神威平台
维护使用说明

(ast2400 版)

2017 年 11 月

成都申威科技有限责任公司

成都市华府大道四段电子科大科技园 D22 栋
Building D22, National University Science and technology park,
Section 4, Huafu Avenue, Chengdu

免责声明

本文档仅提供阶段性信息，所含内容可根据产品的实际情况随时更新，恕不另行通知。如因文档使用不当造成的直接或间接损失，本公司不承担任何责任。

成都申威科技有限责任公司

Chengdu Sunway Technology Corporation Limited

地址：成都市华府大道四段电子科大科技园 D22 栋

Building D22, National University Science and technology park,

Section 4, Huafu Avenue, Chengdu

Mail：sales@swcpu.cn

Tel ：028-68769016

Fax：028-68769019

阅读指南

《神威平台维护使用说明》主要描述了神威调试平台的调试工具介绍、维护环境安装、维护命令相关说明、维护脚本及工具使用说明以及平台相关特性介绍等内容。

文档修订

文档更新记录	文档名	神威平台维护使用说明 (ast2400 版)
	版本号	V1.2
	创建人	研发部
	创建日期	2017-11-8

版本更新

版本号	更新内容	更新日期
V1.0	形成说明手册 2016 版	2017-11-10
V1.1	添加命令对应的界面操作	2017-05-10
V1.2	添加界面操作说明	2017-07-20

技术支持

可通过邮箱或问题反馈网站向我司提交产品使用的问题，并获取技术支持。

售后服务邮箱：sales@swcpu.cn

问题反馈网址：http://www.swcpu.cn/

目　　录

1 调试工具

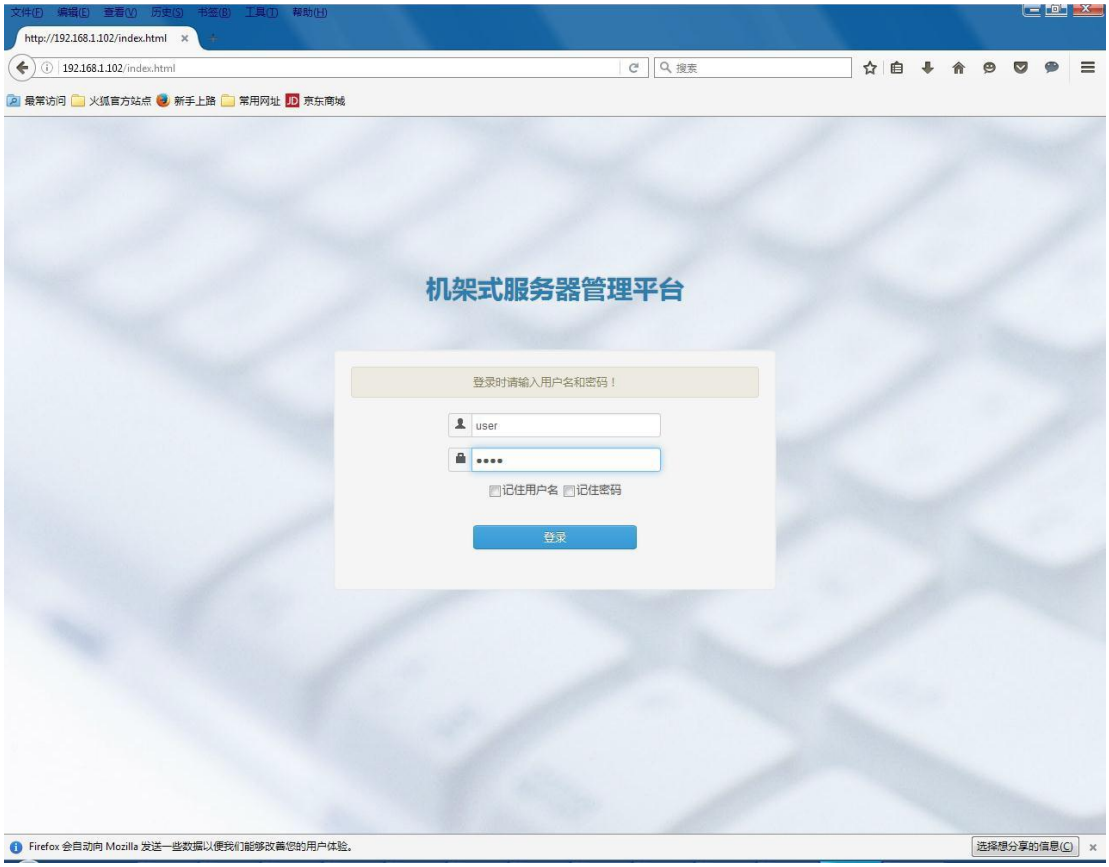
1.1 维护界面操作说明

1.1.1 客户端要求

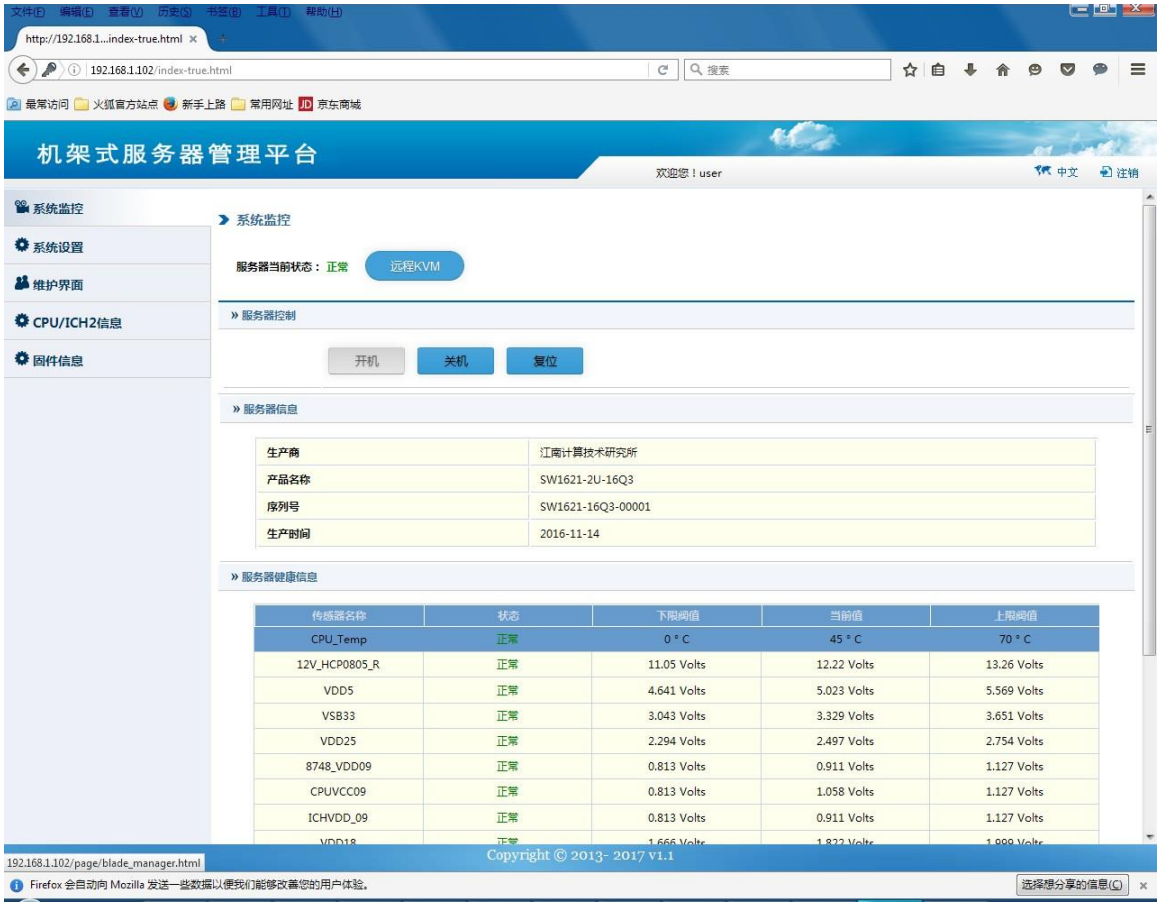
1、 客户端主机与维护机通过网络互联
2、 推荐使用火狐或 Google 浏览器

