MA4000-PX. Configuration and monitoring via SNMP

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.

Operating ONT

Matches between parameter names and digital OIDs are described in the MIB files.

When the number of slot is needed in a command, use the number of needed slot +1 (i.e. when working with the slot 5, use '6' in the commands).

If the command has ONT serial number (dec_serial), this serial number should be specified by 8 decimal digits separated by dots.

Examples:

Serial number	SNMP command parameter
ELTX24A80012	69.76.84.88.36.168.0.18
45-4C-54-58-00-00-00-01	69.76.84.88.0.0.0.1

ONT profiles are specified by the profile index for corresponding OID of the ONT configuration. To know profile index using its name, use the following profile tables:

Profile type	Table
Management	ltp8xONTManagementProfileTable
Ports	ltp8xONTPortsProfileTable
Shaping	ltp8xONTShapingProfileTable
Scripting	ltp8xONTScriptingProfileTable
DBA	ltp8xONTAllocProfileTable
Cross-connect	ltp8xONTCrossConnectProfileTable

To set *unassigned* value, if it is allowed, number 65535 will be transmitted instead of profile index.

Configuring ONT

ONT is configured by the following tables:

- ltp8xONTConfigTable – general parameters;
- ltp8xONTCustomCrossConnectTable – custom parameters;
- ltp8xONTFullServicesConfigTable – Cross-Connect and DBA profiles;
- ltp8xONTSelectiveTunnelTable – selective-tunnel uvids.

Adding ONT

When adding ONT, it is necessary to specify its serial number, gpon-port number and ONT ID.

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTConfigRowStatus.<slot>.8.<dec_serial> i 4
ltp8xONTConfigChannel.<slot>.8.<dec_serial> u <gpon_port>
ltp8xONTConfigID.<slot>.8.<dec_serial> u <ont_id>
```

where:

- <gpon_port> – the real value of the port number.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTConfigRowStatus.15.8.69.76.84.88.36.168.0.18 i 4
ltp8xONTConfigChannel.15.8.69.76.84.88.36.168.0.18 u 6
ltp8xONTConfigID.15.8.69.76.84.88.36.168.0.18 u 0
```

This command creates ONT 14/6/0 with serial number ELTX24A80012.

Editing ONT parameters

ONT General Parameters

General parameters of ONT are configured by using *ltp8xONTConfigTable*.

Command format:

snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid_1>.<slot>.8.<dec_serial> <par1_type> <par1_value>
<parameter_oid_2>.<slot>.8.<dec_serial> <par2_type> <par2_value>
……
<parameter_oid_N>.<slot>.8.<dec_serial> <parN_type> <parN_value>

where:

- <parameter_oid_N> – the names of specific MIB parameters;
- <parN_type> – value type of a parameter;
- <parN_value> – parameter value.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTConfigDescription.15.8.69.76.84.88.36.168.0.18 s "ont_description"
ltp8xONTConfigManagementProfile.15.8.69.76.84.88.36.168.0.18 u 1
ltp8xONTConfigFecUp.15.8.69.76.84.88.36.168.0.18 i 1
```

This command sets the *decription* value, specifies Management profile, and enables error correction for ONT with serial number ELTX24A80012.

Cross-Connect and DBA profiles

Cross-Connect and DBA profiles are configured by using *ltp8xONTFullServicesConfigTable*. Similarly to the custom parameters, additional index (service number) and value increased by 1 should be specified.

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTFullServicesConfigCrossConnectProfile.<slot>.8.<dec_serial>.<service> u <value>

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTFullServicesConfigCrossConnectProfile.15.8.69.76.84.88.36.168.0.18.8 u 1
ltp8xONTFullServicesConfigDBAProfile.15.8.69.76.84.88.36.168.0.18.8 u 2
```

For ONT with serial number ELTX24A80012, this command sets Cross-Connect profile with index 1, and DBA profile with index 2 for service 7.

Custom Cross-Connect parameters

Custom Cross-Connect parameters are configured by using *ltp8xONTCustomCrossConnectTable*. Introduces additional index (service number) including service number increased by 1.

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTCustomCrossConnectEnabled.<slot>.8.<dec_serial>.<service> i <value>

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTCustomCrossConnectEnabled.15.8.69.76.84.88.36.168.0.18.3 i 1
ltp8xONTCustomCrossConnectVID.15.8.69.76.84.88.36.168.0.18.3 i 100
ltp8xONTCustomCrossConnectCOS.15.8.69.76.84.88.36.168.0.18.3 i 1
ltp8xONTCustomCrossConnectSVID.15.8.69.76.84.88.36.168.0.18.3 i 200
```

For ONT with serial number ELTX24A80012, this command activates Custom Cross Connect parameters for service 2 and sets values cvid=100, svid = 200, cos = 1.

Selective-tunnel uvids

Selective-tunnel uvids are configured by using *ltp8xONTSelectiveTunnelTable*. In addition to the service number, an index should be entered – the *selective-tunnel uvid* number. Index takes the values from 1 to 42. The indexes of the assigned *uvid* must follow sequentially without order interruption. For example, if uvid are determined with indexes 1 and 2 the next assigned uvid must have index 3.

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xONTSelectiveTunnelUVID.<slot>.8.<dec_serial>.<service>.<uvid_id> i <SelectiveTunnelUVID_value>

where:

- <SelectiveTunnelUVID_value> – selective-tunnel uvid value.

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xONTSelectiveTunnelUVID.15.8.69.76.84.88.36.168.0.18.8.1 i 200
```

For ONT with serial number ELTX24A80012, this command adds selective-tunnel uvid = 200 for service 7.

Managing PoE on ports

Management of PoE on ONT ports is implemented with the help of *ltp8xOntConfigUniPortTable* table.

Command format:

snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid_1>.1.8.<dec_serial>.<port_number> <par1_type> <par1_value> <parameter_oid_2>.<slot>.8.<dec_serial>.<port_number> <par2_type> <par2_value>
.....
<parameter_oid_N>.<slot>.8.<dec_serial>.<port_number> <parN_type> <parN_value>

To enable PoE on a port use the following command:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xOntConfigUniPortRowStatus.<slot>.8.<dec_serial>.<port_number> i 6

where:

- <parameter_oid_N> – names of the certain parameters in MIB;
- <parN_type> – type of the parameter value;
- <parN_value> – parameter value;
- <port_number> – number of the port.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xOntConfigUniPortRowStatus.15.8.69.76.84.88.130.0.0.2.1 i 4
ltp8xONTPrtsUNIPoeEnable.15.8.69.76.84.88.130.0.0.2.1 i 1
ltp8xONTPortsUNIPoeClassControl.15.8.69.76.84.88.130.0.0.2.1 u 2
ltp8xONTPortsUNIPoePowerPriority.15.8.69.76.84.88.130.0.0.2.1 u 3
```

The command enables PoE on the ONT having the following parameters: ELTX24A80012 serial number, class-control – 2, power-priority – low.

Deleting ONT

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xONTConfigRowStatus.<slot>.8.<dec_serial> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTConfigRowStatus.15.8.69.76.84.88.36.168.0.18 i 6
```

This command deletes ONT configuration with serial number ELTX24A80012.

ONT replacement

You can change ONT by using subsequent commands for deleting and creating configurations with new parameters.

