



# OK3399-C Linux 系统 Docker 编译环境指导

Embedded Development Platform

## Docker Manual

Rev. 1.0

2020/03/02

# 注意事项与维护



## 1、注意事项

- **请勿带电插拔核心板及外围模块！**
- 请遵循所有标注在产品上的警示和指引信息。
- 请保持本产品干燥。如果不慎被任何液体泼溅或浸润，请立刻断电并充分晾干。
- 使用中注意本产品的通风散热，避免温度过高造成元器件损坏。
- 请勿在多尘、脏乱的环境中使用或存放本产品。
- 请勿将本产品应用在冷热交替环境中，避免结露损坏元器件。
- 请勿粗暴对待本产品，跌落、敲打或剧烈晃动都可能损坏线路及元器件。
- 请勿使用有机溶剂或腐蚀性液体清洗本产品。
- 请勿自行修理、拆卸本公司产品，如产品出现故障请及时联系本公司进行维修。
- 擅自修改或使用未经授权的配件可能损坏本产品，由此造成的损坏将不予以保修。

## 2、售后维修

如产品使用过程中出现硬件故障可根据售后服务政策进行维修；

服务政策：参见官方网站 [www.forlinx.com](http://www.forlinx.com) 售后服务说明；

地 址：河北省保定市高开区向阳北大街 2699 号飞凌嵌入式技术有限公司新楼五层售后维修部

联 系 人：售后维修部

电 话：0312-3102650-952、953 邮编：071000

邮寄须知：建议使用顺丰、圆通或韵达，且不接收任何到付。

。

# 技术支持与定制

## 1、技术支持范围

- 1.1 本公司产品的软、硬件资源提供情况咨询；
- 1.2 本公司产品的软、硬件手册使用过程中遇到的问题；
- 1.3 本公司提供的 OEM、ODM 售后技术支持；
- 1.4 本公司产品的故障判断及售后维修服务；

## 2、技术讨论范围

- 2.1 源码的修改以及理解；
- 2.2 操作系统如何移植；
- 2.3 用户在自行修改以及开发中遇到的软硬件问题；

注：以上三点虽不属于技术支持范围，但我公司会尽力为用户提供帮助，如依然没能解决您的问题，敬请谅解；

## 3、技术支持方式

- 3.1 电话：0312-3119192
- 3.2 论坛：bbs.witech.com.cn
- 3.3 邮箱：

Linux 技术支持：	linux@forlinx.com
Android 技术支持：	android@forlinx.com
硬件技术支持：	hardware@forlinx.com

- 3.4 知识库：bbs.witech.com.cn/kb

## 4、技术支持时间

周一至周五：上午 9:00—11:30，下午 13:30—17:00；

公司按照国家法定节假日安排休息，在此期间无法提供技术支持，期间请发邮箱或论坛技术支持区，我们会在工作日尽快给您回复。

## 5、定制开发服务

我公司提供嵌入式操作系统底层驱动、硬件板卡的有偿定制开发服务，以缩短您的产品开发周期；

了解定制流程：<http://www.forlinx.com/OEM.htm>

填写需求文档：<http://www.forlinx.com/docs/PR.docx>

发至项目邮箱：project@forlinx.com

## 资料更新与获取

### 1、资料的更新

产品相关资料会不断的完善更新，包括本手册内容亦然如此；当您在使用这些内容时，请确保其为最新状态；

### 2、更新后如何通知

飞凌嵌入式产品资料更新通知采用微信公众号推送，敬请关注！



订阅号

### 3、资料如何获取

#### 3.1 网络下载：

请注册并登陆“[bbs.witech.com.cn](http://bbs.witech.com.cn)”找到“[开发板资料下载](#)”选择对应平台下载；

下载前请阅读《资料下载说明》：<http://bbs.witech.com.cn/thread-67932-1-1.html>；

#### 3.2 光盘：

请联系我公司销售人员购买；

## 版权声明

本手册版权归保定飞凌嵌入式技术有限公司所有。未经本公司的书面许可，任何单位和个人无权以任何形式复制、传播、转载本手册的任何部分，违者将被追究法律责任。

## 更新记录

日期	手册版本	底板版本	核心板版本	更新内容
2019.03.02	V1.0	V1.0 及其以上版本	V1.1 及其以上版本	第一版

## 目 录

注意事项与维护 .....	- 1 -
技术支持与定制 .....	- 2 -
资料更新与获取 .....	- 3 -
版权声明 .....	- 3 -
更新记录 .....	- 4 -
目 录 .....	- 5 -
第一章 Docker 安装.....	- 6 -
1.1 安装 Ubuntu 编译环境 .....	- 6 -
1.2 安装编译 Docker 系统所需要的库 .....	- 6 -
第二章 Docker 镜像还原 .....	- 7 -
附录 1: .....	- 9 -
F1.1 容器操作.....	- 9 -
F1.2 镜像操作.....	- 9 -

# 第一章 Docker 安装

## 1.1 安装 Ubuntu 编译环境

为什么使用 docker 作为编译环境，根据笔者经验编译环境安装非常麻烦，由于客户安装时候可能软件版本已经有更新，所以客户安装时候与笔者安装时候会遇到不同的问题。Docker 镜像已经被笔者验证可以使用，所以推荐使用 docker 编译。

