

OM10K LUA CLI (Command Line Interface) Overview

The OM10K acts as a layer 2 Ethernet switch without having to access the CLI commands.

If advanced features such as setting up LANs, Monitor Ports, or modifying RJ45/SFP port parameters from the default are required, then you will need to determine the relevant CLI commands and update the OM10K switch configuration file.

This document covers the following:

- CLI (Command Line Interface) overview
- Modifying the OM10K configuration file so the desired CLI commands are executed at startup

Relevant Application Notes:

These following application notes are available in the *DOWNLOADS* sections of:

www.luxcom.com/product/om10k-8/ and www.luxcom.com/product/om10k-24/

App.1 OM10K Management Interface Access.pdf gives instructions on accessing the OM10K management interface.

App.3 OM10K Port Monitor Setup.pdf gives examples of setting up Span/Tap Monitor ports using the CLI commands.

App.4 OM10K VLAN Setup.pdf gives examples of setting up Port and Tag based VLANs using CLI commands.

General

- CLI commands are executed on the Network Management processor.
- Entering “?” at the *Command#* prompt lists the CLI commands.
- Entering a CLI command at the *Command#* prompt executes that command.
- One can paste from the paste buffer a series of CLI commands at the *Command#* and they will execute. You will see any syntax errors by doing this.
- The CLI configuration you entered will be lost if the OM10K loses power.
- To retain the CLI configuration, the desired CLI commands must be added to the configuration file; they will be automatically loaded at power-on and after a system reboot

Updating OM10K Configuration:

Using the USB:

The default configuration file is stored in the Linux file system. It cannot be directly edited from the CLI. In running-config mode, it is possible to save and load configuration files to/from the /cmdFS directory. This directory is part of the file system overwritten by updates using mtdburn.

When you access the Network Management processor you see
At this point enter

OM10K Login:
root

Logging in as *root* gives authority to modify the config.txt file.

You should see:

~ #

To access the CLI interface enter

telnet 127.0.0.1 12345

To exit from the CLI interface to back to Linux enter

CLExit

You should see:

~ #

The config file is found here:

/usr/local/etc/config.txt

It is automatically loaded at power on and after a system reboot.

Using Ethernet Telnet:

The CLI can be accessed directly at port 12345. The Linux system can be accessed at port 23. For example, telnet to 192.168.0.58 12345 to access the CLI. Otherwise, Ethernet access is identical to that using the USB console.

Editing the config.txt file:

The busybox vi editor shipped with the product is the easiest method to modify this configuration.

Documentation for busybox applications is here: <<https://busybox.net/BusyBox.html>>. Most busybox applications are provided. Most accept the “-h” help option.

Default config.txt file:

The default config.txt contains the following line:

```
cpssInitSystem 19,2
```

This line initializes the switch. It must be the first line in the configuration file and it must not be removed.

Lua CLI (Command Line Interface)

The Marvell® Lua Command Line Interface (CLI) is based on following components:

- Lua 5.1: <<http://www.lua.org/ftp/lua-5.1.tar.gz>>
- CLI engine
- mainLuaWrappers – Glue between Lua and CPSS
- CLI scripts – Scripts for Pretera switch management

The Lua CLI scripts are located in the /cmdFS directory. It is possible to add, edit or remove scripts.

Lua CLI Command Modes

The Lua CLI runs in various command modes. The modes are hierarchial. Lua starts at the top/root level in “exec” mode.

Each mode has its own set of commands. Entering a “?” will display a list of the available commands in the current mode. A “?” entered after entering a command or argument will show the next expected argument or value. For a single non-cascaded switch, the default device numer is 0.

The following modes are available at the top level:

- Exec mode – contains cpss-api, init, shell-execute and show commands
- Examples mode – contains sample CLI commands
- Configure mode – contains global and interface commands
- Debug mode – contains commands to trace registers or CPSS-APIs.
- Diagnostic mode
- Traffic mode – contains commands to generate different types of traffic

Lua CLI Command Conventions

In a command line, “[]” indicates an optional entry. “{ }” indicates an obligatory parameter. For example, flowcontrol {auto|on|off} means that for the flowcontrol command either auto, on or off must be entered. “<Key>” specifies a specific key on the keyboard (e.g. <Enter>).

Lua CLI Commands

The CLI commands are numerous and self-documenting. Here are some samples:

Console# ?

clear	Reset functions
cls	clear screen
configure	Enter configuration mode
cpss-api	
cpssInitSystem	Init system
cpssinitsystem	Init system
debug-mode	Exit from the EXEC to debug mode
delete	
diagnostics	Switch to the diagnostics mode
do	Allows to execute command from the USER mode
echo	echo words from command line
examples	Switch to examples of tests
no	Negate command
reload	Reload all stack units or specific unit
shell-execute	allow to execute any 'C' function with it's parameters
shell-execute-in-parallel any	(in parallel - non blocking) allow to execute 'C' function with it's parameters
show	Display running system information
traffic	Switch to traffic mode
exit	Switch to the parent mode
end	Switch to the root mode
CLiexit	Exit CLI

Console# end ?

