

第 3 章 U-BOOT 介绍—系统启动

内容概要

U-Boot 是一个非常复杂的东西，它也许体现了嵌入式系统的一个非常重要的特征：自己定制，他脱胎于 PC 机的 Linux，但是需要你关注更多，我们提供 U-BOOT1.1.1 源码，你也可以在网站上直接下载，或者找到更新的版本，本章主要介绍 U-boot 情况，版本、下载及应用，而对于 U-BOOT 的编译、修改请按照 MakeFile 文件在 Linux 下完成。

1. U-BOOT 能够干什么？

U-BOOT 和其他任何 BOOTLOADER 都是一样的，主要是对系统进行初始化、系统引导、FLASH 操作等功能。EBD9200 的 U-BOOT 主要是对板子的硬件进行初始化，包括：时钟、PLL、定时器和调试串口（Debug UART）等，具体请看源程序。有了 U-BOOT，我们可以在主机的超级终端上通过调试串口和目标机进行通信和设置。

2. 英贝德提供的资源...

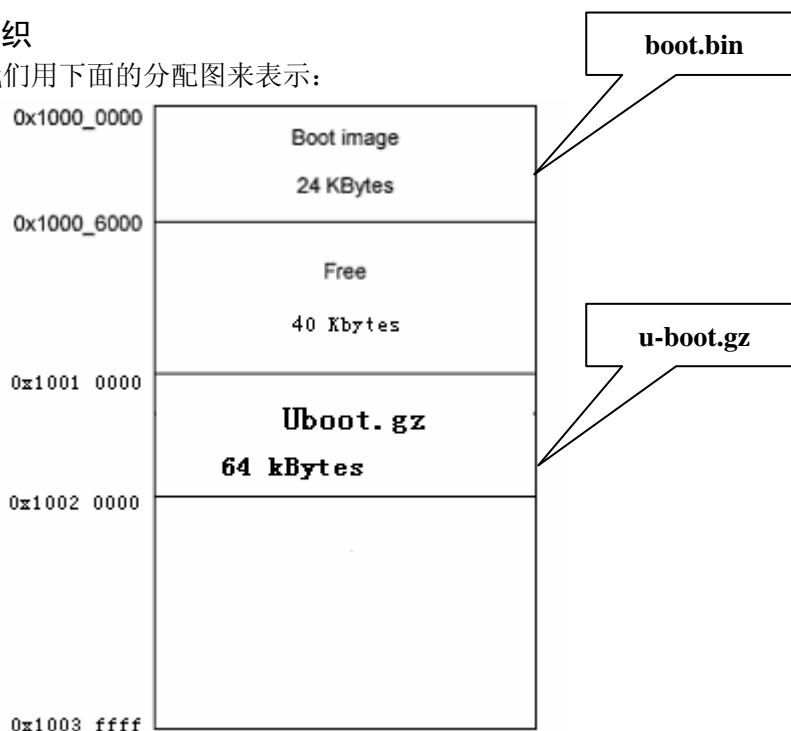
关于 U-BOOT，英贝德提供如下资源：

- U-BOOT、BOOT 和 LOADER 的源代码。
- 编译好的 loader.bin、u-boot.bin、boot.bin、u-boot.gz 文件。
- U-BOOT 使用文献。

3. 你应该更为详细的了解 U-BOOT

3. 1 U-BOOT 的组织

对于 U-BOOT，我们用下面的分配图来表示：



基本的 bootstrap 为 boot.bin，必须驻留在 AT91RM9200 的 Flash 地址处 NCS0，0x10000000。

3. 2 编译 U-BOOT

建议你如果没有特殊需要，不需要编译，但是如果你建造你的特殊要求的板子，在阅读本说明的基础上，还需要详细阅读 U-BOOT 的 readme、makefile 和原代码,编译 U-BOOT 最好在你的 PC 主机按装一个 Linux 系统，建议按装 Redhat 8.0、Redhat 9.0，用的人比较多，交流学习起来方便。

