

A solid blue vertical bar with rounded ends, positioned to the left of the text.

GPON OLT (access and aggregation node)

MA4000-PX

CLI command reference guide
Firmware version 3.34.1

Contents

1	Introduction	4
2	List of changes.....	5
3	Command line usage rules.....	10
4	Command system structure.....	12
5	Interface name system.....	17
6	User interface commands.....	19
7	Network management parameters configuration	24
8	System time configuration	32
9	System user management	35
10	Alarm and event log management.....	41
11	Crate management	82
12	Firmware and configuration management.....	94
13	Device operation debugging	114
14	Creating a configuration backup (BACKUP CONFIG).....	128
15	Remote access configuration	130
16	PLC line board traffic filtering management.....	143
17	Interface mirroring	148
18	Ethernet interface management	151
19	GPON interface management	169
20	ONT configuration management	181
21	Configuring VLAN	207
22	PP4X module stacking configuration.....	215
23	Simple network management protocol (SNMP) configuration	217
24	Spanning Tree protocol configuration.....	223
25	DualHoming protocol configuration	235
26	Multicast management (IGMP, MLD).....	239
27	Channel aggregation groups.....	280
28	MAC table management commands.....	287
29	QoS configuration	291
30	Port isolation group configuration.....	301
31	Selective Q-in-Q feature configuration.....	309
32	DHCP client configuration for management interface.....	314
33	Configuring LLDP.....	317
34	SNTP configuration	325
35	System log configuration	327
36	Syslog filters configuration. LOGFILTER command mode	335

37 ARP configuration (ARP Inspection).....	338
38 VLAN profile configuration (Profile VLAN).....	343
39 DHCP relay agent profile configuration (PROFILE DHCP_RA)	348
40 DHCP IPv6 relay agent profile configuration (PROFILE DHCPv6_RA).....	355
41 Intermediate PPPoE agent profile configuration (PROFILEPPPOE_IA).....	363
42 Address table profile configuration (PROFILE ADDRESS TABLE)	370
43 PLC8 board ONT configuration profile management. cross-connect profile	374
44 PLC8 board ONT configuration profile management. DBA profile.....	384
45 PLC8 board ONT configuration profile management. MANAGEMENT profile.....	390
46 PLC8 board ONT configuration profile management. PORTS profile	395
47 PLC8 board ONT configuration profile management. SCRIPTING profile	423
48 PLC8 board ONT configuration profile management. SHAPING profile.....	426
49 PLC8 board configuration templates ('TEMPLATE')	440
50 Management modules monitoring	450
51 ONT/OLT operational management	453
52 Static Routing.....	464
53 DHCP Relay	467
54 IP Source Guard configuration.....	468
55 PLC8 modules GPON configuration settings.....	472

1 Introduction

- [Abstract](#)
- [Target Audience](#)
- [Notes and warnings](#)

Abstract

This manual describes CLI commands for the administrator of the MA4000-PX subscriber access/aggregation node (hereinafter referred to as the device).

Command Line Interface (CLI) allows to perform the device management and monitor its operation and status. You will require the PC application supporting Telnet or SSH protocol operation or direct connection via the console port (e.g. HyperTerminal).

Target Audience

CLI command reference guide is dedicated to the technical staff that performs the MA4000-PX access node configuration and monitoring using the CLI interface. Qualified technical personnel should be familiar with the operation basics of TCP/IP & UDP/IP protocol stacks and Ethernet and GPON networks design concepts.

Notes and warnings

✔ Notes contain important information, tips or recommendations on device operation and setup.

⚠ Warnings are used to inform the user about harmful situations for the device and the user alike, which could cause malfunction or data loss.

2 List of changes

Document version	Issue date	Revisions
Version 1.8	09.02.2022	Synchronization with firmware version 3.34.1
Version 1.7	14.12.2021	Synchronization with firmware version 3.34.0 Changes in sections: PLC8 board ONT configuration profile management. SHAPING profile – added 'storm-control'
Version 1.6	05.07.2021	Changes in sections: Device operation debugging ONT configuration management PLC8 board configuration templates ('TEMPLATE') PLC8 modules GPON configuration settings
Version 1.5	13.04.2020	Synchronization with firmware version 3.30.0
Version 1.4	08.10.2019	Synchronization with firmware version 3.26.3
Version 1.3	22.12.2017	Chapters added: 11.16 firmware ont auto update edit 19.31 show interface ont <PONSERIAL> services utilization 53.5 ip source-guard ignore-vlan Removed sections: 25.34 show slot <SLOT> gpon olt igmp

Document version	Issue date	Revisions
Version 1.2	01.02.2017	Changes in sections: 8.2 show users config: show users renamed in show users config 11.11 copy: license file upload option has been added 11.26 show running-config: added parameter "ipv6" 14.6 aaa authentication login: added parameter radius 15.6 slot <SLOT> access-list filter: added parameters ipv6-da and ipv6-sa, ipv4-protocol renamed in ip-protocol 17.11 shaper: changed <LIMIT> parameter value 17.12 cnterset: added port type support: plc-front-port, plc-pon-port, plc-slot-port. 19.15 service custom: removed unused value for parameters <VLAN> and <COS> 25.67 show interface gpon-port <GPON-PORT> igmp groups: show slot <SLOT> gpon olt igmp <port> renamed in interface gpon-port <GPON-PORT> igmp groups 38.11 show interface gpon-port <GPON-PORT> dhcp sessions: show ip dhcp snooping table renamed in show interface gpon-port <GPON-PORT> dhcp sessions 38.6 overwrite-option82: added parameter <TEXT FORMAT> 40.6 format: added parameters PONSERIAL and DESCR to form PPPoE IA option 82 45.7 igmp query last-member: igmp query interval last-member renamed in igmp query last-member 45.8 igmp query response: igmp query interval response renamed in igmp query response Chapters added: 8.3 show users who 9.3 show alarm active 11.13 cli max-sessions 11.14 cli display 11.15 show cli 11.21 license set 11.31 show license 14.14-14.17 Added commands to configure radius-server: radius-server timeout, radius-server key, radius-server encrypted key, radius-server host

Document version	Issue date	Revisions
		1.7 show slot <SLOT> ctrset 19.16 service selective-tunnel 19.30 show interface ont <PONSERIAL> connected 22.4 ip snmp agent engine id generate 22.6 ip snmp agent transport 25.24 ip igmp snooping static 25.34 show slot <SLOT> gpon olt igmp 25.35 ipv6 mld snooping enable 25.36 ipv6 igmp snooping pp4x enable 25.37 ipv6 mld snooping slot enable 25.38 ipv6 mld pp4x query-interval 25.39 ipv6 mld slot query-interval 25.40 ipv6 mld unregistered ip4-mc 25.41 ipv6 mld pp4x query-response-interval 25.42 ipv6 mld slot query-response-interval 25.43 ipv6 mld pp4x last-member-query-interval 25.44 ipv6 mld slot last-member-query-interval 25.45 ipv6 mld pp4x robustness 25.46 ipv6 mld slot <ID> robustness 25.47 ipv6 mld snooping mrouter add 25.48 ipv6 mld snooping mrouter del 25.49 ipv6 mld snooping mrouter learning 25.50 ipv6 mld snooping querier enable 25.51 ipv6 mld snooping querier fast-leave 25.52 ipv6 mld version 25.53 ipv6 mld proxy report enable 25.54 ipv6 mld proxy report pp4x enable 25.55 ipv6 mld proxy report slot enable

Document version	Issue date	Revisions
		25.56 ipv6 mld proxy report range 25.57 ipv6 mld proxy report pp4x range 25.58 ipv6 mld proxy report slot range 25.59 show ipv6 mld snooping groups vlan 25.60 show ipv6 mld snooping vlan config 25.61 show ipv6 mld snooping vlan hosts 25.62 show ipv6 mld snooping vlan mrouter 25.63 show ipv6 mld proxy report 25.64 show slot < SLOT > ipv6 mld snooping groups 25.65 show slot < SLOT > ipv6 mld snooping vlan config 25.66 show slot < SLOT > ipv6 mld proxy report 38.9 trusted server 39 Configuring DHCP IPv6 (PROFILE DHCPv6_RA) Relay agent profile: Added DHCPv6_RA profile and all commands to edit it. 40.9 show interface gpon-port < GPON-PORT > pppoe sessions 45.13 mld immediate-leave 45.14 mld multicast dynamic-entry 45.15 mld query interval 45.16 mld query last-member 45.17 mld query response 45.18 mld mode 45.19 mld querier 45.20 mld robustness 45.21 mld version 51.4 show ip route 52. DHCP Relay: DHCP Relaying feature configuration commands added Removed sections: 11.10 save

Document version	Issue date	Revisions
Version 1.1	23.12.2015	Revisions: 19.14 servise custom
Version 1.0	04.12.2015	First issue
Firmware version	3.30.0	

3 Command line usage rules

To simplify the use of the command line, the interface supports automatic command completion. This function is activated when the command is incomplete and the <Tab> character is entered.

Another function that helps to use the command line – context help. At any stage of entering a command, you can get a prompt about the following command elements by entering <?> character.

For the convenience of managing the device via a command line, the **do** command is used, which allows you to execute global level commands (ROOT) when you are at other levels of the command interface.

To simplify the commands, the whole command system has a hierarchical structure. There are special branch commands for transition between levels of the hierarchy. This allows to use brief commands on each level. To designate a current level where a user is located, the system prompt string changes dynamically.

For example,

```
ma4000# configure terminal      /switch to the device configuration mode
```

```
ma4000(config)#
```

```
ma4000(config)# exit          /return to the highest command system layer
```

```
ma4000#
```

For the ease of command line use, shortcut keys listed in the [Table 2.1](#) are supported.

Table 2.1 – Description of CLI shortcut keys

Shortcut key	Description
Ctrl+X	Command execution interruption
Ctrl+D	In the nested section, return to the previous section ('exit' command); in the root section, exit the CLI ('logout' command).
Ctrl+Z ¹	Go to the root section
Ctrl+A	Transition to the beginning of line
Ctrl+E	Transition to the end of line
Ctrl+U	Removal of characters to the left of a cursor
Ctrl+K	Removal of characters to the right of a cursor
Ctrl+C	Clear the line
Ctrl+W	Remove a word
Ctrl+B ¹	Transition of a cursor one position backwards
Ctrl+F ¹	Transition of a cursor one position ahead

¹In the current firmware version 3.34.1 functionality is not implemented

Command line interface enables user authorization and restricts access to commands depending on their access level, provided by the administrator. For access control purposes, MA4000-PX commands are divided into 2 groups based on the user's area of responsibility.

List of groups:

1. commands that control the initial start of the device:
 - crate parameters configuration commands;
 - station part (OLT) configuration commands;
 - subscriber equipment (ONT) configuration commands;
 - profile management commands;
2. active configuration view commands;
3. monitoring commands.

You can create as many users as you like, access rights will be assigned individually to each user.

- ✓ In factory configuration, the system includes one user with the 'admin' name and the 'password' password.

The system allows multiple users to connect to the device simultaneously.

Profile and template names are case sensitive.

4 Command system structure

- [Global mode](#)
- [Configuration management mode](#)

The MA4000-PX command line interface command system is divided into the hierarchical levels – modes (view).

From the global ROOT mode, you can enter the device parameters' configuration mode – **CONFIG** mode. Only users with the access level 15 are able to enter the configuration mode.

To switch from the global mode ROOT, you should run the following commands:

```
ma4000# configure terminal
ma4000(config)#
```

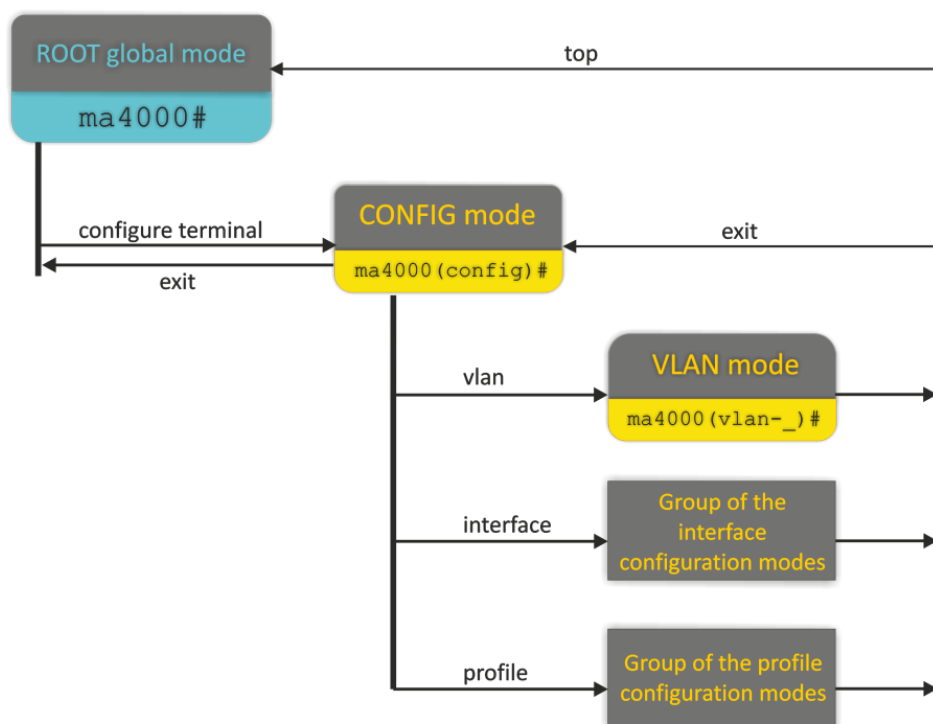


Figure 1 – Command mode hierarchy

Top level of the command hierarchy is shown in [Table 3.1](#).

Table 3.1 – Command modes hierarchy (top level)

Level	Entry command	Prompt line view	Previous level
Global mode (ROOT)		ma4000#	
MA4000 configuration management mode (CONFIG)	configure terminal	ma4000(config)#	ROOT
Interface configuration	For detailed information see Table 3.2	CONFIG	

Level	Entry command	Prompt line view	Previous level
Profiles management	For detailed information see Table 3.3	CONFIG	
VLAN configuration (VLAN)	vlan	ma4000(vlan-N)#	CONFIG
Access-list configuration	access-list-any access-list-MAC access-list-ip	ma4000 (acl)# ma4000(acl-mac)# ma4000(acl-ip)#	CONFIG

Table 3.2 – Interface configuration command modes

Level	Entry command	Prompt line view	Previous level
PP4X module external uplink interface configuration (FRONT-PORT)	interface front-port	ma4000(front-port-_)#	CONFIG
Configuration of PLC8 module external GPON interfaces (GPON-PORT)	interface gpon-port	ma4000(gpon-port-_)#	
ONT GPON configuration (PLC ONT)	interface ont	ma4000(config)(if-ont-_)#	
Configuration of PLC8 module external management interface (mgmt) (PLC FRONT-PORT)	interface plc-front-port	ma4000 (switch-config-front-port-_)#	
Configuration of management interfaces located between the Ethernet switch and PLC8 module olt chips (PLC MGMT-PON-PORT)	interface plc-mgmt-pon-port	ma4000 (plc-mgmt-pon-port-x)#	
Configuration of PON interfaces located between the Ethernet switch and PLC8 module OLT chips (PLC PON-PORT)	interface plc-pon-port	ma4000 (plc-pon-port-x)#	
Configuration of the PLC8 module interface aggregation group used for connection to the PP4X module (PLC SLOT-CHANNEL)	Interface plc-slot-channel	ma4000(plc-slot-channel-x)#	
Configuration of PLC8 module interfaces used for connection to PP4X (PLC SLOT-PORT)	interface plc-slot-port	ma4000 (plc-slot-port-x)#	
LAG configuration of PP4X module uplink interfaces (PORT-CHANNEL)	interface port-channel	ma4000(port-channel-_)#	
LAG configuration of PP4X module interfaces for PLC8 modules (PORT-CHANNEL)	interface slot-channel	ma4000(slot-channel-_)#	

Level	Entry command	Prompt line view	Previous level
Configuration of PP4X module interfaces for PLC8 modules (SLOT-PORT)	interface slot-port	ma4000(slot-port-)#	
Configuration of PP4X module internal stacking interfaces (STACK-PORT)	interface stack-port	ma4000(stack-port-)#	

Table 3.3 – Device profile command mode description

Level	Entry command	Prompt line view	Previous level
Configuring the ONT profiles (template)	template<NAME>	ma4000(ont-template)("name")#	CONFIG
Address table profile configuration (PROFILE ADDRESS TABLE)	profile address_table<NAME>	ma4000(config-address-table)("name")	
ONT GEM port profile configuration (PROFILE CROSS CONNECT)	profile cross-connect<NAME>	ma4000(config-cross-connect)("name")#	
DBA profile configuration (PROFILE DBA)	profile dba<NAME>	ma4000(config-dba)("name")#	
DHCP relay agent profile configuration (PROFILE DHCP_RA)	profile dhcp_ra<NAME>	ma4000(config-dhcp-ra)("name")#	
ONT management profile configuration (PROFILE MANAGEMENT)	profile management<NAME>	ma4000(config-management)("name")#	
ONT port profile configuration (PROFILE PORTS)	profile ports<NAME>	ma4000(config-ports)("name")#	
PPPoE intermediate agent profile configuration (PROFILE PPPoE_IA)	profile pppoe_ia<NAME>	ma4000(config-pppoe-ia)("name")#	
Script profile configuration (PROFILE SCRIPTING)	profile scripting<NAME>	ma4000(scripting)("name")#	
ONT bandwidth management profile configuration (PROFILE SHAPING)	profile shaping<NAME>	ma4000(config-shaping)("name")#	
VLAN profile configuration (PROFILE VLAN)	profile vlan<NAME>	ma4000(config-vlan)("name")#	

Global mode

In base command mode (ROOT) you can perform the following:

- system time configuration;
- operation with configuration files:
 - application;
 - confirmation;
 - reset;
 - saving;
 - undo pending changes;
 - copying;
 - revert to the confirmed configuration;
- module/crate reboot;
- ONT/OLT reconfiguration;
- reboot, reset ONT to default by OMCI commands;
- ONT firmware update files download;
- PP4X module firmware operation:
 - boot;
 - selection of the active firmware image.

You can view the following parameters:

- number of connected subscriber optical terminals (ONT);
- list of configured subscriber optical terminals ONT;
- capacity of the received signal from the subscriber's optical terminal;
- PPPoE session tables;
- MAC address tables on the management module/line card/ONT;
- counters and operation status of interfaces;
- information about interfaces operating with the LACP protocol;
- multicast groups that are learnt via IGMP snooping;
- general information about IGMP snooping parameters.

Configuration management mode

The system settings are controlled in CONFIG mode. The mode is available from the global mode (ROOT).

To switch to the configuration mode, you need to execute the following commands:

```
ma4000#configure terminal
ma4000(config)#
```

In the configuration mode, the adjustment is performed:

- device access parameters: vlan/ip/mask/access-list;
- PP4X/PLC8 module interfaces;
- VLAN parameters;
- DHCP client parameters;
- isolated port groups;
- Selective Q-in-Q feature;
- system log filters;
- OLT/ONT configuration profiles;
- ONT configuration templates;
- adding to configuration and ONT configuration;
- MAC address storage time configuration;
- OLT/ONT configuration parameters;

- access management list;
- queue priorities;
- traffic filtering;
- traffic speed limiting.

5 Interface name system

This chapter describes the naming rules for MA4000-PX interfaces.

Access node interfaces can be divided into two groups: Ethernet interfaces and GPON interfaces. Ethernet interfaces are used for access node connection to operator's network core. GPON interfaces are used for ONT connections.

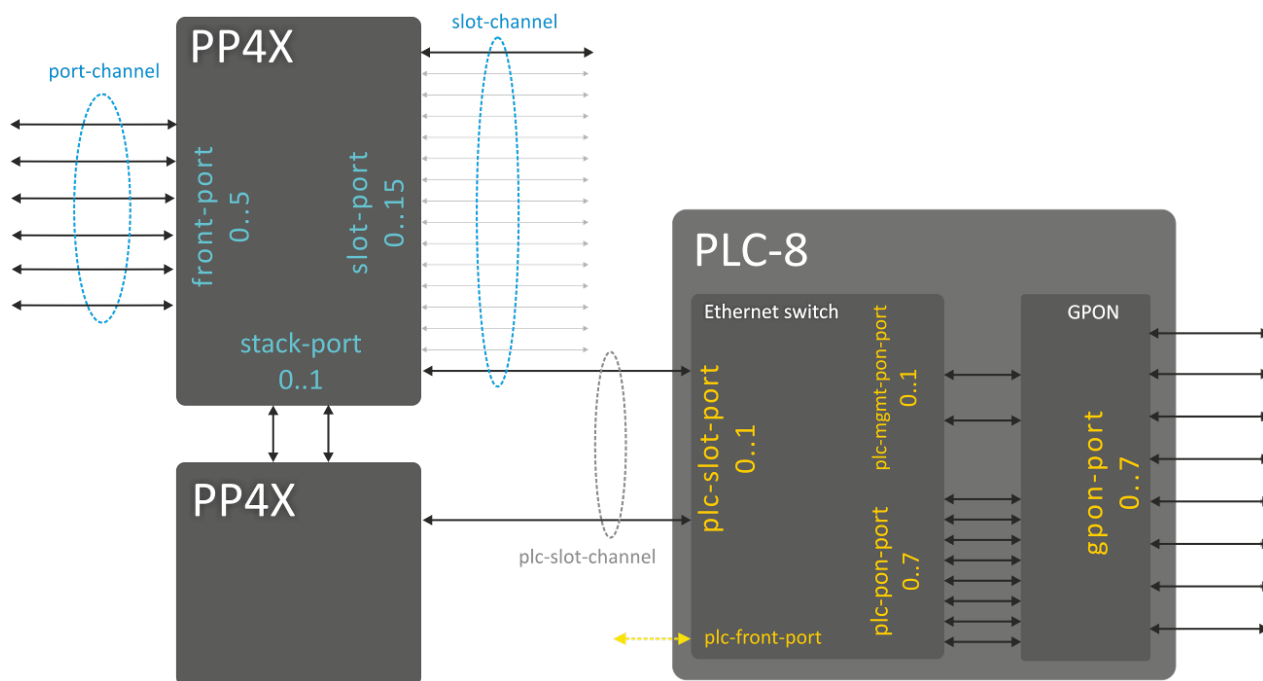


Figure 2 – Access node interface name system

For accepted access node interface name system, see [Table 4.1](#).

Table 4.1 – Access node interface name system and numbering

Interface name	Description	Range
front-port	PP4X module external uplink interfaces	Specified as: U/P - U – PP4X module number [1 .. 2] - P – PP4X uplink interface number [0..5]
port-channel	LAG of PP4X module uplink interfaces	[1..8] – aggregation group number
slot-port	Interfaces of the PP4X module for connecting GPONPLC8 modules	Specified as: U/P - U – PP4X module number [1 .. 2] - P – interface number for PLC8 module [0..15]
slot-channel	LAG of PP4X module interfaces for PLC8 modules	[0..15] – PLC8 module number
stack-port	PP4X module internal stacking interfaces	Specified as: U/P - U – PP4X module number [1 .. 2] - P – interface number for PP4X module [0..1]

Interface name	Description	Range
plc-slot-port	PLC8 module interfaces for connection to central switches – PP4X modules	Specified as: SP • S – PLC8 module number [0 .. 15] • P – interface number for PP4X module [0..1]
plc-slot-channel	PLC8 module interface LAG for connection to central switches – PP4X modules	[0..15] – PLC8 module number
plc-front-port	PLC8 module external management interface (mgmt)	[0..15] – PLC8 module number
plc-pon-port	PLC8 switch downlink interface connected to GPONOLT	Specified as: SP • S – PLC8 module number [0 .. 15] • P – port number [0..7]
plc-mgmt-pon-port	PLC8 switch downlink interface connected to GPONOLT designed to control	Specified as: SP • S – PLC8 module number [0 .. 15] • P – port number [0..1]
gpon-port	PLC8 module external GPON interfaces	Specified as: SP • S – PLC8 module number [0 .. 15] • P – port number [0..7]
ont	Interfaces for the subscriber-side ONT terminals	Specified as: S/P/I • S – PLC8 module number [0..15] • P – PLC8 module port number [0..7] • I – ONT interface number [0..63]

6 User interface commands

- [exit](#)
- [top](#)
- [do](#)
- [show history](#)
- [!](#)
- [reboot](#)
- [configure terminal](#)
- [ping](#)
- [traceroute](#)

This section contains a description of commands that are used to control the device in various modes

exit

This command is used to return to the level up or exit cli if the command is executed in the global command mode.

Syntax

`exit`

Parameters

The command contains no arguments.

Command mode

Return to the top level - all modes, except global.

CLI exit – global mode.

top

The command is used to return to the ROOT command mode.

Syntax

`top`

Parameters

The command contains no arguments.

Command mode

All modes, except global.

do

The do command allows you to execute global level commands (ROOT) when you are at other levels of the command interface.

Syntax

do <COMMAND>

Parameters

<COMMAND> – global level command.

Command mode

All modes, except global.

Example

```
ma4000 (config)# do show firmware
```

Firmware status:

~~~~~

| Unit | Image | Running | Boot | Version         | Date                 |
|------|-------|---------|------|-----------------|----------------------|
| ---  | ---   | ---     | ---  | ---             | ---                  |
| 1    | 0     | No      |      | 1 3 2 379 40889 | 12-Nov-2014 08:41:55 |
| 1    | 1     | Yes     | *    | 1 3 2 380 40897 | 12-Nov-2014 22:58:46 |
| 2    | 0     | Yes     | *    | 1 3 2 380 40897 | 12-Nov-2014 22:58:46 |
| 2    | 1     | No      |      | 1 3 2 379 40889 | 12-Nov-2014 08:41:55 |

"\*" designates that the image was selected **for** the next boot

**show history**

The command displays information on the command used in a current session.

**Syntax**

show history

**Parameters**

The command contains no arguments.

**Command mode**

All modes.

**Example**

```
ma4000# show history
Last CLI commands:

show arp
show cmd-dispatcher
show date
show evt-dispatcher
configure terminal
exit
show firmware
show history
```

**!**

Service command.

The command is used to enter text comments.

**Syntax**

! <COMMAND>

**Parameters**

<COMMAND> – text comment

**Command mode**

All modes.

**Example**

```
ma4000#! hello
```

**reboot**

This command restarts the device components or the whole device.

**Syntax**

reboot <OBJECT>

**Parameters**

<OBJECT> – object to reboot:

- master – master PP4X module;
- slave – slave PP4X module;
- non-stop – reboot the PP4X control modules one by one (without interrupting services)

- slot <NUMBER> – line module, where:
  - <NUMBER> – the number of the slot in which the linear module is set takes values [0 .. 15];
- system – whole device.

## Command mode

ROOT

## Example

```
ma4000# reboot system
```

Crate reboot.

## configure terminal

The command allows you to switch to the device parameters configuration mode.

## Syntax

configure terminal

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# configure terminal
ma4000(config)#
```

Enter the device parameters configuration mode.

## ping

This command allows you to check the availability of the node by exchanging ICMP messages.

## Syntax

ping <IPADDRESS>

## Parameters

<IPADDRESS> – remote device IP address.

## Command mode

ROOT

## Example

```

ma4000# load-1-3-2-3# ping 192.168.16.100
PING 192.168.16.100 (192.168.16.100): 56 data bytes
64 bytes from 192.168.16.100: seq=0 ttl=127 time=0.307 ms
64 bytes from 192.168.16.100: seq=1 ttl=127 time=0.333 ms
64 bytes from 192.168.16.100: seq=2 ttl=127 time=0.214 ms

--- 192.168.16.100 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.214/0.284/0.333 ms

```

## traceroute

This command allows you to view the data path to the remote node.

## Syntax

traceroute <IPADDRESS>

## Parameters

<IPADDRESS> – remote device IP address.

## Command mode

ROOT

## Example

```

ma4000# traceroute 192.168.16.100
ma4000# traceroute to 192.168.16.100 (192.168.16.100), 30 hops max, 38 byte packets
 1  192.168.199.2 (192.168.199.2)  0.307 ms  0.197 ms  0.191 ms
 2  192.168.16.100 (192.168.16.100)  0.224 ms  0.347 ms  0.329 ms

```

## 7 Network management parameters configuration

- [management gateway](#)
- [management ip](#)
- [management vlan](#)
- [hostname](#)
- [show management](#)
- [management access-list default](#)
- [management access-list-any](#)
  - [add](#)
  - [insert](#)
  - [remove](#)
- [management access-list-mac](#)
  - [add](#)
  - [insert](#)
  - [remove](#)
- [management access-list clear](#)
- [show access-list](#)

### management gateway

This command sets the default gateway IP address.

The use of the negative (no) form of the command removes the default gateway IP address.

#### Syntax

```
management gateway <GATEWAY>  
no management gateway
```

#### Parameters

<GATEWAY> – gateway IP address.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# management gateway 192.168.24.15
```

### management ip

This command specifies an IP address and a subnet mask for a crate.

The use of a negative (no) form of the command removes an IP address and a subnet mask for a crate.

#### Syntax

```
management ip <IP> <MASK>  
no management ip
```

**Parameters**

<IP> – IP address;  
<MASK> – subnet mask.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# management ip 192.168.14.15 255.255.255.0
```

**management vlan**

This command sets management VLAN.

The use of a negative form (no) of the command sets VLANID to 1.

**Syntax**

```
management vlan <VID>  
no management vlan
```

**Parameters**

<VID> – VLAN ID, may take values [1 .. 4094].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# management vlan 7
```

The management is available via VLANID 7.

**hostname**

The command allows assigning a name to an object. The maximum length of a string is 32 characters.

**Syntax**

```
hostname <NAME>
```

**Parameters**

<NAME> – device name.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# hostname test
```

The device will have the 'test' name.

**show management**

This command allows to view information about current network settings.

**Syntax**

```
show management
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show management
Network parameters :
  ip           192.168.199.150
  mask         255.255.255.0
  gateway      192.168.199.2
  vlan         199
```

**management access-list default**

This command allows to set the device access policy by access-list

**Syntax**

```
management access-list default <POLICY>
```

**Parameters**

<POLICY> – default access policy, may take values:

- allow – access granted for all hosts,
- deny – access denied for all hosts.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# management access-list default allow
```

**management access-list-any****add**

This command allows to set access-list rules by protocol type and access interface without checking MAC/IP.

**Syntax**

```
add <POLICY> <PROTOCOL> <INTERFACE> [<PORT NUMBER>]
```

**Parameters**

<POLICY> – default access policy, may take values: allow, deny;

<PROTOCOL> – management protocol, may take values: any, http, ssh, telnet;

<INTERFACE> – management interface, may take values: any, front-port, port-channel, slot-channel;

<PORT NUMBER> – port number, if front-port/port-channel/slot-channel is selected as interface.

**Command mode**

MANAGEMENT ACCESS-LIST-ANY

**Example**

```
ma4000(acl)# add allow any front-port 1/1
```

**insert**

This command allows to add access-list rules to an existing list.

**Syntax**

```
insert <POLICY> <PROTOCOL> <INTERFACE> [<PORT NUMBER>] <POSITION>
```

**Parameters**

<POLICY> – default access policy, may take values: allow, deny;

<PROTOCOL> – management protocol, may take values: any, http, ssh, telnet;

<INTERFACE> – management interface, may take values: any, front-port, port-channel, slot-channel;

<PORT NUMBER> – port number, if front-port/port-channel/slot-channel is selected as interface;

<POSITION> – rule sequence number (see section [show access-list](#)).

### Command mode

MANAGEMENT ACCESS-LIST-ANY

### Example

```
ma4000(acl-ip)# insert allow any front-port 1/1 2.2.2.2 255.255.255.255 1
```

### remove

This command allows to delete existing access-list rules.

### Syntax

```
remove <POLICY>/<POSITION> <PROTOCOL> <INTERFACE> [PORT NUMBER] <IP ADDRESS> <MASK>
```

### Parameters

<POLICY> – default access policy, may take values: allow, deny;

<POSITION> – rule sequence number (see table show access-list).

<PROTOCOL> – management protocol, may take values: any, http, ssh, telnet;

<INTERFACE> – management interface, may take values: any, front-port, port-channel, slot-channel;

<PORT NUMBER> – port number, if front-port/port-channel/slot-channel is selected as interface.

<IPADDRESS> – device IP address or network address;

<MASK> – subnet mask.

### Command mode

MANAGEMENT ACCESS-LIST-IP

### Example

```
ma4000(config)# remove allow any front-port 1/1 2.2.2.2 255.255.255.255
```

## management access-list-mac

### add

This command allows to add access-list rules by MAC address.

**Syntax**

```
add <POLICY> <PROTOCOL> <INTERFACE> [<PORT NUMBER>] <MAC ADDRESS>
```

**Parameters**

<POLICY> – default access policy, may take values: allow, deny;

<PROTOCOL> – management protocol, may take values: any, http, ssh, telnet;

<INTERFACE> – management interface, may take values: any, front-port, port-channel, slot-channel;

<PORT NUMBER> – port number, if front-port/port-channel/slot-channel is selected as interface.

<MACaddress> – device MAC address.

**Command mode**

MANAGEMENT ACCESS-LIST-MAC

**Example**

```
ma4000(acl-mac)# add allow any any 02:02:02:02:02:02
```

**insert**

This command allows to add access-list rules to an existing list.

**Syntax**

```
Insert <POLICY > <PROTOCOL> <INTERFACE> [<PORT NUMBER>] <MAC ADDRESS> <POSITION>
```

**Parameters**

<POLICY> – default access policy, may take values: allow, deny;

<PROTOCOL> – management protocol, may take values: any, http, ssh, telnet;

<INTERFACE> – management interface, may take values: any, front-port, port-channel, slot-channel;

<PORT NUMBER> – port number, if front-port/port-channel/slot-channel is selected as interface;

<MACADDRESS> – device MAC address;

<POSITION> – rule sequence number (see table show access-list).

**Command mode**

MANAGEMENT ACCESS-LIST-MAC

**Example**

```
ma4000 (acl-mac)# insert allow any front-port 1/1 03:03:03:03:03:03 0
```

**remove**

This command allows to delete existing access-list rules.

**Syntax**

```
remove <POLICY>/<POSITION> <PROTOCOL> <INTERFACE> [<PORT NUMBER>] <MAC ADDRESS>
<POSITION>
```

**Parameters**

<POLICY> – default access policy, may take values: allow, deny;

<POSITION> – rule sequence number (see table show access-list).

<PROTOCOL> – management protocol, may take values: any, http, ssh, telnet;

<INTERFACE> – management interface, may take values: any, front-port, port-channel, slot-channel;

<PORT NUMBER> – port number, if front-port/port-channel/slot-channel is selected as interface;

<MACADDRESS> – device MAC address;

<POSITION> – rule sequence number (see table show access-list).

**Command mode**

MANAGEMENT ACCESS-LIST-MAC

**Example**

```
ma4000(acl-mac)# remove allow any front-port 1/1 03:03:03:03:03:03 0
```

**management access-list clear**

This command clears access-list.

**Syntax**

```
management access-list clear
```

**Parameters**

The command contains no arguments.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# management access-list clear
```

## show access-list

This command allows to view information about current access-lists.

### Syntax

```
show access-list
```

### Parameters

The command contains no arguments.

### Command mode

ROOT

### Example

```
ma4000# show access-list
```

#### ACL MAC rules

```
~~~~~
```

| Index | Policy | Proto | Interface | Mac                   |
|-------|--------|-------|-----------|-----------------------|
| ----  | -----  | ----- | -----     | -----                 |
| 0     | allow  | any   | any       | mac 02:02:02:02:02:02 |

#### ACL IP rules

```
~~~~~
```

| Index | Policy | Proto  | Interface | Ip                         |
|-------|--------|--------|-----------|----------------------------|
| ----  | -----  | -----  | -----     | -----                      |
| 0     | allow  | ssh    | any       | ip 1.1.1.1 255.255.255.255 |
| 1     | allow  | telnet | any       | ip 1.1.1.1 255.255.255.255 |

#### ACL ANY rules

```
~~~~~
```

| Index | Policy | Proto | Interface      |
|-------|--------|-------|----------------|
| ----  | -----  | ----- | -----          |
| 0     | allow  | http  | front-port 2/2 |
| 1     | allow  | ssh   | front-port 1/1 |
| 2     | allow  | any   | front-port 1/3 |

```
DEFAULT RULE: allow
```

## 8 System time configuration

- [date](#)
- [clock timezone](#)
- [show date](#)
- [clock summer-time](#)

### date

This command sets the system time on the device.

#### Syntax

```
date <TIME> <DAY> <MONTH> <YEAR>
```

#### Parameters

<TIME> – time, specifies in format of hh:mm:ss;

<DAY> – day, may take values [1..31];

<MONTH> – month, may take values: jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec;

<YEAR> – year, may take values [2000..2038].

#### Command mode

ROOT

#### Example

```
ma4000# date 11:00:00 2 jan 2014
```

### clock timezone

This command sets the time zone on the device.

#### Syntax

```
clock timezone hours <HOURS> minutes <MINUTES>
```

#### Parameters

<HOURS> – hourly shift relative to Coordinated Universal Time (UTC);

<MINUTES> – shift in minutes relative to Coordinated Universal Time (UTC).

#### Command mode

CONFIG

## Example

```
ma4000(config)# clock timezone hours 1 minutes 0
```

## show date

This command displays the system time of the basket.

## Syntax

```
show date
```

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show date
Fri May 20 16:18:53 T 2011
```

System time set on the device: 16 hours 18 minutes 53 seconds, May 20, 2011, Friday.

## clock summer-time

This command allows you to configure the transition to daylight saving time of the system time of the basket.

## Syntax

```
clock summer-time recurring zone <DST_ZONE> start-week <WEEK> start-day <DAY> start-month <MONTH> start-time <TIME> end-week <WEEK> end-day <DAY> end-month <MONTH> end-time <TIME> hours <HOURS> minutes <MINUTES>
```

- ✓ The command allows you to set the time of switching the clock without reference to a specific date, for example, the first Saturday of March.

```
clock summer-time date zone <DST_ZONE> start-day <DATE> start-month jan start-year <YEAR> start-time <TIME> end-day <DATE> end-month dec end-year <YEAR> end-time <TIME> hours <HOURS> minutes <MINUTES>
```

- ✓ The command allows you to set the exact date of switching the clock, for example, December 4.

## Parameters

<DST\_ZONE> – zone name;

<WEEK> – number of week in month, may take values [1..5];

<DAY> – day of the week, may take values [mon, tue, wed, thu, fri, sat, sun];

<DATE> – day of the week, may take values [1..31]

<MONTH> – month, may take values: jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec;

<TIME> – time, specified in format of hh:mm;

<YEAR> – year, may take values [2000..2038];

<HOURS> – number of hours, on which time will shift, may take values -23..23;

<MINUTES> – number of minutes, on which time will shift, may take 0..59.

## Command mode

CONFIG

## Example

```
ma4000(configure)# clock summer-time recurring zone AAA start-week 1 start-day sat start-month
mar start-time 11:11 end-week 1 end-day sat end-month nov end-time 11:59 hours 10 minutes 00
```

## 9 System user management

- [user](#)
- [show users config](#)
- [show users who](#)
- [privilege](#)
- [show privileges](#)

### user

This command adds a user to the system, sets a password and access level.

The use of a negative form (no) of the command removes a user from the system.

### Syntax

```
user <USER_NAME> password <USER_PASSWD> privilege <PRIORITY>
nouser <USER_NAME>
```

### Parameters

<USER\_NAME> – user name, specified as string of [1 .. 255] characters;

<USER\_PASSWD> – user password, specified as string of [8 .. 31] characters;

<PRIORITY> – user access level [0..15], where 0 – the lowest level, 15 – the highest access level.

### Command mode

CONFIG

### Example 1

```
ma4000#(config) user name password password privilege 15
```

A user with the test name and password password has been added.

### Example 2

```
ma4000#(config) no user test
```

A user named test has been removed from the system.

### show users config

This command allows to view the list of users and their access rights:

- User name – user name;
- User privilege level – user privilege level.

**Syntax**

```
show users config
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show users config

System users
~~~~~
User name          User privilege level
-----
root               15
admin              15
remote             15
linux              0
test               5
user               8
qwqwe             9
7 system users.
```

**show users who**

This command allows to view the list of active users, as well as the host, time and connection protocol:

- User name – user name;
- Logged in at – date and time of connection;
- Host – protocol and IP address from which the connection was performed;
- User privilege level – user privilege level.

**Syntax**

```
show users who
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

## Example

```
ma4000# show users who
```

```

  User sessions
  ~~~~~
SID/PID User name Logged in at Host Priv
Timers Login/Priv User permissions

eb2cf6a1/8751 admin 16/01/17 16:18:36 SSH 192.168.16.235 no
00:27:59/00:00:00 all
1 user sessions.
```

## privilege

This command defines a set of privileges for a specified level.

The use of a negative form (no) of the command removes privilege from the level.

## Syntax

```
[no] privilege <PRIORITY> <PRIVILEGE>
```

## Parameters

<PRIORITY> – user access level [0..15], where 0 – the lowest level, 15 – the highest access level;

<PRIVELEGE> – a privilege that gives access to a specific command section:

- view-switch – allows to view PP4X switch and slot configuration;
- view-alarm – allows to view active alarms, their configuration and event log;
- view-system – allows to view system settings: logging, user configuration, Tacacs;
- view-general: allows to view basic settings – management, firmware information, status of boards and log message reading;
- view-gpon – allows to view configuration and status of OLT chips, GPON ports, and OLT;
- view-ont – allows to view MAC tables and ONT counters;
- view-ont-profile – allows to view ONT profile configuration;
- view-switch-interfaces – enables Ethernet interface operation monitoring: counters; Ethernet port status, utilization and configuration; MAC table configuration;
- config-switch – enables switch configuration: LACP, QoS, STP;
- config-alarm – enables alarm configuration;
- config-system – enables configuration of system parameters: logging, user configuration, Tacacs;
- config-general – enables configuration of management parameters and operations with software;
- config-gpon – enables configuration of OLT profiles and configuration of OLT chip basic operation parameters;
- config-ont – enables ONT configuration: adding, removing, service activation;
- ont-operation – allows to execute specific ONT management commands: reboot, reconfiguration, firmware update;
- config-ont-profile – enables configuration of ONT profiles;
- config-switch-interfaces – enables Ethernet interface configuration: aggregation, enabling/disabling, VLAN operations.

**Command mode**

CONFIG

**Example 1**

```
ma4000#(config) privilege 14 view-general
```

View-general privilege has been added for Level 14.

**Example 2**

```
ma4000#(config) no privilege 10 view-switch
```

For Level 10, view-switch privilege is not allowed.

**show privileges**

The command allows to view the current privilege levels configuration.

**Syntax**

```
show privileges
```

**Parameters**

The command does not contain parameters

**Command mode**

ROOT

**Example**

```
ma4000# show privileges
```

| Privilege levels<br>~~~~~ |                                                                                                                                                                                                        |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Level                     | Privileges                                                                                                                                                                                             |
| 0                         | !, exit                                                                                                                                                                                                |
| 1                         | view-ont                                                                                                                                                                                               |
| 2                         | ont-operation                                                                                                                                                                                          |
| 3                         | view-ont, ont-operation                                                                                                                                                                                |
| 4                         | view-ont, config-ont, ont-operation                                                                                                                                                                    |
| 5                         | view-ont, view-ont-profile,<br>ont-operation                                                                                                                                                           |
| 6                         | view-gpon, view-ont, view-ont-profile,<br>config-gpon, ont-operation,<br>config-ont-profile                                                                                                            |
| 7                         | view-switch, view-gpon, view-ont,<br>view-ont-profile                                                                                                                                                  |
| 8                         | view-switch, view-gpon, view-ont,<br>view-ont-profile,<br>view-switch-interfaces, config-switch,<br>config-switch-interfaces                                                                           |
| 9                         | view-switch, view-gpon, view-ont,<br>view-ont-profile,<br>view-switch-interfaces, config-switch,<br>config-gpon, config-ont, ont-operation,<br>config-ont-profile,<br>config-switch-interfaces         |
| 10                        | view-switch, view-alarm, view-system,<br>view-general, view-gpon, view-ont,<br>view-ont-profile, view-switch-interfaces                                                                                |
| 11                        | view-switch, view-alarm, view-system,<br>view-general, view-gpon, view-ont,<br>view-ont-profile,<br>view-switch-interfaces, config-alarm,<br>config-system, config-general                             |
| 12                        | view-alarm, view-system, view-general,<br>view-gpon, view-ont, view-ont-profile,<br>view-switch-interfaces, config-switch,<br>config-alarm, config-system,<br>config-general, config-switch-interfaces |
| 13                        | view-gpon, view-ont, view-ont-profile                                                                                                                                                                  |
| 14                        | view-general                                                                                                                                                                                           |
| 15                        | view-switch, view-alarm, view-system,<br>view-general, view-gpon, view-ont,<br>view-ont-profile,<br>view-switch-interfaces, config-switch,                                                             |

```
config-alarm, config-system,
config-general, config-gpon, config-ont,
ont-operation, config-ont-profile,
config-switch-interfaces
```

## 10 Alarm and event log management

- clear alarm
- show alarm
- journal filter
- journal keep-time
- journal max-size
- show alarm events
- show alarm active
- alarm fan
- alarm free-space
- alarm load-average
- alarm ont-high-rx-power
- alarm ont-low-rx-power
- ram
- alarm temperature
- alarm login
- alarm config-save
- alarm firmware-update
- alarm duplicate-mac
- alarm physical-layer-flapping
- alarm pon-channel-no-ont
- alarm ont-physical-layer
- alarm olt-update
- alarm ont-update
- alarm channel-flapping
- alarm ont-flapping
- alarm download
- alarm ont-battery-power
- alarm ont-battery-low
- alarm lan-los
- alarm ont-config
- alarm file-delete
- alarm physical-layer-errors
- alarm physical-layer-block
- alarm link
- alarm logout
- alarm ont-dying-gasp
- alarm ont-rei
- alarm ont-power-off
- alarm config-change
- alarm shutdown
- alarm oms
- alarm ont-state-changed
- alarm ont-config-changed
- alarm channel-state-changed
- alarm pon-alarm-channel
- alarm pon-alarm-onui
- alarm ont-update-inprogress
- alarm olt-device-reset
- alarm ont-signal-degrade
- alarm high-rx-power
- alarm channel-ont-count-overflow
- alarm olt-device-not-working
- alarm redundancy-switch

- [alarm redundancy-fail](#)
- [alarm system-reboot](#)
- [alarmdummy](#)
- [show alarm configuration](#)

The system log management commands are described in this section. In a system operating with central switch redundancy, commands from this section can only be entered and executed on the master module.

## clear alarm

This command is used to delete entries from the system alarm log.

### Syntax

```
clear alarm <TYPE>
```

### Parameters

<TYPE> – log entry type for deletion:

- active – active alarms deletion;
- before – deleting all log entries (alarm events) before the specified date. The date is specified in format of YYYY.MM.DD-hh:mm;
- events – delete all event log entries.

### Command mode

ROOT

### Example

```
ma4000# clear alarm active
```

The log entries about active accidents have been deleted.

## show alarm

This command will show the alarm list.

Time – alarm registration time, DD:MM:YYYY hh:mm:ss;

Priority – alarm priority;

Text – alarm description.

### Syntax

```
show alarm <PARAM> [ALARM]
```

### Parameters

<PARAM> – alarm type:

- active – show active alarm list;
- events – show all event log;
- configuration – show configuration of selected alarm.

## Command mode

ROOT

## Example

```
ma4000(alarms)#show alarm active
Active alarms
~~~~~
Time                Priority  Text
-----
06-09-2011 14:02:36    2      MA4000_ALARM_LINK_DOWN front-port 1/1
06-09-2011 14:02:36    2      MA4000_ALARM_LINK_DOWN front-port 1/2
```

## journal filter

This command sets the rules of making entries for some events in the log.

The use of a negative form (no) of the command sets the default value.

## Syntax

[no] journal filter<FILTER>

## Parameters

<FILTER> – filter events by type:

dying-gasp – set local database to disable the sending of ONT dying gasp alarm messages;

ont-state – set local database to cancel sending alarm messages to the log when the ONT status changes: blocked, ok, free.

## Command mode

CONFIG

## Example

```
ma4000(config)# journal filter ont-state
```

## journal keep-time

This command sets the maximum time for storing entries in the event log.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
journal keep-time <TIME>
```

**Parameters**

<TIME> – time period, may take values in the range [10..10000], in hours.

**Command mode**

CONFIG

**Example**

```
ma4000(config)#journal keep-time 20
```

**journal max-size**

This command is used to set the size of the system log.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
journal max-size <SIZE>
```

**Parameters**

<SIZE> – maximum log size, may take values in the range [1000..50000] Kbyte.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# journal max-size 10000
```

**show alarm events**

This command is used to view the event log entries.

**Syntax**

```
show alarm events
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show alarm events
```

```

  Alarms
  ~~~~~

Time Priority Text

27-11-2014 16:42:05 2 PLC, slot 6: ELTX5F0002A0 2 20 FREE 'NTU-2V'
 '3.50.2.1193'

27-11-2014 16:44:41 2 PLC, slot 6: ELTX5F0002A0 2 20 OK 'NTU-2V'
 '3.50.2.1193'
```

**show alarm active**

This command is used to view the active event entries.

**Syntax**

```
show alarm active
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
MA4000# show alarm active
```

```

 Active alarms
  ~~~~~

Time                Priority      Text
-----
28-12-2016 15:46:37    1      MA4000_ALARM_LINK_DOWN front-port 1/1
28-12-2016 15:46:37    1      MA4000_ALARM_LINK_DOWN front-port 1/2

2 active alarms
```

## alarm fan

This command allows to configure the event generation parameters for logging the fan operation status. The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm fan <PARAM> <VALUE>
no alarm fan <PARAM>
```

### Parameters

<PARAM> – event parameter:

- max-rpm – set the upper limit of fan speed rpm;
- min-rpm – set the lower limit of fan speed rpm;
- in – formation of the event when the parameter value goes abroad;
- out – formation of event normalization when the parameter value is returned to the boundaries;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has several special values.
- 0 – event is in the log until the normalizing event comes;
- 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value, optional parameter.