<CR> Switch to the root mode

Console# configure ?

<CR> Enter configuration mode

Console# show ?

access-list	Show a device's PCL-ID access-list
bpe-802-1-br	
bridge	Bridging information
buffer-management	The dump of all Buffer Management related info
capabilities	Displaing of system capabilities
capwap	Show capwap info configuration

ccfc	The dump of all CCFC related information
counters	Display counters
data-integrity-events	Show data-integrity-events
dce	Show dce data center environment
eee	show EEE - Port's IEEE 802.3az Energy Efficient Ethernet (EEE) related configurations and status
flow-control	The dump of all Flow Control related information
history	Display the session command history
hwDevice	Show hwDevice offset
interfaces	interfaces
ip	IP Information
led	
mac	MAC configuration
mpls	Show mpls info configuration
mpls-transit	Show mpls transit info configuration
nat44	Show NAT44 info configuration
nat66	Show NAT66 info configuration
pcl	Show a device's pcl rule
pfc	The dump of all PFC related information
pip	show pip related configurations and drop counters
pizza-arbiter-dump	
policy	Shows a policy rule list
policy-billing	show policy-billing entry
policy-meter	
policy-meter-db	show policy-meter entry from CPU memory
policy-meter-envelope-hw	show policy-meter envelope from device
policy-meter-hw	show policy-meter entry from device
policy-mng-conters	reset policy-management counter set
port	Show port configuration
port-map	Show port-map
qos	Show qos commands
running-config	Current operating configuration
running-parallel-tasks	Parallel running tasks
simulation	
spbm	
system	Show system information
tail-drop	The dump of all tail drop related information
tpid	Show tpid device
traffic	
tti	show tti configurations
version	System hardware and software version

```

vlan          vlan
vpn           Show vpn info
    
```

Console# show interfaces ?

```

ap
configuration      Interface configuration
led                show port related LED information
mac                Interface(s) MIB Counters
port-channel       Port-channel member ports information
serdes
speed              Interfaces speed
status             Ports information
switchport        switchport
tpid               Show tpid device
    
```

Console# show interfaces status ?

```

all                Apply to all ports of specific device or all available
                  devices
ethernet           Ethernet interface to configure
port-channel       Valid Port-Channel interface
    
```

Console# show interfaces status all

Dev/Port	Mode	Link	Speed	Duplex	Loopback Mode
0/0	QSGMII	Up	1G	Full	None
0/1	QSGMII	Down	10	Half	N/A
0/2	QSGMII	Down	10	Half	N/A
0/3	QSGMII	Down	10	Half	N/A
0/4	QSGMII	Down	10	Half	None
0/5	QSGMII	Down	10	Half	N/A
0/6	QSGMII	Down	10	Half	N/A
0/7	QSGMII	Down	10	Half	N/A
0/8	SGMII	Down	10	Half	None
0/9	SGMII	Down	10	Half	None
0/10	SGMII	Down	10	Half	None
0/11	SGMII	Down	10	Half	None
0/12	SGMII	Down	10	Half	None

0/13	SGMII	Down	10	Half	None
0/14	SGMII	Down	10	Half	None
0/15	SGMII	Down	10	Half	None
0/16	SGMII	Down	10	Half	None
0/17	SGMII	Down	10	Half	None
0/18	SGMII	Down	10	Half	None
0/19	SGMII	Down	10	Half	None
0/20	SGMII	Down	10	Half	None
0/21	SGMII	Down	10	Half	None
0/22	SGMII	Down	10	Half	None
0/23	SGMII	Down	10	Half	None
0/24	SR_LR	Down	10G	Full	None
0/25	SR_LR	Down	10G	Full	None
0/26	SR_LR	Down	10G	Full	None
0/27	SR_LR	Down	10G	Full	None

Console# configure**Console(config)# interface ?**

```

eport                Specific dev/ePort
ethernet             Specific ethernet dev/port
port-channel        Valid Port-Channel interface
range               Select range of interfaces to configure
vlan                Configure an IEEE 802.1 VLAN

```

Console(config)# interface ethernet ?

```

Ethernet port
  0/0                Default dev/port

```

Console(config)# interface ethernet 0/0**Console(config-if)# ?**

```

LED                Sets LED port slot number in the LED's chain
ap-introp
ap-port            Enable AP process on port
auto-detect        Auto-detect port's actual mode and speed
auto-negotiation
auto-tune          Set auto tune type
back-pressure      Enable back-pressure
bpe-802-1-br       BPE system (bridge port extender) according to IEEE-
                   802.1BR standard
bridge            Bridge configuration subcommand