Commands for ONT

Reconfiguration

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTStateReconfigure.<slot>.8.<dec_serial> u 1

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTStateReconfigure.15.8.69.76.84.88.36.168.0.18 u 1
```

Reboot

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTStateReset.<slot>.8.<dec_serial> u 1

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTStateReset.15.8.69.76.84.88.36.168.0.18 u 1
```

Reset ONT to the factory settings

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTStateResetToDefaults.<slot>.8.<dec_serial> u 1

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTStateResetToDefaults.15.8.69.76.84.88.36.168.0.18 u 1
```

Deactivation

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTDisableSlot.0 u <slot>
ltp8xONTDisableONTSerial.0 x <hex_serial>
ltp8xONTDisableChannel.0 u <gpon_port>
ltp8xONTDisableActionDisable.0 u 1

where:

- <hex_serial> – serial number in hex format;
- <gpon_port> – the real value of the port number;
- <slot> – slot number + 1.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTDisableSlot.0 u 15
ltp8xONTDisableONTSerial.0 x "454C545824A80012"
ltp8xONTDisableChannel.0 u 6
ltp8xONTDisableActionDisable.0 u 1
```

The command performs deactivation of ONT ELTX24A80012 on the 6-th channel of the 14-th slot.

Activation

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTDisableSlot.0 u <slot>
ltp8xONTDisableONTSerial.0 x <hex_serial>
ltp8xONTDisableChannel.0 u <ont_channel>
ltp8xONTDisableActionEnable.0 u 1
```

where:

- <hex_serial> – serial number in hex format;
- <gpon_port> – the real value of the port number;
- <slot> – slot number + 1.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTDisableSlot.0 u 15
ltp8xONTDisableONTSerial.0 x "454C545824A80012"
ltp8xONTDisableChannel.0 u 6
ltp8xONTDisableActionEnable.0 u 1
```

The command performs activation of ONT ELTX24A80012 on the 6-th channel of the 14-th slot.

ONT firmware updating via OMCI protocol

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTFirmwareUpdateViaOMCISerial.0 x <hex_serial>
ltp8xONTFirmwareUpdateViaOMCIFilename.0 s <firmware_name>
ltp8xONTFirmwareUpdateViaOMCIAction.0 u 1
```

where:

- <hex_serial> – serial number in hex format;
- <firmware_name> – the file name of the ONT firmware saved on OLT.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTFirmwareUpdateViaOMCISerial.0 x "454C545824A80012"
ltp8xONTFirmwareUpdateViaOMCIFilename.0 s "ntp-rg-revc-3.24.3.41.fw.bin"
ltp8xONTFirmwareUpdateViaOMCIAction.0 u 1
```

The command launches update of ONTELTX24A80012 using "ntp-rg-revc-3.24.3.41.fw.bin" file.

Requests

Common ONT state

The parameters of ONT state can be requested by using *ltp8xONTStateTable*.

Command format:

snmpget -v2c -c <ro_community> <ipaddr> <parameter_oid>.<slot>.8.<dec_serial>

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xONTStateState.15.8.69.76.84.88.36.168.0.18
ltp8xONTStateVersion.15.8.69.76.84.88.36.168.0.18
```

The command requests the status and firmware version for the ONT ELTX24A80012.

PPP session state

To obtain the session list for selected ONT as the list of client MAC addresses, use the following request:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr>
ltp8xOLTPPPoESessionsClientMac.<slot>.<gpon_port>.<ont_id>

where:

- <gpon_port> – port number + 1.

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xOLTPPPoESessionsClientMac.15.7.0
```

The command requests the session list as the list of clients' MAC addresses for ONT 14/6/0.

To request information about a specific PPPoE session, specify gpon port, ONT ID and client MAC address.

Command format:

snmpget -v2c -c <ro_community> <ipaddr>
ltp8xOLTPPPoESessionsPort.<slot>.<gpon_port>.<ont_id>.<dec_client_mac>
ltp8xOLTPPPoESessionsSessionID.<slot>.<gpon_port>.<ont_id>.<dec_client_mac>
ltp8xOLTPPPoESessionsDuration.<slot>.<gpon_port>.<ont_id>.<dec_client_mac>
ltp8xOLTPPPoESessionsUnblock.<slot>.<gpon_port>.<ont_id>.<dec_client_mac>
ltp8xOLTPPPoESessionsSerial.<slot>.<gpon_port>.<ont_id>.<dec_client_mac>

where:

- <gpon_port> – port number + 1;
- <Dec_client_mac> – a client MAC address in decimal notation.

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xOLTPPPoESessionsPort.15.7.0.152.222.208.0.205.252
ltp8xOLTPPPoESessionsSessionID.15.7.0.152.222.208.0.205.252
ltp8xOLTPPPoESessionsDuration.15.7.0.152.222.208.0.205.252
ltp8xOLTPPPoESessionsUnblock.15.7.0.152.222.208.0.205.252
ltp8xOLTPPPoESessionsSerial.15.7.0.152.222.208.0.205.252
```

The command requests PPP session parameters with client MAC address 98:de:d0:00:cd:fc ONT 14/6/0.

MAC table

Ltp8xONTAddressTable is used. To obtain the record list of the selected ONT, use the following request:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr>
ltp8xONTAddressEntryID.<slot>.8.<dec_serial>

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTAddressEntryID.15.8.69.76.84.88.36.168.0.18
```

The command requests MAC table record list of ONT ELTX24A80012.

To request MAC address table, specify ONT serial number and entry ID in the address table of the current ONT.

Command format:

**snmpget -v2c -c <ro_community><ipaddr>
<parameter_oid>.<slot>.8.<dec_serial>.<entry_id>**

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xONTAddressMacAddress.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTAddressCVID.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTAddressSVID.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTAddressUVID.15.8.69.76.84.88.36.168.0.18.1
```

The command requests MAC address, CVID, SVID, UVID for the first record in the MAC address table of ONT ELTX24A80012.

IGMP group table

ltp8xONTMulticastStatsTable is used. To obtain the record list of the selected ONT, use the following request:

Command format:

**snmpwalk -v2c -c <ro_community> <ipaddr>
ltp8xONTMulticastStatsRecordID.<slot>.8.<dec_serial>**

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTMulticastStatsRecordID.15.8.69.76.84.88.36.168.0.18
```

The command requests the record list of the IGMP group table for ONT ELTX24A80012.

To request the records of the IGMP group table, specify ONT serial number and entry ID in the IGMP-group table of the current ONT.

Command format:

**snmpget -v2c -c <ro_community> <ipaddr>
<parameter_oid>.<slot>.8.<dec_serial>.<entry_id>**

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xONTMulticastStatsMulticastAddress.15.8.69.76.84.88.36.168.0.18.153
ltp8xONTMulticastStatsStop.15.8.69.76.84.88.36.168.0.18.153
ltp8xONTMulticastStatsStart.15.8.69.76.84.88.36.168.0.18.153
```

The command requests the IP address of the group, the broadcasting start and stop time for the 153rd entry in IGMP group table of the ONT ELTX24A80012 from the 14-th slot.

ONT connection log

Command format:

**snmpget -v2c -c <ro_community> <ipaddr>
ltp8xONTConnectionLogText.<slot>.8.<dec_serial>**

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xONTConnectionLogText.15.8.69.76.84.88.36.168.0.18
```

The command requests the connection log for ONT ELTX24A80012 from the 14-th slot.

