

# Zynq7000上搭建嵌入式Linux

首先需要在存储设备上存在4个基本的文件

- 1.Linux文件系统（Linaro，BusyBox）
- 2.Linux kernel image
- 3.BOOT.bin 文件
- 4.编译好的device tree

## Formatting the SD Card（制作SD卡）

从SD卡上启动Linux需要在SD card上进行正确的分区，用特别的参数格式化SD card上的前两个分区。

第一个分区一定要是FAT文件系统，同时留下至少1GB的空间。

第二个分区一定要是ext4文件系统，至少留下3GB空间。第二个分区仅当使用Linaro文件系统时才需要。

Digilent要求将SD card按照上述要求格式化为两个分区，为了方便切换文件系统。

遵循如下4步，在Linux系统（PC）上格式化SD card

1)识别SD card设备节点。通过将SD card从Linux机器上移除SD card，然后运行lsblk.

先运行一遍lsblk，插入SD card后，再运行一遍lsblk。

lsblk命令用于列出所有可用块设备的信息，而且还能显示他们之间的依赖关系，但是它不会列出RAM盘的信息。

2)一些Linux版本可能会插入SD card的时候自动挂载到系统上。利用df命令查看SD card是否挂载。

如果挂载上了，确保在重新分区之前卸载这些自动安装在你硬盘分区上的分区。

利用命令umount卸载SD卡上的所有分区

3)一旦卸载所有的分区，你可以开始重新对SD card进行分区利用fdisk。打开SD card，使用fdisk命令：

a.首先使用p命令打印当前的SD card分区表

b.然后使用d命令删除任何存在的分区

c.最后使用n命令建立新的分区，要求建立的分区具有如下特性：分区1（从first cylinder开始占据1GB空间）和分区2（从下一个可用的cylinder开始占据剩余空间）

d.使用w命令将这些改动写到SD card的分区表中，执行完此命令后，也会自动退出fdisk。

4)建立文件系统（使用mkfs格式化你的分区）

将分区1格式化为FAT，label为ZED\_BOOT

将分区2格式化为EXT4，label为ROOT\_FS

## Linux文件系统

ZedBoard支持两种不同的Linux文件系统（BusyBox ramdisk和Linaro Ubuntu）。

BusyBox Ramdisk是一个非常小的文件系统，包括基本的功能，通过RAM运行。BusyBox是非持久性的，这意味着在断电后，不会保存你的任何改变。

Linaro file system拥有图形化的界面（通过HDNI端口显示），会保存一切操作。

在使用ZedBoard之前，要对文件系统进行选择。

### 1.使用BusyBox Ramdisk

已经建立好的BusyBox Ramdisk可以从[www.digilentinc.com/zedboard](http://www.digilentinc.com/zedboard)处获得，具体位置在ZedBoard\_Linux\_Design/sd\_image/ramdisk8M.image.gz.

将ramdisk8M.image.gz放置在FAT分区中。

也可以定制<http://wiki.xilinx.com/zynq-rootfs>。

## 2.使用Linaro file system

首先要从<http://releases.linaro.org/>处获得Linaro的不同版本

不包含图形界面的Linaro: <http://releases.linaro.org/12.09/ubuntu/precise-images/ubuntu-desktop/linaro-precise-ubuntu-desktop-20120923-436.tar.gz>.

一旦下载完成后, 可以通过以下6步将文件系统复制到ext4分区中

- 1)在/tmp下建立文件夹linaro, 将Linaro镜像复制到里面
- 2)利用tar命令解压镜像
- 3)插入SD card, 将任何自动挂载的分区卸载(如上面格式化SD card中的第二步)
- 4)将SD card挂载到/tmp/sd\_ext4
- 5)使用rsync将根文件系统复制到SD card中。
- 6)在移除SD card之前卸载, 确保所有的文件都已经同步。卸载/tmp/sd\_ext4需要几分钟, 一定要等到unmount返回。

## 建立Linux kernel

Digilent维护Linux源代码树(针对Digilent系统板)。这个库包含了custom drivers, attachable Pmods。在构建内核之前, 用户必须首先从Xilinx下载并安装ARM GNU工具。这些Xilinx工具的安装程序和说明可在: <http://wiki.xilinx.com/zynq-tools>.

