ARM 385系统环境操作手册说明

文件版本：V1.2

发行日期：2017/8/31

拟 制 \_

审 核

批 准

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用 指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

更新版本

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **更改后版本号** | **拟制人/修改人** | **主要更改内容**  **（写要点即可）** | **拟制 / 修改日期** | **备注** |
| V1.0 |  | 新发布 | 2017/05/23 |  |
| V1.1 |  | 修改 | 2017/08/09 | 更新ntp时间 |
| V1.2 |  | 升级 | 2017/08/31 | 更新bmc固件方法，ntp时区修改 |
|  |  |  |  |  |
| 注1：每次更改文档内容时，需填写此表。  注2：首次发行时填写新生效。 | | | | |

目录

[1 概述 4](#_Toc494283059)

[1.1嵌入式开发环境 4](#_Toc494283060)

[1.2 linux开发环境 4](#_Toc494283061)

[1.3系统环境搭建及BMC操作 5](#_Toc494283062)

[1.3.1搭建tftp远程服务器 5](#_Toc494283063)

[1.3.2 BMC的IP及MAC地址修改 5](#_Toc494283064)

[1.3.3通过tftp升级BMC 8](#_Toc494283065)

[1.3.4 BMC时间查询及修改 9](#_Toc494283066)

[1.3.5 BMC控制节点上下电 9](#_Toc494283067)

[1.3.6 通过IPMI操作登录ARM节点 10](#_Toc494283068)

[2 网络安装系统以及更新内核 11](#_Toc494283069)

[2.1 U-boot 简介 11](#_Toc494283070)

[2.1.1 ARM板U-boot更新 12](#_Toc494283071)

[2.2嵌入式操作系统网络安装 13](#_Toc494283072)

[2.2.1挂载远程服务器系统 13](#_Toc494283073)

[2.2.2系统分区及拷贝内核及文件系统 14](#_Toc494283074)

[2.2.3设置ARM板默认启动方式 16](#_Toc494283075)

[2.2.4 ARM节点NTP时间同步 16](#_Toc494283076)

[3 通过串口安装ARM385操作系统 16](#_Toc494283077)

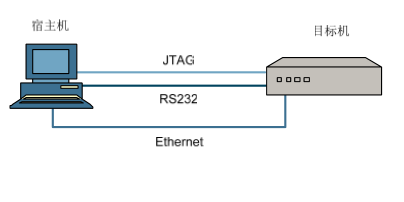
[4 附录Ubuntu14.04系统更新源 17](#_Toc494283078)

# 概述

本文档介绍ARM 385 Linux环境的搭建、文件系统以及内核和根文件系统的网络烧写，本文档主要是让客户更快地了解ARM 385系统环境,配置基本的嵌入式系统。

1.1嵌入式开发环境

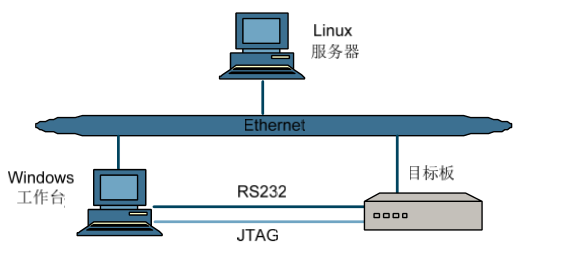
由于嵌入式单板的资源有限，不能在单板上运行开发和调试工具，通常需要交叉编译调试的方式进行开发和调试，即“宿主机＋目标机（评估板）”的形式。宿主机和目标机一般采用串口连接，也可同时通过网口或者JTAG连接，如图所示。



宿主机和目标机的处理器一般不相同，宿主机需要建立适合于目标机的交叉编译环境，程序在宿主机上经过“编译－连接－定位”得到可执行文件，通过一定的方法将可执行文件烧写到目标机中，然后在目标机上运行。目标机上的 Bootloader启动后，目标机中的操作信息通过串口或者网口输出到宿主机上显示，在宿主机上的控制台中输入命令，可以控制目标机。

