GB-II iSCB User's Manual

Preliminary

Rev 0.3

Company **APPRO International, Inc.**	Documentation Number xxxxxxxxxxx	Status **Preliminary**
Document Type **User's Manual**	Project **GB-II Management**	Revision **1.0 -- Aug 4, 2011**
Responsible Engineer **Francis Kunoh Lee**	Manager Review **John Lee**	Department **ATS**

REVISION HISTORY

Revision	Revision Date	Authors	Details of Change
0.1	07/02/2011	Francis	Initial Release – preliminary. Based on iSCB firmware v0.1 rc1
0.2	08/05/2011	Francis	Based on iSCB firmware v0.1 rc2
0.3	09/08/2011	Francis	Based on iSCB firmware v0.1 rc5

TABLE OF CONTENTS

1 Overview

Appro GB-II iSCB provides an easy to use command line interface and API through the network communication on the Subrack. This document describes usage of the command line interface and SNMP.

2 Key Features

Since iSCB is based on legacy iSCB from the GreenBlade-I blade product, it includes all of legacy iSCB capability.

- User Interface: CLI, SNMP
- Power Control: On/Off/Cycle/Reset
- ID-LED Control: On/Off
- GPU: Enable/Disable
- Remote Console: 16 Channel Simultaneous Serial Console (COM1) redirect to TCP port
- Node Identify: Present, On/Off, BMC MAC/IP/Version
- Deploy: Cabinet/Subrack/iSCB enumeration
- PSU Monitoring: Status, Fan1/Fan2 Speed, Temperature, 12VA, ACVA
- CFU Monitoring: Status, Fan1/Fan2 Speed, Current PWM Duty (%)
- Active CFU Control: Low, Normal, High control by BMC temperature sensors
- Hot Swap: Node, PSU, CFU, iSCB
- Redundant: Dual iSCB
- Support Powerman/Conman
- iSCB firmware: Remote update and USB flash for production
- E-Mail and SNMP Trap notification

3 Specification

3.1 Hardware

- 400MHz i.MX253 ARM9 SoC
- 128MB Flash ROM
- 128MB RAM
- 256B NVRAM
- 1 USB 2.0 OTG
- 1 Ethernet port
- 1 Serial Console
- 16 channel UARTs
- Three I2C buses
- 16bit GPIO expanders

3.2 Software

- embedded Linux-2.6.31
- u-boot

- jffs2 filesystem
- net-snmp and APPRO Private MIB (IANA 38067)
- telnet
- tftp
- Cisco style CLI service
- Remote update and usb flash

4 Option Switches, Ports and LEDs

4.1 Reset Button

The engagement of the iSCB reset button does not affect the Compute Nodes in the sub rack and in the case of iSCB redundant configuration, it will not impact the second iSCB module.

4.2 Status LED

GREEN LED: iSCB service is running
Steady: Preparing Service
0.5 seconds blinking: Running
RED LED: iSCB detects an Error
Steady: Error while Startup
0.5 seconds blinking: Node/FAN control error
0.2 seconds blinking: No device or Internal BUS Error
GREEN/RED Steady:
Startup
GREEN/RED Blink:
0.5 seconds blinking: Health is abnormal
0.2 seconds blinking: Identifying
GREEN/RED Wink:
0.5 seconds blinking: Initializing resources
0.2 seconds blinking: Update firmware progress

4.3 Ethernet Port

- 10 or 100 Base-T, Full/Half-Duplex
- Link & Activity indicator
- Default IP Address with IDSW module: ~~192~~10.10.CC.US
  CC : Cabinet ID from the IDSW on the Sub-rack
  U : iSCB UNIT ID
  S : Subrack ID from the IDSW on the Sub-rack
- Default Net Mask: 255.255.0.0
- Default Gateway: 10.10.1.254
- Default IP Address without IDSW module: 10.10.1.10

4.4 Serial Port (CN2)

Debug and Maintenance purposes.

Baud: 115200, N, 8, 1
Flow Control: None

Note: To connect to Standard DB9 COM port, please use cable PN:XXX-XXXX

4.5 Option Switches

SW1

00: Normal Service
11: Flash iSCB firmware using USB cable

SW2

S1: Diagnostic mode. Self diagnostic using iSCB Test Jig for volume production.
S2: Debug mode. Print messages out into the log and allow log-in without authentication.
S3: Reserved. Future use.
S4: Reserved. Future use.

5 Diagnostic

5.1 Flashing firmware

- **Use standard A-B type USB cable**
- **Firmware flashing using iflash and flash.sh command line utility on Linux system.**
- **RHEL 5.x higher**

iflash

iflash is command line iSCB firmware flash utility for linux. Usually it is called by flash.sh.

flash.sh

flash.sh is a front-end iSCB firmware flash tool. It calls iflash to send the firmware image.

Flash procedure

Step 1. Set iSCB SW1 to FWUPDATE mode.
Step 2. Connect USB cable between iSCB and Linux system.
Step 3. Connect iSCB console cable to verify.
Step 4. run flash.sh with the firmware image and iSCB's mac address as follows:
[root@localhost ~]# ./flash.sh iscb-1.0.bin.gz 00:0b:3f:71:00:01
Step 5. When flashing is done, check up the iSCB console to flash and memory test done.
Step 6. Set iSCB SW1 to Normal Service mode (00).
Step 7. Power discharge reset iSCB to start up.

