

# CM4-ET-IND BALENA OS 用户手册



Version: 0.1  
2025-04-18

<a href="http://www.emtop-tech.com">www.emtop-tech.com</a>	<a href="https://github.com/EMTOP-TECH">github.com/EMTOP-TECH</a>
<a href="mailto:sales@emtop-tech.com">sales@emtop-tech.com</a>	<a href="mailto:support@emtop-tech.com">support@emtop-tech.com</a>

修订历史

版本	日期	说明
0.1	2025-04-18	Initial Release

目录

第一章、 Balena OS 介绍 ..... 2

    1.1 BalenaOS 架构 ..... 2

    1.2 Balena CLI ..... 3

第二章、 开始操作：CM4-ET-IND ..... 4

    2.1 准备工作[PC Windows 下操作] ..... 4

    2.2 部署应用程序 ..... 8

    2.3 自动更新到设备 ..... 12

    2.4 启动项配置 ..... 14

    2.5 SSH 登录 ARM 板 ..... 16

附录 ..... 17

    词汇表 ..... 17

## 第一章、 Balena OS 介绍

BalenaOS 是一种针对在嵌入式设备上运行 Docker 容器而优化的操作系统，注重长期运行的可靠性，以及受构建 balena 时的经验教训启发的高效开发人员工作流程。

BalenaOS 的核心理念是，Linux 容器首次提供了在嵌入式设备上使用虚拟化的实用途径。虚拟机和虚拟机管理程序已显著提升了云部署的生产力和自动化程度，但它们对硬件的抽象、资源开销以及硬件支持的缺乏，意味着它们并不适合嵌入式场景。而 Linux 容器所实现的操作系统级虚拟化，则消除了上述两个障碍，使其适用于物联网中众多 Linux 设备。

BalenaOS 是一个旨在轻松移植到多种设备类型（通过 Yocto 框架）的操作系统，并且针对 Linux 容器（尤其是 Docker）进行了优化。

BalenaOS 作为 balena 平台的一部分开发的，并且已经在 balena 上的数千台嵌入式设备上运行，并在许多不同的环境中部署多年，验证了其能够作为独立的操作系统运行，适用于嵌入式设备和容器交互的任何环境。

### 1.1 BalenaOS 架构

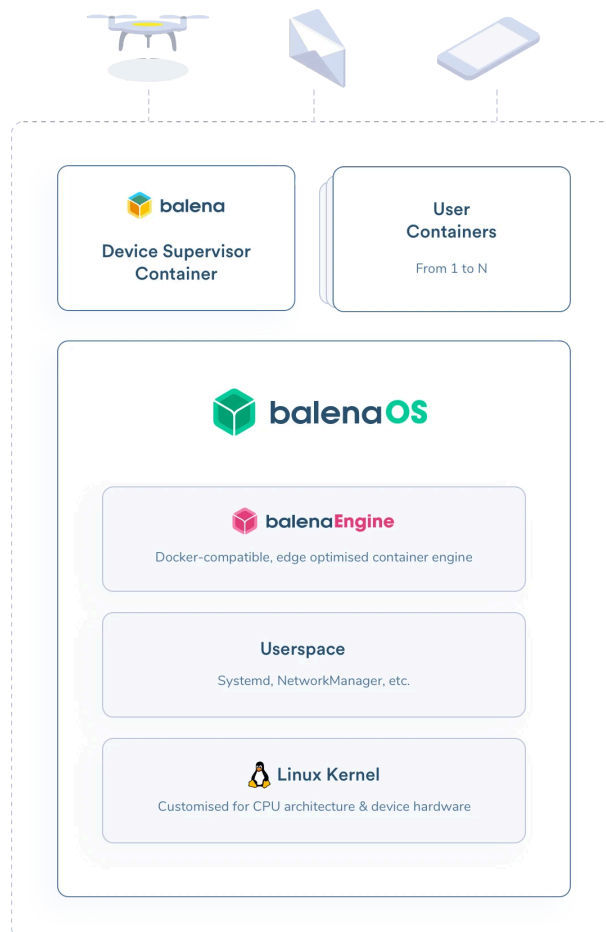


图 1-1 BalenaOS 核心架构

Balena 生态系统中的设备运行 balenaOS，这是一款基于 Yocto Linux 的精简主机操作系统，它与我们轻量级、兼容 Docker 的容器引擎 balenaEngine 一起打包提供。主机操作系统负责启动设备监控器、设备上的 balena 代理以及容器化服务。在每个服务的容器中，您可以指定一个基本操作

系统，该操作系统可以来自任何与您的设备架构兼容的现有 Docker 基本映像。基本操作系统与主机操作系统共享内核，但在其他方面独立工作。如果您愿意，可以将容器配置为以特权运行、直接访问硬件，甚至将模块注入内核。balena 设备监控器在其自己的容器中运行，即使您的应用程序崩溃，设备也可以继续运行并提取新代码。图 1-1 为 BalenaOS 在 ARM 板中运行的系统架构。

## 1.2 Balena CLI

Balena CLI 是 balenaCloud 或 openBalena 的命令行界面。它是一款适用于 Windows、macOS 和 Linux 的软件工具，可通过命令提示符/终端窗口使用。它可以交互使用，也可以在脚本中调用。Balena CLI 基于 balena API 和 balena SDK 构建，也可以直接导入到 Node.js 应用程序中。它的应用场景十分广泛，可用于项目部署、设备状态监控等。

Balena CLI 的主要应用场景：

- IoT 设备管理
- 远程部署和更新
- 多设备管理
- 容器应用程序开发

Balena CLI 是 Github 上的开源项目：<https://github.com/balena-io/balena-cli>。

<a href="http://www.emtop-tech.com">www.emtop-tech.com</a>	<a href="https://github.com/EMTOP-TECH">github.com/EMTOP-TECH</a>
<a href="mailto:sales@emtop-tech.com">sales@emtop-tech.com</a>	<a href="mailto:support@emtop-tech.com">support@emtop-tech.com</a>

