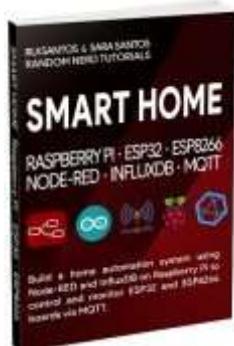
PCBWay
PCB Fabrication & Assembly

ONLY \$5 for 10 PCBs

✓ 24-hour Build Time ✓ Quality Guaranteed
✓ Most Soldermask Colors:
█ █ █ █ █ █ █ █ █ █

Order now

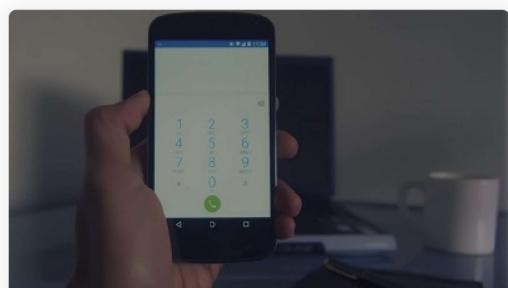
www.pcbway.com



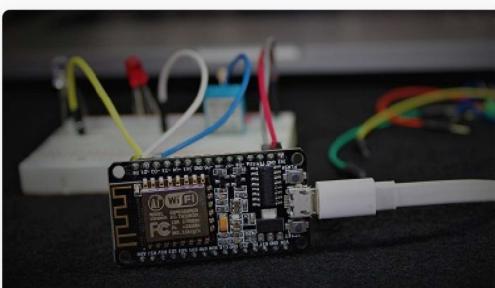
SMART HOME with Raspberry Pi, ESP32, ESP8266 [eBook]

Learn how to build a home automation system and we'll cover the following main subjects: Node-RED, Node-RED Dashboard, Raspberry Pi, ESP32, ESP8266, MQTT, and InfluxDB database [DOWNLOAD »](#)

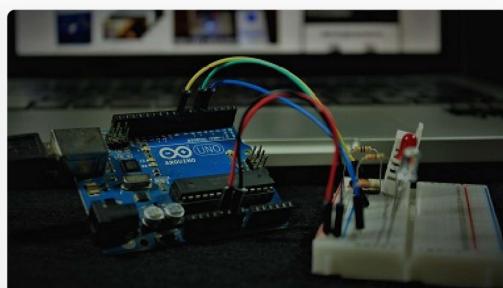
Recommended Resources



[Build a Home Automation System from Scratch](#)



[Home Automation using ESP8266 eBook and video](#)



[Arduino Step-by-Step Projects » Build 25 Arduino](#)

» With Raspberry Pi,
ESP8266, Arduino, and
Node-RED.

[course](#) » Build IoT and
home automation projects.

projects with our course,
even with no prior
experience!

What to Read Next...

[ESP8266 NodeMCU: Getting Started
with InfluxDB](#)

[ESP32 MQTT – Publish
DHT11/DHT22 Temperature and
Humidity Readings \(Arduino IDE\)](#)

[Programming ESP32 with Atom Text
Editor and PlatformIO IDE](#)

**Enjoyed this project? Stay updated by subscribing our
newsletter!**

Your Email Address

[SUBSCRIBE](#)

36 thoughts on “Arduino IDE 2: Install ESP32 LittleFS Uploader (Upload Files to the Filesystem)”



Geraldit Griffiths

May 6, 2024 at 4:57 pm

Thank you so much for this!

As someone who used LittleFS for a couple of projects I did way-back, I found it so frustrating to read messages online that said how difficult it would be to support LittleFS in IDE2, if – indeed – it would be possible at all.

I will read your article with interest and make rude cyber-gestures at the nay-sayers!

[Reply](#)



Sara Santos

May 7, 2024 at 8:56 am

Support for the ESP32 was added very recently to the Arduino IDE 2.

Regards,

Sara

[Reply](#)



Mark D

May 6, 2024 at 5:48 pm

Howdy Again

As usual a great tutorial which worked floorlessly and has answered alot of questions that I had about sketches that needed the data uploaded to the board.

It took me a long time to dicover that older older arduino Ide supported this function and now you have explained wonderfully how to upload LittleFs in latest Arduino IDE.

I thankyou and the team and look forward to your tutorials and the information from your site.

Thanks from the UK

Mark D

[Reply](#)



Eddie

May 6, 2024 at 10:22 pm

Wow, finally the LittleFS is available on Arduino IDE 2.0! Excelent post, by the way, the quality of the tutorials remains the same when I found this site, 5 yrs ago: excelent!