5.2 Diagnostic

iSCB provides a diagnostic utility for the production stage. This utility tests the following items:

- GPIO
  PEER_REQ/PEER_ACK
  RESET_OUT/PEER_PRESENT/IO_INT
  MASTER_ON/ISCB_ID
  IOBD_RESET/MISC_INT
- I2C bus
  IPMB, MISC, IO
- 16 UART Ports
  USB0-11, ttymxc1-4

This utility requires the following components:

- iSCB serial console cable
- ATX Power supply
- iSCB diagnostic module (Test JIG)

Test Procedure

Step 1. Set iSCB SW2-1 to DIAG mode.
Step 2. Connect iSCB console cable to verify.
Step 3. Connect Test JIG to iSCB.
Step 4. Wait for the startup of the diagnostic tool instead of the login prompt.
Step 5. Check the diagnostic result by viewing the LED lights and monitor console.
Step 6. If something is not ok, you can test again by any key. Otherwise <r> to reset.

```
[[[ GreenBlade-II iSCB Diagnostic Tool v0.1           ATS, APPRO Inc. ]]]

<<GPIO>>
   GPIO_PEER_REQ > GPIO_PEER_ACK    : 0:ok  1:ok
  GPIO_RESET_OUT > GPIO_PEER_PRESENT: 0:ok  1:ok
  GPIO_RESET_OUT > GPIO_IO_INT      : 0:ok  1:ok
  GPIO_MASTER_ON > GPIO_ISCB_ID     : 0:ok  1:ok
 GPIO_IOBD_RESET > GPIO_MISC_INT    : 0:ok  1:ok

<<I2C BUS>>
 I2C-0:ok I2C-1:ok I2C-2:ok

<<UART>>
 UART1:ok UART2:ok UART3:ok UART4:ok UART5:ok UART6:ok UART7:ok UART8:ok UART9:o
k UART10:ok UART11:ok UART12:ok UART13:ok UART14:ok UART15:ok UART16:ok

Test done. <q> to quit. <r> to reset. Any key to Test again.:
```

6 Setup

6.1 Installation into GB-II Sub-rack

- iSCB supports Hot-Swap capability. Insertion and removal doesn't affect the running compute nodes and the other side iSCB.

- While iSCB is inserted into Sub-rack, iSCB automatically initializes sub-rack resources and the setup IP address using the NVRAM settings on the sub-rack IDSW module. It takes a minute.
- If you want to setup your ip address, you can do so by using the 'set ip' command.
- If the subrack has a problem and iSCB cannot access it properly, iSCB will set the IP address to 10.10.1.10.

6.2 Log-in

When iSCB is running and ready to service, you can log-in using telnet.

```
[kolee@confucius ~]$ telnet 10.10.1.11 7000
Trying 10.10.1.11...
Connected to 10.10.1.11 (10.10.1.11).
Escape character is '^]'.
1:1:1-iSCB>
```

6.3 Password

- iSCB authenticates the access permission whenever connected if there is a valid password in NVRAM.
- The default password is none. So the administrator should setup a new password when initially setting up iSCB. It can be changed by the 'set passwd' command at the CLI.
- The new password will be stored in the NVRAM area on the subrack. (Not iSCB).
- If you want to apply permanentaly, you have to update NVRAM using 'nvram save' command at the CLI.
- If you lose your password, you can reset the password using the following method:
  Step 1. Set debug mode using iSCB SW2-2 ON.
  Step 2. Login into CLI with bypass password.
  Step 3. Run the 'set passwd' command to set password and then run the 'nvram save' command to save it to the NVRAM area.
  Step 4. Unset debug mode using iSCB SW2-2 OFF.
  Step 5. Restart iSCB.

6.4 IP Address

- When iSCB startup, IP address will be automatically set up by the NVRAM settings.
- There are two cases that iSCB doesn't use NVRAM settings:
  - NVRAM checksum is invalid
  - Cabinet/Sub-rack ID are mismatched between NVRAM and IDSW

  In this case, iSCB setup default IP Address using Cabinet/Sub-rack ID. See Ethernet Port section.
- If you want to change your IP Address, you can use the 'set ip' command as following:
  Step 1. set ip ~~192~~10.~~168~~10.1.100
  Step 2. set nm 255.255.0.0
  Step 3. set gw ~~192~~10.~~168~~10.1.254

Step 4. nvram save
Step 5. reboot

- The changed IP address will be applied after reboot.