## 第二章、 开始操作：CM4-ET-IND

在本章节中，我们将通过以下方式帮助您使用 balenaCloud：

- 设置您的 CM4-ET-IND 设备并使其在 balenaCloud 仪表板上联机
- 在设备上部署 Node.js hello-world 项目
- 开发示例项目：进行更改并在设备上实时测试它们

完成该 Balena 入门指南后，您将掌握使用 BalenaCloud 继续开发应用程序所需的基础知识，并踏上将设备集群部署到生产环境的征程。

### 2.1 准备工作[PC Windows 下操作]

- 访问 <https://www.balena.io>，并注册 balenaCloud 账号
- 登录后创建组织（Organization），可自由命名, 本文中以“emtop-tech”为例
- 创建 fleet，可自由命名，文中以“example-app”为例

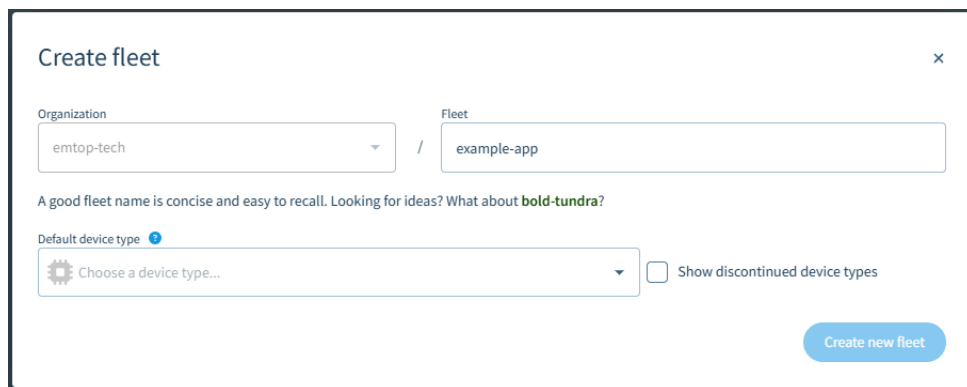


图 2-2 创建 fleet

- 选择设备类型 从 Default device type 下拉菜单中选择设备类型，我们选择“Raspberry Pi 4 (using 64bit OS)”