### Command mode

CONFIG

### Example

```
ma4000(config)# alarm fan max-rpm 2000
```

## alarm free-space

This command allows to configure the event generation parameters for recording to the alarm log if there is not enough free disk space.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm free-space <PARAM> <VALUE>
no alarm free-space <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- level – set the lower limit of free disk space, in %;
- in – formation of the event when the parameter value goes abroad;
- out – formation of event normalization when the parameter value is returned to the boundaries;

- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

### Example

```
ma4000(config)# alarm free-space level 20
```

### alarm load-average

This command allows to configure the event generation parameters for logging the alarms at high CPU load. The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm load-average <PARAM> <VALUE>
no alarm load-average <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- level – set upper CPU load level;
- in – formation of the event when the parameter value goes abroad;
- out – formation of event normalization when the parameter value is returned to the boundaries;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has several special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

### Example

```
ma4000(config)# alarm load-average level 255
```

## alarm ont-high-rx-power

This command allows to configure the event generation parameters for logging alarms when the received signal level on the ONT side is above a threshold value.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm ont-high-rx-power <PARAM> <VALUE>
no alarm ont-high-rx-power <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- level – sets the maximum power level of the signal received by ONT (-127..0) dBm, special value 0xFF (disables sending no alarm ont-high-rx-power level);
- in – formation of the event when the parameter value goes abroad;
- out – formation of event normalization when the parameter value is returned to the boundaries;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has several special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

### Example

```
ma4000(config)# alarm ont-high-rx-power level -12
```

## alarm ont-low-rx-power

This command allows to configure the event generation parameters for logging alarms when the received signal level on the ONT side is below a threshold value.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm ont-low-rx-power <PARAM> <VALUE>
alarm low-rx-power <PARAM> <VALUE>
no alarm ont-low-rx-power <PARAM>
no alarm low-rx-power <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- **level** – sets the minimum power level of the signal received by ONT (-127..0) dBm, special value 0xFF (disables sending no alarm ont-low-rx-power level);
- **in** – formation of the event when the parameter value goes abroad;
- **out** – formation of event normalization when the parameter value is returned to the boundaries;
- **severity** – describes the degree of importance of the event (info, minor, major, critical);
- **ttl** – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ont-low-rx-power level -28
```

## ram

This command allows to configure the event generation parameters for logging alarms when the free RAM amount of the device is below the threshold value.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm ram <PARAM> <VALUE>
no alarm ram <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- **free-ram** – set the threshold value of the minimum free memory capacity in %;
- **in** – formation of the event when the parameter value goes abroad;
- **out** – formation of event normalization when the parameter value is returned to the boundaries;
- **severity** – describes the degree of importance of the event (info, minor, major, critical);
- **ttl** – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ram free-ram 35
```

## alarm temperature

This command allows to configure the event generation parameters for recording to the alarm log when the temperature at one of the sensors exceeds the threshold value.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm temperature <PARAM> <VALUE>
no alarm temperature <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- sensor1 – set the threshold value of maximum temperature at sensor 1 of PLC board, in °C;
- sensor2 – set the threshold value of maximum temperature at sensor 2 of PLC board, in °C;
- in – formation of the event when the parameter value goes abroad;
- out – formation of event normalization when the parameter value is returned to the boundaries;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm temperature sensor1 66
```

## alarm login

This command allows to configure the event generation parameters for logging alarms when a user attempts to connect to the system (ssh/telnet/console).

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm login [<PARAM> <VALUE>]
no alarm login <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm login severity info
```

**alarm config-save**

This command allows to configure the event generation parameters for logging alarms while saving the device configuration.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm config-save <PARAM> <VALUE>
no alarm config-save <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm config-save severity info
```

## alarm firmware-update

This command allows to configure the event generation parameters for logging alarms while updating the device firmware.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm firmware-update <PARAM> <VALUE>
no alarm firmware-update <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

## Example

```
ma4000(config)# alarm firmware-update severity critical
```

## alarm duplicate-mac

This command allows to configure the event generation parameters for logging alarms when the same MAC address is detected within the same VLAN on two different ports of the device.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm duplicate-mac <PARAM> <VALUE>
no alarm duplicate-mac <PARAM>
```

**Parameters**

<PARAM> – event parameter:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm duplicate-mac severity critical
```

**alarm physical-layer-flapping**

This command allows to configure the event generation parameters for logging alarms when the physical state (up/down) of the Ethernet port is changed frequently – flapping.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm physical-layer-flapping [<PARAM> <VALUE>]
no alarm physical-layer-flapping <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm physical-layer-flapping severity critical
```

## alarm pon-channel-no-ont

This command allows to configure the event generation parameters for logging alarms when the last (first) ONT registered on the PON port is disabled.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm pon-channel-no-ont <PARAM> <VALUE>
no alarm pon-channel-no-ont <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

## Example

```
ma4000(config)# alarm pon-channel-no-ont severity info
```

## alarm ont-physical-layer

This command allows to configure the event generation parameters for logging alarms when connecting/disconnecting ONT to OLT.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm ont-physical-layer <PARAM> <VALUE>
no alarm ont-physical-layer <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm ont-physical-layer severity info
```

**alarm olt-update**

This command allows to configure the event generation parameters for logging alarms when the OLT firmware update is successful/in error.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm olt-update <PARAM> <VALUE>
```

```
no alarm olt-update <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm olt-update severity info
```

## alarm ont-update

This command allows to configure the event generation parameters for logging alarms in the event of a successful/in error ONT firmware update by OMCI.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm ont-update [<PARAM> <VALUE>]
no alarm olt-update <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ont-update severity info
```

## alarm channel-flapping

This command allows to configure the event generation parameters for logging alarms when the physical state of the GPON port is changed frequently.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm channel-flapping <PARAM> <VALUE>
no alarm channel-flapping <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm channel-flapping severity info
```

**alarm ont-flapping**

This command allows to configure the event generation parameters for logging alarms when the physical state is changed frequently (flapping).

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm ont-flapping [<PARAM> <VALUE>]
no alarm ont-flapping <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm ont-flapping severity info
```

## alarm download

This command allows to configure the event generation parameters for logging alarms while uploading file to the device.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm download <PARAM> <VALUE>
no alarm download <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm download severity info
```

## alarm ont-battery-power

This command allows to configure the event generation parameters for logging alarms when ONT is switched to power from a backup battery source (there must be hardware support for this mode of operation on ONT).

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm ont-battery-power <PARAM> <VALUE>
no alarm ont-battery-power <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ont-battery-power severity info
```

## alarm ont-battery-low

This command allows to configure event generation parameters for logging at low ONT battery backup source (there should be hardware support for this mode of operation on ONT).

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm ont-battery-low [<PARAM> <VALUE>]
no alarm ont-battery-low <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ont-battery-low severity info
```

## alarm lan-los

This command allows to configure the event generation parameters for logging alarms when changing the physical state of LAN ports to ONT.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm lan-los <PARAM> <VALUE>
no alarm lan-los <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

## Example

```
ma4000(config)# alarm lan-los severity info
```

## alarm ont-config

This command allows to configure the event generation parameters for logging alarms when configuring ONT by OMCI.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm ont-config <PARAM> <VALUE>
no alarm ont-config <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm ont-config severity info
```

**alarm file-delete**

This command allows to configure the event generation parameters for recording to the alarm log when deleting files from the system.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm file-delete <PARAM> <VALUE>
no alarm file-delete <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm file-delete severity info
```

## alarm physical-layer-errors

This command allows to configure the event generation parameters for logging alarms when errors occur on the Ethernet ports of the device.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm physical-layer-errors <PARAM> <VALUE>
no alarm physical-layer-errors <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

## Example

```
ma4000(config)# alarm physical-layer-errors severity info
```

## alarm physical-layer-block

This command allows to configure the event generation parameters for logging alarms when the device Ethernet port is locked.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm physical-layer-block <PARAM> <VALUE>
no alarm physical-layer-block <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm physical-layer-block severity info
```

**alarm link**

This command allows to configure the event generation parameters for logging alarms when the Ethernet port state of the linkup/down device is changed.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm link <PARAM> <VALUE>
no alarm link <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm link severity info
```

## alarm logout

This command allows to configure the event generation parameters for logging alarms when a user is disconnected from the CLI command interface.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm logout <PARAM> <VALUE>
no alarm logout <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

## Example

```
ma4000(config)# alarm logout severity info
```

## alarm ont-dying-gasp

This command allows to configure the event generation parameters for logging alarms in the event of power failure on the ONT. Support on ONT for sending PLOAM messages in case of power failure is required.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm ont-dying-gasp <PARAM> <VALUE>
no alarm ont-dying-gasp <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ont-dying-gasp severity info
```

## alarm ont-rei

This command allows to set the event generation parameters for logging alarms when receiving REI (Remote Error Indication) ERRi (BIP error of ONUi) data from ONT, see T-REC-G.984.3 11.2 Performance monitoring. Requires support on ONT.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm ont-rei <PARAM> <VALUE>
no alarm ont-rei <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ont-rei severity info
```

## alarm ont-power-off

This command allows to configure the event generation parameters for logging alarms when the ONT PLOAM message about manual power failure is received. Requires support on ONT.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm ont-power-off [<PARAM> <VALUE>]
no alarm ont-power-off <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm ont-power-off severity info
```

## alarm config-change

This command allows to configure the event generation parameters for logging alarms while changing the system configuration.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm config-change <PARAM> <VALUE>
no alarm config-change <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm config-change severity info
```

**alarm shutdown**

This command allows to configure the event generation parameters for logging alarms when snmpagent is restarted in the system.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm shutdown <PARAM> <VALUE>
no alarm shutdown <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event (info, minor, major, critical);
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm shutdown severity info
```

## alarm oms

This command allows to configure the event generation parameters for logging alarms when performing the OMS-MIB operation.

This alarms signals a success or an error when performing file operations when operating over SNMP, these operations are loading and unloading the configuration backup over SNMP and firmware updates over SNMP.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm oms <PARAM> <VALUE>
no alarm oms <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm oms severity info
```

## alarm ont-state-changed

This command allows to configure the event generation parameters for logging alarms at each state change (OMCI configuration) of the ONT. It is used for prompt display of ONT lists in the EMS management system.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm ont-state-changed <PARAM> <VALUE>
no alarm ont-state-changed <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm ont-state-changed severity info
```

**alarm ont-config-changed**

This command allows to configure the event generation parameters for logging alarms at each configuration change (OMCI configuration) of the ONT. It is used for prompt display of ONT lists in the EMS management system.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm ont-config-changed <PARAM> <VALUE>
no alarm ont-config-changed <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm ont-config-changed severity info
```

## alarm channel-state-changed

This command allows to configure the event generation parameters for logging alarms at each configuration change of the GPON channel.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm channel-state-changed <PARAM> <VALUE>
no alarm channel-state-changed <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

## Example

```
ma4000(config)# alarm channel-state-changed severity info
```

## alarm pon-alarm-channel

This command allows to configure the event generation parameters for logging alarms when no ONT is connected to the channel. An alarm occurs after the last ONT is disconnected and is removed after the first one is connected. It does not matter if there is ONT in the configuration or not.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm pon-alarm-channel <PARAM> <VALUE>
no alarm pon-alarm-channel <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm pon-alarm-channel severity info
```

## alarm pon-alarm-onui

This command allows to configure the event generation parameters for logging alarms when an ONT emergency condition is detected.

This alarm is designed to broadcast PLOAM alarms generated on the ONT side:

LOSI: Loss of signal for ONUi

DOWI: Drift of Window of ONUi

LOFI: Loss of frame of ONUi

RDII: Remote Defect Indication of ONUi

LOAMI: Loss of PLOAM for ONUi

LCDGI: Loss of GEM channel delineation

SDI: Signal Degraded of ONUi

SFI: Signal Fail of ONUi etc. See T-REC-G.984.3 11.1 Alarms

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm pon-alarm-onui <PARAM><VALUE>
```

```
no alarm pon-alarm-onui <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;

- **tll** – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

### Example

```
ma4000(config)# alarm pon-alarm-onui severity info
```

### alarm ont-update-inprogress

This command allows to configure the event generation parameters for logging alarms during the ONT firmware update procedure via OMCI.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm ont-update-inprogress <PARAM> <VALUE>
```

```
no alarm ont-update-inprogress <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- **in** – formation of the event when an accident occurs;
- **out** – formation of event normalization at normalization of the accident;
- **severity** – describes the degree of importance of the event: info, minor, major, critical;
- **tll** – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

### Example

```
ma4000(config)# alarm ont-update-inprogress severity info
```

### alarm olt-device-reset

This command allows to configure the event generation parameters for logging alarms during the PON chip reboot.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm olt-device-reset <PARAM> <VALUE>
no alarm olt-device-reset <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

### Example

```
ma4000(config)# alarm olt-device-reset severity info
```

## alarm ont-signal-degrade

This command allows to configure the event generation parameters for logging alarms when detecting a low level of optical signal on the ONT side. ONT should have support for RXpower measurements and sending alarms.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm ont-signal-degrade <PARAM> <VALUE>
no alarm ont-signal-degrade <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Default value:**

-28 dBm

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm ont-signal-degrade severity info
```

**alarm high-rx-power**

This command allows to configure the event generation parameters for logging alarms when detecting a high level of optical signal on the ONT side. ONT should have support for RXpower measurements and sending alarms.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm high-rx-power <PARAM> <VALUE>
no alarm high-rx-power <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Default value:**

-8 dBm

## Example

```
ma4000(config)# alarm high-rx-power severity info
```

## alarm channel-ont-count-overflow

This command allows to configure the event generation parameters for logging alarms when the number of connected ONTs on the channel exceeds the maximum allowed value – 64 ONTs.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm channel-ont-count-overflow <PARAM> <VALUE>
no alarm channel-ont-count-overflow <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm channel-ont-count-overflow severity info
```

## alarm olt-device-not-working

This command allows to configure the event generation parameters for logging alarms in case of configuration error or GPON OLT boot.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
alarm olt-device-not-working <PARAM> <VALUE>
no alarm olt-device-not-working <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# alarm olt-device-not-working severity info
```

**alarm redundancy-switch**

This command allows to configure the event generation parameters for logging alarms when switching to the GPON backup channel.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
alarm redundancy-switch <PARAM> <VALUE>
```

```
no alarm redundancy-switch <PARAM>
```

**Parameters**

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

**Command mode**

CONFIG

## Example

```
ma4000(config)# alarm redundancy-switch severity info
```

## alarm redundancy-fail

This command allows to configure the event generation parameters for logging alarms during switching to the GPON backup channel by failure.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm redundancy-fail <PARAM> <VALUE>
no alarm redundancy-fail <PARAM>
```

### Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

### Command mode

CONFIG

## Example

```
ma4000(config)# alarm redundancy-fail severity info
```

## alarm system-reboot

This command allows to configure the event generation parameters for logging alarms while reboot the system.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
alarm system-reboot <PARAM> <VALUE>
no alarm system-reboot <PARAM>
```

## Parameters

<PARAM> – event parameter, may take values:

- in – formation of the event when an accident occurs;
- out – formation of event normalization at normalization of the accident;
- severity – describes the degree of importance of the event: info, minor, major, critical;
- ttl – time an event exists in the active alarms log. (from 1 to 2,147,483,647). Specified in seconds. It has following special values:
  - 0 – event is in the log until the normalizing event comes;
  - 1 – SNMP trap is sent if specified, but the event is not logged in the alarm log.

<VALUE> – parameter value.

## Command mode

CONFIG

## Example

```
ma4000(config)# alarm system-reboot severity info
```

## alarmdummy

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service. The use of a negative form (no) of the command sets the default configuration.

## show alarm configuration

This command is used to view the event settings for generating the alarm log.

## Syntax

show alarm configuration <TYPE>

## Parameters

<TYPE> – event type:

| Event        | Description                                                         | Threshold |
|--------------|---------------------------------------------------------------------|-----------|
| load-average | Average CPU load reached the threshold, estimated time is 1 minute. |           |
| ram          | Free RAM size decreased to the threshold.                           | 30%       |
| login        | Connecting the user to the management system.                       |           |
| config-save  | The configuration was saved on user initiative.                     |           |

| Event                   | Description                                                   | Threshold                         |
|-------------------------|---------------------------------------------------------------|-----------------------------------|
| firmware-update         | Device firmware update.                                       |                                   |
| duplicate-mac           | Two devices with the same MAC addresses detected in one VLAN. |                                   |
| physical-layer-flapping | Flapping on Ethernet ports.                                   |                                   |
| pon-channel-no-ont      | The first ONT connected/the last ONT disconnected on channel. |                                   |
| ont-physical-layer      | ONT connected/disconnected.                                   |                                   |
| olt-update              | OLT chip firmware update completed successfully/with errors.  |                                   |
| ont-update              | ONT chip firmware update completed successfully/with errors.  |                                   |
| channel-flapping        | GPON interface flapping.                                      |                                   |
| ont-flapping            | ONT flapping.                                                 |                                   |
| download                | File download completed successfully/with errors.             |                                   |
| ont-battery-power       | Switch ONT to battery power.                                  |                                   |
| ont-battery-low         | ONT battery low.                                              | Threshold is set on the ONT side. |
| lan-los                 | Link Down on ONT LAN port.                                    |                                   |
| ont-config              | The configuration of the connected ONT is valid/not valid.    |                                   |
| file-delete             | File deleted successfully/with errors.                        |                                   |
| physical-layer-errors   | Physical layer errors on Ethernet ports.                      |                                   |
| physical-layer-block    | Ethernet port blocked.                                        |                                   |
| link                    | Ethernet port status changed (up/down).                       |                                   |
| logout                  | User logout from the management system.                       |                                   |
| ont-dying-gasp          | Dying Gasp signal received from ONT.                          |                                   |
| ont-rei                 | Remote Error Indication (REI).                                |                                   |
| ont-power-off           | ONT power off.                                                |                                   |
| config-change           | OLT configuration changed.                                    |                                   |

| Event                      | Description                                                                          | Threshold |
|----------------------------|--------------------------------------------------------------------------------------|-----------|
| shutdown                   | SNMP agent disabled.                                                                 |           |
| oms                        | OMS-MIB operation completed successfully/with errors.                                |           |
| ont-state-changed          | ONT state changed.                                                                   |           |
| ont-config-changed         | ONT configuration changed.                                                           |           |
| channel-state-changed      | GPON interface configuration changed.                                                |           |
| pon-alarm-channel          | Event related to GPON interface.                                                     |           |
| pon-alarm-onui             | Event related to ONT.                                                                |           |
| ont-update-inprogress      | Updating ONT firmware.                                                               |           |
| olt-device-reset           | Resetting OLT chip.                                                                  |           |
| ont-signal-degrade         | The attenuation level in the line is lower than allowed for this ONT.                |           |
| high-rx-power              | The level of signal received from this ONT is above the acceptable value.            | -8 dBm    |
| free-space                 | Free drive space decreased to the threshold.                                         | 30%       |
| temperature                | The temperature has exceeded the threshold value.                                    | 60 °C     |
| redundancy-switch          | Switching to the redundant channel.                                                  |           |
| dummy                      | Debug message.                                                                       |           |
| channel-ont-count-overflow | The number of ONTs connected per channel exceeds the maximum allowed value (64 ONT). |           |
| ont-low-rx-power           | Input voltage is low or there is no power on one of the power feeders.               |           |
| olt-device-not-working     | GPON OLT configuration was loaded successfully/with errors.                          |           |
| redundancy-fail            | Failure switching to the redundant channel.                                          |           |
| fan                        | Fan rotation speed exceeded the safe operating limits.                               |           |
| system-reboot              | Reboot of the device.                                                                |           |

## Command mode

ROOT

## Example

```
ma4000# show alarm configuration ram
Alarm:
  Severity:          major
  Send on in:        true
  Send on out:       true
  Ttl:               0
  Free space:        30%
```

## 11 Crate management

- [show shelf](#)
- [show system information](#)
- [show uptime](#)
- [show system environment](#)
- [fan min-speed](#)
- [fan speed](#)
- [fan speed-level](#)
- [fan speed-table](#)
- [slot type](#)
- [slot profile](#)
- [slot terminal-vlan](#)
- [slot logging system loglevel](#)
- [slot logging module](#)
- [show slot terminal-vlan](#)
- [show slot gpon olt configuration](#)
- [show slot cpu detailed1](#)
- [show slot channel-group lacp](#)

### **show shelf**

This command is used to view the configuration and status of the crate, installed in the crate cards.

#### **Syntax**

```
show shelf
```

#### **Parameters**

The command contains no arguments.

#### **Command mode**

ROOT

## Example

```
ma4000# show shelf
```

```
Shelf status
```

```
~~~~~
```

| Slot #<br>State | Configured Type | Detected Type | Version    | Serial #   | Link State | Slot        |
|-----------------|-----------------|---------------|------------|------------|------------|-------------|
| 0               | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 1               | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 2               | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 3               | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 4               | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 5               | plc8            | plc8          | 3.22.0.364 | OL04000854 | up         | Operational |
| 6               | plc8            | plc8          | 3.22.0.364 | OL04001738 | up         | Operational |
| 7               | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 8               | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 9               | plc8            | plc8          | 3.22.0.364 | OL04001762 | up         | Operational |
| 10              | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 11              | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 12              | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 13              | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 14              | none            | none          | 0.0.0.0    |            | down       | Absent      |
| 15              | none            | none          | 0.0.0.0    |            | down       | Absent      |

## show system information

This command is used to display the status of the PP4X management module: operation mode (uptime, CPU and memory load), temperature on 3 temperature sensors, firmware version, serial number and basic MAC address of the device.

### Syntax

```
show system information <UNIT>
```

### Parameters

<UNIT> – number of unit to which the PP4X module is installed, may take values [1..2].

### Command mode

ROOT

## Example

```
ma4000# show system information 1
System information (1):
 Uptime (d:h:m:s): 0:1:30:39
 CPU load (1/5/15 minutes): 0.00/0.00/0.00
 RAM (total/free), Mbytes: 498/199
 Partition '/' (total/free), Mbytes: 57/22
 Partition '/mnt/tools' (total/free), Mbytes: 1024/887
 Partition '/mnt/config' (total/free), Mbytes: 64/61
 Partition '/mnt/log' (total/free), Mbytes: 128/41
 Temperature (SFP): 26C
 Temperature (CPU): 31C
 Temperature (Switch) : 44C
 Firmware version: 3.22.0.364 r40769 22:55:08 05/11/2014
 Linux version: Linux version 2.6.22.18 (jenkins@xpon.eltex.loc) (gcc version 4.3.2
(sdk3.2rc1-ct-ng-1.4.1)) #1 Thu Nov 6 05:20:35 NOVT 2014
 MAC address: a8:f9:4b:81:85:f0
 Serial number: OL02000056
```

## show uptime

This command displays the operating time of the PP4X control module from the last reboot.

### Syntax

```
show uptime
```

### Parameters

The command contains no arguments.

### Command mode

ROOT

## Example

```
ma4000# show uptime
up 6 days, 3:37
```

## show system environment

This command displays the status of the basket: operation mode and fan speed, voltage and current on the device power feeders.

### Syntax

```
show system environment [detailed]
```

## Parameters

detailed – command used to view the full system information.

## Command mode

ROOT

## Example

```
ma4000# show system environment
MFC board status: ok
MFC board version: 0x2
MFC firmware:
 Status: 0x00 (ok)
 Version: 8 2 1 1 5 05/11/2013
 Timestamp (UTC): 05-Nov-2013 12:19:22

Fan configured speed, %: auto
Fan current speed, %: 36
Fan minimum speed, %: 15
Fan speed levels, %: 15 25 36 46 57 68 78 89 100

 Fan0 Fan1 Fan2
Status: ok ok ok
RPM: 2460 2442 2388

 Feeder1 Feeder2
Status: ok REVERSED
Current, A: 3.50 0.00
Voltage, V: -54.12 2.23

Shelf voltage, V: -54.13
```

## fan min-speed

This command is used to set the minimum fan speed threshold of the device.

The use of a negative form (no) of the command sets the default configuration.

## Syntax

```
fan min-speed <SPEED>
no fan min-speed
```

## Parameters

<SPEED> – rotation speed, may take values [15..100] %.

## Command mode

CONFIG

## Example

```
ma4000(config)# fan min-speed 20
```

## fan speed

This command is used to set a fixed fan speed for the unit or to select a speed automatically.

The use of a negative form (no) of the command sets the default configuration.

### Syntax

```
fan speed <SPEED>
no fan speed
```

### Parameters

<SPEED> – rotation speed, may take values [15..100/auto] %.

### Command mode

CONFIG

## Example

```
ma4000(config)# fan speed 22
```

## fan speed-level

In the automatic adjustment mode, the fan speed changes in steps.

This command is used to set a fixed fan speed for each step of the device.

### Syntax

```
fan speed-level <LEVEL> <SPEED>
```

### Parameters

<LEVEL> – step number, [0..8];

<SPEED> – rotation speed, may take values [15..100/auto] %.

### Command mode

CONFIG

**Example**

```
ma4000(config)# fan speed-level 5 65
```

**fan speed-table**

In the automatic adjustment mode, the fan speed changes in steps.

This command is used to set a fixed fan speed for each step of the device.

The use of a negative form (no) of the command sets the default configuration.

**Syntax**

```
fan speed-table <SPEED> <SPEED> <SPEED> <SPEED> <SPEED> <SPEED> <SPEED> <SPEED>
no fan speed-table
```

**Parameters**

<SPEED> – rotation speed, may take values [15..100/auto] %.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# fan speed-table 17 22 33 44 55 66 77 88 99
```

**slot type**

This command is used to configure the PLC8 module in a specified slot.

The use of a negative form (no) of the command sets the default configuration for the specified slot.

**Syntax**

```
slot <SLOT> type <TYPE>
no slot <SLOT> type
```

**Parameters**

<SLOT> – slot place in the crate, may take values [0..15];

<TYPE> – board type: PLC8.

**Command mode**

CONFIG

## Example

```
ma4000(config)# slot 5 type plc8
```

## slot profile

This command is used to set the PLC8 configuration profile in a specified slot.

The use of a negative form (no) of the command sets the default configuration for the specified board.

## Syntax

```
slot <SLOT> profile <PROFILE> <NAME>
no slot <SLOT> profile <PROFILE>
```

## Parameters

<SLOT> – slot place in the crate, may take values [0..15];

<PROFILE> – profile type: pppoe-ia, dhcp-ra, dhcpv6-ra;

<NAME> – profile name.

## Command mode

CONFIG

## Example

```
ma4000(config)# slot 5 profile pppoe-ia pppoe-ia-00
```

## slot terminal-vlan

This command is used to edit the parameters VLANID, CoS for terminal-vlan used in the configuration of this slot .

The use of a negative form (no) of the command sets the default configuration for the specified board.

## Syntax

```
slot <SLOT> terminal-vlan <NAME> vid <VLAN ID> cos <COS>
no slot <SLOT> terminal-vlan <NAME>
```

## Parameters

<SLOT> – slot place in the crate, may take values [0..15];

<NAME> – Terminal VLAN name. terminal-vlan with this name should be previously created using the **terminal-vlan <NAME>** command;

<VLANID> – VLAN number, may take values [1..4094];

<COS> – CoS value, [0..7, unused].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# slot 6 terminal-vlan INTERNET vid 1123 cos unused
```

**slot logging system loglevel**

This command is used to configure the parameters of syslog message transmission to the system log for configuring this slot.

The use of a negative form (no) of the command sets the default configuration for the specified board.

**Syntax**

```
slot <SLOT> logging system loglevel <LEVEL>
no slot <SLOT> logging system
```

**Parameters**

<SLOT> – slot place in the crate, may take values [0..15];

<LEVEL> – message level, for description see [Table 10.1](#).

Table 10.1 – Alarm message level description

| Level     | Description                                     |
|-----------|-------------------------------------------------|
| emergency | Further operation of the system is not possible |
| alert     | The system requires emergency intervention      |
| critical  | Critical events                                 |
| error     | Operation errors                                |
| warning   | Warnings                                        |
| notice    | Important events during normal operation        |
| info      | Information messages                            |
| debug     | Debug messages                                  |

**Command mode**

CONFIG

## Example

```
ma4000(config)# slot 6 logging system loglevel debug
```

## slot logging module

This command is used to set the syslog-message filtering level for each system module individually in the slot configuration.

The use of a negative form (no) of the command sets the default configuration for the specified board.

## Syntax

```
slot <SLOT> logging module <TYPE> loglevel <LEVEL>
no slot <SLOT> logging module <TYPE>
```

## Parameters

<SLOT> – slot place in the crate, may take values [0..15];

<TYPE> – system module type, for description see [Table 10.2](#);

<LEVEL> – message level, for description see [Table 10.1](#).

Таблица 10.2 - System module description

| Module           | Description                                                        |
|------------------|--------------------------------------------------------------------|
| alarm            | Alarms log message                                                 |
| snmp             | Messages from the SNMP agent                                       |
| pmchal-ipc       | Messages from the pmchal subsystem of interprocess communication   |
| pmchal-gpon      | GPON messages                                                      |
| pmchal-machine   | Messages on operation of state machines for OLT, channels, and ONT |
| pmchal-olt       | OLT general information                                            |
| pmchal-channel   | Information about GPON channels                                    |
| pmchal-ont       | ONT information                                                    |
| pmchal-scheduler | Messages from the scheduler subsystem                              |
| pmchal-rdn       | Messages on GPON channels reservation                              |
| pmchal-dhcpra    | Messages from DHCP Relay Agent                                     |

| Module         | Description                            |
|----------------|----------------------------------------|
| pmchal-pppoeia | Messages from PPPoE Intermediate Agent |

**Command mode**

CONFIG

**Example**

```
ma4000(config)# slot 6 logging module snmp loglevel info
```

**show slot < SLOT > terminal-vlan**

This command is used to view the values of vlanid and CoS for terminal-vlan used in the configuration of this slot .

**Syntax**

```
show slot < SLOT > terminal-vlan
```

**Parameters**

< SLOT > – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using ( - ).

**Command mode**

ROOT

**Example**

```
ma4000#show slot 6 terminal-vlan
Terminal VLAN "t-222":
 VLAN ID: 222
 CoS: unused
```

**show slot < SLOT > gpon olt configuration**

This command is used to view the set PPPoEIA and DHCPRA profiles used in this slot configuration.

**Syntax**

```
show slot < SLOT > gpon olt configuration
```

## Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using (-).

## Command mode

ROOT

## Example

```
ma4000#show slot 6 gpon olt configuration
 Profile pppoe-ia: pppoe-ia-00 OLT Profile PPPoE Intermediate Agent 0
 Profile dhcp-ra: dhcp-ra-00 OLT Profile DHCP Relay Agent 0
 Profile dhcp-ra per VLAN: <list is empty>
```

## show slot <SLOT> cpu detailed<sup>1</sup>

This command is used to view information on the PLC8 linear board processor load.

## Syntax

```
show slot <SLOT> cpu [detailed]
```

detailed – command that will show detailed information when specified.

## Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using (-).

## Command mode

ROOT

## Example

```
ma4000#show slot 6 cpu detailed
```

---

<sup>1</sup> Not implemented in firmware version 3.34.1

## show slot <SLOT> channel-group lacp

This commands are used to view the state of a channel-group of interfaces for communication with central switches PP4X.

## Syntax

```
show slot < SLOT > channel-group lacp
```

## Parameters

< SLOT > – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using (-).

## Command mode

ROOT

## Example

```
ma4000# show slot 6 channel-group lacp
Active Aggregator: 2

Channel group 56 (Aggregator 2)
Number of ports: 2

System Priority: 32768
System MAC: a8:f9:4b:88:4e:60
Key: 0x0e21

Port plc-slot-port 6/0: [active], link up, 10 Gbps, full duplex
Actor Port Partner Port
Port Number: 53 23
Port Priority: 32768 32768
LACP Activity: active active

Port plc-slot-port 6/1: [active], link up, 10 Gbps, full duplex
Actor Port Partner Port
Port Number: 54 83
Port Priority: 32768 32768
LACP Activity: active active
```

## 12 Firmware and configuration management

- [commit](#)
- [confirm](#)
- [restore](#)
- [rollback](#)
- [default](#)
- [firmware select image-alternate](#)
- [firmware select image-current](#)
- [firmware confirm](#)
- [copy](#)
- [cli session-timeout](#)
- [cli max-sessions](#)
- [cli display](#)
- [show cli](#)
- [firmware ont auto update add](#)
- [firmware ont auto update edit](#)
- [firmware ont auto update delete](#)
- [firmware ont auto update immediate](#)
- [firmware ont auto update postpone](#)
- [firmware ont delete image](#)
- [license set](#)
- [show firmware](#)
- [show firmware ont](#)
- [show firmware ont auto update state](#)
- [show firmware ont auto update entries](#)
- [show running-config](#)
- [show candidate-config](#)
- [show boot](#)
- [show startup-config](#)
- [show license](#)

### **commit**

The command allows to apply (enable) the configuration changes. RUNNING configuration is replaced by CANDIDATE configuration. To enable the changes applied, you need to confirm the operation by 'confirm' command during the time period not exceeding the acknowledgement timer lifetime.

### **Syntax**

```
commit
```

### **Parameters**

Command contains no arguments.

### **Command mode**

ROOT

## Example

```
ma4000# commit
```

The configuration changes made in the current CLI transaction are applied.

## confirm

The command is intended for confirmation of configuration application. If no confirmation has been entered within the specified time (**commit**), after a configuration change, the last confirmed configuration will be automatically restored. Automatic rollback system totally prevents loss of connection with the device.

## Command syntax

```
confirm
```

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# confirm
```

Confirmation of configuration changes.

## restore

This command allows to cancel the unconfirmed application of the configuration and return to the last confirmed one. Undoing changes can be carried out only until 'confirm' command is entered. When executing 'restore' command, there is a loss of unconfirmed configuration.

## Syntax

```
restore [configuration section]
```

## Parameters

[configuration section] – configuration section, optional parameter:

- pp4x – central switch management;
- profiles olt – PLC board OLT profiles;
- profiles ont – ONT board profiles;

Slot <NUMBER> – interface module configuration, where <NUMBER> – interface module number, may take values [0 .. 15].

**Command mode**

ROOT

**Example**

```
ma4000# restore
```

Reversion to the last confirmed configuration is completed.

**rollback**

The command allows to cancel not applied configuration changes. As a result of command execution, CANDIDATE configuration will be deleted. The command may be used only until the **commit** command is entered.

**Syntax**

```
rollback
```

**Parameters**

Command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# rollback
```

Cancellation of all not applied configuration changes has been made.

**default**

This command resets the configuration to default.

The command without specifying a parameter (configuration section) will be executed for the whole device, otherwise the factory default values will be set for the corresponding parts of the configuration.

Global reset of the configuration to the factory settings deletes all log files, ONT files, PP4/profile/slot configuration files.

The command requires confirmation.

**Syntax**

```
default [configuration section]
```

## Parameters

[configuration section] – configuration section, optional parameter:

- pp4x – central switch management;
- profiles olt – PLC board OLT profiles;
- profiles ont – PLC board ONT profiles.

Slot <NUMBER> – interface module configuration, where <NUMBER> – interface module number, may take values [0 .. 15].

## Command mode

ROOT

## Example

```
ma4000# default
```

Entire candidate configuration will be reset to **default**, all settings will be lost upon commit. Additional firmware will be deleted.

Do you really want to **continue** ? y

Candidate configuration **for** PLC8/ONT profiles has been reset to **default**

Candidate configuration **for** PLC8/OLT profiles has been reset to **default**

Candidate configuration **for** slot 0 has been reset to **default**

Candidate configuration **for** slot 1 has been reset to **default**

Candidate configuration **for** slot 2 has been reset to **default**

Candidate configuration **for** slot 3 has been reset to **default**

Candidate configuration **for** slot 4 has been reset to **default**

Candidate configuration **for** slot 5 has been reset to **default**

Candidate configuration **for** slot 6 has been reset to **default**

Candidate configuration **for** slot 7 has been reset to **default**

Candidate configuration **for** slot 8 has been reset to **default**

Candidate configuration **for** slot 9 has been reset to **default**

Candidate configuration **for** slot 10 has been reset to **default**

Candidate configuration **for** slot 11 has been reset to **default**

Candidate configuration **for** slot 12 has been reset to **default**

Candidate configuration **for** slot 13 has been reset to **default**

Candidate configuration **for** slot 14 has been reset to **default**

Candidate configuration **for** slot 15 has been reset to **default**

PP4X candidate configuration has been reset to **default**.

Boot candidate configuration has been reset to **default**.

Firmware deleting finished.

load-1-3-2#

All configuration sections have been reset to their original state.

## firmware select image-alternate

The command is designed to change the active firmware image during reboot on a specified PP4X module.

## Syntax

firmware select image-alternate [unit <NUMBER>]

**Parameters**

<NUMBER> – PP4X module number, may take values [1 .. 2].

The command requires confirmation.

**Command mode**

ROOT

**Example**

```
ma4000# firmware select image-alternate unit 1
WARNING: operations with concrete unit aren't safe !!!
 Set image 1 as active on unit 1? (y/N) y
Verifying image 1 on unit 1, please wait...
Updating unit 1...
Firmware image 1 on unit 1 has been selected as the active image.
When the unit is booted next time, it will use image 1.
You will need to confirm that the active image on the unit is working properly
by entering 'firmware pp4x confirm unit 1' command.
If the command will not be entered in 10 minutes after the unit has booted,
the unit will automatically reboot,
and image 0 will be selected as the active image.
Request complete.
```

**firmware select image-current**

The command is designed to select the current firmware image at reboot on a specified PP4X module. The command is executed if the next reboot is required from the current image, if the image was previously changed with the **firmware select image-alternate** commands.

The command requires confirmation.

**Syntax**

```
firmware select image-current [unit <NUMBER>]
```

**Parameters**

<NUMBER> – module number, may take values [1 .. 2].

**Command mode**

ROOT

## Example

```
ma4000# firmware select image-current unit 1
WARNING: operations with concrete unit aren't safe !!!
WARNING: firmware upgrade is already in progress on unit 1.
```

To complete the firmware upgrade, **do** the following.

1. Reboot the unit.
2. Confirm that the active image on the unit is working properly by entering 'firmware pp4x confirm unit 1' command. If the command will not be entered in 10 minutes after the unit has rebooted, the unit will automatically reboot, and image 0 will be selected as the active image.

Alternatively, you may choose to ignore **this** warning and proceed with the command to set another image as active.

```
Proceed? y
Set image 0 as active on unit 1? (y/N) y
Verifying image 0 on unit 1, please wait...
Updating unit 1...
Firmware image 0 on unit 1 has been selected as the active image.
When the unit is booted next time, it will use image 0.
Request complete.
```

## firmware confirm

The command is used to confirm the correctness of firmware download.

### Syntax

```
firmware confirm
```

### Parameters

Command contains no arguments.

### Command mode

ROOT

## Example

```
ma4000# firmware confirm
```

## copy

This command allows:

- copy the file from the TFTP server to the Flash memory of the device;
- copy the file from the Flash memory of the device to the TFTP server;

- copy the file from the Flash memory of the first device to the Flash memory of the second device;

## Syntax

copy <SOURCE-URL> <DESTINATION-URL>

## Parameters

<SOURCE-URL> – URL source, specified as:

| src:                                   | Description                                                                                                                            |
|----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">tftp://HOST/FILE</a>       | Download FILE file from external TFTP server of HOST to the device.                                                                    |
| <a href="#">http://HOST/FILE</a>       | Download FILE file from external HTTP server of HOST to the device.                                                                    |
| <a href="#">fs://backup</a>            | Download the system configuration file in text form.                                                                                   |
| <a href="#">fs://backup.gz</a>         | Download the system configuration file as tar.gz archive.                                                                              |
| <a href="#">fs://backup.zip</a>        | Download the system configuration file as ZIP archive.                                                                                 |
| <a href="#">fs://pp-config</a>         | Download the management module (PP4X) configuration file in text form.                                                                 |
| <a href="#">fs://slot-config/SLOT</a>  | Download the line card (PLC8) configuration file in text form.                                                                         |
| <a href="#">fs://profiles/olt</a>      | Download the OLT profile configuration file in text form.                                                                              |
| <a href="#">fs://profiles/ont</a>      | Download the ONT profile configuration file in text form.                                                                              |
| <a href="#">fs://log/FILE</a>          | Upload the FILE log file (see the list of existing files: show log) to an external TFTP server. Files from the /tmp/syslog/ directory. |
| <a href="#">fs://logfs/FILE</a>        | Upload the FILE log file to an external TFTP server. Files from the /mnt/log/ directory.                                               |
| <a href="#">fs://ont-firmware/FILE</a> | Upload the ONT firmware file to an external TFTP server.                                                                               |

<DESTINATION-URL> – destination URL, specified as:

| dst :                            | Description                                                                       |
|----------------------------------|-----------------------------------------------------------------------------------|
| <a href="#">tftp://HOST/FILE</a> | Upload the FILE file to an external TFTP server of the HOST.                      |
| <a href="#">http://HOST/FILE</a> | Upload the FILE file to an external HTTP server of the HOST.                      |
| <a href="#">fs://backup</a>      | Restore system configuration from an uploaded file.                               |
| <a href="#">fs://pp-config</a>   | Restore the configuration of the management module (PP4X) from the uploaded file. |

| dst :                                 | Description                                                        |
|---------------------------------------|--------------------------------------------------------------------|
| <a href="#">fs://slot-config/SLOT</a> | Restore the line card configuration (PLC8) from the uploaded file. |
| <a href="#">fs://profiles/olt</a>     | Restore OLT profile configuration from an uploaded file.           |
| <a href="#">fs://profiles/ont</a>     | Restore ONT profile configuration from an uploaded file.           |
| <a href="#">fs://ont-firmware</a>     | Upload the ONT firmware file to the system.                        |
| <a href="#">fs://firmware</a>         | Upload and install the system firmware to an inactive partition.   |
| <a href="#">fs://license</a>          | Upload and install the license                                     |

## Command mode

ROOT

## Example

```
ma4000# copy tftp://192.168.16.100/firmware.3.22.0.372.ma4k fs://firmware
Source:
Protocol: 'tftp'
Hostname: '192.168.16.100'
Path: 'firmware.3.22.0.372.ma4k'
Filename: 'firmware.3.22.0.372.ma4k'
Destination:
Protocol: 'fs'
Kind: container
Copying file from host 192.168.16.100, remote path firmware.3.22.0.372.ma4k...
Copying file: done (rc 0).
Installing firmware, please wait...
Firmware installation finished.
```

## cli session-timeout

This command sets the inactivity time, after which the exit from the current CLI-session will be performed.

## Syntax

```
cli session-timeout <TIME>
```

## Parameters

<TIME> – inactivity time, after which the CLI connection will be closed, [0..2103840] minutes.

## Command mode

CONFIG

## Example

```
ma4000(config)# cli session-timeout 1
```

## cli max-sessions

This command sets the maximum number of CLI sessions running simultaneously.

### Syntax

```
cli max-sessions <VALUE>
```

### Parameters

<VALUE> – maximum number of CLI sessions running simultaneously, [1..10].

### Command mode

CONFIG

## Example

```
ma4000(config)# cli max-sessions 5
```

## cli display

This command sets the output format of the ONT list (show interface ont <SLOT/PORT/ONTID> connected and show interface ont <SLOT/PORT/ONTID> online)

### Syntax

```
cli display <VALUE>
```

### Parameters

<VALUE> – customizable display field:

- config-password – ONT OMCI password, specified in configuration
- description – ONT description
- equipment-id – board type
- gpon-port – channel number to which the ONT is connected
- profile <PROFILE> – ONT profiles:

<PROFILE> – profile type: cross-connect, dba, management, ports, scripting, shaping

- received-password – ONT OMCI password, specified on ONT
- rssi – level of received signal on OLT from ONT
- status – ONT operation status
- version – firmware version

**Command mode**

CONFIG

**Example**

```
ma4000(config)# cli display profile cross-connect
```

**show cli**

This command is used to view CLI settings (maximum number and inactivity time for CLI sessions, display show interface ont <SLOT/PORT/ONTID> connected and show interface ont <SLOT/PORT/ONTID> online command fields).

**Syntax**

```
show cli
```

**Parameters**

Command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show cli
 Cli:
 Session timeout: 5
 Confirm timeout: 5
 Custom ont list showing:
 Ont id: enabled
 GPON-port: enabled
 Enabled: enabled
 Rssi: enabled
 Profile shaping: disabled
 Profile ports: disabled
 Profile management: disabled
 Profile scripting: disabled
 Profile cross connect: enabled
 Profile DBA: disabled
 Version: enabled
 Equipment id: enabled
 Config password: disabled
 Received password: enabled
 Description: enabled
 Max session: 10
```

## firmware ont auto update add

This command adds an entry to the list of rules for ONT firmware autoupdate.

### Syntax

```
firmware ont auto update add <NAME> <TYPE> <VERSION> <FILE>
```

### Parameters

<NAME> – entry name, specified as the string of up to 32 characters;

<TYPE> – ONT type, specified as the string of up to 32 characters;

<VERSION> – ONT firmware version, that requires update;

<FILE> – the name of the ONT firmware file that will be used to update ONT.

### Command mode

ROOT

### Example

```
ma4000# firmware ont auto update add entry 1 NTP-RG 1.1.1 firmware.bin
```

## firmware ont auto update edit

This command edits the entry from the list of rules for ONT firmware autoupdate.

### Syntax

```
firmware ont auto update edit<NAME> <TYPE> <VERSION> <FILE>
```

### Parameters

<NAME> – entry name, specified as the string of up to 32 characters;

<TYPE> – ONT type, specified as the string of up to 32 characters;

<VERSION> – ONT firmware version, that requires update;

<FILE> – the name of the ONT firmware file that will be used to update ONT.

### Command mode

ROOT

### Example

```
ma4000# firmware ont auto update edit 1 NTP-RG 1.1.1 firmware.bin
```

## firmware ont auto update delete

This command removes a certain entry from the list of ONT firmware autoupdate rules.

### Syntax

```
firmware ont auto update delete <NAME>
```

### Parameters

<NAME> – entry name, specified as the string of up to 32 characters;

### Command mode

ROOT

### Example

```
ma4000# firmware ont auto update delete entry 1
```

## firmware ont auto update immediate

This command enables the mode of immediate updating of all connected ONTs.

### Syntax

```
firmware ont auto update immediate
```

### Parameters

The command contains no arguments.

### Command mode

ROOT

### Example

```
ma4000# firmware ont auto update immediate
```

## firmware ont auto update postpone

This command enables the delayed ONT update mode when they are connected.

### Syntax

```
firmware ont auto update postpone
```

### Parameters

The command contains no arguments.

### Command mode

ROOT

### Example

```
ma4000# firmware ont auto update postpone
```

### firmware ont delete image

This command removes ONT files from the system.

### Syntax

```
firmware ont delete image [<NAME>]
```

### Parameters

[<NAME>] – file name, the list of available files can be viewed by the **show firmware ont** command.

### Command mode

ROOT

### Example

```
ma4000# firmware ont delete image ntp-rg-1.2.3.bin
```

### license set

This command is used to upload the license file to the basket.

### Syntax

```
license set “”<LICENSE>””
```

### Parameters

<LICENSE> – full content of the license file obtained from the representative of Eltex Enterprise Ltd.

### Command mode

ROOT

## Example

```
ma4000# MA_135_load# license set ""LICENSE""
License successfully installed.
MA_135_load#
```

## show firmware

This command displays the available firmware on the PP4X modules:

- Unit – PP4X module number;
- Image – firmware file ID;
- Running – specifies whether this configuration is current (yes/no);
- Boot – \* specifies the firmware file that will be selected the next time the system boots up;
- Version – firmware version;
- Date – firmware date.

## Syntax

```
show firmware
```

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show firmware

Firmware status:
~~~~~
Unit   Image   Running   Boot      Version          Date
----   -
1      0        Yes       *          1 3 2 372 40813  07-Nov-2014 08:59:37
1      1        No        *          1 3 2 372 40813  07-Nov-2014 08:59:37
2      0        No        *          1 3 2 372 40813  07-Nov-2014 08:59:37
2      1        Yes       *          1 3 2 372 40813  07-Nov-2014 08:59:37

"*" designates that the image was selected for the next bootshow configuration
```

The command is used to display the current boot configuration.

## show firmware ont

This command displays the list of ONT firmware files uploaded to the system.

**Syntax**

```
show firmware ont
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# load-1-3-2# show firmware ont
```

```
ONT firmware images:
```

```
~~~~~
```

| # | Filename                       | Version | Hardware |
|---|--------------------------------|---------|----------|
| 1 | ntu-rg-3.50.2.1208.fw.bin      |         |          |
| 2 | ntu-2-3.50.2.1193.fw.bin       |         |          |
| 3 | ntp-rg-revc-d3.20.2.471.fw.bin |         |          |

**show firmware ont auto update state**

This command is used to view the set ONT autoupdate mode (immediate or postpone).

**Syntax**

```
show firmware ont auto update state
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show firmware ont auto update state
Auto-update ONT: immediate
```

## show firmware ont auto update entries

This command is used to view the list of rules for ONT firmware autoupdate.

### Syntax

```
show firmware ont auto update entries
```

### Parameters

The command contains no arguments.

### Command mode

ROOT

### Example

```
ma4000# show firmware ont auto update entries
```

## show running-config

This command is used to view the current system configuration.

### Syntax

```
show running-config [PARAM] [VALUE]
```

### Parameters

|                  |                                                                                                            |
|------------------|------------------------------------------------------------------------------------------------------------|
| aaa              | displays current authentication, authorization and accounting settings to manage user access to the system |
| access-list      | displays the current access list settings                                                                  |
| alarm            | displays the current configuration of all alarms                                                           |
| all              | displays the full current configuration of the system                                                      |
| backup           | displays current device backup configuration settings                                                      |
| backup-interface | displays current backup interface settings                                                                 |
| cli              | displays the current command line configuration                                                            |
| clock            | displays the current clock settings                                                                        |
| dhclient         | displays the current configuration                                                                         |

|                |                                                                                                              |
|----------------|--------------------------------------------------------------------------------------------------------------|
| diff           | when specifying this command, the running-candidate configuration differences are viewed, optional parameter |
| fan            | displays the current profile configuration for fans                                                          |
| gpon           | displays the current configuration of the GPON interface module                                              |
| hostname       | displays the current device name                                                                             |
| interface      | displays the current configuration of the selected interface                                                 |
| ip             | displays the current IP routing settings                                                                     |
| ipv6           | displays the current IPv6 settings                                                                           |
| isolation      | displays the current configuration of isolation groups                                                       |
| journal        | displays the current configuration for the event log                                                         |
| lldp           | displays the current LLDP settings                                                                           |
| logging        | displays the current configuration for the system log                                                        |
| mac            | displays the current configuration for the MAC tables                                                        |
| management     | displays the current network settings of the ma4000                                                          |
| mirror         | displays current interface mirroring settings                                                                |
| port-channel   | displays the current configuration of the selected logical interface                                         |
| profile        | displays the current configuration of the selected profile                                                   |
| qos            | displays the current configuration of the qos-mapping                                                        |
| selective-qinq | displays current Selective Q-in-Q settings                                                                   |
| shelf          | displays the current basket slot configuration                                                               |
| slot           | displays the current configuration of the selected slot                                                      |
| spanning-tree  | displays the current STP configuration                                                                       |
| stack          | displays the current configuration of the stack interface                                                    |
| template       | displays the current configuration of the ONT configuration templates                                        |
| user           | displays the current configuration for CLI users                                                             |

|      |                                                 |
|------|-------------------------------------------------|
| vlan | displays the current configuration for the VLAN |
|------|-------------------------------------------------|

## Command mode

ROOT

## Example

```
ma4000# show running-config
hostname ma4000
management ip 192.168.205.234 255.255.255.0
management gateway 192.168.205.230
management vlan 205
```

## show candidate-config

This command is used to view the configuration that will be set after the settings are applied (commit command).

## Syntax

show candidate-config

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show candidate-config
hostname ma4000

management ip 192.168.205.234 255.255.255.0
management gateway 192.168.205.230
management vlan 205
```

## show boot

This command allows to view the basic configuration during system boot: device name, VLAN control, debug message management during boot.

## Syntax

show boot <CONFIG>

**Parameters**

<CONFIG> – configuration type:

- running-config – basic configuration from which the system will be booted;
- candidate-config – basic configuration candidate.

**Command mode**

ROOT

**Example**

```
ma4000# show boot candidate-config
unit 1
priority 240
object load-1-3-2_2
management-vlan 199
boot-proto no
debug-mode 08 00 24 D0 11 38 61 00 00 00 00 00
```

**show startup-config**

This command allows to view the configuration to be applied after rebooting the device.

**Syntax**

show startup-config

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show startup-config
hostname ma4000

no lldp enable

management ip 192.168.205.234 255.255.255.0
management gateway 192.168.205.230
management vlan 205
```

## show license

This command displays information about the installed license:

- License valid – license validity;
- Carrier – name of the carrier that received the license;
- Licensed ONT count – the maximum number of simultaneously connected licensed ONTs;
- Licensed ONT online – number of connected licensed ONTs;
- SN – the list of serial numbers of PP4X boards supported in the license;
- Mac – list of mac addresses supported in PP4X boards license.

## Syntax

```
show license
```

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show license
Active license information:
 License valid: yes
 Version: 1.1
 Carrier: Eltex Enterprise LLC
 Licensed vendor: all
 Licensed ONT count: unlimited
 Licensed ONT online: 2
 SN: OL02000000
 Mac: A8:F9:4B:00:00:00
```

## 13 Device operation debugging

- debug arp
- debug alarm
- debug bonding
- debug boot debug-mode
- debug boot priority
- debug boot unit
- debug cfg-manager
- debug clish
- debug clish-completion
- debug clish-errors
- debug clish-infos
- debug clish-manager
- debug clish-pty
- debug clish-sockets
- debug clish-timers
- debug commands-dump
- debug copy
- debug cscd
- debug cscd election
- debug cscd reserve
- debug cscd topology
- debug dev-exchange sctp-notification
- debug dev-exchange
- debug dhcp
- debug events
- debug fan
- debug firmware
- debug ifm
- debug igmp
- debug iprouting
- debug lacp
- debug license-manager
- debug lldp
- debug locks
- debug mac-sync duplicate-mac
- debug mac-sync sctp-notification
- debug maep-manager
- debug memory
- debug memory pmchal
- debug network
- debug packet
- debug port-states
- debug resources pmchal
- debug sctp
- debug snmp packet
- debug snmp
- debug snmpman
- debug sntp
- debug spanning-tree
- debug stack elections
- stack reserve-channel
- debug storage garbage-collect
- debug syslog

- [debug systemdb](#)
- [debug test](#)
- [debug test-link](#)
- [debug top-manager](#)
- [debug vlan](#)
- [debug vlan pvid](#)
- [debug vlan-manager](#)

### **debug arp**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug alarm**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug bonding**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug boot debug-mode**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug boot priority**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug boot unit**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug cfg-manager**

This command allows to enable debugging of the configuration manager process trace.

The use of a negative form (no) of the command disables debug traces.

### **Syntax**

```
[no] debug cfg-manager [<PARAM>]
```

## Parameters

<PARAM> – allocated action:

- none – enable both types of trace;
- errors – enable enhanced traces for errors;
- routine – enable enhanced traces for standard events.

## Command mode

ROOT

## Example

```
ma4000# debug cfg-manager errors
```

## debug clish

This command enables CLI interface debug traces.

The use of the negative form (no) of the command disables CLI interface debug traces.

## Syntax

[no] debug clish

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# debug clish
```

## debug clish-completion

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

## debug clish-errors

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug clish-infos**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug clish-manager**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug clish-ptype**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug clish-sockets**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug clish-timers**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug commands-dump**

This command allows to display the entire CLI command structure.