6.5 Settings in NVRAM on the Sub-rack

- GB-2 Sub-rack has a NVRAM (non-volatile memory) on the IDSW module and it contains the items listed below. Once you initially access the iSCB, you may need to setup the NVRAM parameters using the 'set' command.
  - User defined sub-rack name (32 chars max)
  - Cabinet/Sub-rack ID
  - iSCB IP Address/Netmask/Gateway/DNS
  - Service Port (TCP, 7000)
  - TTY baud rate (115200)
  - SNMP Trap Community (public, 32 chars max)
  - Receiver IP Address
  - E-mail Address (32 chars max)
  - Mail server/user/pass (16 chars max)
  - Login Password
  - Log Level (info)
  - Monitor Polling interval (180 seconds)
  - Default Fan Speed (normal, 85%)
  - NVRAM Checksum
- 'nvram' command shows the current settings in NVRAM.

```
1:1:1-iSCB> nvram
NVRAM Ver.  : 1.0
Subrack Name: SR5110-1-1
Cabinet ID  : 1
Subrack ID  : 1
SCB1 IP Addr: 10.10.1.11/255.255.0.0
SCB1 Gateway: 10.10.1.254
SCB2 IP Addr: 10.10.2.11/255.255.0.0
SCB2 Gateway: 10.10.2.254
Service Port: 7000
TTY speed   : 115200
SNMP comm   : public
Receiver1   : 0.0.0.0
Receiver2   : 0.0.0.0
E-Mail      :
Mail Server :
Mail UserID :
Mail Passwd :
Password    :
Log Level   : info
Polling intv: 180
ok
```

7 Monitoring

7.1 Sub-rack resource

When iSCB is running, the following areas are being watched by iSCB.

- Compute Nodes on slot
  - Present, Power, ID-LED
  - Remote Console
  - BMC Sensors (Temperature only)

- Power Supply Unit (PSU)
  - Present, Status(PSOK)
  - Temperature, Fan1/Fan2 Speed
  - 12VA, ACV, Watt

- Cooling Fan Unit (CFU)
  - Present, Status
  - Fan1/Fan 2 Speed
  - PWM Duty

- iSCB Service
  - Established CLI service
  - Terminal service ports

7.2 Compute Node Temperature

iSCB can access the BMC on the CN through the IPMB bus and acquire the temperature sensor's status. Then control the CFU speed and send an event log to the administrator via E-mail and SNMP Trap for the WARNING, CRITICAL and NON RECOVERABLE sensors.

```
1:1:1-iSCB> sensor 3
 P0 Therm Margin: ok  ( -48.0 'C)
P0 DTS Therm Mgn: ok  ( -48.0 'C)
DIMM Thrm Mrgn 1: ok  ( -48.0 'C)
   BB Inlet Temp: ok  (  25.0 'C)
     BB BMC Temp: ok  (  33.0 'C)
    CPU0 VR Temp: ok  (  29.0 'C)
     IB QDR Temp: ok  (  34.0 'C)
        PCH Temp: ok  (  43.0 'C)
 Powerville Temp: ok  (  44.0 'C)
 P0 Therm Ctrl %: ok  (   0.0 unspecified)
ok
1:1:1-iSCB>
```

8 Node Control

There are two methods to control the node in iSCB:

- Command Line
- SNMP

8.1 Power Control

- iSCB can power on/off/cycle/reset to a specific node or to all nodes. This operation is not related to OS graceful shutdown. The result of power control will affect the CFU speed.
- An interval time is required for multiple power on/off/cycle operations to protect the power line. The time unit is 100 mili seconds and 0 means no interval.
- You can interrupt the power operation to multiple nodes using <CR> key.

```
1:1:1-iSCB> power off all
interval is required for multiple nodes.
1:1:1-iSCB> power off all 0
ok
1:1:1-iSCB>
```

8.2 ID-LED Control

The ID-LED on the compute node is controlled by iSCB and node BMC. This operation doesn't need interval time for multiple nodes.

8.3 GBGPU Control

If CN has GBGPU expansion and IDSW SW1-6 is on, the GBGPU expansion can be enabled or disabled by the 'gpu on/off' command for power savings. GBGPU power is determined by the combination of node's power and GBGPU enable/disable setting.

Note: If the GBGPU power status is changed by the 'gpu on/off' command when the system is running, the system should be rebooted to apply the new GBGPU status setting.

9 Fan Control

Turning on the IDSW1-7 switch will enable Active Fan Control. Turning off the IDSW1-7 switch will enable Passive Fan Control.

9.1 Passive Fan Control

When IDSW1-7 is off, then the iSCB will operate in a Passive Fan Control mode. In this mode, CFU control is via user defined PWM duty value. The iSCB fan control is independent of the compute node's health status.

Note: Care must be given when selecting a user-defined PWM duty value, because if the fan speed is too low, compute nodes may get damaged due to overheating.

9.2 Active Fan Control

When the IDSW1-7 is on, then the iSCB will operate in an Active Fan Control mode. In this mode, iSCB checks the compute node's health status and automatically adjusts CFU's PWM Duty value as follows:

- CN temperature > 65 or unknown: Set PWM duty to High (100%)
- CN temperature > 50: Set PWM duty to Normal (85%)
- CN temperature <= 50: Set PWM duty to Low (60%)
- No power or empty blade slot: Set PWM duty to 15%

- PWM duty value will be adjusted incrementally by 5% every 10-15 seconds period until pre-set value is met.

9.3 Compute Node Localize

iSCB handles individual CFU using the following slot mappings.