1.1.2 登录界面

1、 打开浏览器，在地址栏中输入维护机的 IP。
2、进入登录界面，用户名：user，密码：user，点击登录按钮，进入维护操作界面。



1.1.3 系统监控



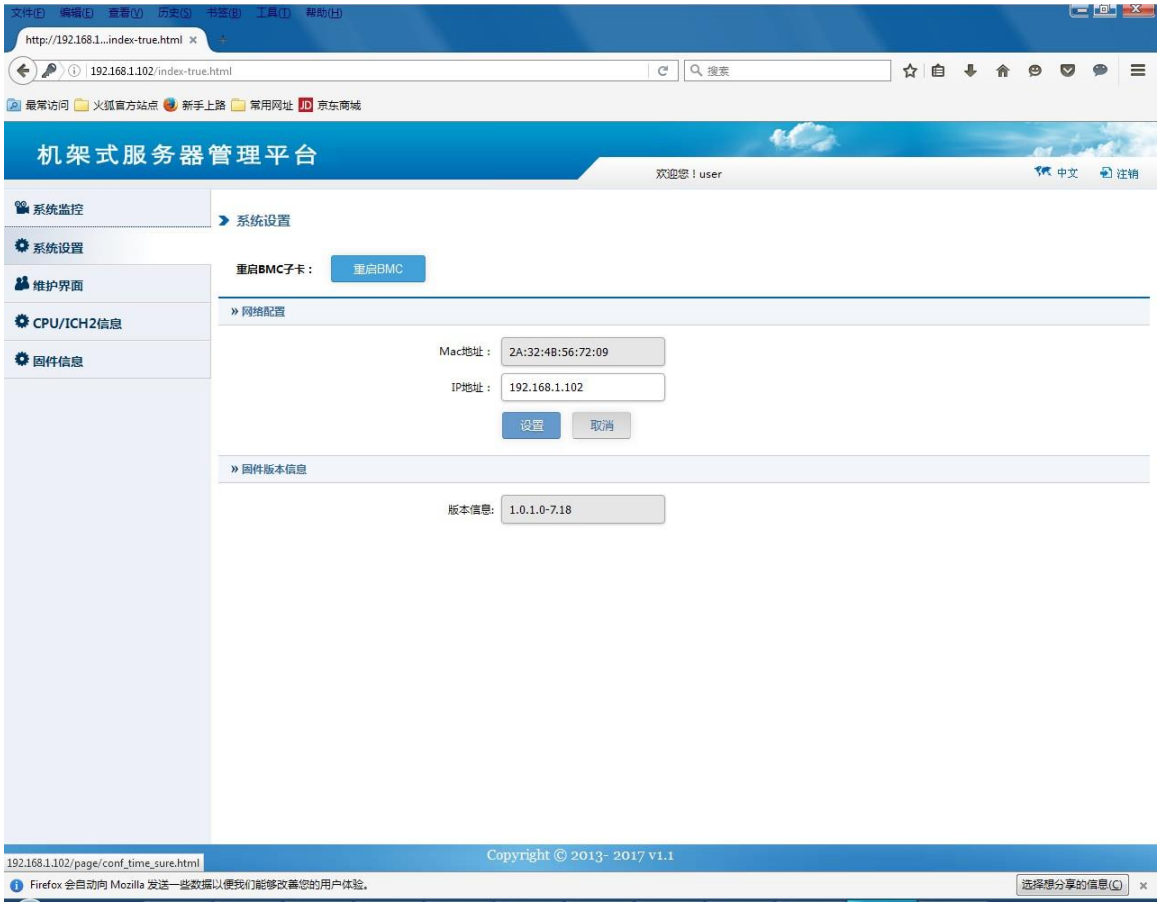
功能说明：

1、 服务器开机、关机和复位功能。

2、 远程 KVM 功能。

3、 服务器健康信息：CPU 温度、主要的电压、风扇监控等。

1.1.4 系统设置



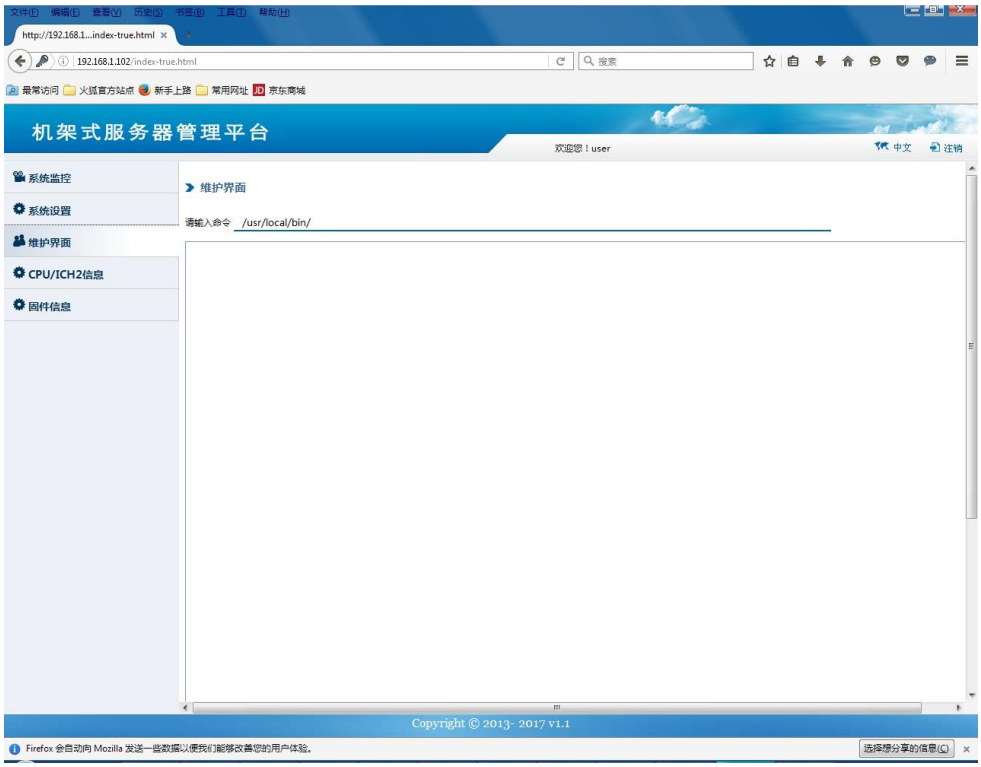
