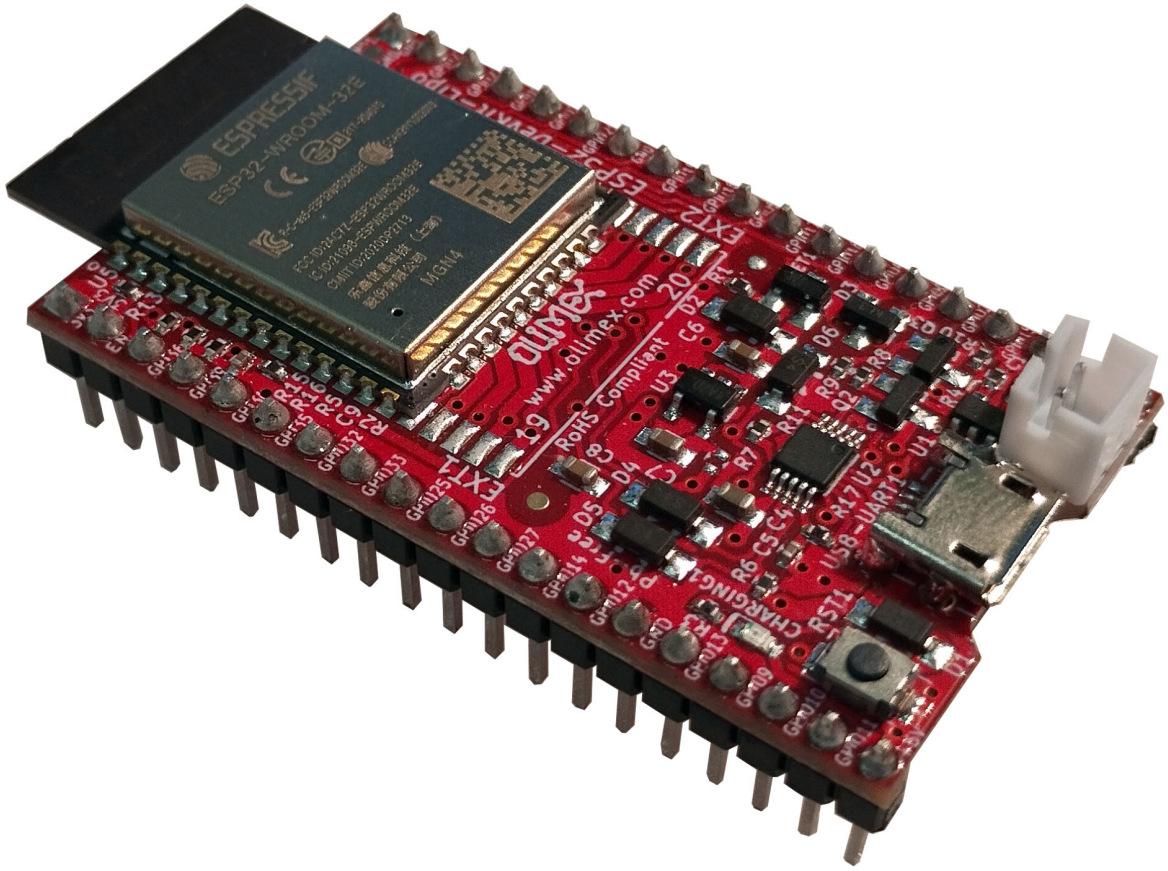
=



ESP32-DevKit-LiPo

User's Manual

Document revision 1.0, Devember 2024

www.olimex.com

Table of Contents

1. What is ESP32-DevKit-LiPo

ESP32-DevKit-LiPo is an Open Source Hardware development board that incorporates an ESP32 module. The ESP32-DevKit-LiPo board is designed and manufactured by Olimex, while the ESP32 module is designed and manufactured by Espressif systems. The design is inspired by ESP32-CoreBoard (ESP32-DevKitC) and has the same pinout, but adds Li-Po battery connector and charger. The board can operate on a Li-Po power when external power supply goes missing, allowing for handheld applications and increasing availability and reliability.