Rack Type	CFU1	CFU2	CFU3	CFU4	CFU5	CFU6
SR5110	Slot 8-10	Slot 4-7	Slot 1-3			
SR5110-GPU	Slot 4-5	Slot 2-4	Slot 1-2			
SR8116	Slot 6-8	Slot 3-6	Slot 1-3	Slot 14-16	Slot 11-14	Slot 8-10
SR8116-GPU	Slot 3-4	Slot 2-3	Slot 1-2	Slot 7-8	Slot 6-7	Slot 5-6
SR8104	Slot 3-4	Slot 2-3	Slot 1-2	Slot 3-4	Slot 2-3	Slot 1-2

9.4 FCB overrides iSCB

- If all compute nodes are not powered on, the FCB (fan control board) overrides the iSCB fan control and will force a slow down (1000rpm) in the fan speed.
- If there is no iSCB installed, CFU will run at100% full speed.

10 Event Log and Notification

10.1 Event Log

iSCB has three severity levels to control and it is handled by syslogd.

- Critical (EMERGENCY, ALERT, CRITICAL, ERR)
- Warning (WARNING)
- Info (NOTICE, INFO)

The event logs are stored in /var/log/messages and you can browse the logs using the ‘log’ command.

10.2 Log level

Every event log is filtered by Log Level Setting. For example, if the log level is critical, iSCB will process critical or higher level events only. You can set the log level using the ‘set loglevel’ command.

- 0: info
- 1: warning
- 2: critical

10.3 Notification

iSCB provides two kinds of user notification methods:

- SNMP Trap(v2c)
- E-Mail(SMTP)

Every event log is sent by SNMP trap and E-mail notification handler.

11 SNMP

11.1 SNMP

iSCB provides SNMP protocol to control and monitor.

- Version: v2c
- Community: public (by default. Can be changed by 'set community' command)
- Port number: UDP 161
- IANA Private Enterprise Number: 38067

To access the SNMP service from iSCB, you may need to get iSCB's private MIB information. You can download the iSCB's private MIB file from the iSCB as below.

```
[root@localhost ~]# tftp 10.10.1.11 -c get /tmp/APPRO-GB2-MIB.txt
```

And then you can access the iSCB using the MIB information through the SNMP protocol. There are three kinds of access methods from the SNMP specifications:

- Get: get information from specific OID
- Walk: get entire information from specific OID and it's subtree
- Set: set value into specific OID

Following are examples of how to access the iSCB through the SNMP protocol. For more detailed OID information, see the MIB file.

```
[root@localhost ~]# snmpwalk -v 2c -c public
10.10.1.11 .1.3.6.1.4.1.38067.1
SNMPv2-SMI::enterprises.38067.1.1 = STRING: "NOTICE: PSU2 is back to
normal
SNMPv2-SMI::enterprises.38067.1.3 = STRING: "SR5110"
SNMPv2-SMI::enterprises.38067.1.4 = INTEGER: 1
SNMPv2-SMI::enterprises.38067.1.5 = INTEGER: 1
SNMPv2-SMI::enterprises.38067.1.6 = INTEGER: 1
SNMPv2-SMI::enterprises.38067.1.7 = INTEGER: 0
SNMPv2-SMI::enterprises.38067.1.8 = STRING: "on
SNMPv2-SMI::enterprises.38067.1.9 = INTEGER: 1
SNMPv2-SMI::enterprises.38067.1.10 = STRING: "static"
SNMPv2-SMI::enterprises.38067.1.11 = INTEGER: 85
SNMPv2-SMI::enterprises.38067.1.12 = INTEGER: 2
SNMPv2-SMI::enterprises.38067.1.13 = INTEGER: 10
SNMPv2-SMI::enterprises.38067.1.14 = INTEGER: 4
SNMPv2-SMI::enterprises.38067.1.15 = INTEGER: 3
SNMPv2-SMI::enterprises.38067.1.16 = INTEGER: 2
SNMPv2-SMI::enterprises.38067.1.17 = INTEGER: 230
```

11.2 SNMP Trap

Once an event log has been generated, iSCB will send the SNMP trap message to the Trap Receiver.

- Version: v2c
- Community: public (by default. Can be changed by 'set community' command)

- Port number: UDP 162
- IANA Private Enterprise Number: 38067
- Trap receiver: A system where SNMP trap daemon is running.

12 Command Line service and Terminal Server

iSCB provides 16-channel simultaneous/redundant remote console for Compute Node and up to 4 channel command line service.

12.1 Command Line Service

iSCB CLI can be accessed by telnet command using port number 7000 (default) and a user can change the port number using the 'set ttyport' command. It supports up to 4 clients and you can check the status or control using the 'console' command.

12.2 Terminal Service Mode

You can select the following mode using IDSW SW1-8. It also can be controlled and checked by the 'tty' command.

