

blog.csdn.net /Mark_md/article/details/121326060


ESP32使用Arduino环境，安装、编译、下载、安装库

一、安装Arduino环境

先下载Arduino：[Arduinp IDE下载](#)

我选择的是免安装压缩包，下载后直接解压即可运行。

Downloads



Arduino IDE 1.8.16

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

~~Windows~~ Win 7 and newer

Windows ZIP file

Windows app Win 8.1 or 10 [Get](#)

Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

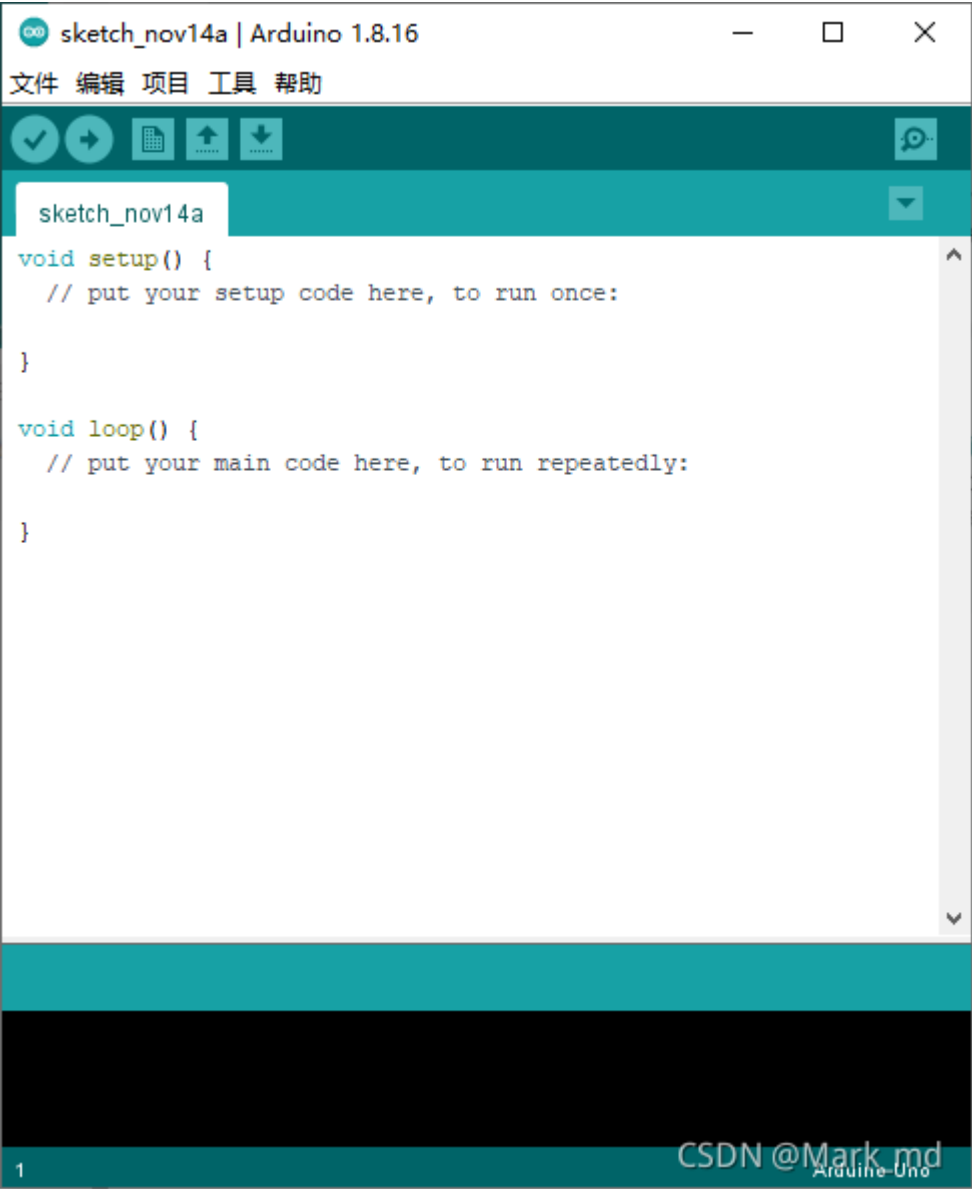
Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#) [Checksums \(sha512\)](#)