## 1.2 linux开发环境

Linux开发环境通常包括 Linux服务器、Windows 工作台和 DMEB（目标板），三者同处于一个网络中，如图所示。



涉及内核镜像编译以及硬件树编译，UBOOT及BMC编译

1.3系统环境搭建及BMC操作

### **1.3.1搭建tftp远程服务器**

这里我们使用windows平台里面安装虚拟机，搭建安装arm板的系统环境，以及内核替换的环境，BMC管理环境。安装好虚拟机，导入虚拟机环境到windows，安装Xshell等远程终端工具，通过Xshell登录虚拟机（这里虚拟机的ip是192.168.2.225）：

ssh 192.168.2.225

输入用户名和密码root/123456进入虚拟机中的linux系统。

### 1.3.2 BMC的IP及MAC地址修改

目前的固件里面BMC的IP将会默认是192.168.1.214

MAC地址默认是6a:7b:8c:9d:ae:b7

注意：需要更改BMC默认的ip地址和MAC地址，不然同一网络里面会出现网络冲突的问题。

一、修改MAC地址

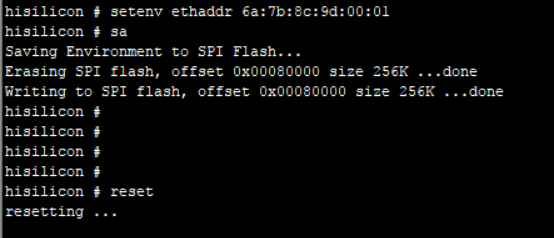
目前只能通过usb转232的串口插针登陆到BMC,在上电瞬间中断自启动进入的uboot的环境下修改。下图是插针的位置：



在uboot的环境下输入

setenv ethaddr 6a:7b:8c:00:00:11

saveenv



重启后可以用print打印看到mac已经修改的信息。

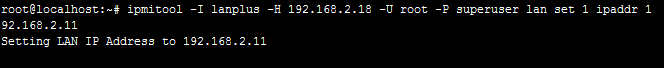
二、修改IP地址及netmask：

可以直接通过串口或者ssh登陆到bmc系统里面vi /etc/network/interfaces 修改ip，修改完使用命令，reboot -f 命令重启设备，

或者用ipmi工具修改：

ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser lan set 1 ipaddr 192.168.2.11

ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser lan set 1 netmask 255.255.255.0

