MYD-CZU3EG Board
Linux 开发手册

V1.0

版本记录

版本号	说明	时间
V1.0	初始版本	2019/04/30

目录

第 1 章概述及软件资源介绍

1.1 概述

MYD-CZU3EG 提供了丰富的系统资源和软件资源，本手册将从环境搭建开始，一步步介绍如何进行 Linux 开发。本手册中开发主机上的命令以 Ubuntu 为例进行讲解。

1.2 软件资源

类别	名称	备注	源码
Tool chains	gcc 5.2.1	gcc version 5.2.1(Linaro GCC 5.2)	
Bootloader	boot.bin	一级引导程序，包括 fsbl、u-boot	Yes
Linux Kernel	Linux 4.9.0	专为 MYD-CZU3EG 硬件制定的 Linux 内核	Yes
Driver	USB2.0/3.0 Host	USB2.0/3.0 Host 驱动	Yes
	Ethernet	千兆以太网驱动	Yes
	MMC/SD/TF	MMC/SD/TF 卡驱动	Yes
	Qspi flash	Qspi flash 驱动	Yes
	CAN	CAN 驱动	Yes
	DP	DP 显示驱动	Yes
	HDMI	HDMI 驱动	Yes
	LCD	LCD 驱动	Yes
	SATA	SATA 硬盘驱动	Yes
	I2C	I2C 驱动	Yes
	UART	串口驱动	Yes
	Watchdog	Watchdog 看门狗驱动	Yes
	GPIO	GPIO 驱动	Yes
	LED	LED 驱动	Yes
	Button	Button 按键驱动	Yes
Application	Led	Led 例程	Yes
	Button	Button 例程	Yes
	CAN	CAN 例程	Yes
	Uart	Rs232 例程	Yes
	Net	Socket 例程	Yes
File system	Ramdisk	Ramdisk 系统镜像	
	Rootfs	Buildroot 制作，包含 Qt	Yes

表 1-1

第 2 章 Linux 开发环境搭建

2.1 建立工作目录

拷贝产品光盘中的资料到主机中，本文中的<WORKDIR>用来表示主机上的工作目录，例如*“/home/myir/Workspace/”*，请保证目录访问权限。

```
# mkdir -p <WORKDIR>
# cp -a <DVDROM>/03-Linux_Source/* <WORKDIR>
```

2.2 设置交叉编译工具

```
# cd <WORKDIR>/Toolchain
# tar xvf aarch64-linux-gnu.tar.gz
# export PATH=$PATH:<WORKDIR>/Toolchain/aarch64-linux-gnu/bin
```

执行完“export”命令后输入 arm 按 Tab 键来检查是否设置成功，该设置只对当前终端有效，如需永久修改，请将以上 export 命令添加到用户启动脚本文件：~/.bashrc。

第 3 章 Linux 系统编译

3.1 编译 Bootloader

进入 Bootloader 目录，解压 U-boot 源码：

```
# cd <WORKDIR>/Bootloader
# tar jxvf u-boot-xlnx.tar.bz2
# cd u-boot-xlnx
```

开始编译：

```
# make ARCH=arm CROSS_COMPILE=aarch64-linux-gnu- distclean
# make ARCH=arm CROSS_COMPILE=aarch64-linux-gnu- zynqmp_myd_defconfig
# make ARCH=arm CROSS_COMPILE=aarch64-linux-gnu-
```

编译完成后，在当前目录下会生成 “u-boot.elf” ELF 文件。使用此镜像制作 boot.bin。

3.2 编译 Linux 内核

进入 Kernel 目录，解压内核源码：

```
# cd <WORKDIR>/Kernel
# tar jxvf linux-xlnx.tar.bz2
# cd linux-xlnx
```

开始编译：

```
# make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- distclean
# make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- zynqmp_myd_defconfig
# make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- Image
# make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- dtbs
```

编译完成后，会在 arch/arm64/boot 目录下生成 Image 文件，在 arch/arm64/boot/dts/xilinx/ 目录下生成 zynqmp-myd.dtb 文件，烧写时需重命名为 devicetree.dtb。

3.3 构建 QT 根文件系统

这里使用 buildroot 来构建根文件系统。

首先进入文件系统目录，解压 buildroot 源码：

```
$ cd <WORKDIR>/Filesystem
$ tar jxvf myir-buildroot.tar.bz2
$ cd myir-buildroot
```