- telnet
- raw (conman)

13 Command Reference

You can access the iSCB CLI service using telnet client using port number 7000(default). iSCB provides Cisco style CLI service and you can complete the proper command using the <tab>, <up> and <down> keys.

```
[root@localhost ~]$ telnet 10.10.1.11 7000
Trying 10.10.1.11...
Connected to 10.10.1.11 (10.10.1.11).
Escape character is '^]'.
1:1:1-iSCB> ver
iSCB Ver 1.0 rc5 (Sep  8 2011)
ok
1:1:1-iSCB>
```

13.1 Command List

status	display Node, PSU and FAN status
power	power on/off/cycle/reset all or specific node
led	id-led on/off all or specific node
gpu	GPU enable/disable all or specific node
node	display information of specific node
sensor	display sensor status of specific node
pmnode	display node power status for powerman
pmled	display node id-led status for powerman
psu	display information of specific PSU
cfu	display information of specific CFU

fan	display CFU status or set fan duty ratio
console	display established cli status or kill connection
tty	display established remote tty console or kill connection
console	display or control the established cli service
identify	blink green/red LED to identify
set	display or set iSCB parameters
nvram	display or save nvram settings
reboot	reboot iSCB for applying setting or firmware updates
lasterr	display recent occurred error message or clear
log	display event logs for all, critical, warning and info severity
history	display command history of what was performed
help	display help message

13.2 Command Syntax

1:1:1-iSCB> command [args]

command	iSCB control or status command
[args]	optional parameters (all, portno, …)

13.3 ver

Syntax:

ver

Description:

Display firmware version and build date.

Parameters:

None

Example:

```
1:1:1-iSCB> ver
iSCB Ver 1.0 rc5 (Sep  5 2011)
ok
1:1:1-iSCB>
```

13.4 status

Syntax:

Status [node|psu|fan]

Description:

Display Node, PSU and CFU or entire status.

Parameters:

node
psu
fan

Example:

```
1:1:1-iSCB> status
iSCB Status
   Node:          03                          10
  Power:
  IDLED:
Console:
    BMC:           *                           *
Temp 'C:          29                          29

    PSU:          01          02          03          04      Total
  Power:          on          on
 Status:          ok          ok
   Temp:       25'C        25'C
    Fan:  3552,4928  3360,5440
    12V:          2A          0A                                 2A
  AC In:        114V        113V
   Watt:         42W         28W                               70W

    CFU:          01          02          03
 Status:          ok         bad          ok
   Fan1:    1550rpm        0rpm    1523rpm
   Fan2:    1548rpm        0rpm    1497rpm
   Duty:         60%         14%         60%
ok
1:1:1-iSCB>
```

13.5 power

Syntax:

power <on|off|cycle|reset> <node(s)> [interval]

Description:

Power on/off/cycle/reset the specific node(s). If the node is already turned on, there will be no action. When multiple nodes are given for power on/off/cycle operation, an interval value is required. <CR> key to abort the power operation.

Parameters:

on	Turn on the specific nodes. If node is already turned on, there will be no action.
off	Turn off the specific nodes. If node is already turned off, there will be no action.
cycle	Turn off, then turn on, the specific nodes.
reset	Reset specific nodes. No interval time required for multiple nodes.
node(s)	Specific node 1 through 16 or all: entire nodes

or
0xffff: Bit fields for multiple specific nodes
0x3: Node 1,2
0x5: Node 1,3
0xff00: Node 9-16

interval Interval time for power on operation of multiple nodes. Unit is 0.1 second. 0 means no interval.

Example:

```
1:1:1-iSCB> power on 1
ok
1:1:1-iSCB> power off 1
ok
1:1:1-iSCB> power on all 1
ok
1:1:1-iSCB> power off all 0
ok
```

13.6 led

Syntax:

led <on|off> <node(s)>

Description:

on/off ID-LED all or specific node.

Parameters:

on Turn on the specific nodes idled. If node is already turned on, there will be no action.

off Turn off the specific nodes idled. If node is already turned off, there will be no action.

node(s) Specific node 1 through 16
or
all: entire nodes
or
0xffff: Bit fields for multiple specific nodes
0x3: Node 1,2
0x5: Node 1,3
0xff00: Node 9-16

Example:

```
1:1:1-iSCB> led on 1
ok
1:1:1-iSCB> led off 1
ok
1:1:1-iSCB> led on all
ok
1:1:1-iSCB> led off all
```

```
ok
```

13.7 gpu

Syntax:

gpu <on|off> <node(s)>

Description:

Enable or disable gpu control all or specific node. When GPU is enabled, GPU power status is controlled by compute node's power. When a GBGPU node is inserted into slot, the default status is Enabled.

Parameters:

on Turn on the GPU expansion blade. If node is already turned on, there will be no action.

off Turn off the GPU expansion blade. If node is already turned off, there will be no action.

node(s) Specific node 1 through 16
or
all: entire nodes
or
0xffff: Bit fields for multiple specific nodes
0x3: Node 1,2
0x5: Node 1,3
0xff00: Node 9-16

Example:

```
1:1:1-iSCB> gpu on 1
ok
1:1:1-iSCB> gpu off 1
ok
1:1:1-iSCB> gpu on all
ok
1:1:1-iSCB> gpu off all
ok
```

13.8 pmnode

Syntax:

pmnode [node]

Description:

Display powerman style power status. n/a means that the node is not installed.

Parameters:

node Specific node 1 through 16
or
All Nodes

Example:

```
1:1:1-iSCB> pmnode
node01: on
node02: off
node03: n/a
node04: n/a
node05: off
node06: off
node07: off
node08: off
node09: off
node10: off
ok
```

13.9 node

Syntax:

node <node>

Description:

Display node's status.