ONT counters

To obtain information about counters, use *ltp8xONTStatistics* table.

Counters for Cross-connect

Matches between counters and OIDs are given in the table below.

Table 1 – Matches between ONT and OID counters

Counter	OID	Description
CrossConnect in the downstream direction	ltp8xONTCrossConnectDSCounterName	CrossConnect counter name
	ltp8xONTCrossConnectDSCounterValue	CrossConnect counter value
CrossConnect in the upstream direction	ltp8xONTCrossConnectUSCounterName	CrossConnect counter name
	ltp8xONTCrossConnectUSCounterValue	CrossConnect counter value
GEM in the downstream direction	ltp8xONTGEMPortPerformMonitoringDSCounterName	GEM counter name
	ltp8xONTGEMPortPerformMonitoringDSCounterValue	GEM counter value
GEM in the upstream direction	ltp8xONTGEMPortPerformMonitoringUSCounterName	GEM counter name
	ltp8xONTGEMPortPerformMonitoringUSCounterValue	GEM counter value
GAL	ltp8xONTGalEthPerformMonitoringHistDataCounterName	GAL counter name
	ltp8xONTGalEthPerformMonitoringHistDataCounterValue	GAL counter value
FEC	ltp8xONTFecPerformMonitoringHistDataCounterName	FEC counter name
	ltp8xONTFecPerformMonitoringHistDataCounterValue	FEC counter value

Command format:

```
snmpwalk -v2c -c <ro_community> <ipaddr>
<parameter_oid>.<slot>.8.<dec_serial>.<cross_connect_id>
```

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTCrossConnectDSCounterName.15.8.69.76.84.88.36.168.0.18.1
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTCrossConnectDSCounterValue.15.8.69.76.84.88.36.168.0.18.1
```

The command requests the name list of Cross-Connect counters for the downstream direction and their values for service 1 of ONT ELTX24A80012.

ETH ports state

To obtain information about ports, use *ltp8xONTUNIPortsStateTable*.

Command format:

```
snmpget -v2c -c <ro_community> <ipaddr>
ltp8xONTUNIPortsStateAvailable.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStateLinkUp.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStateSpeed.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStateDuplex.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStatePoEEnabled.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStatePoEPowerDetectionStatus.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStatePoEPowerClassificationStatus.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStatePoEPSEClassControl.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStatePoEPSEPowerPriority.<slot>.8.<dec_serial>.<eth_port_id>
ltp8xONTUNIPortsStatePoECurrentPowerConsumption.<slot>.8.<dec_serial>.<eth_port_id>
```

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xONTUNIPortsStateAvailable.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStateLinkUp.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStateSpeed.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStateDuplex.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStatePoEEnabled.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStatePoEPowerDetectionStatus.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStatePoEPowerClassificationStatus.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStatePoEPSEClassControl.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStatePoEPSEPowerPriority.15.8.69.76.84.88.36.168.0.18.1
ltp8xONTUNIPortsStatePoECurrentPowerConsumption.15.8.69.76.84.88.36.168.0.18.1
```

The command requests the status of the 1st ETH port for the ONT ELTX24A80012.

ETH port counters

Matches between counters and OIDs are given in the table below.

Table 2 – Matches between ETH port and OID counters

Counter	OID	Description
ETH extended in the downstream direction	ltp8xONTEthFrameExtendedPerformMonitoringDSCounterName	ETH extended counter name
	ltp8xONTEthFrameExtendedPerformMonitoringDSCounterValue	ETH extended counter value
ETH extended in upstream direction	ltp8xONTEthFrameExtendedPerformMonitoringUSCounterName	ETH extended counter name
	ltp8xONTEthFrameExtendedPerformMonitoringUSCounterValue	ETH extended counter value
ETH	ltp8xONTEthPerformMonitoringHistDataCounterName	ETH counter name
	ltp8xONTEthPerformMonitoringHistDataCounterValue	ETH counter value

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> <parameter_oid>.<slot>.8.<dec_serial>.<eth_port_id>

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTEthFrameExtendedPerformMonitoringDSCounterName.15.8.69.76.84.88.36.168.0.18.1
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTEthFrameExtendedPerformMonitoringDSCounterValue.15.8.69.76.84.88.36.168.0.18.1
```

The command requests the name list of ETH extended counters for the downstream direction and their values for the first ETH port of ONT ELTX24A80012.

Counter reset on ONT

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xONTResetCountersAction.<slot>.8.<dec_serial> u 1

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTResetCountersAction.15.8.69.76.84.88.36.168.0.18 u 1
```

The command resets ONT ELTX24A80012 counters on the 14-th slot.

GPON port counter reset

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xPONChannelResetCounters.<slot>.<gpon_port> u 1

where:

- <gpon_port> – port number +1;
- <slot> – slot number +1.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xPONChannelResetCounters.15.7 u 1
```

The command resets the GPON port 14/6 counters.

OLT configuration

Applying and confirming OLT configuration

Before first commit/confirm operation it is necessary to commit the following request:

Command format:

snmpset -v2c -c <rw_community> <ipaddr> eltexRowStatus.100 i 4

Example:

```
snmpset -v2c -c private 192.168.0.1 eltexRowStatus.100 i 4
```

The request should be repeated after the device reboot or after changing pp4x master.

Commit

Command format:

snmpset -v2c -c <rw_community> -t 20 <ipaddr> eltexSourceFileType.100 i 6 eltexSourceLocation.100 i 8 eltexSourceFileName.100 s "candidate" eltexDestinationFileType.100 i 5 eltexDestinationLocation.100 i 8 eltexDestinationFileName.100 s "running"

Example:

```
snmpset -v2c -c <rw_community> -t 20 <ipaddr> eltexSourceFileType.100 i 6
eltexSourceLocation.100 i 8 eltexSourceFileName.100 s "candidate"
eltexDestinationFileType.100 i 5 eltexDestinationLocation.100 i 8
eltexDestinationFileName.100 s "running"
```

Confirm

Command format:

snmpset -v2c -c <rw_community> -t 20 <ipaddr> eltexSourceFileType.100 i 5 eltexSourceLocation.100 i 8 eltexSourceFileName.100 s "running" eltexDestinationFileType.100 i 9 eltexDestinationLocation.100 i 8 eltexDestinationFileName.100 s "confirm"

Example:

```
snmpset -v2c -c private -t 20 192.168.0.1 eltexSourceFileType.100 i 5
eltexSourceLocation.100 i 8 eltexSourceFileName.100 s "running"
eltexDestinationFileType.100 i 9 eltexDestinationLocation.100 i 8
eltexDestinationFileName.100 s "confirm"
```

VLAN configuration (PP4X/PLC8)

PP4X

dot1qVlanStaticTable is used to manage VLAN PP4X.

Adding VLAN (for PP4X)

When adding a new VLAN it is necessary to specify its name and set of ports-members of the VLAN.

Ports membership in VLAN is specified by using 3 bitmasks, in which value 1 in an N-bit (starting from the highest to lowest) means including the port, which corresponds ifindex=N, to the set. For insertion to the *snmpset* command, bitmasks are converted into hex-form.

There are three sets:

- EgressPorts – ports, which are members of the VLAN;
- ForbiddenEgressPorts – ports, which are not members of the VLAN;
- UntaggedPorts – if port-member of the VLAN is incleded to this set, it means that the port is untagged; if the port is out of this set — it is tagged.

According to the structure of ifTable indexes in MA4000, the mask which includes all the possible ports and lacks port-channel has the following form:

- AAA0AAA0AAA022AAAA0AAA0AAA022A00FFFF00.

If port-channel presents in the configuration, the 16-th byte in the mask will be changed. For instance, if there is port-channel 1 and 2 the mask will be as follows:

- AAA0AAA0AAA022AAAA0AAA0AAA022A0FFFF00.

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
dot1qVlanStaticRowStatus.<vlan_id> i 4
dot1qVlanStaticName.<vlan_id> s "<vlan_name>"
dot1qVlanStaticEgressPorts.<vlan_id> x "<ports_mask>"
dot1qVlanForbiddenEgressPorts.<vlan_id> x "<ports_mask>"
dot1qVlanStaticUntaggedPorts.<vlan_id> x "<ports_mask>"
```

Example:

```
snmpset -v2c -c private 192.168.0.1
dot1qVlanStaticRowStatus.72 i 4
dot1qVlanStaticName.72 s "created_by_snmp"
dot1qVlanStaticEgressPorts.72 x "00000000000000002800000000000000000000"
dot1qVlanForbiddenEgressPorts.72 x "AAA0AAA0AAA022AA820AAA0AAA022A00FFFF00"
dot1qVlanStaticUntaggedPorts.72 x "AAA0AAA0AAA022AAA20AAA0AAA022A00FFFF00"
```

The command creates VLAN 72 named "created_by_snmp", VLAN includes frontport 2/3 (untagged) and front-port 2/4(tagged).

Editing VLAN (for PP4X)

The command for VLAN editing is similar to the one for VLAN creation. However, in the command for VLAN editing, you should not specify *dot1qVlanStaticRowStatus* parameter.

Deleting VLAN (for PP4X)

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
dot1qVlanStaticRowStatus.<vlan_id> i 6
```

Example:

```
snmpset -v2c -c private 192.168.0.1
dot1qVlanStaticRowStatus.72 i 6
```

The command deletes VLAN 72 from the PP4X configuration.

Aquiring VLAN list and certain VLAN configuration (for PP4X)

To aquire a VLAN list, you should implement the following request:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> dot1qVlanStaticName

Example:

```
snmpwalk -v2c -c public 192.168.0.1 dot1qVlanStaticName
```

The command shows the VLAN list for PP4X.

To aquire the configuration of a certain VLAN, use the following command:

Command format:

**snmpget -v2c -c <ro_community> <ipaddr> dot1qVlanStaticName.<vid>
dot1qVlanStaticEgressPorts.<vid> dot1qVlanForbiddenEgressPorts.<vid>
dot1qVlanStaticUntaggedPorts.<vid>**

Example:

```
snmpget -v2c -c public 192.168.0.1 dot1qVlanStaticName.72
dot1qVlanStaticEgressPorts.72 dot1qVlanForbiddenEgressPorts.72
dot1qVlanStaticUntaggedPorts.72
```

The command shows the configuration of the certain VLAN for PP4X.

PLC

For managing VLAN PLC8, *ltp8xSwitchVLANTable is used*.

Adding VLAN (for PLC)

Command format:

**snmpset -v2c -c <rw_community> <ipaddr>
ltp8xSwitchVLANRowStatus.<slot>.<vid> i 4**

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xSwitchVLANRowStatus.15.156 i 4
```

The command creates VLAN 156 on the 14-th PLC slot.

Editing VLAN (for PLC)

Ports membership in VLAN is specified by using 2 bitmasks, in which value 1 in an N-bit (starting from the highest to lowest) means including the port with index N to the set. The index allocation is present in *ltp8xSwitchPortsTable.* For insertion to the *snmpset* command, bitmasks are converted into hex-form.

There are two sets: *TaggedPorts* and *UntaggedPorts*.

Command format:

**snmpset -v2c -c <rw_community> <ipaddr>
ltp8xSwitchVLANName.<slot>.<vid> s "vlan_name"
ltp8xSwitchVLANTaggedPorts.<slot>.<vid> x "<tagged_ports_mask>"
ltp8xSwitchVLANUntaggedPorts.<slot>.<vid> x "<utagged_ports_mask>"
ltp8xSwitchVLANIGMPSnoopingEnabled.<slot>.<vid> i 1/2
ltp8xSwitchVLANMLDSnoopingEnabled.<slot>.<vid> i 1/2**

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xSwitchVLANName.15.156 s "edited_by_snmp"
ltp8xSwitchVLANTaggedPorts.15.156 x "40000000"
ltp8xSwitchVLANUntaggedPorts.15.156 x "20000000"
ltp8xSwitchVLANIGMPSnoopingEnabled.15.156 i 1
ltp8xSwitchVLANMLDSnoopingEnabled.15.156 i 2
```

The command sets "edited_by_snmp" name to VLAN 156 on the 14-th PLC slot, adds pon-port 1 tagged, pon-port 2 untagged and enables IGMP snooping.

Deleting VLAN (for PLC)

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xSwitchVLANRowStatus.<slot>.<vid> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xSwitchVLANRowStatus.15.156 i 6
```

The command deletes VLAN 156 from the configuration of the 14-th PLC8 slot.

Aquiring VLAN list and certain VLAN configuration (for PLC)

To aquire a VLAN list, you should implement the following request:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> ltp8xSwitchVLANName.<slot>

Example:

```
snmpwalk -v2c -c public 192.168.0.1 ltp8xSwitchVLANName.15
```

The command shows the VLAN list for the 14-th PLC8 slot.

To aquire the configuration of a certain VLAN, use the following command:

Command format:

snmpget -v2c -c <ro_community> <ipaddr> ltp8xSwitchVLANName.<slot>.<vid>
ltp8xSwitchVLANTaggedPorts.<slot>.<vid>
ltp8xSwitchVLANUntaggedPorts.<slot>.<vid>
ltp8xSwitchVLANIGMPSnoopingEnabled.<slot>.<vid>
ltp8xSwitchVLANIGMPSnoopingQuerierEnabled.<slot>.<vid>
ltp8xSwitchVLANMLDSnoopingEnabled.<slot>.<vid>
ltp8xSwitchVLANMLDSnoopingQuerierEnabled.<slot>.<vid>

Example:

```
snmpget -v2c -c public 192.168.0.1 ltp8xSwitchVLANName.15.156
ltp8xSwitchVLANTaggedPorts.15.156 ltp8xSwitchVLANUntaggedPorts.15.156
ltp8xSwitchVLANIGMPSnoopingEnabled.15.156
ltp8xSwitchVLANIGMPSnoopingQuerierEnabled.15.156
ltp8xSwitchVLANMLDSnoopingEnabled.15.156
ltp8xSwitchVLANMLDSnoopingQuerierEnabled.15.156
```

The command shows the configuration of the certain VLAN for the 14-th PLC8 slot.

Terminal VLAN Configuration

Adding terminal VLAN

Use *ltp8xOLTTerminalVLANsNamesTable* to add Terminal VLAN.