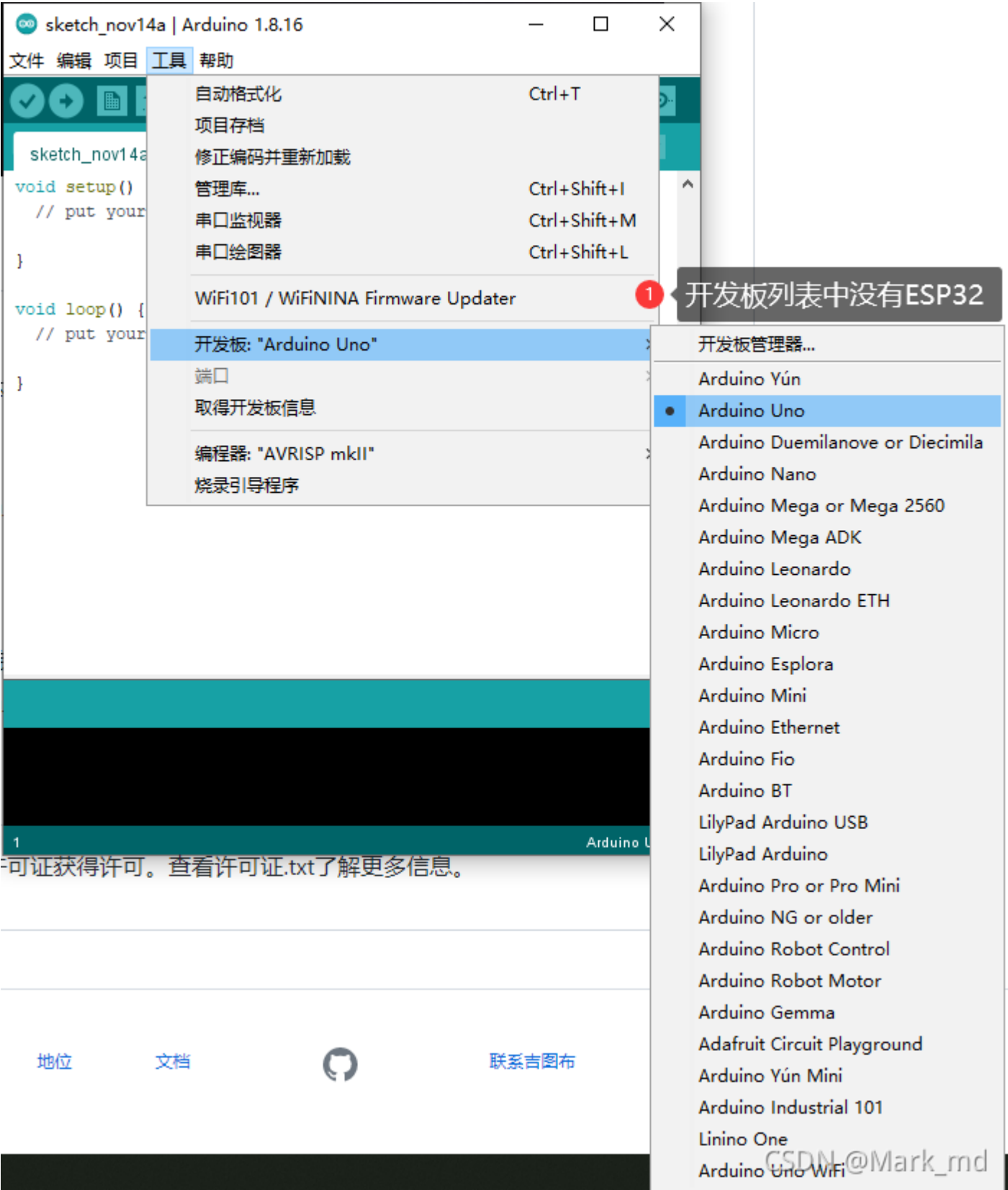
CSDN @Mark_md

界面。

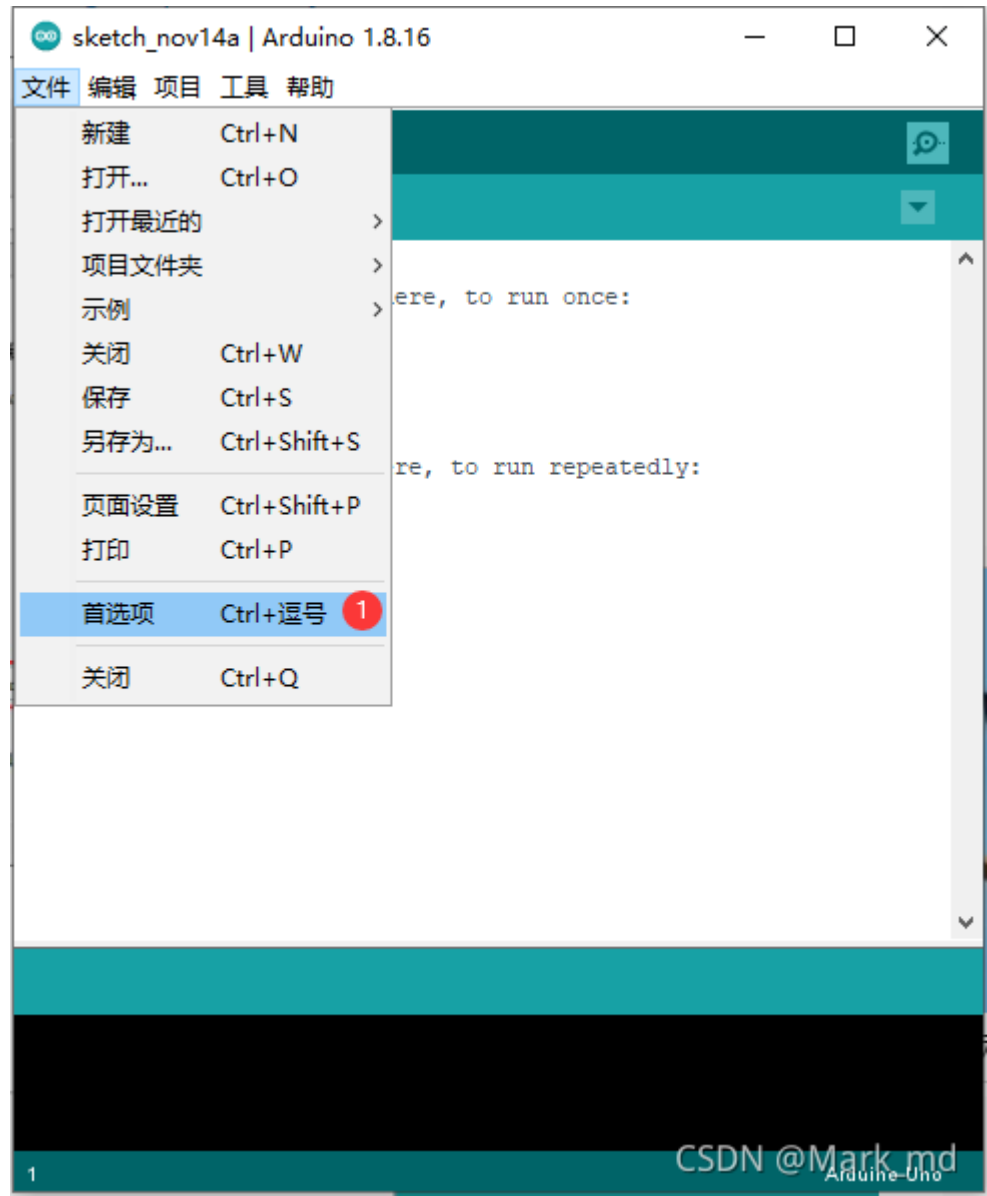


二、添加ESP32开发板

而此时的开发板列表中，并没有ESP32设备。
需要额外安装支持包，以支持ESP32的开发。