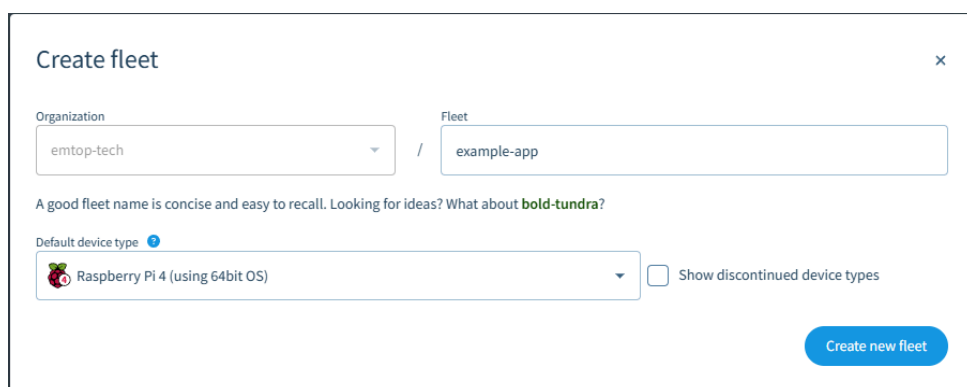


图 2-3 选择设备类型

- 点击“Create new fleet”按钮，创建完成。

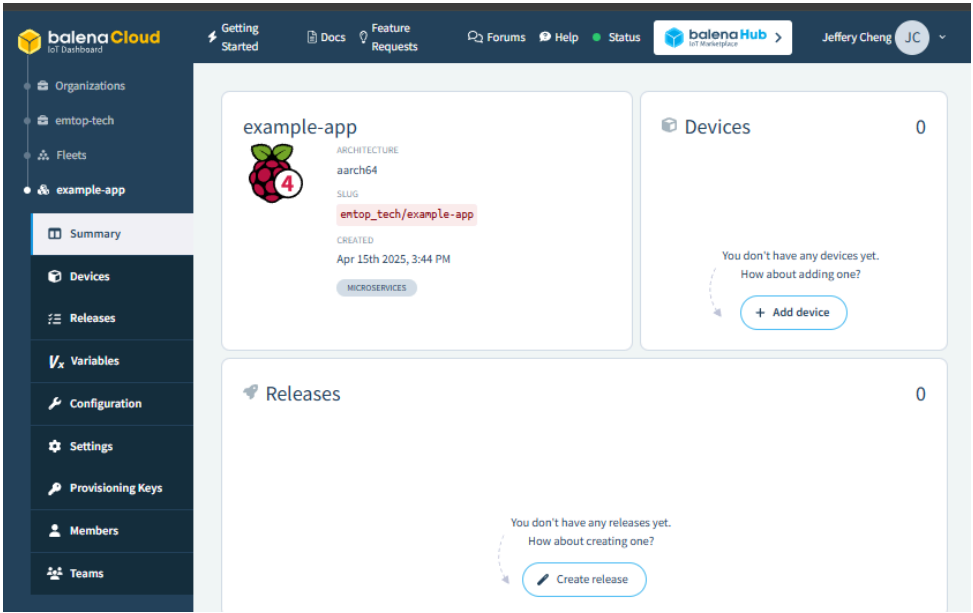


图 2-4 Fleet 创建完成

- 点击“Add device”按钮

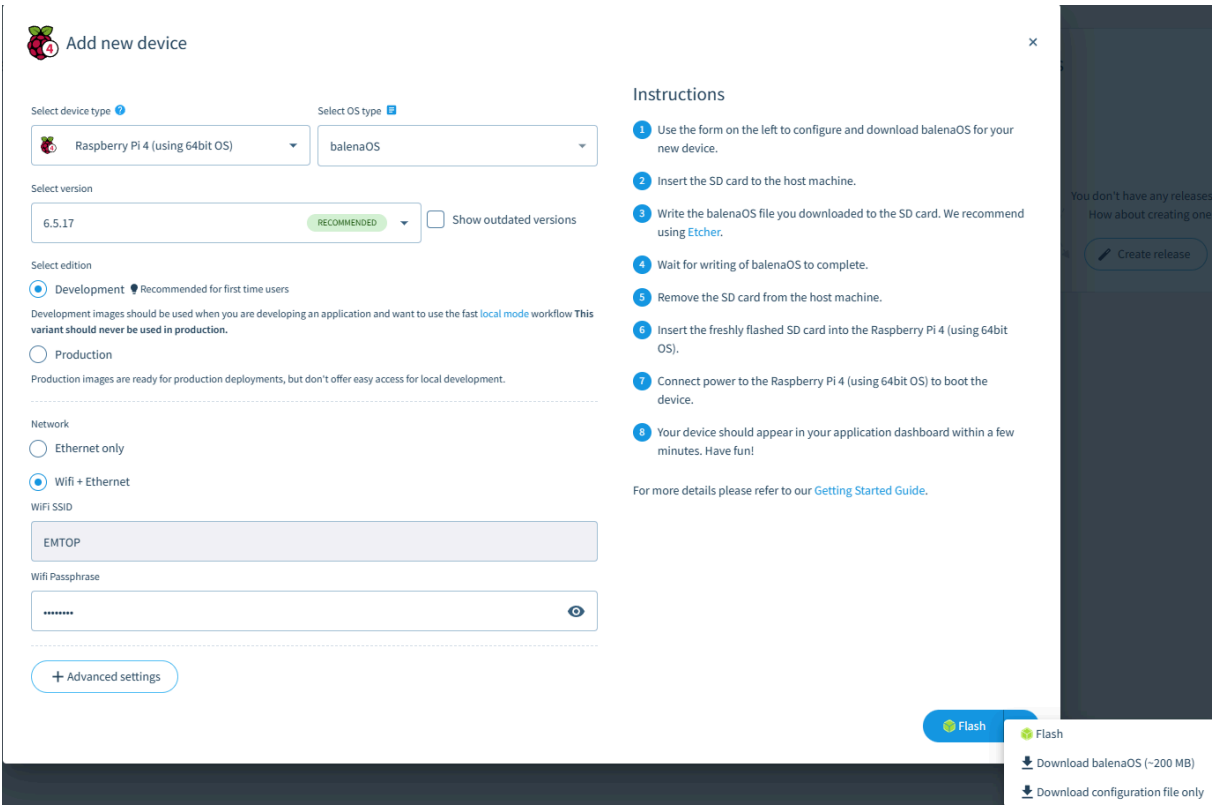


图 2-5 添加设备

[Select edition]

Development：支持调试串口和 ssh 登录，针对新手用户，推荐选择该配置

Production：适用于产品部署，不支持本地登录

点击右下角”Flash“按钮上的三角箭头，可弹出三个下载选项：

Flash	调用 Etcher 烧录工具直接烧录 SD 卡
Download balenaOS	下载 balenaOS 映像
Download configuration file only	仅下载配置文件

我们这里选择“Download balenaOS”，因为获得映像后便于重复烧录制作启动卡。

下载完成后得到 *balena-cloud-example-app-raspberrypi4-64-6.5.17-v17.0.2.img.zip*，该文件名随着官方版本的升级而变化，不过操作方法类似。

- 下载并安装 balenaEtcher：<https://etcher.balena.io>
- 运行 balenaEtcher 烧录 SD 卡

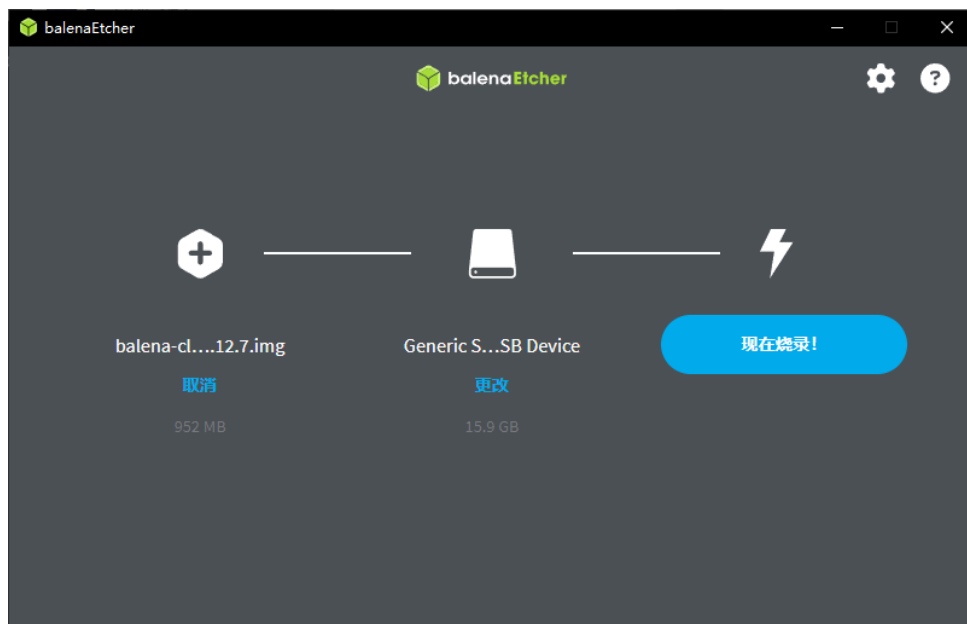


图 2-6 烧录 SD 卡

#### 备选烧录方式

也可将.zip 文件解压后生成.img 文件，运行 Win32diskimager 工具烧录 SD 卡。

- 修改板级支持
  1. 编译 CM4-ET-IND 内核源码，获得 emtop-cm4-et-ind.dtbo 和 emtop-vc4-kms-dsi-7inch.dtbo
  2. 将以上 dtbo 文件复制到 SD 卡的 resin-boot 分区 overlays 目录下
  3. 修改 resin-boot 分区下的 config.txt 文件

```
# Enable UART
enable_uart=1
# Enable VC4 Graphics
dtoverlay=vc4-kms-v3d,cma-320

dtparam=i2c_arm=on
dtparam=spi=on
disable_splash=1
```



```
avoid_warnings=1
dtparam=audio=on
```

```
dtoverlay=emtop-cm4-et-ind
dtoverlay=emtop-vc4-kms-dsi-7inch
```