功能说明：

1、 重启 BMC 子卡

2、设置 BMC 子卡的 IP 地址，设置后需要等待一段时间，方可使用新设置的IP 登录。

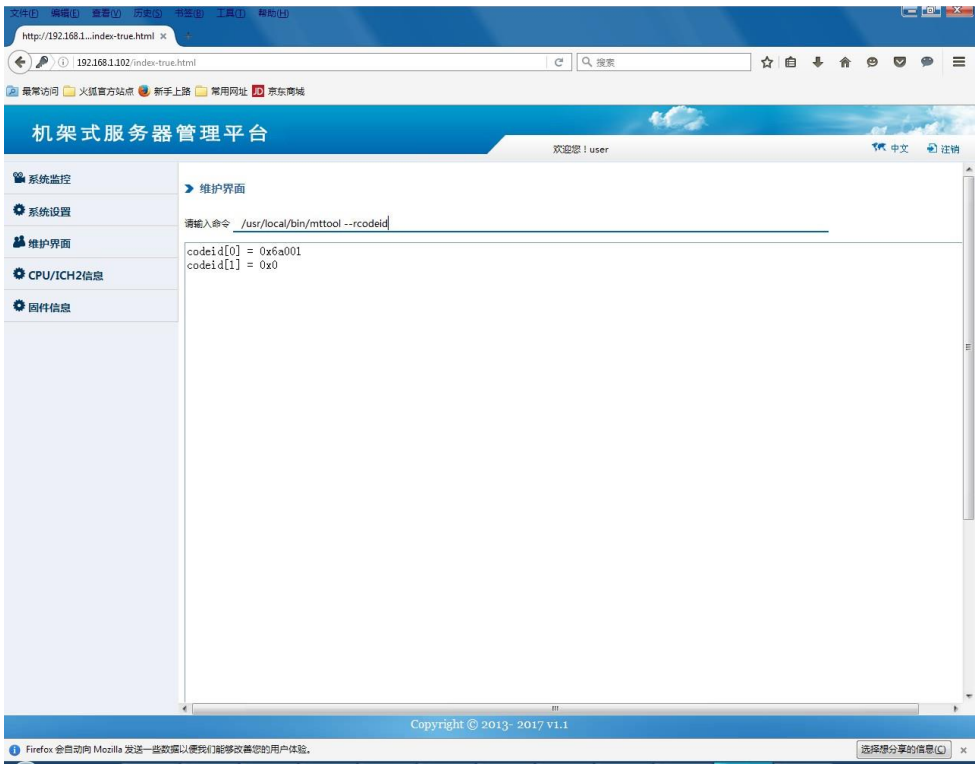
3、 查看 BMC 的版本号，目前版本为 1.0.1.0-7.24。

1.1.5 维护界面



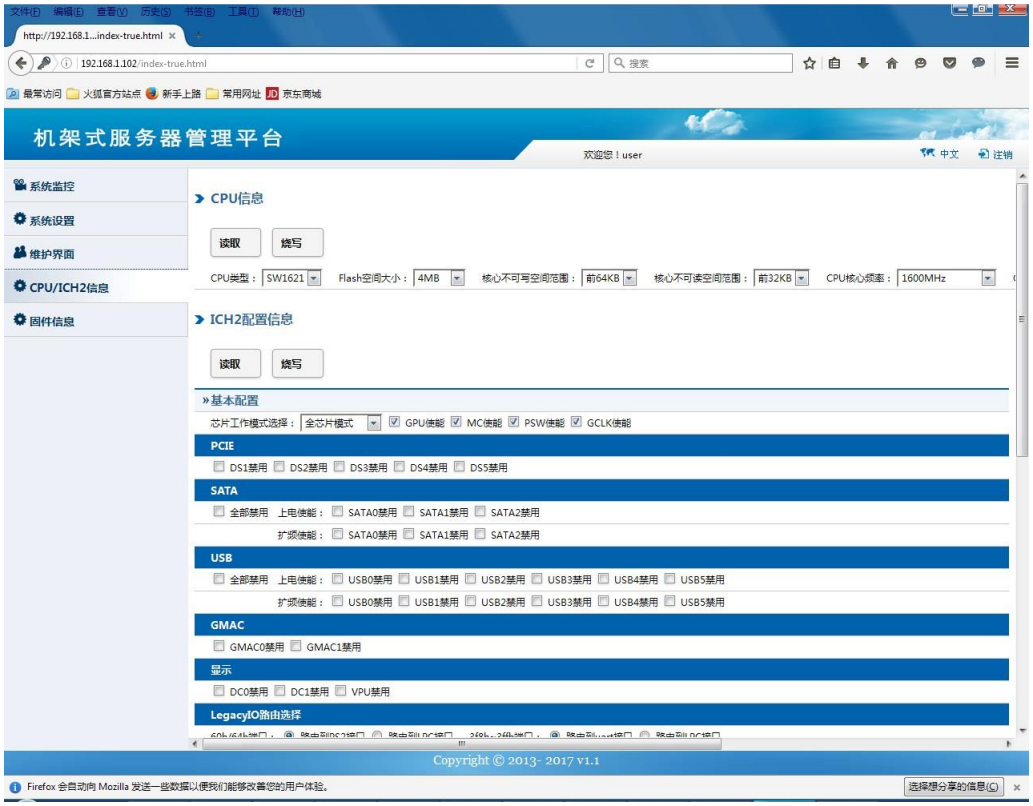
1、 读取 CPU 的 codeid，如下图所示，则说明 CPU 的 JTAG 通路正常。