- 文件 - 首选项。

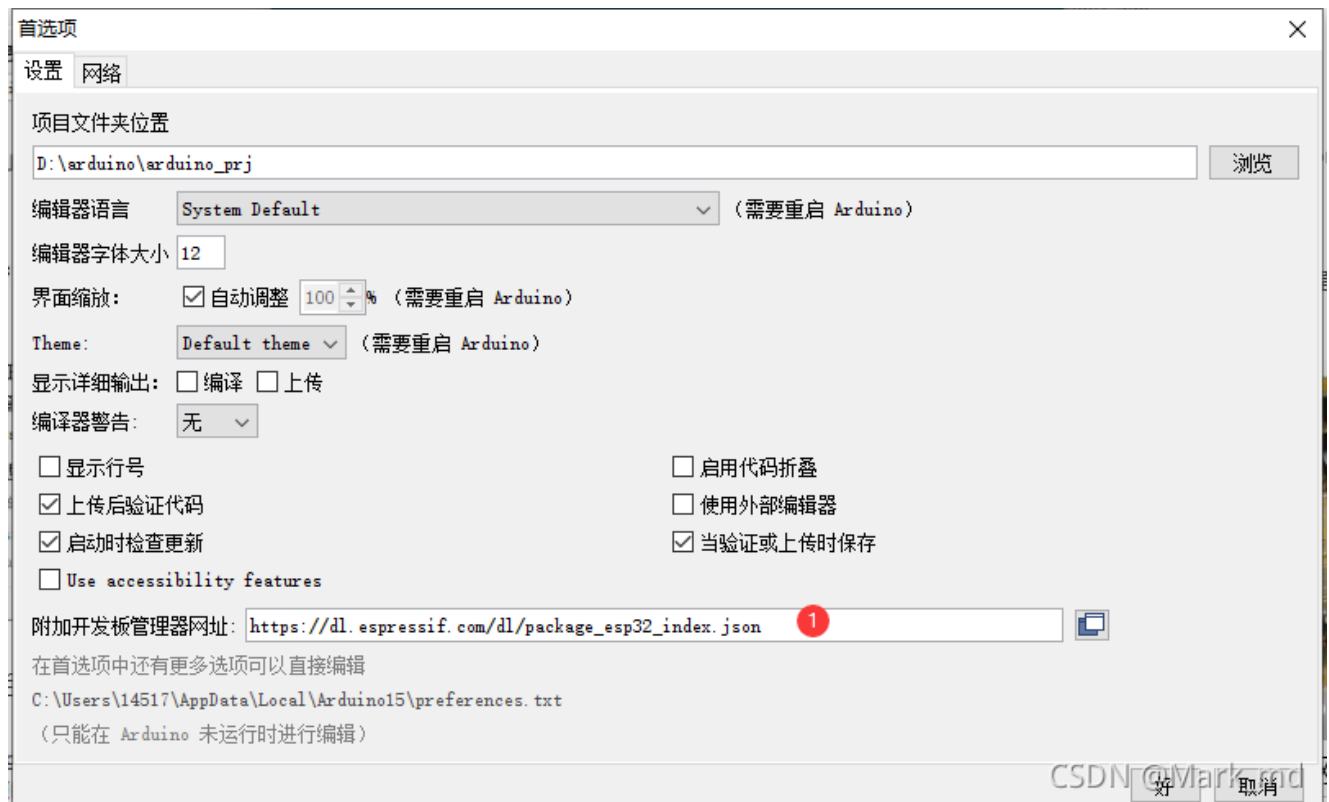


- 在附加开发板管理网站中，输入

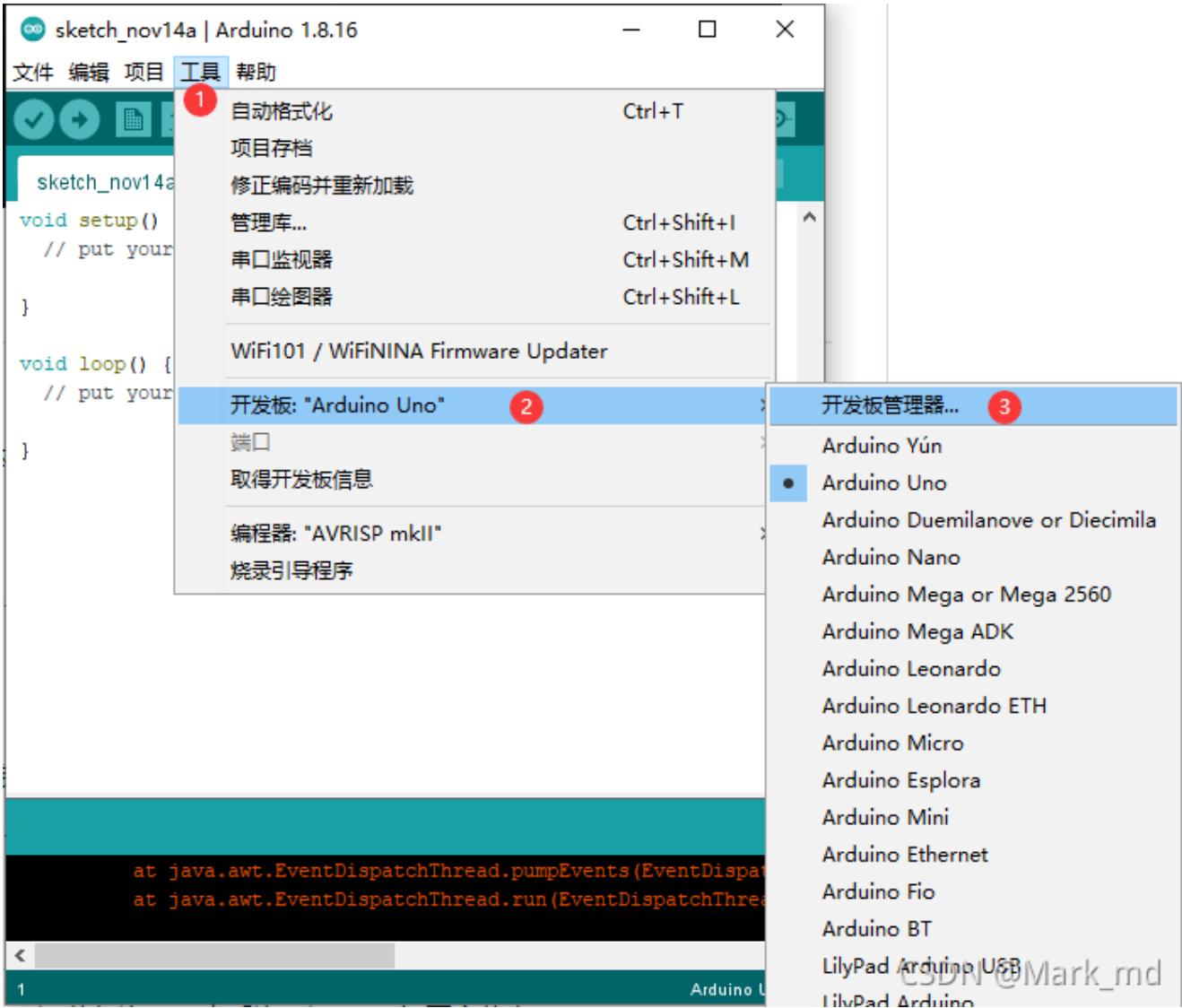
https://dl.espressif.com/dl/package_esp32_index.json