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xOLTTerminalVLANsNamesRowStatus.<t_vlan_id> i 4
ltp8xOLTTerminalVLANsNamesName.<t_vlan_id> s "<t_vlan_name>"

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xOLTTerminalVLANsNamesRowStatus.1 i 4
ltp8xOLTTerminalVLANsNamesName.1 s "created_by_snmp"
```

The command creates Terminal VLAN with index 1 and name 'created_by_snmp'.

Editing terminal VLAN

Use *ltp8xOLTTerminalVLANsTable* to edit Terminal VLAN parameters.

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xOLTTerminalVLANsVID.1.<t_vlan_id> u <vlan_id>
ltp8xOLTTerminalVLANsCOS.1.<t_vlan_id> i <cos>

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xOLTTerminalVLANsVID.1.1 u 80
ltp8xOLTTerminalVLANsCOS.1.1 i 255
```

The command sets the Terminal VLAN with index 1 to vlan_id = 80, cos = unused.

Deleting terminal VLAN

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xOLTTerminalVLANsNamesRowStatus.<t_vlan_id> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xOLTTerminalVLANsNamesRowStatus.1 i 6
```

The command deletes Terminal VLAN with index 1 from the configuration.

Requesting the Terminal VLAN list and configuration of the specific Terminal VLAN

To get the Terminal VLAN list, use the following request:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr>
ltp8xOLTTerminalVLANsNamesName

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xOLTTerminalVLANsNamesName
```

The command lists Terminal VLAN

To obtain configuration of the specific VLAN, you can use the following request:

Command format:

snmpget -v2c -c <ro_community> <ipaddr>
ltp8xOLTTerminalVLANsName.<t_vlan_id>
ltp8xOLTTerminalVLANsVID.<t_vlan_id>
ltp8xOLTTerminalVLANsCOS.<t_vlan_id>

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xOLTTerminalVLANsName.15.1
ltp8xOLTTerminalVLANsVID.15.1
ltp8xOLTTerminalVLANsCOS.15.1
```

The command displays the configuration of a specific Terminal VLAN for the 14-th slot of PLC8.

IGMP/MLD Configuration

Global settings for enabling IGMP/MLD Snooping, Proxy Report

To configure IGMP/MLD on PP4X, use *ltp8xSwitchIGMPSnoopingTable*. For configuring on slots, use *ltp8xSwitchIGMPSnoopingTable* and *ltp8xIGMPProxyReportTable.* OIDs for configuring global parameters are listed in the table below:

Table 3 – OID compliance for global settings

Parameter	OID	Description
PP4X		
IGMP Snooping	pp4IGMPSnoopingEnable	Possible values: 1 – Enable 2 – Disable
IGMP Proxy Report	pp4IGMPProxyReportEnable	
MLD Snooping	pp4MLDSnoopingEnable	
MLD Proxy Report	pp4MLDProxyReportEnable	
PLC8		
IGMP Snooping	ltp8xSwitchIGMPSnoopingEnabled	Possible values: 1 – Enable 2 – Disable
IGMP Proxy Report	ltp8xIGMPProxyReportEnabled	
MLD Proxy Report	ltp8xMLDProxyReportEnabled	

Command format:

`snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid>.<slot> i <value>`

where <slot> has the value: 0 – for PP4X, 1..16 – for PLC8.

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4SwitchIGMPSnoopingEnabled.0 i 1
```

The command enables IGMP Snooping on PP4X.

IGMP Snooping and VLAN Querier configuration

PP4X

Parameter	OID	Description
IGMP Snooping	pp4IGMPSnoopingVLANEnabled	Available values: 1 — Enable 2 — Disable
IGMP Querier	pp4IGMPSnoopingVLANQuerierEnabled	
MLD Snooping	pp4MLDSnoopingVLANEnabled	
MLD Querier	pp4MLDSnoopingVLANQuerierEnabled	

Command format:

`snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid>.<vid> i <value>`

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4IGMPSnoopingVLANEnabled.86 i 1
```

The command enables IGMP Snooping for VLAN 86 on PP4X.

PLC

The configuration is similar to the one described in Editing VLAN.

Adding/deleting IGMP/MLD Proxy Report Range

PP4X

To configure Proxy Report Range on PP4X the following tables are used: *pp4IGMPProxyReportRangesTable, pp4MLDProxyReportRangesTable*.

Adding IGMP Proxy Report Range (for PP4X) :

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4IGMPProxyRowStatus.<range_id> i 4
pp4IGMPProxyReportRangesStart.<range_id> a <range_start>
pp4IGMPProxyReportRangesEnd.<range_id> a <range_end>
pp4IGMPProxyReportRangesFromVLAN.<range_id> u <from_vlan>
pp4IGMPProxyReportRangesToVLAN.<range_id> u <to_vlan>

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4IGMPProxyRowStatus.1 i 4
pp4IGMPProxyReportRangesStart.1 a 233.0.0.1
pp4IGMPProxyReportRangesEnd.1 a 233.0.0.255
pp4IGMPProxyReportRangesFromVLAN.1 u 5
pp4IGMPProxyReportRangesToVLAN.1 u 6
```

The command adds a new IGMP Proxy Report Range entry to PP4X with the following parameters:

- id=1;
- ip_start: 233.0.0.1;
- ip_end: 233.0.0.255;
- from_vlan: 5;
- to_vlan: 6.

Viewing ID of already added IGMP Proxy Report Range entries (for PP4X) :

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> pp4IGMPProxyReportRangesID

Example:

```
snmpwalk -v2c -c public 192.168.0.1 pp4IGMPProxyReportRangesID
```

Deleting IGMP Proxy Report Range (for PP4X):

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4IGMPProxyRowStatus.<range_id> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4IGMPProxyRowStatus.1 i 6
```

The command deletes IGMP Proxy Report Range entry having id=1 on PP4X.

Adding MLD Proxy Report Range (for PP4X) :

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4MLDProxyRowStatus.<range_id> i 4
pp4MLDProxyReportRangesStart.<range_id> s <range_start>
pp4MLDProxyReportRangesEnd.<range_id> s <range_end>
pp4MLDProxyReportRangesFromVLAN.<range_id> u <from_vlan>
pp4MLDProxyReportRangesToVLAN.<range_id> u <to_vlan>

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4MLDProxyRowStatus.1 i 4
pp4MLDProxyReportRangesStart.1 s FF15:0:0:0:0:0:0:1
pp4MLDProxyReportRangesEnd.1 s FF15:0:0:0:0:0:0:FFFF
pp4MLDProxyReportRangesFromVLAN.1 u 7
pp4MLDProxyReportRangesToVLAN.1 u 8
```

The command adds a new MLD Proxy Report Range entry to PP4X with the following parameters:

- id=1;
- ip_start: FF15::1;
- ip_end: FF15::FFFF;
- from_vlan: 7;
- to_vlan: 8.