将目录中的 zynqmp_myd_config 文件重命名为.config：

```
$ cp zynqmp_myd_config .config
```

配置 buildroot：

```
$ make menuconfig
```

设置交叉编译工具路径：

```
Toolchain  --->
    ()  Toolchain path
```

进入 Toolchain path 选项，填入编译器路径：

```
<WORKDIR>/Toolchain/aarch64-linux-gnu/
```

退出配置，开始编译：

```
$ make
```

编译完成之后将在“myir-buildroot/output/images”目录内生产 rootfs.tar 文件。

3.4 修改文件系统

3.4.1 修改根文件系统 tar 包

（1）拆 tar 包

```
# cd <WORKDIR>/Filesystem
# mkdir -p rootfs
# sudo tar xvf rootfs.tar -C ./rootfs/
```

（2）修改文件夹，加入新文件。

（3）打包 tar

```
# cd ./rootfs
# sudo tar cvf ../rootfs.tar ./*
```

3.4.2 修改 Ramdisk 文件系统

Ramdisk 主要作为烧写 QSPI，eMMC 时使用。

（1）挂载 Ramdisk

新建目录 tmp，并将 uramdisk.image.gz 拷贝至该目录

```
# cd <WORKDIR>/Filesystem
```

```
# mkdir tmp
# cp uramdisk.image.gz tmp/
# cd tmp/
```

去掉 mkimage 生成的 64 bytes 的文件头，生成新的 ramdisk.image.gz

```
# dd if=uramdisk.image.gz of=ramdisk.image.gz bs=64 skip=1
```

gunzip 解压 ramdisk.image.gz 生成 ramdisk.image

```
# gunzip ramdisk.image.gz
```

新建挂载目录“ramdiskdir”，并将 ramdisk.image 挂载

```
# mkdir -p ramdiskdir
# sudo mount -o loop,rw ramdisk.image ramdiskdir
```

进入 ramdiskdir 目录，根据需要做修改。

（2）重新生成 ramdisk

同步文件系统并卸载 ramdisk

```
# sync
# sudo umount ramdiskdir
```

用 gzip 压缩 ramdisk.image，生成 ramdisk.image.gz

```
# gzip -9 ramdisk.image
```

用 mkimage 添加文件头，生成新的 uramdisk.image.gz

```
# ./mkimage -A arm -T ramdisk -C gzip -n Ramdisk -d ramdisk.image.gz uramdisk.image.gz
```

删除临时文件 ramdisk.image.gz

```
# rm ramdisk.image.gz
```

第 4 章 Linux 系统烧写

这里提供的烧写方法为通过 TF 卡启动 Linux，进入 Ramdisk 文件系统中的脚本烧写 Bootloader，内核到 QSPI-Flash，烧写根文件系统到 eMMC。

4.1 准备烧写文件

将产品资料光盘 Images 文件夹中的镜像文件复制到 TF 卡中，所需的文件如下表所示：

文件名称	说明
BOOT.bin	系统启动程序，包括 fsbl、和 u-boot。具体的制作方法将在《MYD-CZU3EG 入门指导手册》中描述
Image	Linux 内核
devicetree.dtb	设备树文件
uramdisk.image.gz	Ramdisk 文件系统，烧写时 Linux 挂载的根文件系统
rootfs.tar	根文件系统，准备烧写到 eMMC 的文件系统（光盘中提供的 rootfs.tar 包含 Qt）

表 4-1

4.2 开始烧写

（1）将开发板的启动模式 switch 开关 SW1 的 1 拨到 OFF，2 拨到 ON，3 拨到 OFF，4 拨到 ON，设置成 TF 卡启动模式；

（2）插入已存入烧写文件的 TF 卡，连接串口波特率为 115200，开发板上电；

（3）开发板将引导进入 Ramdisk 文件系统，进入 Linux 命令行，输入命令开始更新：

```
Welcome to myir board
myir login: root
[root@myir ~]#/updatesys.sh /mnt/mmcblk1p1
```

脚本将把 BOOT.bin，devicetree.dtb，Image 烧写到 QSPI-Flash，把 rootfs.tar 烧写到 eMMC。

（4） 烧写完成之后，将开发板的启动模式 switch 开关 SW1 的 1 拨到 ON，2 拨到 OFF，3 拨到 ON，4 拨到 ON，设置成 Qspi flash 启动模式，重新上电，就可以进入烧写

的 rootfs 文件系统。

第 5 章 Linux 应用程序

MYD-CZU3EG Board 提供了常用外设的演示程序，程序以及源码都位于*“<WORKDIR>/Examples/”*，请根据目录内的 Makefile 进行编译：

```
# cd <WORKDIR>/Examples/
# make CROSS_COMPILE=aarch64-linux-gnu- clean
# make CROSS_COMPILE=aarch64-linux-gnu-
```