- 1
- 顺手把上面的项目文件夹位置给改了，这个路径就是以后库的下载路径。

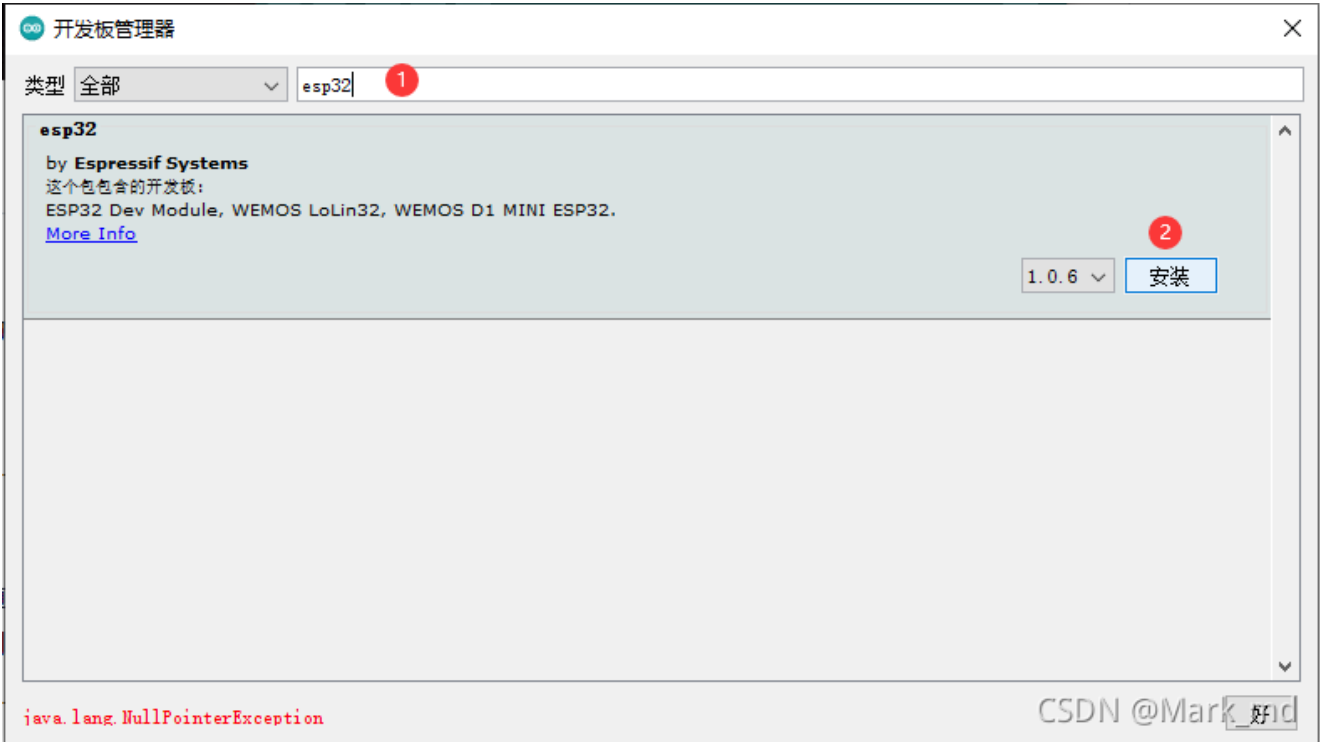
- 点击好，退出。



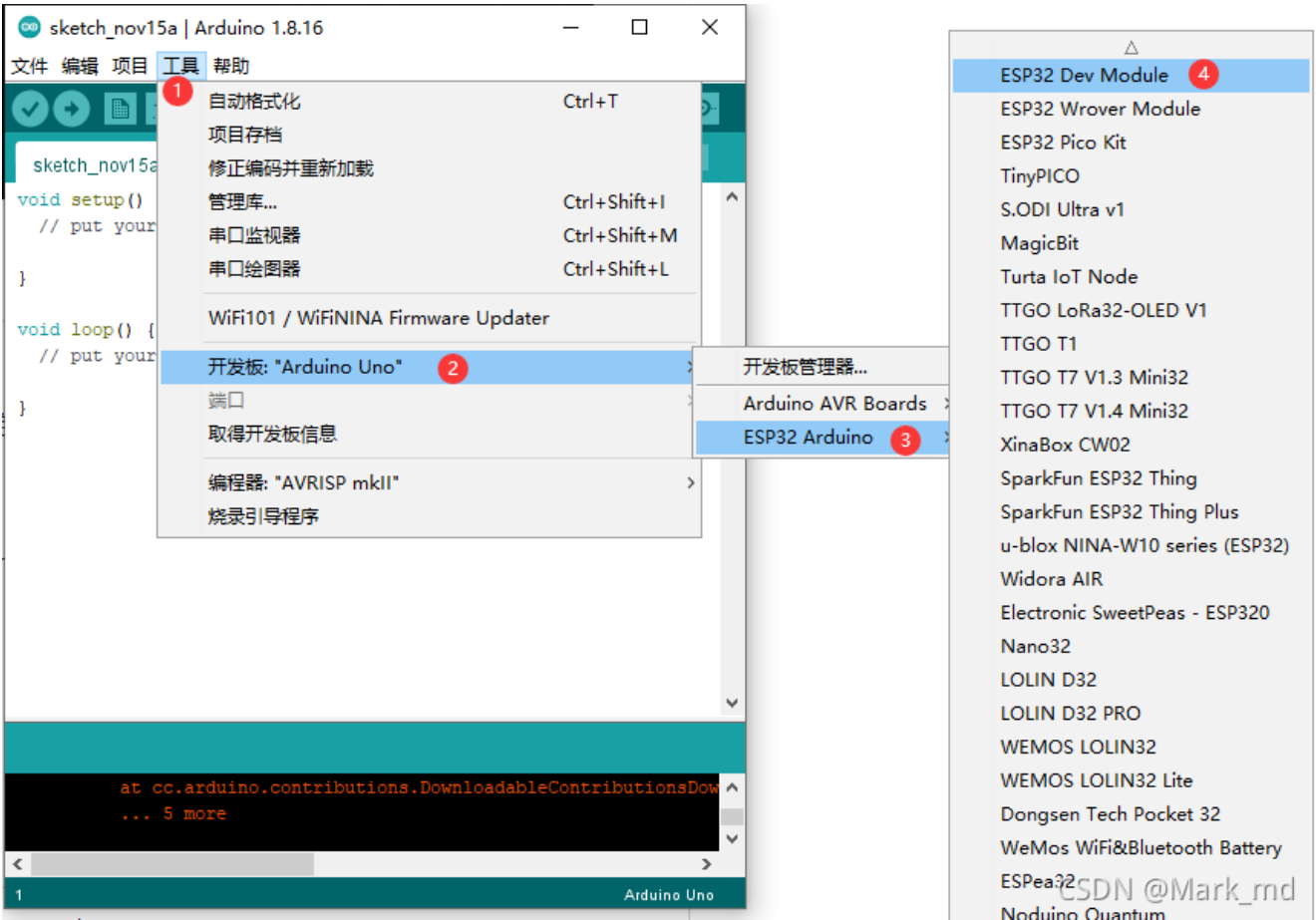
- 接着，工具 - 开发板 - 开发板管理。



- 弹出后，搜索ESP32，点击安装。
- 考验耐心的时候到了，下载非常慢，还总容易断。如果电脑是笔记本的话，可以用手机的热点。（不知道现在网络什么情况，不科学上网，访问Github，手机的基站网络要比家用宽带快得多）