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4MLDProxyRowStatus.<range_id> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4MLDProxyRowStatus.1 i 6
```

the command deletes MLD Proxy Report Range entry with id=1 on PP4X.

PLC

To configure Proxy Report Range on PLC8 slots, the following tables are used *ltp8xIGMPProxyReportRangesTable, ltp8xMLDProxyReportRangesTable*.

Adding IGMP Proxy Report Range (for PLC):

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xIGMPProxyRowStatus.<slot>.<range_id> i 4
ltp8xIGMPProxyReportRangesStart.<slot>.<range_id> a <range_start>
ltp8xIGMPProxyReportRangesEnd.<slot>.<range_id> a <range_end>
ltp8xIGMPProxyReportRangesFromVLAN.<slot>.<range_id> i <from_vlan>
ltp8xIGMPProxyReportRangesToVLAN.<slot>.<range_id> i <to_vlan>

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xIGMPProxyRowStatus.15.1 i 4
ltp8xIGMPProxyReportRangesStart.15.1 a 235.0.0.1
ltp8xIGMPProxyReportRangesEnd.15.1 a 235.0.0.100
ltp8xIGMPProxyReportRangesFromVLAN.15.1 i 55
ltp8xIGMPProxyReportRangesToVLAN.15.1 i 56
```

The command adds a new IGMP Proxy Report Range entry for the 14-th slot with the following parameters:

- id=1;
- ip_start: 233.0.0.1;
- ip_end: 233.0.0.100;
- from_vlan: 55;
- to_vlan: 56.

Deleting IGMP Proxy Report Range (for PLC) :

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xIGMPProxyRowStatus.<slot>.<range_id> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xIGMPProxyRowStatus.15.1 i 6
```

The commands deletes the entry having id=1 from the 14-th slot.

Adding MLD Proxy Report Range (for PLC):

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp8xMLDProxyRowStatus.<slot>.<range_id> i 4
ltp8xMLDProxyReportRangesStart.<slot>.<range_id> a <range_start>
ltp8xMLDProxyReportRangesEnd.<slot>.<range_id> a <range_end>
ltp8xMLDProxyReportRangesFromVLAN.<slot>.<range_id> u <from_vlan>
ltp8xMLDProxyReportRangesToVLAN.<slot>.<range_id> u <to_vlan>
```

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xMLDProxyRowStatus.15.1 i 4
ltp8xMLDProxyReportRangesStart.15.1 s FF15:0:0:0:0:0:0:1
ltp8xMLDProxyReportRangesEnd.15.1 s FF15:0:0:0:0:0:0:100
ltp8xMLDProxyReportRangesFromVLAN.15.1 i 57
ltp8xMLDProxyReportRangesToVLAN.15.1 i 58
```

The command creates a new MLD Proxy Report Range entry for the 14-th slot with the following parameters:

- id=1;
- ip_start: 1;
- ip_end: 100;
- from_vlan: 57;
- to_vlan: 58.

Deleting MLD Proxy Report Range (for PLC):

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp8xMLDProxyRowStatus.<slot>.<range_id> i 6
```

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xMLDProxyRowStatus.15.1 i 6
```

The command deletes the MLD Proxy Report Range entry having id=1 from the 14-th slot.

Globally

Proxy Report Range global configuration for all PLC8 slots is implemented with the help of the following tables: *ltp8xIGMPProxyReportRangesGlobalTable, ltp8xMLDProxyReportRangesGlobalTable*.

Adding IGMP Proxy Report Range (global settings):

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp8xIGMPProxyGlobalRowStatus.<range_id> i 4
ltp8xIGMPProxyReportRangesGlobalStart.<range_id> a <range_start>
ltp8xIGMPProxyReportRangesGlobalEnd.<range_id> a <range_end>
ltp8xIGMPProxyReportRangesGlobalFromVLAN.<range_id> i <from_vlan>
ltp8xIGMPProxyReportRangesGlobalToVLAN.<range_id> i <to_vlan>
```

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xIGMPProxyGlobalRowStatus.1 i 4
ltp8xIGMPProxyReportRangesGlobalStart.1 a 238.0.0.100
ltp8xIGMPProxyReportRangesGlobalEnd.1 a 238.0.0.150
ltp8xIGMPProxyReportRangesGlobalFromVLAN.1 i 107
ltp8xIGMPProxyReportRangesGlobalToVLAN.1 i 108
```

The command creates a new global IGMP Proxy Report Range entry, with the following parameters:

- id=1;
- ip_start: 233.0.0.100;
- ip_end: 233.0.0.150;
- from_vlan: 107;
- to_vlan: 108.

Deleting IGMP Proxy Report Range (global settings):

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xIGMPProxyGlobalRowStatus.<range_id> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xIGMPProxyGlobalRowStatus.1 i 6
```

The command deletes the global IGMP Proxy Report Range entry having id=1.

Adding MLD Proxy Report Range (global settings):

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xMLDProxyGlobalRowStatus.<range_id> i 4
ltp8xMLDProxyReportRangesGlobalStart.<range_id> s <range_start>
ltp8xMLDProxyReportRangesGlobalEnd.<range_id> s <range_end>
ltp8xMLDProxyReportRangesGlobalFromVLAN.<range_id> i <from_vlan>
ltp8xMLDProxyReportRangesGlobalToVLAN.<range_id> i <to_vlan>

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xMLDProxyGlobalRowStatus.1 i 4
ltp8xMLDProxyReportRangesGlobalStart.1 s FF15:0:0:0:0:0:0:100
ltp8xMLDProxyReportRangesGlobalEnd.1 s FF15:0:0:0:0:0:0:200
ltp8xMLDProxyReportRangesGlobalFromVLAN.1 i 200
ltp8xMLDProxyReportRangesGlobalToVLAN.1 i 300
```

The command creates a new MLD Proxy Report Range entry with the following parameters:

- id=1;
- ip_start: FF15::100;
- ip_end: FF15::200;
- from_vlan: 200;
- to_vlan: 300.

Deleting MLD Proxy Report Range (global settings):

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xMLDProxyGlobalRowStatus.<range_id> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xMLDProxyGlobalRowStatus.1 i 6
```

The command deletes the global MLD Proxy Report Range entry having id=1.

Aquiring current configuration on IGMP/MLD Proxy Report Range

PP4X

IGMP report range (for PP4X):

Command format:

```
snmpget -v2c -c <ro_community> <ipaddr>
pp4IGMPProxyReportRangesStart.<range_id>
pp4IGMPProxyReportRangesEnd.<range_id>
pp4IGMPProxyReportRangesFromVLAN.<range_id>
pp4IGMPProxyReportRangesToVLAN.<range_id>
```

MLD report range (for PP4X):

Command format:

```
snmpget -v2c -c <ro_community> <ipaddr>
pp4MLDProxyReportRangesStart.<range_id>
pp4MLDProxyReportRangesEnd.<range_id>
pp4MLDProxyReportRangesFromVLAN.<range_id>
pp4MLDProxyReportRangesToVLAN.<range_id>
```

Example:

```
snmpget -v2c -c public 192.168.0.1 pp4MLDProxyReportRangesStart.1
pp4MLDProxyReportRangesEnd.1 pp4MLDProxyReportRangesFromVLAN.1
pp4MLDProxyReportRangesToVLAN.1
```

The command displays the parameters of MLD Proxy Report Range having id=1 on PP4X.