Parameters:

node Specific node 1 through 16

Example:

```
1:1:1-iSCB> node 3
   slot: 3
present: on
  power: on
 id-led: off
console: 7003(idle)
prod-id: 004d
BMC ver: 0.16
BMC MAC: 00:1e:67:08:51:f6
 ipaddr: 10.10.1.20
netmask: 255.255.0.0
gateway: 10.10.1.254
sensors: 10
maxtemp: 43.0
ok
1:1:1-iSCB>
```

13.10 sensor

Syntax:

sensor <node>

Description:

Display temperature sensors on the node.

Parameters:

node Specific node 1 through 16

Example:

```
1:1:1-iSCB> sensor 3
 P0 Therm Margin: ok  ( -48.0 'C)
P0 DTS Therm Mgn: ok  ( -48.0 'C)
DIMM Thrm Mrgn 1: ok  ( -48.0 'C)
   BB Inlet Temp: ok  (  25.0 'C)
     BB BMC Temp: ok  (  33.0 'C)
    CPU0 VR Temp: ok  (  29.0 'C)
     IB QDR Temp: ok  (  34.0 'C)
        PCH Temp: ok  (  43.0 'C)
 Powerville Temp: ok  (  44.0 'C)
 P0 Therm Ctrl %: ok  (   0.0 unspecified)
ok
1:1:1-iSCB>
```

13.11 pmled

Syntax:

pmled node|all

Description:

Display powerman style ID-LED status.

Parameters:

node Specific node 1 through 16
or
all: All Nodes

Example:

```
1:1:1-iSCB> pmnode
node01: on
node02: off
node03: n/a
node04: n/a
node05: on
node06: on
node07: off
node08: off
node09: off
node10: off
```

```
ok
```

13.12 psu

Syntax:

psu <psuno>

Description:

Display status of specified psu.

Parameters:

psuno PSU number. If given psuno is 0, it will show the total of 12VA and AC Watt.

Example:

```
1:1:1-iSCB> psu 1
psu#: 1
powr: on
stat: ok
temp: 25
fan1: 3552
fan2: 4928
12VA: 2
ACIn: 113
Watt: 42
ok
1:1:1-iSCB> psu 0
12VA: 2
Watt: 226
ok
1:1:1-iSCB>
```

13.13 cfu

Syntax:

cfu <cfuno>

Description:

Display status of specified CFU.

Parameters:

psuno CFU number.

Example:

```
1:1:1-iSCB> cfu 1
cfu#: 1
stat: ok
fan1: 1555
fan2: 1547
duty: 60
```

```
ok
1:1:1-iSCB>
```

13.14 fan

Syntax:

fan [cfuno duty]

Description:

Display or set CFU FAN speed duty. If there is no access of iSCB from outside over 90 seconds, iSCB will change to full speed automatically.

Parameters:

cfuno	CFU number.
high	Set full Speed. (100%)
normal	Set normal Speed. (85%)
low	Set low Speed. (60%)

Example:

```
1:1:1-iSCB> fan
    CFU:         01         02         03
 Status:         ok        bad         ok
   Fan1:   1552rpm       0rpm   1522rpm
   Fan2:   1540rpm       0rpm   1495rpm
   Duty:        60%        14%        60%
ok
1:1:1-iSCB> fan all high
ok
1:1:1-iSCB> fan
    CFU:         01         02         03
 Status:         ok        bad         ok
   Fan1:   1552rpm       0rpm   1522rpm
   Fan2:   1540rpm       0rpm   1495rpm
   Duty:       100%       100%       100%
ok
1:1:1-iSCB>
```

13.15 console

Syntax:

console [kill number]

Description:

Display established CLI connection or kill the connection. * mark means current connection.

Parameters:

kill number	Kill the specified CLI connection number.

Example:

```
1:1:1-iSCB> console
 No Auth Remote                    Timeout         In        Out
*1  pass 10.10.1.252,60372               0         19        989
 2  pass 10.10.1.252,34593               0         73       3905
ok
1:1:1-iSCB> console kill 2
ok
1:1:1-iSCB> console
 No Auth Remote                    Timeout         In        Out
*1  pass 10.10.1.252,60372               0         44       1241
ok
1:1:1-iSCB>
```

13.16 tty

Syntax:

tty [kill channel]

Description:

Display all Remote Console (TTY) status or kill the connection.

Parameters:

kill channel Kill the specified Remote Console connection number.

Example:

```
1:1:1-iSCB> tty
Port  Device        Remote       Timeout     TCPin    TCPout     Devin    Devout
7001  /dev/ttyUSB0  10.10.1.1           0         6         5         5         6
7002  /dev/ttyUSB1  10.10.1.1           0         6         0         0         6
7003  /dev/ttyUSB2  10.10.1.1           0         6         5         5         6
7004  /dev/ttyUSB3  10.10.1.1           0         6         5         5         6
7005  /dev/ttyUSB4  10.10.1.1           0         6         5         5         6
7006  /dev/ttyUSB5  10.10.1.1           0       161    101269    101269       161
7007  /dev/ttyUSB6  10.10.1.1           0         6     74634     74634         6
7008  /dev/ttyUSB7  10.10.1.1           0       425    128353    128353       425
7009  /dev/ttyUSB8  10.10.1.1           0         6         8         8         6
7010  /dev/ttyUSB9  0.0.0.0             0         0         0         0         0
ok
1:1:1-iSCB> tty kill 7009
ok
1:1:1-iSCB>
```

13.17 set

Syntax:

set <param> [value]

Description:

Display or set iSCB parameters. Most updated parameters are applied and permanently stored in NVRAM area by the ‘nvram save’ command.

