

# 英贝德科技技术文档

## u-boot 介绍

深圳市英贝德科技公司制作  
二〇〇六年五月

深圳市英贝德科技公司版权所有 2006.5

## 目 录

第1节 bootloader 简介

第2节 u-boot 简介

第3节 u-boot 使用说明

第4节 u-boot 指令的使用

## 第一节 boot loader 简介

Boot loader 是系统加电后运行的第一段软件代码。由 PC 的体系结构可知，PC 机中的引导加载程序是由 BIOS 和位于硬盘 MBR 中的引导程序一起组成的。BIOS 在完成硬件检测和资源分配后，将硬盘 MBR 中的引导程序读到系统的 RAM 中，然后将控制权交给引导程序。引导程序的主要运行任务就是将内核映像从硬盘上读到 RAM 中，然后跳转到内核的入口点去运行，即开始启动操作系统。而在嵌入式系统中，通常并没有像 BIOS 那样的软硬件程序，因此整个系统的加载启动任务就完全由 Boot loader 来完成。系统在上电或复位时将从地址 0x00000000 处开始执行，而在这个地址处安排的通常就是系统的 Boot loader 程序。

每种不同的 CPU 体系结构都有不同的 Boot loader。除了依赖于 CPU 的体系结构外，Boot loader 实际上也依赖于具体嵌入式板级配置，比如板卡的硬件地址分配、RAM 芯片的类型、其他外设的类型等。对于两块不同的嵌入式板而言，即使是基于同一种 CPU 而构建的，如果其硬件资源和配置不一致，要想让运行在一块板子上的 Boot loader 程序也能运行在另一块板子上，则还需要作一些必要修改。

## 第 2 节 u-boot 简介

u-boot 是由德国 DENX 小组开发的交叉平台 Boot loader，其全称为 universal Boot Loader。u-boot 的开发目标是支持尽可能多的嵌入式处理器和嵌入式操作系统，它提供数百种嵌入式开发板和各种 CPU，包括 PowerPC、ARM、Xscale、MIPS、Coldfire、NIOS、Microblaze 和 X86 等，同时它除了支持 Linux 系统的引导外，还支持 NetBSD、VxWorks、QNX、RTEMS、ARTOS 等多种嵌入式操作系统的引导。u-boot 的用户接口类似于 Linux 的 shell 界面，通过串口连接以后，用户可以交互式的输入指令和察看结果。

u-boot 的启动界面如下所示：

```
boot 1.0 (Oct 16 2004 - 22:21:32)

Uncompressing image...

U-Boot 1.1.1 (Nov 16 2004 - 18:01:43)

U-Boot code: 21F00000 -> 21F16F2C BSS: -> 21F1B368
RAM Configuration:
Bank #0: 20000000 32 MB
Flash: 16 MB
In: serial
Out: serial
Err: serial
Hit any key to stop autoboot: 0
Uboot> _
```

u-boot 提供了多种强大的功能，而且其源码多是从 Linux 内核源码中修改而得，其功能可以实现高度裁减。u-boot 的主要功能有：

- u-boot 内集成了串口、以太网、FLASH、LCD、键盘等多种驱动程序。
- 上电后 u-boot 可以对 CPU 型号，SDRAM、FLASH 大小和故障进行自动检测。
- 支持串口和 TFTP 网口两种程序下载方式。
- 支持 Linux、VxWorks 等多种操作系统的加载与引导。

### 第 3 节 u-boot 使用说明

u-boot 的启动界面：

```
boot 1.0 (Oct 16 2004 - 22:21:32)

Uncompressing image...

U-Boot 1.1.1 (Nov 16 2004 - 18:01:43)

U-Boot code: 21F00000 -> 21F16F2C BSS: -> 21F1B368
RAM Configuration:
Bank #0: 20000000 32 MB
Flash: 16 MB
In: serial
Out: serial
Err: serial
Hit any key to stop autoboot: 0
Uboot> _
```

u-boot 的用户接口类似于 Linux 的 shell 界面，通过串口连接以后，用户可以交互式的输入命令和看到结果。启动时，首先检查系统配置。由上述结果可知，该系统的 RAM 配置为 32M，FLASH 配置为 16M，并且 C 库的 stdin、stdout 和 stderr 文件流连接到串口。然后开始进行命令行界面，以提示符“Uboot>”表示，在该命令行界面中用户可以输入操作命令。

u-boot 中所有操作都是通过其命令行输入命令完成的。这里讲述 u-boot 的命令行接口，注意由于 u-boot 的可配置性，当前配置并不一定支持所有命令。用户可使用“help”命令查看当前配置支持的所有命令。U-boot 中所有命令的数字都以十六进制格式输入，有些命令的处理结果依赖于 u-boot 的配置以及一些环境变量的设置。

在 u-boot 中输入命令并不一定输入全部命令名称，而是可省略后面的一些字符的，例如：“help”命令等同于“h”、“he”和“hel”，这要视实际情况而定。

u-boot 环境被保存于非易失性存储器（如 FLASH）的一段区域，当 u-boot 启动时被复制到 RAM 中。它用于保存配置系统的环境变量数据。U-boot 环境采用 CRC32 校验和保护。以下列举部分重要的环境变量，用户可以使用它们配置 u-boot。