首先要在 PC 机上安装预先编译好的开发环境，后面第 5 章,有详细说明。

建议在根下建个目录 9200/bootldr，把 u-bootFor16M.tar.gz 或 u-bootFor8M.tar.gz 复制到 9200/bootldr 下，解压就成了。（建议以 root 用户登录）

```
$ cd /9200/bootldr          进入 ARM 目录
$ tar -zxvf u-bootFor16M.tar.gz 解开
```

3. 2. 1 编译 U-BOOT

在 Linux 系统下，你也可以用下面的命令编译，

```
> cd (U-Boot 1.1.1目录)      ; 进入目录
> make mrproper (只有初次编译时需要)
> make at91rm9200dk_config    ; 编译
> make all
> gzip -c u-boot.bin > u-boot.gz ; 压缩为gz文件
```

3. 2. 2 编译 boot.bin

```
> cd (AT91RM9200-Boot目录)
> make
```

3. 2. 3 编译 loader.bin

```
> cd (AT91RM9200-Loader目录)
> make
```

3. 3 下载 U-BOOT

直接下载 boot.bin 和 u-boot.gz,并且固化到 flash。

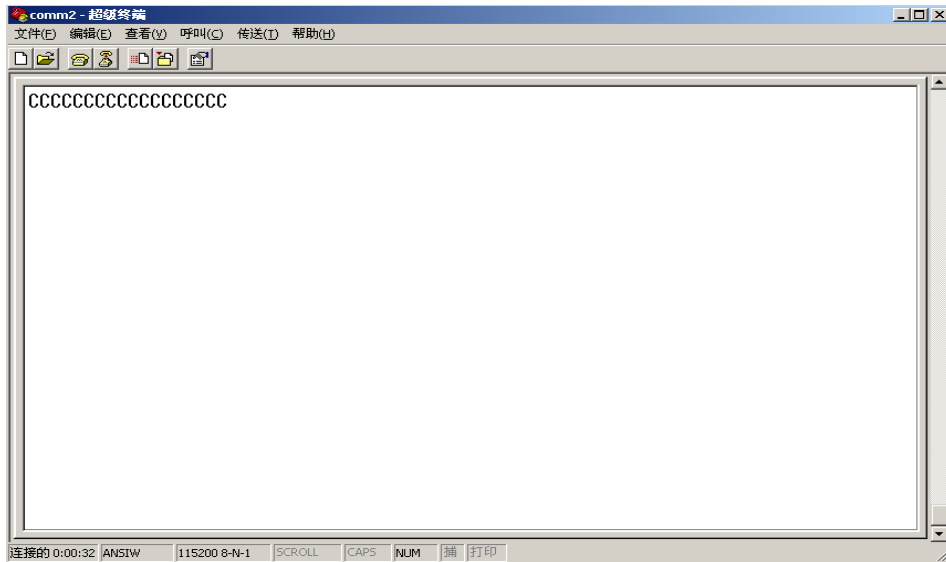
4. 对于更多的需求，你需要做什么？

当然 U-BOOT 还是比较复杂的，如果你只是一个 ARM 的应用者，我觉得足够了，如果你是对产品的开发细节非常关注的工程师，我想你可能会更去读源码，请下载 U-BOOT 的原代码，你才会了解更多，你如果更有兴趣，可以下载 U-BOOT 的各种版本，进行阅读和开发。