**注**

或者可直接烧录我们提供的映像 **CM4-ET-IND-balena-cloud-example-app-raspberrypi4-64-6.5.17-v17.0.2.img**

- 将其烧录到 SD 卡
- 在“Add device”页面 图 2-5 下选择“Download configuration file only”
- 将下载的.json 文件覆盖 SD 卡 resin-boot 分区下的 config.json 文件

如果在 BalenaCloud 页面上删除了设备,想要重新加上的话,无需再下载一份新的 balena 映像,只需下载 config.json 文件覆盖 resin-boot 下的 config.json 即可。

- 将 ARM 板接入互联网,有线网或 WiFi 均可
- 将修改完毕的启动卡安装到 ARM 板中,连接上调试串口,上电启动

**注意**

调试串口不强制连接,新手练习阶段建议连上,主要为了检查网络是否正常联通。

- 观察串口终端,启动完成后将显示如下信息

```
balenaOS 6.5.17 f3a135c serial0
f3a135c login:
```

我们可以用 root 账号,空密码登录。

- 建议用 ping 命令检查 ARM 板是否连入互联网

```
root@f3a135c:~# ping www.baidu.com
```

```
64 bytes from 183.240.99.58: seq=0 ttl=53 time=13.294 ms
64 bytes from 183.240.99.58: seq=1 ttl=53 time=13.507 ms
64 bytes from 183.240.99.58: seq=2 ttl=53 time=13.439 ms
64 bytes from 183.240.99.58: seq=3 ttl=53 time=13.300 ms
...
```

- 等待约 1~2 分钟,刷新 balenaCloud 页面,可看到它发现了个新设备

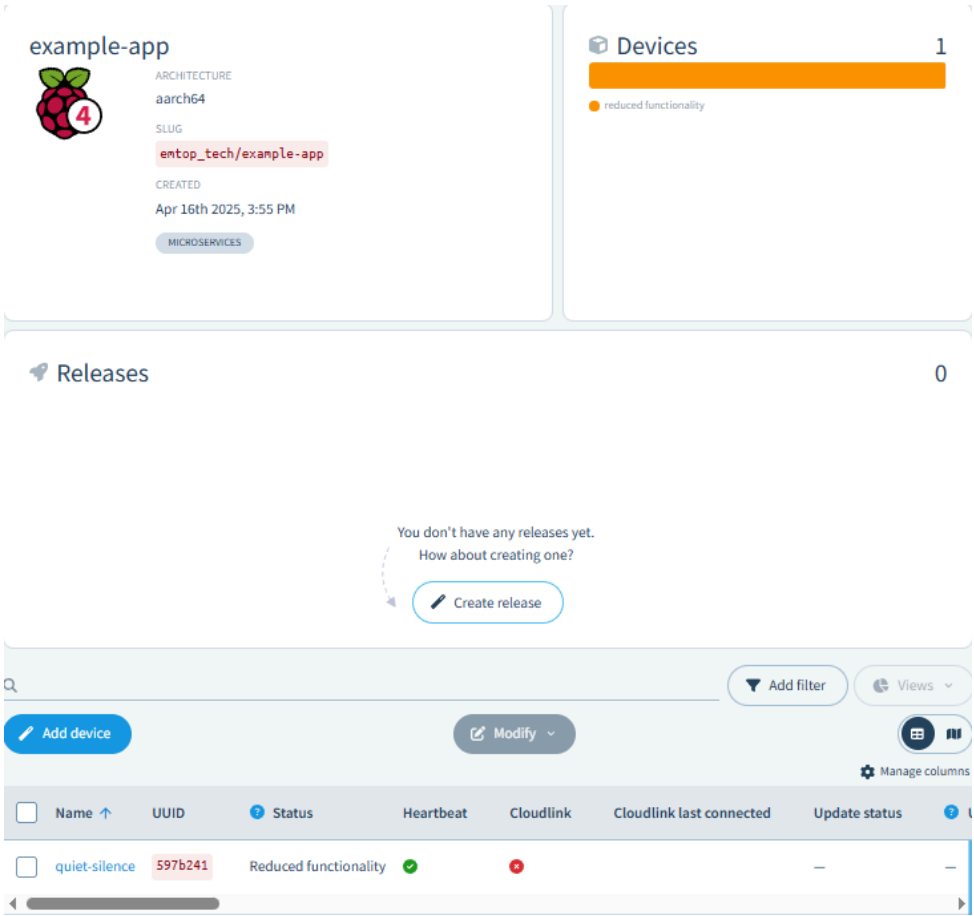


图 2-7 发现新设备

2.2 部署应用程序

我们可从官网中找到一些应用模板：<https://docs.balena.io/learn/more/examples/example-projects>。

# Balena example projects

IMPROVE THIS DOC

Below is a selection of "Hello world" examples and feature demonstrations for a wide variety of device types and languages when using balena.

Name	Description
<a href="#">balena-nodejs-hello-world</a>	Example of how to deploy Node code on a balena supported device.
<a href="#">balena-plant-saver</a>	We're building a plant monitor (and saver) - this is the early stage
<a href="#">multicontainer-getting-started</a>	Get up and running quickly with a multicontainer setup on balena
<a href="#">internetspeedtest</a>	Log your internet download, upload and ping metrics to influxDB, and Grafana it. Grafana all the things!
<a href="#">x11-window-manager</a>	Example project showing how to run a desktop manager with balenaCloud in order to run GUI applications.
<a href="#">balena-python-hello-world</a>	Example of how to deploy Python code on a balena supported device.
<a href="#">staged-releases</a>	demonstration scripts on how to use the resin.io API to do staged releases to a fleet of devices.
<a href="#">coral-streaming-object-detector</a>	This example will help you deploy a streaming camera feed with realtime people detection using the Coral Edge TPU for on-device ML inferencing.