**Syntax**

```
debug commands-dump
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# debug commands-dump
```

## **debug copy**

This command enables debug messages output when copying configuration/firmware files from/to external TFTP server.

The use of the negative form (no) of the command disables debug messages output when copying configuration/firmware files.

### **Syntax**

```
[no] debug copy
```

### **Parameters**

The command contains no arguments.

### **Command mode**

ROOT

### **Example**

```
ma4000# debug copy
```

## **debug cscd**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug cscd election**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug cscd reserve**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug cscd topology**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug dev-exchange sctp-notification**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

## debug dev-exchange

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

## debug dhcp

This command sets permission to output debug messages of DHCP-client/server/relay operation.

The use of the negative form (no) of the command disables debug messages of specified DHCP type.

### Syntax

```
[no] debug dhcp <PARAM>
```

### Parameters

<PARAM> – message type:

- client – DHCP client messages,
- common – common messages;
- errors – error messages;
- proxy – DHCP agent messages;
- server –DHCP server messages.

### Command mode

ROOT

### Example

```
ma4000# debug dhcp client
```

## debug events

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

## debug fan

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

## debug firmware

This command allows to output debug messages when updating the subsystem firmware.

The use of the negative form (no) of the command disables debug messages output.

**Syntax**

```
[no] debug firmware
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000(pp4x)# debug firmware
```

**debug ifm**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug igmp**

This command allows to output debug messages when processing IGMP packets.

The use of the negative form (no) of the command disables debug messages output.

**Syntax**

```
debug igmp <ACT>
[no] debug igmp <ACT>
```

**Parameters**

<ACT> – allocated action:

- fdb – enable output of debugging traces when accessing the IGMP-protocol database;
- group – enable output of debugging traces to events occurring with IGMP groups;
- packet – enable output of debugging traces when receiving/sending IGMP packets.

**Command mode**

ROOT

**Example**

```
ma4000(pp4x)# debug igmp fdb
```

## debug iprouting

This command allows to output debug messages when processing packet routing.  
The use of a negative form (no) of the command disables debug traces.

### Syntax

```
debug iprouting <TYPE>
[no] debug iprouting <TYPE>
```

### Parameters

<TYPE> – event type:

- common – enable common event debugging;
- errors – enable error event debugging.

### Command mode

ROOT

### Example

```
ma4000# debug iprouting common
```

## debug lacp

This command enables the output of debug messages when processing LACP packets.  
The use of the negative form (no) of the command disables debug messages output.  
If parameter is not specified, the debug messages will be enabled for all LAC protocol events.

### Syntax

```
debug lacp <ACT>
[no] debug lacp <ACT>
```

### Parameters

<ACT> – allocated action:

- packet – enable debugging when sending/receiving LACP frames;
- port-channel – enable debug messages of LACP for a given group of LAG aggregation of external uplink interfaces, [1 .. 8];
- slot-channel – enable debug messages of the LACP for a given group of LAG interfaces aggregation to connect modules of linear interfaces, [0 .. 15].

### Command mode

ROOT

**Example**

```
ma4000(pp4x)# debug lacp packet
```

**debug license-manager**

The command enables displaying of license manager debug messages.

The use of the negative form (no) of the command disables debug messages output.

**Syntax**

```
[no] debug licence-manager <TYPE>
```

**Parameters**

<TYPE> – type of events:

- common – enable common event debugging;
- errors – enable error event debugging.

**Command mode**

ROOT

**Example**

```
ma4000(pp4x)# debug license-manager common
```

**debug lldp**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug locks**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug mac-sync duplicate-mac**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug mac-sync sctp-notification**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug maep-manager**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug memory**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug memory pmchal**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug network**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug packet**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug port-states**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug resources pmchal**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug sctp**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

### **debug snmp packet**

This command enables output of debug messages when exchanging via SNMP.

The use of the negative form of the command (no) disables debug messages.

**Syntax**

```
[no] debug snmp packet
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# debug snmp packet
```

**debug snmp**

This command enables output of SNMP agent debug messages.

The use of the negative form of the command (no) disables debug messages.

**Syntax**

```
[no] debug snmp
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000(pp4x)# debug snmp
```

**debug snmpman**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug sntp**

This command enables output of debug traces when the NTP time synchronization protocol is running.

The use of the negative form (no) of the command disables debug messages output.

**Syntax**

```
[no] debug sntp
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# debug sntp
```

**debug spanning-tree**

This command enables the output of debug traces when processing STP/RSTP packets.

The use of the negative form (no) of the command disables debug traces output.

**Syntax**

```
[no] debug spanning-tree <TYPE>
```

**Parameters**

<TYPE> – type:

- common;
- errors;
- sync – synchronization.

**Command mode**

ROOT

**Example**

```
ma4000# debug spanning-tree errors
```

**debug stack elections**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**stack reserve-channel**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug storage garbage-collect**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug syslog**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug systemdb**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug test**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug test-link**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug top-manager**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug vlan**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug vlan pvid**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

**debug vlan-manager**

This command allows you to enable additional output of debug messages.

The full description of the functionality can be clarified by contacting the technical support service.

## 14 Creating a configuration backup (BACKUP CONFIG)

- [backup onchange](#)
- [backup ontimer](#)
- [backup ontimer-period](#)
- [backup path](#)

The commands of the 'backup' family allow saving the configuration of the entire MA4000-PX basket to a remote TFTP-server, as well as setting up the configuration archiving parameters.

Uploading and downloading configuration files in manual mode is described in 'copy' section.

### backup onchange

This command automatically downloads the configuration file every time the configuration is changed (commit).

The use of the negative form (no) of the command disables configuration file downloading.

#### Syntax

```
[no] backup onchange
```

#### Parameters

The command contains no arguments.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# backup onchange
```

### backup ontimer

This command automatically downloads the configuration file at a specified interval.

The use of the negative form (no) of the command disables configuration file downloading.

#### Syntax

```
[no] backup ontimer
```

#### Parameters

The command contains no arguments.

#### Command mode

CONFIG

## Example

```
ma4000(config)# backup ontimer
```

## backup ontimer-period

This command automatically downloads the configuration file at a specified interval.

### Syntax

```
backup ontimer-period [<INTERVAL>]
```

### Parameters

<INTERVAL> – downloading interval, may take values [600..32000000] seconds.

### Command mode

CONFIG

## Example

```
ma4000(config)# backup ontimer-period 600
```

## backup path

This command specifies the path for saving the configuration archives.

The use of the negative form (no) of the command cancels the previously specified path for saving the configuration archives.

### Syntax

```
backup path <PATH>
no backup path
```

### Parameters

<PATH> – string in format of: tftp://<ip>/<tftpdirectiory>.

### Command mode

CONFIG

## Example

```
ma4000(config)# backup path tftp://192.168.18.252/pp4x/
```

## 15 Remote access configuration

- [ip ssh server](#)
- [ip telnet port](#)
- [ip telnet server](#)
- [show ip ssh](#)
- [show ip telnet](#)
- [aaa authentication login](#)
- [line](#)
- [login authentication](#)
- [enable authentication](#)
- [tacacs-server timeout](#)
- [tacacs-server key](#)
- [tacacs-server encrypted key](#)
- [tacacs-server host](#)
- [radius-server timeout](#)
- [radius-server key](#)
- [radius-server encrypted key](#)
- [radius-server host](#)
- [aaa accounting commands tacacs+](#)
- [aaa accounting start-stop tacacs+](#)
- [show authentication methods](#)
- [show tacacs](#)
- [show accounting](#)

### ip ssh server

This command enables the server to manage the device with access via SSH protocol.

The use of the negative form (no) of the command disables the server to manage the device with access via SSH protocol.

#### Syntax

[no] ip ssh server

#### Parameters

The command contains no arguments.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# ip ssh server
```

### ip telnet port

This command specifies the port for the Telnet server.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
ip telnet port <PORT>
no ip telnet port
```

**Parameters**

<PORT> – port number, takes values of [1..65535].

**Default value**

23

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip telnet port 24
```

**ip telnet server**

This command enables the server to manage the device with access via Telnet protocol.

The use of a negative form (no) of the command disables Telnet server.

**Syntax**

```
[no] ip telnet server
```

**Parameters**

The command contains no arguments.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip telnet server
```

**show ip ssh**

This command allows to view information about the state of the SSH-server (allowed/denied).

**Syntax**

```
show ip ssh
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show ip ssh
SSH server state: enabled
```

**show ip telnet**

This command allows to view information about the status of the Telnet server and the port number from which the Telnet connection is available.

**Syntax**

```
show ip telnet
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show ip telnet
Telnet server state: enabled
port : 23
```

**aaa authentication login**

This command sets the authentication method for logging in. The created lists can be used to specify an authentication method different from the default authentication method for a certain type of connection (console, telnet, ssh).

In order for users authenticated through the TACACS+/RADIUS server to be able to edit the system configuration, a local user named 'remote' must be given the appropriate rights.

The use of the negative form (no) of the command enables local authentication.

**Syntax**

```
[no] aaa authentication login default <METHODS>
[no] aaa authentication login list <LIST NAME> <METHODS>
```

**Parameters**

<METHODS> – authentication methods list, which may contain one or more of following values:

- tacacs+ – use TACACS+ server for authentication;
- radius – use RADIUS server for authentication.

<LISTNAME> – authentication list name.

**Default value**

local

**Command mode**

CONFIG

**Example**

```
ma4000(config)# aaa authentication login default tacacs+ local
```

**line**

This command is used to enter the configuration mode of a certain type of connection.

**Syntax**

```
line <TYPE>
```

**Parameters**

<TYPE> – connection type, may take one of the following values:

- console – connection via console;
- telnet – connection via Telnet;
- ssh – connection via SSH.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# line console
ma4000(pp4x-config-line-console)#
```

## login authentication

This command sets the authentication method to log in for a specific type of connection (console, telnet, ssh). The use of a negative form (no) of the command sets the default value.

### Syntax

```
[no] login authentication <NAME>
```

### Parameters

<NAME> – authentication methods list name. Takes the 'default' value and the names of the lists created by the user.

### Default value

default

list

### Command mode

CONFIGURE LINE

### Example

```
ma4000(pp4x-config-line-console)# login authentication mylist
```

## enable authentication

This command sets the authentication method to obtain privileged access for a specific type of connection (console, telnet, ssh).

The use of a negative form (no) of the command sets the default value.

### Syntax

```
[no] enable authentication <NAME>
```

### Parameters

<NAME> – authentication methods list name. Takes the 'default' value and the names of the lists created by the user.

### Default value

default

list

**Command mode**

CONFIGURE LINE

**Example**

```
ma4000(pp4x-config-line-console)# enable authentication enable list
```

**tacacs-server timeout**

This command sets the default time to wait for a response from the TACACS+ server.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
[no] tacacs-server timeout <TIMEOUT>
```

**Parameters**

<TIMEOUT> – TACACS+ server response waiting time, may take values [1..30] seconds.

**Default value**

5 seconds

**Command mode**

CONFIG

**Example**

```
ma4000(config)# tacacs-server timeout 10
```

**tacacs-server key**

This command sets the default key for authentication and encryption of data between the device and the TACACS+ server.

The use of a negative form (no) of the command removes the default key.

**Syntax**

```
[no] tacacs-server key <KEY>
```

**Parameters**

<KEY> – authentication key, contain [1..64] characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# tacacs-server key 12345
```

**tacacs-server encrypted key**

This command sets the default key for authentication and encryption of data between the device and the TACACS+ server in encrypted state.

The use of a negative form (no) of the command removes the default key.

**Syntax**

```
[no] tacacs-server encrypted key <KEY>
```

**Parameters**

<KEY> – authentication key, contain [1..128] characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# tacacs-server encrypted key 98C7D37909
```

**tacacs-server host**

This command adds the specified server to the list of used TACACS+ servers and moves to the configuration mode of a certain TACACS+ server.

The use of a negative form (no) of the command removes server.

**Syntax**

```
[no] tacacs-server host <IP>
```

**Parameters**

<IP> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

In the server configuration mode, the following parameters can also be set:

- timeout – server response waiting time, may take values [1..30] seconds;
- port-number – number of port to exchange data with a server, takes values of [1..65535];

- key – key for authentication and encryption of data between the device and the TACACS+ server, the key contains [1...64] characters;
- encrypted key – key for authentication and encryption of data between the device and the TACACS+ server in encrypted form, the key contains [1...128] characters;
- priority – remote server priority, takes values in the range of [0-65535].

## Command mode

CONFIG

## Example

```
ma4000(config)# tacacs-server host 10.10.10.10
ma4000(pp4x-config-tacacs)# key 123
ma4000(pp4x-config-tacacs)# timeout 12
ma4000(pp4x-config-tacacs)# priority 0
ma4000(pp4x-config-tacacs)# port-number 3000
```

## radius-server timeout

This command sets the default time to wait for a response from the RADIUS server.

The use of a negative form (no) of the command sets the default value.

## Syntax

[no] radius-server timeout <TIMEOUT>

## Parameters

<TIMEOUT> – RADIUS server response waiting time, may take values [1..30] seconds.

## Default value

5 seconds

## Command mode

CONFIG

## Example

```
ma4000(config)# radius-server timeout 10
```

## radius-server key

This command sets the default key for authentication and encryption of data between the device and the RADIUS server.

The use of a negative form (no) of the command removes the default key.

**Syntax**

```
[no] radius-server key <KEY>
```

**Parameters**

<KEY> – authentication key, contain [1..64] characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# radius-server key 12345
```

**radius-server encrypted key**

This command sets the default key for authentication and encryption of data between the device and the RADIUS server in encrypted state.

The use of a negative form (no) of the command removes the default key.

**Syntax**

```
[no] radius-server encrypted key <KEY>
```

**Parameters**

<KEY> – authentication key, contain [1..128] characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# radius-server encrypted key 98C7D37909
```

**radius-server host**

This command adds the specified server to the list of used RADIUS servers and moves to the configuration mode of a certain RADIUS server.

The use of a negative form (no) of the command removes server.

**Syntax**

```
[no] radius-server host <IP>
```

## Parameters

<IP> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

In the server configuration mode, the following parameters can also be set:

- timeout – server response waiting time, may take values [1..30] seconds;
- port-number – number of port to exchange data with a server, takes values of [1..65535];
- key – key for authentication and encryption of data between the device and the RADIUS server, the key contains [1...64] characters;
- encrypted key – key for authentication and encryption of data between the device and the RADIUS server in encrypted form, the key contains [1...128] characters;
- priority – remote server priority, takes values in the range of [0-65535].

## Command mode

CONFIG

## Example

```
ma4000(config)# radius-server host 10.10.10.10
ma4000(pp4x-config-radius)# key 123
ma4000(pp4x-config-radius)# timeout 12
ma4000(pp4x-config-radius)# priority 0
ma4000(pp4x-config-radius)# port-number 3000
```

## aaa accounting commands tacacs+

This command enables keeping records of the commands entered by the user.

The use of a negative form (no) of the command sets the default value. By default is disabled.

## Syntax

[no] aaa accounting commands tacacs+

## Parameters

The command contains no arguments.

## Command mode

CONFIG

## Example

```
ma4000(config)# aaa accounting commands tacacs+
```

**aaa accounting start-stop tacacs+**

This command enables logging in/out of the system.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
[no] aaa accounting start-stop tacacs+
```

**Parameters**

The command contains no arguments.

**Default value**

accounting disabled

**Command mode**

CONFIG

**Example**

```
ma4000(config)# aaa accounting start-stop tacacs+
```

**show authentication methods**

This command allows to view the authentication methods selected in the system.

**Syntax**

```
show authentication methods
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

## Example

```
ma4000# show authentication methods
```

```
 Login Authentication Method Lists
```

```
    ~~~~~
```

| Name    | Methods |
|---------|---------|
| default | local   |

```
    ~~~~~
```

```
    ~~~~~
```

```
    Lines Authentication Method Lists
```

```
    ~~~~~
```

| Line    | Login Method List | Enable Method List |
|---------|-------------------|--------------------|
| console | default           | default            |
| telnet  | default           | default            |
| ssh     | default           | default            |

```
    ~~~~~
```

```
    ~~~~~
```

```
    ~~~~~
```

```
    console
```

```
    default
```

```
    default
```

```
    telnet
```

```
    default
```

```
    default
```

```
    ssh
```

```
    default
```

```
    default
```

## show tacacs

This command is used to view the list of TACACS+ servers.

## Syntax

```
show tacacs
```

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show tacacs
```

```
Global Values:
```

```
    Timeout 5 sec
```

```
    TACACS Configuration
```

```
    ~~~~~
```

| IP address  | Port | Timeout | Priority |
|-------------|------|---------|----------|
| 10.10.10.10 | 49   | 0       | 0        |

```
    ~~~~~
```

```
    ~~~~~
```

```
    ~~~~~
```

```
    ~~~~~
```

## show accounting

This command allows to view the accounting settings.

**Syntax**

show accounting

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show accounting
Login start-stop: Disable
Commands: Disable
```

## 16 PLC line board traffic filtering management

- [slot access-list mode](#)
- [slot access-list create](#)
- [slot access-list delete](#)
- [slot access-list bind](#)
- [slot access-list unbind](#)
- [slot access-list filter](#)
- [show slot access-list](#)

### slot <SLOT> access-list mode

This command sets the ACL operation mode for line boards.

Filtering is performed for received traffic for the specified interface.

#### Syntax

```
slot <SLOT> access-list mode <MODE>
```

#### Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<MODE> – filtering list type:

- whitelist – packets that meet the ACL rules are transmitted;
- blacklist – packets that meet the ACL rules are discarded.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# slot 13 access-list mode whitelist
```

### slot <SLOT> access-list create

This command creates a new ACL for PLC line cards.

#### Syntax

```
slot <SLOT> access-list create <NAME>
```

#### Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<NAME> – ACL name, specified as the string of up to 32 characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# slot 13 access-list create test
```

**slot <SLOT> access-list delete**

This command removes the ACL by its name.

**Syntax**

```
slot <SLOT> access-list delete <NAME>
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<NAME> – ACL name, specified as the string of up to 32 characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# slot 13 access-list delete test
```

**slot <SLOT> access-list bind**

This command assigns the ACL to certain interfaces.

**Syntax**

```
slot <SLOT> access-list bind <INTERFACE> <RANGE> <NAME>
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<INTERFACE> – interface type: plc-front-port; plc-pon-port; plc-slot-channel. Interfaces description is in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#).

<NAME> – ACL name, specified as the string of up to 32 characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# slot 13 access-list bind plc-front-port 0/0 test
```

**slot <SLOT> access-list unbind**

This command removes compliance of the ACL with specified interfaces.

**Syntax**

```
slot <SLOT> access-list unbind <INTERFACE> <RANGE> <NAME>
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<INTERFACE> – interface type: plc-front-port; plc-pon-port; plc-slot-channel. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#).

<NAME> – ACL name, specified as the string of up to 32 characters.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# slot 13 access-list unbind plc-front-port 0/0
```

**slot <SLOT> access-list filter**

This command sets (add) or removes (del) a packet filtering rule by one of the parameters for a certain ACL.

**Syntax**

```
slot <SLOT> access-list [add|del] <TYPE> <VALUE> <NAME>
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<TYPE> – filtering method:

- mac-sa – packet filtering is performed by the MAC address of the sender, MAC address is specified as XX:XX:XX:XX:XX, where each part takes the value 00-FF;
- mac-da – packet filtering is performed by the MAC address of the recipient, MAC address is specified as XX:XX:XX:XX:XX, where each part takes the value 00-FF;
- l2-protocol – packet selection is performed by ethertype, specified in 0xXXXX format;
- ip-protocol – packet filtering is performed via IPV4/IPV6 protocol at L4 level, specified in 0xXX format;
- ip-sa – packet filtering is performed by the IP address of the sender, IP address is specified as AAA.BBB.CCC.DDD, where each part takes values 0-255;
- ip-da – packet filtering is performed by the IP address of the recipient, IP address is specified as AAA.BBB.CCC.DDD, where each part takes values 0-255;
- ip-sa – packet filtering is performed by the IP address of the sender, IP address is specified as XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX, where each part takes values 0-FFFF;
- ip-da – packet filtering is performed by the IP address of the recipient, IP address is specified as XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX, where each part takes values 0-FFFF;
- tcp-sport – packet filtering is performed by the number of the TCP port of the sender, the port is specified in 0xXXXX format;
- tcp-dport – packet filtering is performed by the number of the TCP port of the recipient, the port is set in 0xXXXX format;
- udp-sport – packet filtering is performed by the number of the UDP port of the sender, the port is specified in 0xXXXX format;
- udp-dport – packet filtering is performed by the number of the UDP port of the recipient, the port is set in 0xXXXX format.

<VALUE> – filter value;

<NAME> – filter name.

## Command mode

CONFIG

## Example

```
ma4000(config)# slot 13 access-list filter add ip-sa 192.168.2.2 test
```

## show slot <SLOT> access-list

This command is used to view access control lists on a PLC8 line card.

## Syntax

```
show slot <SLOT> access-list
```

## Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

**Example**

```
ma4000# show slot 6 access-list
Global mode: blacklist
```

## 17 Interface mirroring

- [mirror interface](#)
- [mirror analyzer](#)
- [mirror add-tag](#)
- [mirror added-tag-config](#)
- [mirror vlan](#)

### mirror <rx|tx>interface

This command enables the operation of mirroring the interfaces of the central switch PP4X and PLC8 modules for incoming/outgoing traffic. Interface mirroring allows you to copy traffic going from one interface to another for external analysis.

The use of the negative form of the command (no) disables the interface mirroring operation.

### Syntax

```
[no] mirror <rx|tx> interface <INTERFACE> <RANGE>
no mirror all
```

### Parameters

<rx|tx> – traffic type:

- rx – incoming;
- tx – outgoing.

<INTERFACE> – interface type: front-port; plc-front-port; plc-pon-port; plc-slot-channel; port-channel; slot-channel . The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If the value 'all' is specified, all interfaces of the specified type will be specified in the command.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

CONFIG

### Example

```
ma4000(config)# mirror rx interface front-port all
```

For incoming traffic coming to front-port interfaces, port mirroring operation is enabled. Traffic is copied from front-port interfaces to the interface analyzer installed by the command mirror rx analyzer for further analysis without interfering with the front-port data stream.

### mirror <rx|tx> analyzer

This command allows you to install an interface to which packets will be duplicated to analyze incoming/outgoing traffic from the interfaces installed by the mirror rx port/mirror tx port command.

## Syntax

```
mirror <rx|tx> analyzer <INTERFACE><RANGE>
```

## Parameters

<rx|tx> – traffic type:

- rx – incoming;
- tx – outgoing.

<INTERFACE> – interface type: front-port; plc-front-port; plc-pon-port; plc-slot-channel; port-channel; slot-channel . The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

## Command mode

CONFIG

## Example

```
ma4000(config)# mirror rx analyzer front-port 1/2
```

Data for external analysis will be duplicated to the 2 uplink ports of the first PP4X board from the port/ports where the option 'mirror incoming traffic' is set.

## mirror add-tag

This command allows adding the 802.1q mark to the traffic being analyzed. Setting the tag value can be done by the command **mirror <rx|tx> added-tag-config**.

The use of the negative form of the command (no) removes the tag.

## Syntax

```
[no] mirror add-tag
```

## Parameters

The command contains no arguments.

## Command mode

CONFIG

## Example

```
ma4000(config)# mirror add-tag
```

**mirror <rx|tx> added-tag-config**

This command allows you to set a tag value that can be added to the analyzed incoming/outgoing traffic.

**Syntax**

```
mirror <rx|tx> added-tag-config vlan <VID> [user-prio <USER-PRIO>]
```

**Parameters**

<VID> – identification VLAN number, takes values from [1 .. 4094];

<USER-PRIO> – COS priority, takes values from [0 .. 7].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# mirror rx added-tag-config vlan 4 user-prio 5
```

**mirror <rx|tx> vlan**

The command specifies the VLANID to be used in the mirroring operation when transmitting incoming/outgoing traffic.

**Syntax**

```
[no] mirror <rx|tx> vlan <VID>
```

**Parameters**

<rx|tx> – traffic type:

- rx – incoming;
- tx – outgoing.

<VID> – VLAN ID, takes values of [1..4094].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# mirrorrxvlan 3
```

## 18 Ethernet interface management

- [interface](#)
- [shutdown](#)
- [bridging to](#)
- [flow-control](#)
- [frame-types](#)
- [ingress-filtering](#)
- [pvid](#)
- [speed](#)
- [rate-limit](#)
- [load-average](#)
- [shaper](#)
- [cntrset](#)
- [clear counters](#)
- [show interfaces configuration](#)
- [show interface counters](#)
- [show interface utilization](#)
- [show interfaces status](#)
- [show cntrset](#)
- [show slot cntrset](#)
- [show slot shaping](#)
- [show slot rate-limits](#)

### interface

The command allows you to switch to the management mode of one or group of interfaces.

#### Syntax

```
interface <INTERFACE> <RANGE>
```

#### Parameters

<INTERFACE> – interface type: front-port; gpon-port; ont; plc-front-port; plc-mgmt-pon-port; plc-pon-port; plc-slot-channel; plc-slot-port; port-channel; slot-channel; slot-port; stack-port. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

#### Command mode

CONFIG

#### Example

```
ma4000(config)# interface front-portall
ma4000(front-port-all)#
```

## shutdown

The command disables an interface being configured.

The use of a negative form (no) of the command enables the interface being configured.

### Syntax

```
[no] shutdown
```

### Parameters

The command contains no arguments.

### Command mode

FRONT-PORT

PORT-CHANNEL

SLOT-PORT

SLOT-CHANNEL

STACK-PORT

PLC-SLOT-PORT

PLC-PON-PORT

PLC-MGMT-PON-PORT

PLC-FRONT-PORT

### Example

```
ma4000(front-port-all)# shutdown
```

Configured interface is disabled.

## bridging to

This command sets permission for traffic transfer between interfaces.

The use of the negative form (no) of the command sets prohibition for traffic transfer between interfaces.

### Syntax

```
[no] bridging to <INTERFACE> <RANGE>
```

### Parameters

<INTERFACE> – interface type: front-port; port-channel, slot-channel. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

### Command mode

FRONT-PORT

PORT-CHANNEL

SLOT-CHANNEL

PLC-SLOT-CHANNEL

PLC-PON-PORT

PLC-FRONT-PORT

### Example

```
ma4000(front-port-all)# bridging to slot-channel 2
```

### flow-control

This command enables/disables the data flow control mechanism (flowcontrol).

The *flowcontrol* mechanism makes it possible to compensate for differences in transmitter and receiver speeds. If the traffic amount exceeds a certain level, the receiver will transmit frames informing the transmitter of the need to reduce the amount of traffic to reduce the number of lost packets.

To implement this mechanism, it is necessary that the remote device also supports this function.

### Syntax

```
flow-control <ACT>
```

### Parameters

<ACT> – allocated action:

- on – enable;
- off – disable.

### Default value

disabled

### Command mode

FRONT-PORT

PORT-CHANNEL

SLOT-CHANNEL

PLC-SLOT-CHANNEL

PLC-SLOT-PORT

PLC-MGMT-PON-PORT

## PLC-FRONT-PORT

### Example

```
ma4000(front-port-all)# flow-control on
```

Data flow control is enabled on the configured port.

### frame-types

The command allows to assign specific rules for receiving packets for the port:

receive tagged and untagged packages;

receive only packets with VLAN tag.

### Syntax

```
frame-types <ACT>
```

### Parameters

<ACT> – allocated action:

- all – receive tagged and untagged packages;
- tagged – receive only packets with VLAN tag.

### Default value

Receive all packets (tagged and untagged)

### Command mode

FRONT-PORT

PORT-CHANNEL

SLOT-CHANNEL

PLC-SLOT-CHANNEL

PLC-SLOT-PORT

PLC-PON-PORT

PLC-FRONT-PORT

### Example

```
ma4000(front-port-all)# frame-types all
```

Untagged traffic is allowed on configured ports.

## ingress-filtering

This command is intended to enable packet filtering based on the VLAN tag. The use of the negative form (no) of the command disables packet filtering based on the VLAN tag.

### Syntax

```
[no] ingress-filtering
```

### Parameters

The command contains no arguments.

### Default value

The feature is enabled by default.

### Command mode

FRONT-PORT

PORT-CHANNEL

SLOT-CHANNEL

PLC-SLOT-CHANNEL

PLC-SLOT-PORT

PLC-FRONT-PORT

### Example

```
ma4000(front-port-all)# ingress-filtering
```

Inbound packet filtering is enabled on custom ports.

## pvid

This command sets the default VID value for packets received by the port. When an untagged packet or a packet with VID value in the VLAN tag equal to 0 is received, the packet is assigned a VID value equal to PVID.

### Syntax

```
pvid <VID>
```

### Parameters

<VID> – VLAN port ID, may take values [1 .. 4094].

### Default value

1

**Command mode**

FRONT-PORT

PORT-CHANNEL

SLOT-CHANNEL

PLC-SLOT-CHANNEL

PLC-SLOT-PORT

PLC-PON-PORT

PLC-FRONT-PORT

**Example**

```
ma4000(front-port-2/5)# pvid 5
```

PVID 5 is assigned to the configured port.

**speed**

The command sets speed value for interface being configured. The command sets the following modes: 1000 Mbps, 10 Gbps, 10 Mbps, 100 Mbps or auto.

**Syntax**

```
speed { 10G | 1000M | 100M { full-duplex | half-duplex } | 10M { full-duplex | half-duplex } | auto }
```

**Parameters**

10M – 10 Mbps speed value with the operation mode of the transceiver:

- full-duplex – duplex,
- half-duplex.

100M – 100 Mbps speed value with the operation mode of the transceiver:

- full-duplex – duplex,
- half-duplex.

1000M – 1000 Mbps speed value:

- 10G – 10Gbps speed;
- auto – automatic mode selection.

**Default value**

auto

**Command mode**

FRONT-PORT

PORT-CHANNEL

SLOT-CHANNEL

PLC-SLOT-CHANNEL

PLC-SLOT-PORT

PLC-FRONT-PORT

### Example 1

```
ma4000(front-port-2/5)# speed 10G
```

10 Gbps speed limit is set.

### Example 2

```
ma4000 (front-port-2/5)# speed 10M full-duplex
```

10 Mbps speed limit is set, duplex.

## rate-limit

This command sets speed limits for multicast and broadcast traffic on the configured interface/interfaces of the PLC8 module switch.

The use of the negative form (no) of the command removes speed limits for multicast and broadcast traffic on the configured interface/interfaces of the PLC8 module switch.

### Syntax

```
rate-limit <TYPE> <LIMIT> <BURST>
no rate-limit <TYPE>
```

### Parameters

<TYPE> – traffic type:

- bc – broadcast traffic;
- mc – multicast traffic.

<LIMIT> – speed limit level, may take values [1..10000000] kbps;

<BURST> – maximum length of a continuous packet transmission, may take values [1..10000000] bytes. It is not recommended to change it.

### Command mode

PLC-SLOT-PORT

PLC-PON-PORT

## Example

```
ma4000(slot-0-switch-config-slot-port-0)# rate-limit bc
```

## load-average

This command sets the time of statistics collection to calculate the port load on data transfer/reception.

### Syntax

```
load-average <TIME>
```

### Parameters

<TIME> – statistics collecting time: 30, 60, 90, 120, 150, 180, 210, 240, 270, 300, 330, 360, 390, 420, 450, 480, 510, 540, 570, 600, seconds.

### Command mode

CONFIG

## Example

```
ma4000(config)# load-average 330
```

## shaper

This command configures speed limit.

The use of a negative form (no) of the command removes the speed limit.

### Syntax

```
shaper <LIMIT> <BURST>
no shaper
```

### Parameters

<LIMIT> – speed limit, may take values (193..789000) kbps;

<BURST> – buffer size, may take values (1..4000) bytes.

### Command mode

PLC-SLOT-PORT

PLC-PON-PORT

## Example

```
ma4000(config)(if-gpon-1)# profile address-table TEST
```

## cntreset

This command assigns the queue statistics collector to the queues with specified criteria.

The full description of the functionality can be clarified by contacting the technical support service.

## Syntax

```
cntreset <PORT> <UNIT> <SET> <VLAN> <QUEUE> <DROP PRECEDENCE>
```

## Parameters

<PORT> – type of the port for counting, may take values:

- all – all ports;
- cpu – CPU port;
- front-port – front-port count;
- plc-front-port – plc-front-port count;
- plc-pon-port – pon-port count;
- plc-slot-port – plc-slot-port count;
- slot-port – slot-port count;
- stack-port – stack-port count.

<UNIT> – port sequential number:

- for cpu: may take values [1 .. 2];
  - for front port: <unit/port>, where:
    - unit – PP4X module number, may take values [1 .. 2];
    - port – port number, may take values [0 .. 5].
- for slot-port: <unit/port>, where:
  - unit – PP4X module number, may take values [1 .. 2];
  - port – port number, may take values [0 .. 15].
- for plc-front-port: <unit/port>, where:
  - unit – PLC8 module number, may take values [0 .. 15];
  - port – port number, may take values [0].
- for plc-pon-port: <unit/port>, where:
  - unit – PLC8 module number, may take values [0 .. 15];
  - port – port number, may take values [0 .. 7].
- for plc-slot-port: <unit/port>, where:
  - unit – PLC8 module number, may take values [0 .. 15];
  - port – port number, may take values [0 .. 1].
- for stack-port: <unit/port>, where:
  - unit – PLC8 module number, may take values [1 .. 2];
  - port – port number, may take values [0 .. 1].

<SET> – statistics collector number, may take values [0 .. 1];

<VLAN> – VLAN ID, may take values [1 .. 4094] or all;

<QUEUE> – queue number, may take values [0 .. 7] or all;

<DROP PRECEDENCE> – drop precedence value [0 .. 1] or all.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# cntreset cpu 1 0 59 7 1
```

**clear counters**

The command performs the reset of specified interface/interface group counter.

**Syntax**

```
clear counters <INTERFACE> <RANGE>
```

**Parameters**

<INTERFACE> – interface type: front-port; plc-front-port; plc-mgmt-pon-port; plc-pon-port; plc-slot-port; slot-port; stack-port. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified. You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```
ma4000# clear counters front-port 1/1-4,2/3-4
```

Reset counters for 1,2,3,4 ports of the first PP4X module and 3,4 ports of the second PP4X module.

**show interfaces configuration**

This command allows to view the configuration of a certain interface or group of interfaces.

**Syntax**

```
show interfaces <INTERFACE> <RANGE> configuration
```

**Parameters**

<INTERFACE> – interface type: front-port; port-channel. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified. You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show interface front-port 1/4 configuration

interface front-port 1/4
 speed 1000M
 frame-types tagged
 channel-group 1
exit
```

## show interface counters

This command is used to view information on interface counters. Use the **detail** command to display detailed information on interface counters.

Counters description:

- Port – port number;
- UC sent – number of unicast packets sent;
- MC sent – number of multicast packets sent;
- BC sent – number of broadcast packets sent;
- Octets sent – number of bytes sent;
- UC rcv – number of unicast packets received;
- MC rcv – number of multicast packets received;
- BC rcv – number of broadcast packets received;
- Octets rcv – number of bytes received;;
- Bad octets rcv – number of bad bytes received;
- MAC transmit err – number of frames that were not transmitted successfully due to an internal reception error at the MAC level;
- Bad frames rcv – number of bad frames received;
- Frames 64 octets pass – number of processed frames with the size of 64 bytes;
- Frames 65-127 octets pass – number of processed frames with the size of 65-127 bytes;
- Frames 128-255 octets pass – number of processed frames with the size of 128-255 bytes;
- Frames 256-511 octets pass – number of processed frames with the size of 256-511 bytes;
- Frames 512-1023 octets pass – number of processed frames with the size of 512-1023 bytes;
- Frames 1024-max octets pass – number of processed frames with the size of more than 1024 bytes;
- Excessive collisions – number of frames that were not sent due to excessive number of collisions;
- Unrec MAC cntr rcv – number of MAC Control Frames with an unknown operation code;
- FC sent – number of transmitted Flow Control frames;
- Good fc rcv – number of received Flow Control frames;
- Drop events – packet discard event counter;
- Undersize packets – number of received packets that are smaller than the minimum allowed frame size;
- Fragments packets – number of packet fragments;
- Oversize packets – number of received packets whose size exceeds the maximum allowed frame size;
- Jabber packets – number of jabber packets;
- MAC receive err – number of frames that were not received successfully due to an internal reception error at the MAC level;
- Bad CRC – number of frames whose byte number matches the length that failed frame check sequence validation;

- Collisions – collisions counter;
- Late collisions – number of cases when collision is identified after transmitting the first 64 bytes of the packet to the communication link (slotTime);
- Bad FC recv – number of Flow Control frames received that are not in the correct format.
- *Current load* – average load of the interface for receiving/transmitting data in Kbps or frames/s;
- *5:00 average* – average interface load per 5-minute interval for receiving/transmitting data in Kbps or frames/s.

## Syntax

```
show interface <INTERFACE> <RANGE> counters
show interface <INTERFACE> <RANGE> counters detail
```

## Parameters

<INTERFACE> – interface type: front-port. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified. You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show interface front-port 1/0 counters detail
```

| Counter                      | Value    |
|------------------------------|----------|
| UC sent                      | 35805    |
| MC sent                      | 5339     |
| BC sent                      | 13       |
| Octets sent                  | 3614086  |
| UC rcv                       | 39918    |
| MC rcv                       | 94563    |
| BC rcv                       | 33307    |
| Octets rcv                   | 22392248 |
| Bad octets rcv               | 0        |
| MAC transmit err             | 0        |
| Bad frames rcv               | 0        |
| Frames 64 octets pass        | 41028    |
| Frames 65-127 octets pass    | 65715    |
| Frames 128-255 octets pass   | 100476   |
| Frames 256-511 octets pass   | 1572     |
| Frames 512-1023 octets pass  | 86       |
| Frames 1024-max octets pass  | 68       |
| Excessive collisions         | 0        |
| Unrec MAC cntr rcv           | 0        |
| FC sent                      | 0        |
| Good fc rcv                  | 0        |
| Drop events                  | 0        |
| Undersize packets            | 0        |
| Fragments packets            | 0        |
| Oversize packets             | 0        |
| Jabber packets               | 0        |
| MAC receive err              | 0        |
| Bad CRC                      | 0        |
| Collisions                   | 0        |
| Late collisions              | 0        |
| Bad FC rcv                   | 0        |
| Current load Kbits sent/sec  | 1        |
| Current load Kbits rcv/sec   | 2        |
| Current load frames sent/sec | 2        |
| Current load frames rcv/sec  | 3        |
| 5:00 average Kbits sent/sec  | 1        |
| 5:00 average Kbits rcv/sec   | 1        |
| 5:00 average frames sent/sec | 1        |
| 5:00 average frames rcv/sec  | 2        |

## show interface <INTERFACE> utilization

This command allows to view the port load status for data transmission/reception:

- Last utilization counters – average port load;
- 5m:00s utilization average – values averaged over a 5-minute interval (interval setting is performed by the load-average command).

## Syntax

```
show interface <INTERFACE> <RANGE> utilization
```

## Parameters

<INTERFACE> – interface type: front-port; slot-port. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified. You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show interface front-port all utilization
<cr>

ma4000# show interface front-port all utilization

 Last utilization counters
  ~~~~~
Port          Kbits sent/sec    Kbits rcv/sec    Frames sent/sec    Frames
rcv/sec
-----
front-port 1/0    2                4                4                6
front-port 1/1    0                0                0                0
front-port 1/2    0                0                0                0
front-port 1/3    0                0                0                0
front-port 1/4    0                0                0                0
front-port 1/5    0                0                0                0

  5m:00s utilization average
  ~~~~~
Port Kbits sent/sec Kbits rcv/sec Frames sent/sec Frames
rcv/sec

front-port 1/0 0 2 0 2
front-port 1/1 0 0 0 0
front-port 1/2 0 0 0 0
front-port 1/3 0 0 0 0
front-port 1/4 0 0 0 0
front-port 1/5 0 0 0 0
```

## show interfaces <INTERFACE> status

This command is used to view information about the interface status.

Description:

- Interface – interface name;
- Status – connection state:
  - up – connection is established;
  - down – no connection.

- Media – storage type:
  - none – none;
  - error – error;
  - copper – copper;
  - fiber – fiber;
  - unknown – unknown.
- Speed – port data rate, Mbps;
- Duplex – transceiver operation mode:
  - full – full duplex;
  - half – half-duplex.
- Flow control – «Flow control» (PFC) feature state:
  - no – active;
  - yes – not active.

## Syntax

show interfaces status <INTERFACE> <RANGE>

## Parameters

<INTERFACE> – interface type: front-port; port-channel; slot-channel; stack-port; slot-port. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified. You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

Example

```
ma4000# show interface front-port 1/0-5 status
```

| Interface      | Status | Media  | Speed   | Duplex | Flow control |
|----------------|--------|--------|---------|--------|--------------|
| -----          | -----  | -----  | -----   | -----  | -----        |
| front-port 1/0 | up     | copper | 1 Gbps  | full   | no           |
| front-port 1/1 | down   | none   | 10 Mbps | half   | no           |
| front-port 1/2 | down   | none   | 10 Mbps | half   | no           |
| front-port 1/3 | down   | none   | 10 Mbps | half   | no           |
| front-port 1/4 | down   | none   | 10 Mbps | half   | no           |
| front-port 1/5 | down   | none   | 10 Mbps | half   | no           |

## show cntreset

This command is used to view the queue collector information.

## Syntax

show cntreset <SET>

## Parameters

<SET> – counter number, may take values [0 .. 1].

**Command mode**

ROOT

**Example**

```

ma4000# show cnterset 1
[0A093204] Configuration Register: 0x00000000
[0A093214] Outgoing Unicast Packet Count: 587
[0A093224] Outgoing Multicast Packet Count: 0
[0A093234] Outgoing Broadcast Packet Count: 1
[0A093244] Bridge Egress Filtered Packet Count: 37296
[0A093254] Tail Dropped Packet Counter: 0
[0A093264] Control Packet Counter: 712
[0A093274] Egress Forwarding Restriction Dropped Packet Counter: 0

```

**show slot <SLOT> cnterset**

This command is used to view the queue collector for port information.

**Syntax**

```
show slot <SLOT> cnterset <SET>
```

**Parameters**

<SET> – counter number, may take values [0 .. 1].

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```

ma4000# show slot 1 cnterset 0

Counter Set
~~~~~
Name                                     Value
-----
Outgoing Unicast Packets                 0
Outgoing Multicast Packets               0
Outgoing Broadcast Packets               0
Bridge Egress Filtered Packets           0
Tail Dropped Packets                     0
Control Packets                           4528
Egress Forwarding Restriction Dropped Packets 0
Multicast FIFO Dropped Packets

```

**show slot <SLOT>shaping**

This command is used to view shaper settings on the ports of a certain PLC8 module (bandwidth restriction on traffic transfer from the port).

**Syntax**

```
show slot <SLOT> shaping
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```
ma4000# show slot 6 shaping
Interface name      Limits      Burst
plc-pon-port 6/0    0           0
plc-pon-port 6/1    0           0
plc-slot-port 6/0    0           0
plc-slot-port 6/1    0           0
plc-pon-port 6/2    0           0
plc-pon-port 6/3    0           0
plc-pon-port 6/4    0           0
plc-pon-port 6/5    0           0
plc-pon-port 6/6    0           0
plc-pon-port 6/7    100000     4000
```

**show slot <SLOT>rate-limits**

This command is used to view bandwidth limiting settings for received (BC, MC, Unknown UC) traffic on ports of a certain PLC8 module.

**Syntax**

```
show slot <SLOT> rate-limits
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```
ma4000# show slot 6 rate-limits
```

| Interface name    | Limits bc | Burst bc | Limits mc | Burst mc |
|-------------------|-----------|----------|-----------|----------|
| plc-pon-port 6/0  | 0         | 0        | 0         | 0        |
| plc-pon-port 6/1  | 0         | 0        | 0         | 0        |
| plc-slot-port 6/0 | 0         | 0        | 0         | 0        |
| plc-slot-port 6/1 | 0         | 0        | 0         | 0        |
| plc-pon-port 6/2  | 0         | 0        | 0         | 0        |
| plc-pon-port 6/3  | 0         | 0        | 0         | 0        |
| plc-pon-port 6/4  | 0         | 0        | 0         | 0        |
| plc-pon-port 6/5  | 0         | 0        | 0         | 0        |
| plc-pon-port 6/6  | 0         | 0        | 0         | 0        |
| plc-pon-port 6/7  | 0         | 0        | 0         | 0        |

## 19 GPON interface management

- [interface](#)
- [shutdown](#)
- [fec](#)
- [optics](#)
- [profile](#)
- [show interface gpon-port / state](#)
- [show interface gpon-port / redundancy](#)
- [show interface gpon-port / igmp groups](#)
- [show interface gpon-port / configuration](#)
- [show interface gpon-port / counters v-interface](#)
- [clear counters interface gpon-port](#)
- [show interface gpon-port / downstream-ber](#)
- [show interface gpon-port / optics state](#)
- [show interface gpon-port / optics configuration](#)

### interface

The command allows you to switch to the management mode of one or group of interfaces.

#### Syntax

```
interface gpon-port <RANGE>
```

#### Parameters

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

#### Command mode

CONFIG

#### Example

```
ma4000(config)# interface gpon-port 2/2
ma4000(config)(if-gpon-2/2)#
```

### shutdown

The command disables an interface being configured.

The use of a negative form (no) of the command enables the interface being configured.

#### Syntax

```
[no] shutdown
```

### Parameters

The command contains no arguments.

### Command mode

GPON-PORT

### Example

```
ma4000(if-gpon-1)# shutdown
```

Configured interface is disabled.

### fec

This command allows to enable downstream error correction on the GPON interface.

The use of the negative form of the command (no) disables this function.

### Syntax

[no] fec

### Parameters

The command contains no arguments.

### Command mode

GPON-PORT

### Example

```
ma4000(config)(if-gpon-1)# fec
```

### optics

This command is used to configure the optical parameters of the GPON interface.

### Syntax

optics <PARAM> <VALUE>

### Parameters

<PARAM> – optical parameter name;

<VALUE> – parameter value.

**Command mode**

GPON-PORT

**Example**

```
ma4000(config)(if-gpon-1)# optics reset data burst bcdr d1 23
```

**profile**

This command assigns a certain VLAN or ADDRESS-TABLE profile to a given interface/group of interfaces.

**Syntax**

```
profile <PROFILE> <NAME>
```

**Parameters**

<PROFILE> – profile type:

- vlan – VLAN profile;
- address-table – address-table profile.

<NAME> – profile name, case sensitive.

**Command mode**

GPON-PORT

**Example**

```
ma4000(config)(if-gpon-1)# profile address-table TEST
```

**show interface gpon-port <SLOT>/<CHANNEL> state**

This command is used to view the state of the GPON interface.

**Syntax**

```
show interface gpon-port <SLOT>/<CHANNEL> state
```

**Parameters**

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value «all» is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```

ma4000# # show interface gpon-port 13/all state
Reading:
Channels status information:
  Channel:
  State:
  ONT count:
  SFP vendor:
  SFP product number:
  SFP vendor revision:
  SFP temperature [C]:
  SFP voltage [V]:
  SFP tx bias current [mA]:
  SFP tx power [dBm]:

```

**show interface gpon-port <SLOT>/<CHANNEL> redundancy****Syntax**

```
show interface gpon-port <SLOT>/<CHANNEL> redundancy
```

**Parameters**

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```

ma4000# show interface gpon-port 13/all redundancy
Channel:
Redundancy mode:
Channel state:

```

**show interface gpon-port <SLOT>/<CHANNEL> igmp groups**

This command is used to view IGMP groups that are assigned to this PLC8 channel.

## Syntax

```
show interface gpon-port<SLOT>/<CHANNEL>igmp groups
```

## Parameters

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show interface gpon-port 13/all igmp groups
All IGMP groups (0):
<no groups>
```

## show interface gpon-port <SLOT>/<CHANNEL> configuration

This command is used to view the PLC8 channel configuration.

