



## IAC-IMX8MM-Kit Linux用户手册

版本号: V2.0  
2021年04月

浙江启扬智能科技有限公司版权所有

QIYANG TECHNOLOGY Co., Ltd

Copyright Reserved

## 版本更新记录

版本	硬件平台	描述	日期	修订人
1.0	IAC-IMX8MM-MB V1.10	初始版本, 首次发布	2020-04	zhujh
2.0	IAC-IMX8MM-MB V1.20	更新图片	2021-04	wwx

## 目 录

前言 .....	4
公司简介 .....	4
IAC-IMX8MM-KIT 开发/评估板的使用建议 .....	4
一、准备工作.....	5
二、系统镜像烧写（固件烧写） .....	6
三、功能说明与测试.....	7
四、搭建嵌入式 Linux 开发环境 .....	8
4.1 安装交叉编译器 .....	8
4.2 NFS 网络文件系统 .....	8
五、测试源码编译.....	9
六、U-Boot 编译 .....	10
6.1 编译 imx_v2018.03_4.14.98_2.0.0_ga.....	11
6.2 编译 imx-atf.....	11
6.3 编译 imx-mkimage .....	11
七、内核的编译.....	12
八、应用程序的开发.....	13
8.1 Hello World.....	13
8.2 交叉编译 .....	14
8.3 运行应用程序 .....	14
8.4 添加应用程序到文件系统.....	15
九、小结.....	16

## 前言

### 公司简介

浙江启扬智能科技有限公司2007年成立于杭州，是一家专注于ARM嵌入式产品研发、生产与销售的国家高新技术企业。10余年的积累与沉淀，成功构建了产品从开发到量产的服务链。

作为公司的核心，启扬研发团队由30余位嵌入式工程师组成，致力于为用户提供简单易用的嵌入式硬件、软件工具以及定制化的产品解决方案。已广泛应用于工控、物联网、新零售、医疗、电力、环境监测、充电桩等领域。

设立于诸暨的生产基地为启扬提供了强有力的保障，占地面积5000平米，拥有2条SMT产线，通过并严格遵循ISO9001质量管理体系认证指导生产。依托雄厚的生产实力，年产能可达100万套，保证用户交期，解决后顾之忧。

启扬拥有完善的销售市场网络，专业的销售和售后团队为用户提供全方位的技术支持与服务。业务已遍及120多个国家和地区，成功帮助2000多家用户将产品快速高效地推向市场。

研发、产能、市场的结合与延伸，为启扬智能成为专业化、全球化的嵌入式软硬件供应商奠定了坚实的基础。

我们为您提供：

- **多平台软/硬件产品**

NXP、Rockchip、MTK、Renesas、TI、Atmel、Cirrus Logic等多平台ARM开发板/核心板/工控板和周边硬件产品以及支持用户快速二次开发的配套工具与软件资源。

- **定制服务**

充分发挥在ARM平台及Linux、Android、Ubuntu操作系统上的技术累积，为用户提供量身定制嵌入式产品服务（OEM/ODM）。

感谢您使用启扬智能的产品，我们会尽最大努力为您提供技术协助！祝愿您工作顺利！

### IAC-IMX8MM-KIT 开发/评估板的使用建议

- 1) 使用开发板之前，请务必先阅硬件说明书《IAC-IMX8MM-Kit 硬件说明书》；
- 2) 使用前请详细核对装箱单，是否齐全；
- 3) 了解开发板的基本结构和组成，包括硬件资源的分配，核心板与底板的各个引脚定义，以及扩展引脚定义等等；
- 4) 如果您需要在Linux系统或者Android系统下进行设计开发或者功能测试，还需要阅读《IAC-IMX8MM-KIT 系统镜像烧写手册》、《IAC-IMX8MM-KIT 测试手册》；
- 5) IAC-IMX8MM-KIT 嵌入式开发板，接受批量订购。

## 一、准备工作

- ◆ 准备一台安装 Windows 7 或 Windows 10 的 PC 机。
- ◆ PC 机安装了虚拟机，虚拟机中安装有 Linux 系统（Ubuntu 或其它 Linux 发行版）或者其他安装 Linux 系统 PC 机，若未安装，可参考提供的《虚拟机安装 Ubuntu 指导手册》进行安装，本手册以 Windows 10 下虚拟机安装的 Ubuntu 14.04 操作为例。
- ◆ 串口连接：通过 3Pin 串口线,将开发板 DEBUG 调试串口（J18）与 PC 机串口连接。  
（如 PC 机没有串口或者使用笔记本电脑，还需要自己准备一根 USB 转串口线）
- ◆ 网络连接：通过网线将开发板的以太网接口(J11)与 PC 机的网络接口连接，或者通过交换机连接。
- ◆ USB 连接:通过 USB 连接线将开发板的 USB Device(J42)与 PC 机的 USB 接口连接。
- ◆ 串口设置：打开 Windows 下终端通讯软件或光盘资料工具目录下提供的 SecureCRT 串口终端软件，选择连接开发板所用的串口并设置如下参数：波特率（115200）、数据位（8 位）、停止位（1 位）、校验位（无）、数据流控制（无）。