- 安装好后，按照下图，选择开发板。



三、为什么选用ESP32呢？

先来说下市面上常用的MCU开发板。

Arduino自家UNO开发板，其MCU型号为Atmega328p。频率仅有20MHz，2+32KB显然与时代脱节。最重要的是用了这么多年快停产了，价格每年都在涨，钱包日渐消瘦表示只好另寻佳人。

stm32系列因为近两年价格猛涨，现已大部分被国产取代。但国产替代最大的问题在于手册不配套、SDK不维护，不是量产客户，很难得到FAE的技术支持。所以如果仅仅是diy爱好者，国产替代的使用体验肯定不如ST原厂的好。虽然过两年的价格可能会回落，但囤货的供应商显然不会让价格跌的那么快。现在的价格让人高攀不起，Cortex-A的价格买Cortex-M的芯片，而且还极有可能买到二手翻新，H750的价格没怎么变但没有小型易焊接的封装。对不起，再也不见。

树莓派Pico的RP2040，普通而又自信的M0+，性能没多少亮点但结合价格来看中规中矩，外设不多但该有的也都有。最尴尬的是生态，目前支持c++和MicroPython两种开发方式，Arduino未来会有支持。虽然不想说，但就个人以前的mpy使用体验来讲，mpy真是个无比尴尬的存在。乍一听很美好，低代码开发，不用在电脑安装编译工具链，也不用等待漫长的编译。不过泼冷水的来了，MicroPython虽然语法上与Python相似，但实际用起来完全就是两个东西。用过mpy的应该多少都痛恨其调试过程，解释器的运行机制，导致其比c慢的多得多；代码只有在MCU上运行后，才能知道之前写的对不对；且不得不人为添加许多异常处理，以免触发异常报错后导致程序卡死；还有扩展库的用法与标准Python的并不完全一致，用的库越多越感觉mpy与py的相似性越低。种种原因，都让MicroPython的体验没预想的那么好。

乐鑫ESP32系列在疫情期的价格稳定，虽因台积电涨价，价格略涨，但相比较同行，简直业界良心。热门型号有比较新的ESP32C3，单核160MHz，WIFI+BLE5.0，虽然RAM略有缩水，仅400 KB SRAM，但得益于SDK优化，最终可供用户使用的内存与ESP32相差无几。QFN封装，有内置Flash的型号，除了GPIO较少其他方面都是吊打同行。另外ESP32的SDK可以说是在MCU界无敌强的存在，与其他国产的画风完全不同。开发方式多种多样，可以借助强大的ESP-IDF，也可以用Arduino、Platform、Lua、MicroPython。现阶段推荐型号有ESP32、ESP32C3。S2被阉割太多了搞的没法用，S3要等等但价格或许要比以前贵很多。

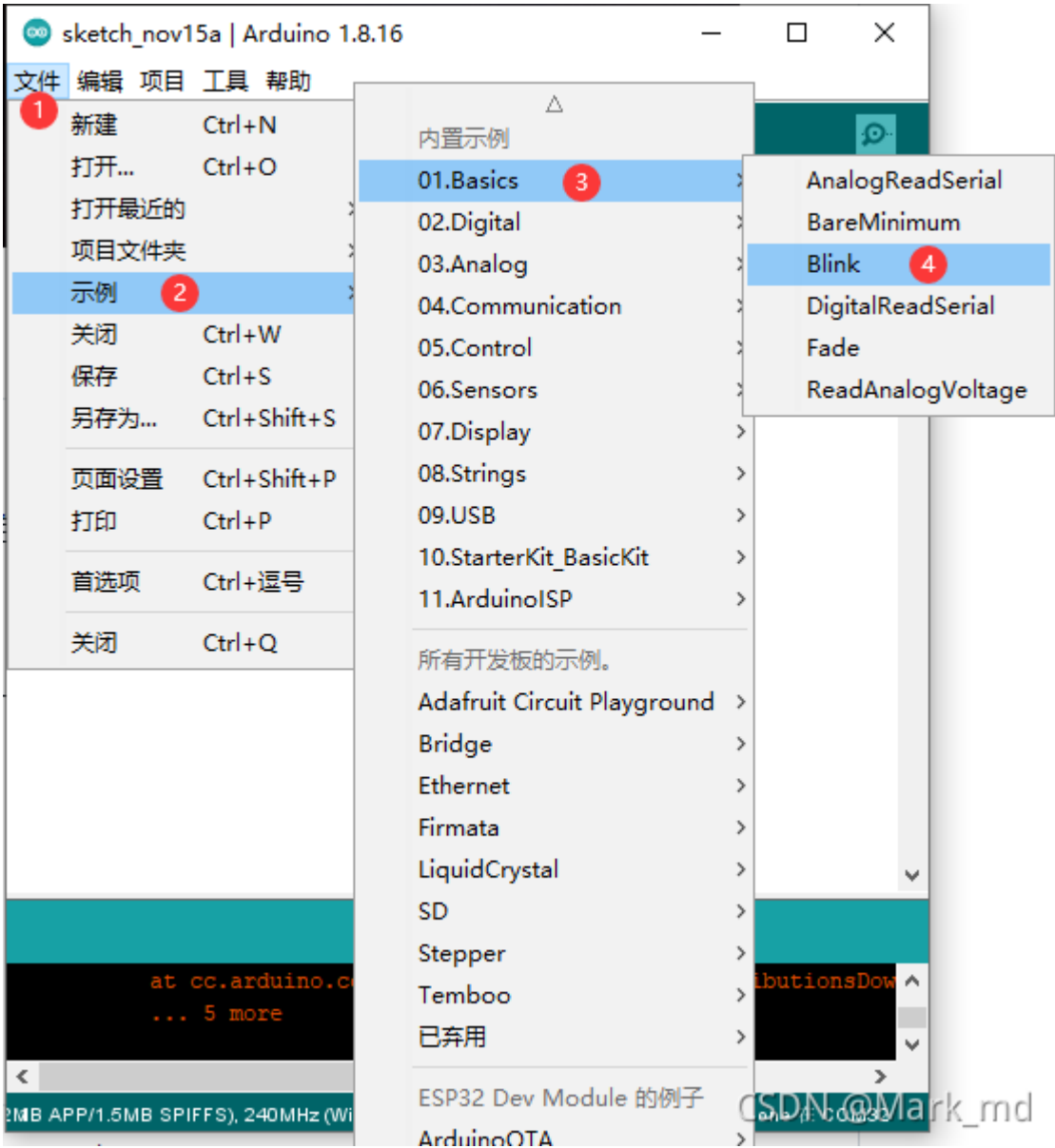
本文介绍Arduino。而ESP-IDF开发可以参照我的专栏：[ESP32开源驱动库Easyio的使用，基于ESP-IDF开发框架](#)