## Syntax

```
show interface gpon-port<SLOT>/<CHANNEL>configuration
```

## Parameters

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show interface gpon-port 6/0 configuration
```

```
-----  
Slot 6 GPON-port 0 configuration  
-----
```

|                             |                 |                     |
|-----------------------------|-----------------|---------------------|
| Status:                     | UP              |                     |
| Fec down:                   | false           |                     |
| Profile vlan:               | vlan-00         | OLT Profile Vlan 0  |
| Profile address table:      | addresstable-00 | OLT Profile Address |
| Table 0                     |                 |                     |
| Use custom optics settings: | false           |                     |

## show interface gpon-port <SLOT>/<CHANNEL> counters v-interface

This command is used to view the v-interface counters (between the switch and the PLC8 module olt chips).

### Syntax

```
show interface gpon-port <SLOT>/<CHANNEL> counters v-interface
```

### Parameters

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

## Example

```
ma4000# show interface gpon-port 13/all counters v-interface
```

```
##      Downstream counters for channels:
```

```

 1      RX Alignment errors
 2      RX Pause frames
 3      RX CRC-32 errors
 4      RX Oversize errors
 5      RX Bad FCS
 6      RX Too long frames
 7      RX Undersize errors
 8      RX Range errors
 9      RX Ok frames
10      RX total frames
11      RX 64 octets frames
12      RX 65-127 octets frames
13      RX 128-255 octets frames
14      RX 256-511 octets frames
15      RX 512-1023 octets frames
16      RX 1024-1518 octets frames
17      RX 1519-MAX octets frames
18      RX Total unicast packets
19      RX Total multicast packets
20      RX Total broadcast packets
22      RX Total octets
24      RX Ok octets
25      RX FIFO overflow errors
26      RX Bad FCS and <64 octets
27      RX Frame errors

```

```
##      Upstream counters for channels:
```

```

 1      TX frames without errors
 2      TX valid pause frames
 3      TX frames with errors
 4      TX good unicast packets
 5      TX good multicast packets
 6      TX good broadcast packets
 8      TX octets

```

## clear counters interface gpon-port

This command is used to clear the counters on a given GPON interface(s) from the ONT or OLT side.

### Syntax

```
clear counters interface gpon-port <SL0T>/<CHANNEL> <SIDE>
```

### Parameters

<SL0T> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<SIDE> – the side from which the counters will be reset: OLT or ONT.

### Command mode

ROOT

### Example

```
ma4000# clear counters interface gpon-port all/all ont-side
```

### show interface gpon-port <SLOT>/<CHANNEL> downstream-ber

This command is used to view downstream counters on the specified channel of the PLC8 module.

### Syntax

```
show interface gpon-port <SLOT>/<CHANNEL> downstream-ber
```

### Parameters

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

## Example

```
ma4000# show interface gpon-port 6/0 downstream-ber
```

```
-----  
Slot 6 GPON-port 0 BER table  
-----
```

| ## | Ch/ Id | Errors | Intervals | BER Interval | BER |
|----|--------|--------|-----------|--------------|-----|
| 1  | 0/ 0   | 0      | 0         | 0            | 0   |
| 2  | 0/ 1   | 0      | 0         | 0            | 0   |
| 3  | 0/ 3   | 0      | 0         | 0            | 0   |
| 4  | 0/ 4   | 0      | 0         | 0            | 0   |
| 5  | 0/ 5   | 0      | 0         | 0            | 0   |
| 6  | 0/ 6   | 0      | 0         | 0            | 0   |
| 7  | 0/ 7   | 0      | 0         | 0            | 0   |
| 8  | 0/ 8   | 0      | 0         | 0            | 0   |
| 9  | 0/ 9   | 0      | 0         | 0            | 0   |
| 10 | 0/ 10  | 0      | 0         | 0            | 0   |
| 11 | 0/ 11  | 0      | 0         | 0            | 0   |
| 12 | 0/ 12  | 0      | 0         | 0            | 0   |
| 13 | 0/ 13  | 0      | 0         | 0            | 0   |
| 14 | 0/ 14  | 0      | 0         | 0            | 0   |
| 15 | 0/ 15  | 0      | 0         | 0            | 0   |
| 16 | 0/ 16  | 0      | 0         | 0            | 0   |
| 17 | 0/ 17  | 0      | 0         | 0            | 0   |
| 18 | 0/ 18  | 0      | 0         | 0            | 0   |
| 19 | 0/ 19  | 0      | 0         | 0            | 0   |
| 20 | 0/ 20  | 0      | 0         | 0            | 0   |
| 21 | 0/ 21  | 0      | 0         | 0            | 0   |
| 22 | 0/ 22  | 0      | 0         | 0            | 0   |
| 23 | 0/ 23  | 0      | 0         | 0            | 0   |
| 24 | 0/ 24  | 0      | 0         | 0            | 0   |
| 25 | 0/ 25  | 0      | 0         | 0            | 0   |
| 26 | 0/ 26  | 0      | 0         | 100000       | 0   |
| 27 | 0/ 27  | 0      | 0         | 100000       | 0   |
| 28 | 0/ 29  | 0      | 0         | 100000       | 0   |
| 29 | 0/ 30  | 0      | 0         | 100000       | 0   |
| 30 | 0/ 31  | 0      | 0         | 100000       | 0   |

## show interface gpon-port <SLOT>/<CHANNEL> optics state

This command is used to view the optical parameters of the PLC8 module channel.

### Syntax

```
show interface gpon-port <SLOT>/<CHANNEL> optics state
```

### Parameters

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

### Example

```
ma4000# show interface gpon-port 13/all optics state
Reading:
Channels status information:
  Channel:
  State:
  ONT count:
  SFP vendor:
  SFP product number:
  SFP vendor revision:
  SFP temperature [C]:
  SFP voltage [V]:
  SFP tx bias current [mA]:
  SFP tx power [dBm]:
```

### show interface gpon-port <SLOT>/<CHANNEL> optics configuration

This command is used to view the configuration of optical parameters of the PLC8 module channel.

### Syntax

```
show interface gpon-port <SLOT>/<CHANNEL> optics configuration
```

### Parameters

<SLOT> – PLC8 module number, may take values [0..15]. If the value 'all' is specified, all modules will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<CHANNEL> – PLC8 module channel number, may take values [0..7]. If the value 'all' is specified, all channels of this module will be specified in the command. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

## Example

```
ma4000# show interface gpon-port 6/0 optics configuration
```

```
-----  
Slot 6 GPON-port 0 optics configuration  
-----
```

```
Laser reset enable:           enabled
Laser reset polarity:        low
Laser sd polarity:           low
Sd source:                   laser sd
Sd hold normal:              disabled
Sd hold snr ranging:         disabled
Reset type snr ranging:      delay based
Reset type normal:           start burst based
Reset data burst bcdr d1:    11
Reset data burst bcdr d2:    1
Reset data burst laser d1:   5
Reset data burst laser d2:   2
Reset snr burst bcdr d1:     9
Reset snr burst bcdr d2:     2
Reset snr burst laser d1:    1
Reset snr burst laser d2:    2
Reset rng burst bcdr d1:     9
Reset rng burst bcdr d2:     2
Reset rng burst laser d1:    1
Reset rng burst laser d2:    2
Reset single bcdr d1:        1
Reset single bcdr d2:        1
Reset single laser d1:       1
Reset single laser d2:       1
Reset double bcdr d4:        1
Reset double bcdr d3:        1
Reset double laser d4:       1
Reset double laser d3:       1
Delay snr timer:             8
Delay snr preamble:          32
Delay snr delimiter:         128
Delay snr burst:             128
Delay rng timer:             8
Delay rng preamble:          32
Delay rng delimiter:         128
Delay rng burst:             0
Delay single burst:          1
Delay double burst:          1
Preamble correlation preamble length: 8
Preamble length snr rng:     119
Preamble guard time data mode: 32
Preamble type1 size data:    0
Preamble type2 size data:    0
Preamble type3 size data:    5
Preamble type3 pattern:     170
Preamble delimiter size:     20
Preamble delimiter byte1:    171
Preamble delimiter byte2:    89
Preamble delimiter byte3:    131
```

Voltage **if** mode:

lvpecl

Rssi trigger pulse width:

3000

## 20 ONT configuration management

- interface ont
- ber interval
- ber update-period
- broadcast-downstream
- custom-model
- description
- fec
- omci-error-tolerant
- password
- pppoe-sessions-unlimit
- port shutdown
- port poe
- profile
- rf-port-state
- serial
- service
- service custom
- service selective-tunnel
- shutdown
- template
- clear counters interface ont
- show interface ont connections
- show interface ont laser
- show interface ont ports
- show interface ont rssi
- show interface ont state
- show interface ont configuration
- show interface ont counters
- show interface ont online
- show interface ont offline
- show interface ont unactivated
- show interface ont connected
- show interface ont configured
- show interface ont services utilization
- show interface ont downstream-ber

### interface ont

This command allows to switch to ONT configuration management mode.

#### Syntax

```
interface ont <RANGE>
```

#### Parameters

<RANGE> – ONT number, specified in format of S/P/I, where

- S – PLC8 module number, may take values [0..15];
- P – GPON port number, may take values [0..7];
- I – ONT number, may take values [0..63].

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).  
E.g.: 1/1-3/3,5,7,15-33

### Command mode

CONFIG

### Example

```
ma4000(config)# interface ont 1/0/1
ma4000(config)(if-ont-1/0/1)#
```

### ber interval

This command allows to set the interval for calculating the error rate in the downstream direction. At the end of the interval, ONT sends a REI message.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ber interval <INTERVAL>
no ber interval
```

### Parameters

<INTERVAL> – number of packets, may take values [0..4294967294]. If 'none' is specified, ONT will not send the REI message.

### Command mode

ONT

### Example

```
ma4000(config)(if-ont-1/0/1)# ber interval 500000
```

### ber update-period

This command allows to set the error rate update interval for the downstream direction.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ber update-period <PERIOD>
no ber update-period
```

**Parameters**

<PERIOD> – time period, may take values [0..4294967295], in seconds.

**Command mode**

ONT

**Example**

```
ma4000(config)(if-ont-1/0/1)# ber update-period 500000
```

**broadcast-downstream**

This command enables sending broadcast packets in a downstream direction through the dedicated GEM port.

The use of a negative form (no) of the command sets the default value.

**Syntax**

[no] broadcast-downstream

**Parameters**

The command contains no arguments.

**Command mode**

ONT

**Example**

```
ma4000(config)(if-ont-1/0/1)# broadcast-downstream
```

**custom-model**

This command sets the script profile to configure the configured ONT.

The use of a negative form (no) of the command sets the default value.

**Syntax**

[no] custom-model <NAME>

**Parameters**

<NAME> – profile to configure the configured ONT:

- none – so not set profile;

- Sets the script profile to configure the configured ONT.

### Command mode

ONT

### Example

```
ma4000(config)(if-ont-1/0/1)# custom-model none
```

### description

The command allows you to add a description for the configured ONT.

The use of a negative form (no) of the command removes description.

### Syntax

[no] description <STRING>

### Parameters

<STRING> – text description.

### Command mode

ONT

### Example

```
ma4000(config)(if-ont-1/0/1)# description name
```

### fec

This command enables upstream error correction mode when transmitting data from ONT.

The use of a negative form (no) of the command sets the default value.

### Syntax

[no] fec

### Parameters

The command contains no arguments.

### Command mode

ONT

## Example

```
ma4000(config)(if-ont-1/0/1)#fec
```

## omci-error-tolerant

This command disables the processing of errors that occur during the ONT configuration by OMCI. The use of a negative form (no) of the command sets the default value.

### Syntax

```
[no] omci-error-tolerant
```

### Parameters

The command contains no arguments.

### Command mode

ONT

## Example

```
ma4000(config)(if-ont-1/0/1)# omci-error-tolerant
```

## password

This command sets the OMCI configuration of the ONT password. The use of a negative form (no) of the command sets the default value.

### Syntax

```
password <STRING>  
no password
```

### Parameters

<STRING> – authorization OMCI password, 10 characters.

### Command mode

ONT

## Example

```
ma4000(config)(if-ont-1/0/1)#password
```

## **pppoe-sessions-unlimit**

The command disables PPPoE session limit.

The use of a negative form (no) of the command sets the default value.

### **Syntax**

```
[no] pppoe-sessions-unlimit
```

### **Parameters**

The command contains no arguments.

### **Command mode**

ONT

### **Example**

```
ma4000(config)(if-ont-10/4/0)# pppoe-sessions-unlimit
```

## **port <PORT\_ID> shutdown**

The command is used for LAN port management.

The use of a negative form (no) of the command sets the default value.

### **Syntax**

```
[no] port <PORT_ID> shutdown
```

### **Parameters**

<PORT\_ID> – number of LAN port, takes the value [0-3].

### **Command mode**

ONT

### **Example**

```
ma4000(config)(if-ont-10/4/0)# port 0 shutdown
```

## **port <PORT\_ID> poe**

The command is used for managing PoE on LAN ports of ONT.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
port <PORT_ID> poe [enable] [pse-class-control <CLASS>] [power-priority <PRIORITY>]
no port <PORT_ID> poe [enable] [pse-class-control] [power-priority]
```

**Parameters**

<PORT\_ID> – number of LAN port, takes the value [0-3];

<CLASS> – power class, takes the value [0-5];

<PRIORITY> – port priority:

- critical;
- high;
- low.

**Command mode**

ONT

**Example**

```
ma4000(config)(if-ont-10/4/0)# port 0 poe power-priority critical
```

**profile**

This command assigns a configuration profile for this ONT.

The use of the negative form (no) of the command removes assignment of profile to ONT.

**Syntax**

```
profile <TYPE> <NAME>
no profile <TYPE>
```

**Parameters**

<TYPE> – configuration profile type:

- management – set management profile;
- ports – set ports profile;
- shaping – set shaping profile;
- scripting – set scripting profile;

<NAME> – configuration profile name, case sensitive.

**Command mode**

ONT

## Example

```
ma4000(config)(if-ont-1/0/1)# profile ports ports-00
```

## rf-port-state

This command sets the operation mode of the ONT RF port in the configuration. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default value.

## Syntax

```
rf-port-state <STATE>
norf-port-state
```

## Parameters

<STATE> – RF port operation mode:

- disabled – port disabled;
- enabled – port enabled;
- no-change – do not change port state.

## Command mode

ONT

## Example

```
ma4000(config)(if-ont-1/0/1)#rf-port-state enabled
```

## serial

This command sets the serial number for ONT (PONSERIAL).

The negative command form (no) removes the ONT serial number.

## Syntax

```
serial <PONSERIAL>
no serial
```

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

**Command mode**

ONT

**Example**

```
ma4000 (config)(if-ont-1/0/1)# serial ELTX00000000
```

**service**

This command assigns service configuration profiles to this ONT.

The use of the negative form (no) of the command removes service configuration profile assignment from this ONT.

**Syntax**

```
service <SERVICEID> profile <TYPE> <NAME>
no service <SERVICEID> profile <TYPE>
```

**Parameters**

<SERVICEID> – service number, may take values [0..7];

<TYPE> – profile type:

- dba – set dba profile;
- cross-connect – set cross-connect profile.

<NAME> – configuration profile name, case sensitive.

**Command mode**

ONT

**Example**

```
ma4000(config)(if-ont-1/0/1)# service 0 profile dba dba-00
```

**service custom**

This command assigns configuration profiles to replace the profile parameters on the given ONT.

The use of the negative form (no) of the command disables the use of a replacement profile on this ONT.

**Syntax**

```
service <SERVICEID> custom [cvid <VLAN> svid <VLAN> cos <COS>]
no service <SERVICEID> profile <TYPE>
```

**Parameters**

<SERVICEID> – service number, may take values [0..7];

<VLAN> – replacing VLAN value:

<1-4094> – set replacing VLAN tag;

<COS> – replacing COS value:

<0-7> – set replacing COS value.

**Command mode**

ONT

**Example**

```
ma4000(config)(if-ont-1/0/1)# service 0 custom cvid 100 svid 200
```

- ✓ For cross-connect profiles with tag-mode single-tagged type, the VLAN tag replaces the cvid parameter.

**service selective-tunnel**

This command assigns a list of uvid permissions when using cross-connect type selective-tunnel.

The use of the negative form (no) of the command removes the specified vid (or range) from the list of uvid allowed.

**Syntax**

```
service <SERVICEID> selective-tunnel uvid <VLAN>
no service <SERVICEID> profile <VLAN>
```

**Parameters**

<SERVICEID> – service number, may take values [0..7];

<VLAN> – replacing VLAN value:

<1-4094> – set replacing VLAN tag.

**Command mode**

ONT

**Example**

```
ma4000(config)(if-ont-1/0/1)# service 0 selective-tunnel uvid 135-139,141-159
```

## shutdown

This command disables an ONT being configured.

The use of a negative form (no) of the command enables the ONT being configured.

### Syntax

```
[no] shutdown
```

### Parameters

The command contains no arguments.

### Command mode

ONT

### Example

```
ma4000(config)(if-ont-1/0/1)# shutdown
```

Configured ONT is disabled.

## template

This command assigns a template for ONT.

### Syntax

```
template <NAME>
```

### Parameters

<NAME> – template name, case sensitive.

### Command mode

ONT

### Example

```
ma4000(config)(if-ont-1/0/1)# template name
```

The 'NAME' template is assigned to the configured ONT.

## clear counters interface ont

This command resets the counters of a given ONT interface, interface group on the ONT or OLT side.

## Syntax

```
clear counters interface ont <SLOT/PORT/ONTID> <SIDE>
```

## Parameters

<SLOT/PORT/ONTID> – ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63);

<SIDE> – selecting the counters to be reset:

- ont-side – OMCI message will be sent to ONT to reset the counters;
- olt-side – counters reset on the OLT side.

## Command mode

ROOT

## Example

```
ma4000# clear counters interface ont 1/1/1 ont-side
```

## show interface ont <PONSERIAL> connections

This command is used to view the ONT connection log:

- Serial number – ONT serial number;
- Equipment ID – device model;
- Version – device firmware version;
- LinkUp – time to connect the ONT to the OLT;
- LinkDown – time to disconnect the ONT from the OLT;
- Last state – the last state.

## Syntax

```
show interface ont <PONSERIAL> connections
show interface ont <SLOT/PORT/ONTID> connections
```

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

## Command mode

ROOT

## Example

```
ma4000# show interface ont 45-4C-54-58-1A-00-2E-73 connections
```

```
-----  
[ONT13/0/0] connections  
-----
```

```
-----  
Serial number:      45-4C-54-58-1A-00-2E-73  
Equipment ID       NTP-RG-1402G-W:rev.C  
Version:           3.20.1.6075  
LinkUp :           2014-11-25 08:50:32  
LinkDown :         2014-11-25 08:51:11  
Last state :       Disconnected  
-----
```

```
-----  
Serial number:      45-4C-54-58-1A-00-2E-73  
Equipment ID       NTP-RG-1402G-W:rev.C  
Version:           3.20.1.6075  
LinkUp :           2014-11-25 08:52:31  
LinkDown :         -----  
Last state :       Working  
-----
```

## show interface ont <PONSERIAL> laser

This command is used to show parameters of the ONT optical module: transmitter power, level at reception, temperature, current and voltage.

The implementation of the measurement functionality of these indicators should be on the ONT side.

## Syntax

```
show interface ont <PONSERIAL> laser  
show interface ont <SLOT/PORT/ONTID> laser
```

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAAAAAAXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

## Command mode

ROOT

## Example

```
ma4000# show interface ont 6/5/9 laser
```

```
-----  
[ONT6/5/9] laser state  
-----
```

```
Voltage:           3.32 [V]  
Bias current:      9.40 [mA]  
Temperature:       43.38 [C]  
Tx power:          2.52 [dBm]*  
Rx power:          -16.29 [dBm]*
```

```
RF port status:    n/a  
Video power:       n/a
```

## show interface ont <PONSERIAL> ports

This command is used to view the state of ONT LAN ports: speed, operation mode.

## Syntax

```
show interface ont <PONSERIAL> ports  
show interface ont <SLOT/PORT/ONTID> ports
```

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

**Command mode**

ROOT

**Example**

```
ma4000# show interface ont 13/0/0 ports
```

```
-----
[ONT13/0/0] ports state
-----
```

| UNI ##  | 0    | 1    | 2    | 3    |
|---------|------|------|------|------|
| Link:   | down | down | down | down |
| Speed:  | n/a  | n/a  | n/a  | n/a  |
| Duplex: | n/a  | n/a  | n/a  | n/a  |

**show interface ont <PONSERIAL> rssi**

This command to view the level of received signal on OLT from ONT (RSSI).

**Syntax**

```
show interface ont <PONSERIAL> rssi
show interface ont <SLOT/PORT/ONTID> rssi
```

**Parameters**

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

**Command mode**

ROOT

## Example

```
ma4000# show interface ont 13/0/0 rssi

-----
[ONT13/0/0] RSSI
-----

Received signal strength indication:   -15.361070 [dBm]
```

## show interface ont <PONSERIAL> state

This command is used to view the state of the ONT:

- Serial number – ONT serial number;
- Slot – number of PLC8 module, to which ONT is connected;
- Channel – channel number;
- ONT ID – ONT number;
- Equipment ID – board type;
- Version – firmware version;
- Equalization delay – delay coorection;
- FEC state – error correction function status;
- OMCI port – GEM port number for OMCI transmission;
- Alloc IDs – allocation numbers (T-CONT) allocated for transmission of user traffic (first digit) and OMCI (second digit);
- Hardware state – devoice state;
- State – ONT operation state;
- ONT distance – distance from MA4000-PX to ONT;
- RSSI – level of received signal from ONT to OLT.

## Syntax

```
show interface ont <PONSERIAL> state
show interface ont <SLOT/PORT/ONTID> state
```

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

## Command mode

ROOT

## Example

```
ma4000# show interface ont 13/0/0 state

-----
[ONT13/0/0] state
-----

Serial number:      45-4C-54-58-1A-00-2E-73
Slot:               13
Channel:            0
ONT ID:             0
Equipment ID        NTP-RG-1402G-W:rev.C
Version:            3.20.1.6075
Equalization delay: 1409
FEC state:          Disable
OMCI port:          1
Alloc IDs:           320 0
Hardware state:      ACTIVATE
State:              OK
ONT distance:        0.317 [km]
RSSI:               -15.32 [dBm]
```

## show interface ont <PONSERIAL> configuration

This command is used to view the ONT configuration:

## Syntax

show interface ont <PONSERIAL> configuration [verbose]

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID } (0..63).

Verbose – when entering the command, the detailed configuration of the ONT will be shown.

**Command mode**

ROOT

**Example**

```
ma4000# show interface ont 13/0/0 configuration
```

```
-----  
[ONT13/0/0] configuration  
-----
```

```

Description:      ''
Status:           UP
Serial:           45-4C-54-58-1A-00-2E-73
Password:         '0000000000'
Fec up:           false
Downstream broadcast: true
Ber interval:     100000
Ber update period: 60
Rf port state:    disabled
Omc error tolerant: false
Service [0]:
    Profile cross connect: unassigned
    Profile dba:           unassigned
Service [1]:
    Profile cross connect: unassigned
    Profile dba:           unassigned
Service [2]:
    Profile cross connect: unassigned
    Profile dba:           unassigned
Service [3]:
    Profile cross connect: unassigned
    Profile dba:           unassigned
Service [4]:
    Profile cross connect: acs                ONT Profile Cross Connect
4
    Profile dba:           dba-00            ONT Profile DBA 0
Service [5]:
    Profile cross connect: unassigned
    Profile dba:           unassigned
Service [6]:
    Profile cross connect: unassigned
    Profile dba:           unassigned
Service [7]:
    Profile cross connect: unassigned
    Profile dba:           unassigned
Profile shaping:   shaping-00                ONT Profile Shaping 0
Profile ports:     ports-00                  ONT Profile Ports 0
Profile management: management-00           ONT Profile Management
0
Profile scripting: unassigned
Custom model:      none
Template:          unassigned

```

**show interface ont <PONSERIAL> counters**

This command is used to view the ONT counters:

**Syntax**

```
show interface ont <PONSERIAL> counters <NAME>
```

**Parameters**

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

<NAME> – counter group name:

- cross-connect – traffic statistics for services;
- gem-port-performance-monitoring – traffic statistics for the GEM port;
- gem-port-nctp-performance-monitoring – traffic statistics for the GEM port;
- ethernet-performance-monitoring-history-data – traffic statistics for the LAN port;
- ethernet-performance-monitoring-history-data2 – traffic statistics for the LAN port;
- ethernet-performance-monitoring-history-data3 – traffic statistics for the LAN port;
- gal-ethernet-performance-monitoring-history-data – traffic statistics for the LAN port;
- fec-performance-monitoring-history-data – error correction operation statistics;
- ethernet-frame-extended-performance-monitoring – traffic statistics for the LAN port;
- multicast-subscriber-monitor – MC traffic statistics and list of requested MC groups.

**Command mode**

ROOT

## Example

```
ma4000# show interface ont 13/0/0 counters cross-connect
```

```
-----  
[ONT13/0/0] counters  
-----
```

```
## Downstream counters for cross-connects:  0    1    2    3          4    5    6  
7  
1    Total octets          ---  ---  ---  ---          0    ---  ---  
---  
3    Data                  ---  ---  ---  ---          0    ---  ---  
---  
4    GEM Fragments        ---  ---  ---  ---          0    ---  ---  
---  
  
## Upstream counters for cross-connects:  0    1    2    3          4    5    6  
7  
1    US octets             ---  ---  ---  ---        101352  ---  ---  
---  
2    US packets           ---  ---  ---  ---          309    ---  ---  
---  
3    US Packed dropped (CPU) ---  ---  ---  ---          0    ---  ---  
---  
4    GEM fragments        ---  ---  ---  ---          1712   ---  ---  
---  
6    Bytes                ---  ---  ---  ---        101352  ---  ---  
---''
```

## show interface ont <PONSERIAL> online

This command is used to view the list of connected ONT.

### Syntax

```
show interface ont <PONSERIAL> online [<VALUE>]
```

### Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

<VALUE> – optional parameter, which will display a list of ONTs that meet a certain criterion:

- ok – normal operation state;
- authinprogress – in authorization process;
- authfailed – authorization failed;
- cfginprogress – in configuration application process;
- cfgfailed – configuration application error;
- blocked – ONT blocked;
- mibreset – in the process of resetting the OMCI configuration;
- fwupdating – firmware update;
- disable – disabled.

## Command mode

ROOT

## Example

```
ma4000# show interface ont 13/0/0 online ok
-----
Slot 13 GPON-port 0 ONT online list
-----
##          Serial          ONT ID Channel Status RSSI[dBm]   Version   EquipmentID
Description
1  45-4C-54-58-1A-00-2E-73 0    0      OK    -14.96  3.20.1.6075 NTP-RG-1402G-W:rev.C

Slot 13 total ONT count: 1
```

## show interface ont <PONSERIAL> offline

This command is used to view a list of configured but not connected and registered ONTs in the configuration.

## Syntax

show interface ont <PONSERIAL> offline

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

## Command mode

ROOT

## Example

```
ma4000# ma4000# show interface ont 1/4 offline
```

```
-----  
Slot 1 GPON-port 4 ONT offline list  
-----
```

| ## | Serial       | ONT ID | GPON-port | Description     |
|----|--------------|--------|-----------|-----------------|
| 1  | ELTX660000CC | 0      | 4         | max_ont11112400 |
| 2  | ELTX5D000070 | 1      | 4         | max_ont11112401 |
| 3  | ELTX5D000040 | 2      | 4         | max_ont11112402 |
| 4  | ELTX5D000078 | 3      | 4         | max_ont11112403 |
| 5  | ELTX5D000084 | 5      | 4         | max_ont11112405 |
| 6  | ELTX5D000074 | 6      | 4         | max_ont11112406 |

Slot 1 total ONT count: 6

## show interface ont <PONSERIAL> unactivated

This command is used to view a list of connected but not configured ONTs.

### Syntax

```
show interface ont <PONSERIAL> unactivated
```

### Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

### Command mode

ROOT

## Example

```
ma4000# show interface ont 13/0/0 unactivated
```

```
Slot 13 GPON-port 0 has no unactivated ONTs
```

```
Slot 13 total ONT count: 0
```

**show interface ont <PONSERIAL> connected**

This command is used to view the list of connected ONT.

**Syntax**

```
show interface ont <PONSERIAL> connected
```

**Parameters**

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT[/PORT[/ONTID]], where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

**Command mode**

ROOT

**Example**

```
ma4000# show interface ont 13/0/0 online ok
-----
Slot 13 GPON-port 0 ONT online list
-----
##      Serial          ONT ID Channel Status RSSI[dBm]   Version   EquipmentID
Description
1  45-4C-54-58-1A-00-2E-73 0    0    OK    -14.96  3.20.1.6075 NTP-RG-1402G-W:rev.C

Slot 13 total ONT count: 1
```

**show interface ont <PONSERIAL> configured**

This command is used to view the list of ONT configurations (prescribed in the configuration).

**Syntax**

```
show interface ont <PONSERIAL> configured
```

**Parameters**

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];

- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

## Command mode

ROOT

## Example

```
ma4000# show interface ont 13/0/0 configured
```

```
-----  
Slot 13 GPON-port 0 ONT configured list  
-----
```

| ## | Serial                  | ONT ID | Assigned channel | Description |
|----|-------------------------|--------|------------------|-------------|
| 1  | 45-4C-54-58-1A-00-2E-73 | 0      | 0                |             |

## show interface ont <PONSERIAL> services utilization

This command allows to view the ont load status for each data transfer/receipt service:

Last utilization counters – average load;

5m:00s utilization average – values averaged over a 5-minute interval (interval setting is performed by the load-average command).

## Syntax

```
show interface ont <PONSERIAL> services utilization
```

## Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

Command mode

ROOT

Example

```
ma4000# show interface ont ELTX1A002E79 services utilization

-----
[ONT0/0/1] services utilization
-----

Last utilization for 30 seconds:

      Services    0    1    2    3    4    5    6    7
Upstream, kbits/sec 0    0    ---    ---    ---    ---    ---    ---
Downstream, kbits/sec 0    0    ---    ---    ---    ---    ---    ---

Average utilization for 5 minutes:

      Services    0    1    2    3    4    5    6    7
Upstream, kbits/sec 0    0    ---    ---    ---    ---    ---    ---
Downstream, kbits/sec 0    0    ---    ---    ---    ---    ---    ---
```

show interface ont <PONSERIAL> downstream-ber

This command is used to view downstream ONT error rate counters.

Syntax

show interface ont <PONSERIAL> downstream-ber

Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F];

or ONT interface number, specified in format: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID} (0..63).

Command mode

ROOT

## Example

```
ma4000# show interface ont 13/0/0 downstream-ber
```

```
-----  
Slot 13 GPON-port 0 BER table  
-----
```

| ## | Ch/ | Id | Errors | Intervals | BER Interval | BER |
|----|-----|----|--------|-----------|--------------|-----|
| 1  | 0/  | 0  | 0      | 1         | 100000       | 0   |

## 21 Configuring VLAN

- [terminal-vlan](#)
- [show terminal-vlan](#)
- [vlan](#)
- [name](#)
- [tagged](#)
- [untagged](#)
- [forbidden](#)
- [show interface vlans](#)
- [show vlan](#)
- [gpon network cvlan-ethertype](#)
- [gpon network svlan-ethertype](#)
- [show gpon network](#)

### terminal-vlan

This command allows adding a new terminal-vlan to the system.

The use of the negative form of the command (no) removes the specified VLAN.

#### Syntax

```
terminal-vlan <NAME>  
no terminal-vlan <NAME>
```

#### Parameters

<NAME> – terminal-vlan name.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# terminal-vlan INTERNET
```

### show terminal-vlan

This command allows to view the list of terminal VLAN of the system.

#### Syntax

```
show terminal-vlan
```

#### Parameters

Command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000(config)# show terminal-vlan
##          Name
  1         t-111
  2         t-222
  3        INTERNET
```

**vlan**

This command allows you to add a new VLAN and go to its configuration mode.

The use of the negative form of the command (no) removes the specified VLAN.

**Syntax**

```
[no] vlan <VID>
```

**Parameters**

<VID> – VLAN ID, may take values [1 .. 4094]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

CONFIG

**Example**

Adding VLANs with numbers 1, 6-10.

```
ma4000(config)# vlan 1,6-10
ma4000(vlan-1,6-10)#
```

Deletion of VLAN with number 6.

```
ma4000(config)# no vlan 6
```

**name**

This command assigns a name to the VLAN being configured.

This command is not available when configuring several VLANs at once.

Default name is VLAN <VLAN ID>.

**Syntax**

name <NAME>

**Parameters**

<NAME> – VLAN name , string up to 32 characters.

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# name test
```

VLAN 2 is named 'test'.

**tagged**

This command allows you to add a tagging port to the VLAN group.

All packets sent via this port will be transmitted with a tag.

**Syntax**

tagged <INTERFACE> <RANGE>

**Parameters**

<INTERFACE> – interface type: front-port; plc-front-port; plc-pon-port; plc-slot-channel; port-channel; slot-channel. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If the value 'all' is specified, all interfaces of the specified type will be specified in the command.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

VLAN

**Example**

```
ma4000(vlan-100)# tagged front-port 1/0
```

Uplink port with serial number 0 of PP4X 1 module is added to the current VLAN as a tagging port.

## untagged

This command allows you to add a port to the VLAN group through which packets will be transmitted without adding a tag.

### Syntax

untagged <INTERFACE> <RANGE>

### Parameters

<INTERFACE> – interface type: front-port; plc-front-port; plc-pon-port; plc-slot-channel; port-channel; slot-channel. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If the value 'all' is specified, all interfaces of the specified type will be specified in the command.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

VLAN

### Example

```
ma4000(vlan-100)# untagged front-port 1/1
```

Uplink port with serial number 1 of PP4X first module is added to the current VLAN as a untagging port. Packets will be transmitted through this port without changes.

## forbidden

This command allows you to remove a port from the VLAN group.

### Syntax

forbidden <INTERFACE> <RANGE>

### Parameters

<INTERFACE> – interface type: front-port; plc-front-port; plc-pon-port; plc-slot-channel; port-channel; slot-channel. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If the value 'all' is specified, all interfaces of the specified type will be specified in the command.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

VLAN

## Example

```
ma4000(vlan-100)# forbidden front-port 0
```

Uplink-port 0 is removed from the current VLAN group.

## show interface vlans

This command is used to view information about VLAN parameters on a given interface/group of interfaces.

### Syntax

```
show interfaces <INTERFACE> <RANGE> vlans
```

### Parameters

<INTERFACE> – interface type: front-port; gpon-port; ont; plc-front-port; plc-mgmt-pon-port; plc-pon-port; plc-slot-port; port-channel; slot-channel; slot-port; stack-channel; stack-port. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If the value 'all' is specified, all interfaces of the specified type will be specified in the command.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

## Example

```
ma4000# show interface front-port 1/0 vlans
```

```
VLAN Table
~~~~~
```

| Interface      | PVID | Frame types | Ingress filtering | VLAN tag | VLAN untag |
|----------------|------|-------------|-------------------|----------|------------|
| front-port 1/0 | 1    | All         | yes               | 200      | 1          |

N/A - interface doesn't exist

N/S - interface is not a 802.1Q bridge port

ERR - can't get vlan setting for interface

## show vlan

This command allows to view VLANs that are registered on the device.

### Syntax

```
show vlan <VID>
```

## Parameters

<VID> – VLAN ID, may take values [1 .. 4094]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show vlan 1-5
```

Vlans:

~~~~~

| VID  | Name     | Tagged               |     | Untagged              |     |
|------|----------|----------------------|-----|-----------------------|-----|
| ---- | -----    | -----                |     | -----                 |     |
| 1    | VLAN0001 | slot-channel 0       | (M) | front-port 1/0        | (S) |
|      |          | slot-channel 1       | (M) | front-port 1/1        | (S) |
|      |          | slot-channel 2       | (M) | front-port 1/2        | (S) |
|      |          | slot-channel 3       | (M) | front-port 1/3        | (S) |
|      |          | slot-channel 4       | (M) | front-port 1/4        | (S) |
|      |          | slot-channel 5       | (M) | front-port 1/5        | (S) |
|      |          | slot-channel 6       | (M) | front-port 2/0        | (S) |
|      |          | slot-channel 7       | (M) | front-port 2/1        | (S) |
|      |          | slot-channel 8       | (M) | front-port 2/2        | (S) |
|      |          | slot-channel 9       | (M) | front-port 2/3        | (S) |
|      |          | slot-channel 10      | (M) | front-port 2/4        | (S) |
|      |          | slot-channel 11      | (M) | front-port 2/5        | (S) |
|      |          | slot-channel 12      | (M) | -                     |     |
|      |          | slot-channel 13      | (M) | -                     |     |
|      |          | slot-channel 14      | (M) | -                     |     |
|      |          | slot-channel 15      | (M) | -                     |     |
|      |          | plc-front-port 0/0   | (D) | plc-mgmt-pon-port 0/0 | (D) |
|      |          | plc-pon-port 0/0     | (D) | plc-mgmt-pon-port 0/1 | (D) |
|      |          | plc-pon-port 0/1     | (D) | -                     |     |
|      |          | plc-pon-port 0/2     | (D) | -                     |     |
|      |          | plc-pon-port 0/3     | (D) | -                     |     |
|      |          | plc-pon-port 0/4     | (D) | -                     |     |
|      |          | plc-pon-port 0/5     | (D) | -                     |     |
|      |          | plc-pon-port 0/6     | (D) | -                     |     |
|      |          | plc-pon-port 0/7     | (D) | -                     |     |
|      |          | plc-slot-channel 0/0 | (D) | -                     |     |
| ---- | -----    | -----                |     | -----                 |     |

Membership type: (S) – **static**, (D) – dynamic, (M) – slot-channels in management vlan

## gpon network cvlan-ethertype

This command allows you to set the ETHERTYPE value for the inner VLAN for DHCP/PPPoE upstream packets generated by the corresponding L3 agent on the OLT.

The use of a negative form (no) of the command sets the ethertype 0x8100 value.

**Syntax**

```
gpon network cvlan-ethertype <ETHERTYP>
no gpon network cvlan-ethertype
```

**Parameters**

<ETHERTYP> – ETHERTYPE for QinQ packets, assigned on inner VLAN, may take values [<0x0000-0xFFFF>].

**Command mode**

CONFIG

**Example**

```
ma4000# gpon network cvlan-ethertype 0x8100
```

**gpon network svlan-ethertype**

This command allows you to set the ETHERTYPE value for the outer VLAN for DHCP/PPPoE upstream packets generated by the corresponding L3 agent on the OLT.

The use of a negative form (no) of the command sets the ethertype 0x8100 value.

**Syntax**

```
gpon network svlan-ethertype <ETHERTYP>
no gpon network svlan-ethertype
```

**Parameters**

<ETHERTYP> – ETHERTYPE for packets, assigned on outerVLAN, may take values [<0x0000-0xFFFF>].

**Command mode**

CONFIG

**Example**

```
ma4000# gpon network svlan-ethertype 0x8100
```

**show gpon network**

This command allows you to view the lifetime of the MAC table, the ETHERTYPE value for the inner/outer VLAN DHCP/PPPoE upstream packets generated by the corresponding L3 agents on the OLT.

**Syntax**

```
show gpon network
```

## Parameters

Command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show gpon network
Network:
 Mac age time: 1200
 S-VLAN ethertype: 0x8100
 C-VLAN ethertype: 0x8100
```

## 22 PP4X module stacking configuration

- [stack upgrade-enable](#)
- [stack sync-allow](#)
- [show stack](#)

### stack upgrade-enable

This command enables the automatic firmware update for the 'slave' module in the stack.

The use of the negative form (no) of the command disables the automatic firmware update for the 'slave' module in the stack.

#### Syntax

[no] stack upgrade-enable

#### Parameters

The command contains no arguments.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# stack upgrade-enable
```

### stack sync-allow

This command sets permission for synchronization of configuration files in the stack between current devices.

The use of the negative form (no) of the command prohibits synchronization of configuration files in the stack between current devices.

#### Syntax

[no] stack sync-allow

#### Parameters

The command contains no arguments.

#### Command mode

ROOT

## Example

```
ma4000# stack sync-allow
```

Set permission for synchronization of configuration files.

## show stack

This command is used to view the state of the stack, the booted firmware version, and permissions for synchronization of configuration between modules in the stack.

## Syntax

```
show stack
```

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show stack
```

Stack Units

```
~~~~~
Unit   Position   Role    Prio   MAC Address      Version
----   -
*1     Left        MASTER  240    a8:f9:4b:88:33:a0  1 3 2 333 40586
```

Synchronization state in the stack: Unknown

Stack-channel State

```
~~~~~
Interface Status

stack-port 1/0 down (on discovery)
stack-port 1/1 down (on discovery)
```

## 23 Simple network management protocol (SNMP) configuration

- [ip snmp agent community](#)
- [ip snmp agent enable](#)
- [ip snmp agent engine id](#)
- [ip snmp agent engine id generate](#)
- [ip snmp agent system name](#)
- [ip snmp agent transport](#)
- [ip snmp agent traps](#)
- [ip snmp agent user add](#)
- [ip snmp agent user delete](#)
- [ip snmp agent user private add](#)
- [show ip snmp agent users](#)

### **ip snmp agent community**

This command sets SNMP communities in the device.

The use of a negative form (no) of the command removes SNMP communities.

#### **Syntax**

```
[no] ip snmp agent community <MODE> <COMMUNITY>
```

#### **Parameters**

<MODE> – set access mode:

- readonly – read;
- readwrite – edit;
- trap – snmp trap transmission list.

<COMMUNITY> – community name, 63 characters max.

#### **Command mode**

CONFIG

#### **Example**

```
ma4000(config)# ip snmp agent community readonly test
```

The test user group has read rights.

### **ip snmp agent enable**

This command enables management and monitoring of the device via SNMP protocol.

The use of a negative form (no) of the command prohibits device management via SNMP protocol.

#### **Syntax**

```
[no] ip snmp agent enable
```

**Parameters**

The command contains no arguments.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip snmp agent enable
```

**ip snmp agent engine id**

This command defines Engine ID of SNMPv3.

The use of a negative form (no) of the command removes SNMPv3 Engine ID.

**Syntax**

```
[no] ip snmp agent engine id <ENGINEID>
```

**Parameters**

<ENGINEID > – identifier, 63 characters max.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip snmp agent engine id test
```

**ip snmp agent engine id generate**

This command sets random Engine ID of SNMPv3.

**Syntax**

```
ip snmp agent engine id generate
```

**Parameters**

The command contains no arguments.

**Command mode**

CONFIG

## Example

```
ma4000(config)# ip snmp agent engine id generate
```

## ip snmp agent system name

This command sets the system name of the device in SNMP.

The use of a negative form (no) of the command removes system name of the device in SNMP.

### Syntax

```
[no] ip snmp agent system name <NAME>
```

### Parameters

<NAME> – device name, 255 characters max.

### Command mode

CONFIG

## Example

```
ma4000(config)# ip snmp agent system name ma4000
```

## ip snmp agent transport

This command defines the transport protocol and port for SNMP exchange.

The use of a negative form (no) of the command removes transport protocol configuration.

### Syntax

```
[no] ip snmp agent transport <PROTOCOL> [<PORT>]
```

### Parameters

<PROTOCOL> – transport protocol: tcp/udp;

<PORT> - tcp/udp port, may take values [1 ..65535].

### Command mode

CONFIG

## Example

```
ma4000(config)# ip snmp agent transport udp 157
```

## ip snmp agent traps

This command specifies the server to which SNMP TRAP messages will be sent.

The use of a negative form (no) of the command removes a specified TRAP server.

### Syntax

```
[no] ip snmp agent traps <VERSION> <IP_ADDRESS>
```

### Parameters

<VERSION> – SNMP Trap version:

- Informs;
- trapsv1;

<IP\_ADDRESS> – server IP address.

### Command mode

CONFIG

## Example

```
ma4000(config)# ip snmp agent traps informs
```

## ip snmp agent user add

This command adds an SNMP user.

### Syntax

```
ip snmp agent user add <USER_NAME> <USER_PASSWD> <USER_ACCESS>
```

### Parameters

<USER\_NAME> – user name;

<USER\_PASSWD> – user password, string of 8 to 31 characters;

<USER\_ACCESS> – access level:

- ro – read only;
- rw – read and edit.

**Command mode**

CONFIG

**Example**

```
ma4000(config)#ip snmp agent user add test test ro
```

**ip snmp agent user delete**

This command removes an SNMP user.

**Syntax**

```
ip snmp agent user add <USER_NAME>
```

**Parameters**

<USER\_NAME> – user name.

**Command mode**

CONFIG

**Example**

```
ma4000(config)#ip snmp agent user delete test
```

**ip snmp agent user private add**

This command adds a privileged SNMP user.

**Syntax**

```
ip snmp agent user private add <USER_NAME> <USER_PASSWD> <USER_ACCESS>
```

**Parameters**

<USER\_NAME> – user name;

<USER\_PASSWD> – user password;

<USER\_ACCESS> – access level:

- ro – read only;
- rw – read and edit.

**Command mode**

CONFIG

## Example

```
ma4000(config)# ip snmp agent user private add test tested ro
```

## show ip snmp agent users

This command is used to view information about SNMPv3 users.

## Syntax

```
show ip snmp agent users
```

## Parameters

The command contains no arguments.

## Command mode

ROOT

## Example

```
ma4000# show ip snmp agent users
clish_ip_snmp_agent_user: action list, value (null)
 SNMP users
  ~~~~~
User name          User permissions
-----
```

## 24 Spanning Tree protocol configuration

- [spanning-tree enable](#)
- [spanning-tree fdelay](#)
- [spanning-tree hello1](#)
- [spanning-tree holdcount](#)
- [spanning-tree maxage](#)
- [spanning-tree mode](#)
- [spanning-tree bpdu](#)
- [spanning-tree priority](#)
- [spanning-tree pathcost](#)
- [spanning-tree admin-edge](#)
- [spanning-tree admin-p2p](#)
- [spanning-tree auto-edge](#)
- [spanning-tree mst configuration](#)
- [spanning-tree mst instance priority](#)
- [spanning-tree mst max-hops](#)
- [show spanning-tree active](#)
- [show spanning-tree bridge](#)
- [show spanning-tree interface](#)
- [show spanning-tree bpdu](#)
- [show spanning-tree mst-configuration](#)

### spanning-tree enable

This command enables STP on the device globally in the CONFIG mode and on certain interfaces in configuration modes of interfaces (FRONT-PORT, PORT-CHANNEL). The use of the negative form of the command (no) denies the use of STP feature.

#### Syntax

```
[no] spanning-tree enable
```

#### Parameters

The command contains no arguments.

#### Command mode

CONFIGURE

FRONT-PORT

PORT-CHANNEL

#### Example

```
ma4000(config)# spanning-tree enable
```

STP feature enabled.

## spanning-tree fdelay

This command sets the delay time for data transmission. Forwarding delay is the time, during which the interface remains in 'Listening' and 'Learning' states prior to going into 'Forwarding' state.

The use of the negative form of the command (no) sets the default data transmission delay time.

### Syntax

```
spanning-tree fdelay <DELAY>  
no spanning-tree fdelay
```

### Parameters

<DELAY> – data transmission delay time, [4 .. 30] seconds.

### Default value

15 seconds

### Command mode

CONFIG

### Example

```
ma4000(config)# spanning-tree fdelay 20
```

## spanning-tree hello<sup>1</sup>

This command sets the time of sending hello-packages.

The exchange of hello-packets is carried out between the Root Bridge and dedicated bridges (Designated Bridges) and serves to exchange information on the topology of the entire switched local network.

The use of the negative form of the command (no) sets the default hello packets sending time.

### Syntax

```
spanning-tree hello <TIME>  
no spanning-tree hello
```

### Parameters

<TIME> – hello packets sending time.

### Default value

2

**Command mode**

CONFIG

**Example**

```
ma4000(config)# spanning-tree hello 2
```

<sup>1</sup> In the current firmware version this parameter can not be changed. Default value is 2.

**spanning-tree holdcount**

This command sets the maximum number of BPDU packets the device can receive within a second.

The use of the negative form of the command (no) sets the default maximum number of BPDU packets the device can receive within a second.

**Syntax**

```
spanning-tree holdcount <COUNT>  
no spanning-tree holdcount
```

**Parameters**

<COUNT> – number of BPDU packets, may take values [1 .. 10].

**Default value**

6

**Command mode**

CONFIG

**Example**

```
ma4000(config)# spanning-tree holdcount 5
```

**spanning-tree maxage**

This command sets the value of BPDU packet waiting timer.

The use of the negative form of the command (no) sets the default value of BPDU packet waiting timer.

**Syntax**

```
spanning-tree maxage <TIME>  
no spanning-tree maxage
```

**Parameters**

<TIME> – BPDU packet waiting time, may take values [6 .. 40].

**Default value**

20

**Command mode**

CONFIG

**Example**

```
ma4000(config)# no spanning-tree maxage 15
```

**spanning-tree mode**

This command sets the type of the spanningtree protocol: STP, RSTP, MST.  
The use of a negative form (no) of the command sets RSTP (default value).

**Syntax**

```
spanning-tree mode <MODE>  
no spanning-tree mode
```

**Parameters**

<MODE> – protocol type:

- stp – IEEE 802.1D Spanning Tree Protocol;
- rstp – IEEE 802.1W Rapid Spanning Tree Protocol;
- mstp – IEEE 802.1S Multiple Spanning Tree Protocol.

**Default value**

RSTP

**Command mode**

CONFIG

**Example**

```
ma4000(config)# spanning-tree mode rstp
```

## spanning-tree bpdud

The command is used to set the BPDU packet processing mode by the interface with STP disabled. The use of a negative form (no) of the command sets the default value.

### Syntax

```
spanning-tree bpdud <MODE>
```

### Parameters

<MODE> – processing method:

- filtering – BPDU packets are filtered on the interface with disabled STP;
- flooding – untagged BPDU packets are transmitted on the interface with disabled STP, tagged ones are filtered.

### Default value

flooding

### Command mode

CONFIG

### Example

```
ma4000(config)# spanning-tree bpdud flooding
```

## spanning-tree priority

This command in the mode PP4X CONFIGURE sets the bridge priority of the binding tree STP, (0-65535). The root switch is assigned a lower priority switch. 32768 is set by default. The priority value should be a multiple of 4096.

This command in the FRONT-PORT, PORT-CHANNEL configuration mode sets the interface priority in the STP, (0-240). The priority value should be a multiple of 16. 128 is set by default.

The use of a negative form (no) of the command sets the default STP operation priority.

### Syntax

```
spanning-tree priority <PRIORITY>
no spanning-tree priority
```

### Parameters

<PRIORITY > – priority;

In CONFIG mode takes the value [0...65535], which should be a multiple of 4096;

In FRONT-PORT modes, PORT-CHANNEL takes [0...240], which should be a multiple of 16.

**Default value**

in CONFIG mode – 32768;

in FRONT-PORT, PORT-CHANNEL mode – 128.

**Command mode**

CONFIG

FRONT-PORT

PORT-CHANNEL

**Example**

```
ma4000(front-port-2/5)# spanning-tree priority 128
```

Priority 128 is set.

**spanning-tree pathcost**

This command sets the value of the STP path for the configurable port.

The use of a negative form (no) of the command sets the default path cost value.

**Syntax**

spanning-tree pathcost <PATHCOST>

no spanning-treepath cost

**Parameters**

<PATHCOST> – path cost, may take values [0.. 200000000].

**Default value**

0

**Command mode**

FRONT-PORT

PORT-CHANNEL

**Example**

```
ma4000(front-port-2/5)# spanning-tree pathcost 1
```

Path cost 1 is set.

## spanning-tree admin-edge

This command sets the connection type as an edge link to the host. In this case, data transmission is enabled automatically for the port, when the link is established.

The use of a negative form (no) of the command restores the default value.

### Syntax

```
[no] spanning-tree admin-edge
```

### Parameters

The command contains no arguments.

### Default value

disabled

### Command mode

FRONT-PORT

PORT-CHANNEL

### Example

```
ma4000(front-port-2/5)# spanning-tree admin-edge
```

For the configured port, the type of edgeline connection is enabled.

## spanning-tree admin-p2p

This command defines the type of p2p connection definition.

The use of a negative form (no) of the command sets the default p2p connection definition type.

### Syntax

```
spanning-tree admin-p2p <TYPE>
no spanning-tree admin-p2p
```

### Parameters

<TYPE> – connection definition type:

- auto – the definition is based on bpdu;
- force-false – forced link as p2p;
- force-true – forced link as a non p2p.

### Default value

auto

**Command mode**

FRONT-PORT

PORT-CHANNEL

**Example**

```
ma4000(front-port-2/5)# spanning-tree admin-p2p auto
```

For the configured port, the connection type p2p is defined on the basis of bdpu.

**spanning-tree auto-edge**

This command sets the automatic bridge detection on the configured interface/interfaces.

The use of a negative form (no) of the disables automatic bridge detection on the configured interface/interfaces.

**Syntax**

```
[no] spanning-tree auto-edge
```

**Default value**

enabled

**Parameters**

The command contains no arguments.

**Command mode**

FRONT-PORT

PORT-CHANNEL

**Example**

```
ma4000(front-port-2/5)# spanning-tree auto-edge
```

The automatic bridge detection feature is enabled.

**spanning-tree mst configuration**

Switch to mst configuration mode.

**Syntax**

```
spanning-tree mst configuration
```

**Parameters**

The command contains no arguments.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# spanning-tree mst configuration
```

**spanning-tree mst instance priority**

This command allows you to set a priority for this access node over others that use a shared MSTP instance. The value should be a multiple of 4096.

The use of a negative form (no) of the command sets the default priority value.

**Syntax**

```
[no] spanning-tree mst instance priority <INSTANCE> <PRIORITY>
```

**Parameters**

<INSTANCE> – MSTP instance ID, may take values [1..15];

<PRIORITY> – priority, may take values in the range [0..61440], the value should be a multiple of 4096.

**Default value**

32768

**Command mode**

CONFIG

**Example**

```
ma4000(config)# spanning-tree mst instance
```

**spanning-tree mst max-hops**

The command sets the maximum amount of hops for BPDU packet that are required to build a tree and to keep its structure information. If the packet has already passed the maximum amount of hops, it is dropped on the next hop.

The use of a negative form (no) of the command sets the default value.

## Syntax

```
[no] spanning-tree mst max-hops <COUNT>
```

## Parameters

<COUNT> – maximum amount of hops, may take values [6..40].

## Command mode

CONFIG

## Example

```
ma4000(config)# spanning-tree mst max-hops
```

## show spanning-tree active

This command allows to view the STP status on the device stack. If you specify the **instance** command, you can view the STP status for a specific MSTP instance identifier.

## Syntax

```
show spanning-tree active [instance <INSTANCE>]
```

## Parameters

<INSTANCE> – MSTP instance ID, may take values [1..15].

## Command mode

ROOT

## Example

```
ma4000# show spanning-tree active
spanning tree: off
```

## show spanning-tree bridge

The command is used to view the bridge status. If you specify the **instance** command, you can view the bridge status for a specific MSTP instance identifier.

## Syntax

```
show spanning-tree bridge [instance <INSTANCE>]
```

**Parameters**

<INSTANCE> – MSTP instance ID, may take values [1..15].

**Command mode**

ROOT

**Example**

```
ma4000# show spanning-tree bridge
Cannot get spanning-tree status
```

**show spanning-tree interface**

This command allows to view the STP status for the specified interface range.

**Syntax**

show spanning-tree interface <INTERFACE> <RANGE>

**Parameters**

<INTERFACE> – interface type: front-port; gpon-port; ont; plc-front-port; plc-mgmt-pon-port; plc-pon-port; plc-slot-channel; plc-slot-port; port-channel; slot-channel; slot-port; stack-port. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```
ma4000# show spanning-tree interface front-port 1/0-1
spanning tree: off
```

**show spanning-tree bpdu**

The command is used to view the way BPDU packets are processed for interfaces where STP function is disabled.