[Reply](#)



Sara Santos

May 7, 2024 at 9:07 am

Thank you 😊

Regards,

Sara

[Reply](#)



meebox

May 10, 2024 at 4:10 am

Is there any plugins that support SPIFFS? Thanks.

[Reply](#)



Sara Santos

May 10, 2024 at 9:53 am

For spiffs, you have to use the old Arduino IDE.

<https://randomnerdtutorials.com/install-esp32-filesystem-uploader-arduino-ide/>

Regards,

Sara

[Reply](#)



Martin Maly

May 12, 2024 at 4:31 pm

See here: <https://github.com/espx-cz/arduino-spiffs-upload>

[Reply](#)



meebox

May 13, 2024 at 12:21 pm

cool.

[Reply](#)



Adent

May 13, 2024 at 6:45 am

Use this for Arduino IDE V2: <https://github.com/espx-cz/arduino-spiffs-upload>

[Reply](#)



Jop van der Werff

May 10, 2024 at 7:22 am

Hi Sara and Rui,

I was very happy to see your new tutorial about upload LittleFS now is implemented in the Arduino IDE 2.

I'm working on MAC and stuck on one problem. My upload to LittleFS ends with this error:

A fatal error occurred: Could not open /dev/cu.usbserial-1420, the port doesn't exist
ERROR: Upload failed, error code: 2

Maybe it has to do with an open Serial-monitor connection. In the old IDE 1, you had to close this connection before starting upload of data and this was done very simple by just closing the (distinct) window were Serial-monitor was dumping it's data.

In the IDE 2 it's not very clear how to close Serial-monitor. As Serial monitor is in the bottom panel of my sketch-window.

Here the complete log output of my uploading to LittleFS:

Using partition: default

Building LittleFS filesystem

```
/Users/jop/Library/Arduino15/packages/esp32/tools/mklittlefs/3.0.0-gnu12-
dc7f933/mklittlefs -c
```

```
/Users/jop/STACK/ArduinoSketches/@KENNIS/LittleFS/TestLittleFS/data -p 256 -b
4096 -s 1441792 /var/folders/vd/rwhbr1s5j77gkn7yq_qfmlm0000gn/T/tmp-7661-
```

q2cK0GA8qGrt-.littlefs.bin

/test_example.txt

Uploading LittleFS filesystem

```
/Users/jop/Library/Arduino15/packages/esp32/tools/esptool_py/4.5.1/esptool –chip esp32 –port /dev/cu.usbserial-1420 –baud 921600 –before default_reset –after hard_reset write_flash -z –flash_mode dio –flash_freq 80m –flash_size detect 2686976 /var/folders/vd/rwhbr1s5j77gkn7yq_qfmlm0000gn/T/tmp-7661-q2cK0GA8qGrt-.littlefs.bin  
esptool.py v4.5.1  
Serial port /dev/cu.usbserial-1420
```

A fatal error occurred: Could not open /dev/cu.usbserial-1420, the port doesn't exist

ERROR: Upload failed, error code: 2

The port cu.usbserial-1420 does exist, as I used it to upload my test sketch.
My macOS is Sonoma.

With kind regards,

Jop

[Reply](#)



Martin Maly

May 12, 2024 at 4:30 pm

Close Serial terminal first, then try it again.

[Reply](#)



Martin Maly

May 12, 2024 at 2:04 pm

Here is a SPIFFS clone: <https://github.com/espx-cz/arduino-spiffs-upload>

[Reply](#)



Sara Santos

May 13, 2024 at 9:49 am

Great!

Thanks for sharing.

Regards,

Sara

[Reply](#)



meebox

November 22, 2024 at 3:13 am

With both the ESP32 and ESP8266 package installed on the same IDE, the plugin always finds the mkspiffs tool in the ESP32 package. This would be a problem while uploading SPIFFS to ESP8266 since the mkspiffs tool is not compatible with ESP8266. I've made a [PR](#) to fix it.

[Reply](#)



Doug Leppard

May 25, 2024 at 6:52 pm

FYI, I got this to work in the sample with a XIAO ESP32C3.

Thanks

[Reply](#)



Kevin B

May 31, 2024 at 5:03 pm

I have Arduino 2.3.2 on Windows. I followed the instructions and reopened the IDE, did the Fn+CTRL+P and the file uploader is not on the list!

First I put the latest release (arduino-littlefs-upload-1.1.7.vsix) in the plugins folder, with the following path: C:\Users\kevin.arduinoIDE\plugins. When it didn't work, I deleted that file and replaced it with the version mentioned in the tutorial (arduino-littlefs-upload-1.1.5.vsix) but it still did not work. The board I selected is ESP32 Dev Module. Any ideas?