## 二、系统镜像烧写（固件烧写）

IAC-IMX8MM-Kit 开发板镜像烧写采用原厂提供的专用烧写工具 `uuu.exe`，具体烧写方法请参考《IAC-IMX8MM-Kit 系统镜像烧写手册.pdf》。

正常出货已根据订单要求烧写镜像；

### 三、功能说明与测试

IAC-IMX8MM-Kit 开发板系统镜像已集成测试程序，系统启动后可在/usr/test/ 目录下找到相应测试程序，具体测试请参考《IAC-IMX8MM-Kit 功能说明与测试手册.pdf》

## 四、搭建嵌入式 Linux 开发环境

### 4.1 安装交叉编译器

Uboot 修改、Kernel 驱动裁剪、应用程序开发等都需要通过 ARM-Linux 交叉编译，这里需要先安装一下交叉编译工具链，交叉编译器在光盘资料/交叉编译器有提供，用户可直接安装使用。

将交叉编译工具拷贝到已经安装好的 Ubuntu 环境或者其他 Linux 发行版环境中，如没有 Ubuntu 环境，请参考《虚拟机安装 Ubuntu 指导手册》进行安装；

交叉编译器安装：

```
$ ./fsl-imx-wayland-glibc-x86_64-fsl-image-qt5-validation-imx-aarch64-toolchain-4.14-sumo.sh
```

输入安装路径这个直接敲回车，采用默认路径

输入 Y，确定，如下图所示，安装成功。

```
./fsl-imx-wayland-glibc-x86_64-fsl-image-qt5-validation-imx-aarch64-toolchain-4.14-sumo.sh
NXP i.MX Release Distro SDK installer version 4.14-sumo
=====
Enter target directory for SDK (default: /opt/fsl-imx-wayland/4.14-sumo):
You are about to install the SDK to "/opt/fsl-imx-wayland/4.14-sumo". Proceed[Y/n]? Y
[sudo] password for zhujh:
Extracting SDK.....done
.....done
Setting it up...done
SDK has been successfully set up and is ready to be used.
Each time you wish to use the SDK in a new shell session, you need to source the environment setup script e.g.
$ . /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux
```

### 4.2 NFS 网络文件系统

NFS 是 Network FileSystem 的简称，可以让不同的主机通过网络访问远端的 NFS 服务器共享出来的文件。这样，将主机当作 NFS 服务器，我们就可以在开发板上通过网络访问主机的文件。在嵌入式中使用 NFS 使得应用程序的开发十分方便，如在 Ubuntu 环境搭建时已安装，可跳过此节。

具体的使用方法如下：

#### 1. 安装服务

```
$sudo apt-get install nfs-kernel-server
```



2. 创建共享目录, 修改 /etc/exports 文件末尾

```
$sudo vi /etc/exports
```

在文件末尾加入下面一行:

```
/*(rw, sync, no_root_squash)
```

```
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes      hostname1(rw, sync, no_subtree_check) hostname2(ro, sync, no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4       gss/krb5i(rw, sync, fsid=0, crossmnt, no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw, sync, no_subtree_check)
#
# *(rw, sync, no_root_squash)
~
~
~
~
~
```

这样, 就将主机的根目录当作了共享目录, 也可自己配置共享目录。

3. 启动 NFS 服务器, 重启 nfs-kernel-server 服务:

```
$ sudo /etc/init.d/nfs-kernel-server restart
```

4. 本地验证 NFS

```
$ sudo mount local:/ /mnt/
```

```
$ ls /mnt/
```

这时可以在/mnt/目录下查看到根目录下的内容, 则说明 NFS 配置正确。

## 五、测试源码编译

在光盘/源码/测试源码目录下, 提供了各测试部分源码, 可根据需要进行修改编译。所  
有任何技术问题或需要帮助, 请联系: [supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品, 请联系销售: [sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问: <http://www.qiytech.com>