## u-boot 环境变量说明

- **bootcmd**: 该变量定义一个命令字符串，当初始化倒计时没有被中断时，该字符串将被自动执行。该命令仅在 **bootdelay** 也被定义时执行。
- **bootdelay**: 系统复位后，在执行 **bootcmd** 变量的内容前，u-boot 将等待 **bootdelay** 秒倒数。在这段时间可通过按下任意键中断 **bootcmd** 的运行。如果不希望有任何延时，则设置该变量为 0 即可。
- **bootargs**: 该变量的内容被传递到 Linux 内核作为启动参数。
- **bootfile**: 使用 TFTP 加载缺省的映像名。
- **ipaddr**: IP 地址，TFTP 命令中设置目标板的 IP 地址。
- **serverip**: TFTP 服务器 IP 地址，TFTP 命令使用。

## 第 5 节 u-boot 指令的使用

u-boot 命令大部分支持扩展命令“.b”（以 8 位宽度执行），“.l”（以 32 位宽度执行），“.w”（以 16 位宽度执行）。它们可以直接跟在普通命令的后面。

- **flinfo**: 显示可用的 FLASH 的信息。使用事例如下:

```
Uboot>flinfo
10000000      10020000      10040000      10060000      10080000
101E0000      10200000      10220000      10240000      10260000
10280000      102A0000      102C0000      102E0000      10300000
10320000      10340000      10360000      10380000      103A0000
103C0000      103E0000      10400000      10420000      10440000
10460000      10480000      104A0000      104C0000      104E0000
10500000      10520000      10540000      10560000      10580000
105A0000      105C0000      105E0000      10600000      10620000
10640000      10660000      10680000      106A0000      106C0000
106E0000      10700000      10720000      10740000      10760000
10780000      107A0000      107C0000      107E0000      10800000
10820000      10840000      10860000      10880000      108A0000
108C0000      108E0000      10900000      10920000      10940000
10AA0000      10AC0000      10AE0000      10B00000      10B20000
10B40000      10B60000      10B80000      10BA0000      10BC0000
10BE0000      10C00000      10C20000      10C40000      10C60000
10C80000      10CA0000      10CC0000      10CE0000      10D00000
10D20000      10D40000      10D60000      10D80000      10DA0000
10DC0000      10DE0000      10E00000      10E20000      10E40000
10E60000      10E80000      10EA0000      10EC0000      10EE0000
10F00000      10F20000      10F40000      10F60000      10F80000
10FA0000      10FC0000      10FE0000
Uboot>
```

- protect: 使能或禁止 FLASH 保护功能。使用事例如下:

```
Uboot> protect off all (全片禁止 FLASH 保护功能)
Un-Protect Flash Bank # 1
Uboot>
```

- cp: 存储区复制 (支持扩展命令)。使用事例如下:

```
Uboot> cp.b 20000000 10000000 10504
Copy to Flash...\done
Uboot>
```

Cp 后面跟 3 个数据: 被复制地址, 复制到的地址, 复制文件的大小。“b”表示以 8 位宽度复制数据。

- erase: 擦除 FLASH 存储器。

在 u-boot 中, 扇区是一次擦除操作的最小区域, 擦除操作都是以扇区为单位的。在 u-boot 中, bank 的编号从 1 开始, 而扇区编号从 0 开始。该命令使用比较复杂, 最常用的用法就是传递待擦除区的开始地址和结束地址到命令中, 而且这两个地址必须是扇区的开始和结尾地址。使用事例如下:

```
Uboot> erase 10000000 1001ffff (擦除第 1 个扇区)
- Warning: 1 protected sectors will not be erased!
Erased 1 sectors
Uboot>
```

- loadb: 经由串口线加载二进制文件 (kermit 协议)。Loadb 后跟文件加载到的位置地址。使用事例如下:

```
Uboot> loadb 20000000
## Ready for binary (kermit) download to 0x20000000 at 115200
bps...
## Total Size      = 0x0000b776 = 46966 Bytes
## Start Addr     = 0x20000000
Uboot>
```

- tftpboot: 使用 TFTP 协议经由网络加载映像文件。使用事例如下:

```
Uboot> tftpboot
MAC: error during MII initialization
*** Warning: no boot file name; using '6600A8C0.img'
TFTP from server 192.168.0.125; our IP address is
192.168.0.102
Filename '6600A8C0.img'.
Load address: 0x20000000
Loading: #####
```

#### 其它命令简介:

- help: 显示 u-boot 可用命令。
- bootm: 从存储器启动应用程序。
- go: 开始某地址处的应用程序。
- saveenv: 保存环境变量到非易失性存储介质。
- setenv: 设置环境变量。
- run: 运行一个环境变量中的命令。
- mtest: 简单的存储区测试。
- mm: 存储区修改。
- mw: 内存填充。
- cmp: 存储区比较。
- md: 存储区显示。
- base: 设置或打印地址偏移。
- crc32: 校验和计算。

(更详细说明请参考 u-boot 官方技术文档)

深圳市英贝德科技公司  
2006-5-10