本教程使用ESP32-WROOM-E模组，4MB Flash。（其他的如WROVER也可以）

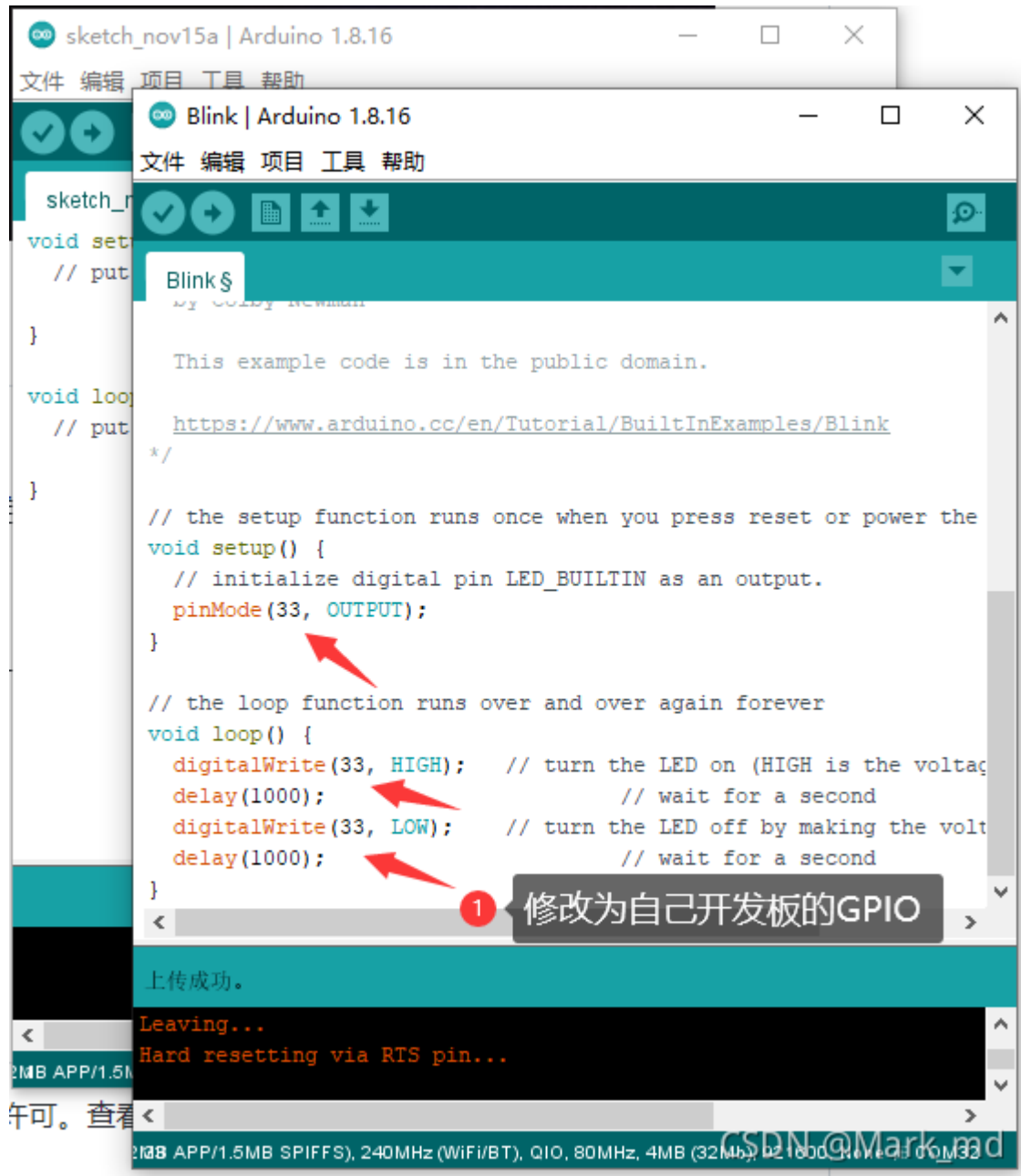
四、编译下载运行示例工程-Blink

尝试建立一个简单的工程，并编译下载运行。

- 以比较简单的Blink为例。

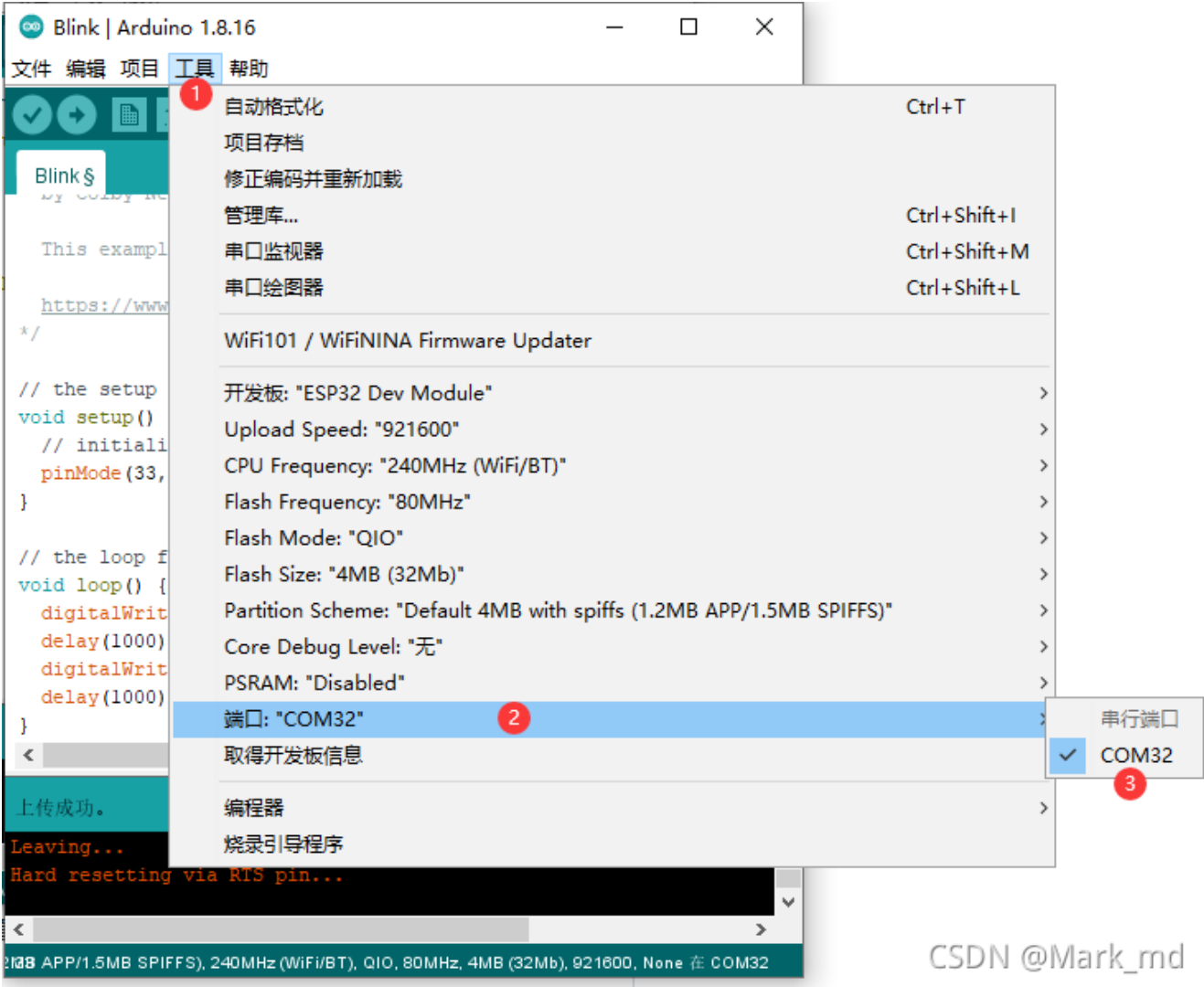


- 针对自己的开发板，对GPIO做一些修改。



- 随后将开发板插入，选择端口，连接开发板。
- 在上方的配置项为 下载波特率、ESP32运行频率、Flash频率、Flash模式、Flash容量、分区表 等等。我们在购置开发板或者模组时，一般都会选择4MB或者以上的版本，这时候可以不用修改，用