PLC

IGMP proxy report range (for PLC):

Command format:

```
snmpget -v2c -c <ro_community> <ipaddr>
ltp8xIGMPProxyReportRangesStart.<slot>.<range_id>
ltp8xIGMPProxyReportRangesEnd.<slot>.<range_id>
ltp8xIGMPProxyReportRangesFromVLAN.<slot>.<range_id>
ltp8xIGMPProxyReportRangesToVLAN.<slot>.<range_id>
```

MLD proxy report range (for PLC):

Command format:

```
snmpget -v2c -c <ro_community> <ipaddr>
ltp8xMLDProxyReportRangesStart.<slot>.<range_id>
ltp8xMLDProxyReportRangesEnd.<slot>.<range_id>
ltp8xMLDProxyReportRangesFromVLAN.<slot>.<range_id>
ltp8xMLDProxyReportRangesToVLAN.<slot>.<range_id>
```

Example:

```
snmpget -v2c -c public 192.168.0.1 ltp8xIGMPProxyReportRangesStart.15.1
ltp8xIGMPProxyReportRangesEnd.15.1
ltp8xIGMPProxyReportRangesFromVLAN.15.1
ltp8xIGMPProxyReportRangesToVLAN.15.1
```

The command displays the parameters of IGMP Proxy Report Range having id=1 on the 14-th slot.

Globally

IGMP proxy report range (global settings):

Command format:

```
snmpget -v2c -c <ro_community> <ipaddr>
ltp8xIGMPProxyReportRangesGlobalStart.<range_id>
ltp8xIGMPProxyReportRangesGlobalEnd.<range_id>
ltp8xIGMPProxyReportRangesGlobalFromVLAN.<range_id>
ltp8xIGMPProxyReportRangesGlobalToVLAN.<range_id>
```

MLD proxy report range (global settings):

Command format:

```
snmpget -v2c -c <ro_community> <ipaddr>
ltp8xMLDProxyReportRangesGlobalStart.<range_id>
ltp8xMLDProxyReportRangesEnd.<slot>.<range_id>
ltp8xMLDProxyReportRangesFromVLAN.<slot>.<range_id>
ltp8xMLDProxyReportRangesToVLAN.<slot>.<range_id>
```

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xMLDProxyReportRangesGlobalStart.1
ltp8xMLDProxyReportRangesGlobalEnd.1
ltp8xMLDProxyReportRangesGlobalFromVLAN.1
ltp8xMLDProxyReportRangesGlobalToVLAN.1
```

The command displays the global parameters of MLD Proxy Report Range with id=1.

Configuring Cross-connect, DBA, Ports profiles

Cross-connect profiles

The configuration of cross-connect profiles is implemented with the help of *ltp8xONTCrossConnectProfileTable*.

Adding cross-connect profiles

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTCrossConnectRowStatus.<profile_index> i 4
```

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTCrossConnectRowStatus.2 i 4
```

The command adds Cross-connect profile with the index 2.

Editing cross-connect profiles

Cross-connect profiles have a peculiarity – if you need to specify the link to terminal-vlan as vlan-id, the value from -100 (terminal-vlan-0) to -131 (terminal-vlan-31) will be transmitted.

Command format:

```
snmpset -v2c -c <rw_community> <ipaddr>
<parameter_oid_1>.<profile_id> <par1_type> <par1_value>
<parameter_oid_2>.<profile_id> <par2_type> <par2_value>
…
<parameter_oid_N>.<profile_id> <parN_type> <parN_value>
```

where:

- <parameter_oid_N> – names of the specified parameters in MIB;
- <profile_id> – profile index;
- <parN_type> – type of the parameter value;
- <parN_value> – parameter value.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTCrossConnectName.2 s "edited_by_snmp"
ltp8xONTCrossConnectModel.2 i 1
ltp8xONTCrossConnectBridgeGroup.2 u 5
ltp8xONTCrossConnectUVID.2 i -101
```

The command sets name – "edited_by_snmp", type – "ont", bridge group = 5 and UVID which links to the terminal-vlan-1 for the Cross-connect profile having index 2.

Deleting cross-connect profiles

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTCrossConnectRowStatus.<profile_index> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTCrossConnectRowStatus.2 i 6
```

The command deletes Cross-connect profile having index 2.

Aquiring cross-connect profiles list

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr>
ltp8xONTCrossConnectName

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTCrossConnectName
```

DBA

The operation with DBA profiles is implemented with the help of *ltp8xONTAllocProfileTable*.

Adding DBA profiles

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTAllocRowStatus.<profile_index> i 4

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTAllocRowStatus.3 i 4
```

The command adds DBA profile with the index 3.

Editing DBA profiles

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
<parameter_oid_1>.<profile_id> <par1_type> <par1_value>
<parameter_oid_2>.<profile_id> <par2_type> <par2_value>
……
<parameter_oid_N>.<profile_id> <parN_type> <parN_value>

where:

- <parameter_oid_N> – names of the specified parameters in MIB;
- <profile_id> – profile index;

- <parN_type> – type of the parameter value;
- <parN_value> – parameter value.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTAllocName.3 s "edited_by_snmp"
ltp8xONTAllocServiceClass.3 i 3
ltp8xONTAllocFixedBandwidth.3 u 269248
```

The command sets name – "edited_by_snmp", service class = cbr and fixed bandwidth 269248 for the DBA profile having index 3.

Deleting DBA profiles

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTAllocRowStatus.<profile_index> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTAllocRowStatus.3 i 6
```

The commend deletes DBA profile having index 3.

Aquiring DBA profiles list

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr>
ltp8xONTAllocName

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTAllocName
```

Ports

The operation with Ports profiles is implemented with the help of the following tables:

- ltp8xONTPortsProfileTable – general parameters of a profile;
- ltp8xONTPortsProfileUNITable – UNI ports;
- ltp8xONTPortsProfileMCDynamicEntriesTable – IGMP multicast dynamic entries;
- ltp8xONTPortsProfileMLDDynamicEntriesTable – MLD multicast dynamic entries.

Adding ports profiles

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTPortsRowStatus.<profile_index> i 4

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xONTPortsRowStatus.4 i 4
```

The command adds Ports profile with the index 4.

Editing ports profiles

General parameters:

Command format:

snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid_1>.<profile_id> <par1_type> <par1_value> <parameter_oid_2>.<profile_id> <par2_type> <par2_value> … <parameter_oid_N>.<profile_id> <parN_type> <parN_value>

where:

- <parameter_oid_N> – names of the specified parameters in MIB;
- <profile_id> – profile index;
- <parN_type> – type of the parameter value;
- <parN_value> – parameter value.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTPortsName.4 s "edited_by_snmp"
ltp8xONTPortsMulticastIPVersion.4 i 1
ltp8xONTPortsMLDVersion.4 u 1
ltp8xONTPortsMLDQueryInterval.4 u 120
```

The command sets name – "edited_by_snmp", use of IPv6, MLD v1 and MLD query interval – 120 for the Ports profile having index 4.

UNI ports parameters:

Additionally to a profile index, port index (0 – 3) is specified.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTPortsUNIBridgeGroup.4.0 i 100
ltp8xONTPortsUNIMulticastEnabled.4.0 i 1
ltp8xONTPortsUNIMaxGroups.4.0 u 500
```

The command sets , bridge group = 100, max groups = 500 parameters and enables multicast for the Ports profile having index 4.

IGMP multicast dynamic entries:

Additionally to a profile index, index of a dynamic entry (0-19) is specified.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTPortsMCVLANID.4.14 u 200
ltp8xONTPortsMCFirstGroupIP.4.14 a 224.0.0.0
ltp8xONTPortsMCLastGroupIP.4.14 a 239.255.255.255
```

The command sets vlan_id = 200, first group ip = 224.0.0.0, last group ip = 239.255.255.255 for multicast dynamic entry (having index 14) of the Ports profile (having index 4).