```

capwap	Configure capwap
cascade	Setting of the port to be DSA-tagged
channel-group	Configure channel-group
counters	Configure counters
crc	CRC subcommads
dce	DCE subcommands
echo	echo words from command line
eee	Support EEE - IEEE 802.3az Energy Efficient Ethernet (EEE)
flow-control	Configure flow-control mode
force	force link state
interface	Interface configuring
ip	IP configuration commands
jumbo-frame	Enable jumbo frames for the device
led	
loopback	Configure loopback
mpls	Set MPLS configuration on port
mpls-transit	Set MPLS transit configuration on port
negotiation	Enable auto negotiation
nmi	
no	Negate command
phy	Support PHY features on this interface.
pip	Support for Pre-Ingress Prioritization (PIP)
policy-id	Sets the port to a policy rule
port	Perform port configuration
port-mac	Port MAC address subcommands
prbs	Prbs
qos	Configure QoS
rmon	Determine what kind of traffic the histogram mac
counters	will count
serdes	
service-acl	Apply an ACL to particular interface
shutdown	Shutdown the selected interface
spbm	
speed	Set the transmit and receive speeds
storm-control	Enabled Ingress Port Storm Rate Limit Enforcement for current port
switchport	Configure switch port definition in vlan
tail-drop	Configures tail drop limits on Port
tail-drop-queue	Configures tail drop limits on Port
tpid	

traffic-generator	Handle traffic generator
ttn	TTI configurations
uni	create UNI interface
vss	VSS: state interface related info.
wred	Enables WRED for TC/DP limits
do	Allows to execute command from the USER mode
exit	Switch to the parent mode
end	Switch to the root mode
CLIexit	Exit CLI

Console(config-if)# speed ?

Port speed

10	Force operation at 10Mbps
100	Force operation at 100Mbps
1000	Force operation at 1Gbps
10000	Force operation at 10Gbps
100000	Force operation at 100Gbps
11800	Force operation at 11.8Gbps
12000	Force operation at 12Gbps
13600	Force operation at 13.6Gbps
140000	Force operation at 140Gbps
15000	Force operation at 15Gbps
16000	Force operation at 16Gbps
20000	Force operation at 20Gbps
2500	Force operation at 2.5Gbps
40000	Force operation at 40Gbps
47200	Force operation at 47.2Gbps
5000	Force operation at 5Gbps
50000	Force operation at 50Gbps
75000	Force operation at 75Gbps

Console(config-if)# speed 1000 ?

mode	interface mode
------	----------------

Console(config-if)# speed 1000 mode ?

Port interface mode

1000Base_X	1000 Base X interface mode
100Base_FX	100 Base-FX interface mode
CHGL	CHGL interface mode
GMII	GMII interface mode
HGL	HGL interface mode
HX	Half X interface mode
ILKN12	ILKN12 interface mode

ILKN16	ILKN16 interface mode
ILKN24	ILKN24 interface mode
ILKN4	ILKN4 interface mode
ILKN8	ILKN8 interface mode
KR	KR interface mode
LOCAL_XGMII	XGMII interface mode
MGMII	MGMII interface mode
MII	MII interface mode
MII_PHY	MII PHY interface mode
NO_SERDES_PORT	generic LOCAL_XGMII
QSGMII	Quad SGMII interface mode
QX	Quarter X interface mode
REDUCED_10BIT	Reduced 10-bit interface mode
REDUCED_GMII	Reduced GMII interface mode
RXAUI	RXAUI interface mode
SGMII	SGMII interface mode
SR_LR	Short/Long reach interface mode
XAUI	XGMII interface mode (alias)
XHGS	XHGS interface mode
XHGS_SR	XHGS SR interface mode
XLG	XLG(40G) interface mode

```
Console(config-if)# speed 1000 mode QSGMII
```

```
Console(config-if)# end
```

```
Console#
```

The “cpss-api” command provides access to Marvell's Core Prestera® Software Suite. This enables access to various device features that are not implemented as Lua scripts. For example,

```
Console# cpss-api ?
```

call	Call CPSS function
man	Show CPSS API function info
mode	Set mode for cpss-api call output
search	Call CPSS function

```
Console# cpss-api search temperature
```

```
cpssDxChDiagDeviceTemperatureGet
```

```
cpssDxChDiagDeviceTemperatureSensorsSelectGet
```

```
cpssDxChDiagDeviceTemperatureSensorsSelectSet
```

```
cpssDxChDiagDeviceTemperatureThresholdGet
```

```
cpssDxChDiagDeviceTemperatureThresholdSet
```

```
cpssDxChPortDiagTemperatureGet
```

```
Console# cpss-api call cpssDxChDiagDeviceTemperatureGet devNum 0
```

```
result=GT_OK = Operation succeeded
values={
  temperature=67
}
```

Console# cpss-api man cpssDxChDiagDeviceTemperatureGet

```
/******
***
* cpssDxChDiagDeviceTemperatureGet
*
* DESCRIPTION:
*     Gets the PP temperature.
*
* . . .
*
* INPUTS:
*     devNum          - device number
*
* OUTPUTS:
*     temperaturePtr - (pointer to) temperature in Celsius degrees
*                    (can be negative)
*
* RETURNS :
*     GT_OK           - on success
*     GT_BAD_PARAM   - wrong parameter
*     GT_BAD_PTR     - null pointer
*     GT_NOT_APPLICABLE_DEVICE - on not applicable device
*     GT_HW_ERROR    - on hardware error
*
* COMMENTS:
*
*****
**/
```