©2020 Qiyangtech 版权所有

有以下操作均需在系统已安装交叉编译器并加入到环境变量中前提下执行，若未安装交叉编译器，则需参照 4.1 安装交叉编译器。

选择要进行编译的模块源码，此处以 `buzzer` 为例，将目录：`光盘/源码/测试源码/buzzer_test`，拷贝到 `ubuntu 14.04` 目录下，打开 `linux` 终端，进入到 `buzzer_test` 目录，执行 `make` 命令，在该目录下将得到编译后的可执行文件 `buzzer_test`：

```
$ cd buzzer_test/
```

```
$ ls
```

```
buzzer_test.c  Makefile
```

```
$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux
```

```
$ make
```

## 六、U-Boot 编译

光盘提供了已经编译好的引导镜像及源码，请尽量使用光盘自带镜像作为引导程序，若需对该引导程序进行重新编译，请参照如下步骤执行，执行如下命令前，请务必已经正确安装交叉编译器，并添加到环境变量中，若未安装，请参考本手册 4.1 小结进行安装：

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

第 10 页 共 18 页

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2020 Qiyangtech 版权所有

源码:

```
imx-atf_imx_4.14.98_2.0.0_ga.tar.gz
imx-mkimage_imx_4.14.98_2.0.0_ga.tar.g
uboot-imx_v2018.03_4.14.98_2.0.0_ga.tar.gz
```

解压光盘中的 u-boot 源码压缩包

```
$tar zxvf uboot-imx_v2018.03_4.14.98_2.0.0_ga.tar.gz
$tar zxvf imx-atf_imx_4.14.98_2.0.0_ga.tar.gz
$tar zxvf imx-mkimage_imx_4.14.98_2.0.0_ga.tar.gz
```

## 6.1 编译 imx\_v2018.03\_4.14.98\_2.0.0\_ga

```
$ cd imx_v2018.03_4.14.98_2.0.0_ga/
$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux
$ make imx8mm_qiyang_defconfig
$ make -j40
```

编译无误后，生成如下文件：

```
u-boot-nodtb.bin
spl/u-boot-spl.bin
arch/arm/dts/fsl-imx8mm-qiyang.dtb
```

## 6.2 编译 imx-atf

```
$ cd imx-atf/
$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux
$ LDFLAGS="" make PLAT=imx8mm
```

编译无误后，生成如下文件：

```
build/imx8mm/release/bl31.bin
```

## 6.3 编译 imx-mkimage

编译 imx-mkimage 之前确保已经编译 imx-atf 和 imx\_v2018.03\_4.14.98\_2.0.0\_ga

```
$ cd imx-mkimage/
$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux
$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/u-boot-nodtb.bin iMX8M/
$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/spl/u-boot-spl.bin iMX8M/
$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/arch/arm/dts/fsl-imx8mm-qiyang.dtb iMX8M/
```

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

第 11 页 共 18 页

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2020 Qiyangtech 版权所有

```
$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/tools/mkimage ./mkimage_uboot
$ cp ../imx-atf/build/imx8mm/release/bl31.bin iMX8M/
$ make SOC=iMX8MM flash_evk
```

编译无误后，生成如下文件：

iMX8M/flash.bin

flash.bin 就是烧写到开发板的 uboot 镜像文件

## 七、内核的编译

光盘中有配置好的内核源码文件 kernel\_imx\_4.14.98\_2.0.0\_ga.tar.gz

压缩包，将它拷到一个 Ubuntu14.04 目录下，用户可以自行配置内核。编译内核之前，请确保当前的交叉编译器已正确安装到系统中。

```
$ tar zxvf kernel_imx_4.14.98_2.0.0_ga.tar.gz
$ cd imx_4.14.98_2.0.0_ga/
$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux
$ make defconfig
$ LDFLAGS="" CC="$CC" make -j40
```

编译无误后，生成如下文件：

```
arch/arm64/boot/Image  
arch/arm64/boot/dts/freescale/fsl-imx8mm-qiyang.dtb
```

## 八、应用程序的开发

### 8.1 Hello World

您可以在 PC 机上自由地开发应用程序，这里以 Hello World 为例。首先编写一个 Hello World 程序代码如下：

```
#include <stdio.h>  
  
int main(void)  
{  
  
    printf("Hello World ! \n");
```

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

第 13 页 共 18 页

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2020 Qiyangtech 版权所有

```
return 0;  
  
}
```