图 2-8 示例项目


我们以 balena-python-hello-world 为例讲解部署步骤（以下操作在 PC Ubuntu24.04 系统下完成）：

- 安装 Balena CLI
  1. 打开网页：<https://github.com/balena-io/balena-cli>
  2. 进入 releases 页面，下载最新版本的 linux-x64-standalone.zip
  3. 将其解压后复制到任意目录，比如/home/emtop/balena-cli
  4. 创建软链接

```
emtop@Ubuntu24:~$ sudo ln -sf /home/emtop/balena-cli/balena /usr/sbin/balena
```

5. 登录

```
emtop@Ubuntu24:~$ balena login
```



```
Logging in to balena-cloud.com
? How would you like to login? (Use arrow keys)
? Web authorization (recommended)
  Credentials
  Authentication token
  I don't have a balena account!
```

选择（Web authorization），回车键确认后打印 https 网址：

```
Opening web browser for URL:
https://dashboard.balena-cloud.com/login/cli/http%253A%252F%252F127.0.0.1%253A37811%252Fauth
```

将其复制到 Ubuntu 浏览器打开，填写您的 balenaCloud 账号和密码，登录成功后显示如下界面

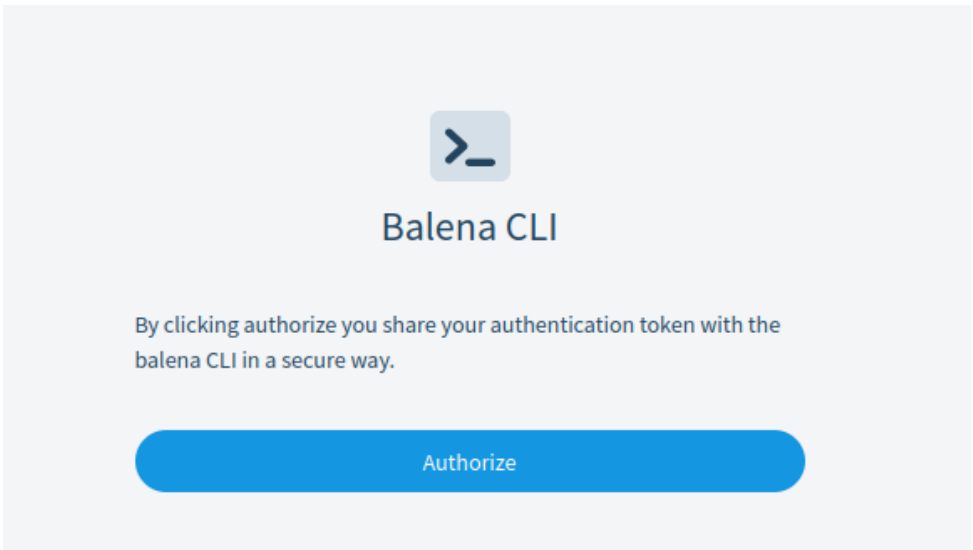


图 2-9 Balena 登录授权

点击“Authorize”按钮，成功后显示

```
Successfully logged in as: ██████████
```

- 查询 balenaCloud 上的 Fleets

```
emtop@Ubuntu24:~$ balena fleet list
```

ID	NAME	SLUG	DEVICE
TYPE	DEVICE COUNT	ONLINE DEVICES	
2238495	example-app	emtop_tech/example-app	
raspberrypi4-64	1	0	

记住目标 Fleet 的 NAME，本例中的是 example-app，下面部署命令中需要用到它。

- 下载 balena-python-hello-world 代码

```
emtop@Ubuntu24:~$ git clone https://github.com/balena-io-examples/balena-python-hello-world.git
```

- 上传部署

```
emtop@Ubuntu24:~$ balena push example-app
```

```
[Info] Starting build for example-app, user ██████
[Info] Dashboard link: https://dashboard.balena-cloud.com/apps/2238495/devices
[Info] Building on 4b2d7d0
[Info] No suitable previous release for caching, skipping cache pull
[balena-hello-world] Step 1/6 : FROM balenalib/aarch64-python:3-run
[balena-hello-world] --> 9b5accc91e93
[balena-hello-world] Step 2/6 : WORKDIR /usr/src/app
[balena-hello-world] Using cache
[balena-hello-world] --> 22c8baecec70
[balena-hello-world] Step 3/6 : COPY requirements.txt requirements.txt
[balena-hello-world] Using cache
[balena-hello-world] --> c65a18db75b5
[balena-hello-world] Step 4/6 : RUN pip install -r requirements.txt
[balena-hello-world] Using cache
[balena-hello-world] --> 52c5306d4416
[balena-hello-world] Step 5/6 : COPY . ./
[balena-hello-world] Using cache
[balena-hello-world] --> d6304b4699a1
[balena-hello-world] Step 6/6 : CMD ["python","-u","src/app.py"]
[balena-hello-world] Using cache
[balena-hello-world] --> ca358b1de50c
[balena-hello-world] Successfully built ca358b1de50c
[Info] Uploading images
[Success] Successfully uploaded images
[Info] Built on 4b2d7d0
[Success] Release successfully created!
[Info] Release: a728d2b6fb77df41de56a5ae6495af60 (id: 3450223)
```

Service	Image Size	Build Time
balena-hello-world	310.69 MB	< 1 second

```
Build finished in 14 seconds
```

$$\begin{array}{r} \backslash \\ \backslash \backslash \\ \backslash \backslash \\ > \backslash / 7 \\ \_ \cdot \cdot (6' \backslash \\ (= \_ \cdot \_ / \backslash \\ ) \backslash | \\ / / | \\ / > / \end{array}$$