The ESP32 modules are extremely popular WIFI and BT modules due to their size, price, and very good documentation. The typical use is for WiFi and BT enabled devices. ESP32- supports low power modes including deep sleep that goes as low as 10uA.

The design has three variants, there is external antenna variant and variant with ESP32 module with extra PSRAM.

The ESP32-DevKit-LiPo board has the following features:

- ESP32-WROOM-32E (or ESP32-WROOM-32UE or ESP32-WROVER-E module, depending on variant)
- Reset button
- Micro USB connector for powering and programming
- Built-in LiPo battery charger
- LiPo battery connector
- Battery measurement and external power sense circuits
- Two columns of pins with headers soldered for easy access to all the board’s GPIOs
- Low-power design for extended operation on battery
- PCB dimensions: (1.9×1.1)" ~ (4.8×2.8)cm
- Operating temperature: -40+85C

1.1 ESP32-DevKit-LiPo variants

The board has 3 variants:

- **ESP32-DevKit-LiPo**
- **ESP32-DevKit-LiPo-EA**
- **ESP32-WROVER-DevKit-LiPo**

All board variants work in the industrial temperature range (-40+85 degrees Celsius).

ESP32-DevKit-LiPo comes with ESP32-WROOM-32E-N4 module;

ESP32-DevKit-LiPo-EA uses ESP32-WROOM-32UE-N4 module and comes with external 3dB antenna compatible with the module's u.FL connector (ANTENNA-WIFI-U.FL);

ESP32-WROVER-DevKit-LiPo uses ESP32-WROVER-E-N4R8 module that has extra 8MB PSRAM. Notice that the WROVER module uses two extra GPIO pins for the PSRAM. Refer to the schematic and the notes on it.

1.2 Board use requirements

You only need a fitting USB cable and a personal computer. The board requires USB micro connector. Usually only such cable is required:

https://www.olimex.com/Products/Components/Cables/USB-CABLE-A-MICRO-1.8M/

The computer needs software compatible with ESP32 modules. Most commonly used tools are ESP-IDF and Arduino IDE with ESP32 package. You can use ESP32-DevKit-LiPo with any software tool that supports the main ESP32 module.

The board has auto-programming enabled, you don’t need to force bootloader mode. Make sure not to influence the bootstrap pins since this might influence the auto-programming feature.