连接调试串口 J15，设置波特率为 115200，数据位为 8，停止位为 1，无奇偶校验，将对应的可执行程序拷贝到开发板，在开发板上运行程序时需要注意权限，如果无法执行请使用如下命令：

```
# chmod +x <program-to-be-executed>
```

5.1 Led

本例程演示如何使用 Linux API 操作开发板上的 LED，详情请参考源码。

将目录*“<WORKDIR>/Examples/leds”*中的可执行程序 led-test 拷贝至开发板，把调试串口 J15 连到 PC 上。执行以下命令后，开发板上 LED 闪烁：

```
# ./led-test
```

5.2 Button

本例程演示如何使用 Linux API 读取开发板上的按键，详情请参考源码。

将目录*“<WORKDIR>/Examples/buttons”*中的可执行程序 btn-test 拷贝至开发板，把调试串口 J15 连到 PC 上。执行以下命令后，按下开发板的 S1 键，串口会输出相关信息：

```
# ./btn-test
```

5.3 CAN

本例程演示如何使用 Linux API 操作开发板上的 can，详情请参考源码。

将目录*“<WORKDIR>/Examples/can”*中的可执行程序 can-test 拷贝至开发板，把调试串口 J15 连到 PC 上。执行以下命令：

（注：请将板子上的 can 接口与外部 can 设备连接，来测试 can 通讯）

```
# ./can-test
```

5.4 Net

本例程演示如何使用 Linux API 进行网络通讯，详情请参考源码。

将目录 *“<WORKDIR>/Examples/network”* 中的可执行程序 arm-client 和 arm-server 拷贝至开发板，将 pc-client 和 pc-server 拷贝到主机的 ubuntu 系统中，把调试串口 J15 连到 PC 上。执行以下命令后，串口会输出相关信息：

主机端作为 server，输入如下命令

```
# ./pc-server
```

板子端作为 client，输入如下命令

```
# ./arm-client  192.168.xxx.xxx
```

附录一售后服务与技术支持

凡是通过米尔科技直接购买或经米尔科技授权的正规代理商处购买的米尔科技全系列产品，均可享受以下权益：

1、6 个月免费保修服务周期

2、终身免费技术支持服务

3、终身维修服务

4、免费享有所购买产品配套的软件升级服务

5、免费享有所购买产品配套的软件源代码，以及米尔科技开发的部分软件源代码

6、可直接从米尔科技购买主要芯片样品，简单、方便、快速；免去从代理商处购买时，漫长的等待周期

7、自购买之日起，即成为米尔科技永久客户，享有再次购买米尔科技任何一款软硬件产品的优惠政策

8、OEM/ODM 服务

如有以下情况之一，则不享有免费保修服务：

1、超过免费保修服务周期

2、无产品序列号或无产品有效购买单据

3、进液、受潮、发霉或腐蚀

4、受撞击、挤压、摔落、刮伤等非产品本身质量问题引起的故障和损坏

5、擅自改造硬件、错误上电、错误操作造成的故障和损坏

6、由不可抗拒自然因素引起的故障和损坏

产品返修：用户在使用过程中由于产品故障、损坏或其他异常现象，在寄回维修之前，请先致电米尔科技客服部，与工程师进行沟通以确认问题，避免故障判断错误造成不必要的运费损失及周期的耽误。

维修周期：收到返修产品后，我们将即日安排工程师进行检测，我们将在最短的时间内维修或更换并寄回。一般的故障维修周期为 3 个工作日（自我司收到物品之日起，不计运

输过程时间），由于特殊故障导致无法短期内维修的产品，我们会与用户另行沟通并确认维修周期。

维修费用：在免费保修期内的产品，由于产品质量问题引起的故障，不收任何维修费用；不属于免费保修范围内的故障或损坏，在检测确认问题后，我们将与客户沟通并确认维修费用，我们仅收取元器件材料费，不收取维修服务费；超过保修期限的产品，根据实际损坏的程度来确定收取的元器件材料费和维修服务费。

运输费用：产品正常保修时，用户寄回的运费由用户承担，维修后寄回给用户的费用由我司承担。非正常保修产品来回运费均由用户承担。

购买请联系：

电话：0755-25622735

传真：0755-25532724

邮箱：sales@myirtech.com

网站：www.myir-tech.com

技术支持请联系：

电话：0755-25622735

传真：0755-25532724

邮箱：support@myirtech.com

网站：www.myir-tech.com