

ESP32C3

通常你不需要这份文档，这是用于自行扩展固件的高级文档。

我们提供的固件包就包含编译好的固件（soc后缀）

如果你只是想将已有的库添加到固件中，可以使用我们提供的[在线云编译](#)生成自定义固件。

如果你是在找刷机/编译lua脚本之类的应用型文档，那么这份文档不是你需要查看的内容。

视频教程链接: [B站每日喝粥](#)

准备环境

安装好乐鑫官方的idf环境，目前为 [ESP-IDF V4.4.1](#)，具体请见项目readme

Windows平台用户

装好7-zip，建议保持默认文件夹！！

安装IDF。对于国内用户，推荐直接安装离线包：[esp-idf-tools-setup-offline-4.4.1.exe](#)

详细信息请见乐鑫官方文档：[Windows 平台工具链的标准设置](#)

Linux平台用户

装好7-zip，各个平台装法不一样（apt/yum/pacman），自己搜吧

接着请直接参考乐鑫官方文档的步骤进行安装：[Linux 和 macOS 平台工具链的标准设置](#)

准备项目

将LuatoS主仓库clone到适当位置,推荐用git, 如果下载zip一定一定要解压后改文件夹名称!!!

如果不熟悉git, 强烈建立[入门](#)一下

```
git clone https://gitee.com/openLuat/LuatOS.git
```

再将ESP32C3项目clone到与LuatoS主仓库的同级目录

```
git clone https://gitee.com/dreamcmi/LuatOS-ESP32.git
```

必须按以下目录结构进行摆放, 以 [D:\gitee](#) 为例

```
LuatoS 主库代码 D:\gitee\LuatoS
ESP32C3 适配代码 D:\gitee\LuatoS-ESP32
```

检查点，路径正确的情况下，以下文件路径必存在，找不到就肯定是命名问题，手动添加是徒劳的

```
D:\gitee\LuatoS\lua\src\lgc.c
D:\gitee\LuatoS-ESP32\components\luat\include\luat_conf_esp.h
```

任何附加字符都不可用, 例如 [LuatoS-master](#) 就是错误的命名

定制固件里的库

打开 `D:\gitee\LuatOS-ESP32\components\luat\include\luat_conf_esp.h`，按需注释或取消注释。注意，如果功能太大导致固件放不下，会编译失败。

打开idf命令行工具

该步骤仅限 windows 用户需要做

配置项目目标芯片

在 IDF 命令行，进入 `LuatOS-ESP32` 仓库目录，执行命令 `idf.py set-target esp32c3`

如果需要开启USB打印日志（新版开发板）

经典版开发板用户无需按此步骤操作，默认就是串口输出日志

在 IDF 命令行执行 `idf.py menuconfig` 命令

进入 `Component config` -> `ESP System Settings` -> `Channel for console output`

勾选 `USB Serial/JTAG Controller`，按S保存，回车，再按Q退出

```
(Top) → Component config → ESP System Settings → Channel for console output
Espressif IoT Development Framework Configuration
( ) Default: UART0
(X) USB Serial/JTAG Controller
( ) Custom UART
( ) None

[Space/Enter] Toggle/enter  [ESC] Leave menu          [S] Save
[O] Load                   [?] Symbol info          [/] Jump to symbol
[F] Toggle show-help mode  [C] Toggle show-name mode [A] Toggle show-all mode
[Q] Quit (prompts for save) [D] Save minimal config (advanced)
```

编译

在 IDF 命令行，进入 `LuatOS-ESP32` 仓库目录，执行 `idf.py build`，开始编译

```
_httpd.c.obj
[1021/1868] Linking C static library esp-idf\protobuf-c\libprotobuf-c.a
[1022/1868] Building C object esp-idf/protocomm/CMakeFiles/__idf_protocomm.dir/src/simple_ble/simple_ble.c.obj
[1023/1868] Building C object esp-idf/protocomm/CMakeFiles/__idf_protocomm.dir/src/transport/protocomm_ble.c.obj
[1024/1868] Building C object esp-idf/esp_local_ctrl/CMakeFiles/__idf_esp_local_ctrl.dir/src/esp_local_ctrl_handler.c.obj
[1025/1868] Building C object esp-idf/esp_local_ctrl/CMakeFiles/__idf_esp_local_ctrl.dir/src/esp_local_ctrl_transport_ble.c.obj
[1026/1868] Building C object esp-idf/esp_local_ctrl/CMakeFiles/__idf_esp_local_ctrl.dir/proto-c/esp_local_ctrl.pb-c.c.obj
[1027/1868] Building C object esp-idf/esp_local_ctrl/CMakeFiles/__idf_esp_local_ctrl.dir/src/esp_local_ctrl.c.obj
[1028/1868] Building C object esp-idf/mdns/CMakeFiles/__idf_mdns.dir/mdns_networking_lwip.c.obj
[1029/1868] Linking C static library esp-idf\protocomm\libprotocomm.a
[1030/1868] Building C object esp-idf/expat/CMakeFiles/__idf_expat.dir/expat/expat/lib/xmlrole.c.obj
[1031/1868] Building C object esp-idf/mdns/CMakeFiles/__idf_mdns.dir/mdns_console.c.obj
[1032/1868] Building C object esp-idf/expat/CMakeFiles/__idf_expat.dir/expat/expat/lib/xmltok_ns.c.obj
[1033/1868] Building C object esp-idf/expat/CMakeFiles/__idf_expat.dir/expat/expat/lib/xmltok_impl.c.obj
[1034/1868] Building CXX object esp-idf/wear_levelling/CMakeFiles/__idf_wear_levelling.dir/crc32.cpp.obj
[1035/1868] Building CXX object esp-idf/wear_levelling/CMakeFiles/__idf_wear_levelling.dir/WL_Ext_Perf.cpp.obj
```

当出现 `Project build complete.` 字样则表示编译成功

生成SOC烧录文件

进入 `LuatOS-ESP32/tools` 目录，执行下面的命令

```
pip install -r requirements.txt
python esp32v3.py -t esp32c3 -p
```

即可在 `LuatOS-ESP32/tools` 目录找到生成完毕的SOC文件