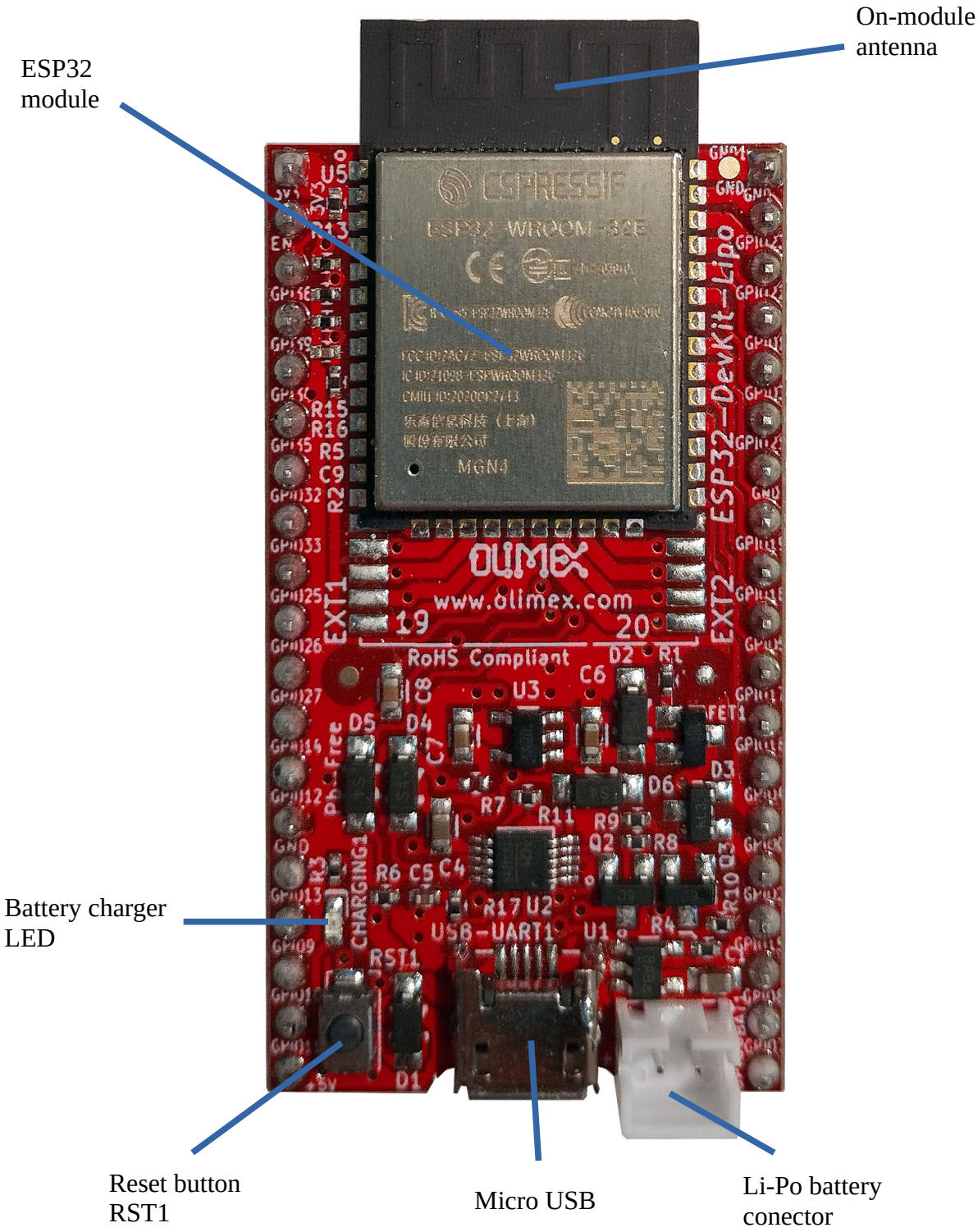
1.3 ESP32-DevKit-LiPo Open Source Licenses

ESP32-DevKit-LiPo is Open Source Hardware, listed in OSHWA.org here:

https://certification.oshwa.org/bg000019.html

The hardware files are released under CERN OSHW license.
The software is released under GPL 3 license.
The documentation is released under CC BY-SA 3.0 license.

2. General board layout



3. Power supply and consumption

ESP32-DevKit-LiPo typically consumes around 50mA of current depending on the software. The board can consume much less using the power saving modes – down to around 10uA in deep sleep mode.

The absolute maximum power ESP32-DevKit-LiPo can draw from the power supply would be the determined by the maximum input of the regulator on the power input line. Of course, consider that on-board peripherals and the main module would use some of that current. The board uses low power linear voltage regulator, it was chosen since it consumes only 1.5uA in power down mode. The downside is that it doesn't provide a lot of current, up to a maximum of 500mA. Leave 100-150mA for the ESP32 module, you are left with 350-400mA for additional circuits.

4. Schematics and dimensions

ESP32-DevKit-LiPo was designed with KiCAD (free and open-source CAD tool). ESP32-DevKit-LiPo schematics and sources can be found at GitHub here:

https://github.com/OLIMEX/ESP32-DevKit-LiPo/tree/main/HARDWARE

There are also PDF exports if you don't want to install KiCAD.

5. ESP32-DevKit-LiPo pinout description

All pin names and functions are printed in white print at the bottom of the board. The board's pinout can also be seen in the schematic's bottom left corner.

The ESP32 chip has very good multiplexer so you can set the free GPIO pins for alternative functions via software means.