- 此时 BalenaCloud 设备管理面板上可看到该项目出现在 Releases 栏中

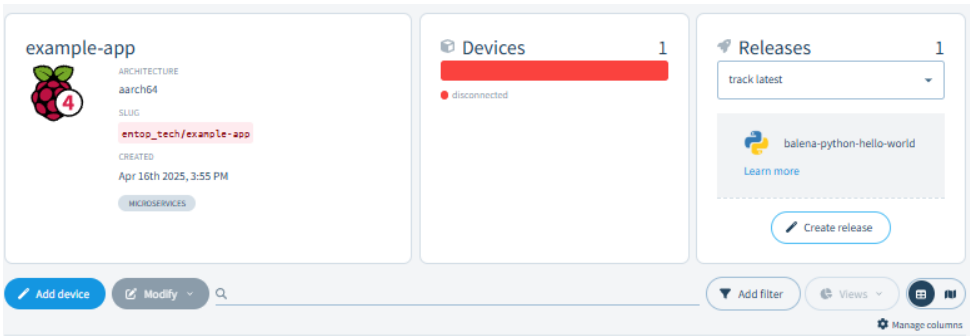


图 2-10 项目部署完成

2.3 自动更新到设备

BalenaCloud 上的应用部署完成后，所有能连上目标 Fleet 的设备都能自动同步安装该应用程序。ARM 板启动完成后，dashBoard 上将显示 Updating 进度：

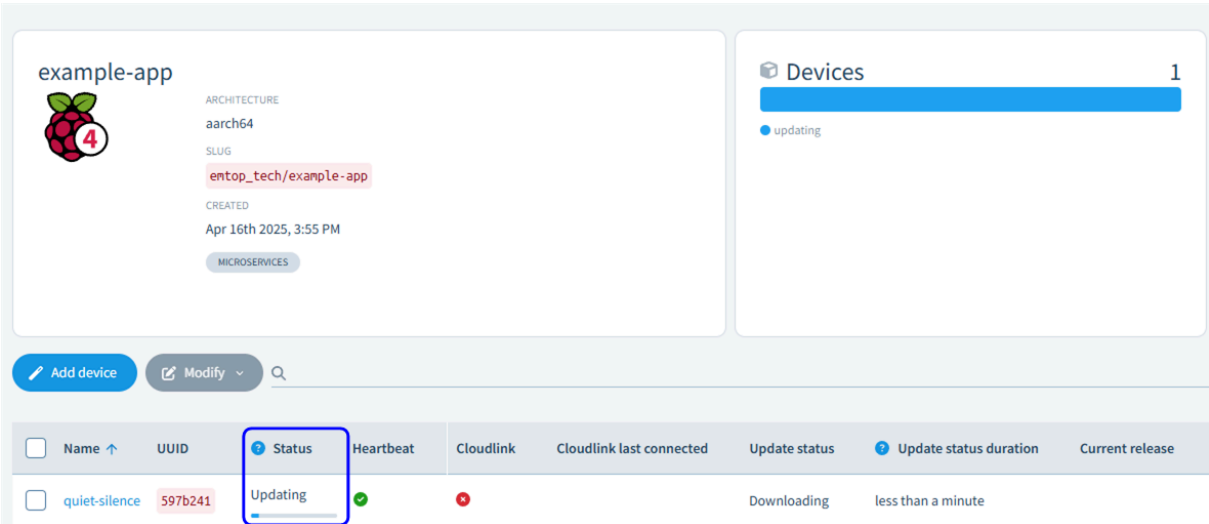


图 2-11 设备更新

等待更新完成后，点击设备列表中目标设备的 Name，本例中为“quiet-silence”，切换到设备管理页面，如图：图 2-12

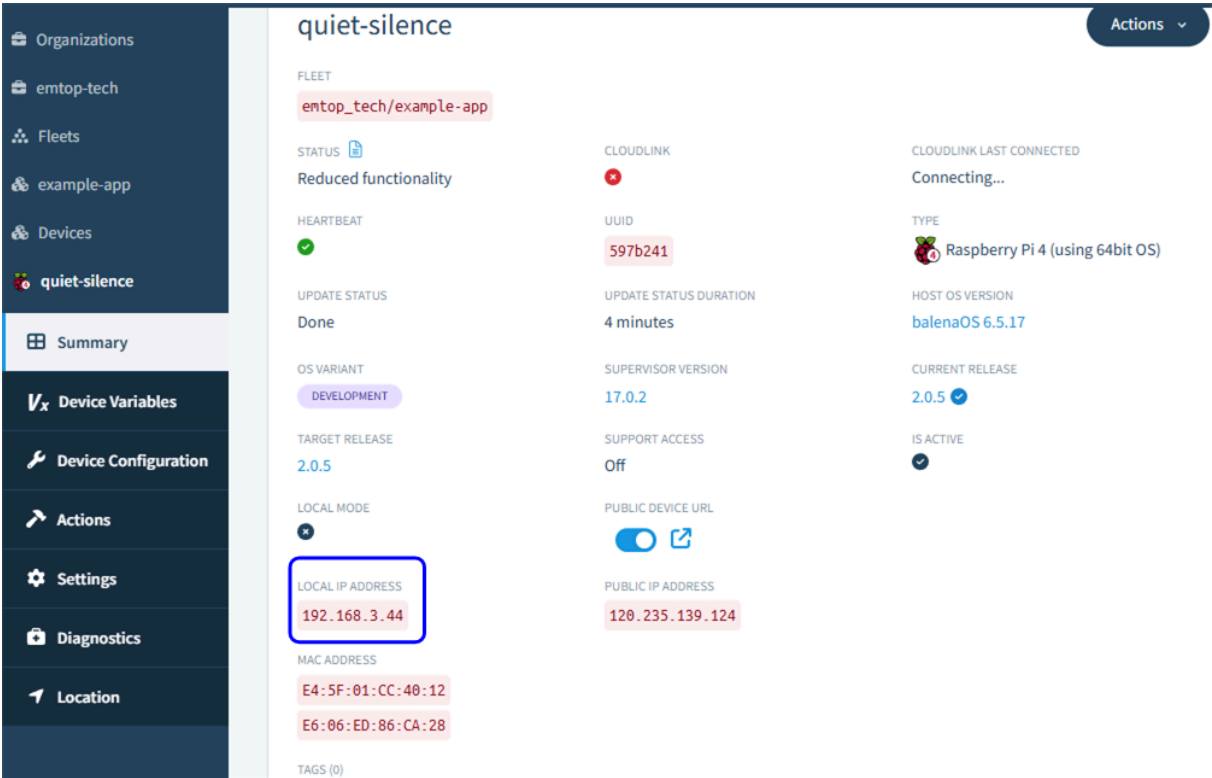


图 2-12 设备管理

复制设备的 LOCAL IP ADDRESS，粘贴到浏览器中打开，即可看到 balena-python-hello-world 程序显示的主页，如图：图 2-13