注：CPU 需单独连接到维护卡的 JTAG 控制器上，不支持与其他设备的 JTAG 连接。

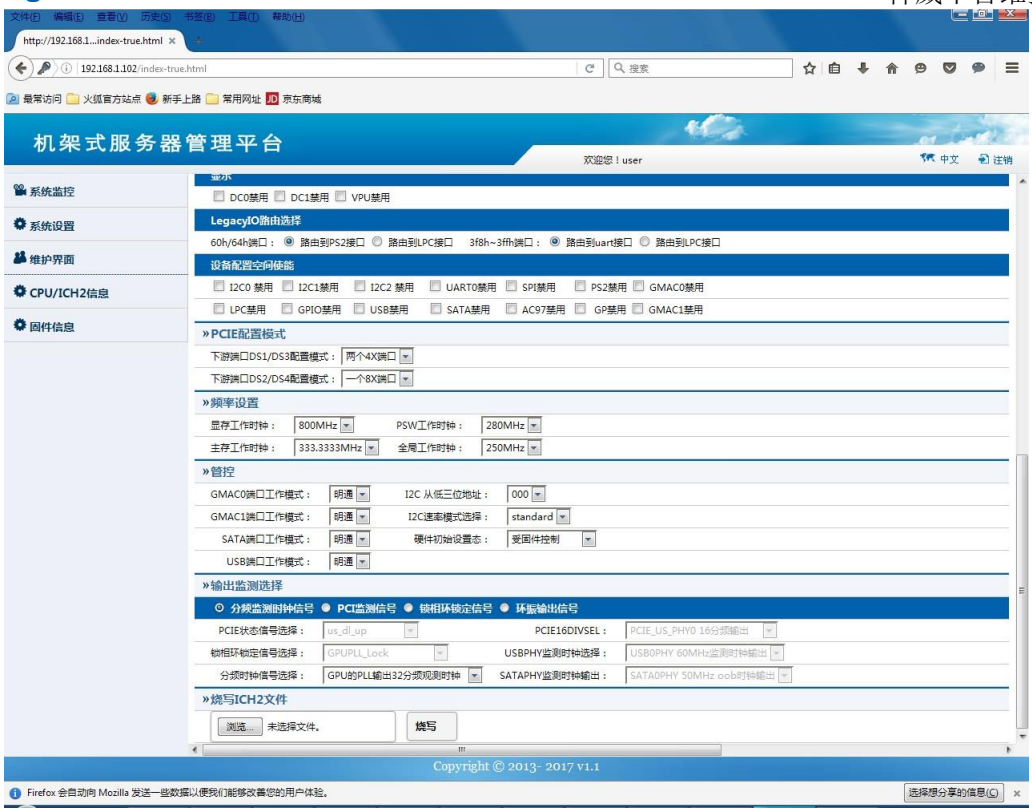


注：其它命令在维护命令中说明

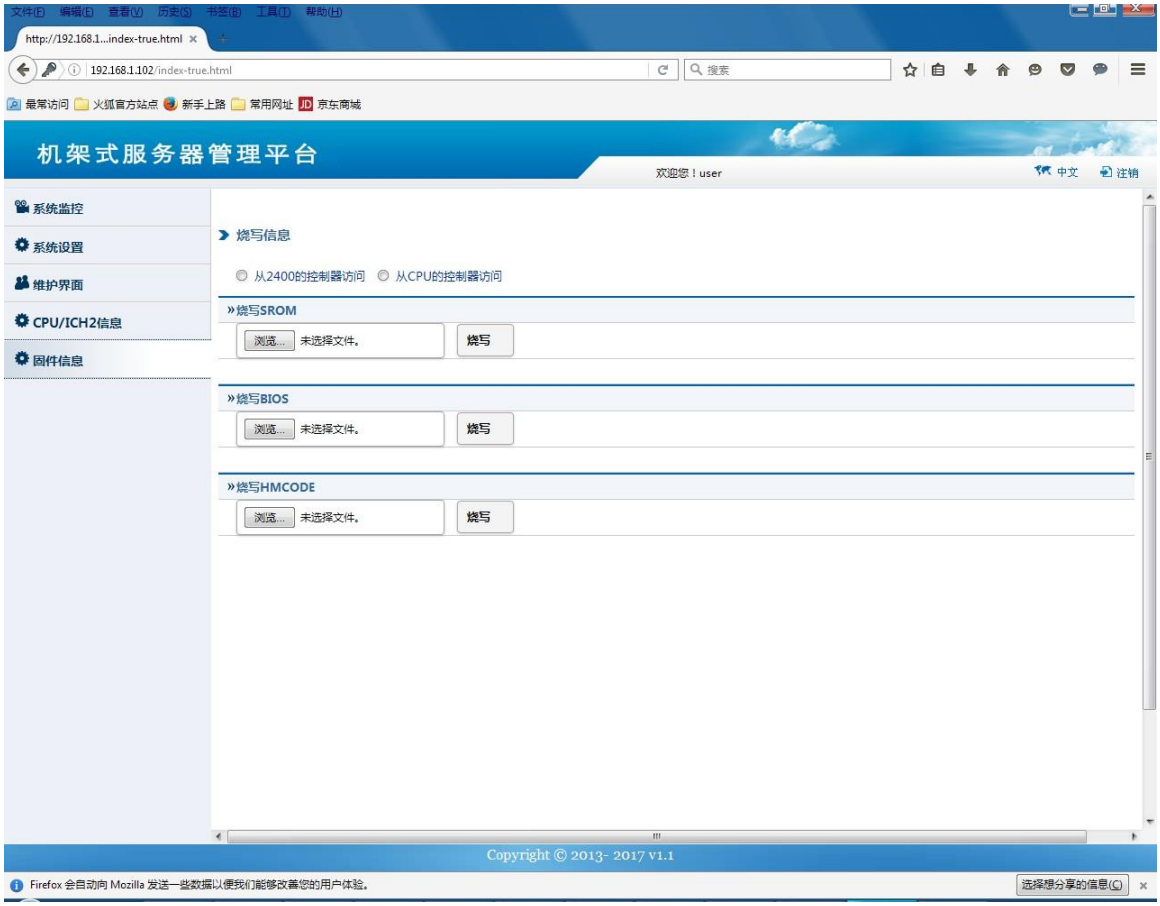
1.1.6 CPU/ICH2 信息配置



注：要使用 ICH2 的界面配置功能，必须确保 ICH2 的 flash 中已经有正确的配置文件，如果 没有，可通过烧写 ICH2 文件功能，加载配置文件到 ICH2 的 flash 中。


机架式服务器管理平台
欢迎您！user
中文
注销
系统监控
系统设置
维护界面
CPU/ICH2信息
固件信息
DC0禁用 DC1禁用 VPU禁用
LegacyIO路由选择
60h/64h端口： 路由到PS2接口 路由到LPC接口
3f8h~3ffh端口： 路由到uart接口 路由到LPC接口
设备配置空间硬能
I2C0 禁用 I2C1禁用 I2C2 禁用 UART0禁用 SPI禁用 PS2禁用 GMAC0禁用
LPC禁用 GPIO禁用 USB禁用 SATA禁用 AC97禁用 GP禁用 GMAC1禁用
»PCIE配置模式
下游端口DS1/DS3配置模式： 两个4X端口
下游端口DS2/DS4配置模式： 一个8X端口
»频率设置
显存工作时钟： 800MHz
PSW工作时钟： 280MHz
主存工作时钟： 333.3333MHz
全局工作时钟： 250MHz
»管控
GMAC0端口工作模式： 明通
I2C 从低三位地址： 000
GMAC1端口工作模式： 明通
I2C速率模式选择： standard
SATA端口工作模式： 明通
硬件初始设置态： 受固件控制
USB端口工作模式： 明通
»输出监测选择
分频监测时钟信号 PCI监测信号 锁相环锁定信号 环振输出信号
PCIE状态信号选择： us_dl_up
PCIE16DIVSEL： PCIE_US_PHY0 16分频输出
锁相环锁定信号选择： GPUPLL_Lock
USBPHY监测时钟选择： USB0PHY 60MHz监测时钟输出
分频时钟信号选择： GPU的PLL输出32分频观测时钟
SATAPHY监测时钟输出： SATA0PHY 50MHz oob时钟输出
»烧写ICH2文件
浏览... 未选择文件。
烧写
Copyright © 2013- 2017 v1.1
Firefox 会自动向 Mozilla 发送一些数据以便我们能够改善您的用户体验。
选择想分享的信息(C)

1.1.7 固件信息



功能说明：

1、 烧写 SROM：请选择从 CPU 的控制器访问，从本地选择 SROM 文件，点击烧写按钮。
2、 烧写 BIOS：请选择从 2400 的控制器访问，从本地选择 BIOS 文件，点击烧写按钮。
3、 烧写 HMCODE：请选择从 2400 的控制器访问，从本地选择 HMCODE 文件，点击烧 写按钮。

2 维护环境安装

2.1 完全新环境

在执行维护命令时，必须在客户端进行如下操作：

1) 安装维护命令包。在 Linux 环境下，需要设置 PATH 和 CLASSPASS 环境变量。PATH 指向维护系统的 bin 目录；CLASSPATH 指向维护系统的 classes 目录）

2) 安装 JAVA 虚拟机。下载 Linux 系统下对应的 JDK，要求版本在 1.6 以上，下载完成后，进行安装。

- ◆ 拷贝 1.6 的安装包：
- ◆ 安装：./jdk-6u26-linux-i586.bin
- ◆ 设置环境变量：$PATH 查看当前 PATH 的设置，vi /root/.bash_profile,export PATH=/usr/java/jdk1.6.0_11/bin:$PATH
- ◆ 退出，重新登录，查看 java 版本 java -version，看是否已经变为 1.6， 如果还是没有，依次查看一下文件：.bash_profile .bashrc /etc/profile