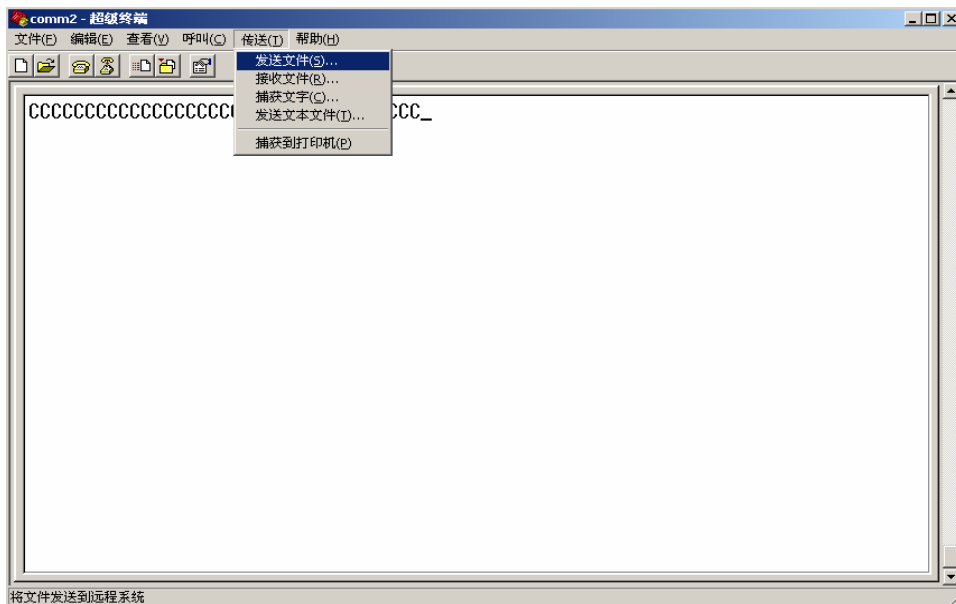
4. 对于如何把 Bootloader 固化到你的裸板上

你拿到板子时,我们已经预先为你在 flash(E28Fxxx)固化好 Bootloader,所以你的板子能够自动引导,如果在操作中无意破坏了 Bootloader,板子不能自动在超级终端打印信息,不能自动引导,怎么办呢?下面我们教你一步一步重新固化好 Bootloader。

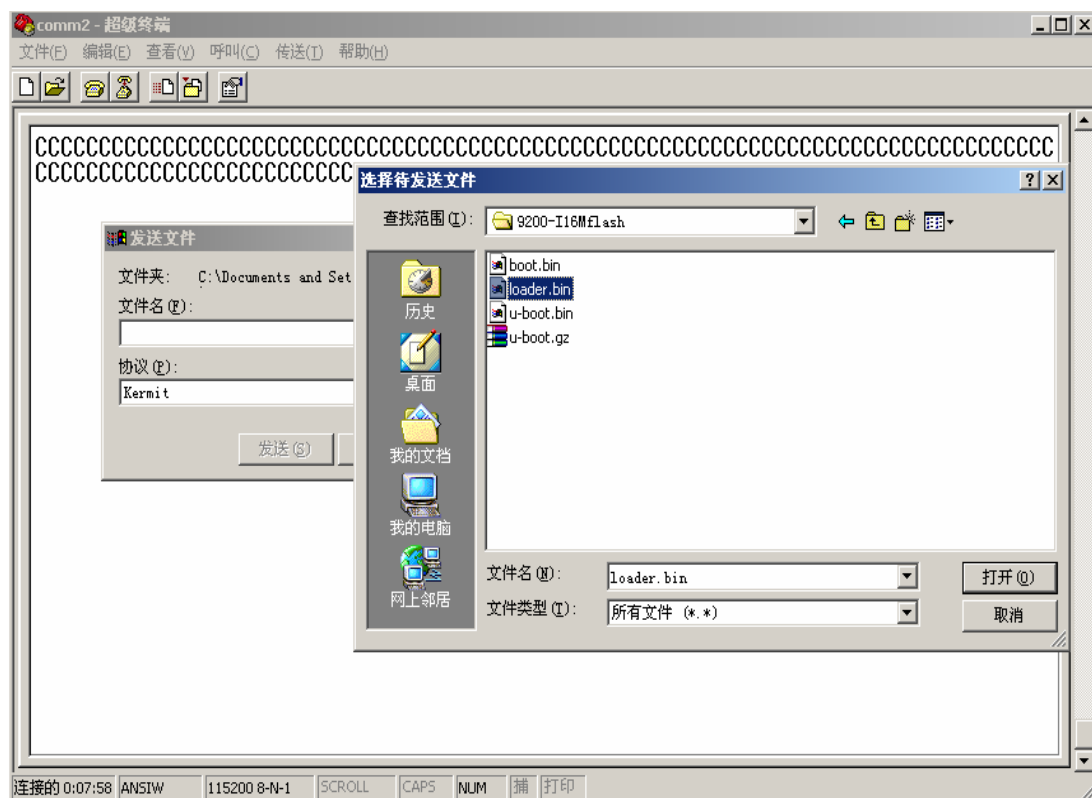
- 1) 把开发板左上角的串口 (EBGU) P1 和你的 PC 主机的串口连上,同时核心板上的跳线 J2 跳到 1、2 脚短接,重新上电后 CPU 便可从内部启动,在超级终端打印 CCCCCC 如下图所示:



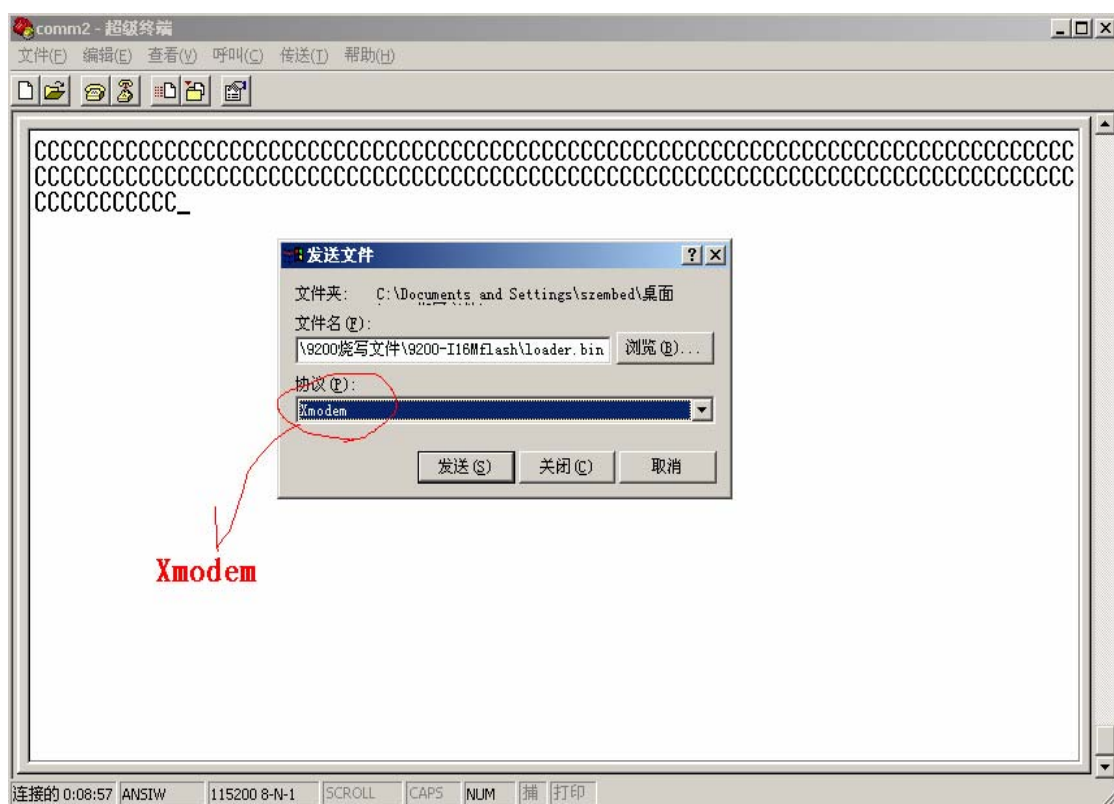
- 2) 点菜单 发送文件:



- 3) 找到光盘里目录(u-boot16bitFlash 启动)的对应的 Bootloader 文件 loader.bin 目录 (u-boot16bitFlash 启动) 有几个 Bootloader 映射子目录, EBD9200_I 16Mflash、EBD9200_I 8MFLASH, 分别和 Flash E28F128、E28F128 对应, 要发送与你开发板并行 Flash 对应的文件。



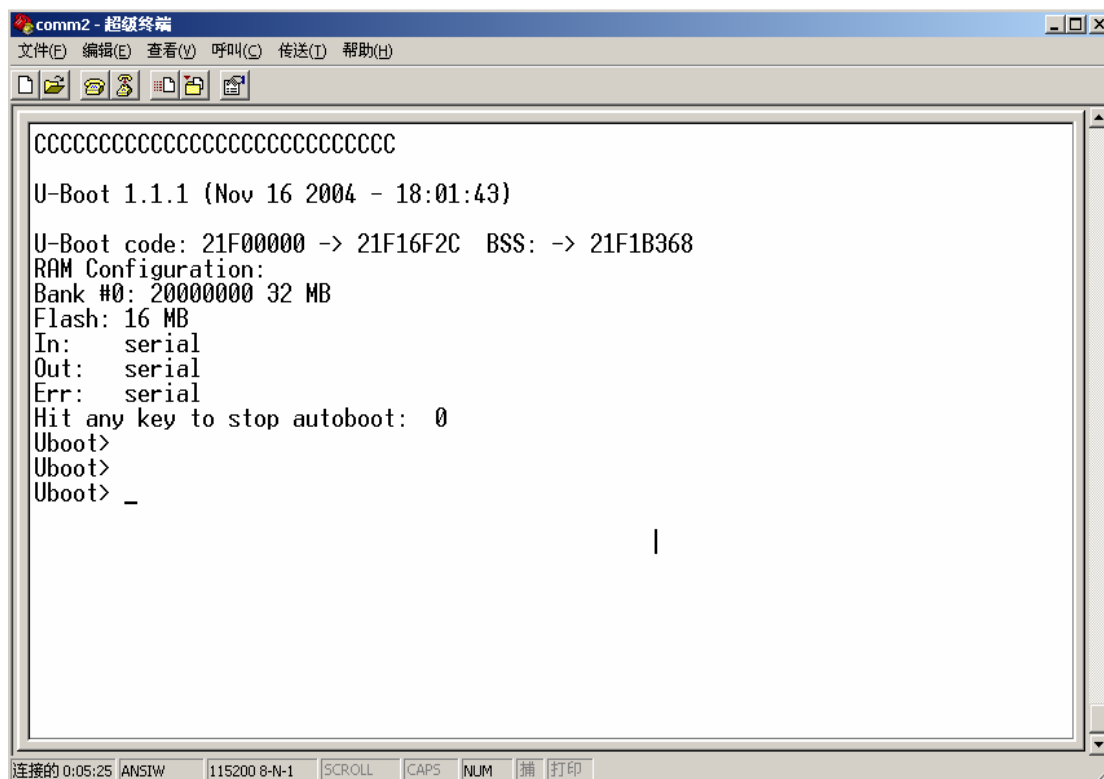
4) 发送文件:



5) 接着发送文件 u-boot.bin:



6) 发送 u-boot.bin 成功后出现熟悉信息如下:



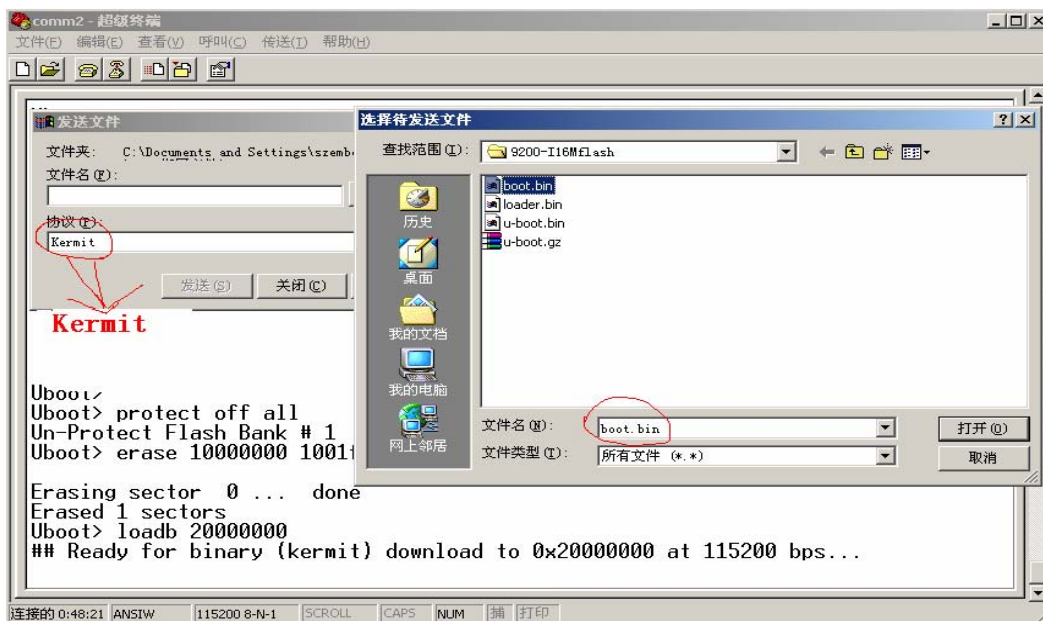
输入 **Uboot->protect off all** 回车 如下图所示:

连接的 0:15:45	ANSIW	115200 8-N-1	SCROLL	CAPS	NUM	捕	打印
-------------	-------	--------------	--------	------	-----	---	----