[Reply](#)



Freddy

August 30, 2024 at 9:50 am

i have the same problem like you. Sometimes i get the message “Upload LittleFS to Pico/ESP8266/esp32” on the list, but only once. When i try a second time, the message is no more there.

[Reply](#)



Michel

June 8, 2024 at 5:14 pm

Hello.

Here is a list created by the community containing plugins compatible with the Arduino IDE 2.x: github.com/MicSG-dev/list-of-compatible-plugins-with-arduino-ide-2-x

[Reply](#)



Michel

June 20, 2024 at 9:33 am

Link: github.com/MicSG-dev/list-of-compatible-plugins-with-arduino-ide-2-x/

[Reply](#)



qualia universe

June 20, 2024 at 5:09 am

Only site that gives actual solution.

[Reply](#)



AI

June 20, 2024 at 2:12 pm

Hello,

Thanks for the useful tutorials you keep sharing 😊

I used the tool. Upload went well but it erased (overwrote ?) the NVS preferences on the board.

Do you know how to upload /data to littleFS and keep “preferences” untouched ?

Thanks

[Reply](#)



Sara Santos

June 20, 2024 at 2:33 pm

Hi.

Thanks for your feedback.

I didn't know about that.

I searched for a while and found people with the same issue, but no answers... I don't know how to fix that issue.

If you find a solution, please share.

Regards,

Sara

[Reply](#)



AI

June 23, 2024 at 11:18 am

I tried on another board (Mini D1 ESP32) and this time upload did not erase preferences. Same Arduino IDE and tool version.

I could not find a systematic way to reproduce the problem ! It is strange.

[Reply](#)



RU183

July 7, 2024 at 12:50 am

Activating extension 'arduino-littlefs-upload' failed: Cannot find module 'c:\Users.....arduinoIDE\plugins\arduino-littlefs-upload-1.1.8\out\extension.js' Require stack: - C:\Users.....\AppData\Local\Programs\arduino-ide\resources\app\lib\backend\plugin-host.js

If anyone has this problem, you need to change the file extension extension.ts from the src folder to js and move the file to the created “out” folder

ARDUINO IDE 2.3.2

esptool 4.5.1

ESP32 S3 WROOM

For Gyver project WIFI panel matrix on WS2812b 1024 LEDs

[Reply](#)



Fu

August 1, 2024 at 5:44 am

The previous steps were executed according to your steps, but an error message appeared :

ERROR: Partition entry not found in csv file!

How to fix it?

[Reply](#)



Sara Santos

August 6, 2024 at 8:41 am

Hi.

When do you get that error?

What board are you selecting in Tools > Board?

Regards,

Sara

[Reply](#)



Jan E

August 11, 2024 at 8:38 am

Try replacing partitions.csv with this one: github.com/espressif/arduino-esp32/blob/master/tools/partitions/default.csv

[Reply](#)



hans

August 5, 2024 at 5:55 pm

As always, this place is my first rescue whenever I run into an Arduino related problem and once again I found a crystal clear tutorial that works. Thank you so much!

[Reply](#)



Edzman

August 29, 2024 at 9:24 pm

I have a question about Step #7. Before asking, I would like to say this website has helped me a lot when building my projects. Thanks in advance for your help.

QUESTION:

What would happen if we DON't perform step #7:

“Then, upload the files to the ESP32 board. Press [Ctrl] + [Shift] + [P] on Windows or [⌘] + [Shift] + [P] on MacOS to open the command palette. Search for the Upload LittleFS to Pico/ESP8266/ESP32 command and click on it.”

...and just programatically write:

```
File file = LittleFS.open("example_data.json", "w");
```

then we serialize some JSON data to the file “example_data.json”

and finally we close the file using “file.close();”

Wouldnt that technically CREATE the file if it doesnt exist?

Or its mandatory to ALWAYS perform manually step#7 in ArduinoIDE 2.0?

I mean, I know we need to create a “data” folder at the root of our sketch, but do we also NEED to manually create the file we want to manipulate and ALSO perform step #7?

Can step #7 be performed programatically inside the sketch?

Thank you!!

[Reply](#)



Jorge Vila

September 9, 2024 at 7:52 pm

I have a problem

The plugin doesn't work

Wisdoms 10 (PT_BR) + Arduino IDE 2.3.2 + arduino-littlefs-upload-1.1.8.vsix

When I press SHIFT+CTRL+P some options appear, but not “Upload Little FS to Pico/ESP8266/ESP32”

Any tips to solve the problem?

or

Any other way to send data to spiffs

[Reply](#)



Sara Santos

September 9, 2024 at 9:34 pm

Hi.