■ 说明： 登录 linux 时，一般先启动 etc/profile,再启动 bash_profile,bashrc，etc/profile 为系统每个用户配置信息，bash_profile 每个用户自己的配置文 件

3)修改 hosts 文件。将安装包中的 hosts 文件，拷贝到/etc 目录下。 注：

目前支持两种模式，如访问目标机 192.168.1.40，host 设置

192.168.1.40 bmc40

192.168.1.40 bmc00040

-o 40:0:0:0 和-o 0:0:40:0 都可以访问到目标机(说明：若访问 -o 0:0:0:0 则 只能访问 bmc0)

3 维护命令格式说明

维护命令的基本格式为：

命令名 <参数…> [参数…] [-help]

说明：<>内的参数表示该命令的必需参数；[]内的参数表示该命令的可选参 数。[-help]参数将打印出该命令简单的命令格式说明。

4 维护命令说明

说明：

1、 客户端：在安装的维护环境主机上，通过维护命令访问维护卡上的维护服务。

2、 维护卡：维护卡上支持的命令操作，可以在维护界面页使用的命令。

3、 界面：维护界面支持的图形化操作。

4.1 读存储器

客户端：

rmem [-o bmcid:midpid:cardid:cupid[:coreid]] [-cache] [-bwdqA] <-a address> [-l length] [-f filename]

维护卡：

rmem [-cache] [-bwdqA] <-a address> [-l length] [-f filename]

读 CPU 核心主存。

参数说明：

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0

-cache：可 cache 方式读，缺省为不可 cache 方式读

-bwdqA：以字节/字/双字/四倍字/ASCII 方式显示读出结果，缺省为双字方式。

-a address：指定主存起始地址

-l length：指定读主存的长度，缺省为 128 字节。

-f filename：将读出结果按照二进制数据格式保存到指定文件，不打印输出。

正确返回：

read mem result 0xXXX(地址)：

XX_XX_XX_XX_XX_XX_XX_XX_XX_XX_XX_XX_XX_XX_X X(数

据)

错误返回： 见附录 A

4.2 写存储器

客户端：

wmem [-o bmcid:midpid:cardid:cupid[:coreid]] <-a address> [-v value[,value[,value...]]] [-cache] [-bwdqA]

维护卡：

wmem <-a address> [-v value[,value[,value...]]] [-cache] [-bwdqA]

写数据到 CPU 核心主存。 **参数说明：**

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0

-a address：指定主存起始地址

-cache：可 cache 方式写，缺省为不可 cache 方式写。

-bwdqA：指定输入数据以字节/字/双字/四倍字/ASCII 方式为单位，缺省为双 字方式。

-v value[,value[,value...]]：指定将写入的数据，缺省时接受命令行输入。

正确返回：

write success

错误返回： 见附录 A

4.3 加载文件到存储器

客户端：loadfile [-o bmcid:midpid:cardid:cupid[:coreid]] <-a address> <-f filename> [-cache] [-c] [-w]

维护卡：

loadfile <-a address> <-f filename> [-cache] [-c] [-w]

加载文件中数据到 CPU 核心主存。 **参数说明：**

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：

CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0

-a address：指定主存起始地址。

-f filename：包含加载数据的文件。

-cache：可 cache 方式写，缺省为不可 cache 方式写。

-c：将写入数据读出与源数据进行比较。

-w：指定数据比较不等时继续操作，缺省时比较不等时中断操作。 **正确返回：**

write file to memory XX bytes succeed!

错误返回： 见附录 A

4.4 读 IO 寄存器

客户端：

rio [-o bmcid:midpid:cardid:cpuid[:coreid]] [-t regType] <-a regAddress|-n regName> [-m mask] [-s serialNo] [-i index] [-p]

维护卡：

rio [-t regType] <-a regAddress|-n regName> [-m mask] [-s serialNo] [-i index] [-p]

读 CPU 的 IO 寄存器。

参数说明：-o bmcid:midpid:cardid:cpuid[:coreid] ：指定 BMC 号：中板号：计算刀片号：CPU 号[：核心号]。对于读 CG 相关 IO 寄存器，可不指定寄存器类型， 直接通过目标号中的核心号来指定。

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修 改.

-t regType:

cg0: registers of CG0 cpm: registers of CPM

mcbox0: registers of MCBOX0 mcbox1: registers of MCBOX1 pub0: registers of PUB0

pub1: registers of PUB1 pcie0: registers of PCI-E0pcie1: registers of PCI-E1

mi: registers of CPU's maintenance interface si: registers of CPU's system interface

-a regaddress：指定寄存器地址。

-n regname：指定寄存器名称。当通过寄存器名称来读 CG 相关寄存器时，可 通过在目标中指定核心号来得到 CG 号。

-m mask：寄存器数据有效指示位，0xf 表示寄存器低 32 位有效，0xf0 表示 寄存器高 32

位有效，0xff 表示寄存器 64 位数据有效，缺省为 0xff。本 参数仅对 PCI-E 相关 IO 寄存器有效。

-p ：打印所有的 IO 寄存器名称。

正确返回：

0xXXXX,XXXX,XXXX,XXXX （注：64 位数据）

错误返回： 见附录 A

例如：查看 CPUID

rio –o bmcnum:0:0:0 –t mi –n cpuid

4.5 写 IO 寄存器

客户端：

wio [-o bmcid:midpid:cardid:cpuid[:coreid]] [-t regType] <-a regAddress|-n regName> <-v value> [-m mask] [-s serialNo] [-i index] [-or|-xor|-and] [-p]

维护卡：

wio [-t regType] <-a regAddress|-n regName> <-v value> [-m mask] [-s serialNo] [-i index] [-or|-xor|-and] [-p]

写数据到 CPU 的 IO 寄存器。

参数说明：

-o bmcid:midpid:cardid:cpuid[:coreid] ：指定 BMC 号：中板号：计算刀片号：CPU 号[：核心号]。对于写 CG 相关 IO 寄存器，可不指定寄存器类型， 直接通过目标号中的核心号来指定。

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修 改.

-t regType：寄存器分类

cg0: registers of CG0 cpm: registers of CPM

mcbox0: registers of MCBOX0 mcbox1: registers of MCBOX1 pub0: registers of PUB0

pub1: registers of PUB1 pcie0: registers of PCI-E0 pcie1: registers of PCI-E1

mi: registers of CPU's maintenance interfacesi: registers of CPU's system interface

-a regaddress：指定寄存器地址。

-n regname：指定寄存器名称。当通过寄存器名称来读 CG 相关寄存器时，可通过在目标中指定核心号来得到 CG 号。

-v value：将写入的数据，64 位或低 32 位有效，仅 PCIE 寄存器可能为 32 位。

-m mask：寄存器数据有效指示位，0xf 表示寄存器低 32 位有效，0xf0 表示寄存器高 32 位有效，0xff 表示寄存器 64 位数据有效，缺省为 0xff。本 参数仅对 PCI-E 相关 IO 寄存器有效。