保存在 `hello.c` 文件，我们用如下命令对它进行交叉编译。

## 8.2 交叉编译

要使得程序能够运行在开发板上，需要使用之前安装好的交叉编译器对应用程序进行编译。将以上文件放到 `Ubuntu14.04` 目录中，我们可以使用如下命令进行编译：

```
$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux  
$ $CC -o hello hello.c
```

编译后可以看到在当前目录生成了用于 `ARM` 平台的可执行的二进制文件 `hello`。

## 8.3 运行应用程序

这样我们就得到了可以在开发板上执行的程序了，下面介绍如何在主板运行应用程序，这里介绍了两种方法：挂载 `NFS` 服务器和添加到文件系统。

我们可以使用之前配置好的 `NFS` 服务器，在开发板上挂载主机的 `NFS` 服务器后，就可以在开发板上操作主机上的文件，如复制文件、运行程序等。这种方式十分方便于调试，具体实现方法如下：

- 1、确保通过网线将开发板与 `PC` 机连接好，并在 `PC` 机上开启了 `NFS` 服务
- 2、设置开发板的 `IP` 与 `PC` 机的 `IP` 在同一网段，如：

PC 机 IP : 192.168.1.75

目标板 IP: 192.168.1.203

Network Mask: 255.255.255.0

Broadcast IP: 192.168.1.255

- 3、测试网络连接

在开发板上 `ping` 主机，在超级终端运行如下命令：

```
# ping 192.168.1.75
```

```

root@imx8mmqiyang:~# ping 192.168.1.75
PING 192.168.1.75 (192.168.1.75) 56(84) bytes of data.
64 bytes from 192.168.1.75: icmp_seq=1 ttl=64 time=2.81 ms
64 bytes from 192.168.1.75: icmp_seq=2 ttl=64 time=1.30 ms
64 bytes from 192.168.1.75: icmp_seq=3 ttl=64 time=1.30 ms
64 bytes from 192.168.1.75: icmp_seq=4 ttl=64 time=1.31 ms
64 bytes from 192.168.1.75: icmp_seq=5 ttl=64 time=2.10 ms
64 bytes from 192.168.1.75: icmp_seq=6 ttl=64 time=2.11 ms
64 bytes from 192.168.1.75: icmp_seq=7 ttl=64 time=2.10 ms
64 bytes from 192.168.1.75: icmp_seq=8 ttl=64 time=2.10 ms
64 bytes from 192.168.1.75: icmp_seq=9 ttl=64 time=2.10 ms
    
```

在主机上以同样的方式 ping 开发板，如果主机和开发板可以互相 ping 通，则说明网络连接正常。

#### 4、挂载主机 NFS 服务器

在超级终端输入以下命令：

```

# mount -o nolock 192.168.1.75:/mnt/

# cd /mnt/
    
```

挂载正确后，可以在开发板的/mnt 目录下看到主机的根目录，然后可以进入到程序所在的目录运行应用程序。通过这种方式，可以方便调试程序，直到程序调试正确后，再将应用程序添加到文件系统中，烧写到 emmc。这样，避免了不断的烧写程序到 emmc。

## 8.4 添加应用程序到文件系统

在将应用程序调试好后，可以将它直接添加到文件系统，这样您就可以把它和文件系统一起烧到 emmc 上了。下面介绍如何将应用程序添加到根文件系统，从光盘中拷贝根文件系统源码到 Ubuntu14.04，并解压。

先新建一个文件夹 filesystem，解压出来的文件就放在这个文件夹里。

```
$ mkdir filesystem
```

把光盘中的文件系统压缩文件 rootfs.tar.bz2 拷贝到该文件目录下。

```
$ tar jxvf rootfs.tar.bz2 //对解压出来的 filesystem 目录进行操作
```

```
$ ls
```

```
bin boot dev etc home lib media mnt opt proc run sbin sys tmp unit_tests usr var
```

可以在所列的目录中加入您的应用程序。

压缩该目录下的文件，压缩之前先删除原先的 rootfs.tar.bz2

```
$ rm -rf rootfs.tar.bz2
```

```
$ tar -jcvf rootfs.tar.bz2 -R *
```

压缩完后，在当前目录下就有一个新生成的 rootfs.tar.bz2 文件。

## 九、小结

可能遇到的问题汇总：



## 浙江启扬智能科技有限公司

电话：0571-87858811 / 87858822

传真：0571-89935912

技术支持：0571-87858811 转 805

E-MAIL: [supports@qiyangtech.com](mailto:supports@qiyangtech.com)

网址： <http://www.qiytech.com>

地址：浙江省杭州市西湖科技园西园八路6号A幢3楼

邮编：310030