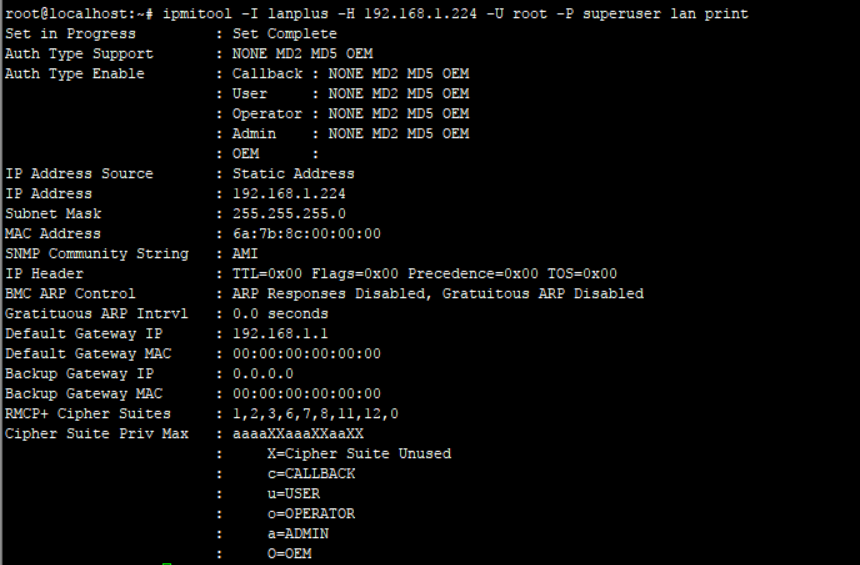


这一步后网络BMC的网络地址将更改，用ctrl+c退出修改（这个地方会需要等待的时间长一点），

可用：

ipmitool -I lanplus -H 192.168.1.224 -U root -P superuser lan print

查看网络更改后的情况



ipmitool -I lanplus -H 192.168.1.224 -U root -P superuser bmc reset cold

冷重启BMC

ipmitool -I lanplus -H 192.168.1.224 -U root -P superuser bmc reset warm

热重启BMC

当不记得ip地址的时候通过按reset键将可以重置IP位192.168.1.118

1.3.3通过tftp升级BMC

1、将升级的固件包拷贝的tftproot的服务器里面，用usb转串口线材插到bmc的插针上，上电后中断系统自动加载进程

首先对uboot的环境的网络进行配置:

setenv ipaddr 192.168.1.111

setenv serverip 192.168.1.18（IP是tftp服务器的ip）

然后进行

hisilicon#sf probe 0 /\*对 SPI Flash 进行初始化设置\*/

hisilicon#sf erase 0x500000 0xb00000 /\*首先擦除 FLASH 的文件系统分区，FLASH 分区信息参见后面 的启动参数设置\*/

hisilicon#tftp 0x82000000 jiffs2-root.img /\*将 jffs2 文件下载到 0x82000000\*/ 正常的下载过程超级终端中显示的信息如下所示：

MAC: 00-10-85-18-01-30 TFTP from server 10.85.180.211; our IP address is 10.85.180.130 Download Filename 'rootfs-FULL\_REL.jffs2'. Download to address: 0x80800000 Downloading: %# [ Connected ] ################################

[ 1.000 MB] ################################

[ 2.000 MB] ################################

[ 3.000 MB] ################################

[ 4.000 MB] ################################

[ 5.000 MB] ################################

[ 6.000 MB] ################## 6.591 MB download ok. Bytes transferred = 6897136 (693df0 hex)

hisilicon#sf write 0x82000000 0x500000 0xb00000 由于 SPI Flash 的写操作速度较慢，如果下载的文件较大，则需要花费一定的时间，等 到重新回到“hisilicon#”的提示符，表示更新完成。

1.3.4 BMC时间查询及修改：

通过ssh登陆到bmc的IP使用密码 登陆进去，将时间设好：

date -s "2017-08-09 10:00:00"

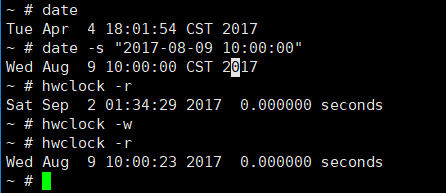
然后将时间同步到cmos时间里面去

hwclock -w

这样主板将插拔bmc的板子

hwclock -r

查询cmos的时间是否和系统时间一致



这里是NTP服务器的关键点，所有的arm节点都是同步BMC的时间，所以这个时间设置好，BMC板上必须安装好纽扣电池。

### 1.3.5 BMC控制节点上下电

使用BMC控制整机或者节点开机|关机|重启：

目前定义为，0号为整机1-12为各个节点，此步骤是选取节点后再进行操作：

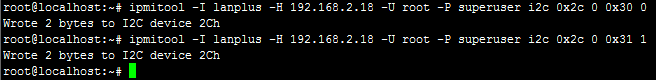
ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser i2c 0x2c 0 0x30 0

ipmitool -I lanplus -H 192.168.1.214 -U root -P superuser i2c 0x2c 0 0x30 0

选取整机（0x30后面的0代表整机）

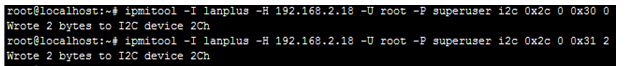
ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser i2c 0x2c 0 0x31 1