-or|-xor|-and：分别表示或写、异或写和与写，缺省为覆盖写。

-p ：打印所有的 IO 寄存器名称。

正确返回：

write success

错误返回： 见附录 A

4.6 读 PC 值

客户端：

rpc6a bmnum cgid coreid

维护卡：

rpc <-c coreid> [-cg cgid] 读各个核的 PC 值。 **参数说明：**

-c coreid:cpu core id

-cg cgid: cpu cgid

注：

rpc6a_all bmcnum

读取 16 个核心的 PC 值。

4.7 维护复位

客户端：

mtrst4a <-o bmcid:midpid:cardid:cpuid>

维护卡： 不支持。 维护复位。 **正确返回：**

```
Maintenance Reset succeed!
```

错误返回： 见附录 A

4.8 加载 SROM 到 icache 中

客户端：

loadicache <-o bmcid:midpid:cardid:cpuid> <-f filename>

维护卡：

loadicache <-f filename>

参数说明：

-o bmcid:midpid:cardid:cpuid ： 指定 BMC 号：中板号：计算刀片号：CPU 号

-f filename:需要加载到 flash 中的文件名

4.9 状态监测和扫描

客户端：

scan_out <-o cabid:midpid[:cardid[:cpuid]> <-a startAddr> <-l length> <-s startBit> [-bw bitWidth] [-ra rightAlignBitWidth] [-bwdq] [-t]

维护卡：

scan_out <-a startAddr> <-l length> <-s startBit> [-bw bitWidth] [-ra rightAlignBitWidth] [-bwdq] [-t]

参数说明：

-o bmcid:midpid:cardid:cpuid ： 指定 BMC 号：中板号：计算刀片号：CPU 号

-a startAddr：起始地址，缺省值为 0

-l length：读出的字节长度，缺省值为 128

-s startBit： 读出字节的有效起始位: [0~7]

-bw bitWidth:：读出字节的位宽: [1~8]. 缺省为 1

-ra： 右对齐位宽。.

-bwdq：打印模式:：字节/字/双字/四字 . 缺省为字.

-t：用于测试平台。

4.10 写扫描链

客户端：

scanin [-o bmcid:midpid:cardid:cupid[:coreid]] <-a startAddr> <-f filename>|[-v data0[,data1[,data2...]]]

维护卡：

不支持。

将数据扫描到 CPU 的扫描链中。 **参数说明：**

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0

-a startAddr：指定扫描链起始地址

-f filename：二进制写入文件。

-v data0[,data1[,data2...]]：指定将写入的数据，缺省时接受命令行输入。

-bwdq：输入数据长度模式：字节/字/双字/四字 .

错误返回： 见附录 A

4.11 加载文件到 CPU 的 flash

客户端：

loadflash <-o bmcid:midpid:cardid:cpuid> <-a address> <-f filename>[-c]

维护卡： 不支持。 **参数说明：**

-o bmcid:midpid:cardid:cpuid ：指定 BMC 号：中板号：计算刀片号：CPU 号

-a address：flash 中的偏移地址

-f filename:需要加载到 flash 中的文件名

4.12 读 CPU 的 flash 中的内容

客户端：

rflash <-o bmcid:midpid:cardid:cpuid>[-bwdqA] <-a address> [-l length] <-f filename>

维护卡：

不支持。 **参数说明：**

-o bmcid:midpid:cardid:cpuid ：指定 BMC 号：中板号：计算刀片号：CPU 号

-a address：flash 中的偏移地址，比如，bios 存放的起始地址为 0x100000； hmcode 存放的起始地址为 0x10000；配置信息存放的起始地址为 0xf000

-l length：读出数据的字节数

-f filename：读出的数据保存到文件中（可选）

- bwdqA：结果显示格式，b–字节 w–字（2 字节）d—双字（4 字节） A—ASCII 码（必须是大写）

例：

rflash -o 11:0:0:0 -a 0xf000 -l 0x20 –b

返回：

0x000000f000: 00 00 10 00 c0 a8 01 0b 0b 01 a8 c0 aa 00 06 00

0x000000f010: 00 0f 0d 06 09 01 00 00 00 00 00 00 00 00 00 00

4. 13 加载 BIOS 到 CPU 的 flash（界面支持）

客户端：

loadbios <-o bmcid:midpid:cardid:cpuid> <-f filename>[-c]

维护卡： 不支持。 **参数说明：**

-o bmcid:midpid:cardid:cpuid ：指定 BMC 号：中板号：计算刀片号：CPU 号

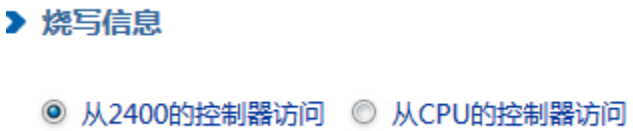
-f filename:需要加载到 flash 中的文件名

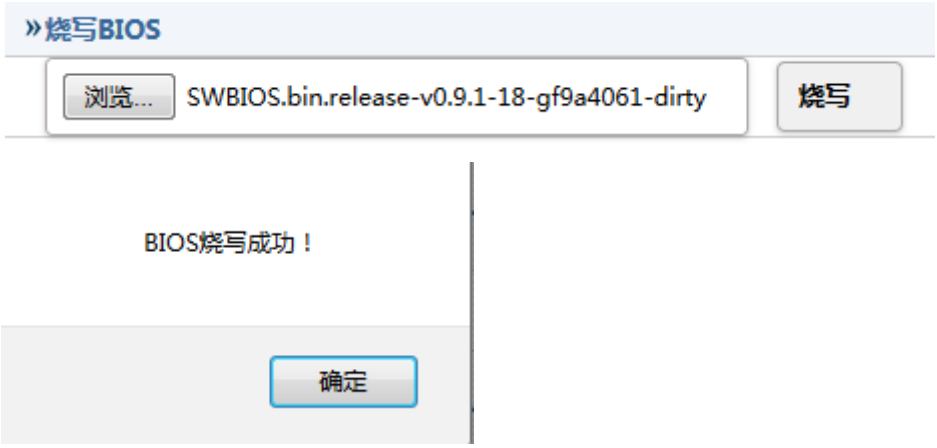
-c 与源文件进行比较。

注：该命令自动将 BIOS 加载到第二片 flash 中定义的地址，并自动将 BIOS 的文件长度写入配置文件所对应的位置，并烧录到 flash 中。 **界面：**

使用说明：

在固件信息—>烧写信息，选择从 2400 控制器访问，在烧写 BIOS 一栏中， 选择需要烧写的文件，点击烧写按钮，烧写正确后，界面跳出烧写成功提示框。





4.14 加载 hmcode 到第二片 flash（界面支持）

客户端：

loadhmcod <-o bmcid:midpid:cardid:cpuid> <-f filename>[-c]

维护卡： 不支持。参数说明：

-o bmcid:midpid:cardid:cpuid ： 指定 BMC 号：中板号：计算刀片号：CPU 号

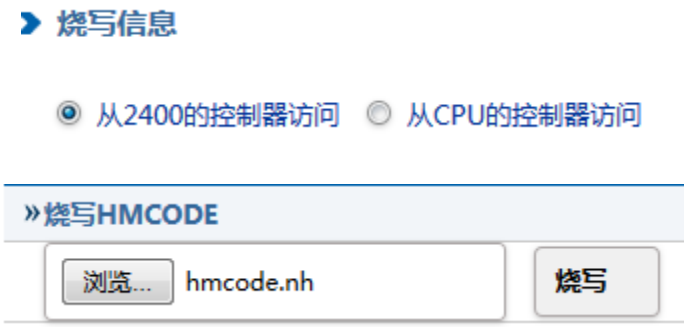
-f filename:需要加载到 flash 中的文件名

-c 与源文件进行比较。

注：该命令自动将 hmcode 加载到第二片 flash 中定义的地址，并自动将 hmcode 的文件长度写入配置文件所对应的位置，并烧录到 flash 中。

界面： 使用说明：

在固件信息—>烧写信息，选择从 2400 控制器访问，在烧写 HMCODE 一栏中，选择需要烧写的文件，点击烧写按钮，烧写正确后，界面跳出烧写成功提示框。



4.15 加载 SROM 到 CPU 的 flash（界面支持）

客户端：

loadsrom <-o bmcid:midpid:cardid:cpuid><-f filename>[-c]