**Syntax**

show spanning-tree bpdu

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show spanning-tree bpdu
Global: flooding
```

**show spanning-tree mst-configuration**

This command is used to view information about configured MSTP instances.

**Syntax**

```
show spanning-tree mst-configuration
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show spanning-tree mst-configuration
Name:
Revision: 0
```

```

MST Configuration
~~~~~
Inst State Mapped Vlans
---- -
0 enabled 1, 30, 200
```

## 25 DualHoming protocol configuration

- [backup interface](#)
- [backup-interface mac-per-second](#)
- [backup-interface mac-duplicate](#)
- [backup-interface preemption](#)
- [show backup interface](#)

### backup interface

This command specifies the redundancy interface to which the switch will occur when communication is lost on the primary. Redundancy is enabled only on those interfaces on which the SPANNING TREE protocol is disabled and VLAN Ingress Filtering is enabled. If the reserve is specified globally for the interface, it will be used for all VLANs. If another reserve is specified for some VLANs, this setting will take priority over the global setting.

The use of a negative form (no) of the command removes configuration from the interface.

### Syntax

```
[no] backup interface <INTERFACE> <RANGE> vlan <VLAN_ID_RANGE>
```

### Parameters

<INTERFACE> – interface type: front-port; port-channel . The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<VLAN\_ID\_RANGE> – can take the following values:

- [1..4094] – specified VLAN (VLAN range) identifier, for which the redundancy must be enabled.
- ignore – enable redundancy regardless of existing VLANs on the port.

### Command mode

FRONT-PORT

PORT-CHANNEL

### Example

Global redundancy

```
ma4000(port-channel-1)# no backup interface front-port 1/1 vlan ignore
ma4000(port-channel-1)# backup interface front-port 1/1 vlan ignore
```

Redundancy in a specific VLAN

```
ma4000(front-port-1/0)# no backup interface front-port 1/1 vlan 10
ma4000(front-port-1/0)# backup interface port-channel 1 vlan 10
```

## backup-interface mac-per-second

This command specifies the packet quantity per second, that will be sent into the active interface during the fallback.

The use of a negative form (no) of the command restores the default value (400 packets).

### Syntax

```
[no] backup-interfacemac-per-second <COUNT>
```

### Parameters

<COUNT> – amount of MAC addresses per second, takes value of [50..400].

### Default value

400 packets

### Command mode

CONFIG

### Example

```
ma4000(config)# backup-interface mac-per-second 200
```

## backup-interface mac-duplicate

This command defines the number of copies of packets with the same MAC address that will be sent to the active interface when switching.

The use of a negative form (no) of the command restores the default value (1 packet).

### Syntax

```
[no] backup-interfacemac-duplicate <COUNT>
```

### Parameters

<COUNT> – amount of packets copies, takes values of [1..4].

### Default value

1 packet

### Command mode

CONFIGURE

## Example

```
ma4000(config)# backup-interface mac-duplicate 4
```

## backup-interface preemption

This command specifies that it is necessary to switch traffic to the main interface when restoring communication. If the recovery of the main interface is configured while the backup is active, then when the link is raised on the main interface, the traffic will be switched to it.

The use of a negative form (no) of the command restores the default configuration.

### Syntax

```
[no] backup-interface preemption
```

### Parameters

The command contains no arguments.

### Default value

Switch disabled.

### Command mode

CONFIG

## Example

```
ma4000(config)# backup-interface preemption
```

## show backup interface

This command allows to view interface redundancy settings.

### Syntax

```
show backup interface
```

### Parameters

The command contains no arguments.

### Command mode

ROOT

Example

```
ma4000# show backup interface
Backup Interface Options:
 Preemption is disabled.
 MAC recovery packets rate 400 pps.
 Recovery packets repeats count 1.

Backup Interface Pairs
~~~~~
```

| VID | Master Interface | Backup Interface | State                    |
|-----|------------------|------------------|--------------------------|
| 30  | front-port 1/0   | front-port 2/0   | Master Up/Backup Standby |
| 150 | front-port 1/0   | front-port 2/0   | Master Up/Backup Standby |

## 26 Multicast management (IGMP, MLD)

- ip igmp snooping enable
- ip igmp snooping pp4x enable
- ip igmp snooping slot enable
- ip igmp pp4x query-interval
- ip igmp slot query-interval
- ip igmp unregistered ip4-mc
- ip igmp pp4x query-response-interval
- ip igmp slot query-response-interval
- ip igmp pp4x last-member-query-interval
- ip igmp slot last-member-query-interval
- ip igmp pp4x robustness
- ip igmp slot robustness
- ip igmp snooping mrouter add
- igmp snooping mrouter del
- ip igmp snooping mrouter learning
- ip igmp snooping querier enable
- ip igmp snooping querier fast-leave
- ip igmp version
- ip igmp proxy report enable
- ip igmp proxy report pp4x enable
- ip igmp proxy report slot enable
- ip igmp proxy report range
- ip igmp proxy report pp4x range
- ip igmp snooping static
- ip igmp proxy report slot range
- show ip igmp snooping groups vlan
- show ip igmp snooping vlan config
- show ip igmp snooping vlan hosts
- show ip igmp snooping vlan mrouter
- show ip igmp proxy report
- show slot ip igmp snooping groups
- show slot ip igmp snooping vlan config
- show slot ip igmp proxy report
- ipv6 mld snooping enable
- ipv6 igmp snooping pp4x enable
- ipv6 mld snooping slot enable
- ipv6 mld pp4x query-interval
- ipv6 mld slot query-interval
- ipv6 mld unregistered ip4-mc
- ipv6 mld pp4x query-response-interval
- ipv6 mld slot query-response-interval
- ipv6 mld pp4x last-member-query-interval
- ipv6 mld slot last-member-query-interval
- ipv6 mld pp4x robustness
- ipv6 mld slot robustness
- ipv6 mld snooping mrouter add
- ipv6 mld snooping mrouter del
- ipv6 mld snooping mrouter learning
- ipv6 mld snooping querier enable
- ipv6 mld snooping querier fast-leave
- ipv6 mld version
- ipv6 mld proxy report enable
- ipv6 mld proxy report pp4x enable

- [ipv6 mld proxy report slot enable](#)
- [ipv6 mld proxy report range](#)
- [ipv6 mld proxy report pp4x range](#)
- [ipv6 mld proxy report slot range](#)
- [show ipv6 mld snooping groups vlan](#)
- [show ipv6 mld snooping vlan config](#)
- [show ipv6 mld snooping vlan hosts](#)
- [show ipv6 mld snooping vlan mrouter](#)
- [show ipv6 mld proxy report](#)
- [show slot ipv6 mld snooping groups](#)
- [show slot ipv6 mld snooping vlan config](#)
- [show slot ipv6 mld proxy report](#)
- [show interface gpon-port igmp groups](#)

## ip igmp snooping enable

This command is used to enable IGMP snooping on all PLC modules in CONFIG mode for all VLANs and in VLAN mode for the configured VLAN.

The use of the negative form of the command (no) disables the IGMP snooping function.

IGMP Snooping is a feature that allows to determine which devices on the network participate in multicast groups and address traffic to the appropriate ports.

### Syntax

```
[no] ip igmp snooping enable
```

### Parameters

The command contains no arguments.

### Command mode

CONFIG

VLAN

### Example

```
ma4000(vlan-2)# ip igmp snooping enable
```

IGMP snooping function is enabled for VLAN 2.

## ip igmp snooping pp4x enable

This command is used to enable the IGMP snooping function in CONFIG mode for the PP4X module and in VLAN mode for the configured VLAN.

The use of the negative form of the command (no) disables the IGMP snooping function.

### Syntax

```
[no] ip igmp snooping pp4x enable
```

## Parameters

The command contains no arguments.

## Command mode

CONFIG

VLAN

## Example

```
ma4000(vlan-2)# ip igmp snooping pp4x enable
```

## ip igmp snooping slot enable

This command is used to enable *IGMP snooping* function:

- in CONFIG mode for a specific PLC8 module globally;
- in VLAN mode for configured VLAN on a specific PLC8 module.

The use of the negative form of the command (no) disables the IGMP snooping function.

## Syntax

```
[no] ip igmp snooping slot <ID> enable
```

## Parameters

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

## Command mode

CONFIG

VLAN

## Example 1

```
ma4000(config)# ip igmp snooping slot 3 enable
```

IGMP snooping function is enabled for 3rd PLC8 module.

## Example 2

```
ma4000(vlan-2)# ip igmp snooping slot 3 enable
```

For VLAN 2, IGMP snooping is enabled on 3rd PLC8 module.

## ip igmp pp4x query-interval

This command is used to set the request interval for the current VLAN for the PP4X module.

The timeout by which the system sends requests to all members of the multicast group to check their activity. If a subscriber does not send a reply to this request within a certain period of time (sets by the **command**), the system considers that the subscriber has left the multicast group and deletes it from the multicast group.

The use of a negative form (no) of the command sets the default request interval value.

### Syntax

```
ip igmp pp4x query-interval <PARAM>
no ip igmp pp4x query-interval
```

### Parameters

<PARAM> – request interval, may take values [30 .. 600], specified in seconds.

### Default value

125 seconds

### Command mode

VLAN

### Example

```
ma4000(vlan-2)# ip igmp pp4x query-interval 100
```

The request period is set to 100 sec.

## ip igmp slot query-interval

This command is used to set the request interval for the current VLAN for the certain PLC8 module.

The use of a negative form (no) of the command sets the default request interval value.

Default value is 125 seconds.

### Syntax

```
ip igmp slot <ID> query-interval <PARAM>
no ip igmp slot <ID> query-interval
```

### Parameters

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers in using comma ( , ) or specify the range using hyphen ( - );

<PARAM> – request interval, may take values [30 .. 600], specified in seconds.

**Default value**

125 seconds

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ip igmp slot 4 query-interval 100
```

The request period is set to 100 s for the 4th PLC8 module.

**ip igmp unregistered ip4-mc**

This command is designed for processing unregistered multicast traffic of IPv4. Defines whether to discard unregistered multicast traffic or distribute it to all ports.

**Syntax**

```
ip igmp unregistered ip4-mc <ACT>
```

**Parameters**

<ACT> – allocated action:

- drop – drop packets, for which destination port is not defined;
- flood – pass packets, for which destination port is not defined;

**Default value**

flood

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip igmp unregistered ip4-mc drop
```

Unregistered multicast traffic of IPv4 will be discarded.

**ip igmp pp4x query-response-interval**

This command is used to set the reply interval for the current VLAN for the PP4X module.

The reply interval is the time during which a subscriber must send a request for the system to consider it active in the multicast group.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ip igmp pp4x query-response-interval <PARAM>
no ip igmp pp4x query-response-interval
```

### Parameters

<PARAM> – reply interval, may take values [5 .. 200], specified in seconds.

### Default value

100 seconds

### Command mode

VLAN

### Example

```
ma4000(vlan-2)# ip igmp pp4x query-response-interval 125
```

Reply interval is set to 125 seconds

### ip igmp slot query-response-interval

This command is used to set the reply interval for the configured VLAN for the PLC8 module.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ip igmp slot <ID> query-response-interval <PARAM>
no ip igmp slot <ID> query-response-interval
```

### Parameters

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers in using comma ( , ) or specify the range using hyphen ( - );

<PARAM> – reply interval, may take values [5 .. 200], specified in seconds.

### Default value

100 seconds

### Command mode

VLAN

## Example

```
ma4000(vlan-2)# ip igmp slot 4 query-response-interval 125
```

The request response interval is set to 125 seconds for the 4th PLC8 module.

## ip igmp pp4x last-member-query-interval

This command is used to set the request interval for the last participant for the PP4X module.

When leaving the multicast group, the subscriber sends the corresponding packets to the IGMP proxy server. The system sends a request to the last participant in response before deleting him from the multicast group. If a subscriber does not send a reply to this request within a certain period of time, the system considers that the subscriber has left the multicast group and deletes it from the multicast group.

The use of a negative form (no) of the command sets the default value.

## Syntax

```
ip igmp pp4x last-member-query-interval <PARAM>
no ip igmp pp4x last-member-query-interval
```

## Parameters

<PARAM> – last participant request interval, my take values [1 .. 25] seconds.

## Default value

10 seconds

## Command mode

VLAN

## Example

```
ma4000(vlan-2)# ip igmp pp4x last-member-query-interval 25
```

The last participant request period is set to 25 sec.

## ip igmp slot last-member-query-interval

This command sets the request interval for the last participant for the specified PLC8 module. The use of a negative form (no) of the command sets the default value.

## Syntax

```
ip igmp slot <ID> last-member-query-interval <PARAM>
no ip igmp slot <ID> last-member-query-interval <PARAM>
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<PARAM> – last participant request interval, may take values [1 .. 25] seconds.

**Default value**

10 seconds

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ip igmp slot 4 last-member-query-interval 25
```

The last participant request period is set to 25 s for the 4th PLC8 module.

**ip igmp pp4x robustness**

This command sets the robustness value (measure that determine the system reliability) for the PP4X module. Defines the validity period for the member and the packet retransmission counter. If the subnet is unstable and subject to packet loss, the robustness value should be increased. The use of a negative form (no) of the command sets the default value.

**Syntax**

```
ip igmp pp4x robustness <PARAM>
no ip igmp pp4x robustness
```

**Parameters**

<PARAM> – robustness value, specified in the range [1 .. 10].

**Default value**

2

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ip igmp pp4x robustness 5
```

**ip igmp slot <ID> robustness**

This command sets the robustness value for PLC8 module.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
ip igmp slot <ID> robustness <PARAM>
no ip igmp slot <ID> robustness
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<PARAM> – robustness value, specified in the range [1 .. 10].

**Default value**

2

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ip igmp slot 3 robustness 5
```

For the 3rd PLC8 module the system reliability value 5 is set for the current VLAN.

**ip igmp snooping mrouter add**

This command is used to add the port that is connected to a multicast router for the selected VLAN.

**Syntax**

```
ip igmp snooping mrouter add <INTERFACE> <RANGE>
```

**Parameters**

<INTERFACE> – interface type: front-port; port-channel. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

VLAN

## Example

```
ma4000(vlan-2-5)# ip igmp snooping mrouter add front-port all
```

## igmp snooping mrouter del

This command is used to remove the port that is connected to a multicast router for the selected VLAN.

### Syntax

```
ip igmp snooping mrouter del <INTERFACE> <RANGE>
```

### Parameters

<INTERFACE> – interface type: front-port; port-channel. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

VLAN

## Example

```
ma4000(vlan-2-5)# ip igmp snooping mrouter del front-port all
```

## ip igmp snooping mrouter learning

This command enables automatic detection of port connection to the multicast router.

The use of the negative form (no) of the command disables automatic detection of port connection to the multicast router.

### Syntax

```
[no] ip igmp snooping mrouter learning
```

### Parameters

The command contains no arguments.

### Command mode

VLAN

## Example

```
ma4000(vlan-2)# ip igmp snooping mrouter learning
```

## ip igmp snooping querier enable

This command enables the querier mode.

*Querier* – a device that transmits IGMP queries.

The use of a negative form (no) of the command disables the querier mode.

## Syntax

```
[no] ip igmp snooping querier enable
```

## Parameters

The command contains no arguments.

## Command mode

VLAN

## Example

```
ma4000(vlan-2)# ip igmp snooping querier enable
```

## ip igmp snooping querier fast-leave

This command enables the fast-leave mode for the current VLAN. If the device receives the 'igmp-leave' message, the port is immediately excluded from the IGMP group.

The use of the negative form (no) of the command disables the fast-leave mode for the current VLAN.

## Syntax

```
[no] ip igmp snooping querier fast-leave
```

## Parameters

The command contains no arguments.

## Command mode

VLAN

## Example

```
ma4000(vlan-2)# ip igmp snooping querier fast-leave
```

## ip igmp version

This command sets the IGMP compatibility version for the specified VLAN.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ip igmp version <VERSION>  
no ip igmp version
```

### Parameters

<VERSION> – IGMP version, may take values: v1-only; v2-only; v3-only; v1-v2; v1-v3; v2-v3; v1-v2-v3.

### Default value

3 version

### Command mode

VLAN

## Example

```
ma4000(vlan-2)# ip igmp snooping version v3-only
```

## ip igmp proxy report enable

This command enables IGMP request proxying mode on all PLC modules.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
[no] ip igmp proxy report enable
```

### Parameters

The command contains no arguments.

### Default value

Proxy mode is disabled by default.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip igmp proxy report enable
```

**ip igmp proxy report pp4x enable**

This command enables the IGMP request proxying mode for PP4X.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
[no] ip igmp proxy report pp4x enable
```

**Parameters**

The command contains no arguments.

**Default value**

Proxy mode is disabled by default.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip igmp proxy report pp4x enable
```

**ip igmp proxy report slot enable**

This command enables the IGMP request proxying mode for specified PLC8 module.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
[no] ip igmp proxy report slot <ID> enable
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Default value**

disabled

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip igmp proxy report slot 3 enable
```

**ip igmp proxy report range**

This command sets the range of IGMP groups to be proxied on all PLC modules.

The use of a negative form (no) of the command removes the range.

**Syntax**

```
[no] ip igmp proxy report range <START RANGE> <END RANGE> from <FROM VLAN> to <TO VLAN>
```

**Parameters**

<START RANGE> – initial IP address of the range of IGMP groups to be proxied;

<END RANGE> – ending IP address of the range of IGMP groups to be proxied;

<FROM VLAN> – VLAN identification number from which IGMP requests must be proxied, may take values [1 .. 4094, all];

<TO VLAN> – VLAN identification number to which IGMP requests must be proxied, may take values [1 .. 4094].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ip igmp proxy report range 233.7.70.1 233.7.70.255 from all to 30
```

**ip igmp proxy report pp4x range**

This command specifies the range of IGMP groups to be proxied for the PP4X module.

The use of a negative form (no) of the command removes the range.

## Syntax

```
[no] ip igmp proxy report pp4x range <START RANGE> <END RANGE> from <FROM VLAN> to <TO VLAN>
```

## Parameters

<START RANGE> – initial IP address of the range of IGMP groups to be proxied;

<END RANGE> – ending IP address of the range of IGMP groups to be proxied;

<FROM VLAN> – VLAN identification number from which IGMP requests must be proxied, may take values [1 .. 4094];

<TO VLAN> – VLAN identification number to which IGMP requests must be proxied, may take values [1 .. 4094].

## Command mode

CONFIG

## Example

```
ma4000(config)# ip igmp proxy report pp4x range 233.7.70.1 233.7.70.254 from 500 to 30
```

## ip igmp snooping static

This command adds a static entry to the IGMP snooping table.

The use of a negative form (no) of the command removes the entry.

## Syntax

```
[no] ip igmp snooping static <IP> interface
```

## Parameters

<INTERFACE> – interface type: front-port, port-channel, slot-channel, plc-slot-channel, plc-pon-port, plc-front-port. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

## Command mode

CONFIG VLAN

## Example

```
ma4000(vlan-100)# ip igmp snooping static 239.255.26.5 interface plc-pon-port 1/1
```

## ip igmp proxy report slot range

This command specifies the range of IGMP groups to be proxied.

The use of a negative form (no) of the command removes the range.

### Syntax

```
[no] ip igmp proxy report slot <ID> range <START RANGE> <END RANGE> from <FROM VLAN> to <TO VLAN>
```

### Parameters

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<START RANGE> – initial IP address of the range of IGMP groups to be proxied;

<END RANGE> – ending IP address of the range of IGMP groups to be proxied;

<FROM VLAN> – VLAN identification number from which IGMP requests must be proxied, may take values [1 .. 4094];

<TO VLAN> – VLAN identification number to which IGMP requests must be proxied, may take values [1 .. 4094].

### Command mode

CONFIG

### Example

```
ma4000(config)# ip igmp proxy report slot 3 range 233.7.70.1 233.7.70.254 from 500 to 30
```

## show ip igmp snooping groups vlan

This command allows viewing information about registered IGMP-groups in the specified VLAN.

### Syntax

```
show ip igmp snooping groups vlan <VID>
```

### Parameters

<VID> – VLAN ID, may take values [1 .. 4094].

### Command mode

ROOT

## Example

```
ma4000# show ip igmp snooping groups vlan 100
```

```

  IMGP Groups
  ~~~~~
 Vlan Group Address Group Expires Filter
 ----- -
 Mode
 ----- -
 Group Member
 ----- -
 Member Expires

```

## show ip igmp snooping vlan config

This command is intended for viewing information about IGMP configuration in the specified VLAN.

IGMP snooping is disable/enable globally – IGMP snooping function disabled/enabled globally;

IGMP snooping is disable/enable for this VLAN – IGMP snooping function disabled/enabled on the current VLAN;

Querier disabled/enabled – querier mode is disabled/enabled;

IGMP version compatibility – IGMP compatibility version;

Query Interval – query interval;

Query Response Interval – reply interval (the time during which a subscriber must send a request for the system to consider it active in the multicast group);

Robustness Variable – robustness value;

Group Membership Int – amount of time that must pass before a group router decides that there are no more group members or a specific source on the network who want to participate in the mailing list;

DSCP – DSCP field value for IGMP packets generated by the device;

User-prio – p-bit value for IGMP packets generated by the device;

Fast Leave – fast-leave mode state;

Last Member Query Int – IGMP request interval for the last participant (in seconds);

Last Member Query Time – IGMP reply interval for the last participant (in seconds).

## Syntax

```
show ip igmp snooping vlan config <VID>
```

## Parameters

<VID> – VLAN ID, may take values [1 .. 4094]. You can specify several VIDs separated by commas ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show ip igmp snooping vlan config 1-6
VLAN 1
 IGMP snooping is disabled globally
 IGMP snooping is disabled for this VLAN
 IGMP version compatibility: v2-v3
 Querier disabled
 Query Interval: 125 seconds
 Query Response Interval: 10 seconds
 Robustness Variable: 2
 Group Membership Int.: 260 seconds
 DSCP: 0
 User-prio: 0
 Fast Leave: disabled
 Last Member Query Int.: 1 seconds
 Last Member Query Time: 2 seconds
```

## show ip igmp snooping vlan hosts

This command allows viewing information about IGMP hosts in the specified VLAN.

### Syntax

show ip igmp snooping vlan hosts <VID>

### Parameters

<VID> – VLAN ID, may take values [1 .. 4094].

You can specify several VIDs separated by commas ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

## Example

```
ma4000# show ip igmp snooping vlan hosts 100

Hosts ports. VLAN 100.
~~~~~
Interface                      Timer
-----
```

## show ip igmp snooping vlan mrouter

This command is used to view ports to which multicast routers are connected for the selected VLAN.

**Syntax**

```
show ip igmp snooping vlan mrouter <VID>
```

**Parameters**

<VID> – VLAN ID, may take values [1 .. 4094]. You can specify several VIDs separated by commas (,) or specify the range using hyphen (-).

**Command mode**

ROOT

**Example**

```
ma4000# show ip igmp snooping vlan mrouter 1

Multicast routers ports. VLAN 1.
~~~~~
Interface Static Timer

```

**show ip igmp proxy report**

This command allows viewing the ranges of IGMP groups that were specified for proxying in certain VLANs on the PP4x module.

**Syntax**

```
show ip igmp proxy report
```

**Parameters**

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show ip igmp proxy report
IGMP PROXY for reports: disabled

IGMP PROXY for reports ranges
~~~~~
#      First group      Last group      From VID      To VID
--

```

**show slot <SLOT> ip igmp snooping groups**

This command allows viewing information about registered IGMP-groups in the specified VLAN on the PLC line card.

**Syntax**

```
show slot <SLOT> ip igmp snooping groups <VID>
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<VID> – VLAN ID, may take values [1 .. 4094].

**Command mode**

ROOT

**Example**

```
ma4000# show slot 6 ip igmp snooping groups 30
```

**show slot <SLOT> ip igmp snooping vlan config**

This command is intended for viewing information about IGMP configuration in the specified VLAN on the PLC line card.

IGMP snooping is disable/enable globally – IGMP snooping function disabled/enabled globally;

IGMP snooping is disable/enable for this VLAN – IGMP snooping function disabled/enabled on the current VLAN;

Query Interval – query interval;

Query Response Interval – reply interval (the time during which a subscriber must send a request for the system to consider it active in the multicast group);

Robustness Variable – robustness value;

Group Membership Int – amount of time that must pass before a group router decides that there are no more group members or a specific source on the network who want to participate in the mailing list;

Fast Leave – fast-leave mode state;

Last Member Query Int – IGMP request interval for the last participant (in seconds);

Last Member Query Time – IGMP reply interval for the last participant (in seconds).

**Syntax**

```
show slot <SLOT> ip igmp snooping vlan config <VID>
```

## Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

<VID> – VLAN ID, may take values [1 .. 4094]. You can specify several VIDs separated by commas (,) or specify the range using hyphen (-).

## Command mode

ROOT

## Example

```
ma4000# show slot 6 ip igmp snooping vlan config 30
VLAN 30
  IGMP snooping is disabled globally
  IGMP snooping is disabled for this VLAN
    Query Interval:                125 seconds
    Query Response Interval:       10 seconds
    Robustness Variable:           2
    Group Membership Int.:         260 seconds
    Fast Leave:                    disabled
      Last Member Query Int.:       1 seconds
      Last Member Query Time:       2 seconds
```

## show slot <SLOT>ip igmp proxy report

This command allows viewing the ranges of IGMP groups, which were specified for proxying in certain VLANs for PLC line cards.

## Syntax

```
show slot <SLOT> ip igmp proxy report
```

## Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

## Command mode

ROOT

## Example

```
ma4000# show slot 6 ip igmp proxy report
IGMP PROXY for reports: disabled

IGMP PROXY for reports ranges
~~~~~
First group Last group From VID To VID
--- -
```

## ipv6 mld snooping enable

This command is used to enable MLD snooping on all PLC modules in CONFIG mode for all VLANs and in VLAN mode for the configured VLAN.

The use of the negative form of the command (no) disables the MLD snooping function.

MLD Snooping – function that allows you to determine which devices on the network participate in multicast groups and address traffic to the appropriate ports.

## Syntax

```
[no] ipv6 mld snooping enable
```

## Parameters

The command contains no arguments.

## Command mode

CONFIG

VLAN

## Example

```
ma4000(vlan-2)# ipv6 mld snooping enable
```

For VLAN 2, MLD snooping is enabled on all PLC modules.

## ipv6 igmp snooping pp4x enable

This command is used to enable the MLD snooping function in CONFIG mode for the PP4X module and in VLAN mode for the configured VLAN.

The use of the negative form of the command (no) disables the MLD snooping function.

## Syntax

```
[no] ipv6 mld snooping pp4x enable
```

**Parameters**

The command contains no arguments.

**Command mode**

CONFIG

VLAN

**Example**

```
ma4000(vlan-2)# ipv6 mld snooping pp4x enable
```

For VLAN 2, MLD snooping is enabled on PP4X module.

**ipv6 mld snooping slot enable**

This command is used to enable MLD *snooping* function:

- in CONFIG mode for a specific PLC8 module globally;
- in VLAN mode for configured VLAN on a specific PLC8 module.

The use of the negative form of the command (no) disables the MLD snooping function.

**Syntax**

```
[no] ipv6 mld snooping slot <ID> enable
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

CONFIG

VLAN

**Example 1**

```
ma4000(config)# ipv6 mld snooping slot 3 enable
```

MLD snooping function is enabled for 3rd PLC8 module.

**Example 2**

```
ma4000(vlan-2)# ipv6 mld snooping slot 3 enable
```

For VLAN 2, MLD snooping is enabled on 3rd PLC8 module.

### ipv6 mld pp4x query-interval

This command is used to set the request interval for the current VLAN for the PP4X module.

The timeout by which the system sends requests to all members of the multicast group to check their activity. If a subscriber does not send a reply to this request within a certain period of time (sets by the **command**), the system considers that the subscriber has left the multicast group and deletes it from the multicast group.

The use of a negative form (no) of the command sets the default request interval value.

#### Syntax

```
ipv6 mld pp4x query-interval <PARAM>
no ipv6 mld pp4x query-interval
```

#### Parameters

<PARAM> – request interval, may take values [30 .. 600], specified in seconds.

#### Default value

125 seconds

#### Command mode

VLAN

#### Example

```
ma4000(vlan-2)# ipv6 mld pp4x query-interval 100
```

The request period is set to 100 sec.

### ipv6 mld slot query-interval

This command is used to set the request interval for the current VLAN for the certain PLC8 module.

The use of a negative form (no) of the command sets the default request interval value.

Default value is 125 seconds.

#### Syntax

```
ipv6 mld slot <ID> query-interval <PARAM>
no ipv6 mld slot <ID> query-interval
```

#### Parameters

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<PARAM> – request interval, may take values [30 .. 600], specified in seconds.

**Default value**

125 seconds

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ipv6 mld slot 4 query-interval 100
```

The request period is set to 100 s for the 4th PLC8 module.

**ipv6 mld unregistered ip4-mc**

This command is designed for processing unregistered multicast traffic of IPv6. Defines whether to discard unregistered multicast traffic or distribute it to all ports.

**Syntax**

```
ipv6 mld unregistered ip6-mc <ACT>
```

**Parameters**

<ACT> – allocated action:

- drop – drop packets, for which destination port is not defined;
- flood – pass packets, for which destination port is not defined.

**Default value**

flood

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ipv6 mld unregistered ip4-mc drop
```

Unregistered multicast traffic of IPv6 will be discarded.

**ipv6 mld pp4x query-response-interval**

This command is used to set the reply interval for the current VLAN for the PP4X module.

The reply interval is the time during which a subscriber must send a request for the system to consider it active in the multicast group.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ipv6 mld pp4x query-response-interval <PARAM>
no ipv6 mld pp4x query-response-interval
```

### Parameters

<PARAM> – reply interval, may take values [5 .. 200], specified in seconds.

### Default value

100 seconds

### Command mode

VLAN

### Example

```
ma4000(vlan-2)# ipv6 mld pp4x query-response-interval 125
```

Reply interval is set to 125 seconds

## ipv6 mld slot query-response-interval

This command is used to set the reply interval for the configured VLAN for the PLC8 module.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ipv6 mld slot <ID> query-response-interval <PARAM>
no ipv6 mld slot <ID> query-response-interval
```

### Parameters

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<PARAM> – reply interval, may take values [5 .. 200], specified in seconds.

### Default value

100 seconds

### Command mode

VLAN

## Example

```
ma4000(vlan-2)# ipv6 mld slot 4 query-response-interval 125
```

The request response interval is set to 125 seconds for the 4th PLC8 module.

## ipv6 mld pp4x last-member-query-interval

This command is used to set the request interval for the last participant for the PP4X module.

When leaving the multicast group, the subscriber sends the corresponding packets to the MLD proxy server. The system sends a request to the last participant in response before deleting him from the multicast group. If a subscriber does not send a reply to this request within a certain period of time, the system considers that the subscriber has left the multicast group and deletes it from the multicast group.

The use of a negative form (no) of the command sets the default value.

## Syntax

```
ipv6 mld pp4x last-member-query-interval <PARAM>
no ipv6 mld pp4x last-member-query-interval
```

## Parameters

<PARAM> – last participant request interval, my take values [1 .. 25] seconds.

## Default value

10 seconds

## Command mode

VLAN

## Example

```
ma4000(vlan-2)# ipv6 mld pp4x last-member-query-interval 25
```

The last participant request period is set to 25 sec.

## ipv6 mld slot last-member-query-interval

This command sets the request interval for the last participant for the specified PLC8 module. The use of a negative form (no) of the command sets the default value.

## Syntax

```
ipv6 mld slot <ID> last-member-query-interval <PARAM>
no ipv6 mld slot <ID> last-member-query-interval
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<PARAM> – last participant request interval, may take values [1 .. 25] seconds.

**Default value**

10 seconds

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ipv6 mld slot 4 last-member-query-interval 25
```

The last participant request period is set to 25 s for the 4th PLC8 module.

**ipv6 mld pp4x robustness**

This command sets the robustness value (measure that determine the system reliability) for the PP4X module. Defines the validity period for the member and the packet retransmission counter. If the subnet is unstable and subject to packet loss, the robustness value should be increased. The use of a negative form (no) of the command sets the default value.

**Syntax**

```
ipv6 mld pp4x robustness <PARAM>
no ipv6 mld pp4x robustness
```

**Parameters**

<PARAM> – robustness value, specified in the range [1 .. 10].

**Default value**

2

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ipv6 mld pp4x robustness 5
```

**ipv6 mld slot <ID> robustness**

This command sets the robustness value for PLC8 module.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
ipv6 mld slot <ID> robustness <PARAM>
no ipv6 mld slot <ID> robustness
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<PARAM> – robustness value, specified in the range [1 .. 10].

**Default value**

2

**Command mode**

VLAN

**Example**

```
ma4000(vlan-2)# ipv6 mld slot 3 robustness 5
```

For the 3rd PLC8 module the system reliability value 5 is set for the current VLAN.

**ipv6 mld snooping mrouter add**

This command is used to add the port that is connected to a multicast router for the selected VLAN.

**Syntax**

```
ipv6 mld snooping mrouter add <INTERFACE> <RANGE>
```

**Parameters**

<INTERFACE> – interface type: front-port; port-channel. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

VLAN

## Example

```
ma4000(vlan-2-5)# ipv6 mld snooping mrouter add front-port all
```

## ipv6 mld snooping mrouter del

This command is used to remove the port that is connected to a multicast router for the selected VLAN.

### Syntax

```
ipv6 mld snooping mrouter del <INTERFACE> <RANGE>
```

### Parameters

<INTERFACE> – interface type: front-port; port-channel. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

### Command mode

VLAN

## Example

```
ma4000(vlan-5)# ipv6 mld snooping mrouter del front-port all
```

## ipv6 mld snooping mrouter learning

This command enables automatic detection of port connection to the multicast router.

The use of the negative form (no) of the command disables automatic detection of port connection to the multicast router.

### Syntax

```
[no] ipv6 mld snooping mrouter learning
```

### Parameters

The command contains no arguments.

### Command mode

VLAN

## Example

```
ma4000(vlan-2)# ipv6 mld snooping mrouter learning
```

## ipv6 mld snooping querier enable

This command enables the querier mode.

*Querier* – a device that transmits IGMP queries.

The use of a negative form (no) of the command disables the querier mode.

## Syntax

```
[no] ipv6 mld snooping querier enable
```

## Parameters

The command contains no arguments.

## Command mode

VLAN

## Example

```
ma4000(vlan-2)# ipv6 mld snooping querier enable
```

## ipv6 mld snooping querier fast-leave

This command enables the fast-leave mode for the current VLAN. If the device receives the 'igmp-leave' message, the port is immediately excluded from the MLD group.

The use of the negative form (no) of the command disables the fast-leave mode for the current VLAN.

## Syntax

```
[no] ipv6 mld snooping querier fast-leave
```

## Parameters

The command contains no arguments.

## Command mode

VLAN

## Example

```
ma4000(vlan-2)# ipv6 mld snooping querier fast-leave
```

## ipv6 mld version

This command sets the IGMP compatibility version for the specified VLAN.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
ipv6 mld version <VERSION>
no ipv6 mld version
```

### Parameters

<VERSION> – IGMP version, may take values: v1-only; v1-v2.

### Default value

v1-v2

### Command mode

VLAN

## Example

```
ma4000(vlan-2)# ipv6 mld snooping version v1-only
```

## ipv6 mld proxy report enable

This command enables MLD request proxying mode on all PLC modules.

The use of a negative form (no) of the command sets the default value.

### Syntax

```
[no] ipv6 mld proxy report enable
```

### Parameters

The command contains no arguments.

### Default value

Proxy mode is disabled by default.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ipv6 mld proxy report enable
```

**ipv6 mld proxy report pp4x enable**

This command enables the MLD request proxying mode for PP4X.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
[no] ipv6 mld proxy report pp4x enable
```

**Parameters**

The command contains no arguments.

**Default value**

Proxy mode is disabled by default.

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ipv6 mld proxy report pp4x enable
```

**ipv6 mld proxy report slot enable**

This command enables the MLD request proxying mode for specified PLC8 module.

The use of a negative form (no) of the command sets the default value.

**Syntax**

```
[no] ipv6 mld proxy report slot <ID> enable
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Default value**

disabled

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ipv6 mld proxy report slot 3 enable
```

**ipv6 mld proxy report range**

This command sets the range of MLD groups to be proxied on all PLC modules.

The use of a negative form (no) of the command removes the range.

**Syntax**

```
[no] ipv6 mld proxy report range <START RANGE><END RANGE> from <FROM VLAN> to <TO VLAN>
```

**Parameters**

<START RANGE> – initial IP address of the range of MLD groups to be proxied;

<END RANGE> – ending IP address of the range of MLD groups to be proxied;

<FROM VLAN> – VLAN identification number from which MLD requests must be proxied, may take values [1 .. 4094, all];

<TO VLAN> – VLAN identification number to which MLD requests must be proxied, may take values [1 .. 4094].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ipv6 mld proxy report range ff15:: ff15::ffff from all to 30
```

**ipv6 mld proxy report pp4x range**

This command specifies the range of MLD groups to be proxied for the PP4X module.

The use of a negative form (no) of the command removes the range.

**Syntax**

```
[no] ipv6 mld proxy report pp4x range <START RANGE> <END RANGE> from <FROM VLAN> to <TO VLAN>
```

**Parameters**

<START RANGE> – initial IP address of the range of MLD groups to be proxied;

<END RANGE> – ending IP address of the range of MLD groups to be proxied;

<FROM VLAN> – VLAN identification number from which MLD requests must be proxied, may take values [1 .. 4094, all];

<TO VLAN> – VLAN identification number to which MLD requests must be proxied, may take values [1 .. 4094].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ipv6 mld proxy report pp4x range ff15:: ff15::ffff from all to 30
```

**ipv6 mld proxy report slot range**

This command specifies the range of MLD groups to be proxied.

The use of a negative form (no) of the command removes the range.

**Syntax**

```
[no] ipv6 mld proxy report slot <ID> range <START RANGE> <END RANGE> from <FROM VLAN> to <TO VLAN>
```

**Parameters**

<ID> – PLC8 module number, may take values [0..15]. You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<START RANGE> – initial IP address of the range of MLD groups to be proxied;

<END RANGE> – ending IP address of the range of MLD groups to be proxied;

<FROM VLAN> – VLAN identification number from which MLD requests must be proxied, may take values [1 .. 4094, all];

<TO VLAN> – VLAN identification number to which MLD requests must be proxied, may take values [1 .. 4094].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# ipv6 mld proxy report slot 3 range ff15:: ff15::ffff from all to 30
```

## show ipv6 mld snooping groups vlan

This command allows viewing information about registered MLD groups in the specified VLAN.

### Syntax

```
show ipv6 mld snooping groups vlan <VID>
```

### Parameters

<VID> – VLAN ID, may take values [1 .. 4094].

### Command mode

ROOT

### Example

```
ma4000# show ipv6 mld snooping groups vlan 100
 MLD Groups
  ~~~~~
```

| Vlan  | Group Address | Group Expires | Filter Mode | Group Member | Member Expires |
|-------|---------------|---------------|-------------|--------------|----------------|
| ----- | -----         | -----         | -----       | -----        | -----          |

## show ipv6 mld snooping vlan config

This command is intended for viewing information about MLD configuration in the specified VLAN.

MLD snooping is disable/enable globally – MLD snooping function disabled/enabled globally;

MLD snooping is disable/enable for this VLAN – MLD snooping function disabled/enabled on the current VLAN;

Querier disabled/enabled – querier mode is disabled/enabled;

MLD version compatibility – MLD compatibility version;

Query Interval – query interval;

Query Response Interval – reply interval (the time during which a subscriber must send a request for the system to consider it active in the multicast group);

Robustness Variable – robustness value;

Group Membership Int – amount of time that must pass before a group router decides that there are no more group members or a specific source on the network who want to participate in the mailing list;

DSCP – DSCP field value for MLD packets generated by the device;

User-prio – p-bit value for IGMP packets generated by the device;

Fast Leave – fast-leave mode state;

Last Member Query Int – MLD request interval for the last participant (in seconds);

Last Member Query Time – MLD reply interval for the last participant (in seconds).

## Syntax

show ipv6 mld snooping vlan config <VID>

## Parameters

<VID> – VLAN ID, may take values [1 .. 4094]. You can specify several VIDs separated by commas ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```

ma4000# show ipv6 mld snooping vlan config 1-6
VLAN 1
  MLD snooping is disabled globally
  MLD snooping is enabled for this VLAN
  MLD version compatibility:      v1-v2
  Query Interval:                125 seconds
  Query Response Interval:       10 seconds
  Robustness Variable:           2
  Group Membership Int.:         260 seconds
  Last Member Query Int.:        1 seconds
  Last Member Query Time:        2 seconds
  Querier disabled
    Fast Leave:                  disabled
    VLAN user prio:              0
    IP DSCP:                     0

```

## show ipv6 mld snooping vlan hosts

This command allows viewing information about MLD hosts in the specified VLAN.

## Syntax

show ipv6 mld snooping vlan hosts <VID>

## Parameters

<VID> – VLAN ID, may take values [1 .. 4094].

You can specify several VIDs separated by commas ( , ) or specify the range using hyphen ( - ).

## Command mode

ROOT

## Example

```
ma4000# show ip igmp snooping vlan hosts 100

Hosts ports. VLAN 100.
~~~~~
Interface Timer

```

## show ipv6 mld snooping vlan mrouter

This command is used to view ports to which multicast routers are connected for the selected VLAN.

### Syntax

```
show ipv6 mld snooping vlan mrouter <VID>
```

### Parameters

<VID> – VLAN ID, may take values [1 .. 4094]. You can specify several VIDs separated by commas ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

## Example

```
ma4000# show ipv6 mld snooping vlan mrouter 1

Multicast routers ports. VLAN 1.
~~~~~
Interface                               Static   Timer
-----
```

## show ipv6 mld proxy report

This command allows viewing the ranges of MLD groups that were specified for proxying in certain VLANs on the PP4x module.

### Syntax

```
show ipv6 mld proxy report
```

### Parameters

The command contains no arguments.

**Command mode**

ROOT

**Example**

```
ma4000# show ipv6 mld proxy report
MLD PROXY for reports: disabled

MLD PROXY for reports ranges
~~~~~
First group Last group From VID To VID
-- - - - - - - - - - - - - - - - - - - - -
```

**show slot <SLOT> ipv6 mld snooping groups**

This command allows viewing information about registered MLD groups in the specified VLAN on the PLC line card.

**Syntax**

```
show slot <SLOT> ipv6 mld snooping groups <VID>
```

**Parameters**

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<VID> – VLAN ID, may take values [1 .. 4094].

**Command mode**

ROOT

**Example**

```
ma4000# show slot 6 ipv6 mld snooping groups 30
```

**show slot <SLOT> ipv6 mld snooping vlan config**

This command is intended for viewing information about MLD configuration in the specified VLAN on the PLC line card.

MLD snooping is disable/enable globally – MLD snooping function disabled/enabled globally;

MLD snooping is disable/enable for this VLAN – MLD snooping function disabled/enabled on the current VLAN;

Query Interval – query interval;

Query Response Interval – reply interval (the time during which a subscriber must send a request for the system to consider it active in the multicast group);

Robustness Variable – robustness value;

Group Membership Int – amount of time that must pass before a group router decides that there are no more group members or a specific source on the network who want to participate in the mailing list;

Fast Leave – fast-leave mode state;

Last Member Query Int – MLD request interval for the last participant (in seconds);

Last Member Query Time – MLD reply interval for the last participant (in seconds).

### Syntax

```
show slot < SLOT > ipv6 mld snooping vlan config < VID >
```

### Parameters

< SLOT > – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

< VID > – VLAN ID, may take values [1 .. 4094]. You can specify several VIDs separated by commas ( , ) or specify the range using hyphen ( - ).

### Command mode

ROOT

### Example

```
ma4000# show slot 6 ipv6 mld snooping vlan config 30
VLAN 30
 MLD snooping is disabled globally
 MLD snooping is disabled for this VLAN
 Query Interval: 125 seconds
 Query Response Interval: 10 seconds
 Robustness Variable: 2
 Group Membership Int.: 260 seconds
 Fast Leave: disabled
 Last Member Query Int.: 1 seconds
 Last Member Query Time: 2 seconds
```

### show slot < SLOT > ipv6 mld proxy report

This command allows viewing the ranges of MLD groups, which were specified for proxying in certain VLANs for PLC line cards.

### Syntax

```
show slot < SLOT > ipv6 mld proxy report
```

### Parameters

< SLOT > – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

**Command mode**

ROOT

**Example**

```
ma4000# show slot 6 ipv6 mld proxy report
MLD PROXY for reports: disabled
```

```
MLD PROXY for reports ranges
```

```
~~~~~
#      First group      Last group      From VID      To VID
--      -----      -
```

**show interface gpon-port <GPON-PORT> igmp groups**

The command is used for viewing the log of MC channels for the plc line card.

**Syntax**

```
show interface gpon-port <GPON-PORT> igmp groups
```

**Parameters**

<GPON-PORT> - gpon-port number in format of <SLOT>/<port>;

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - );

<port> – GPON port number of PLC8 module.

**Command mode**

ROOT

**Example**

```
ma4000# show interface gpon-port 1/0 igmp groups
All IGMP groups (0):
<no groups>
```

## 27 Channel aggregation groups

- [port-channel load-balance](#)
- [no interface port-channel](#)
- [lacp system-priority](#)
- [mode](#)
- [channel-group](#)
- [lacp mode](#)
- [lacp port-priority](#)
- [lacp rate](#)
- [show channel-group counters](#)
- [show channel-group hw](#)
- [show channel-group lacp](#)
- [show channel-group summary](#)

### port-channel load-balance

This command allows you to select the type of IPv4 traffic balancing.

#### Syntax

```
port-channel load-balance <METHOD>
```

#### Parameters

< METHOD > – balancing type:

- ip – based on IP address of sender and receiver;
- ip-l4 – based on IP address of sender and receiver, and L4;
- mac – based on MAC address of sender and receiver;
- mac-ip – based on MAC address and IP address of sender and receiver;
- mac-ip-l4 – based on MAC and IP addresses and L4 of sender and receiver.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# port-channel load-balance ip
```

### no interface port-channel

This command removes the group of port-channel uplink-interfaces.

#### Syntax

```
no interface port-channel <NUMBER>
```

#### Parameters

<NUMBER> – group number, may take values [1 .. 8].

**Command mode**

CONFIG

**Example**

```
ma4000(config)# no interface port-channel 2
```

**lacp system-priority**

This command sets the system priority LACP, which determines which of the switches in the interaction on the LACP protocol determines the priority ports.

The use of a negative from of the command (no) sets the default LACP system priority.

**Syntax**

```
lacp system-priority <PRIORITY>  
no lacp system-priority
```

**Parameters**

<PRIORITY> – LACP priority, may take values [0 .. 65535].

**Default value**

32768

**Command mode**

CONFIG

**Example**

```
ma4000(config)# lacp system-priority 32541
```

**mode**

This command sets the channel aggregation mode:

- use LACP channel aggregation protocol;
- do not use channel aggregation.

**Syntax**

```
mode <ACT>
```

**Parameters**

<ACT> – mode:

- lacp – use LACP;
- static – don't use LACP.

**Command mode**

PORT-CHANNEL

SLOT-CHANNEL

PLC-SLOT-CHANNEL

**Example**

```
ma4000(switch-config-slot-channel-0/0)# mode lacp
```

The channel aggregation mode is allowed on the configurable interface.

**channel-group**

This command adds a port/ports to the aggregation group.

The use of the negative form of the command (no) removes a port/port group from the aggregation group.

**Syntax**

```
channel-group <ID> <FORCE>
no channel-group
```

**Parameters**

<ID> – sequence number of the aggregation group, to which the port will be added, takes values [1 ... 8];

<FORCE> – compatibility with other members of the aggregation group:

- force – set automatic port speed and duplex;
- normal – do not change the port speed and duplex mode.

**Command mode**

FRONT-PORT

**Example**

```
ma4000(front-port-all)# channel-group 3 normal
```

All uplink ports are combined in aggregation group 3.

## lacp mode

This command allows you to select the channel aggregation mode:

- Passive – switch does not initiate creation of a logical link, but processes incoming LACP packets;
- Active – in this mode it is necessary to form an aggregated communication line and initiate coordination. Connection of communication lines is formed if the other side works in LACP active or passive modes.

The use of the negative form of the command (no) sets the default channel aggregation mode.

### Syntax

```
lacp mode <NAME>
no lacp mode
```

### Parameters

<NAME> – mode:

- active;
- passive.

### Default value

active

### Command mode

FRONT-PORT

PLC-SLOT-PORT

### Example

```
ma4000(front-port-all)# lacp mode active
```

The channel aggregation mode 'active' is enabled on the configurable ports.

## lacp port-priority

This command sets the priority for the configurable port. The priority is set in the range [1 .. 65535]. 1 is the highest priority.

The use of a negative form (no) of the command sets the default priority value.

### Syntax

```
lacp port-priority <PRIORITY>
no lacp port-priority
```

### Parameters

<PRIORITY> – the priority for this port is [0 .. 65535].

**Default value**

The default priority for all ports is 32768.

**Command mode**

FRONT-PORT

PLC-SLOT-PORT

**Example**

```
ma4000(front-port-all)# lacp port-priority 256
```

Port 256 priority is set on the configurable ports.

**lacp rate**

This command sets the transmission interval of control packets of LACPDU protocol. The use of the negative form of the command (no) sets the transmission interval of control packets of the LACPDU protocol by default.

**Syntax**

```
lacp rate <RATE>
```

```
no lacp rate
```

**Parameters**

<RATE> – transmission interval:

- fast – transmission interval is 1 second;
- slow – transmission interval is 30 seconds.

**Default value**

The default setting is 1 second (fast).

**Command mode**

FRONT-PORT

PLC-SLOT-PORT

**Example**

```
ma4000(front-port-1/1)# lacp rate slow
```

LACPDU control packets transmission interval is set to 30 seconds.

## show channel-group counters

This command allows you to view information about LACPDU counters as well as connection breaks.

### Syntax

show channel-group counters <NUMBER>

### Parameters

<NUMBER> – group number, may take values [1 .. 8].

### Command mode

ROOT

### Example

```
ma4000# show channel-group counters 1
Channel group 1
  Mode: static
  Port front-port 1/4:      Link failure count: 0      LACPDU Rx:      5      LACPDU Tx:
0
  Port front-port 2/4:      Link failure count: 0      LACPDU Rx:      5      LACPDU Tx:
0
```

## show channel-group hw

This command allows you to view the LAG aggregation group composition.

### Syntax

show channel-group hw <NUMBER>

### Parameters

<NUMBER> – group number, may take values [1 .. 8].

### Command mode

CONFIG

### Example

```
ma4000# show channel-group hw 1
Channel group 1 (2 members): front-port 2/4 [E], front-port 1/4 [E]
```

## show channel-group lacp

This command is used to view LACP information for a given aggregation group.

### Syntax

```
show channel-group lacp <NUMBER>
```

### Parameters

<NUMBER> - group number, takes the following values [1 ... 8] or '0' to view information about all groups.

### Command mode

CONFIG

### Example

```
ma4000# show channel-group lacp 1
Channel group 1
Mode: static
```

## show channel-group summary

This command allows you to view general information about the group.

### Syntax

```
show channel-group summary <NUMBER>
```

### Parameters

<NUMBER> - group number, takes the following values [1 ... 8] or '0' to view information about all groups.

### Command mode

CONFIG

### Example

```
ma4000# show channel-group summary 1
Channel group 1
Mode: static
Port front-port 1/4: [active], link up, 1 Gbps , full duplex
Port front-port 2/4: [active], link up, 1 Gbps , full duplex
```

## 28 MAC table management commands

- [mac address-table aging-time](#)
- [mac address-table learning](#)
- [show gpon network](#)
- [clear mac address-table](#)
- [show mac](#)

### mac address-table aging-time

The command sets MAC address lifetime in the table.