Parameters:

ip	iSCB IP Address (10.10.1.11). **
nm	iSCB Netmask (255.255.0.0). **
gw	iSCB Gateway (10.10.1.254). **
ns	iSCB NameServer. **
loglevel	Event logging level. One of ‘critical’, ‘warning’ and ‘info’. *
debug	debug option.
community	SNMP Trap community name of the event notification. *
receiver	SNMP Trap receiver IP address for the event notification. **
email	administrators email address for the event notification. **
mailserver	mail server address for the email delivery. **
mailuser	user account of the mail server if required. **
mailpass	user password of the mail server if required. **
port	TCP port number for the CLI and Terminal Service. Default is 7000. **
baud	UART baudrate of the Terminal Service. Default is 115200. **
time	System time. Setting format is [[[[[YY]YY]MM]DD]hh]mm[.ss]
name	Subrack alias name. Default is Type-CID-SID (ie., SR5110-1-1). *

* Apply immediately. ‘nvram save’ required to make permanent
** ‘nvram save’ and ‘reboot’ required to apply on iSCB operations

Example:

```
1:1:1-iSCB> set ip
NVRAM  : 10.10.1.9
Current: 10.10.1.9
ok
1:1:1-iSCB> set ip 10.10.1.11
ok
1:1:1-iSCB> set ip
NVRAM  : 10.10.1.11
Current: 10.10.1.9
ok
1:1:1-iSCB> nvram save
ok
1:1:1-iSCB> reboot
ok
1:1:1-iSCB> Connection closed by foreign host.
```

13.18 nvram

Syntax:

nvram [save|clear]

Description:

Display or update stored configuration parameters.

Cabinet ID
Subrack ID
IP Address
NetMask

Gateway
Password
Interval
Checksum

Parameters:

save Store current configuration parameters into the NVRAM.
clear Clear NVRAM data then set factory default.

Example:

```
1:1:1-iSCB> nvram
NVRAM Ver.  : 1.0
Subrack Name: SR5110-1-1
Cabinet ID  : 1
Subrack ID  : 1
SCB1 IP Addr: 10.10.1.11/255.255.0.0
SCB1 Gateway: 10.10.1.254
SCB2 IP Addr: 10.10.2.11/255.255.0.0
SCB2 Gateway: 10.10.2.254
Service Port: 7000
TTY speed   : 115200
SNMP comm   : public
Receiver1   : 0.0.0.0
Receiver2   : 0.0.0.0
E-Mail      :
Mail Server :
Mail UserID :
Mail Passwd :
Password    :
Log Level   : info
Polling intv: 180
ok
1:1:1-iSCB>
```

13.19 reboot

Syntax:

reboot

Description:

Restart iSCB for applying settings or to update firmware. The embedded linux of iSCB will do a graceful shutdown and then boot up with the new settings.

Parameters:

None

Example:

```
1:1:1-iSCB> reboot
ok
```

```
1:1:1-iSCB> Connection closed by foreign host.
```

13.20 lasterr

Syntax:

lasterr [clear]

Description:

Display recently occurred event log or clear the last log.

Parameters:

clear Clear the last log and reset the RED LED on the iSCB front panel.

Example:

```
1:1:1-iSCB> lasterr
ALERT: CFU02 failed
ok
1:1:1-iSCB> lasterr clear
ok
1:1:1-iSCB> lasterr
ok
1:1:1-iSCB>
```

13.21 log

Syntax:

log [level]

Description:

Display event logs by given log level.

Parameters:

level There are three log levels: critical, warning and info.

critical : display err, crit, alert and emerg level even logs

warning : display warning and higher level event logs

info : display notice, info and higher logs

Example:

```
1:1:1-iSCB> log critical
5: Jan  3 23:00:01 iscb daemon.alert iscbd: CFU02 failed
11: Jan  3 23:00:54 iscb daemon.crit iscbd: slot03: P0 Therm Margin: n/a
(   0.0)
12: Jan  3 23:00:54 iscb daemon.crit iscbd: slot03: P0 DTS Therm Mgn: n/a
(   0.0)
13: Jan  3 23:00:54 iscb daemon.crit iscbd: slot03: DIMM Thrm Mrgn 1: n/a
(   0.0)
ok
1:1:1-iSCB>
```

13.22 identify

Syntax:

identify

Description:

Blink ACT/FAIL LED for 10 seconds.

13.23 help

Syntax:

help [command]

Description:

Display available commands and usage

Parameters:

command insert an appropriate command that you wish to learn about

Example:

```
1:1:1-iSCB> help power
power on [STRING:nodes][INT:interval]
; power on specific node(s) or all

power off [STRING:nodes][INT:interval]
; power off specific node(s) or all

power cycle [STRING:nodes][INT:interval]
; power cycle specific node(s) or all

power reset <STRING:nodes>
; reset specific node(s) or all

ok
1:1:1-iSCB>
```

14 Remote Update

14.1 Send firmware image to iSCB

iSCB is ready to receive a firmware image through tftp protocol so that you can send iscb the firmware image using following command on a system.

```
[root@localhost ~]# tftp 10.10.1.11 –c put iscb-xx.bin.gz
```

Please wait for the action to be completed. It may take a minute or two to complete.