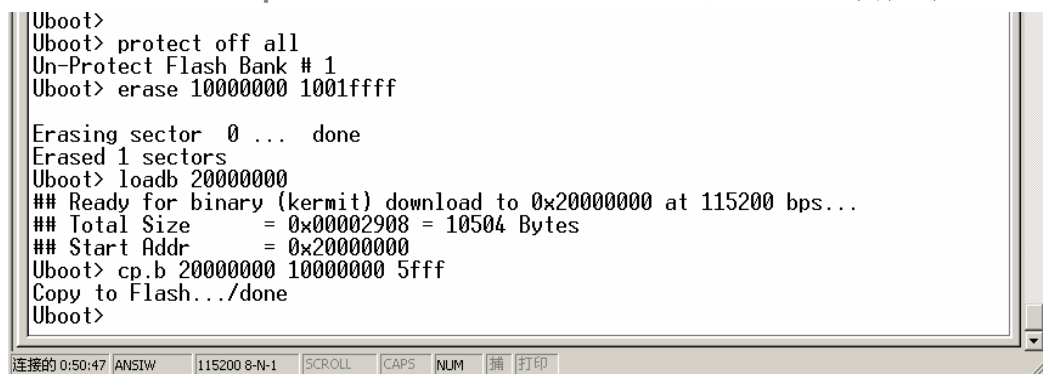
这样就把 flash 的第一个扇区刷除干净了。

连接的 0:47:06 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

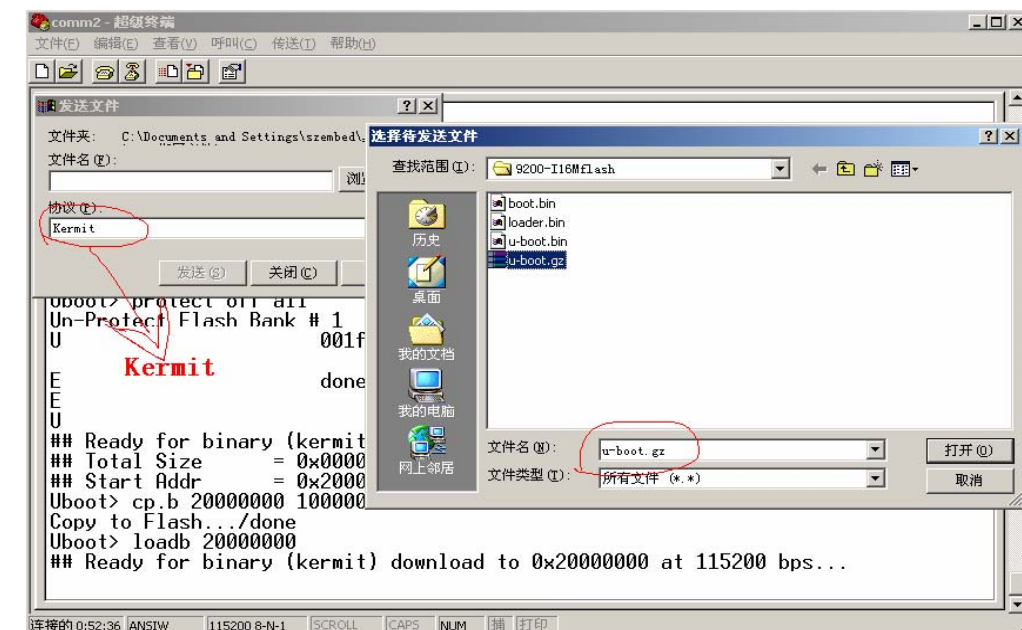
9) 接着 **Uboot>loadb 20000000** 发送文件 boot.bin:



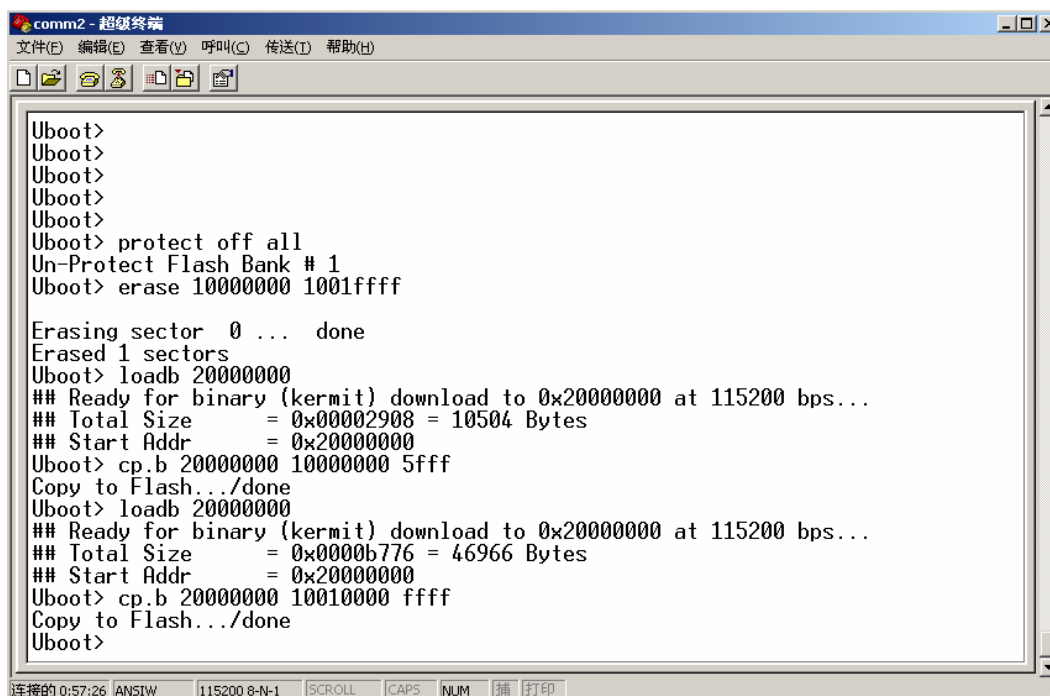
10) **Uboot >cp.b 20000000 10000000 5FFF** 把 boot.bin 固化到 flash.