The use of the negative form of the command (no) sets the default MAC address lifetime.

#### Syntax

```
mac address-table aging-time <AGING TIME>
no macaddress-table aging-time
```

#### Parameters

<AGINGTIME> – MAC address lifetime, may take values [0, 10..86400] seconds. Value 0 – for the PP4X module, the lifetime of the MAC address is not limited.

#### Default value

300 seconds.

#### Command mode

CONFIG

#### Example

```
ma4000(config)# mac address-table aging-time 400
```

### mac address-table learning

This command enables the learn of MAC addresses in a given interface or VLAN.

The use of the negative form of the command (no) disables learning MAC addresses.

#### Syntax

```
[no] mac address-table learning interface <INTERFACE> <RANGE>
[no] mac address-table learning vlan <VLAN>
```

#### Parameters

<INTERFACE> – interface type: front-port; port-channel; slot-channel. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma ( , ) or specify the range using hyphen ( - ).

<VLAN> – VLAN ID, may take values [1..4094].

### Command mode

CONFIG

### Example

```
ma4000(config)# mac address-table learning vlan 1
```

## show gpon network

This command allows you to view the lifetime of the MAC address table.

### Syntax

show gpon network

### Parameters

Command contains no arguments.

### Command mode

ROOT

### Example

```
ma4000# showgponnetwork
Network:
    Mac age time:                1200
    S-VLAN ethertype:            0x8100
    C-VLAN ethertype:            0x8100
```

## clear mac address-table

This command allows to clear MAC table:

- for the whole device;
- by PLC8 or PP4X module number;
- by interface type and number;
- by VLAN.

### Syntax

clear mac address-table [interface <INTERFACE> <RANGE>|pp4|slot <ID>|vlan <VLAN>]

## Parameters

If no parameters are set, the command will clear the MAC table for the whole device.

**interface** <INTERFACE> <RANGE> – if you specify this command, the MAC table of a certain interface/group of interfaces will be cleared where:

<INTERFACE> – interface type: front-port, port-channel, slot-channel. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1.

**pp4** – when you specify this command will be cleared MAC table of the central PP4X module switch.

**slot** <ID> – If you specify this command, the MAC table of a certain PLC8 module will be cleared, where <ID> takes values [0 ..15].

**vlan** <VID> – If you specify this command, the MAC table of a particular VLAN will be cleared, where <VID> is the <VLAN> number, takes the values [1..4094].

## Command mode

ROOT

## Example

```
ma4000# clear mac address-table
```

## show mac

This command allows you to view the MAC table according to the specified filter:

- by interface type and number;
- by MAC address;
- by VLAN number.

## Syntax

```
show mac <PARAM> [<TYPE> <RULE>]
```

## Parameters

<PARAM> – selection of the MAC table:

- all – show all;
- interface <INTERFACE> <RANGE> – MAC table of a certain interface/interface group <INTERFACE>: gpon-port; ont; front-port; port-channel; slot-channel; plc-front-port; plc-slot-port; plc-mgmt-pon-port; plc-pon-port. A description of the interfaces and their <RANGE> values is given in [Table 4.1](#).
- pp4 – MAC table of the central PP4X switch modules;
- slot <ID> – MAC table of a certain PLC8 module, where <ID> takes values [0...15];

<TYPE> – filter type:

- exclude – excludes from the table entries by the given rules;
- include – includes entries according to specified rules.

<RULE> – filter operation rule:

- mac-address <MAC> – MAC selection, where <MAC> – MAC address value in format of XX:XX:XX:XX:XX:XX;
- interface <INTERFACE> – selection by selected interface;
- vlan <VID> – VLAN number selection, where <VID> – <VLAN> number, may take values [1..4094].

## Command mode

ROOT

## Example

```
ma4000# show mac interface front-port 1/0
```

Mac table (shadow)

~~~~~

VID	MAC address	Port	Type	From	To
----	-----	-----	-----	-----	-----
1	00:02:11:22:e3:b8	front-port 1/0	Dynamic	Forward	Forward
1	00:1b:21:4f:f5:ad	front-port 1/0	Dynamic	Forward	Forward
1	00:24:21:a0:9a:80	front-port 1/0	Dynamic	Forward	Forward
1	20:cf:30:bf:ac:61	front-port 1/0	Dynamic	Forward	Forward
1	20:cf:30:e8:0f:28	front-port 1/0	Dynamic	Forward	Forward
1	20:cf:30:e8:0f:66	front-port 1/0	Dynamic	Forward	Forward
1	50:46:5d:8e:27:68	front-port 1/0	Dynamic	Forward	Forward
1	90:e6:ba:1f:c0:41	front-port 1/0	Dynamic	Forward	Forward
1	90:e6:ba:9f:09:99	front-port 1/0	Dynamic	Forward	Forward
1	a8:f9:4b:80:e7:00	front-port 1/0	Dynamic	Forward	Forward
1	a8:f9:4b:80:e7:27	front-port 1/0	Dynamic	Forward	Forward
1	bc:ee:7b:73:dd:af	front-port 1/0	Dynamic	Forward	Forward
12 valid mac entries					

29 QoS configuration

- [qos default](#)
- [qos type](#)
- [qos map](#)
- [show qos](#)
- [qos cpu input-rate slot](#)
- [qos cpu quota queues](#)
- [qos cpu rate-limit](#)
- [qos cpu replication-swap-queues](#)
- [qos cpu replication-task-priority](#)
- [qos cpu wrr enable](#)
- [qos cpu wrr queues](#)
- [show slot qos cpu](#)
- [qos wrr enable](#)
- [qos wrr queues](#)
- [qos downstream-qinq-prio enable](#)
- [show slot qos general](#)

qos default

This command defines the queue, that will be used for packets without any preconfigured rules. The queue with the value of 6 is considered the highest priority.

Syntax

```
qos default <QUEUE>
```

Parameters

<QUEUE> – priority queue number, may take values [0 .. 6].

Default value

0

Command mode

CONFIG

Example

```
qos default 6
```

Packets for which no other rules are set are queued with priority 6.

qos type

This command allows you to set a rule by which to select the priority field for the package.

The traffic prioritization method will be chosen depending on the configured system rules (IEEE 802.1p/DSCP).

The system distinguishes the following traffic prioritization methods:

- all the priorities are equal;
- packet selection according to IEEE 802.1p;
- packet selection only according to IP ToS (Type of Service) on 3 level - support for Differentiated Services Codepoint (DSCP);
- interaction according to either 802.1p or DSCP/TOS.

Syntax

qos type <TYPE>

Parameters

<TYPE> – traffic prioritization type:

- 0 – all the priorities are equal;
- 1 – package selection only by 802.1p only (Priority field in 802.1Q tag);
- 2 – packet selection only by DSCP/TOS (field Differentiated Services IP packet header, senior 6 bits);
- 3 – interaction either via 802.1p or DSCP/ToS.

Default value

All the priorities are equal.

Command mode

CONFIG

Example

```
ma4000(config)# qos type 2
```

Traffic prioritization will be done only via DSCP/TOS.

qos map

This command sets the parameters for the priority queue:

- specifies the value of the field Differentiated Services IP packet header, senior 6 bits;
- value of the Priority field in 802.1Q tag.

Based on the rules set by the qos type command and the specified priority values, packages are selected for this priority queue.

The use of the negative form of the command (no) allows you to remove the entry from the queue settings table.

Syntax

[no] qos map <TYPE> <VALUES> to <QUEUE>

Parameters

<TYPE> – traffic prioritization type:

- 0 – by the 1p standard (used on level 2);

- 1 – by the DSCP/TOS standard (used on level 3).

<VALUES> – value of the field by which the packets are selected is set according to <TYPE> (the values of the fields are entered with a comma or as a range with hyphen (-)):

- if <TYPE> = 0, then the Priority field value is set to 802.1Q Tag: [0 ... 7];
- if <TYPE> = 1, then set the values of the fields Differentiated Services of the IP packet header, the highest 6 bits. The value is entered in hexadecimal format: [0 .. 63].

<QUEUE> – priority queue number, may take values [0 .. 6].

Command mode

CONFIG

Example

```
ma4000(config)# qosmap 0 0 to 1
```

For the 7th priority queue the value of the field priority = 7 in 802.1Q tag.

show qos

This command is used to view the priorities assigned to the queues. The default queue priority is 0. The priority value for the queue is set in the range [0 .. 6], the queue with the priority 6 is considered the most priority.

Syntax

show qos

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```
ma4000# show qos
Priority assignment by NONE packet field, all priorities are equal
Default priority queue is 0
DSCP/TOS queues:
0:
1:
2:
3:
4:
5:
6:
802.1p queues:
0:
1:
2:
3:
4:
5:
6:
```

qos cpu input-rate slot

This command enables a restriction on packet transfer to the CPU for PLC8 line cards.

The negative form of the command sets the default value for this parameter.

Syntax

```
qos cpu input-rate slot <SLOT>
no qos cpu input-rate slot <SLOT>
```

Parameters

<SLOT> – slot number [0..15]

Command mode

CONFIG

Example

```
ma4000# qos cpu input-rate slot 6
```

qos cpu quota queues

This command allows you to set the buffer size for each of the queues to transmit packets to the CPU for PLC line cards.

Syntax

```
qos cpu quota queues <Quota7> <Quota6> <Quota5> <Quota4> <Quota3> <Quota2> <Quota1>
<Quota0> slot <SLOT>
```

Parameters

<QuotaN> – buffer size for the corresponding queue;

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000# qos cpu quota queues 22 33 33 22 33 33 3 3 slot 6
```

qos cpu rate-limit

This command sets a limit on the number of packets intercepted to be sent to the CPU for PLC8 line cards.

Syntax

```
qos cpu rate-limit <TYPE> <RATE> slot <SLOT>
```

Parameters

<TYPE> – traffic type:

- mgmt – management traffic;
- mgmt-pon – PON port management traffic;
- arp – ARP packets;
- igmp – IGMP packets;
- lacp – LACP packets;
- green-replication – packets for replication by green queue;
- yellow-replication – packets for replication by yellow queue;
- pon-interception;

<RATE> – amount of MAC addresses per second, takes value of [1..10000];

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000# qos cpu rate-limitmgmt 10000 slot 6
```

qos cpu replication-swap-queues

This command inverts the green and yellow packet replication queue processing mechanisms on the CPU for PLC8 line cards.

The negative form of the command sets the default value for this parameter.

Syntax

```
qos cpu replication-swap-queues slot <SLOT>  
no qos cpu replication-swap-queues slot <SLOT>
```

Parameters

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000# qos cpu replication-swap-queue slot 6
```

qos cpu replication-task-priority

This command allows to set the priority for processing the replication of traffic by the system to the CPU for PLC8 line cards.

Syntax

```
qos cpu replication-task-priority <TASK> slot <SLOT>
```

Parameters

<TASK> – task execution priority, may take values [0..20];

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000# qos cpu replication-task-priority 2 slot 6
```

qos cpu wrr enable

This command enables wrr-mode of priority queues processing on CPU for PLC8 line cards.
The negative form of the command sets the default value for this parameter.

Syntax

```
qos cpu wrr enable slot <SLOT>
no qos cpu wrr enable slot <SLOT>
```

Parameters

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000# qos cpu wrr enable slot 6
```

qos cpu wrr queues

This command sets the buffer size for each of the queues on the CPU for PLC8 line cards.

Syntax

```
qos cpu wrr queues <Quota7> <Quota6> <Quota5> <Quota4> <Quota3> <Quota2> <Quota1>
<Quota0> slot <SLOT>
```

Parameters

<QuotaN> – buffer size for the corresponding queue;

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000# qos cpu wrr queues 2 2 2 2 2 2 2 2 slot 6
```

show slot qos cpu

This command displays the configuration of traffic processing queues on the CPU for PLC8 line cards.

Syntax

show slot < SLOT > qos cpu

Parameters

< SLOT > – slot number, may take values [0..15].

Command mode

ROOT

Example

```

ma4000# show slot 6 qos cpu

WRR enabled
WRR values for queues 7..0: 8 7 6 5 4 3 2 1
Quota values for queues 7..0: 1 1 1 1 1 1 1 1
Rate-limit for Mgmt: 2500
Rate-limit for Mgmt for PON: 2500
Rate-limit for ARP: 500
Rate-limit for IGMP: 500
Rate-limit for LACP: 50
Rate-limit for Replication green queue: 30
Rate-limit for Replication yellow queue: 30
Rate-limit for Intercepted packets by PON: 500
Rate-limit for Unknown: 100
Rate-limit for Unknown: 100
Rate-limit for Unknown: 1000
Rate-limit for Unknown: 2500
CPU input rate service: enabled
Replication task priority: 0
Swapping replications green and yellow queues: disabled

```

qos wrr enable

This command enables wrr mode of priority queues processing on the PLC line card switch.

The negative form of the command sets the default value for this parameter.

Syntax

qos wrr enable slot < SLOT >
no qos wrr enable slot < SLOT >

Parameters

< SLOT > – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000(config)# qos wrr enable slot 6
```

qos wrr queues

This command sets the buffer size for each of the queues on the PLC8 line cards switch.

Syntax

```
qos wrr queues <Quota6> <Quota5> <Quota4> <Quota3> <Quota2> <Quota1> <Quota0> slot
<SLOT>
```

Parameters

<QuotaN> – buffer size for the corresponding queue;

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000(config)# qos wrr queues 1 1 1 1 22 22 33 slot 2
```

qos downstream-qinq-prio enable

This command sets the p-bit mode from outer-vid to inner-vid Q-in-Q packets replacing on the PLC8 line card switch to provide QoS traffic transfer.

Syntax

```
qos downstream-qinq-prio enable slot <SLOT>
```

Parameters

<SLOT> – slot number, may take values [0..15].

Command mode

CONFIG

Example

```
ma4000(config)# qos downstream-qinq-prio enable slot 6
```

show slot<SLOT> qos general

This command displays the QoS settings on the PLC8 line card switch.

Syntax

```
show slot <SLOT> qos general
```

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show slot 6 qos general

Priority assignment by 802.1p packet field
Priority assignment by inner tag of Q-in-Q packet in downstream is disabled
Default priority queue is 0
DSCP/TOS queues:
6:
5:
4:
3:
2:
1:
0:
802.1p queues:
6: 7
5: 6
4: 5
3: 4
2: 3
1: 2
0: 0-1
WRR enabled
WRR values for queues 6..0: 7 6 5 4 3 2 1
```

30 Port isolation group configuration

- [isolation group](#)
- [allow](#)
- [isolation enable](#)
- [isolation assign](#)
- [show bridging](#)
- [show isolation vlans](#)
- [show isolation groups](#)
- [show slot isolation vlans](#)
- [show show slot isolation groups](#)

isolation group

This command allows you to switch to the group/group of isolated ports configuration mode.

✓ The function is not compatible with operation on model 1.

Syntax

isolation group <RANGE>

Parameters

<RANGE> – isolation group number, may take values [0..29].

Command mode

CONFIG

Example

```
ma4000(config)# isolation group 1
ma4000(pp4x-config-isolation-1)#
```

allow

This command adds interfaces to the port isolation group, with which information exchange in this group will be allowed.

The use of a negative form (no) of the command removes port/ports from port isolation group.

Syntax

[no] allow <INTERFACE> <RANGE>

Parameters

<INTERFACE> – interface type: front-port; port-channel; slot-channel. The description of interfaces is given in Table 4.1.

<RANGE> – interface number. The range of values and numbering rules are described in Table 4.1. If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ISOLATION GROUP

Example

```
ma4000(pp4x-config-isolation-2)# allow front-port 1/5
```

isolation enable

This command enables port isolation feature for configured VLAN ports.

The use of a negative form (no) of the command disables this feature.

Syntax

[no] isolation enable

Parameters

The command contains no arguments.

Command mode

VLAN

Example

```
ma4000(vlan-1)# isolation enable
```

isolation assign

This command is used to assign an isolation group for the interface globally in CONFIG mode or in VLAN mode for the interface of the configured VLAN.

The use of a negative form (no) of the command removes port from the group.

Syntax

[no] isolation assign <INTERFACE> <RANGE> group <GROUP RANGE>

Parameters

<INTERFACE> – interface type: front-port; port-channel; slot-channel. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

<GROUPRANGE> – port isolation group number, may take values [0..29].

Command mode

VLAN

CONFIG

Example

```
ma4000(config)# isolation assign front-port all group 1
```

show bridging

This command is used to view the port isolation settings.

Syntax

```
show bridging <INTERFACE> <RANGE>
```

Parameters

<INTERFACE> – interface type: front-port; gpon-port; ont; plc-front-port; plc-mgmt-pon-port; plc-pon-port; plc-slot-channel; plc-slot-port; port-channel; slot-channel; slot-port; stack-port. The description of interfaces is given in [Table 4.1](#).

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show bridging front-port 1/5

  Bridging settings
  ~~~~~

Source          Destination          Traffic restriction
-----          -
front-port 1/5  front-port 1/0        Allow
                front-port 1/1        Allow
                front-port 1/2        Allow
                front-port 1/3        Allow
                front-port 1/4        Allow
                front-port 1/5        Allow
                front-port 2/0        Allow
                front-port 2/1        Allow

...

                slot-channel 7        Allow
                slot-channel 8        Allow
                slot-channel 9        Allow
                slot-channel 10       Allow
                slot-channel 11       Allow
                slot-channel 12       Allow
                slot-channel 13       Allow
                slot-channel 14       Allow
                slot-channel 15       Allow

-----          -
```

show isolation vlans

This command allows to view the port isolation status by VLAN.

Syntax

show isolation vlans <VID>

Parameters

<VID> – VLAN ID, may take values [1 .. 4094].

Command mode

ROOT

Example

```
ma4000# show isolation vlans 1

Isolation per vlan:
~~~~~

VID      State      Interface      Destination
-----  -
1        disabled  front-port 1/0  Any
          front-port 1/1  Any
          front-port 1/2  Any
          front-port 1/3  Any
          front-port 1/4  Any
          front-port 1/5  Any
          front-port 2/0  Any
          front-port 2/1  Any
          front-port 2/2  Any
          front-port 2/3  Any
          front-port 2/4  Any
          front-port 2/5  Any
          slot-channel 0   0
          slot-channel 1   0
          slot-channel 2   0
          slot-channel 3   0
          slot-channel 4   0
          slot-channel 5   0
          slot-channel 6   0
          slot-channel 7   0
          slot-channel 8   0
          slot-channel 9   0
          slot-channel 10  0
          slot-channel 11  0
          slot-channel 12  0
          slot-channel 13  0
          slot-channel 14  0
          slot-channel 15  0
-----  -
```

show isolation groups

This command allows to view the port isolation status by groups.

Syntax

show isolation groups <GROUP RANGE>

Parameters

<GROUPRANGE> – isolation group number, may take values [0 .. 29].

Command mode

ROOT

Example

```
ma4000# show isolation groups 1
```

```
Isolation groups:
```

```
~~~~~
```

Group	Interface	Traffic restriction flag

1	front-port 1/0	Allow
	front-port 1/1	Allow
	front-port 1/2	Allow
	front-port 1/3	Allow
	front-port 1/4	Allow
	front-port 1/5	Allow
	front-port 2/0	Allow
	front-port 2/1	Allow
	front-port 2/2	Allow
	front-port 2/3	Allow
	front-port 2/4	Allow
	front-port 2/5	Allow
	slot-channel 0	Deny
	slot-channel 1	Deny
	slot-channel 2	Deny
	slot-channel 3	Deny
	slot-channel 4	Deny
	slot-channel 5	Deny
	slot-channel 6	Deny
	slot-channel 7	Deny
	slot-channel 8	Deny
	slot-channel 9	Deny
	slot-channel 10	Deny
	slot-channel 11	Deny
	slot-channel 12	Deny
	slot-channel 13	Deny
	slot-channel 14	Deny
	slot-channel 15	Deny

show slot <SLOT> isolation vlans

This command allows to view the port isolation status by VLAN for PLC8 line cards.

Syntax

```
show slot <SLOT> isolation vlans <VID>
```

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

<VID> – VLAN ID, may take values [1 .. 4094].

Command mode

ROOT

Example

```
ma4000# show slot 6 isolation vlans 1

Isolation per vlan:
~~~~~
VID      State      Interface              Destination group
-----
1        disabled    plc-front-port 6/0     0
                plc-pon-port 6/0       1
                plc-pon-port 6/1       1
                plc-pon-port 6/2       1
                plc-pon-port 6/3       1
                plc-pon-port 6/4       1
                plc-pon-port 6/5       1
                plc-pon-port 6/6       1
                plc-pon-port 6/7       1
                plc-slot-channel 6/0    0
-----
```

show show slot <SLOT> isolation groups

This command allows to view the port isolation status by groups for PLC8 line cards.

Syntax

show slot <SLOT> isolation groups <GROUP RANGE>

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

<GROUPRANGE> – isolation group number, may take values [0 .. 29].

Command mode

ROOT

Example

```
ma4000# show slot 6 isolation groups 1
```

```
Isolation groups:
```

```
~~~~~
```

Group	Interface	Traffic restriction flag
-----	-----	-----
1	plc-front-port 6/0	allow
	plc-pon-port 6/0	deny
	plc-pon-port 6/1	deny
	plc-pon-port 6/2	deny
	plc-pon-port 6/3	deny
	plc-pon-port 6/4	deny
	plc-pon-port 6/5	deny
	plc-pon-port 6/6	deny
	plc-pon-port 6/7	deny
	plc-slot-channel 6/0	allow
-----	-----	-----

31 Selective Q-in-Q feature configuration

- [selective-qinq common](#)
- [selective-qinq list](#)
- [add-tag](#)
- [overwrite-tag](#)
- [remove](#)
- [clear](#)
- [selective-qinq enable](#)
- [show selective-qinq](#)

For general settings of the Selective Q-in-Q feature the **SELECTIVE Q-IN-Q COMMON** command mode is intended. The **SELECTIVE Q-IN-Q LIST** command mode is used to set the Selective Q-in-Q rules list.

selective-qinq common

This command transitions to the mode of general settings of the Selective Q-in-Q function.

Syntax

```
selective-qinq common
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# selective-qinq common
ma4000(config-selective-qinq)#
```

selective-qinq list

In the CONFIG mode, this command transitions to the configuration mode of the Selective Q-in-Q rules list.

In the interface/interface group configuration mode (SLOT-CHANNEL, PORT-CHANNEL, FRONT-PORT) this command assigns a certain list of rules to the interface being configured.

Syntax

```
selective-qinq list <NAME>
```

Parameters

<NAME> – Selective Q-in-Q rules list name, 31 characters max.

Command mode

CONFIG

SLOT-CHANNEL

PORT-CHANNEL

FRONT-PORT

Example

```
ma4000(config)# selective-qinq list test
ma4000(config-selective-qinq-test)#
```

add-tag

This command adds an outer tag based on the inner tag.

The use of a negative form (no) of the command removes the set rule.

Syntax

```
[no] add-tag svlan <S-VLAN> cvlan <C-VLAN>
```

Parameters

<S-VLAN> – outer tag number, may take values [1..4095];

<C-VLAN> – inner tag number(s), may take values 1-4094 or ignore. The C-VLAN list is defined using commas (,). If the option 'ignore' is selected, the tag will be added to any incoming package.

Command mode

SELECTIVE Q-IN-Q COMMON

SELECTIVE Q-IN-Q LIST

Example

```
ma4000(config-selective-qinq-test)# add-tag svlan 3 cvlan 2,4-100
```

overwrite-tag

This command is used to substitute SVLAN in the required direction.

The use of a negative form (no) of the command removes the set rule.

Syntax

```
[no] overwrite-tag new-vlan <NEW-VLAN> old-vlan <OLD-VLAN> <RULE_DIRECTION>
```

Parameters

<NEW-VLAN> – new VLAN number, may take values [1 ..4095];

<OLD-VLAN> – number of VLAN, which should be substituted, may take values [1 .. 4094];

<RULE_DIRECTION> – traffic direction:

- ingress – incoming;
- egress – outgoing.

Command mode

SELECTIVE Q-IN-Q COMMON

SELECTIVE Q-IN-Q LIST

Example

```
ma4000(config-selective-qinq-test)# overwrite-tag new-vlan 555 old-vlan 111 ingress
```

remove

This command removes the Selective Q-in-Q rule by the specified number.

Syntax

remove <RULE_INDEX>

Parameters

<RULE_INDEX> – rule number, may take values [0 .. 511].

Command mode

SELECTIVE Q-IN-Q COMMON

SELECTIVE Q-IN-Q LIST

Example

```
ma4000(config-selective-qinq-test)# remove 0
```

clear

This command removes all Selective Q-in-Q rules.

Syntax

clear

Parameters

The command contains no arguments.

Command mode

SELECTIVE Q-IN-Q COMMON

SELECTIVE Q-IN-Q LIST

Example

```
ma4000(config-selective-qinq-test)# clear
```

selective-qinq enable

This command enables the Selective Q-in-Q feature on the configured interface.

Syntax

```
[no] selective-qinq enable
```

Parameters

The command contains no arguments.

Command mode

SLOT-CHANNEL

PORT-CHANNEL

FRONT-PORT

Example

```
ma4000(express-config-port-channel-1)# no selective-qinq enable
```

show selective-qinq

This command allows to view the list of Selective Q-in-Q feature rules.

Syntax

```
show selective-qinq <PARAM>
```

Parameters

<PARAM> – output to display:

- common – view common rules;

- list – view rules of the certain list. It is necessary to specify the 'name' of the list;
- lists – view rules in all lists.

Command mode

ROOT

Example

```
ma4000# show selective-qinq common

      Selective Q-in-Q common rules
      ~~~~~
Index  Rule                                     Direction
-----
0      Add S-VLAN 20 to C-VLAN 100-201        Ingress
```

32 DHCP client configuration for management interface

- [dhclient lease-time](#)
- [dhclient reboot](#)
- [dhclient retry](#)
- [dhclient select-timeout](#)
- [dhclient timeout](#)

dhclient lease-time

This command sets the requested lease time for the IP address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dhclient lease-time <TIME>  
no dhclient lease-time
```

Parameters

<TIME> – IP address leasing time, may take values [0 .. 3600] s.

Default value

100 seconds

Command mode

CONFIG

Example

```
ma4000(config)# dhclient lease-time 200
```

dhclient reboot

This command specifies the time during which a restarted DHCP client requests a previously issued IP address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dhclient reboot <TIME>  
no dhclient reboot
```

Parameters

<TIME> – IP address request time, may take values [0 .. 3600] s.

Default value

10 seconds

Command mode

CONFIG

Example

```
ma4000(config)# dhclient reboot 20
```

dhclient retry

This command sets the time between attempts to obtain an IP address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dhclient retry <TIME>  
no dhclient retry
```

Parameters

<TIME> – timeout for IP address re-obtaining, may take values [0 .. 3600] s.

Default value

120 seconds

Command mode

CONFIG

Example

```
ma4000(config)# dhclient retry 120
```

dhclient select-timeout

This command sets the time of waiting for the response from the DHCP server to the request.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dhclient select-timeout <TIME>  
no dhclient select-timeout
```

Parameters

<TIME> – DHCP server response waiting time, take values [0 .. 3600] s.

Default value

3 seconds

Command mode

CONFIG

Example

```
ma4000(config)# dhclientselct-timeout 3
```

dhclient timeout

This command specifies the time during which a DHCP client tries to obtain an IP address.

The use of a negative form (no) of the command sets the default value.

Syntax

```
dhclient timeout <TIME>
```

```
no dhclient timeout
```

Parameters

<TIME> – time for IP address obtaining by DHCP client, may take values [0 .. 3600] s.

Default value

60 seconds

Command mode

CONFIG

Example

```
ma4000(config)# dhclient timeout 90
```

33 Configuring LLDP

- [lldp enable](#)
- [lldp hold-multiplier](#)
- [lldp reinit](#)
- [lldp timer](#)
- [lldp tx-delay](#)
- [lldp lldpdu](#)
- [lldp mode](#)
- [show lldp configuration](#)
- [show lldp neighbor](#)
- [show lldp local](#)
- [show lldp statistics](#)

lldp enable

This command enables switch operation via LLDP.

The use of a negative form (no) of the command disables LLDP protocol usage by switch.

Syntax

```
[no] lldp enable
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# lldp enable
```

lldp hold-multiplier

This command specifies the amount of time for the receiver to keep LLDP packets before dropping them.

This value will be transmitted to the receiving side in the LLDP update packets; and should be an increment for the LLDP timer. Thus, the lifetime of LLDP packets is calculated by the formula: $TTL = \min(65535, \text{LLDP-Timer} * \text{LLDP-HoldMultiplier})$.

The use of a negative form (no) of the command sets the default value.

Syntax

```
lldp hold-multiplier <HOLD>  
no lldp hold-multiplier
```

Parameters

<HOLD> – time, may take values [2 .. 10] seconds.

Default value

4 seconds

Command mode

CONFIG

Example

```
ma4000(config)# lldp hold-multiplier 5
```

lldp reinit

This command sets minimum amount of time for the LLDP port to wait before LLDP reinitialization.
The use of a negative form (no) of the command sets the default value.

Syntax

```
lldp reinit <REINIT>  
no lldp reinit
```

Parameters

<REINIT> – time, may take values [1 .. 10] seconds.

Default value

2 seconds

Command mode

CONFIG

Example

```
ma4000(config)# lldp reinit 3
```

lldp timer

This command specifies how frequently the device will send LLDP information updates.
The use of a negative form (no) of the command sets the default value.

Syntax

```
lldp timer <TIMER>  
no lldp timer
```

Parameters

<TIMER> – time, may take values [5..32768] seconds.

Default value

30 seconds

Command mode

CONFIG

Example

```
ma4000(config)# lldp timer 60
```

lldp tx-delay

This command specifies the delay between the subsequent LLDP packet transmissions caused by the changes of values or status in the local LLDP MIB database.

It is recommended that this delay be less than 0.25* LLDP-Timer.

The use of a negative form (no) of the command sets the default value.

Syntax

```
lldp tx-delay <TXDELAY>  
no lldp tx-delay
```

Parameters

<TXDELAY> – time, may take values [1..8192] seconds.

Default value

2 seconds

Command mode

CONFIG

Example

```
ma4000(config)# lldp tx-delay 3
```

lldp lldpdu

This command sets the LLDP packet processing mode when LLDP is disabled.

The use of a negative form (no) of the command sets the default value (filtering).

Syntax

```
lldp lldpdu <MODE>
no lldp lldpdu
```

Parameters

<MODE> – LLDP packets processing mode:

- filtering – LLDP packets are filtered if LLDP is disabled on the switch;
- flooding – LLDP packets are transmitted if LLDP is disabled on the switch.

Command mode

CONFIG

Example

```
ma4000(config)# lldp lldpdu flooding
```

lldp mode

This command sets the operating mode of the LLDP protocol for a specific interface.

The use of a negative form (no) of the command sets the default value.

Syntax

```
lldp mode <MODE>
no lldp mode
```

Parameters

<MODE> – operation mode of LLDP on interface:

- disabled – denies operation via LLDP;
- transmit-only – allows only the transmission of packets via LLDP;
- receive-only – allows only the reception of packets via LLDP;
- transmit-receive – allows transmission and reception of packets via LLDP.

Default value

transmit-receive

Command mode

FRONT-PORT

Example

```
ma4000(front-port-all)# lldp mode receive-only
```

show lldp configuration

This command allows to view the LLDP configuration of all physical interfaces of the device or specified interfaces.

Syntax

```
show lldp configuration [front-port <RANGE>]
```

Parameters

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified. You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show lldp configuration
LLDP state: Enabled
Timer: 30 Seconds
Hold multiplier: 4
Reinit delay: 2 Seconds
Tx delay: 2 Seconds
LLDP packets handling: Flooding
```

```

  LLDP configuration
  ~~~~~
Interface          Status          Optional TLVs
-----
front-port 1/0     transmit-receive
front-port 1/1     transmit-receive
front-port 1/2     transmit-receive
front-port 1/3     transmit-receive
front-port 1/4     transmit-receive
front-port 1/5     transmit-receive
front-port 2/0     transmit-receive
front-port 2/1     transmit-receive
front-port 2/2     transmit-receive
front-port 2/3     transmit-receive
front-port 2/4     transmit-receive
front-port 2/5     transmit-receive
PD - port description, SN - system name, SD - system description, SC - system capabilities
```

show lldp neighbor

This command allows to view information on the neighbour devices on which LLDP is enabled.

Syntax

```
show lldp neighbor [front-port <RANGE>]
```

Parameters

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show lldp neighbor

LLDP neighbors
~~~~~
Interface      Device ID      Port ID      System Name      Capabilities
TTL
-----
front-port 1/0  00:02:11:22:e3:a9  g15          94
/120
```

show lldp local

This command allows to view the LLDP information announced by this interface.

Syntax

```
show lldp local [front-port <RANGE>]
```

Parameters

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.

You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show lldp local

  LLDP local TLVs
  ~~~~~
Interface          Device ID          Port ID
TTL
-----
front-port 1/0     a8:f9:4b:88:33:a0  front-port 1/0
120
```

show lldp statistics

This command allows to view LLDP statistics for front-port, port-channel interfaces.

Syntax

```
show lldp statistics [front-port <RANGE>]
```

Parameters

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified.
You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show lldp statistics
```

```
Tables Last Change Time: 0:0:2:3
```

```
Tables Inserts: 1
```

```
Tables Deletes: 0
```

```
Tables Dropped: 0
```

```
Tables Ageouts: 0
```

```
LLDP statistics
```

```
~~~~~
```

```
Interface      Tx total Rx total Rx errors Rx discarded TLVs discarded TLVs unrecognized Agouts
total
```

```
-----
```

front-port 1/0	1060	1059	0	0	0	0	0
front-port 1/1	0	0	0	0	0	0	0
front-port 1/2	0	0	0	0	0	0	0
front-port 1/3	0	0	0	0	0	0	0
front-port 1/4	0	0	0	0	0	0	0
front-port 1/5	0	0	0	0	0	0	0
front-port 2/0	0	0	0	0	0	0	0
front-port 2/1	0	0	0	0	0	0	0
front-port 2/2	0	0	0	0	0	0	0
front-port 2/3	0	0	0	0	0	0	0
front-port 2/4	0	0	0	0	0	0	0
front-port 2/5	0	0	0	0	0	0	0

34 SNTP configuration

- [ip sntp client](#)
- [ip sntp poll-period](#)
- [ip sntp server](#)

ip sntp client

This command enables the SNTP client to synchronize time with the server.

The use of a negative form (no) of the command disables SNTP client.

Syntax

```
[no] ip sntp client
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# ip sntp client
```

ip sntp poll-period

This command sets the synchronization period with the SNTP server.

The use of a negative form (no) of the command sets the default synchronization period with the SNTP server.

Syntax

```
ip sntp poll-period <PERIOD>  
no ip sntp poll-period
```

Parameters

<PERIOD> – synchronization period, may take values [10 ... 1440] minutes.

Default value

300 minutes

Command mode

CONFIG

Example

```
ma4000(config)# ip sntp poll-period 500
```

ip sntp server

This command sets the IP address or host name of the SNTP server for synchronization.

The use of a negative form (no) of the command deletes the IP address or host name of the SNTP server.

Syntax

```
[no] ip sntp server <HOST>
```

Parameters

<HOST> – IP address/SNTP server hostname.

Command mode

CONFIG

Example

```
ma4000(config)# ip sntp server
```

35 System log configuration

- [logging console](#)
- [logging file](#)
- [logging file-size](#)
- [logging filter](#)
- [logging builtin-filter](#)
- [logging host](#)
- [logging max-files](#)
- [logging storage persistent](#)
- [logging monitor](#)
- [clear log](#)
- [show log](#)
- [show logging config](#)

The **LOGGING** commands allow you to configure such parameters as log storage location (local file, remote file), number and size of these files, message filtering rules.

logging console

The command allows you to set the level of syslog messages that will be displayed in the console in one of two modes: either display all messages with the level not lower than the specified one, or only with this level.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] logging console <SEVERITY>
```

Parameters

<SEVERITY> – syslog message level for output: 'emerg', 'alert', 'crit', 'error', 'warning', 'notice', 'info', 'debug'.

Default value

info

Command mode

CONFIG

Example

```
ma4000(config)# logging console debug
```

logging file

This command specifies the name of the local file where the log is stored, as well as the level of messages that are stored in this file.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] logging file <FILE> <SEVERITY>
```

Parameters

<FILE> – file name, up to 255 characters long;

<SEVERITY> – level of messages, that saved in file.

Command mode

CONFIG

Example

```
ma4000(config)# logging file test alert
```

logging file-size

This command allows to set the maximum log file size in kilobytes. When one file is exhausted, the system will create the next one according to the max-files setting (see loggingmax-files).

The use of a negative form (no) of the command restores the default value.

Syntax

```
logging file-size <SIZE>  
no logging file-size
```

Parameters

<SIZE> – allowed size of log file. Values between 100 and 10000 Kbytes are allowed.

Default value

500 KB

Command mode

CONFIG

Example

```
ma4000(config)# logging file-size 1000
```

logging filter

The command allows to switch to the configuration mode of a separate filter rule by its name, a detailed description of the LOGFILTER mode is given in [Syslog filters configuration. LOGFILTER command mode](#) section.

The use of the negative form (no) of the command remove filter by its name.

Syntax

```
[no] logging filter <FILE>
```

Parameters

<FILE> – file name, up to 255 characters.

Command mode

CONFIG

Example

```
ma4000(config)# logging filter test
ma4000(pp4x-config-log-filter-test)#
```

logging builtin-filter

The command allows to switch to the configuration mode of the filter rule of a predefined slot.

Syntax

```
logging builtin-filter <NAME>
```

Parameters

<NAME> – slot name or PP4X card, may take values: 'pp', 'pp-other', 'slot0', 'slot1', 'slot2', 'slot3', 'slot4', 'slot5', 'slot6', 'slot7', 'slot8', 'slot9', 'slot10', 'slot11', 'slot12', 'slot13', 'slot14', 'slot15'.

Command mode

CONFIG

Example

```
ma4000(config)# logging builtin-filter pp
ma4000(pp4x-config-log-filter-pp)#
```

logging host

This command allows to set the mode of log saving to a remote address.

The use of a negative form (no) of the command cancels the remote logging.

Syntax

```
logging host <HOST> port <PORT> transport <TRANSPORT> <SEVERITY>  
no logging host
```

Parameters

<HOST> – IP address of the remote host to save the log file;

<PORT> – port number for communication with the remote host from 1 to 65535;

<TRANSPORT> – transmission packets type: tcp or udp;

<SEVERITY> – level of messages, that saved in file.

Command mode

CONFIG

Example

```
ma4000(config)# logging host 192.168.1.2 port 1024 transport tcp debug
```

logging max-files

This command specifies the maximum number of log-files to be stored in the system. When the last available one is filled in, the system will go back to the first one.

The use of a negative form (no) of the command returns the default value.

Syntax

```
logging max-files <FILE_NUM>  
no logging max-files
```

Parameters

<FILE_NUM> – maximum log file number, may take values [1 .. 1000].

Default value

3

Command mode

CONFIG

Example

```
ma4000(config)# logging max-files 20
```

logging storage persistent

The command allows to specify the mode of saving log-files in memory. Log files will either be created from scratch every time you start the MA4000, or saved between restarts.

The use of a negative form (no) of the command returns the storage mode to temporary.

Syntax

```
[no] logging storage persistent
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# logging storage persistent
```

logging monitor

The command allows to set the level of SYSLOG messages that will be displayed at remote SSH and TELNET sessions in one of two modes: either display all messages with the level not lower than the specified one, or only with this level.

The use of a negative form (no) of the command removes a set mode.

Syntax

```
logging monitor <SEVERITY>  
no logging monitor
```

Parameters

<SEVERITY> – syslog message level for output: 'emerg', 'alert', 'crit', 'error', 'warning', 'notice', 'info', 'debug'.

Default value

info

Command mode

CONFIG

Example

```
ma4000(config)# logging monitor debug
```

clear log

This command allows to delete log-files stored in local memory of MA4000-PX.

Syntax

```
clear log [file <FILE>]
```

Parameters

<FILE> – log file name to delete, optional . If file name is not specified, all log files stored in local device memory will be deleted.

Command mode

ROOT

Example

```
ma4000# clear log file slot0
```

show log

This command allows to view the list of log files stored in local memory of MA4000-PX. If file name is specified, the content of the log file will be displayed.

Syntax

```
show log [<FILE>]
```

Parameters

<FILE> – log file name, optional.

Command mode

ROOT

Example

```
ma4000# showlog
```

```
Log files
```

```
~~~~~
```

##	Name	Size in bytes	Date of last modification
1	daemon	447	Tue Feb 4 12:01:39 2014
2	pp	188073	Tue Feb 4 12:01:39 2014
3	slot0	6132	Tue Feb 4 12:03:41 2014
4	slot0.1	511983	Tue Feb 4 09:23:56 2014
5	slot0.2	511867	Tue Feb 4 09:23:56 2014
6	slot0.3	511915	Tue Feb 4 09:23:56 2014

```
-----
```

```
Total files: 6
```

show logging config

This command allows to view log settings for saving MA4000 log files.

Syntax

```
show logging config
```

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```
ma4000# show logging config
```

```
    Logging max files - 3
```

```
    Logging file size - 500
```

```
Criteria filter table:
```

```
~~~~~
```

Name	Emerg	Alert	Crit	Error	Warn	Notce	Info	Debug
console	+	+	+	+	+	+	-	-
monitor	+	+	+	+	+	+	+	+
slot0	+	+	+	+	+	+	+	-
slot1	+	+	+	+	+	+	+	-
slot2	+	+	+	+	+	+	+	-
slot3	+	+	+	+	+	+	+	-
slot4	+	+	+	+	+	+	+	-
slot5	+	+	+	+	+	+	+	-
slot6	+	+	+	+	+	+	+	-
slot7	+	+	+	+	+	+	+	-
slot8	+	+	+	+	+	+	+	-
slot9	+	+	+	+	+	+	+	-
slot10	+	+	+	+	+	+	+	-
slot11	+	+	+	+	+	+	+	-
slot12	+	+	+	+	+	+	+	-
slot13	+	+	+	+	+	+	+	-
slot14	+	+	+	+	+	+	+	-
slot15	+	+	+	+	+	+	+	-
pp	+	+	+	+	+	+	+	-
pp-other	+	+	+	+	+	+	+	-

```
File target table:
```

```
~~~~~
```

Filter name	File name
slot0	slot0
slot1	slot1
slot2	slot2
slot3	slot3
slot4	slot4
slot5	slot5
slot6	slot6
slot7	slot7
slot8	slot8
slot9	slot9
slot10	slot10
slot11	slot11
slot12	slot12
slot13	slot13
slot14	slot14
slot15	slot15
pp	pp
pp-other	pp-other
load-1-3-2_51#	

36 Syslog filters configuration. LOGFILTER command mode

- [destination](#)
- [match-source](#)
- [facility](#)
- [match](#)

SYSLOG message filters are set up in the **LOGFILTER mode**. This mode is available from the **CONFIG** mode.

To configure a custom SYSLOG filter, run the command:

```
ma4000(config)# logging filter <NAME>
ma4000(pp4x-config-log-filter-<NAME>)#
```

where <NAME> – filter name.

To set a predefined SYSLOG filter for a given PLC8 or PP4X module (by default), execute the command:

```
ma4000(pp4x-config)# logging builtin-filter <NAME>
ma4000(pp4x-config-log-filter-<NAME>)#
```

where <NAME> – PLC8 or PP4X module name, may take values: 'pp' 'pp-other' 'slot0' 'slot1' 'slot2' 'slot3' 'slot4' 'slot5' 'slot6' 'slot7' 'slot8' 'slot9' 'slot10' 'slot11' 'slot12' 'slot13' 'slot14' 'slot15'.

destination

This command configures the destination for outputting SYSLOG messages of the filter being configured:

- to the file;
- to the console;
- to the remote host.

The use of a negative form (no) of the command cancels the specified direction.

Syntax

[no] destination <TYPE>

Parameters

<TYPE> – SYSLOG message destination:

- console – output to the console;
- file – output to the file, specified in the format: file <FILE>, where: <FILE> – file name in the local file system;
- host – output to the remote address, specified in the format: host <HOST> port <PORT> transport <TRANSPORT>, where:
 - <HOST> – remote host IP address;
 - <PORT> – connection port number [1..65535];
 - <TRANSPORT> – connection protocol: tcp, udp.
- monitor – output to the PC monitor.

Command mode

LOG FILTER

Example

```
ma4000(pp4x-config-log-filter-test)# destination console
```

match-source

This command configures the message source for analysis: PP4X module (master/slave), PLC8 module number. Multiple sources can be set for analysis.

For the filter to be active, you must specify at least one source.

The use of a negative form (no) of the command cancels the specified source analysis.

Syntax

```
[no] match-source <SOURCE>
```

Parameters

<SOURCE> – source, used for SYSLOG messages analysis:

- pp – master PP4X board;
- pp-other – slave PP4X board;
- slot – slot number, specified in the format: slot <num>, where:
- <num> – slot number [0..15].

Command mode

LOG FILTER

Example

```
ma4000(pp4x-config-log-filter-test)# match-source pp-other
```

facility

This command specifies the level of SYSLOG messages to be analyzed.

The use of a negative form (no) of the command cancels the specified analysis level.

Syntax

```
[no] facility <FACILITY> <SEVERITY>
```

Parameters

<FACILITY> – SYSLOG message source class:

- kernel – core;
- user – user level;
- any – any.

<SEVERITY> – syslog message level: 'emerg', 'alert', 'crit', 'error', 'warning', 'notice', 'info', 'debug'.

Command mode

LOG FILTER

Example

```
ma4000(pp4x-config-log-filter-test)# facility all debug
```

match

This command adds a regexp expression that serves as a filter. The line entering the filter must be checked by this expression.

The use of the **not-match** command configures the regexp filter for the opposite purpose – the incoming line should NOT be checked by this expression.

The use of a negative form (no) of the command removes the regexp filter.

Syntax

```
match <REGEXP>  
not-match <REGEXP>  
no match
```

Parameters

<REGEXP> – regular expression, string up to 63 characters.

Command mode

LOG FILTER

Example

```
ma4000(pp4x-config-log-filter-test)# no match
```

37 ARP configuration (ARP Inspection)

- [ip arp inspection enable](#)
- [ip arp inspection trusted](#)
- [ip arp inspection static-table](#)
- [show ip arp inspection](#)
- [show arp](#)
- [show slot ip arp inspection](#)
- [show slot ip arp table](#)

ip arp inspection enable

This command in the CONFIG mode enables global control of ARP (ARP Inspection feature) on the device. The use of a negative form (no) of the command disables the ARP Inspection feature.

Syntax

```
[no] ip arp inspection enable
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# ip arp inspection enable
```

ip arp inspection trusted

This command adds the configured interface to the list of 'trusted' when using ARP control. The ARP traffic of the 'trusted' interface is considered safe and is not monitored.

The use of the negative (no) form of the command removes interface from the 'trusted' list and makes it 'untrusted'.

Syntax

```
[no] ip arp inspection trusted
```

Parameters

The command contains no arguments.

Command mode

FRONT-PORT

PORT-CHANNEL

SLOT-CHANNEL
 SLOT-CHANNEL
 PLC-PON-PORT
 PLC-SLOT-CHANNEL
 PLC-FRONT-PORT

Example

```
ma4000(config-front-port-2/5)# ip arp inspection trusted
```

ip arp inspection static-table

This command adds new IP + MAC + VLAN compliance to static table for ARP inspection.

The use of the negative (no) form of the command removes an existing entry from the static Arp inspection table..

Syntax

```
ip arp inspection static-table ip <IP-ADDRESS> mac-address <MAC-ADDRESS> vlan <VID>
no ip arp inspection static-table ip <IP-ADDRESS>
```

Parameters

<IP-ADDRESS> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255]. For a command with the prefix (no), only this parameter is sufficient;

<MAC-ADDRESS> – subscriber terminal MAC address, defined as XX:XX:XX:XX:XX:XX, where each part takes values of [00-FF];

<VID> – VLAN ID, in which to make the assignment. May take values [1..4094, ignore]. The 'ignore' value corresponds to any VLAN.

Command mode

CONFIG
 FRONT-PORT
 PORT-CHANNEL
 SLOT-CHANNEL
 SLOT-CHANNEL
 PLC-PON-PORT
 PLC-SLOT-CHANNEL
 PLC-FRONT-PORT

Example

```
ma4000(config)# ip arp inspection static-table ip 192.168.1.2 mac-address 00:22:68:77:D0:94
vlan ignore
```

show ip arp inspection

ARP Inspection function configuration view command.

Syntax

```
show ip arp inspection
```

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```
ma4000# show ip arp inspection
Arp inspection settings:
Global mode: disabled
Per VLAN enable:
  None

Trusted ports:
front-port 1/0, front-port 1/1, front-port 1/2, front-port 1/3, front-port 1/4
front-port 1/5, front-port 2/0, front-port 2/1, front-port 2/2, front-port 2/3
front-port 2/4, front-port 2/5

Static mapping
~~~~~
IP          MAC          VID          IFACE
-----

```

show arp

Command to view the ARP table.

ip arp inspection should be enabled.

Syntax

```
show arp
```

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```
ma4000# show arp
```

show slot <SLOT> ip arp inspection

ARP Inspection function configuration view command for PLC8 line board.

Syntax

```
show slot <SLOT> ip arp inspection
```

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show slot 6 ip arp inspection
Arp inspection settings:
Global mode: disabled
Per VLAN enable: 0
  None

Trusted ports: 1
  plc-front-port 6/0

  Static mapping
  ~~~~~
IP      MAC      VID      IFACE
-----
```

show slot <SLOT> ip arp table

ARP table view command for PLC8 line board.

ip arp inspection should be enabled.

Syntax

```
show slot <SLOT> ip arp table
```

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
test1_3_2# show slot 6 ip arp table  
ARP table is empty
```

38 VLAN profile configuration (Profile VLAN)

- [profile vlan](#)
- [name](#)
- [description](#)
- [downlink extended](#)
- [downlink insertion](#)
- [uplink extended](#)
- [uplink insertion](#)
- [show profile vlan](#)

profile vlan

The command allows you to switch to the VLAN profile configuration mode.

The use of a negative form (no) of the command removes a specified profile.

Syntax

```
[no] profile vlan <NAME>
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile vlan test
ma4000(config-vlan)("test")#
```

name

The command changes the name of the configured profile.

The use of a negative form (no) of the command removes a name.

Syntax

```
[no] name <STRING>
```

Parameters

<STRING> – profile name.

Command mode

PROFILE VLAN

Example

```
ma4000(config-vlan)("test"># name EL
```

description

The command allows you to add a description for the configured VLAN profile.

The use of a negative form (no) of the command removes description.

Syntax

```
[no] description <STRING>
```

Parameters

<STRING> – text description.

Command mode

PROFILE VLAN

Example

```
ma4000(config-vlan)("test"># description TEST
```

downlink extended

This command sets an additional Ethertype in the downstream to recognize the service VLAN.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] downlink extended <TYPE> <VALUE>
```

Parameters

<TYPE> – Ethertype type:

- cvlan-type – ETHERTYPE value for inner VID;
- svlan-type – ETHERTYPE value for outer VID.

<VALUE> – ethertype value, may take values [0x0..0xFFFF].

Command mode

PROFILEVLAN

Example

```
ma4000(config-vlan)("test")# downlink extended svlan-type 0x8100
```

downlink insertion

This command sets the value of the Ethertype field inserted into the package for SVLAN or CVLAN in the downstream.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] downlink insertion <TYPE> <VALUE>
```

Parameters

<TYPE> – Ethertype type:

- cvlan-type – ETHERTYPE value for inner VID;
- svlan-type – ETHERTYPE value for outer VID.

<VALUE> – ethertype value, may take values [0x0..0xFFFF].

Command mode

PROFILEVLAN

Example

```
ma4000(config-vlan)("test")# downlink insertion svlan-type 0x8100
```

uplink extended

This command sets an additional Ethertype in the upstream to recognize the service VLAN. Regardless of the set value, the chip always recognizes Ethertype 0x9100 and 0x88A8.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] uplink extended <TYPE> <VALUE>
```

Parameters

<TYPE> – Ethertype type:

- cvlan-type – ETHERTYPE value for inner VID;
- svlan-type – ETHERTYPE value for outer VID.

<VALUE> – ethertype value, may take values [0x0..0xFFFF].