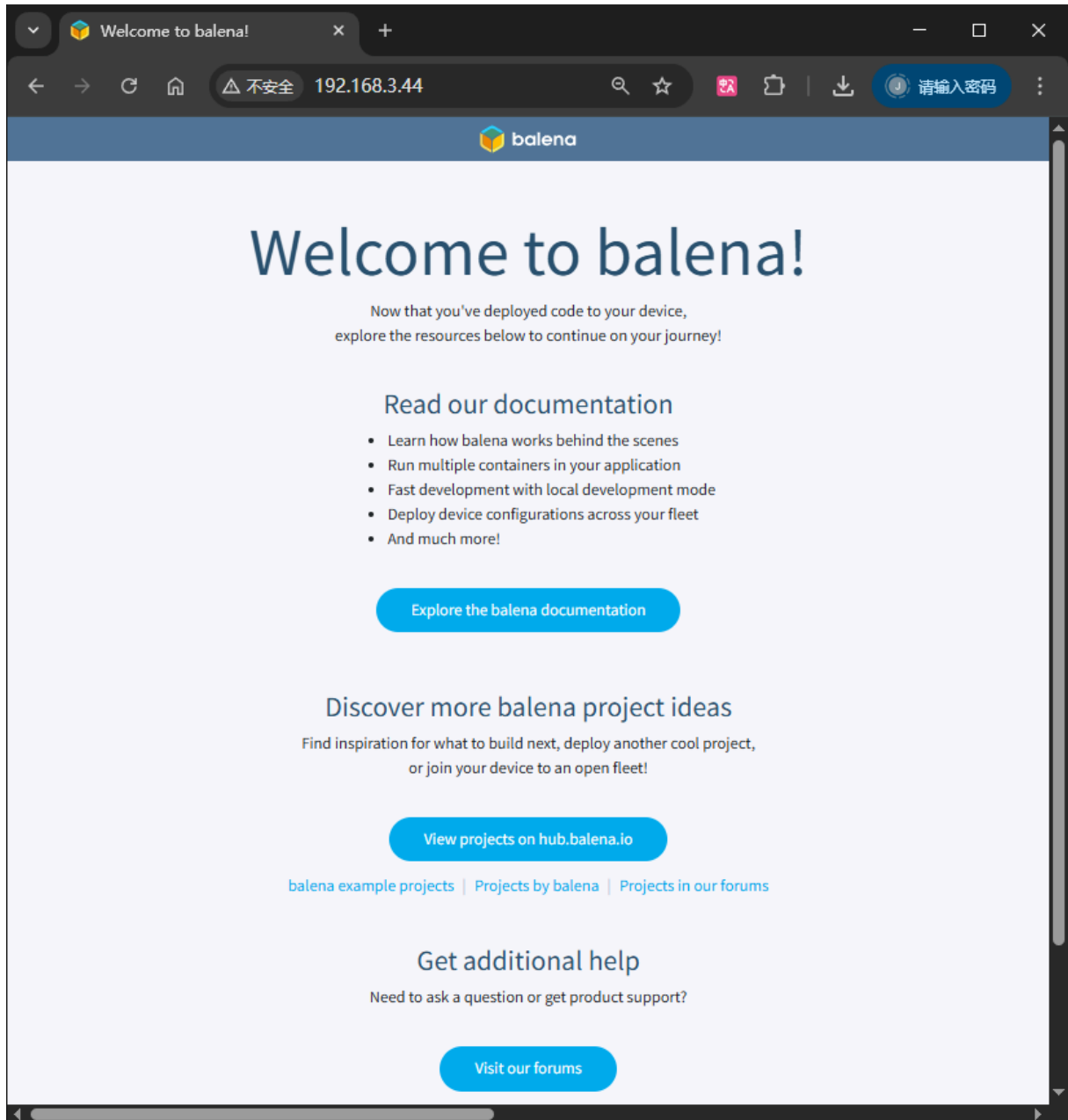


图 2-13 balena-python-hello-world 主页

至此，我们已经利用 BalenaCloud 完成了首个项目从部署到测试的完整流程。

## 2.4 启动项配置

如果您正在使用调试串口操作 ARM 板，会发现首次启动后运行一段时间后就无法操作调试终端了，即便重启也无法恢复。这是因为 BalenaOS 自动修改了 config.txt 禁用了调试串口。我们用如下方法恢复它：

- 进入 BalenaCloud 的设备管理页面

www.emtop-tech.com	github.com/EMTOP-TECH
sales@emtop-tech.com	support@emtop-tech.com