操作整机开机（0x31后面的1表示开机）



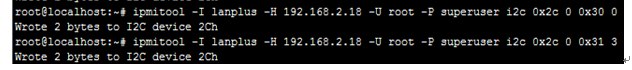
操作整机关机（0x31后面的2表示关机）

ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser i2c 0x2c 0 0x31 1



操作整机重启（0x31后面的3表示重启）

ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser i2c 0x2c 0 0x31 3



### 1.3.6 通过IPMI操作登录ARM节点

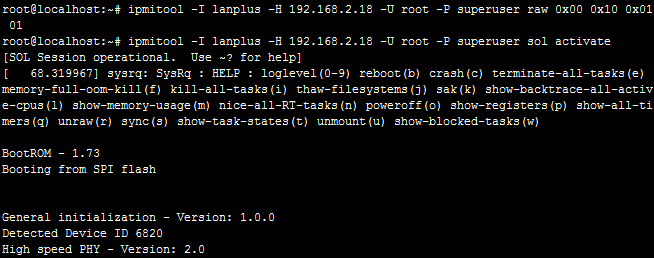
使用BMC中的ipmitool登录节点：

ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser raw 0x00 0x10 0x01

(0x01表示01号ARM的第一个节点，依次可以输入到0x0c)

ipmitool -I lanplus -H 192.168.2.18 -U root -P superuser sol activate

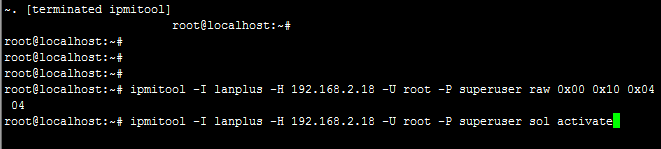
激活串口登陆



可以看到节点正常的信息打印。

切换节点使用的是输入~.，将会把操作从节点退回虚拟机，再去切换到挂入别的节点。

1-12是与节点从上至下，从左至右的顺序一一对应的。（旧版本的设备是0-11）



# 网络安装系统以及更新内核

## 2.1 U-boot 简介

U-boot 是在Bootloader操作系统内核运行之前运行的一段小程序。通过这段小程序，可以实现 以下功能：初始化硬件设备。建立内存空间的映射图。使系统的软硬件环境处于一个合适的状态，为最终调用操作系统内核准备好正确 的环境。 U-boot 除了作为一个 Bootloader 外，还是一个烧写器。在 U-boot下，可以通过串口、网口下载 Linux 内核或者应用程序到内存或 Flash中。