Command mode

PROFILEVLAN

Example

```
ma4000(config-vlan)("test")# uplink extended svlan-type 0x9100
```

uplink insertion

This command sets the value of the Ethertype field inserted into the package for SVLAN or CVLAN in the upstream.

Syntax

```
uplink insertion <TYPE><VALUE>
```

Parameters

<TYPE> – Ethertype type:

- cvlan-type – ETHERTYPE value for inner VID;
- svlan-type – ETHERTYPE value for outer VID.

<VALUE> – ethertype value, may take values [0x0..0xFFFF].

Command mode

PROFILEVLAN

Example

```
ma4000(config-vlan)("test")# uplink insertion svlan-type 0x9100
```

show profile vlan

This command is used to view the created VLAN profiles.

When you specify a profile name, you will see the settings of the configured VLAN profile:

Syntax

```
show profile vlan [<NAME>]
```

Parameters

<NAME> – profile name, optional parameter.

Command mode

ROOT

Example 1

```

ma4000# show PROFILE VLAN vlan-00
Description:                               'OLT Profile Vlan 0'
  Config general [uplink]:
    Extended svlan type:                   0x8100
    Extended cvlan type:                   0x8100
    Insertion svlan ethertype:             0x8100
    Insertion cvlan ethertype:             0x8100
  Config general [downlink]:
    Extended svlan type:                   0x8100
    Extended cvlan type:                   0x8100
    Insertion svlan ethertype:             0x8100
    Insertion cvlan ethertype:             0x8100

```

Example 2

```

ma4000# show profile vlan
  ##          Name      Description
  1          vlan-00    OLT Profile Vlan 0

```

39 DHCP relay agent profile configuration (PROFILE DHCP_RA)

- [profile dhcp-ra](#)
- [name](#)
- [description](#)
- [enable](#)
- [dos-block](#)
- [overwrite-option82](#)
- [trusted \[primary|secondary\]](#)
- [trusted timeout](#)
- [trusted server](#)
- [show profile dhcp-ra](#)
- [show interface gpon-port dhcp sessions](#)

In the **PROFILE DHCP_RA**, you can configure the DHCP relay agent profile parameters. After configuration, the DHCP relay agent profile can be assigned to a specific PLC8 module with the **slot <id> profile dhcp-ra** command in the CONFIG command mode.

profile dhcp-ra

This command allows you to create a new DHCP relay agent profile and switch to its configuration mode.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] profile dhcp-ra [<NAME>]
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile dhcp-ra test
ma4000(config-dhcp-ra)("test")#
```

name

The command changes the name of the configured profile.

The use of a negative form (no) of the command removes a name.

Syntax

```
[no] name <STRING>
```

Parameters

<STRING> – profile name.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("test"># name EL
```

description

The command allows you to add a description for the configured DHCP relay agent profile.

The use of a negative form (no) of the command removes description.

Syntax

[no] description <STRING>

Parameters

<STRING> – text description.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("test"># description TEST
```

enable

This command enables DHCP relay (Option 82).

The use of a negative form (no) of the command disables this feature.

Syntax

[no] enable

Parameters

The command contains no arguments.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("TEST")# enable
```

dos-block

This command enables DoS attack prevention.

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] set dos-block [packet-limit <LIMIT>|block-time <TIME>]
```

Parameters

packet-limit <LIMIT> – the command sets the DoS attack threshold (number of packets per second), where <LIMIT> – number of packets per second, may take values [10 .. 1000];

block-time <TIME> – the command sets the time to block the port when a DoS attack is detected, where <TIME> – blocking time, may take values [30 .. 3600] in seconds.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("TEST")# dos-block packet-limit 13 block-time 222
```

overwrite-option82

This command sets the formats of CircuitID and RemouteID fields for Option82.

Circuit ID – contains information about the port from which the request came to the DHCP repeater;

Remote ID – identifier of the DHCP repeater itself.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] overwrite-option82 <FORMAT> <STRING> <TEXT FORMAT>
```

Parameters

<FORMAT> – configured field:

- circuit_id_format;
- remote_id_format

<STRING> – string up to 240 characters. Has the following form: PARAM_name1=PARAM_1... PARAM_name2=PARAM_2... PARAM_nameN= PARAM_N, where parameters (PARAM_1.. PARAM_N) are the following formats:

%HOSTNAME% – LTP device name;

%SLOTID% – MA4000 slot number;

%MNGIP% – management interface IP address;

%GPON-PORT% – optical channel identifier;

%ONTID% – ONT identifier, assigned by administrator;

%PONSERIAL% – ONT device serial number;

%GEMID% – GEM port identifier;

%VLAN0% – VLAN external identifier;

%VLAN1% – VLAN internal identifier;

%MAC% – subscriber device MAC address;

%OPT60% – string is extracted from DHCP option 60 of the incoming packet;

%OPT82_CID% – DHCP option82 Circuit ID string is extracted from the incoming packet;

%OPT82_RID% – DHCP option82 Remote ID string is extracted from the incoming packet;

%DESCR% – first 20 characters of ONT configuration description.

The separators between parameters may be any characters, but each parameter FORMAT must be enclosed in '% '.

<TEXT FORMAT> – option presentation type:

- binary – binary;
- text – text.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("test")# circuit_id_format host=%HOSTNAME%,ont=%ONTID%, slot=%SLOTID%'
```

trusted [primary|secondary]

This command sets the IP addresses of the primary and secondary DHCP server.

The use of a negative form (no) of the command removes a specified value.

Syntax

[no] trusted primary <IP>

[no] trusted secondary <IP>

Parameters

trusted primary <IP> – the command sets primary DHCP server IP address;

trusted secondary <IP> – the command sets secondary DHCP server IP address;

<IP> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("test")# trusted primary 192.168.52.2
```

trusted timeout

This command sets the time of waiting for the response from the DHCP server.

The use of a negative form (no) of the command removes a specified value.

Syntax

[no] trusted timeout <VALUE>

Parameters

<VALUE> – time period in seconds [200..1500].

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("test")# trusted timeout 1000
```

trusted server

This command enables the use of trusted DHCP servers.

The use of a negative form (no) of the command disables this feature.

Syntax

[no] trusted server

Parameters

The command contains no arguments.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("TEST")# trusted server
```

show profile dhcp-ra

This command is used to view the created DHCPRA profiles.

When you specify a profile name, you will see the settings of the configured profile:

- Description – profile description;
- Relay agent – DHCP relay state (enabled/disabled);
- Circuit id format – Circuit id field mode for Option82 (text or binary);
- Remote id format – Remote id field mode for Option82 (text or binary);
- Overwrite client option82 – allow/deny the adding of information of Option 82, that was received from the client;
- Dos block enabled – DoS attack protection (enabled/disabled);
- Bc packet per second – number of packets per second, at which the system registers DoS attack;
- Port block time – port blocking time on detection of DOS attack, seconds;
- Trusted server enable – allow/deny the use of trusted DHCP servers;
- Trusted primary – primary DHCP server address;
- Trusted secondary – secondary DHCP server address;
- Trusted server timeout – response time from DHCP server, seconds.

Syntax

```
show profile dhcp-ra [<NAME>]
```

Parameters

<NAME> – profile name, optional parameter.

Command mode

ROOT

Example 1

```
ma4000# show profile dhcp-ra dhcp-ra-00
Description:                               'OLT Profile DHCP Relay Agent 0'
Relay agent:                             disabled
Circuit id format:                        ''
Remote id format:                         ''
Overwrite client option82:               false
Dos block enabled:                        false
Bc packet per second:                     128
Port block time:                          600
Trusted server:                           disabled
Trusted primary:                          0.0.0.0
Trusted secondary:                        0.0.0.0
Trusted server timeout:                   1000
```

Example 2

```
ma4000# show profile dhcp-ra
##          Name      Description
  1         dhcp-ra-00 OLT Profile DHCP Relay Agent 0
```

show interface gpon-port <GPON-PORT> dhcp sessions

This command allows you to view currently active DHCP-sessions on PLC8 line cards.

DHCP Relay Agent should be enabled.

Syntax

```
show interface gpon-port <GPON-PORT> pppoe sessions
```

Parameters

<GPON-PORT> - gpon-port number in format of <SLOT>/<port>

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-);

<port> – GPON port number of PLC8 module.

Command mode

ROOT

Example

```
ma4000# show interface gpon-port 1/0 dhcp sessions
No active DHCP sessions
```

40 DHCP IPv6 relay agent profile configuration (PROFILE DHCPv6_RA)

- [profile dhcpv6-ra](#)
- [name](#)
- [description](#)
- [enable](#)
- [dos-block](#)
- [add-interface-id](#)
- [add-remote-id](#)
- [add-suboptions](#)
- [trusted \[primary|secondary\]](#)
- [trusted timeout](#)
- [trusted server](#)
- [show profile dhcpv6-ra](#)

In the **PROFILE DHCPv6_RA**, you can configure the DHCP relay agent profile parameters. After configuration, the DHCPv6 relay agent profile can be assigned to a specific PLC8 module with the **slot <id> profile dhcpv6-ra** command in the CONFIG command mode.

profile dhcpv6-ra

This command allows you to create a new DHCP relay agent profile and switch to its configuration mode. The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] profile dhcpv6-ra [<NAME>]
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile dhcpv6-ra test
ma4000(config-dhcpv6-ra)("test")#
```

name

The command changes the name of the configured profile. The use of a negative form (no) of the command removes a name.

Syntax

```
[no] name <STRING>
```

Parameters

<STRING> – profile name.

Command mode

PROFILE DHCPV6_RA

Example

```
ma4000(config-dhcp-ra)("test"># name EL
```

description

The command allows you to add a description for the configured DHCPv6 relay agent profile.

The use of a negative form (no) of the command removes description.

Syntax

[no] description <STRING>

Parameters

<STRING> – text description.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("test"># description TEST
```

enable

This command enables DHCPv6 Relay (Option82).

The use of a negative form (no) of the command disables this feature.

Syntax

[no] enable

Parameters

The command contains no arguments.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcpv6-ra)("TEST")# enable
```

dos-block

This command enables DoS attack prevention.

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] set dos-block [packet-limit <LIMIT>|block-time <TIME>]
```

Parameters

packet-limit <LIMIT> – the command sets the DoS attack threshold (number of packets per second), where <LIMIT> – number of packets per second, may take values [10 .. 1000];

block-time <TIME> – the command sets the time to block the port when a DoS attack is detected, where <TIME> – blocking time, may take values [30 .. 3600] in seconds.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcp-ra)("TEST")# dos-block packet-limit 13 block-time 222
```

add-interface-id

This command sets the formats of Option 38 in DHCPv6.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] add-interface-id <STRING> <TEXT FORMAT>
```

Parameters

<STRING> – string up to 240 characters. Has the following form: PARAM_name1=PARAM_1... PARAM_name2=PARAM_2... PARAM_nameN= PARAM_N, where parameters (PARAM_1.. PARAM_N) are the following formats:

%HOSTNAME% – LTP device name;

%SLOTID% – MA4000 slot number;
 %MNGIP% – management interface IP address;
 %GPON-PORT% – optical channel identifier;
 %ONTID% – ONT identifier, assigned by administrator;
 %PONSERIAL% – ONT device serial number;
 %GEMID% – GEM port identifier;
 %VLAN0% – VLAN external identifier;
 %VLAN1% – VLAN internal identifier;
 %MAC% – subscriber device MAC address;
 %DESCR% – first 20 characters of ONT configuration description.

The separators between parameters may be any characters, but each parameter FORMAT must be enclosed in '% '.

<TEXT FORMAT> – option presentation type:

- binary – binary;
- text – text.

Command mode

PROFILE DHCPV6_RA

Example

```
ma4000(config-dhcpv6-ra)("test")# add-interface-id host=%HOSTNAME%,ont=%ONTID%,slot=%SLOTID%
```

add-remote-id

This command sets the formats of Option 37 in DHCPv6.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] add-remote-id <STRING> <TEXT FORMAT>
```

Parameters

<STRING> – string up to 240 characters. Has the following form: PARAM_name1=PARAM_1... PARAM_name2=PARAM_2... PARAM_nameN= PARAM_N, where parameters (PARAM_1.. PARAM_N) are the following formats:

%HOSTNAME% – LTP device name;
 %SLOTID% – MA4000 slot number;
 %MNGIP% – management interface IP address;
 %GPON-PORT% – optical channel identifier;
 %ONTID% – ONT identifier, assigned by administrator;

%PONSERIAL% – ONT device serial number;

%GEMID% – GEM port identifier;

%VLAN0% – VLAN external identifier;

%VLAN1% – VLAN internal identifier;

%MAC% – subscriber device MAC address;

%DESCR% – first 20 characters of ONT configuration description.

The separators between parameters may be any characters, but each parameter FORMAT must be enclosed in '%'.

<TEXT FORMAT> – option presentation type:

- binary – binary;
- text – text.

Command mode

PROFILE DHCPV6_RA

Example

```
ma4000(config-dhcpv6-ra)("test")# add-remote-id host=%HOSTNAME%,ont=%ONTID%,slot=%SLOTID%
```

add-suboptions

This command enables adding a suboption to DHCPv6.

The use of a negative form (no) of the command disables suboptions adding.

Syntax

[no] add-suboptions

Parameters

The command contains no arguments.

Command mode

PROFILE DHCPV6_RA

Example

```
ma4000(config-dhcpv6-ra)("test")# add-suboptions
```

trusted [primary|secondary]

This command sets the IP addresses of the primary and secondary DHCP server.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] trusted primary <IP>
[no] trusted secondary <IP>
```

Parameters

trusted primary <IP> – the command sets primary DHCP server IP address;

trusted secondary <IP> – the command sets secondary DHCP server IP address;

<IP> – IP address, defined as AAA.BBB.CCC.DDD where each part takes values of [0..255].

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcpv6-ra)("test"># trusted primary 192.168.52.2
```

trusted timeout

This command sets the time of waiting for the response from the DHCPv6 server.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
[no] trusted timeout <VALUE>
```

Parameters

<VALUE> – time period in seconds [200..1500].

Command mode

PROFILE DHCPV6_RA

Example

```
ma4000(config-dhcpv6-ra)("test"># trusted timeout 1000
```

trusted server

This command enables the use of trusted DHCPv6 servers.

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] trusted server
```

Parameters

The command contains no arguments.

Command mode

PROFILE DHCP_RA

Example

```
ma4000(config-dhcpv6-ra)("TEST")# trusted serve
```

show profile dhcpv6-ra

This command is used to view the created DHCPv6_RA profiles.

When you specify a profile name, you will see the settings of the configured profile:

- Description – profile description;
- Relay agent – DHCPv6 relay state (enabled/disabled);
- Interface id format – Option 38 format;
- Remote id format – Option 37 format;
- Write interface id option, Write remote id option – allow/deny adding suboptions;
- Dos block enabled – DoS attack protection (enabled/disabled);
- Bc packet per second – number of packets per second, at which the system registers DoS attack;
- Port block time – port blocking time on detection of DOS attack, seconds;
- Trusted server enable – allow/deny the use of trusted DHCP servers;
- Trusted primary – primary DHCP server address;
- Trusted secondary – secondary DHCP server address;
- Trusted server timeout – response time from DHCP server, seconds.

Syntax

```
show profile dhcpv6-ra [<NAME>]
```

Parameters

<NAME> – profile name, optional parameter.

Command mode

ROOT

Example 1

```
ma4000# show profile dhcpv6-ra test
Description:                'OLT Profile DHCPv6 Relay Agent 1'
Relay agent:                disabled
Interface id format:       ''
Remote id format:          '%DESCR%'
Write interface id option:  true
Write remote id option:    true
Dos block enabled:         false
Bc packet per second:      128
Port block time:           600
Trusted server:            disabled
Trusted primary:           ::
Trusted secondary:         ::
Trusted server timeout:    1000
```

Example 2

```
ma4000# show profile dhcpv6-ra
##          Name      Description
 1         dhcpv6-ra-00  OLT Profile DHCP Relay Agent 0
```

41 Intermediate PPPoE agent profile configuration (PROFILEPPPOE_IA)

- [profile pppoe-ia](#)
- [name](#)
- [description](#)
- [dos-block](#)
- [enable](#)
- [format](#)
- [sessions-limit](#)
- [vendor-id](#)
- [show interface gpon-port pppoe sessions](#)
- [show profile pppoe-ia](#)

In the **PROFILEPPPoE_IA** mode, you can configure the profile parameters of the intermediate agent PPPoE of PLC8 module. After performing the settings, the PPPoE agent profile can be assigned to a certain PLC8 module with the command 'set general profile_olt_pppoeia' in the PROFILESPCOLT command mode.

profile pppoe-ia

This command allows to create a PPPoE profile of Intermedia Agent and switch to its configuration mode. The use of a negative form (no) of the command removes the profile.

Syntax

```
profile pppoe-ia [<NAME>]
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile pppoe-ia TEST
ma4000(config-pppoe-ia)("TEST")#
```

name

The command changes the name of the configured profile.

Syntax

```
name <STRING>
```

Parameters

<STRING> – profile name, case sensitive.

Command mode

PROFILE PPPOE_IA

Example

```
ma4000(config-pppoe-ia)("test"># name EL
```

description

The command allows you to add a description for the configured PPPoE Intermediate Agent profile.
The use of a negative form (no) of the command removes description.

Syntax

```
description <STRING>
```

Parameters

<STRING> – text description.

Command mode

PROFILE PPPOE_IA

Example

```
ma4000(config-pppoe-ia)("TEST"# description TEST
```

dos-block

This command enables DoS attack prevention.
The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] set dos-block [packet-limit <LIMIT>|block-time <TIME>]
```

Parameters

packet-limit <LIMIT> – the command sets the DoS attack threshold (number of packets per second), where
<LIMIT> – number of packets per second, may take values [10 .. 1000];

block-time <TIME> – the command sets the time to block the port when a DoS attack is detected, where
<TIME> – blocking time, may take values [30 .. 3600] in seconds.

Command mode

PROFILE PPPOE_IA

Example

```
ma4000(config-pppoe-ia)("TEST")# dos-block packet-limit 13 block-time 222
```

enable

This command enables PPPoE Intermediate Agent (PPPoE+).

The use of a negative form (no) of the command disables this feature.

Syntax

```
[no] enable
```

Parameters

The command contains no arguments.

Command mode

PROFILE PPPOE_IA

Example

```
ma4000(config-pppoe-ia)("TEST")# enable
```

format

This command sets the formats of CircuitId and RemouteId fields for Vendor Specific Tag.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
format <FORMAT> <STRING>
```

Parameters

<FORMAT> – configured field:

- circuit_id;
- remote_id.

<STRING> – string up to 240 characters. Has the following form: PARAM_name1=PARAM_1... PARAM_name2=PARAM_2... PARAM_nameN= PARAM_N, where parameters (PARAM_1.. PARAM_N) are the following formats:

%HOSTNAME% – LTP device name;

%SLOTID% – MA4000 slot number;

%MNGIP% – management interface IP address;

%GPON-PORT% – optical channel identifier;

%ONTID% – ONT identifier, assigned by administrator;

%PONSERIAL% – ONT device serial number;

%GEMID% – GEM port identifier;

%VLAN0% – VLAN external identifier;

%VLAN1% – VLAN internal identifier;

%MAC% – subscriber device MAC address.

%DESCR% – first 20 characters of ONT configuration description.

The separators between parameters may be any characters, but each parameter FORMAT must be enclosed in '% '.

Command mode

PROFILE PPPOE_IA

Example

```
ma4000(config-pppoe-ia)("test")# format circuit-id host=%HOSTNAME%,ont=%ONTID%, slot=%SLOTID%
```

sessions-limit

This command sets the maximum number of PPPoE sessions for a PPPoE Intermediate Agent and the maximum number of PPPoE sessions for a single ONT.

The use of a negative form (no) of the command removes a specified value.

Syntax

```
sessions_limit <LIMIT> [per-user <LIMIT_ONT>]
```

Parameters

<LIMIT> – number of PPPoE sessions for PPPoE Intermediate Agent, may take values [0 .. 8192].

per-user<LIMIT_ONT> – the command sets maximum amount of PPPoE sessions for a single ONT, where

<LIMIT_ONT> – number of PPPoE sessions for a single ONT.

Command mode

PROFILE PPPOE_IA

Example

```
ma4000(config-pppoe-ia)("test"># sessions-limit 424 per-user 3
```

vendor-id

This command sets Vendor Id.

The use of a negative form (no) of the command removes a specified value.

Syntax

vendor-id <VALUE>

Parameters

<VALUE> – Vendor-Id, may take values [0x000000..0xffffffff].

Command mode

PROFILEPPPOE_IA

Example

```
ma4000(config-pppoe-ia)("test"># vendor-id 0x000de9
```

show interface gpon-port <GPON-PORT> pppoe sessions

This command allows you to view currently active PPP-sessions on PLC8 line cards.

PPPoE Intermediate Agent should be enabled.

Syntax

show interface gpon-port <GPON-PORT> pppoe sessions

Parameters

<GPON-PORT> - gpon-port number in format of <SLOT>/<port>;

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-);

<port> – GPON port number of PLC8 module.

Command mode

ROOT

Example

```
ma4000# show interface gpon-port 1/0 pppoe sessions
No active PPPoE sessions
```

show profile pppoe-ia

This command allows to view the created PPPoE Intermediate Agent profiles.

When you specify a profile name, you will see the settings of the configured profile:

- Description – profile description;
- Intermediate agent – PPPoE IA state (enabled/disabled);
- Circuit id format – Circuit id field format for VendorSpecificTag;
- Remote id format – Remote id field format for VendorSpecificTag;
- Vendor id – ID number;
- Max number pppoe sessions – maximum number of PPPoE sessions for PPPoE Intermediate Agent;
- Max number pppoe sessions per user – maximum number of PPPoE sessions for ONT;
- Dos block enabled – DoS attack protection (enabled/disabled);
- Bc packet per second – number of packets per second, at which the system registers DoS attack;
- Port block time – port blocking time on detection of DOS attack, seconds.

Syntax

```
show profile pppoe-ia [<NAME>]
```

Parameters

<NAME> – profile name, optional parameter.

Command mode

ROOT

Example 1

```
ma4000#show profile pppoe-ia pppoe-ia-00
Description:                               'OLT Profile PPPoE Intermediate Agent 0'
Intermediate agent:                        disabled
Circuit id format:                         ''
Remote id format:                         ''
Vendor id:                                0x000DE9
Max number pppoe sessions:                 0
Max number pppoe sessions per user:        0
Dos block enabled:                         false
Bc packet per second:                      128
Port block time:                           600
```

Example 2

```
ma4000# show profile pppoe-ia
##          Name      Description
 1         pppoe-ia-00 OLT Profile PPPoE Intermediate Agent 0
```

42 Address table profile configuration (PROFILE ADDRESS TABLE)

- [profile address-table](#)
- [name](#)
- [description](#)
- [show profile address-table](#)
- [discard_pid_unlearned_sa](#)
- [remove_when_aged](#)
- [svlan](#)

In the **PROFILE ADDRESS TABLE** mode, the parameters for the address table profile are adjusted. After making the settings, the profile address-table can be assigned to a specific GPON interface of the PLC8 module with the 'profile address-table <NAME>' command in the GPON-PORT command mode.

profile address-table

This command allows you to create a new ADDRESS-TABLE profile and switch to its configuration mode.

Syntax

```
profile address-table <NAME>
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

PROFILE ADDRESS TABLE

Example

```
ma4000(config)# profile address-table TEST
ma4000(config-address-table)("TEST">#
```

name

The command changes the name of the configured profile of the address table.

Syntax

```
name <STRING>
```

Parameters

<STRING> – profile name.

Command mode

PROFILE ADDRESS TABLE

Example

```
ma4000(config-address-table)("TEST")# name EL
```

description

The command allows you to add a description for the configured profile of the address table.

The use of a negative form (no) of the command removes a profile description.

Syntax

```
[no] description <STRING>
```

Parameters

<STRING> – text description.

Command mode

PROFILE ADDRESS TABLE

Example

```
ma4000(config-address-table)("TEST")# description address_table_Test
```

show profile address-table

This command is used to view the created profiles of the address table.

When you specify a profile name, you will see the settings of the configured address table profile:

- Description – description
- Remove when aged – state of the obsolete entries removal function (enabled/disabled);
- Discard pid unlearned sa – frame redirection state from an unknown source address (enabled/disabled).

Syntax

```
show profile address-table [<NAME>]
```

Parameters

<NAME> – profile name, optional parameter.

Command mode

ROOT

Example

```
ma4000# show profile address-table addresstable-00
  Description:                               'OLT Profile Address Table 0'
  Table config:
    Remove when aged:                        true
    Discard pid unlearned sa:                true
```

discard_pid_unlearned_sa

Enable redirection of all frames from an unknown source address when the entry limit value for a certain interface is reached.

The use of the negative form of the command (no) disables the function.

Syntax

```
[no] discard_pid_unlearned_sa
```

Parameters

The command contains no arguments.

Command mode

PROFILE ADDRESS TABLE

Example

```
ma4000(config-address-table)("TEST")# discard-pid-unlearned-sa
```

remove_when_aged

This command configures a procedure for deleting entries that have reached the end of their lifetime.

The use of the negative form of the command (no) disables the function.

Syntax

```
[no] remove_when_aged
```

Parameters

The command contains no arguments.

Command mode

PROFILE ADDRESS TABLE

Example

```
ma4000(config-address-table)("TEST"># remove-when-aged
```

svlan

This command creates a new configuration for the service VLAN (SVLAN).

The use of the negative form of the command (no) removes the service VLAN configuration.

Syntax

```
[no] svlan <VID> [use [c-vlan|s-vlan|pbits]] forwarding-mode <MODE>| discard-unknown]
```

Parameters

<VID> – VLAN ID, may take values [1..4094];

usec-vlan – when this command is specified, the use of the service VLAN is enabled as part of the address table key;

uses-vlan – when this command is specified, the use of the service VLAN is enabled as part of the address table key;

use pbits – when this command is specified, the use of the priority bit is enabled as part of the address table key;

forwarding-mode <MODE> – this command sets the redirection mode, where **<MODE>** – redirection mode: N_to_1, 1_to_1;

discard-unknown – this command enables the frame rejection with the source address, which is not found in the address table.

Command mode

PROFILE ADDRESS TABLE

Example

```
ma4000(config-address-table)("TEST"># s-vlan 1 use c-vlan forwarding-mode 1-to-1 discard-unknown
```

43 PLC8 board ONT configuration profile management. cross-connect profile

- [cross-connect](#)
- [name](#)
- [description](#)
- [tag-mode](#)
- [outer vid](#)
- [outer cos](#)
- [inner vid](#)
- [user vid](#)
- [user cos](#)
- [type](#)
- [bridge](#)
- [bridge group](#)
- [iphost eid](#)
- [priority](#)
- [mac-table-limit](#)
- [show profile cross-connect](#)

In the **PROFILE cross-connect** mode, the VLAN parameters of the GEM-port operations are set to transmit traffic from/to ONT.

After configuration, the **cross-connect** profile can be assigned to a specific template or directly in the ONT configuration.

cross-connect

This command allows to create a new profile and/or switch to its configuration mode.

The use of a negative form (no) of the command removes the profile.

Syntax

```
profile cross-connect [<NAME>]
[no] profile cross-connect [<NAME>]
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile cross-connect name
ma4000(config-cross-connect) ("name")#
```

name

The command changes the name of the configured profile.

Syntax

name <STRING>

Parameters

<STRING> – profile name.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name"># name test
```

description

This command allows to add a description for the profile to be configured.

The use of a negative form (no) of the command removes description.

Syntax

description <STRING>

[no] description <STRING>

Parameters

<STRING> – text description.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name"># description doc
```

tag-mode

The command allows to set the service operation mode: transfer once tagged traffic or qinq traffic.

The use of a negative form (no) of the command removes description.

Syntax

tag-mode<MODE>

[no] tag-mode

Parameters

<MODE> – transmission mode, may take values:

- single-tagged – to transmit dot1.q packets;
- double-tagged – to transmit Q-in-Q packets;
- tunnel – for tunneling;
- selective-tunnel – for the selective tunneling.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name"># tag-mode single-tagged
```

outer vid

This command allows to set the VLANID value for the external upstream packages tag.

The use of a negative form (no) of the command sets the default value.

Syntax

```
outer vid <VID>
[no] outer vid
```

Parameters

<VID> – VLAN number, may take values:

- terminal-vlan <NAME> – use VLANID value, that corresponds to terminal-vlan <NAME> of the slot (see section SLOT TERMINAL VLAN);
- 1 to 4094.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name"># outer vid 2149
```

outer cos

The command allows to set the COS value for the external packet tag in an upstream direction.

The use of a negative form (no) of the command sets the default value.

Syntax

```
outer cos <COS>
[no] outer cos
```

Parameters

<COS> – 802.1p tag value, may take values:

- terminal-vlan – use CoS value, that corresponds to terminal-vlan <NAME> of the slot (see section SLOT TERMINAL VLAN);
- 0-7;
- unused – When cos unused is set, the value of the 'priority' field of the received packets from the 'user' side will be copied for upstream packets.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name")# outer cos 1
```

inner vid

This command allows to set the VLANID value for the inner upstream QinQ packages tag.

The CoS value of the internal tag is copied from the external one in this case.

The use of a negative form (no) of the command sets the default value.

Syntax

```
inner vid <VID>
[no] inner vid
```

Parameters

<VID> – VLAN number, may take values:

- terminal-vlan <NAME> – use VLANID value, that corresponds to terminal-vlan <NAME> of the slot (see section SLOT TERMINAL VLAN);
- 1 to 4094.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name")# inner vid 2149
```

user vid

The command allows to set VLANID value for outer tag of downstream packets transmitted to uni part of ONT (to the ONT VeIP or ETHERNET port).

The use of a negative form (no) of the command sets the default value.

Syntax

```
user vid <VID>
[no] user vid
```

Parameters

<VID> – VLAN number, may take values:

- terminal-vlan <NAME> – use VLANID value, that corresponds to terminal-vlan <NAME> of the slot (see section SLOT TERMINAL VLAN);
- 1-4094;
- untagged.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name")# user vid 10
```

user cos

The command allows to set COS value for outer tag of downstream packets transmitted to uni part of ONT (to the ONT VeIP or ETHERNET port).

The use of a negative form (no) of the command sets the default value.

Syntax

```
user cos <COS>
[no] user cos
```

Parameters

<COS> – 802.1p tag value, may take values:

- terminal-vlan – use CoS value, that corresponds to terminal-vlan <NAME> of the slot (see section SLOT TERMINAL VLAN);
- 0-7;
- unused – when **cos unused** is set, the value of the 'priority' field of the received packets from the network side will be copied for downstream packets.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name")# user cos 1
```

type

This command allows to set the interface type to connect the service provided through this service. The use of a negative form (no) of the command sets the general operation mode.

Syntax

```
type <TYPE>
[no] type
```

Parameters

<TYPE> – interface type for service activation:

- general – GEM-port for traffic transmission, connected to the VeIP-interface of ONT, serves for bidirectional traffic transmission in accordance with TR-142;
- iphost – GEM port for traffic transmission, connected to the ONT IP interface;
- management – GEM port for traffic transmission, connected to the IP interface used for ONT management via TR-069;
- multicast – GEM port for IGMP traffic transmission;
- voice – GEM port for traffic transmission, connected to the IP interface used for VoIP traffic transmission.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name")# type management
```

bridge

This command allows to set the connection type for the service provided through this service. The *routed* (routed traffic through an ONT router) or *bridged* (connected via a bridge circuit) service. The use of a negative form (no) of the command sets the routed operation mode.

Syntax

```
bridge
[no] bridge
```

Parameters

The command contains no arguments.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name"># bridge
```

bridge group

This command allows to set the connection type for the service provided through this service.

The *routed* (routed traffic through an ONT router) or *bridged* (connected via a bridge circuit) service.

The use of a negative form (no) of the command sets the default value.

Syntax

```
bridge group <group ID>
[no] bridge group
```

Parameters

<groupID> – number of bridge froup on ONT <1-255>.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name"># bridge group 2
```

iphost eid

This command allows to set the IP interface instance number on the ONT, through which the service traffic will be transmitted.

For correct configuration, the service must be of bridge type: management or voip.

The use of a negative form (no) of the command sets the default value.

Syntax

```
iphost eid <ID>
[no] iphost
```

Parameters

<ID> – ONT IP interface number, may take values [0..65535];

- value 0 – usually used to connect the control interface via tr-069,
- value 1 – for VoIP services.

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name")# iphost eid 1
```

priority

The command allows to set the priority queue value for traffic transmission (there must be support for priority queue processing on the ONT side) for the case of transferring traffic of different GEM ports within a single T-CONT.

The use of a negative form (no) of the command sets the default value.

Syntax

```
priority <PRIORITY>
[no] priority
```

Parameters

<PRIORITY> – priority value, may take values [0 .. 7].

Command mode

PROFILECROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name")# priority 7
```

mac-table-limit

This command allows to set a limit on the MAC address table size for the GEM port.

The use of a negative form (no) of the command sets the default value.

Syntax

```
mac-table-limit <COUNT>
[no] mac-table-limit
```

Parameters

<COUNT> – number of entries in the MAC table, may take values [1..126, unlimited].

Command mode

PROFILE CROSS-CONNECT

Example

```
ma4000(config-cross-connect)("name"># mac-table-limit 30
```

show profile cross-connect

This command allows to view the existing profiles, view the configuration of the specified profile.

Syntax

```
show profile cross-connect <NAME>
```

Parameters

<NAME> – profile name, optional parameter.

If no profile name is specified – the list of all profiles of this type is displayed, if the name is specified – the configuration of this profile is displayed.

Command mode

ROOT

Example 1

```
ma4000# show profile cross-connect
##          Name      Description
 1  crossconnect-00  ONT Profile Cross Connect 0
 2      ppp-v1314    ONT Profile Cross Connect 1
 3      ppp-v1315    ONT Profile Cross Connect 2
 4      ppp-v1316    ONT Profile Cross Connect 3
 5      ppp-v1317    ONT Profile Cross Connect 4
```

Example 2

```
ma4000# show profile cross-connect ppp-v1314
  Name: 'ppp-v1314'
  Description: 'ONT Profile Cross Connect 1'
  Model: ont-rg
  Bridge group: -
  Tag mode: single-tagged
  Outer vid: 1314
  Outer cos: unused
  Inner vid: -
  U vid: 10
  U cos: unused
  Mac table entry limit: unlimited
  Type: general
  Iphost eid: 0
  Priority queue: 0
```

44 PLC8 board ONT configuration profile management. DBA profile

- [DBA](#)
- [name](#)
- [description](#)
- [alloc size](#)
- [alloc period](#)
- [bandwidth](#)
- [sla class](#)
- [sla status-reporting](#)
- [show profile dba](#)

In the **PROFILE DBA** mode, the DBA parameters are set for the GEM port transmitting traffic from ONT.

DBA profile configuration and T-CONT types

	T-CONT type 1	T-CONT type 2	T-CONT type 3	T-CONT type 4	T-CONT type 5
service-class	cbr	voip	type5	type5	type5
status-reporting	-	+	+	+	+
fixed-bandwidth	+	-	-	-	+
guaranteed-bandwidth	-	+	+	-	+
besteffort-bandwidth	-	-	+	+	+

After configuration, the **dba** profile can be assigned to a specific template or directly in the ONT configuration.

DBA

This command allows to create a new profile and/or switch to its configuration mode.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] profile dba <NAME>
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile dba name  
ma4000(config-dba)("name")#
```

name

The command changes the name of the configured profile.

Syntax

name <STRING>

Parameters

<STRING> – profile name.

Command mode

PROFILE DBA

Example

```
ma4000(config-dba)("name")# name test
```

description

This command allows to add a description for the profile to be configured.

The use of a negative form (no) of the command removes description.

Syntax

description <STRING>
[no] description <STRING>

Parameters

<STRING> – text description.

Command mode

PROFILE DBA

Example

```
ma4000(config-dba)("name")# description doc
```

alloc size

The command allows to set the buffer size to the transfer for a given location.

The use of a negative form (no) of the command sets the default value.

Syntax

```
alloc size <SIZE>  
[no] alloc size
```

Parameters

<SIZE> – buffer size, may take values [0..194400].

Command mode

PROFILE DBA

Example

```
ma4000(config-dba)("name")# alloc size 12000
```

alloc period

The command allows to set the period of window provision for traffic transfer.

The use of a negative form (no) of the command sets the default value.

Syntax

```
alloc period <PERIOD>  
[no] alloc period
```

Parameters

<PERIOD> – window provision interval, may take values [0, 1, 2, 4, 8, 16, 32].

Command mode

PROFILE DBA

Example

```
ma4000(config-dba)("name")# alloc period 32
```

bandwidth

The command allows to set the bandwidth parameters for a current service (guaranteed, fixed and maximum possible bandwidth).

The use of a negative form (no) of the command sets the default value.

Syntax

```
bandwidth <TYPE> <SIZE>
[no] description <TYPE>
```

Parameters

<TYPE> – bandwidth type:

- besteffort – maximum permissible;
- fixed – fixed;
- guaranteed – guaranteed;

<SIZE> – allocated bandwidth, may take values [0..1244000] kbps.

Command mode

PROFILE DBA

Example

```
ma4000(config-dba)("name")# bandwidth guaranteed 300000
```

sla class

This command allows to set the type of container.

The use of a negative form (no) of the command sets the default value.

Syntax

```
sla class <TYPE>
[no] sla class
```

Parameters

<TYPE> – container type, may take values: cbr; data; periodic-allocation; type5; voip.

Command mode

PROFILE DBA

Example

```
ma4000(config-dba)("name")# sla class voip
```

sla status-reporting

The command allows to set the type of message about the state of the queue for transmission.

The use of a negative form (no) of the command sets the default value.

Syntax

```
sla status-reporting <TYPE>
[no] sla status-reporting
```

Parameters

<TYPE> – message type:

- nsr – without status messages;
- type0 – with message forming, type 0;
- type1 – with message forming, type 1.

Command mode

PROFILE DBA

Example

```
ma4000(config-dba)("name")# sla status-reporting nsr
```

show profile dba

This command allows to view the existing profiles, view the configuration of the specified profile.

Syntax

```
show profile dba <NAME>
```

Parameters

<NAME> – profile name, optional parameter.

If no profile name is specified – the list of all profiles of this type is displayed, if the name is specified – the configuration of this profile is displayed.

Command mode

ROOT

Example 1

```

ma4000# show profile dba
##          Name      Description
  1         dba-00     ONT Profile DBA 0
  2          name     ONT Profile DBA 1

```

Example 2

```

ma4000# # show profile dba dba-00
Name:                                     'dba-00'
Description:                             'ONT Profile DBA 0'
Dba:
  Sla data:
    Service class:                        type5
    Status reporting:                     nsr
    Alloc size:                           0
    Alloc period:                         0
    Fixed bandwidth:                      0
    Guaranteed bandwidth:                 64
    Besteffort bandwidth:                 1244000

```

45 PLC8 board ONT configuration profile management. MANAGEMENT profile

- [profile management](#)
- [name](#)
- [description](#)
- [omci-configuration](#)
- [url](#)
- [username](#)
- [password](#)
- [show profile management](#)

In the **PROFILE MANAGEMENT** mode, the IP parameters of the ONT management interface are configured (for client TR-069).

After configuration, the **management** profile can be assigned to a specific template or directly in the ONT configuration.

profile management

This command allows to create a new profile and/or switch to its configuration mode.

The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] profile management [<NAME>]
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile management name
ma4000(config-management) ("name")#
```

name

The command changes the name of the configured profile.

Syntax

```
name <STRING>
```

Parameters

<STRING> – profile name, case sensitive.

Command mode

PROFILE MANAGEMENT

Example

```
ma4000(config-management)("name"># name test
```

description

This command allows to add a description for the profile to be configured.

The use of a negative form (no) of the command removes description.

Syntax

```
[no] description <STRING>
```

Parameters

<STRING> – text description.

Command mode

PROFILE MANAGEMENT

Example

```
ma4000(config-management)("name"># description doc
```

omci-configuration

This command allows to enable setting of IP interface parameters and TR-069 server configuration by OMCI.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
omci-configuration <STRING>
[no] omci-configuration
```

Parameters

The command does not contain parameters

Command mode

PROFILE MANAGEMENT

Example

```
ma4000(config-management)("name"># omci-configuration
```

url

The command allows to set the URL of the ACS server.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
url <URL>  
[no] url
```

Parameters

<URL> – ACS server in format <http://ipaddr:port>.

Command mode

PROFILE MANAGEMENT

Example

```
ma4000(config-management)("name"># url http://tr.ru:9595
```

username

This command allows to set a user name to connect to the ACS server.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
username <NAME>  
[no] username
```

Parameters

<NAME> – user name, 25 characters max.

Command mode

PROFILE MANAGEMENT

Example

```
ma4000(config-management)("name"># username acsacs
```

password

This command allows to set a password to connect the user to the ACS server.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
password <STRING>
[no] password
```

Parameters

<STRING> – user password, 25 characters max.

Command mode

PROFILE MANAGEMENT

Example

```
ma4000(config-management)("name"># password password123
```

show profile management

This command allows to view the existing profiles, view the configuration of the specified profile.

Syntax

```
show profile management <NAME>
```

Parameters

<NAME> – profile name, optional parameter.

If no profile name is specified – the list of all profiles of this type is displayed, if the name is specified – the configuration of this profile is displayed.

Command mode

ROOT

Example 1

```
ma4000# show profile management
##          Name      Description
 1      management-00  ONT Profile Management 0
 2          name      ONT Profile Management 1
```

Example 2

```
ma4000# show profile management management-00
Name:                               'management-00'
Description:                         'ONT Profile Management 0'
Enable omci configuration:           true
Url:                                 ''
Username:                            ''
Password:                            ''
```

46 PLC8 board ONT configuration profile management. PORTS profile

- profile ports
- name
- description
- igmp immediate-leave
- igmp multicast dynamic-entry
- igmp query interval
- igmp query last-member
- igmp query response
- igmp mode
- igmp querier
- igmp robustness
- igmp version
- mld immediate-leave
- mld multicast dynamic-entry
- mld query interval
- mld query last-member
- mld query response
- mld mode
- mld querier
- mld robustness
- mld version
- veip multicast
- veip max groups
- veip max bandwidth
- veip downstream tag-control
- veip downstream vid
- veip downstream priority
- veip upstream tag-control
- veip upstream vid
- veip upstream priority
- port bridge group
- port bridge group spanning-tree
- port multicast
- port igmp max groups
- port igmp max bandwidth
- port igmp downstream tag-control
- port igmp downstream vid
- port igmp downstream priority
- port igmp upstream tag-control
- port igmp upstream vid
- port igmp upstream priority
- port shaper
- port shaper downstream committed-rate
- port shaper upstream committed-rate
- port shaper downstream peak-rate
- port shaper upstream peak-rate
- show profile ports

In the **PROFILE PORTS** mode, you can set the parameters and modes of ONT ETHERNET/VEIP ports operation, the parameters of multicast traffic transmission.

After configuration, the ports profile can be assigned to a specific template or directly in the ONT configuration.

profile ports

This command allows to create a new profile and/or switch to its configuration mode.
The use of a negative form (no) of the command removes the profile.

Syntax

```
[no] profile ports <NAME>
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile ports name  
ma4000(config-ports)("name")#
```

name

The command changes the name of the configured profile.

Syntax

```
name <STRING>
```

Parameters

<STRING> – profile name, case sensitive.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# name test
```

description

This command allows to add a description for the profile to be configured.
The use of a negative form (no) of the command removes description.

Syntax

```
[no] description <STRING>
```

Parameters

<STRING> – text description.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# description doc
```

igmp immediate-leave

The command allows to configure the IGMP immediate-leave parameters for quick disconnection from the group.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
[no] igmp immediate-leave
```

Parameters

The command contains no arguments.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp immediate-leave
```

igmp multicast dynamic-entry

The command allows to configure the IGMP parameters of MS-channels distribution over VLAN.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp multicast dynamic-entry <NUMBER> vid <VLANID> group <FIRSTIP> <LASTIP>
[no] igmp multicast dynamic-entry <NUMBER>
```

Parameters

<NUMBER> – number of the entry in the list, may take values [0..19];

<VLANID> – number of VLAN, via which the MC stream is transmitted, may take values [1..4094];

<FIRSTIP> – start multicast address of the range: [224.0.0.0 – 239.255.255.255];

<LASTIP> – end multicast address of the range: [224.0.0.0 – 239.255.255.255].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp multicast dynamic-entry 1 vid 23 group 224.1.1.1 224.1.1.240
```

igmp query interval

The command allows to configure the IGMP query interval – IGMP Query message sending interval.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp query interval <TIME>
[no] igmp query interval
```

Parameters

<TIME> – IGMP message sending interval, may take values [30..600] seconds.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp query interval 150
```

igmp query last-member

The command allows to configure the IGMP last-member interval parameters – the interval of waiting for the response to the IGMP group-specific queries.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp query last-member <TIME>
[no] igmp query last-member
```

Parameters

<TIME> – IGMP message waiting interval, may take values [30..600] *0,1 s.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp query interval last-member 100
```

igmp query response

The command allows to configure the IGMP query interval response – IGMP Query general message response waiting interval. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp query response <TIME>
[no] igmp query response
```

Parameters

<TIME> – IGMP message waiting interval, may take values [50..2000] seconds.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp query interval response 100
```

igmp mode

The command allows to configure the IGMP parameters: device operation mode. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp mode <TYPE>
[no] igmp mode
```

Parameters

<TYPE> – operation mode:

- proxy – IGMP proxy mode;
- snooping – IGMP snooping mode;
- spr – IGMP snooping snooping with proxy reporting.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp mode spr
```

igmp querier

This command allows to configure the IGMP parameters: querier IP – the address on which behalf the IGMP messages will be sent.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp querier <IP>
[no] igmp querier
```

Parameters

<IP> – IP address, on behalf of which IGMP messages will be sent.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp querier 192.168.55.5
```

igmp robustness

The command allows to configure the IGMP parameters: robustness – indicator of the reliability of receiving IGMP messages (number of report messages sent by the client).

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp robustness <COUNT>  
[no] igmp robustness
```

Parameters

<COUNT> – number of messages, may take values [1..10].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp robustness 3
```

igmp version

The command allows to configure the IGMP parameters: version – support of operation in accordance with the specified version of IGMP.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
igmp version <VERSION>  
[no] igmp version
```

Parameters

<VERSION> – IGMP version, may take values [1..3].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# igmp version 2
```

mld immediate-leave

The command allows to configure the MLD immediate-leave parameters for quick disconnection from the group.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
[no] mld immediate-leave
```

Parameters

The command contains no arguments.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# mld immediate-leave
```

mld multicast dynamic-entry

The command allows to configure the MLD parameters of MS-channels distribution over VLAN.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld multicast dynamic-entry <NUMBER> vid <VLANID> group <FIRSTIP> <LASTIP>
[no] mld multicast dynamic-entry <NUMBER>
```

Parameters

<NUMBER> – number of the entry in the list, may take values [0..19];

<VLANID> – number of VLAN, via which the MC stream is transmitted, may take values [1..4094];

<FIRSTIP> – start multicast address of the range;

<LASTIP> – end multicast address of the range:

[FF00:: – FF7F:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# mld multicast dynamic-entry 1 vid 23 group ff15:: ff15::ffff
```

mld query interval

The command allows to configure the MLD query interval – MLD Query message sending interval.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld query interval <TIME>  
[no] mld query interval
```

Parameters

<TIME> – MLD message sending interval, may take values [30..600] seconds.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># mld query interval 150
```

mld query last-member

The command allows to configure the MLD last-member interval parameters – the interval of waiting for the response to the MLD group-specific queries.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld query last-member <TIME>  
[no] mld query last-member
```

Parameters

<TIME> – MLD message waiting interval, may take values [30..600] *0,1 s.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># mld query interval last-member 100
```

mld query response

The command allows to configure the MLD query response – MLD Query general message response waiting interval. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld query response <TIME>
[no] mld query response
```

Parameters

<TIME> – MLD message waiting interval, may take values [50..2000] seconds.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# mld query interval response 100
```

mld mode

The command allows to configure the MLD parameters: device operation mode. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld mode <TYPE>
[no] mld mode
```

Parameters

<TYPE> – operation mode:

- proxy – MLD proxy mode;
- snooping – MLD snooping mode;
- spr – MLD snooping with proxy reporting.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# mld mode spr
```

mld querier

This command allows to configure the MLD parameters: querier IP – the address on which behalf the MLD messages will be sent.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld querier <IP>
[no] mld querier
```

Parameters

<IP> – IP address, on behalf of which MLD messages will be sent.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># mld querier 192.168.55.5
```

mld robustness

The command allows to configure the MLD parameters: robustness – indicator of the reliability of receiving MLD messages (number of report messages sent by the client).

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld robustness <COUNT>
[no] mld robustness
```

Parameters

<COUNT> – number of messages, may take values [1..10].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># mld robustness 3
```

mld version

The command allows to configure the MLD parameters: version – support of operation in accordance with the specified version of MLD.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
mld version <VERSION>  
[no] mld version
```

Parameters

<VERSION> – MLD protocol version, takes values in the range of [1..2].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# mld version 2
```

veip multicast

This command allows to enable multicast traffic processing in the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
[no] veip multicast
```

Parameters

The command contains no arguments.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip multicast
```

veip max groups

This command allows to set a limit on the maximum number of multicast groups transmitted on the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip max groups <COUNT>  
[no] veip max groups
```

Parameters

<COUNT> – number of groups <0-65535>, when '0' is specified there are no limits.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip max groups 50
```

veip max bandwidth

This command allows to set a limit on the maximum channel bandwidth for multicast traffic on the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip max bandwidth <BANDWIDTH>
[no] veip max bandwidth
```

Parameters

<BANDWIDTH> – bandwidth, may take values [0..4294967295] bytes/s.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip max bandwidth 102400
```

veip downstream tag-control

This command allows to set VLAN manipulation rules to transmit downstream multicast traffic on the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip downstream tag-control <RULE>
[no] veip downstream tag-control
```

Parameters

<RULE> – processing rule:

- add-tag – add VLAN tag (vlan id and p-bits);
- add-tag-from-subscriber-info – not realised in the current firmware version;
- pass – pass without modifications;
- remove-tag – remove VLAN tag (vlan id and p-bits);
- replace-tag – replace VLAN tag (vlan id and p-bits);
- replace-tag-from-subscriber-info – not realised in the current firmware version;
- replace-vid – replace VLAN ID;
- replace-vid-from-subscriber-info – not realised in the current firmware version.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip downstream tag-control pass
```

veip downstream vid

This command allows to set vlanid value for multicast traffic on the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip downstream vid <VLANID>
[no] veip downstream vid
```

Parameters

<VLANID> – VLAN number, may take values [1..4094].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip downstream vid 50
```

veip downstream priority

This command allows to set p-bit value for multicast traffic on the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip downstream priority <PRIORITY>
[no] veip downstream priority
```

Parameters

<PRIORITY> – p-bit value for the multicast traffic on the VEIP interface, may take values [0..7].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip downstream priority 5
```

veip upstream tag-control

This command allows to set VLAN manipulation rules for upstream transfer of IGMP traffic from the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip upstream tag-control <RULE>
[no] veip upstream tag-control
```

Parameters

<RULE> – processing rule:

- add-tag – add VLAN tag (vlanid+p-bits);
- pass – pass without modifications;
- replace-tag – replace VLAN tag (vlanid+p-bits), for untag traffic the tag is adding;
- replace-vid – replace VLANID.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip upstream tag-control pass
```

veip upstream vid

This command allows to set the VLANID value for upstream IGMP traffic transmitted from the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip upstream vid <VLANID>
[no] veip upstream vid
```

Parameters

<VLANID> – VLAN number, may take values [1..4094].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip upstream vid 50
```

veip upstream priority

This command allows to set the p-bit value for upstream IGMP traffic received from the VEIP interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
veip upstream priority <PRIORITY>
[no] veip upstream priority
```

Parameters

<PRIORITY> – p-bit value for the upstream IGMP traffic, may take values [0..7].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# veip upstream priority 5
```

port bridge group

This command is used to set LAN port binding of ONT to VEIP interface or OMCI-bridge

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> bridge group <GROUP>
[no] port <NUMBER> bridge group
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<GROUP> – group number, may take values [0..255], when 0 is specified, the VEIP binding is performed.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 bridge group 10
```

port bridge group spanning-tree

This command enables STP support on the OMCI-bridge interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> bridge group spanning-tree
[no] port <NUMBER> bridge group spanning-tree
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 bridge group spanning-tree
```

port multicast

This command allows to enable multicast traffic processing in the ONT LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> multicast
[no] port <NUMBER> multicast
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># port 0 multicast
```

port igmp max groups

This command allows to set a limit on the maximum number of multicast groups transmitted on the LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp max groups <COUNT>
[no] port <NUMBER> igmp max groups
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<COUNT> – number of groups, may take values <0-65535>, when 0 is specified there are no limits.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># port 0 igmpmaxgroups 100
```

port igmp max bandwidth

This command allows to set a limit on the maximum channel bandwidth for multicast traffic on the LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp max bandwidth <BANDWIDTH>
[no] port <NUMBER> igmp max bandwidth
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<BANDWIDTH> – bandwidth, may take values [0..4294967295] bytes/s.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 igmp max bandwidth 102400
```

port igmp downstream tag-control

This command allows to set VLAN manipulation rules to transmit downstream multicast traffic on the LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp downstream tag-control <RULE>
[no] port <NUMBER> downstream tag-control
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<rule> – processing rule:

- add-tag – add VLAN tag (vlan id and p-bits);
- add-tag-from-subscriber-info – not realised in the current firmware version;
- pass – pass without modifications;
- remove-tag – remove VLAN tag (vlan id and p-bits);
- replace-tag – replace VLAN tag (vlan id and p-bits);
- replace-tag-from-subscriber-info – not realised in the current firmware version;
- replace-vid – replace VLAN ID;
- replace-vid-from-subscriber-info – not realised in the current firmware version.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 downstream tag-control pass
```

port igmp downstream vid

This command allows to set vlanid value for multicast traffic on the LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp downstream vid <VLANID>
[no] port <NUMBER> igmp downstream vid
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<VLANID> – VLAN number, may take values [1..4094].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 downstream vid 50
```

port igmp downstream priority

This command allows to set p-bit value for multicast traffic on the LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp downstream priority <PRIORITY>
[no] port <NUMBER> igmp downstream priority
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<PRIORITY> – priority, may take values [0 .. 7].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 downstream priority 5
```

port igmp upstream tag-control

This command allows to set VLAN manipulation rules for upstream transfer of IGMP traffic from the LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp upstream tag-control<RULE>
[no] port <NUMBER> igmp upstream tag-control
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<rule> – processing rule:

- add-tag – add VLAN tag (vlanid+p-bits);
- pass – pass without modifications;
- replace-tag – replace VLAN tag (vlanid+p-bits), for untag traffic the tag is adding;
- replace-vid – replace VLAN ID.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 upstream tag-control pass
```

port igmp upstream vid

This command allows to set the VLANID value for upstream IGMP traffic transmitted from the LAN interface.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp upstream vid <VLANID>
[no] port <NUMBER> igmp upstream vid
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<VLANID> – VLAN number, may take values [1..4094].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># port 0 upstream vid 50
```

port igmp upstream priority

This command allows to set the p-bit value for upstream IGMP traffic received from the LAN interface. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> igmp upstream priority <PRIORITY>
[no] port <NUMBER> igmp upstream priority
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<PRIORITY> – priority, may take values [0 .. 7].