默认的即可。



CSDN @Mark_md

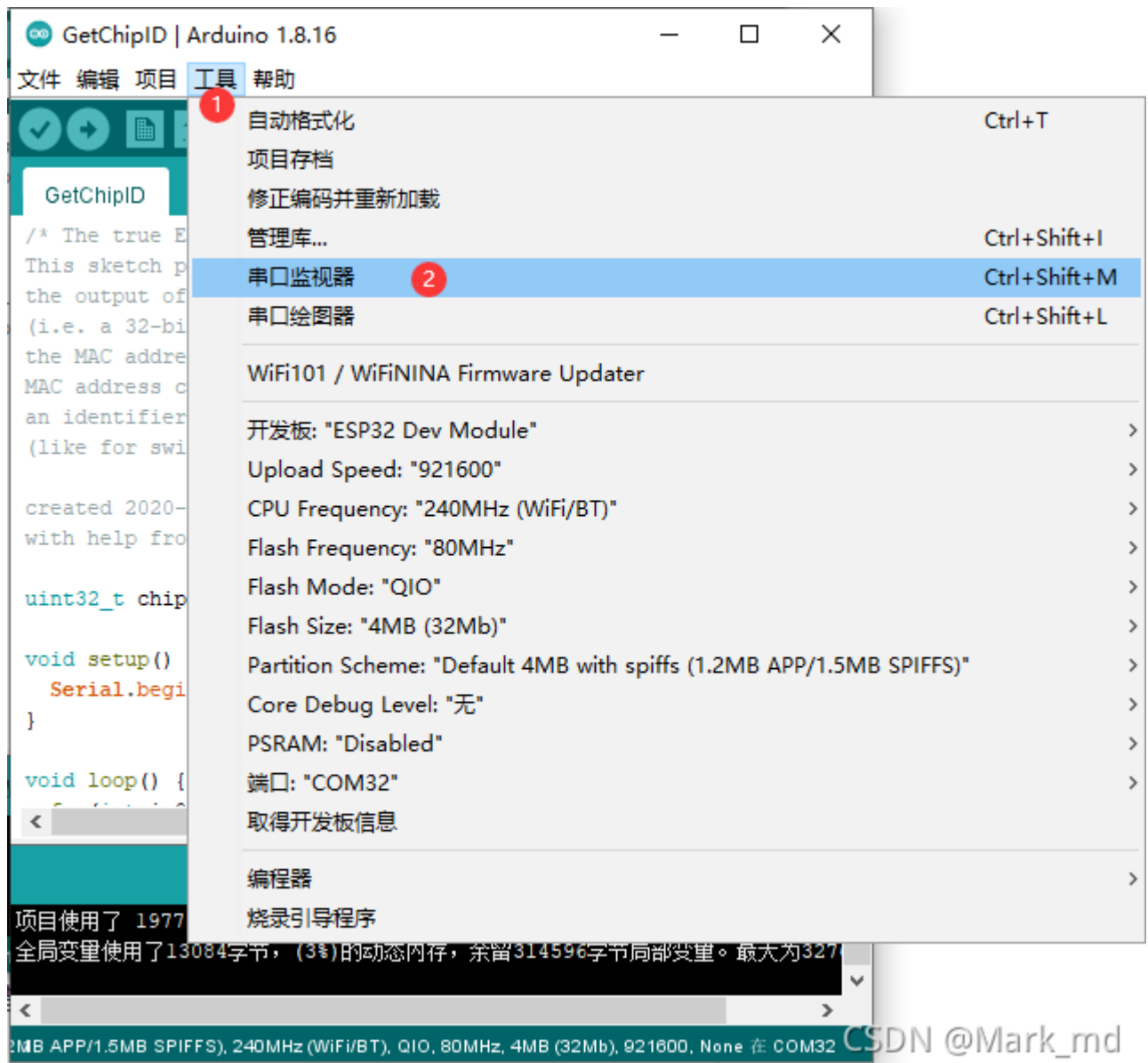
- 最后点击上传。就能看到板载的LED在闪烁。



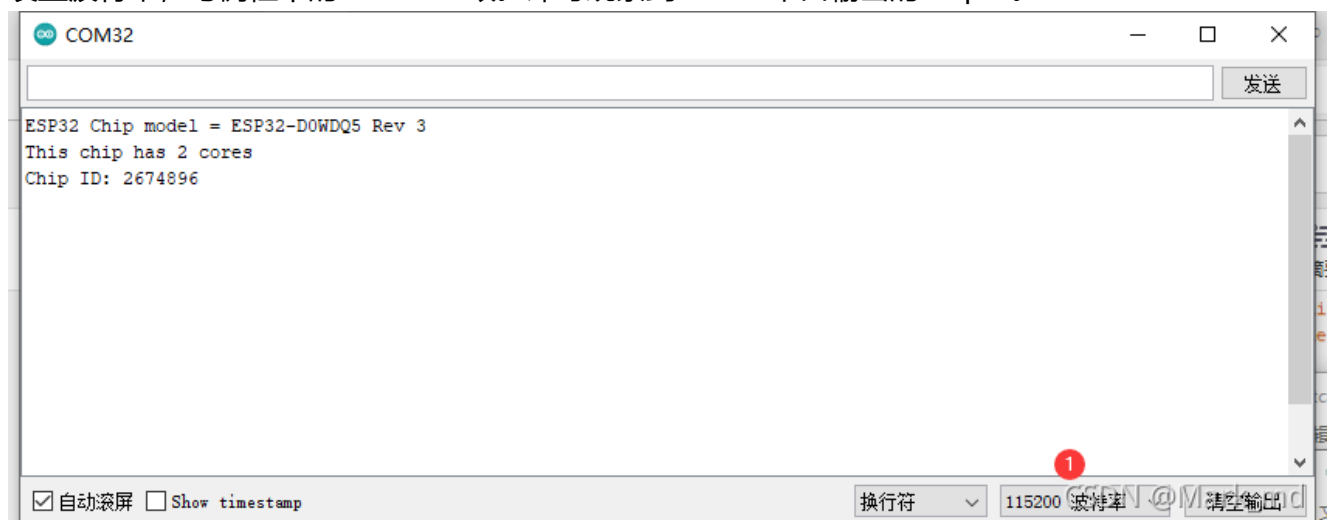
五、串口监视器使用

- 重建一个工程。文件 - 示例 - ESP32 - ChipID - GetChipID。
- 上传。

- 打开 工具 - 串口监视器。来监视串口信息。



- 设置波特率，与例程中的115200一致。即可观察到ESP32串口输出的ChipID。



ESP32开发板开源计划

现已开源一款ESP32物联网开发板，支持WIFI、蓝牙、以太网、RS485、CAN通信。板载6轴IMU、温湿度传感器、38K红外遥控收/发、独立RTC、TF卡座。USB和18650双电源供电，有2.0寸单点电容屏可用，配套各外设驱动和LVGL源码。Github、立创EDA、CSDN全云端同步更新。

详情可见：

- **ESP32-IOT-KIT 开发板介绍：**