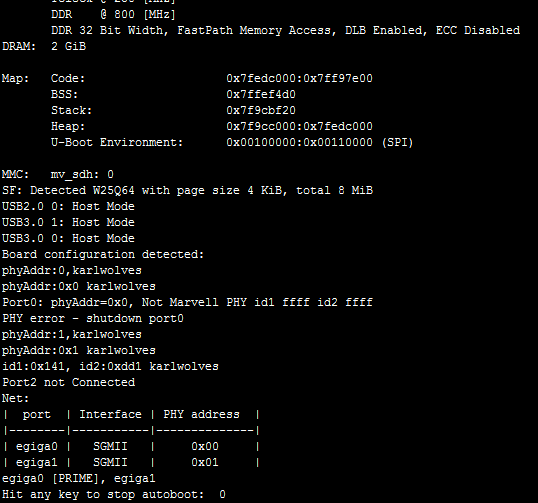
### 2.1.1 ARM板U-boot更新

ARM板的U-boot更新烧写有两种办法：

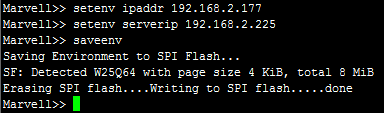
1.使用烧录器将文件烧录进flash中。

2.通过网口下载更新烧录进flash。这里介绍使用网络更新U-boot方式。

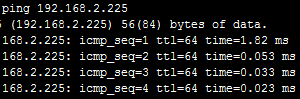
首先我们拿到手的arm板将会是已经烧好了启动环境的U-boot，启动会如下：



此时按键盘中断自启动，进入uboot环境，设好ip以及服务器ip保存好



Ping tftp 服务器的ip测试网络是否正常。



烧录过程：

sf probe 0

sf erase 0x0 0x100000

tftp 0x2000000 u-boot-a38x-2015\_T1.0-spi.bin

sf write 0x2000000 0x0 0x100000

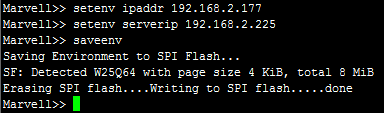
reset后从启动过程查看是否更新到新的版本。

## 2.2嵌入式操作系统网络安装

### 2.2.1挂载远程服务器系统

首先我们拿到手的arm板将会是已经烧好了启动环境的U-boot，启动后

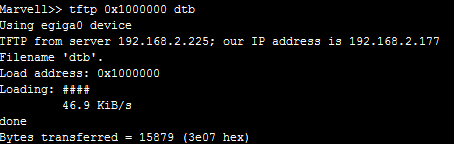
设好ip以及服务器ip保存好



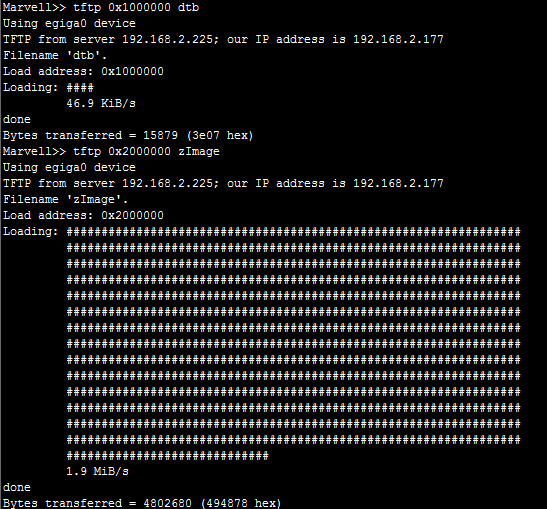
可以用print命令查看是否设置正确。

(先得将dtb 和zImage拷贝到远程服务器的root/tftproot和root/rootfs/home下)

下载dtb文件（硬件树）



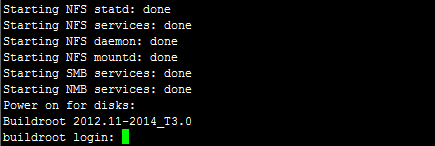
下载zImage文件（内核文件）



设置uboot启动引导

setenv bootargs console=ttyS0,115200 root=/dev/nfs rw nfsroot=192.168.2.225:/root/rootfs ip=192.168.2.177:192.168.2.225:192.168.2.1:255.255.255.0:Armada38x:eth0:none init=/linuxrc