The board has battery power sense and battery power measurement optional feature, that can be enabled by modifying the state of jumpers BAT_PWR_E1, PWR_SENS_E1, and BAT_SENS_E1. Close them (solder the pads together) to enable both functions, on pins GPI35 and GPI39.

6. Software installation

Espressif guide for Arduino IDE installation – after installation – there is own entry for the board, it should be listed as OLIMEX ESP32-DevKit-LiPo in the board selection.

Olimex provides an Arduino example for deep sleep testing here:

https://www.olimex.com/Products/IoT/ESP32/ESP32-DevKit-LiPo/resources/timerweake-up-sketch.zip

Espressif ESP-IDF installation.

Espressif guide for PlatformIO installation.

7. Step-by-step Arduino IDE installation

The instructions were made under Windows 10 but should be pretty similar for different operating systems.

7.1. Download and install Arduino IDE

If you still don't have Arduino IDE, navigate to the following web-address to download it:

https://www.arduino.cc/en/software/

Download the version suitable for your operating system and install it (or extract it if you use the stand-alone version). After the installation is complete, launch the Arduino IDE application.

7.2. Install the ESP32 support for Arduino IDE

7.2.1. Install Espressif Arduino-ESP32 stable release. To do so follow the detailed instructions here:

https://docs.espressif.com/projects/arduino-esp32/en/latest/installing.html

Short summary of what needs to be done:

- In Arduino IDE navigate to "File" → "Preferences" and in the filed that says "Additional Boards Manager URLs" append the following json link (copy-paste):

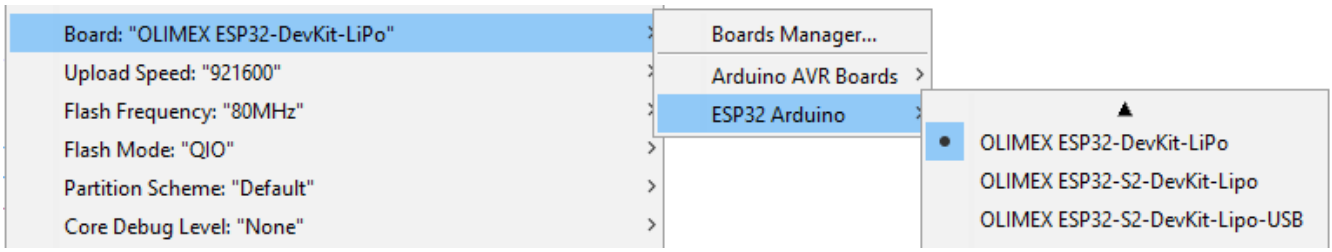
https://espressif.github.io/arduino-esp32/package_esp32_index.json

- Navigate to "Tools" → "Boards" → "Board Manager" and search for the *esp32* platform. Install the latest version (3.0.4 at the time of writing this document).

If you encounter problems during the package installation make sure to check all details on the official guide from the link on the previous page.

7.2.2. Restart Arduino IDE and select the configuration for the board. Navigate to "Tools" → "Board" →under "ESP32 Arduino" tab. Find in the list "OLIMEX ESP32-DevKit-LiPo" and select it.

You can also change the properties (ESP32-WROOM and ESP32-WROVER require slightly different settings, since ESP32-WROVER has extra 8MB PSRAM). If you have WROVER version, enable PSRAM.



7.3. Connect the board to the computer

First connect the ESP32-DevKit-LiPo board to your computer via the micro USB connector using USB cable. In the “Windows Device Manager” you should be seeing a new CH340 device. It should be listed under “Ports” and listed as “USB-SERIAL CH340”.

7.4. Upload a sketch

7.4.1. After you enter the bootloader mode as mentioned above, navigate to “Tools” → “Port…” select the COM of your device (make sure to identify it from “Windows Device Manager”):

7.4.2. Compile and upload your sketch.

8. Document revision history

Revision 1.0 December 2024