[ESP32开发板开源啦 ESP32-IOT-KIT全开源物联网开发板](#)

- **PCB硬件：**

[立创EDA：ESP32开发板 IOT-KIT原理图、PCB、BOM](#)

打样免费的时代，速去JLC白嫖PCB。

开源学习交流Q群：827686418



群名称:ESP32全开源Easyio物联学...

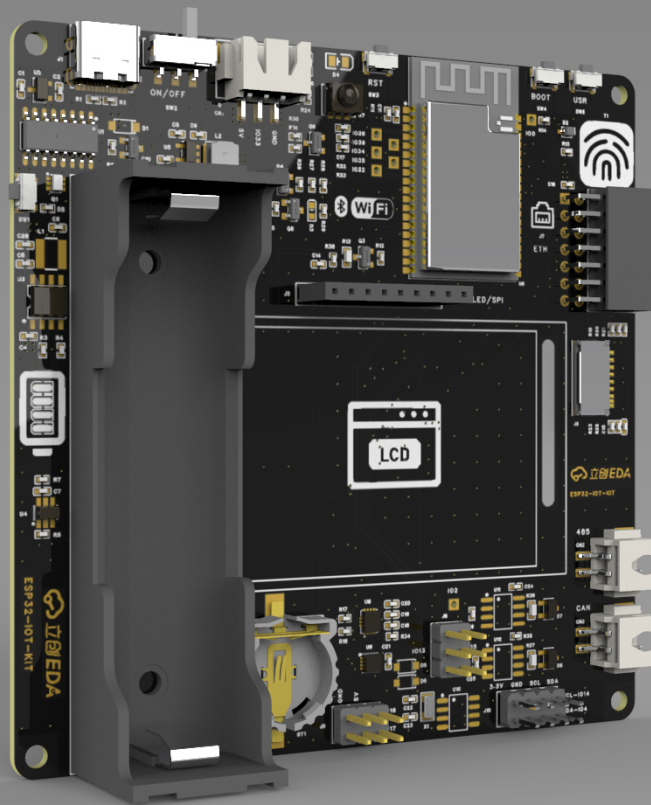
群 号:827686418

ESP32 IOT KIT

ESP32-WROVER-E

16M Flash + 8M PSRAM

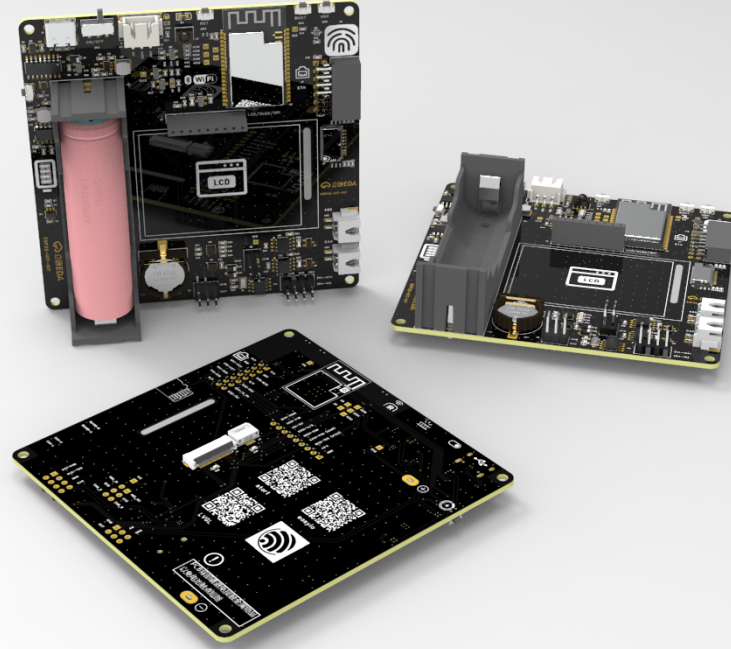
I/O全引出



- 蓝牙、WIFI
- 有线以太网（插接）
- RS485、CAN
- TF卡座（MicroSD）
- 原配2.0寸LCD
- 电容触摸屏
- 板载18650电池
- 方便脱机调试
- 集成充放电管理
- TypeC接口
- CH340下载、调试
- 集成 温湿度传感器
- 六轴IMU、RTC
- 红外收发、光感



CSDN @Mark_md



CSDN @Mark_md