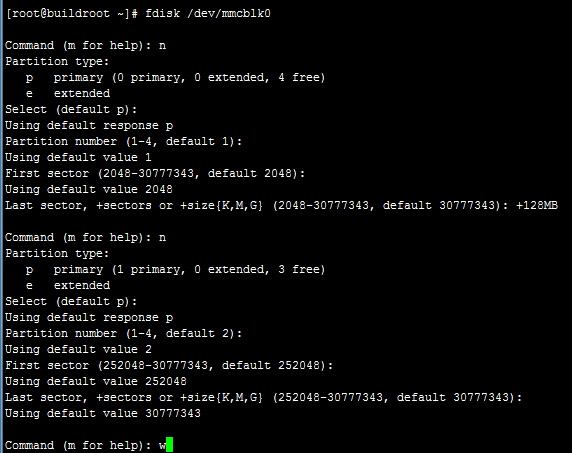
bootz 0x2000000 - 0x1000000



使用root/123456登录

### 2.2.2系统分区及拷贝内核及文件系统

格式化emmc：



制作文件格式：

mkfs.ext2 /dev/mmcblk0p1

mkfs.ext4 /dev/mmcblk0p2

fsck /dev/mmcblk0p1

fsck /dev/mmcblk0p2

挂载下载内核及文件系统：

mount /dev/mmcblk0p1 /mnt

cd /home

cp zImage dtb /mnt

这个地方的拷贝命令就是将内核拷贝到第一个分区，升级内核就是通过修改整个dtb zImage文件，就能替换嵌入式系统的内核。(具体的文件请咨询正式发布的版本)

umount /mnt

mount /dev/mmcblk0p2 /mnt

tar -zxvf ubuntu-12.04-mars200\_v1.0\_20160310.tgz -C /mnt

做完后就reboot

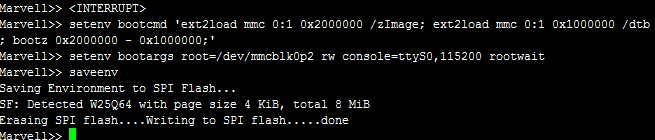
### 2.2.3设置ARM板默认启动方式

在uboot的环境下输入：

setenv bootcmd 'ext2load mmc 0:1 0x2000000 /zImage; ext2load mmc 0:1 0x1000000 /dtb; bootz 0x2000000 - 0x1000000;'

setenv bootargs root=/dev/mmcblk0p2 rw console=ttyS0,115200 rootwait //设置从emmc启动

saveenv //保存网络设置



reset

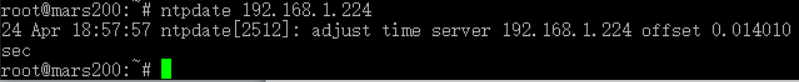
后将自动进入到emmc已经设置好的嵌入式linux系统。

### 2.2.4 ARM节点NTP时间同步

安装好系统后使用ipmi的命令将BMC的时间设定好后，所有的节点可以通过命令：

ntpdate 192.168.1.224

这里的IP是对应的BMC的IP地址，请确认BMC和节点的网络在同一网段且可以ping通。



通过date可以观察到时间已经同步好：



# 通过串口安装ARM385操作系统

通过串口刷入操作系统到emmc，此方法还是需要开发环境搭建好，用U盘或者硬盘制作成母盘，将系统备份出来，通过母盘使用串口将系统拷贝到emmc中去。具体待更新

# 附录Ubuntu14.04系统更新源

在新的版本的系统上安装ubuntu14.04的系统后，存在没有ssh和源无法更新软件的情况。

这里修改源：

deb http://ports.ubuntu.com/ubuntu-ports/ trusty main

deb-src http://ports.ubuntu.com/ubuntu-ports/ trusty main

deb http://ports.ubuntu.com/ubuntu-ports/ trusty-updates main

deb-src http://ports.ubuntu.com/ubuntu-ports/ trusty-updates main

deb http://ports.ubuntu.com/ubuntu-ports/ trusty-security main

deb-src http://ports.ubuntu.com/ubuntu-ports/ trusty-security main

然后再修改vi /etc/resolv.conf

添加

nameserver 8.8.8.8

nameserver 202.96.128.166

通过源更新安装ssh

apt-get update

apt-get install openssh-server

有时会出现无法写入的问题

mount -o remount,rw /

然后确认sshserver是否启动了：

ps -e |grep ssh

如果看到sshd那说明ssh-server已经启动了。

修改SSH的配置文件/etc/ssh/sshd\_config

修改 vim /etc/ssh/sshd\_config

找到# Authentication:

LoginGraceTime 120

PermitRootLogin without passwd

StrictModes yes

改成

# Authentication:

LoginGraceTime 120

PermitRootLogin yes

StrictModes yes

然后重启SSH服务（/etc/init.d/ssh restart）即可。