维护卡： 不支持。 **参数说明：**

-o bmcid:midpid:cardid:cpuid ： 指定 BMC 号：中板号：计算刀片号：CPU 号

-f filename:需要加载到 flash 中的文件名

-c 与源文件进行比较。

注：该命令自动将 SROM 加载到第二片 flash 中定义的地址，并自动将 SROM 的文件长度写入配置文件所对应的位置，并烧录到 flash 中。界面： 使用说明：

在固件信息—>烧写信息，选择从 CPU 控制器访问，在烧写 SROM 一栏中， 选择需要烧写的文件，点击烧写按钮，烧写正确后，界面跳出烧写成功提示 框。

4.16 读 IIC 设备

客户端：

riic<-o bmcid:midpid:cardid:cpuid> <-m master> <-b busid> <-a address>

<-rega regAddr> <-l readlen> [-f filename]|[-ps]

参数说明：

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：

CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0

-m master：IIC 主控制器选择。

-b busid：IIC 通道号选择。

-a address：IIC 设备的 SLAVE 地址

-rega regaAddr：IIC 设备的寄存器地址。

-l readlen：需要读取的字节长度。

-f filename：将读出结果按照二进制数据格式保存到指定文件，不打印输出。

-ps：采用 GBK 码显示结果。

正确返回：

0x address ： XX XX XX(显示字节数=readlen)

或 Write data to file filename Succeed!

错误返回：见附录 A

维护卡：

i2c-test <-b busnumber> <-s address> <-m mode> <-rc count> <-d bytes>

参数说明：

-b busnumber：IIC 通道号选择。

-s address：IIC 设备的 SLAVE 地址，维护卡使用七位地址，需要进行地址转 化，如 0xc0(11000000)应该换为 0x60（01100000）。

-m mode：模式选择，默认选择 1.

-rc count：需要读取的字节长度。

-d bytes：IIC 设备的寄存器地址，该位必须放在最后。

正确返回：

0x address ： XX XX XX(显示字节数=readlen)

4.17 写 IIC 设备

wiic <-o bmcid:midpid:cardid:cpuid> <-m master> <-b busid> <-a address> <-rega regAddr> <-s string>| <-v value1[,value2…]>|<-f filename> [-c]

参数说明：

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0

-m master：IIC 主控制器选择。

-b busid：IIC 通道号选择。

-a address：IIC 设备的 SLAVE 地址

-rega regaAddr：IIC 设备的寄存器地址。

-f filename：二进制写入文件。

-c 与写入的文件做比较。 **正确返回：**

```
Write data to IIC device Succeed!
```

错误返回： 见附录 A **维护卡：**

i2c-test <-b busnumber> <-s address> <-m mode> <-w> <-d bytes> < value1 > < value2 >…

参数说明：

-b busnumber：IIC 通道号选择。

-s address：IIC 设备的 SLAVE 地址，维护卡使用七位地址，需要进行地址转 化，如 0xc0(11000000)应该换为 0x60（01100000）。

-m mode：模式选择，默认选择 1.

-w：写操作。

-d bytes：IIC 设备的寄存器地址，该位必须放在最后。

4.18 读版本信息

客户端：

rversion <-o bmcid:midpid:cardid:cpuid>

维护卡： 不支持。 **参数说明：**

-o bmcid:midpid:cardid:cpuid ： 指定 BMC 号：中板号：计算刀片号：CPU 号

正确返回：

BMC firmware Version ： XX.XX.XX.XX Srom Information ： XX.XX.XX.XX Hmcode Information：XX.XX.XX.XX BIOS Information：XX.XX.XX.XX 各文件信息

错误返回：

见附录 A

4.19 读取/修改 CPU 的频率配置（界面支持）

客户端：

cpufreq <-o bmcid:midpid:cardid:cpuid >（暂时不支持 SW6A） 读取 CPU 的核心频率、存控频率以及核心互联频率 **参数说明：**

-o bmcid:midpid:cardid:cpuid ： 指定 BMC 号：中板号：计算刀片号：CPU 号 midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修 改；

正确返回：

例：BMC 15: core/memory/xbx frequency(set) : 1200MHz / 266.67MHz / 1000MHz

BMC 15: core/memory/xbx frequency(effect) : 1200MHz / 266.67MHz / 1000MHz

cpufreq <-o bmcid:midpid:cardid:cpuid> [-c core_freq] [-m mm_freq] [-x xbx_freq] [-p]

修改 CPU 的核心频率、存控频率以及核心互联频率

参数说明：

-o bmcid:midpid:cardid:cpuid ： 指定 BMC 号：中板号：计算刀片号：CPU 号

-c core_freq：需要配的核心频率值

-m mm_freq：需要配的存控频率值

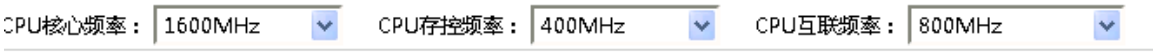
-x xbx_freq：需要配的核心互联频率值

-p 打印出频率和值的对应关系

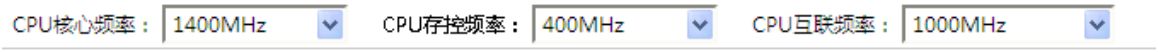
注：每个值都可单独设置

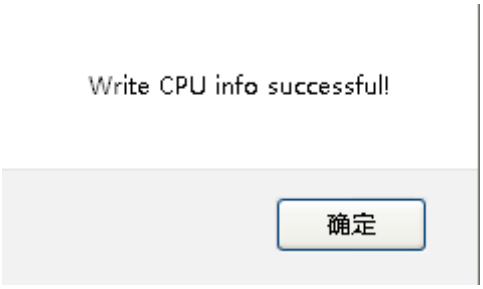
界面：支持读取与修改 CPU 的核心频率、存控频率以及核心互联频率 **使用说明：**

读取：在 CPU/ICH2 信息—>CPU 信息，选择对应的 CPU 类型，点击读取按 钮，读取正常时界面会弹出提示框，点击确定后即可获取到相关频率值。



烧写：点开各频率的下拉框，选择需要修改的频率值，点击烧写按钮，烧写完成后会弹出提示框。





4. 20 读取/修改 CPU 的温度阈值（界面支持）

客户端：

cputemper<-o bmcid:midpid:cardid:cpuid >（暂时不支持 SW6A） 读取 CPU 的当前温度、下限阀值以及上限阀值

参数说明：-o bmcid:midpid:cardid:cpuid ：指定 BMC 号：中板号：计算刀片号：CPU 号 midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修 改；

正确返回：

例：cpu temperature: XX.XX

cpu lower temperature alarm trip:- XX cpu upper temperature alarm trip: XX

cputemper <-o bmcid:midpid:cardid:cpuid> [-h high_temp] [-l low_temp]