在这里建议用户使用 Ubuntu 真机安装 docker，我们运行 docker 的环境 为 **Ubuntu18.04.1 LTS**，建议使用 Ubuntu14 及以上版本安装 docker。

注：推荐电脑配置处理器：Core(TM) i7 内存：8G 以上；

## 1.2 安装编译 Docker 系统所需要的库

Docker 一些工具包。本节操作前必须确保您的计算机或虚拟机能正常连接互联网，如您在安装中出现网络断开连接请再按照以下步骤进行安装。

本章所有操作均以在 **普通用户** 下为例，前缀为 '\$' 的操作代表是在普通用户下的操作。

步骤 1. 安装必要的一些系统工具

```
$ sudo apt-get update
```

```
$ sudo apt-get -y install apt-transport-https ca-certificates curl software-properties-common
```

步骤 2. 安装 GPG 证书

```
$ curl -fsSL http://mirrors.aliyun.com/docker-ce/linux/ubuntu/gpg | sudo apt-key add -
```

步骤 3. 写入软件源信息

```
$ sudo add-apt-repository "deb [arch=amd64] http://mirrors.aliyun.com/docker-ce/linux/ubuntu  
$(lsb_release -cs) stable"
```

步骤 4. 更新并安装 Docker-CE

```
$ sudo apt-get -y update
```

查找 Docker-CE 的版本

```
$ apt-cache madison docker-ce
```

输出： **docker-ce | 18.06.3~ce~3-0~ubuntu | http://mirrors.aliyun.com/docker-ce/linux/ubuntu/  
trusty/stable amd64 Packages**

安装指定版本的 Docker-CE

例如：

```
$ sudo apt-get -y install docker-ce=18.06.3~ce~3-0~ubuntu
```

步骤 5. 安装校验

```
$ sudo docker version
```

## 第二章 Docker 镜像还原

特别注意 **容器和镜像** 术语

请使用飞凌提供的 **docker** 镜像包,文件所在路径如下:

📁 用户资料\工具\docker 镜像\docker\_image.tar.bz2

复制此压缩包到虚拟机或者真实机器 **/root/workspace/rk3399/**路径下。

生产目录结构, 命令如下

```
$ mkdir -p /root/workspace/rk3399
$ cd /root/workspace/rk3399
```

### Docker 镜像释放

注意: 最终的 **docker** 镜像为 ok3399\_linux\_18.04.tar.bz2

**步骤 1:** 验证压缩包完整性, 与 docker\_image.tar.bz2-md5sum.txt 对比, 命令如下

```
$ md5sum docker_image.tar.bz2
```

**步骤 2:** 释放压缩包, 命令如下

```
$ tar xvf docker_image.tar.bz2
```

**步骤 3:** 从备份处加载 **docker** 镜像, 命令如下

```
$ docker load -i ok3399_linux_18.04.tar.bz2
```

**步骤 4:** 查看加载的**镜像 id** 和名字

```
$ docker image ls
```

REPOSITORY	TAG	IMAGE ID
ok3399_linux_18.04	latest	012b4d871cf7
days ago	1.23GB	2

**步骤 5:** 创建并运行**容器**

```
$ docker run -it --name test_ubuntu --privileged=true --cap-add=SYS_ADMIN -v /root/worksp
ace/rk3399:/home/forlinx/OK3399 -p 10900:22 012b4d871cf7 /bin/bash
```

说明: 此命令将本地文件夹/root/workspace/rk3399 (冒号前路径) 和容器里文件夹

/home/forlinx/OK3399 (冒号后路径) 建立映射关系, 注意此处的两个文件夹的路径都要是绝对路径。

之后变成下边提示符说明进入**容器**

```
root@630df1b5b0f2:/#
```

退出**容器**并关闭**容器**的命令: **exit**

**步骤 6:** 查看所有**容器**, 可以看到刚才创建的**容器**:

```
$ docker container ls -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	
630df1b5b0f2	012b4d871cf7	"/bin/bash"	5 minutes ago
minutes ago	test_ubuntu	Exited (0) 4	

**步骤 7:** 开启**容器**, 进入**容器**, 准备编译

```
# docker start 630df1b5b0f2
# docker exec -it 630df1b5b0f2 bin/bash
```

进入**容器**后状态

```
root@630df1b5b0f2:/#
root@630df1b5b0f2:/# cd /root/workspace/rk3399
```

**步骤 8:** 切换为普通用户



容器默认的用户是 `root` 用户，`root` 用户拥有很高的权限，同时也意味着更高的风险。同时，`linux` 的源码编译是在普通用户状态下操作的，所以建议切换为普通用户再进行编译等操作。

容器中提供了 `forlinx` 的普通用户。`root` 用户的密码是 `forlinx`，`forlinx` 用户的密码是 `123456`。

通过如下指令切换为普通用户 `forlinx`：

```
root@630df1b5b0f2:/# su forlinx
```

```
forlinx@630df1b5b0f2:/$
```

**步骤9：**在容器内进行代码编译。

## 附录 1:

### F1.1 容器操作

容器关闭: `# docker stop 630df1b5b0f2`  
容器开启: `# docker start 630df1b5b0f2`  
进入容器: `# docker exec -it 630df1b5b0f2 bin/bash`  
退出容器: `# exit`  
查看所有容器: `# docker ps -a`  
查看所有正在运行容器: `# docker ps`

### F1.2 镜像操作

查看所有存在镜像: `# docker images ls`