Make sure you're placing the file on the right folder.

It works well for us.

If it doesn't work, it's better to post an issue in the developer github page:

<https://github.com/earlephilhower/arduino-littlefs-upload/issues>

Regards,

Sara

[Reply](#)



Ramon

September 19, 2024 at 10:22 am

I have the same problem, I have checked the folder and its destination and the file.

but the option does not appear.

did you solve the problem??

Arduino 2.3.2

[Reply](#)



flat four

January 10, 2025 at 12:16 pm

I have the same problem. Has it been solved?

I have followed the instructions carefully and placed the “arduino-littlefs-upload-1.5.3.vsix” in the .arduinoIDE/plugins directory. restarted the IDE, hit CTRL/SHIFT/P and cannot see the upload function in the palette.

[Reply](#)



Juan

October 22, 2024 at 5:35 am

Hello, excellent tutorial that has helped me a lot to solve doubts and problems that I had, I have managed to load the files into the esp32 memory but I cannot read them. I can read the files that I create directly from the sketch, but not the imported ones. Any advice?

All the best

[Reply](#)



Yves

November 21, 2024 at 5:04 pm

Hello, i have installed the plugin on arduino ide 2.3 (windows11). When i use the command “upload little fs” i get an error about the baud rate option :

esptool: error: argument –baud/-b: invalid arg_auto_int value: ‘NaN’

ERROR: Upload failed, error code: 2

Any idea ?

Best Regards

Yves

[Reply](#)

Leave a Comment

Name *

Email *

Website

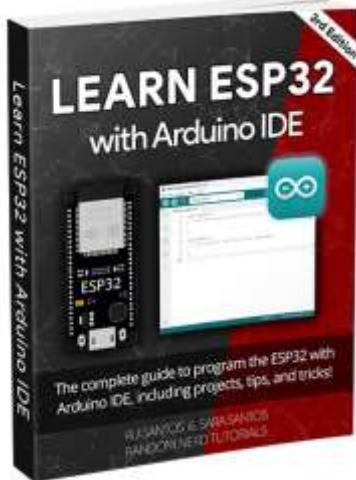
Notify me of follow-up comments by email.

Notify me of new posts by email.

Post Comment

Affiliate Disclosure: *Random Nerd Tutorials* is a participant in affiliate advertising programs designed to provide a means for us to earn fees by linking to Amazon, eBay, AliExpress, and other sites.

We might be compensated for referring traffic and business to these companies.

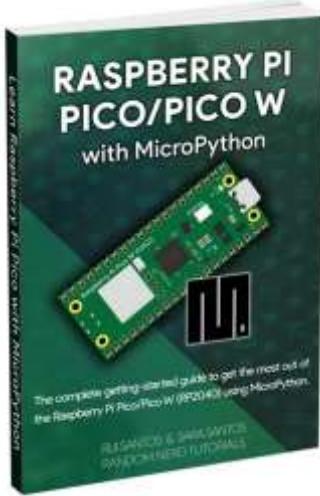


[Learn ESP32 with Arduino IDE eBook »](#)

Complete guide to program the ESP32 with Arduino IDE!



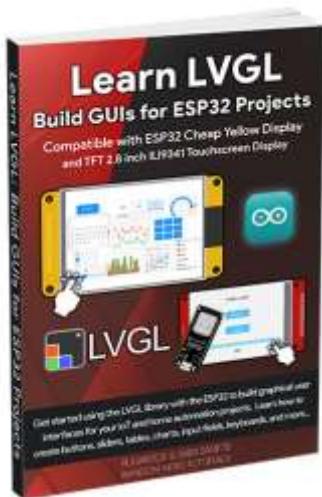
[SMART HOME with Raspberry Pi, ESP32, and ESP8266 »](#) learn how to build a complete home automation system.



Learn Raspberry Pi Pico/Pico W with

MicroPython » The complete getting

started guide to get the most out of the the
Raspberry Pi Pico/Pico W (RP2040)
microcontroller board using MicroPython
programming language.



🔥 Learn LVGL: Build GUIs for ESP32

Projects » Learn how to build Graphical

User Interfaces (GUIs) for ESP32 Projects
using LVGL (Light Versatile Graphics
Library) with the Arduino IDE.

[About](#) [Support](#) [Terms and Conditions](#) [Privacy Policy](#) [Refunds](#) [Complaints' Book](#)
[MakerAdvisor.com](#) [Join the Lab](#)

Copyright © 2013-2025 · RandomNerdTutorials.com · All Rights Reserved