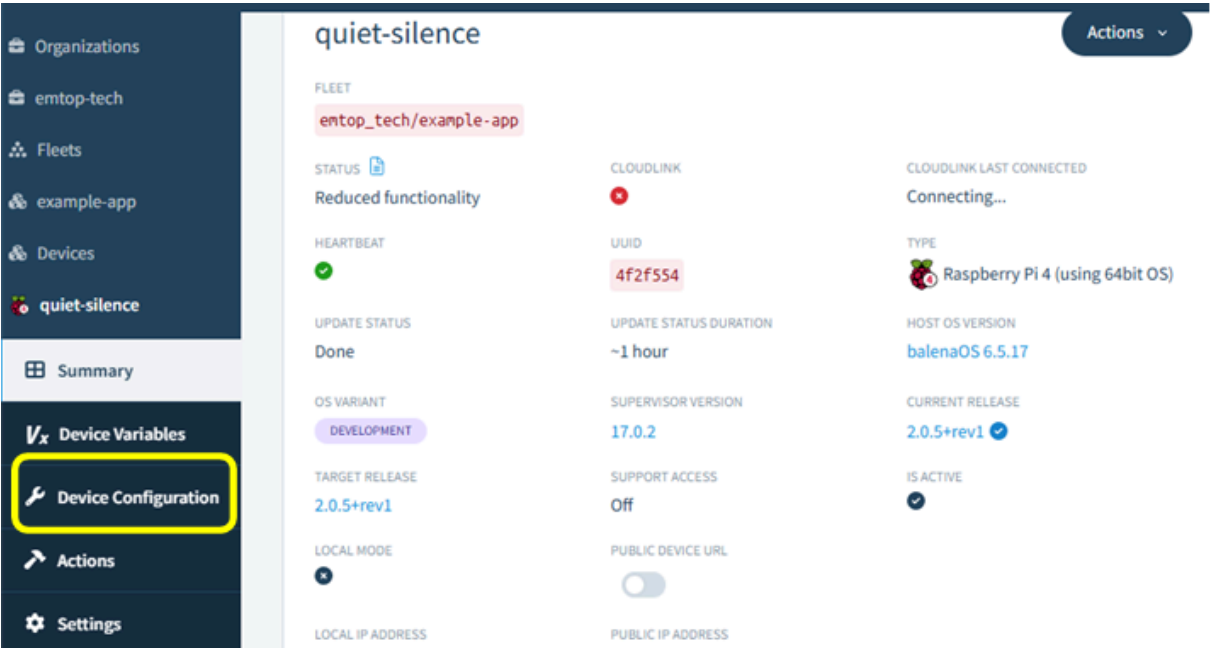


图 2-14 设备管理

- 点击侧边栏的“Device Configuration”，找到“Enable/Disable UART”一栏，点击蓝色框内的链接，如下图：

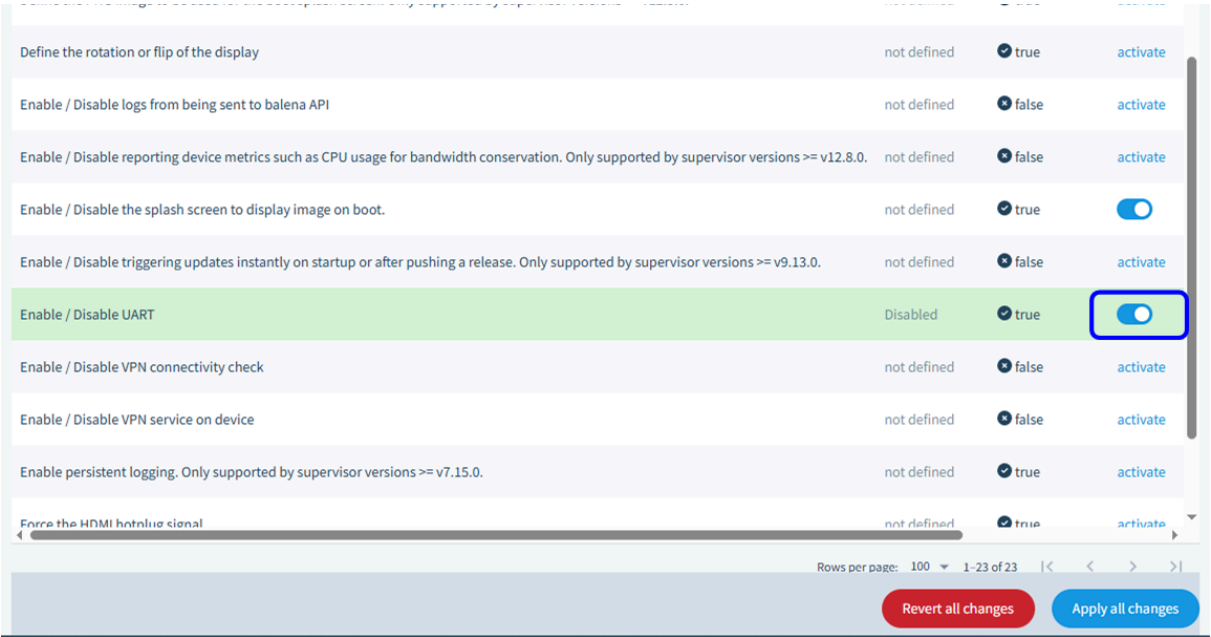


图 2-15 Enable UART

- 点击右下角的“Apply all changes”。手动重启 ARM 板，稍等片刻后就能看到调试串口上的登录信息又回来了。

设备配置

- 在 Device configuration 页面里，我们能看到很多选项可以配置，您可根据实际应用进行针对性的设置。
- 在页面的下半部还有“Custom configuration”，这里能增加自定义的配置。

## 2.5 SSH 登录 ARM 板

当我们的系统可以流畅运行且网络也能稳定联通之后，SSH 登录比调试串口更加便捷。

- 从 BalenaCloud 的设备管理页面获取 LOCAL IP ADDRESS，本例中 ARM 板的 IP 为 192.168.3.44。
- 在 PC Ubuntu 的终端窗口执行如下命令：

```
emtop@Ubuntu24:~# ssh -p 22222 root@192.168.3.44
```

```
The authenticity of host '[192.168.3.44]:22222 ([192.168.3.44]:22222)' can't be established.  
ED25519 key fingerprint is SHA256:vG/M++K6dWPExBT+8zdmA1MiQDjbv/HaFcLPHQSGwA.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '[192.168.3.44]:22222' (ED25519) to the list of known hosts.  
Last login: Fri Apr 18 08:07:34 2025 from 192.168.3.246
```

www.emtop-tech.com	github.com/EMTOP-TECH
sales@emtop-tech.com	support@emtop-tech.com

## 附录

### 词汇表

- **dashBoard**

BalenaCloud 平台中用于管理设备的功能面板。

- **openBalena**

Balena 平台的开源产品。设备运行 BalenaOS 并通过 Balena CLI 进行管理，仅适合本地局域网部署。OpenBalena 不包含 [dashBoard](#)。

- **BalenaCloud**

设备部署和管理基础设施，由 balena 托管。可理解为运行在公网上的增强版 openBalena。

- **BalenaEtcher**

运行在 Windows 下的 SD 卡/U 盘映像烧录工具，可支持裸数据和多种压缩格式文件烧录。