11) 接着 **Uboot>loadb 20000000** 发送文件 u-boot. gz。



12) Uboot >cp.b 20000000 10010000 FFFF 把 u-boot. gz 固化到 flash。



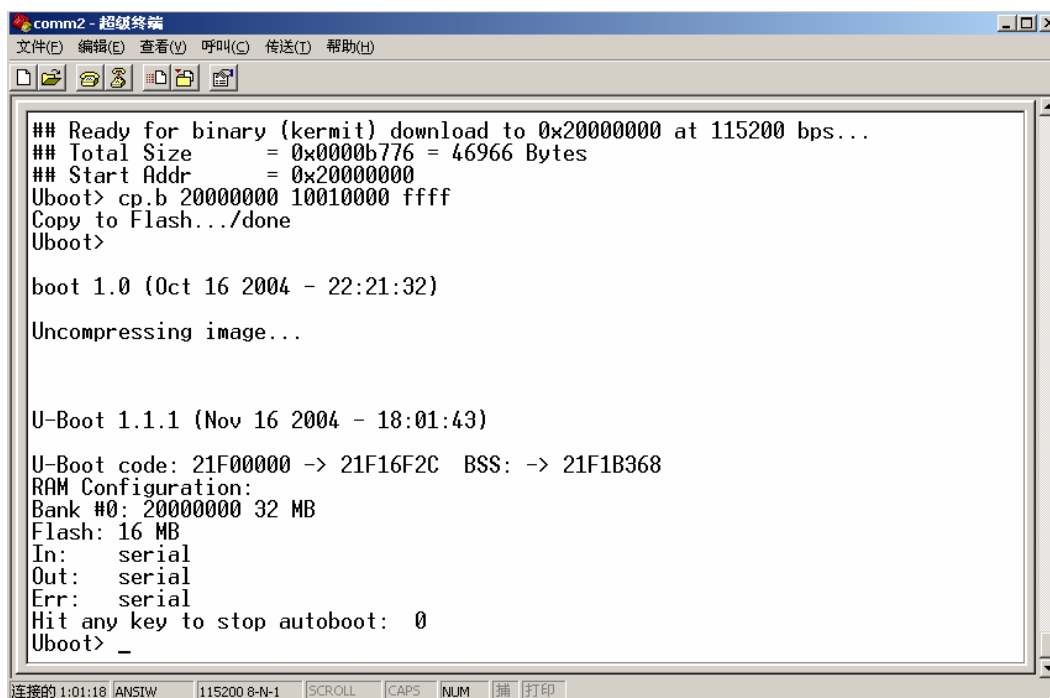
```

comm2 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)

Uboot>
Uboot>
Uboot>
Uboot>
Uboot> protect off all
Un-Protect Flash Bank # 1
Uboot> erase 10000000 1001ffff

Erasing sector 0 ... done
Erased 1 sectors
Uboot> loadb 20000000
## Ready for binary (kermit) download to 0x20000000 at 115200 bps...
## Total Size      = 0x00002908 = 10504 Bytes
## Start Addr      = 0x20000000
Uboot> cp.b 20000000 10000000 5fff
Copy to Flash.../done
Uboot> loadb 20000000
## Ready for binary (kermit) download to 0x20000000 at 115200 bps...
## Total Size      = 0x0000b776 = 46966 Bytes
## Start Addr      = 0x20000000
Uboot> cp.b 20000000 10010000 ffff
Copy to Flash.../done
Uboot>
  
```

13) 这样，Bootloader 已固化完毕，把跳线 J2 跳到 2、3 脚短接，重新上电如下：



```

comm2 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)

## Ready for binary (kermit) download to 0x20000000 at 115200 bps...
## Total Size      = 0x0000b776 = 46966 Bytes
## Start Addr      = 0x20000000
Uboot> cp.b 20000000 10010000 ffff
Copy to Flash.../done
Uboot>

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> _
  
```

现在 u-boot 的参数还没设置，欲了解 u-boot 的参数设置详见第五章。