14.2 Update firmware

Perform ‘reboot’ command (NOT reset) to update iscb firmware during reboot sequence.

```
[root@localhost ~]# telnet 10.10.1.11 7000
1:1:1-iSCB> reboot
```

iSCB will automatically startup the flashing sequence and it will take about two minutes and rebooting twice until the flashing sequence is complete. DO NOT RESET or INTERRUPT during the flashing sequence.
Wait for the updating complete message. If not successful, try to send the image again after resetting iSCB.

15 Lookup the iSCB internal settings

If you want to get more detailed iSCB internal settings, you can get the information via tftp command as follows:

```
[root@localhost tmp]# tftp 10.10.1.11 -c get /tmp/iscb.settings
[root@localhost tmp]# cat iscb.settings
ISCB_UNITID=1
ISCB_DIAG_MODE=0
ISCB_DEBUG_MODE=0
ISCB_MODE_SW3=0
ISCB_MODE_SW4=0
ISCB_IDSW_CABINETID=1
ISCB_IDSW_SUBRACKID=1
ISCB_IDSW_RACKTYPE=SR5110
ISCB_IDSW_GBGPU=0
ISCB_IDSW_CFUCONTROL=1
ISCB_IDSW_CONMAN=0
ISCB_IDSW_IPSRC=static
ISCB_NVRAM_VERSION=1.0
ISCB_NAME=SR5110-1-1
ISCB_CABINETID=1
ISCB_SUBRACKID=1
ISCB_INTERVAL=10
ISCB_CYCLEDELAY=10
ISCB_FANSPEED=85
ISCB_POLLING=180
ISCB_IPADDR1=10.10.1.11
ISCB_NETMASK1=255.255.0.0
ISCB_GATEWAY1=10.10.1.254
ISCB_IPADDR2=10.10.2.11
ISCB_NETMASK2=255.255.0.0
ISCB_GATEWAY2=10.10.2.254
ISCB_NAMESERVER=0.0.0.0
ISCB_COMMUNITY=public
ISCB_RECEIVER1=0.0.0.0
ISCB_RECEIVER2=0.0.0.0
ISCB_LOGLEVEL=info
```

```
ISCB_EMAIL=
ISCB_MAILSERVER=
ISCB_MAILUSER=
ISCB_MAILPASS=
ISCB_TTYPORT=7000
ISCB_TTYSPEED=115200
ISCB_PASSWORD=''
```

16 Powerman/Conman support

iSCB supports powerman and conman control interface.

16.1 Powerman

You can get the GB2 device file from running iSCB.

```
[root@localhost tmp]# tftp 10.10.1.11 -c get /tmp/appro-gb2.dev
```

There are five types of GB-2 device types:

- sr5110 : Standard 10 Nodes
- sr5110_gpu : 5 Nodes with 5 GBGPU expansions
- sr8116 : Standard 16 Nodes
- sr8116_gpu : 8 Nodes with 8 GBGPU expansions
- sr8104 : 2U 4 Nodes

Following are examples of the /etc/powerman/powerman.conf

```
[root@localhost ~]# cat /etc/powerman/powerman.conf
include "/etc/powerman/appro-gb2.dev"

tcpwrappers yes

device "gb1" "sr5110"          "10.10.1.11:7000"
device "gb2" "sr5110_gpu"      "10.10.1.12:7000"
device "gb3" "sr8116"          "10.10.1.13:7000"

node "cn01" "gb1" "01"
node "cn02" "gb1" "02"
node "cn03" "gb1" "03"
node "cn04" "gb1" "04"

node "gn01" "gb2" "01"
node "gn02" "gb2" "02"

node "cn05" "gb3" "01"
node "cn06" "gb3" "02"
node "cn07" "gb3" "03"
node "cn08" "gb3" "04"
```

16.2 Conman

Following are examples of the /etc/conman.conf

```
[root@localhost ~]# cat /etc/conman.conf
SERVER keepalive=ON
SERVER logdir="/var/log/conman"
SERVER logfile="/var/log/conman.log"
SERVER loopback=ON
SERVER pidfile="/var/run/conman.pid"
SERVER resetcmd="powerman -0 %N; sleep 3; powerman -1 %N"
SERVER tcpwrappers=ON
SERVER timestamp=1h
GLOBAL seropts="115200,8n1"
GLOBAL log="console.%N"
GLOBAL logopts="sanitize,timestamp"

CONSOLE name="cn01" dev="10.10.1.11:7001"
CONSOLE name="cn02" dev="10.10.1.11:7002"
CONSOLE name="cn03" dev="10.10.1.11:7003"
CONSOLE name="cn04" dev="10.10.1.11:7004"

CONSOLE name="gn01" dev="10.10.1.12:7001"
CONSOLE name="gn02" dev="10.10.1.12:7002"

CONSOLE name="cn05" dev="10.10.1.13:7001"
CONSOLE name="cn06" dev="10.10.1.13:7002"
CONSOLE name="cn07" dev="10.10.1.13:7003"
CONSOLE name="cn08" dev="10.10.1.13:7004"
```

- End of Document