修改 CPU 的上限阀值以及下限阀值 **参数说明：**

-o bmcid:midpid:cardid:cpuid ：指定 BMC 号：中板号：计算刀片号：CPU 号

-h high_temp：需要配的高温上限阀值

-l low_temp：需要配的低温下限阀值 **注：**每个值都可单独设置

界面：支持读取 CPU 的当前温度、下限阀值以及上限阀值，不支持直接修 改上下限阈值。

使用说明：

在系统监控—>服务器健康信息中的 CPU_Temp 一行，显示CPU 温度的状态、 当前温度，上下限阈值，不支持修改温度阈值。

4. 21 修改维护端口 IP 地址（界面支持）

cmtip <-o bmcid:midpid:cardid:cpuid> [-v ip] （暂时不支持） **参数说明：**

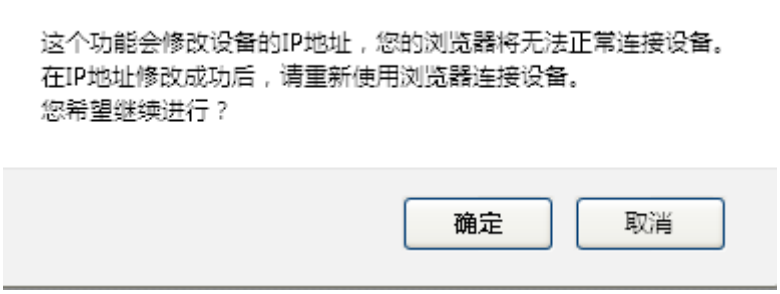
-v: 维护网口的 ip 地址 例：cmtip -o 10:0:0:0 -v 192.168.1.11

说明：修改完后需要维护复位或断电重启后才能生效

界面：支持修改 IP

使用说明：

在系统设置—>网络配置中—>IP 地址，填写需要修改的 IP 地址，点击设置 按钮，在弹出的对话框中点击确定，等待 5s 后，打开浏览器重新登录新的 IP 号。



4. 22 读 ICH2 内部寄存器

rport <-o bmcid:midpid:cardid:cpuid> <-t type> <-a address>

参数说明：

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0

-t type：访问类型选择：1，2 or 4 字节寄存器。

-a address：IO 端口地址。 **正确返回：** 0xXX_XX_XX_XX **错误返回：** 见附录 A

注：读地址 0x8000000c 时，为读取 CPU 的 FSM 状态，与 SW4A 兼容。 基地址为 0xf0000000 时，为读取维护卡上的 GPIO，

例：读 GPIOB0 的值

readioport bmcnum f0000008

4. 23 写 ICH2 内部寄存器

wport <-o bmcid:midpid:cardid:cpuid><-t type> <-a address> <-v value>

参数说明：

-o bmcid:midpid:cardid:cpuid:coreid ：指定 BMC 号：中板号：计算刀片号：CPU 号：核心号

midpid，cardid，cpuid 都为 0，为了与 1P 命令兼容，减少软件脚本修改；

coreid：缺省表示访问 CG0 -t type：访问类型选择：1，2 or 4 字节寄存器。

-a address：IO 端口地址。

-v value：写入寄存器的数据。 **正确返回：**

```
Write data to IO port success!
```

错误返回： 见附录 A 注：

基地址为 0xf0000000 时，为写维护卡上的 GPIO， 例：写 1 到 GPIOB0

readioport bmcnum f0000008 1

4. 24 写 1 字节 ICH2 内部寄存器

命令格式：

writeioport1 bmcnum addr data

参数说明：

bmcnum：BMC 号。 addr：寄存器的地址。 data：向寄存器写入的数据。

4. 25 读 1 字节 ICH2 内部寄存器

命令格式：

readioport1 bmcnum addr

参数说明：

bmcnum：BMC 号。

addr：寄存器的地址

4. 26 写 2 字节 ICH2 内部寄存器

命令格式：

writeioport2 bmcnum addr data

参数说明：

bmcnum：BMC 号。 addr：寄存器的地址。 data：向寄存器写入的数据。

4. 27 读 2 字节 ICH2 内部寄存器

命令格式：

readioport2 bmcnum addr

参数说明：

bmcnum：BMC 号。

addr：寄存器的地址。

4.28 1.4.28. 写 4 字节 ICH2 内部寄存器

命令格式：

writeioport4(writeioport) bmcnum addr data

参数说明：

bmcnum：BMC 号。 addr：寄存器的地址。 data：向寄存器写入的数据。

4.29 读 4 字节 ICH2 内部寄存器

命令格式：

readioport4(readioport) bmcnum addr

参数说明：

bmcnum：BMC 号。

addr：寄存器的地址。

4.30 显示客户端版本信息

命令格式：

readme [-i]

参数说明：

i：版本更改信息记录。

5 维护脚本和工具使用说明

5.1 查看各核打印信息

lazyrrk bmcnum core_id 查看 bios 打印信息。 **参数说明：** bmcnum：指定 BMC 号。core_id：核号。

6 CPU 监测工具的使用

1) 确认 JDK 版本是否为 1.6 以上，否则，升级至 1.6 以上版本。
2) 运行 java -jar socmainplatform_sw6a.jar ，会出现图型界面。

7 神威平台频率配置（SW6A）

1）核心时钟配置：根据参考时钟输入引脚（RCLK）的时钟频率和核心时钟 配置引脚 CFG_CORE [3:0]_H 或者维护配置寄存器 INIT_CTL[CORE_FREQ]，通过核 心 PLL 产生核心时钟。核心时钟配置引脚的具体定义如表 7-5 所示。

表 2-1：核心时钟配置表

CFG_CORE_H[3:0]	核心时钟工作频率（MHz）
0	旁路（200MHz）
1	800
2	1000
3	1200
4	1400
5	1500
6	1600
7	1700
8	1750
9	1800
10	1850
11	1900

12	1950
13	2000
14	2050
15	2100

2）存储控制器时钟配置：根据参考时钟输入引脚（RCLK）和存储器接口配 置引脚 CFG_MM [2:0]_H 或者维护寄存器 INIT_CTL[MM_FREQ]，通过存控 PLL 产 生存控时钟。存储控制器时钟配置引脚的具体定义如表 7-6 所示。

表 2-2：存储控制器时钟配置表

CFG_MM_H[2:0]	存控时钟频率（MHz）
0	旁路（200MHz）
1	266
2	300
3	333
4	366
5	400
6	433
7	466

3）互连时钟配置：根据参考时钟输入引脚（RCLK）的时钟频率和互连时钟 配置引脚 CFG_XBX [3:0]_H 或者维护寄存器 INIT_CTL[XBX_FREQ]，通过互联 PLL 产生互连时钟。互连时钟配置引脚的具体定义如表 7-7 所示。

表 2-3：互连时钟配置表

CFG_XBX_H[2:0]	互连时钟工作频率（MHz）
0	旁路（200MHz）
1	800
2	1000
3	1050
4	1150
5	1250
6	1300
7	1333

附录 A-错误码定义

错误码	错误信息	说明
0x50	SW2 report read back failed,control error	错误读响应（带数据，含控制错）
0x51	SW2 report read back failed,illegal address	非法地址读响应（不带数据）
0x52	SW2 report write failed,illegal address	非法地址写结束
0x53	SW2 report req package parity Error	维护串口校验错响应（不带数据）
0x54	SW2 report illegal cmd ack respond	维护串口非法命令响应（不带数据）
0x55	SW2 report write failed,control error	带控制错写结束
0x56	BMC report wait ack timeout	BMC 维护超时
0x57	BMC report console req package parity Error	BMC 命令校验错响应
0x58	BMC report read or write error	BMC 读写错误
0x59	Console report ack cmd not defined	未定义的响应包