- 1.在安装完ARM GNU工具链, 我建议打开.bashrc文件(home目录下)将下列的代码添加进去。

```
PATH=~:/CodeSourcery/Sourcery_CodeBench_Lite_for_Xilinx_GNU_Linux/bin:$PATH export
```

```
CROSS_COMPILE=arm-xilinx-linux-gnueabi-
```

- 2.使用git下载Linux kernel 源代码。

```
git clone https://github.com/Digilent/linux-digilent.git
```

- 3.在下载完成后, 切换到linux-digilent目录, 运行如下命令去配置ZedBoard

```
make ARCH=arm diligent_zed_defconfig, 这会将内核配置正确以便在板子上运行。
```

想要查看或修改默认配置使用如下命令:

```
make ARCH=arm menuconfig, 会打开一个图形化的接口来修改内核
```

- 4.运行如下命令来建立内核

```
make ARCH=arm
```

如果建立没有任何错误, 将会看到内核镜像文件在linux-digilent/arch/arm/boot/zImage, 将这个文件copy到SD card的FAT分区。

## 获得BOOT.BIN 文件

BOOT.BIN的文件是一个容器文件, 包含了各种Xilinx的特定文件, 用来初始化配置Zynq AP SoC, the programmable logic and processing systems。这个容器文件中也有u-boot, 第二阶段的bootloader(负责加载Linux)。

从sd\_image文件夹中复制预制好的boot.bin文件到SD卡的FAT分区。

如果配置Zynq部分感兴趣, 可以查看“ZedBoard Linux Hardware Design”项目, [www.digilentinc.com/zedboard](http://www.digilentinc.com/zedboard).

## 编译设备树(Compiling the Device Tree)

设备树是一种数据结构, 它将系统中存在的硬件描述为Linux内核。树将设备列为“节点”, 其中包含相应的驱动程序。内核通过对这些节点解析, 在启动过程中初始化每个节点的驱动程序。

Digilent Linux 目录中包含了一个默认的设备树, 与Linux硬件设计相一致。在linux-digilent/arch/arm/boot/digilent-zed.dts中找到默认的设备树。

用户必须更新设备树, 以反映在Xilinx平台工作室中对Linux硬件设计的任何更改。

除了描述硬件之外，设备树也是在启动时对内核进行配置。启动参数用来指示内核做许多不同的事情。最值得关注的是，这也是内核被告知要加载什么文件系统的地方。这意味着你需要修改diligent-zed.dts文件来说明正在使用哪个文件系统。

1)如果使用Linaro文件系统来启动ZedBoard，打开diligent-zed.dts文件，用下面的内容替换包含bootargs定义的代码行。

```
bootargs = "console=ttyPS0,115200 root=/dev/mmcblk0p2 rw earlyprintk rootfstype=ext4 rootwait  
devtmpfs.mount=0";
```

2)如果使用ramdisk来启动ZedBoard，使用下面的内容进行替换

```
bootargs = "console=ttyPS0,115200 root=/dev/ram rw initrd=0x800000,8M earlyprintk";
```

在正确配置启动参数后，使用dtc工具来建立device tree。

内核源码包含了dtc工具，在build kernel后可以在linux-diligent/scripts/dtc/dtc中找到。

切换到linux-diligent目录，并运行如下命令来建立device tree。需要修改输入文件，修改diligent-zed.dts文件位置。

编译好的设备树在linux-diligent/devicetree.dtb，将这个文件copy到SD card的FAT分区中。

## 引导SD card

上述步骤执行完后，利用下面8步来检测SD card

1.将SD card插入到ZedBoard

2.将jumpers按照如下设置

MIO 6: set to GND

MIO 5: set to 3V3

MIO 4: set to 3V3

MIO 3: set to GND

MIO 2: set to GND

VADJ Select: Set to 1V8

JP6: Shorted

JP2: Shorted

All other jumpers should be left unshorted

3.将调试工具的串口配置为如下

Baud: 115200

8 data bits

1 stop bit

no parity

4.连接您想在Linux使用的任何外设。如果使用Linaro的文件系统，我们建议您连接显示器的HDMI端口，一个USB集线器的USB OTG接口。然后您可以将鼠标和键盘附加到USB集线器上。

5.对ZedBoard供电

6.连接terminal emulator中合适的端口

7.等待boot process完成。

如果使用的是BusyBox file system，则会显示命令行终端

如果使用的是Linaro file system，这会显示图形界面

8.拥有了一个完整版的Linux 系统