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name"># port 0 upstreampriority 5
```

port shaper

This command is used to enable a baud rate limit for traffic transmitted/received from the ONT LAN interface. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> shaper <DIRECTION>
[no] port <NUMBER> shaper downstream
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<DIRECTION> – traffic direction:

- upstream – upstream;
- downstream – downstream.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 shaper downstream
```

port shaper downstream committed-rate

This command is used to set a baud rate limit for traffic transmitted from the ONT LAN interface. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> shaper downstream committed-rate <BANDWIDTH>
[no] port <NUMBER> shaper downstream committed-rate
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<BANDWIDTH> – speed limit, may take values [0..2488320] kbps.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 shaper downstream committed-rate 102400
```

port shaper upstream committed-rate

This command is used to set a baud rate limit for traffic received from the ONT LAN interface. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> shaper upstream committed-rate <BANDWIDTH>
[no] port <NUMBER> shaper upstream committed-rate
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<BANDWIDTH> – speed limit, may take values [0..1244160] kbps.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 shaper upstream committed-rate 102400
```

port shaper downstream peak-rate

This command is used to set a peak baud rate limit for traffic transmitted from the ONT LAN interface. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> shaper downstream peak-rate <BANDWIDTH>
[no] port <NUMBER> shaper downstream peak-rate
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<BANDWIDTH> – speed limit, may take values [0..2488320] kbps.

Command mode

PROFILE PORTS

Example

```
ma4000(config-ports)("name")# port 0 shaper downstream peak-rate 102400
```

port shaper upstream peak-rate

This command is used to set a peak baud rate limit for traffic received from the ONT LAN interface. The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
port <NUMBER> shaper upstream peak-rate <BANDWIDTH>
[no] port <NUMBER> shaper upstream peak-rate
```

Parameters

<NUMBER> – ONT LAN port number, may take values [0..3]. Defined by ONT type.

<BANDWIDTH> – speed limit, may take values [0..1244160] kbps.

Command mode**PROFILE PORTS****Example**

```
ma4000(config-ports)("name"># port 0 shaper upstream peak-rate 102400
```

show profile ports

This command allows to view the existing profiles, view the configuration of the specified profile.

Syntax

```
show profile ports <NAME>
```

Parameters

<NAME> – profile name, optional parameter.

If no profile name is specified – the list of all profiles of this type is displayed, if the name is specified – the configuration of this profile is displayed.

Command mode**ROOT****Example 1**

```
ma4000# show profile ports
##          Name      Description
 1      ports-00      ONT Profile Ports 0
 2          name      ONT Profile Ports 1
```

Example 2

```

ma4000# show profile ports ports-00
Name: 'ports-00'
Description: 'ONT Profile Ports 0'
Igmp settings:
  Version: 3
  Mode: snooping
  Immediate leave: false
  Robustness: 2
  Querier ip: 0.0.0.0
  Query interval: 125
  Query response interval: 100
  Last member query interval: 10
  Multicast dynamic entry [0]:
    Vlan id: unused
    First group ip: 0.0.0.0
    Last group ip: 0.0.0.0
  Multicast dynamic entry [1]:
    Vlan id: unused
    First group ip: 0.0.0.0
    Last group ip: 0.0.0.0
  Multicast dynamic entry [2]:
    Vlan id: unused
    First group ip: 0.0.0.0
    Last group ip: 0.0.0.0
  Multicast dynamic entry [3]:
    Vlan id: unused
    First group ip: 0.0.0.0
    Last group ip: 0.0.0.0
  Multicast dynamic entry [4]:
    Vlan id: unused
    First group ip: 0.0.0.0
    Last group ip: 0.0.0.0
Veip:
  Multicast enable: false
  Multicast port settings:
    Upstream igmp vid: 1
    Upstream igmp prio: 0
    Upstream igmp tag control: pass
    Downstream multicast vid: 1
    Downstream multicast prio: 0
    Downstream multicast tag control: pass
    Max groups: 0
    Max multicast bandwidth: 0
Port [0]:
  Bridge group: 0
  Spanning tree for bridge group: false
  Multicast enable: false
  Multicast port settings:
    Upstream igmp vid: 1
    Upstream igmp prio: 0
    Upstream igmp tag control: pass
    Downstream multicast vid: 1
    Downstream multicast prio: 0
    Downstream multicast tag control: pass
    Max groups: 0

```

```

    Max multicast bandwidth: 0
    Shaper downstream:
      Enable: false
      Committed rate: 1000000
    Shaper upstream:
      Enable: false
      Committed rate: 1000000
Port [1]:
  Bridge group: 0
  Spanning tree for bridge group: false
  Multicast enable: false
  Multicast port settings:
    Upstream igmp vid: 1
    Upstream igmp prio: 0
    Upstream igmp tag control: pass
    Downstream multicast vid: 1
    Downstream multicast prio: 0
    Downstream multicast tag control: pass
    Max groups: 0
    Max multicast bandwidth: 0
  Shaper downstream:
    Enable: false
    Committed rate: 1000000
  Shaper upstream:
    Enable: false
    Committed rate: 1000000
Port [2]:
  Bridge group: 0
  Spanning tree for bridge group: false
  Multicast enable: false
  Multicast port settings:
    Upstream igmp vid: 1
    Upstream igmp prio: 0
    Upstream igmp tag control: pass
    Downstream multicast vid: 1
    Downstream multicast prio: 0
    Downstream multicast tag control: pass
    Max groups: 0
    Max multicast bandwidth: 0
  Shaper downstream:
    Enable: false
    Committed rate: 1000000
  Shaper upstream:
    Enable: false
    Committed rate: 1000000
Port [3]:
  Bridge group: 0
  Spanning tree for bridge group: false
  Multicast enable: false
  Multicast port settings:
    Upstream igmp vid: 1
    Upstream igmp prio: 0
    Upstream igmp tag control: pass
    Downstream multicast vid: 1
    Downstream multicast prio: 0
    Downstream multicast tag control: pass
    Max groups: 0
    Max multicast bandwidth: 0
  Shaper downstream:

```

Enable:	false
Committed rate:	1000000
Shaper upstream:	
Enable:	false
Committed rate:	1000000

47 PLC8 board ONT configuration profile management. SCRIPTING profile

- [scripting](#)
- [name](#)
- [description](#)
- [show profile scripting](#)

In the **profile scripting** mode, the ONT script configuration profiles are configured.

After configuration, the **cross-connect** profile can be assigned to a specific template or directly in the ONT configuration.

The full description of the functionality can be clarified by contacting the technical support service.

scripting

This command allows to create a new profile and/or switch to its configuration mode.

The use of a negative form (no) of the command removes the profile.

Syntax

```
profile scripting <NAME>
[no] profile scripting <NAME>
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile scripting name
ma4000(config-scripting)("name")#
```

name

The command changes the name of the configured profile.

Syntax

```
name <STRING>
```

Parameters

<STRING> – profile name, case sensitive.

Command mode

PROFILE SCRIPTING

Example

```
ma4000(config-scripting)("name"># name test
```

description

This command allows to add a description for the profile to be configured.

The use of a negative form (no) of the command removes description.

Syntax

```
description <STRING>  
[no] description <STRING>
```

Parameters

<STRING> – text description.

Command mode

PROFILE SCRIPTING

Example

```
ma4000(config-scripting)("name"># description doc
```

show profile scripting

This command allows to view the existing profiles, view the configuration of the specified profile.

Syntax

```
show profile scripting <NAME>
```

Parameters

<NAME> – profile name, optional parameter, case sensitive.

If no profile name is specified – the list of all profiles of this type is displayed, if the name is specified – the configuration of this profile is displayed.

Command mode

ROOT

Example 1

```
ma4000# show profile scripting
##          Name      Description
 1      scripting-00  ONT Profile Scripting 0
```

Example 2

```
ma4000# show profile scripting scripting-00
Name:                                     'scripting-00'
Description:                             'ONT Profile Scripting 0'
```

48 PLC8 board ONT configuration profile management. SHAPING profile

- [shaping](#)
- [name](#)
- [description](#)
- [downstream one-policer](#)
- [downstream policer](#)
- [downstream policer peak-rate](#)
- [upstream](#)
- [upstream committed-rate](#)
- [upstream peak-rate](#)
- [upstream broadcast](#)
- [upstream broadcast committed-rate](#)
- [upstream broadcast peak-rate](#)
- [upstream multicast](#)
- [upstream multicast committed-rate](#)
- [upstream multicast peak-rate](#)
- [upstream unicast](#)
- [upstream unicast committed-rate](#)
- [upstream unicast peak-rate](#)
- [storm-control](#)
- [show profile shaping](#)

In **profile shaping** mode, upstream/downstream data rate limitation is set for GEM ONT port.

After configuration, the **shaping** profile can be assigned to a specific template or directly in the ONT configuration.

shaping

This command allows to create a new profile and/or switch to its configuration mode.

The use of a negative form (no) of the command removes the profile.

Syntax

```
profile shaping <NAME>
[no] profile shaping <NAME>
```

Parameters

<NAME> – profile name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# profile shaping name
ma4000(config-shaping)("name")#
```

name

The command changes the name of the configured profile.

Syntax

```
name <STRING>
```

Parameters

<STRING> – profile name, case sensitive.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name"># name test
```

description

This command allows to add a description for the profile to be configured.

The use of a negative form (no) of the command removes description.

Syntax

```
description <STRING>  
[no] description <STRING>
```

Parameters

<STRING> – text description.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name"># description doc
```

downstream one-policer

The command allows to configure the bandwidth limit for all services simultaneously.

The use of the negative form (no) of the command sets a bandwidth limit for each service by a separate rule.

Syntax

```
[no] downstream one-policer
```

Parameters

The command contains no arguments.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# downstream one-policer
```

downstream policer

The command allows to enable band limit for a particular service.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
[no] downstream policer <SERVICE>
```

Parameters

<SERVICE> – service number, may take values [0..7].

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# downstream policer 0
```

downstream policer peak-rate

The command allows to set the peak bandwidth limit for a particular service.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
downstream policer <SERVICE> peak-rate <RATE>
[no] downstream policer <SERVICE> peak-rate
```

Parameters

<SERVICE> – service number, may take values [0..7];
 <RATE> – peak rate, may take values [0..2488320] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# downstream policer 0 peak-rate 102400
```

upstream

The command allows to enable the general bandwidth limit for a particular service.
 The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream <SERVICE>
[no] upstream <SERVICE>
```

Parameters

<SERVICE> – service number, may take values [0..7].

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream 7
```

upstream committed-rate

The command allows to set the general bandwidth limit for a particular service.
 The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream <SERVICE> committed-rate
[no] upstream <SERVICE> committed-rate <RATE>
```

Parameters

<SERVICE> – service number, may take values [0..7];
 <RATE> – rate value, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream 7 committed-rate
```

upstream peak-rate

The command allows to set the peak bandwidth limit for a particular service.
 The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream <SERVICE> peak-rate  

[no] upstream <SERVICE> peak-rate
```

Parameters

<SERVICE> – service number, may take values [0..7];
 <RATE> – peak rate, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream 7 peak-rate
```

upstream broadcast

The command allows to enable the bandwidth limit for a particular service broadcast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream broadcast <SERVICE>  

[no] upstream broadcast <SERVICE>
```

Parameters

<SERVICE> – service number, may take values [0..7].

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream broadcast 7
```

upstream broadcast committed-rate

The command allows to set the bandwidth limit for a particular service broadcast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream broadcast <SERVICE> committed-rate
[no] upstream broadcast <SERVICE> committed-rate <RATE>
```

Parameters

<SERVICE> – service number, may take values [0..7];

<RATE> – rate value, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream broadcast 7 committed-rate
```

upstream broadcast peak-rate

The command allows to set the peak bandwidth limit for a particular service broadcast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream broadcast <SERVICE> peak-rate
[no] upstream broadcast <SERVICE> peak-rate
```

Parameters

<SERVICE> – service number, may take values [0..7];
 <RATE> – peak rate, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream broadcast 7 peak-rate
```

upstream multicast

The command allows to enable the bandwidth limit for a particular service multicast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream multicast <SERVICE>
[no] upstream multicast <SERVICE>
```

Parameters

<SERVICE> – service number, may take values [0..7].

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream multicast 7
```

upstream multicast committed-rate

The command allows to set the bandwidth limit for a particular service multicast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream multicast <SERVICE> committed-rate
[no] upstream multicast <SERVICE> committed-rate <RATE>
```

Parameters

<SERVICE> – service number, may take values [0..7];
 <RATE> – rate value, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstreammulticast 7 committed-rate
```

upstream multicast peak-rate

The command allows to set the peak bandwidth limit for a particular service multicast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream multicast <SERVICE> peak-rate  

[no] upstream multicast <SERVICE> peak-rate
```

Parameters

<SERVICE> – service number, may take values [0..7];
 <RATE> – peak rate, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream multicast 7 peak-rate
```

upstream unicast

The command allows to enable the bandwidth limit for a particular service unicast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream unicast <SERVICE>  

[no] upstream unicast <SERVICE>
```

Parameters

<SERVICE> – service number, may take values [0..7].

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream unicast 7
```

upstream unicast committed-rate

The command allows to set the bandwidth limit for a particular service unicast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream unicast <SERVICE> committed-rate
[no] upstream unicast <SERVICE> committed-rate <RATE>
```

Parameters

<SERVICE> – service number, may take values [0..7];

<RATE> – rate value, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstreamunicast 7 committed-rate
```

upstream unicast peak-rate

The command allows to set the peak bandwidth limit for a particular service unicast traffic. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
upstream unicast <SERVICE> peak-rate
[no] upstream unicast <SERVICE> peak-rate
```

Parameters

<SERVICE> – service number, may take values [0..7];
 <RATE> – rate value, may take values [0..1244160] kbps.

Command mode

PROFILE SHAPING

Example

```
ma4000(config-shaping)("name")# upstream unicast 7 peak-rate
```

storm-control

The command enables storm-control in upstream or downstream direction. The limit is set for broadcast or multicast.

Syntax

<DIRECTION> <TYPE> rate-limit <RATE> <shutdown> <logging>

Parameters

<DIRECTION> – traffic direction [upstream \ downstream];
 <RATE> – limiting (packets per second) <1-1000000>;
 <shutdown> – to block ONT when <RATE> is exceeded;
 <logging> – to log ONT blocking caused by storm-control.

Example

```
ma4000(config-shaping)("name")# upstream multicast storm-control rate-limit 300 shutdown  
logging
```

show profile shaping

This command allows to view the existing profiles, view the configuration of the specified profile.

Syntax

show profile shaping <NAME>

Parameters

<NAME> – profile name, optional parameter.

If no profile name is specified – the list of all profiles of this type is displayed, if the name is specified – the configuration of this profile is displayed.

Command mode

ROOT

Example 1

```
ma4000# show profile shaping
##          Name      Description
 1         shaping-00  ONT Profile Shaping 0
```

Example 2

```

ma4000# show profile shaping shaping-00
Name:                                     'shaping-00'
Description:                             'ONT Profile Shaping 0'
Downstream:
  One policer:                           true
  Policer [0]:
    Enable:                             false
    Peak rate:                          2488320
  Policer [1]:
    Enable:                             false
    Peak rate:                          2488320
  Policer [2]:
    Enable:                             false
    Peak rate:                          2488320
  Policer [3]:
    Enable:                             false
    Peak rate:                          2488320
  Policer [4]:
    Enable:                             false
    Peak rate:                          2488320
  Policer [5]:
    Enable:                             false
    Peak rate:                          2488320
  Policer [6]:
    Enable:                             false
    Peak rate:                          2488320
  Policer [7]:
    Enable:                             false
    Peak rate:                          2488320
Upstream:
  Shaper [0]:
    Enable:                             true
    Committed rate:                     1244160
    Peak rate:                          1244160
  Shaper [1]:
    Enable:                             true
    Committed rate:                     1244160
    Peak rate:                          1244160
  Shaper [2]:
    Enable:                             true
    Committed rate:                     1244160
    Peak rate:                          1244160
  Shaper [3]:
    Enable:                             true
    Committed rate:                     1244160
    Peak rate:                          1244160
  Shaper [4]:
    Enable:                             true
    Committed rate:                     1244160
    Peak rate:                          1244160
  Shaper [5]:
    Enable:                             true
    Committed rate:                     1244160
    Peak rate:                          1244160
  Shaper [6]:

```

```

    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper [7]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [0]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [1]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [2]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [3]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [4]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [5]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [6]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper unicast [7]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper multicast [0]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper multicast [1]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper multicast [2]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper multicast [3]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160
Shaper multicast [4]:
    Enable: true
    Committed rate: 1244160
    Peak rate: 1244160

```

```

Shaper multicast [5]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper multicast [6]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper multicast [7]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [0]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [1]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [2]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [3]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [4]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [5]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [6]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160
Shaper broadcast [7]:
  Enable: true
  Committed rate: 1244160
  Peak rate: 1244160

```

49 PLC8 board configuration templates ('TEMPLATE')

- [template](#)
- [name](#)
- [description](#)
- [ber interval](#)
- [ber update-period](#)
- [broadcast-downstream](#)
- [fec](#)
- [omci-error-tolerant](#)
- [password](#)
- [rf-port-state](#)
- [serial](#)
- [profile](#)
- [service](#)
- [define](#)
- [undefine](#)
- [show template](#)

In the ONT configuration 'TEMPLATE' mode, you can configure a set of profiles and ONT configuration parameters which will be used as a template. You can assign the template to an ONT or a group of ONTs, and all configuration template settings will overwrite those performed on ONT.

template

This command allows to create a new template and/or switch to its configuration mode.

The use of a negative form (no) of the command removes the template.

Syntax

```
[no] template <NAME>
```

Parameters

<NAME> – template name, case sensitive.

Command mode

CONFIG

Example

```
ma4000(config)# template name
ma4000(ont-template) ("name")#
```

name

The command changes the name of the configured template.

Syntax

name <STRING>

Parameters

<STRING> – template name, case sensitive.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name"># name test
```

description

This command allows to add a description for the profile to be configured.

The use of a negative form (no) of the command removes description.

Syntax

[no] description <STRING>

Parameters

<STRING> – text description.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name"># description name
```

ber interval

This command allows to set the error rate count interval for the downstream direction. At the end of the interval, ONT sends a REI message.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ber interval <INTERVAL>
no ber interval
```

Parameters

<INTERVAL> – number of packets, may take values [0..4294967294]. If 'none' is specified, ONT will not send the REI message.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")# ber interval 500000
```

ber update-period

This command allows to set the interval of sending messages to the OLT with error rate data for the downstream direction.

The use of a negative form (no) of the command sets the default value.

Syntax

```
ber update-period <PERIOD>
no ber update-period
```

Parameters

<PERIOD> – time period, may take values [0..4294967295], in seconds.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")#ber update-period 500000
```

broadcast-downstream

This command enables sending broadcast packets in a downstream direction through the dedicated GEM port.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] broadcast-downstream
```

Parameters

The command contains no arguments.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")#broadcast-downstream
```

fec

This command enables upstream error correction mode when transmitting data from ONT.

The use of a negative form (no) of the command sets the default value.

Syntax

[no] fec

Parameters

The command contains no arguments.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")#fec
```

omci-error-tolerant

This command disables the processing of errors that occur during the ONT configuration by OMCI.

The use of a negative form (no) of the command sets the default value.

Syntax

[no] omci-error-tolerant

Parameters

The command contains no arguments.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")# omci-error-tolerant
```

password

This command sets the OMCI configuration of the ONT password.

The use of a negative form (no) of the command sets the default value.

Syntax

```
password <STRING>
no password
```

Parameters

<STRING> – authorization OMCI password, 10 characters.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")#password
```

rf-port-state

This command sets the operation mode of the ONT RF port in the configuration. Support for this feature should be implemented on ONT.

The use of a negative form (no) of the command sets the default value.

Syntax

```
rf-port-state <STATE>
no rf-port-state
```

Parameters

<STATE> – RF port operation mode:

- disabled – port disabled;
- enabled – port enabled;
- no-change – do not change port state.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")#rf-port-state enabled
```

serial

This command sets the ONT serial number (PONSERIAL ONT) in the template configuration.
The use of a negative form (no) of the command sets the default value.

Syntax

```
serial <PONSERIAL>
no serial
```

Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")#serial ELTX00000000
```

profile

This command sets ONT configuration profiles to the specified template.
The use of a negative form (no) of the command sets the default value.

Syntax

```
profile <TYPE> <NAME>
no profile <TYPE>
```

Parameters

<TYPE> – configuration profile type:

- management – set management profile;

- ports – set ports profile;
- shaping – set shaping profile;
- scripting – set scripting profile;

<NAME> – configuration profile name, case sensitive.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")#profile ports ports-00
```

service

This command sets ONT service configuration profiles to the specified template.

The use of a negative form (no) of the command sets the default value.

Syntax

```
service <SERVICEID> profile <TYPE> <NAME>
no service <SERVICEID> profile <TYPE>
```

Parameters

<SERVICEID> – service number, may take values [0..15];

<TYPE> – profile type:

- dba – set dba profile;
- cross-connect – set cross-connect profile.

<NAME> – configuration profile name, case sensitive.

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")# service 0 profile dba dba-00
```

define

This command adds a parameter to the ONT configuration template.

Syntax

```
define <PARAM>
```

Parameters

<PARAM> – ONT configuration parameter:

- Ber interval – add ber-interval setting;
- Ber update-period – add ber update-period setting;
- broadcast-downstream – add broadcast downstream sending parameters setting;
- profile management – add profile management setting;
- profile ports – add profile ports setting;
- profile shaping – add shaping setting;
- profile scripting – add profile scripting setting;
- serial – add GPON serial setting;
- password – add GPON password setting;
- fec – add fec configuration setting;
- omci-error-tolerant – add omci error processing configuration setting;
- rf-port-state – add rf port configuration setting;
- service <serviceid> profile dba – add DBA profile setting for service;
- service <serviceid> profile cross-connect – add cross-connect profile setting for service.

<SERVICEID>– service number, may take values [0..15].

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")# define rf-port-state
```

undefine

This command removes a parameter from the ONT configuration template.

Syntax

undefine <PARAM>

Parameters

<PARAM> – ONT configuration parameter:

- ber interval – remove ber-interval setting from the template;
- ber update-period – remove ber update-period setting from the template;
- broadcast-downstream – remove broadcast downstream sending parameters setting from the template;
- profile management – remove profile management setting from the template;
- profile ports – remove profile ports setting from the template;
- profile shaping – remove shaping setting from the template;
- profile scripting – remove profile scripting setting from the template;
- serial – remove GPON serial setting from the template;
- password – remove GPON password setting from the template;
- fec – remove fec configuration setting from the template;
- omci-error-tolerant – remove omci error processing configuration setting from the template;
- rf-port-state – remove rf port configuration setting from the template;

- service <serviceid> profile dba – remove DBA profile setting for service from the template;
- service <serviceid> profile cross-connect – remove cross-connect profile setting for service from the template.

<SERVICEID>– service number, may take values [0..7].

Command mode

TEMPLATE

Example

```
ma4000(ont-template)("name")# undefine ber interval
```

show template

This command displays the configuration template parameters.

Syntax

show template <NAME>

Parameters

<NAME> – template name, case sensitive. Optional parameter

If no template name is specified – the list of all templates is displayed, if the name is specified – the configuration of this template is displayed.

Command mode

ROOT

Example

```

Ma4000# show template template-00
Name:                               'template-00'
Description:                         'ONT Template 0'
-----
Fec up:                             false
Downstream broadcast:               true
Ber interval:                       100000
Ber update period:                   60
Rf port state:                       disabled
Omci error tolerant:                 false
-----
Profile shaping:                     shaping-00      ONT Profile Shaping 0
Profile ports:                       ports-00      ONT Profile Ports 0
Service [0]:
    Profile cross connect:           unassigned
    Profile dba:                     dba-00      ONT Profile DBA 0
Service [1]:
    Profile cross connect:           unassigned
    Profile dba:                     unassigned
Service [2]:
    Profile cross connect:           unassigned
    Profile dba:                     unassigned
Service [3]:
    Profile cross connect:           unassigned
    Profile dba:                     unassigned
Service [4]:
    Profile cross connect:           unassigned
    Profile dba:                     unassigned
Service [5]:
    Profile cross connect:           unassigned
    Profile dba:                     unassigned
Service [6]:
    Profile cross connect:           unassigned
    Profile dba:                     unassigned
Service [7]:
    Profile cross connect:           unassigned
    Profile dba:                     unassigned
Profile management:                  management-00  ONT Profile Management
0
Profile scripting:                   unassigned

```

50 Management modules monitoring

- [show cmd-dispatcher](#)
- [show evt-dispatcher](#)
- [show queue](#)
- [show sfp front-port](#)

show cmd-dispatcher

This command is used to view information about the command manager status.

Syntax

```
show cmd-dispatcher
```

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```
ma4000# show cmd-dispatcher
Command Dispatcher memory state:
  overload count      0
  errors              0
  size of element     1192
  free                500
length                500
```

show evt-dispatcher

This command is used to view information about the event manager status.

Syntax

```
show evt-dispatcher
```

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```

ma4000# show evt-dispatcher
Command Dispatcher memory state:
    overload count      0
    errors              0
    size of element     992
    free                500
length                 500

```

show queue

This command allows to view statistics for a given (or all) queue system.

Syntax

show queue <QUEUE>

Parameters

<QUEUE> – queue number, may take values [0 .. 199].

Command mode

ROOT

Example

```

ma4000#show queue 0
Queue event top manager      :
    tx count                 13
    rx count                 13
    overload count           0
    read from empty count    0
    pipe read errors         0
    pipe write errors        0
    size of element          4
    free                     500
    length                   500

```

show sfp front-port

This command is used to view the status of PP4X board SFP modules.

Syntax

show sfp front-port <RANGE>

Parameters

<RANGE> – interface number. The range of values and numbering rules are described in [Table 4.1](#). If you specify the value 'all' in the command, all interfaces of the given type will be specified. You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show sfp front-port all
SFP status:
~~~~~
Port          Temp [C]   Voltage [Volt]   Current [mA]   RX power [mWatt]   TX power
[mWatt]
-----
front-port 1/0  N/A      N/A             N/A            N/A                N/A
front-port 1/1  N/A      N/A             N/A            N/A                N/A
front-port 1/2  N/A      N/A             N/A            N/A                N/A
front-port 1/3  N/A      N/A             N/A            N/A                N/A
front-port 1/4  N/A      N/A             N/A            N/A                N/A
front-port 1/5  N/A      N/A             N/A            N/A                N/A
-----
N/A - Not Available, N/S - Not Supported
```

51 ONT/OLT operational management

- [reconfigure olt slot](#)
- [reconfigure interface gpon-port](#)
- [reconfigure interface ont](#)
- [send omci pptp-video-uni administrative-state](#)
- [send omci reset](#)
- [send omci restore interface ont](#)
- [send ploam disable-sn mode enable-all gpon-port](#)
- [send ploam disable-sn mode enable ont from-configuration gpon-port](#)
- [send ploam disable-sn mode enable ont serial](#)
- [send ploam disable-sn mode enable ont](#)
- [send ploam disable-sn mode disableont from-configuration gpon-port](#)
- [send ploam disable-sn mode disableont serial](#)
- [send ploam disable-sn mode disable ont](#)
- [send ploam ber-interval gpon-port](#)
- [send ploam ber-interval ont](#)
- [show slot gpon olt state](#)
- [show slot information](#)

reconfigure olt slot

This command applies the configuration and resets the current state (reconfig) of the GPON OLT interface to the specified slot or slot range.

Syntax

reconfigure olt slot <SLOT>.

Parameters

<SLOT> – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15).

Command mode

ROOT

Example

```
ma4000# reconfigure olt slot 1-2
```

reconfigure interface gpon-port

This command applies the configuration and resets the current state (reconfig) of the GPON interface (gpon-port).

Syntax

reconfigure interface gpon-port <SLOT>/<PORT>.

Parameters

<SLOT> – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);

<PORT> – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7).

Command mode

ROOT

Example

```
ma4000# reconfigure interface gpon-port 6/1
Slot 6
GPON-port 1 reconfigured successfully
```

reconfigure interface ont

This command applies the configuration and resets the current state (reconfig) of the ONT.

Syntax

reconfigure interface ont <SLOT>/<PORT>/<ONTID>.

Parameters

<SLOT> – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15).

<PORT> – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7).

<ONTID> – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

Command mode

ROOT

Example

```
ma4000# reconfigure interface ont 6/1/1-5,11,22
Slot 6
Reconfigure [ONT1/1]
[ONT1/1] (ELTX06001500) reconfigured successfully
Reconfigure [ONT1/2]
[ONT1/2] (ELTX0600208D) reconfigured successfully
Reconfigure [ONT1/3]
[ONT1/3] (ELTX06002671) reconfigured successfully
Reconfigure [ONT1/4]
[ONT1/4] (ELTX06002663) reconfigured successfully
Reconfigure [ONT1/5]
[ONT1/5] (ELTX0600266A) reconfigured successfully
Reconfigure [ONT1/11]
[ONT1/11] is not connected currently
Reconfigure [ONT1/22]
[ONT1/22] (ELTX06002654) reconfigured successfully
```

send omci pptp-video-uni administrative-state

This command performs operational management of the ONT RF port by OMCI command. Support for this feature should be implemented on ONT.

Syntax

send omci pptp-video-uni administrative-state <COMMAND> interface ont <INTERFACE>.

Parameters

<COMMAND> – port management command: enable/disable;

<INTERFACE> – ONT interface: SLOT/PORT/ONTID, where

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

Command mode

ROOT

Example

```
ma4000# send omci pptp-video-uni administrative-state enable interface ont 6/1/1
```

send omci reset

This command restarts ONT by OMCI command.

Syntax

send omci reset interface ont <interface>.

Parameters

<INTERFACE> – ONT interface: SLOT/PORT/ONTID;

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

Command mode

ROOT

Example

```
ma4000#send omci reset interface ont 6/1/1
[ONT 6/1/1] has been reset successfully
```

send omci restore interface ont

This command resets ONT to the factory settings by the OMCI command. Support for this feature should be implemented on ONT.

Syntax

send omci restore interface ont <INTERFACE>

Parameters

<INTERFACE> – ONT interface: slot/port/ontid;

- <SLOT> – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- <PORT> – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7).
- <ONTID> – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

Command mode

ROOT

Example

```
ma4000#send omci restore interface ont 6/1/2
[ONT 6/1/2] was successfully restored to default config and rebooted
```

send ploam disable-sn mode enable-all gpon-port

This command controls the operation of the optical transmitter ONT: enable on all ONTs connected to the specified GPON-port OLT.

Syntax

send ploam disable-sn mode enable-all gpon-port <interface>.

Parameters

<INTERFACE> – ONT interface, specified in format of SLOT/PORT/ONTID. Optional parameter. If an interface is not specified, the command to enable will be sent to all ONTs;

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

Command mode

ROOT

Example

```
ma4000# send ploam disable-sn mode enable-all gpon-port 6/1
```

send ploam disable-sn mode enable ont from-configuration gpon-port

This command controls the operation of the optical transmitter ONT: enable on all ONTs registered in configuration of the specified GPON-port OLT.

Syntax

send ploam disable-sn mode enable ont from-configuration [gpon-port <INTERFACE>]

Parameter

gpon-port – optional parameter, command to enable will be sent to all ONTs;

<INTERFACE> – ONT interface, specified in format of SLOT/PORT/ONTID:

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);

- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

Command mode

ROOT

Example

```
ma4000#send ploam disable-sn mode enable ont from-configuration gpon-port 6/6
```

send ploam disable-sn mode enable ont serial

This command controls the operation of the ONT optical transmitter: enable on ONT with the specified PON serial number.

Syntax

```
send ploam disable-sn mode enable ont serial <PONSERIAL>
```

Parameters

<PONSERIAL> – ONT serial number in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

Command mode

ROOT

Example

```
ma4000#send send ploam disable-sn mode enable ont serial ELTX06002628
```

send ploam disable-sn mode enable ont

This command controls the operation of the optical transmitter ONT: enable on ONT registered in configuration of the specified GPON-port OLT with specified ONTID.

Syntax

```
send ploam disable-sn mode enable ont<INTERFACE>
```

Parameters

<INTERFACE> – ONT interface, specified in format of SLOT/PORT/ONTID:

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

Command mode

ROOT

Example

```
ma4000#send ploam disable-sn mode enable ont 6/1/2
```

send ploam disable-sn mode disableont from-configuration gpon-port

This command controls the operation of the optical transmitter ONT: disable on all ONTs registered in configuration of the specified GPON-port OLT.

Syntax

```
send ploam disable-sn mode disableont from-configuration [gpon-port <INTERFACE>]
```

Parameters

gpon-port – optional parameter, command to enable will be sent to all ONTs;

<INTERFACE> – ONT interface, specified in format of SLOT/PORT/ONTID:

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

Command mode

ROOT

Example

```
ma4000#send ploam disable-sn mode disableont from-configuration gpon-port 6/6
```

send ploam disable-sn mode disableont serial

This command controls the operation of the ONT optical transmitter: disable on ONT with the specified PON serial number.

Syntax

```
send ploam disable-sn mode disableont serial <PONSERIAL>
```

Parameters

<PONSERIAL> – ONT serial number, specified in format:

- AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];
- XXXXXXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

Command mode

ROOT

Example

```
ma4000#send send ploam disable-sn mode disableont serial ELTX06002628
```

send ploam disable-sn mode disable ont

This command controls the operation of the optical transmitter ONT: disable on ONT registered in configuration of the specified GPON-port OLT with specified ONTID.

Syntax

```
send ploam disable-sn mode disable ont<INTERFACE>
```

Parameters

<INTERFACE> – ONT interface, specified in format of SLOT/PORT/ONTID:

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

Command mode

ROOT

Example

```
ma4000#send ploam disable-sn mode disableont 6/1/2
```

send ploam ber-interval gpon-port

This command sends a command to ONT: enable downstream error rate calculation for all ONTs registered in the configuration on the specified GPON-port OLT.

Syntax

```
send ploam ber-interval <COUNT>[gpon-port <INTERFACE>]
```

Parameters

<COUNT> – number of frames, may take values [1000..1000000000];

gpon-port – optional parameter, command to enable will be sent to all ONTs;

<INTERFACE> – ONT interface: SLOT/PORT/ONTID;

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range { ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID } (0..63).

Command mode

ROOT

Example

```
ma4000#send ploam ber-interval 1000 gpon-port 6/6
```

send ploam ber-interval ont

This command sends a command to ONT: enable downstream error rate calculation for ONT registered in the configuration on the specified GPON-port OLT with a specified ONTID.

Syntax

```
send ploam ber-interval <COUNT> ont <INTERFACE>
```

Parameters

<COUNT> – number of frames, may take values [1000..1000000000];

<INTERFACE> – ONT interface, specified in format of SLOT/PORT/ONTID:

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

Command mode

ROOT

Example

```
ma4000#send ploam ber-interval ont 6/6/6
```

show slot <SLOT> gpon olt state

Commands to view the software version on a specific PLC8 module:

- Device count – GPON chips number;
- Channels per device – number of ports on GPON chip;
- Driver version – driver version;
- Device 0 – information on GPON chip:
- Firmware version – GPON chip firmware version;
- Hardware version – GPON chip hardware version.

Syntax

```
show slot <SLOT> gpon olt state
```

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```

ma4000#show slot 6 gpon olt state
Device count:          2
Channels per device:   4
Driver version:        1.2.561
Device 0:
  Firmware version:    2.3.37.1012
  Hardware version:    5211.2
Device 1:
  Firmware version:    2.3.37.1012
  Hardware version:    5211.2

```

show slot <SLOT>information

Commands to view system information about a specific PLC8 module:

- Module type – module type;
- Hardware version – module hardware version;

- Software version – module firmware version;
- Build revision – firmware version build;
- Build time – firmware file creation time;
- Uptime (d:h:m:s) – module operating time since the last power-on;
- CPU load average (1m, 5m, 15m) – module CPU load;
- Free RAM/Total RAM (Mbytes) – free memory/memory size;
- Temperature (sensor1/sensor2) – sensor temperature.

Syntax

```
show slot <SLOT> information
```

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
show slot 6 information
-----
[slot 6] System information:
-----
Module type:                PLC8
Hardware version:           2
Software version:           3.22.0.414
Build revision:             41050
Build time:                 09:26:00 27/11/2014
Uptime (d:h:m:s):          0:4:15:53
CPU load average (1m, 5m, 15m): 0.29 0.86 1.00
Free RAM/Total RAM (Mbytes): 118/241
Temperature (sensor1/sensor2): 24C/23C
```

52 Static Routing

- [ip routing](#)
- [ip route allow](#)
- [ip route reject](#)
- [show ip route](#)
- [show running-config ip route](#)

ip routing

This command enables the L3 routing mode on PP4X. The device will only use route information to transfer DHCP towards Relay Server, full L3 routing is not available in this version.

The negative form of the command sets the default value of the parameter.

Syntax

```
[no] ip routing
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# ip routing
```

ip route allow

This command adds a new entry to the RR4X routing table: the route to transfer traffic to the specified subnet.

The negative form of the command removes the entry from the table.

Syntax

```
ip route allow <NETWORK> <GATEWAY> <METRIC>  
no ip route allow <NETWORK> <GATEWAY>
```

Parameters

<NETWORK> – network address/mask in format of A.B.C.D/F.G.H.I;

<GATEWAY> – gateway address;

<METRIC> – route metrics.

Command mode

CONFIG

Example

```
ma4000(config)# ip route allow 192.168.1.1 192.168.199.2
```

ip route reject

This command adds a new entry to the RR4X routing table: prohibition to transfer traffic from the specified subnet.

The negative form of the command removes the entry from the table.

Syntax

```
ip route reject <NETWORK>  
no ip route reject <NETWORK>
```

Parameters

<NETWORK> – network address/mask in format of A.B.C.D/F.G.H.I.

Command mode

CONFIG

Example

```
ma4000(config)# ip route reject 192.168.22.0/32
```

show ip route

This command allows to view the static routing table on PP4X.

Syntax

```
show ip route
```

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```
ma4000# show ip route
IP Forwarding: enabled

Codes: C - connected, S - static

S 192.168.22.0/32          rejected route
S 192.168.32.0/24          via 192.168.33.1 metric 10    vlan 333
C 192.168.33.0/24          is directly connected       vlan 333
C 192.168.34.0/24          is directly connected       vlan 334
C 192.155.155.0/24         is directly connected       vlan 1234
```

show running-config ip route

This command allows to view the static routing configuration on PP4X.

Syntax

```
show running-config ip route
```

Parameters

The command contains no arguments.

Command mode

ROOT

Example

```
ma4000# show running-config ip route
route all
ma4000# show running-config ip route
configure terminal
ip routing
ip route allow 192.168.1.0/24 192.168.199.2
ip route reject 192.168.22.0/32
exit
```

53 DHCP Relay

- [relay](#)
- [ip interface](#)

relay

This command specifies the DHCP relay address to which DHCP requests in this VLAN will be redirected. The negative form of the command removes the specified address.

Syntax

```
[no] relay <IP>
```

Parameters

<IP> - IP address in format of XXX.XXX.XXX.XXX

Command mode

CONFIG

Example

```
ma4000(vlan-100)# relay 172.15.2.45
```

ip interface

This command sets the address of the subinterface from which DHCP packets in this VLAN will be sent. The negative form of the command removes the specified address.

Syntax

```
[no] ip interface <IP>
```

Parameters

<IP> - IP address in format of XXX.XXX.XXX.XXX

Command mode

CONFIG

Example

```
ma4000(vlan-100)# ip interface 172.15.3.64
```

54 IP Source Guard configuration

- [ip source-guard enable](#)
- [ip source-guard mode](#)
- [ip source-guard bind](#)
- [no ip source-guard bind](#)
- [ip source-guard ignore-vlan](#)
- [show slot ip source-guard](#)

ip source-guard enable

This command enables IP Source Guard.

The negative form of the command disables IP Source Guard.

Syntax

```
[no] ip source-guard enable
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# ip source-guard enable
```

ip source-guard mode

This command selects the IP Source Guard operation mode.

The negative form of the command sets the default value of the parameter.

Syntax

```
ip source-guard mode <MODE>
```

Parameters

<MODE> – IP Source Guard operation mode:

- dynamic – IP Source Guard dynamic mode;
- static – IP Source Guard static mode.

Command mode

CONFIG

Example

```
ma4000(config)# ip source-guard mode dynamic
```

ip source-guard bind

This command creates a static binding of the sender IP address to the MAC address and service on ONT.

Syntax

```
ip source-guard bind ip <IP> mac <MAC> interface-ont <INTERFACE> service <NUM>
```

Parameters

<IP> – IP address in the format of XXX.XXX.XXX.XXX

<MAC> – MAC address in the format of XX:XX:XX:XX:XX:XX

<INTERFACE> – ONT interface: SLOT/PORT/ONTID:

- SLOT – slot number or range {slot or slot, slot or slot – slot or slot, slot – slot or slot – slot, slot – slot} (0..15);
- PORT – number or port range of GPON port on the line card {port or port, port or port – port or port, port – port or port – port, port – port} (0..7);
- ONTID – ONT number or number range {ONTID or ONTID, ONTID or ONTID – ONTID or ONTID, ONTID – ONTID or ONTID – ONTID, ONTID – ONTID} (0..63).

<NUM> - number of service on ONT, 0-7

Command mode

CONFIG

Example

```
ip source-guard bind ip 192.168.10.5 mac AA:BB:CC:DD:EE:FF interface-ont 0/0/0 service 2
```

no ip source-guard bind

This command deletes a static binding entry for a specified IP address.

Syntax

```
ip source-guard bind ip <IP>
```

Parameters

<IP> - IP address in format of XXX.XXX.XXX.XXX

Command mode

CONFIG

Example

```
ma4000(config)# no ip source-guard bind ip 192.168.10.5
```

ip source-guard ignore-vlan

This command ignores the specified vlan by the ip source-guard mechanism.

Syntax

```
[no] ip source-guard ignore-vlan <VID>
```

Parameters

<VID> – VLAN ID, may take values [1 .. 4094].

Command mode

CONFIG

Example

```
ma4000(config)# ip source-guard ignore-vlan 2000
```

show slot <SLOT> ip source-guard

This command is used to view the mapping table.

Syntax

```
show slot <SLOT> ip source-guard
```

Parameters

<SLOT> – PLC8 module number, may take values (0..15). You may specify the list of numbers using comma (,) or specify the range using hyphen (-).

Command mode

ROOT

Example

```
ma4000# show slot 0 ip source-guard
IP Source Guard:
  Enabled: true
  Mode: dynamic
  Bind [0]:
    Ip: 192.168.10.5
    Mac: AA:BB:CC:DD:EE:FF
    Interface-ont: 0/0/0
    Service: 2
```

55 PLC8 modules GPON configuration settings

- [gpon olt authentication](#)
- [gpon olt block-duplicated-mac](#)
- [gpon olt broadcast-gem-port](#)
- [gpon olt dba-reduced-latency](#)
- [gpon olt dhcpra-shaper](#)
- [gpon olt encryption](#)
- [gpon olt encryption key-update](#)
- [gpon olt model](#)
- [gpon olt multicast-gem-port](#)
- [gpon olt ont-block-time](#)
- [gpon olt ont-sn-format](#)
- [show gpon olt](#)

gpon olt authentication

The command sets the ONT authentication mode: by password, by serial number, by a combination of serial number and password. The use of a negative form (no) of the command sets the default value.

Syntax

```
gpon olt authentication <VALUE>  
no gpon olt authentication
```

Parameters

<VALUE> – ONT authentication mode:

- password – by password;
- serial – by serial number;
- both – by combination of serial number and password.

Default value:

both – ONT authentication by combination of serial number and password

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt authentication password
```

gpon olt block-duplicated-mac

This command enables protection against duplicate MAC addresses (MAC spoofing).

The use of the negative form (no) disables ONT blocking when duplicated MAC addresses are detected.

Syntax

```
[no] gpon olt block-duplicated-mac
```

Parameters

The command contains no arguments.

Default value:

disable – blocking disabled

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt block-duplicated-mac
```

gpon olt broadcast-gem-port

The command sets the GEM port number for broadcast traffic.

The use of a negative form (no) of the command sets the default value.

Syntax

```
gpon olt broadcast-gem-port <PORT>  
no gpon olt broadcast-gem-port
```

Parameters

<PORT> – number of GEM port for broadcast traffic, may take values [0..4095].

Default value

4095

Command mode

CONFIG

Example

```
ma4000(config)#gpon olt broadcast-gem-port 4095
```

gpon olt dba-reduced-latency

The command enable the mode to lower DBA latency.

The use of a negative form (no) of the command sets the default value.

Syntax

```
[no] gpon olt dba-reduced-latency enable
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt dba-reduced-latency enable
```

gpon olt dhcpra-shaper

This command sets the maximum number of DHCP messages per second (for each ONT)

The use of a negative form (no) of the command sets the default value.

Syntax

```
gpon olt dhcpra-shaper <VALUE>  
no gpon olt dhcpra-shaper
```

Parameters

<VALUE> – maximum number of DHCP messages per second for ONT.

Default value:

100

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt dhcpra-shaper 33
```

gpon olt encryption

The command enables GPON traffic encryption.

The use of the negative form (no) of the command disables GPON traffic encryption.

Syntax

```
[no] gpon olt encryption
```

Parameters

The command contains no arguments.

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt encryption
```

gpon olt encryption key-update

The command is used to set an encryption key change period.

The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
gpon olt encryption key-update <TIME>  
no gpon olt encryption key-update
```

Parameters

<TIME> – encryption key update period, may take values [0..65535], in minutes.

Default value:

1 minute

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt encryption key-update 33333
```

gpon olt model

The command sets the datapath construction model – the order of forming rules for traffic transfer.
The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
gpon olt model <VALUE>
no gpon olt model
```

Parameters

<VALUE> – datapath construction model:

1. transmission of Broadcast and unicast packets in one GEM port, VLAN operation (U-VID assignment) is performed on OLT side;
2. transmission of Broadcast traffic in dedicated GEM port, VLAN operation (U-VID assignment) is performed on OLT side;
3. transmission of Broadcast traffic in dedicated GEM port, VLAN operation (U-VID assignment) is performed on ONT side.

Default value

model1

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt model 1
```

gpon olt multicast-gem-port

The command sets the GEM port number for multicast traffic.
The use of a negative form (no) of the command sets the default parameter value.

Syntax

```
gpon olt multicast-gem-port <PORT>
no gpon olt multicast-gem-port
```

Parameters

<PORT> – number of GEM port for multicast traffic, may take values [0..4095].

Default value

4094

Command mode

CONFIG

Example

```
ma4000(config)# gpon olt multicast-gem-port 32
```

gpon olt ont-block-time

This command sets the time to lock the port when duplicate MAC addresses are detected.
The use of a negative form (no) of the command sets the default value.

Syntax

```
gpon olt ont-block-time <TIME>
no gpon olt ont-block-time
```

Parameters

<TIME> – blocking time, may take values in the range [1..2147483647] minutes.

Default value

5 minutes

Command mode

CONFIG

Example

```
ma4000(config)#gpon olt ont-block-time 10
```

gpon olt ont-sn-format

This command sets the format of displaying the ONT serial number.
The use of a negative form (no) of the command sets the default value.

Syntax

```
gpon olt ont-sn-format <VALUE>
no gpon olt ont-sn-format
```

Parameters

<VALUE> – display format:

- literal – AAAAXXXXXXXXXX, where A is uppercase letter, and X in HEX format [0-F];

- numerical – XXXXXXXXXXXXXXXX, where X in HEX format [0-F];
- section-numerical – XX-XX-XX-XX-XX-XX-XX-XX, where X in HEX format [0-F].

Default value

literal

Command mode

CONFIG

Example

```
ma4000(config)#gpon olt ont-sn-format literal
```

show gpon olt

The command to view the settings of GPON-configuration of PLC8 modules:

- Block duplicated mac – protection against duplicate MAC addresses;
- Ont block time – port blocking time in minutes when detecting duplicate MAC addresses;
- Dhcpna shaper – maximum number of DHCP messages per second for each ONT;
- Datapath – datapath configuration:
- Model – datapath construction model;
- Broadcast gem port – number of GEM port for broadcast traffic;
- Multicast gem port – number of GEM port for multicast traffic;
- Encryption – GPON traffic encryption settings
- Enable – GPON traffic encryption state: enabled, disabled;
- Key update interval – encryption key update period, in minutes;
- ONT authentication mode – ONT authentication mode;
- Auto reconfigure ont – ONT auto reconfiguration state: enabled, disabled;
- Auto reconfigure channel – channel auto reconfiguration state: enabled, disabled;
- Auto reconfigure olt – OLT auto reconfiguration state: enabled, disabled;
- Ont sn format – ONT serial number display format.

Syntax

show gpon olt

Parameters

Command contains no arguments.

Command mode

ROOT

Example

```
ma4000(config)# showgponolt
Block duplicated mac: disabled
  Ont block time: 5
  Dhcpna shaper: 100
  Datapath:
    Model: model1
    Broadcast gem port: 4095
    Multicast gem port: 4094
  Encryption:
    Enable: false
    Key update interval: 1
  ONT authentication mode: both
  Auto reconfigure ont: true
  Auto reconfigure channel: true
  Auto reconfigure olt: true
  Ont sn format: literal
```