MLD multicast dynamic entries:

Additionally to a profile index, index of a dynamic entry (0-19) is specified.

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xONTPortsMLDVLANID.4.12 u 30
ltp8xONTPortsMLDMCFirstGroupIP.4.12 x "FF010000000000000000000000000FC"
ltp8xONTPortsMLDMCLastGroupIP.4.12 x "FF010000000000000000000000000FD"
ltp8xONTPortsMLDMCPreviewLength.4.12 u 1024
```

The command sets vlan_id = 30, first group ip = FF01:0:0:0:0:0:0:FC, last group ip = FF01:0:0:0:0:0:0:FD and preview length = 1024 for the multicast dynamic entry (having index 12) of the Ports profile (having index 4).

Deleting ports profiles

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xONTPortsRowStatus.<profile_index> i 6

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTPortsRowStatus.4 i 6
```

The command deletes Port profile which has index 4.

Aquiring the list of ports profiles

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> ltp8xONTPortsName

Example:

```
snmpwalk -v2c -c public 192.168.0.1 ltp8xONTPortsName
```

Configuring PLC slots

Changing type of a PLC module

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4ShelfConfigBoardType.<slot> i <board_type>

where:

- board_type may take values – 3 (plc8) and 0 (none).

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4ShelfConfigBoardType.15 i 3
```

The command sets "PLC8" as type of the module on the 14-th slot.

Aquiring PLC module state

Command format:

snmpget -v2c -c <ro_community> <ipaddr> pp4SlotsState.<slot>

Example:

```
snmpget -v2c -c public 192.168.0.1 pp4SlotsState.15
```

The possible states of modules:

- absent (0);
- discovery (1);
- booting (2);
- operational (3);
- lost (4);
- sand (5);
- fail (6);
- notBooting (7).

PON channels

Enabling/disabling PON channels

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xPONChannelEnabled.<slot>.<gpon_port> i <value>

where:

- <gpon_port> – port number +1;
- <value> – possible values: 1 – Enable; 2 – Disable.

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xPONChannelEnabled.15.1 i 2
```

The command disables gpon-port 14/0.

Reconfiguration of PON channels

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xPONChannelReconfigure.<slot>.<gpon_port> u 1

where:

- <gpon_port> – port number +1.

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xPONChannelReconfigure.15.1 u 1
```

The command reconfigures gpon-port 14/0.

Viewing counters on PON channels

Viewing counters on a pon channel is performed by sending a request for the corresponding switch pon port.

Correspondence of pon-channels to port indexes in switch are listed in the *ltp8xSwitchPortsName* table.

PON channel	Index of switch port
0	2
1	3
2	8
3	9
4	10
5	11
6	12
7	13

Ethernet counters

Request is performed by using *ltp8xSwitchPortCountersTable*.

Command format:

snmpget -v2c -c <ro_community> <ipaddr> <counter_oid>.<slot>.<port_index>

Example:

```
snmpget -v2c -c public 192.168.0.1 ltp8xSwitchPortGoodOctetsRcv.15.9
ltp8xSwitchPortGoodPktsRcv.15.9
```

The command requests the number of received octets and packets for the 3rd pon channel of PLC on the 14th slot.

Interface utilization

Request is performed by using *ltp8xSwitchPortsUtilization*.

Command format:

snmpget -v2c -c <ro_community> <ipaddr> <utilization_oid>.<slot>.<port_index>

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xPortsUtilizationLastKbitsSent.15.13
ltp8xPortsUtilizationLastKbitsRecv.15.13
ltp8xPortsUtilizationLastFramesSent.15.13
ltp8xPortsUtilizationLastFramesRecv.15.13
ltp8xPortsUtilizationAverageKbitsSent.15.13
ltp8xPortsUtilizationAverageKbitsRecv.15.13
ltp8xPortsUtilizationAverageFramesSent.15.13
ltp8xPortsUtilizationAverageFramesRecv.15.13
```

The command requests parameters of utilization for the 7-th pon channel of PLC on the 14-th slot.

The interval of utilization counting is set by using the following command:

Command format:

snmpset -v2c -c <rw_community> <ipaddr> ltp8xPortsUtilizationInterval.0 u <interval>

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xPortsUtilizationInterval.0 u 300
```

Uploading/downloading configuration, update

OLT firmware (tftp/http)

OLT firmware download

To update OLT firmware the following parameters group is used: pp4FirmwareUpdate.

Parameter	Description
pp4FirmwareUpdateFileName	The file name of OLT firmware
pp4FirmwareUpdateIpAddress	IP address of the server storing the firmware file
pp4FirmwareUpdateSwitchVersion	Enable or disable automatic changing of firmware image to the uploaded one
pp4FirmwareUpdateNeedRestart	Enable or disable automatic system reboot after uploading firmware
pp4FirmwareUpdateProtocol	Protocol used for the file access (http/tftp)
pp4FirmwareUpdatePort	Port of the server storing the firmware
pp4FirmwareUpdateAction	Initiate the update

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4FirmwareUpdateFileName.0 s "<file_name>"
pp4FirmwareUpdateIpAddress.0 a <server_ip>
pp4FirmwareUpdateSwitchVersion.0 i <change_version_value>
pp4FirmwareUpdateNeedRestart.0 i <need_restart_value>
pp4FirmwareUpdateProtocol.0 i <download_protocol>
pp4FirmwareUpdateAction.0 u 1

where:

- <server_ip> – ip address of tftp/http server storing firmware file;
- <download_protocol> – takes values "download"/"httpdownload" for update via tftp/http correspondingly.

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4FirmwareUpdateFileName.0 s "ma4000_fw/firmware.3.26.0.1356.ma4k"
pp4FirmwareUpdateIpAddress.0 a 192.168.0.55
pp4FirmwareUpdateSwitchVersion.0 i 1
pp4FirmwareUpdateNeedRestart.0 i 1
pp4FirmwareUpdateProtocol.0 i 1
pp4FirmwareUpdateAction.0 u 1
```

The command implements uploading of firmware file named "ma4000_fw/firmware.3.26.0.1356.ma4k" from tftp server 192.168.0.55? automatically changes active firmware image and reboots the device.

The update progress can be monitored with the help of snmp-trap messages received from the device.

Changing active OLT firmware image

Command format:

snmpset -v2c -c <rw_community> -t 20 <ipaddr>
pp4UnitsActivePartition.<unit_value> i <image_value>

Example:

```
snmpset -v2c -c private -t 20 192.168.0.1 pp4UnitsActivePartition.2 i 2
```

The command sets the second firmware image as active for Unit2.

Confirming OLT firmware substitution

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4FirmwareUpdateConfirm.0 u 1

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4FirmwareUpdateConfirm.0 u 1
```

The command implements conformation of applying firmware image.

Aquiring current OLT firmware information

Command format:

snmpget -v2c -c <ro_community> <ipaddr> pp4SystemUnit1FirmwareVersion.0
pp4SystemUnit2FirmwareVersion.0

Example:

```
snmpget -v2c -c public 192.168.0.1 pp4SystemUnit1FirmwareVersion.0
pp4SystemUnit2FirmwareVersion.0
```

Rebooting OLT

Rebooting chassis

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4RebootCommand.102 u 1

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4RebootCommand.102 u 1
```

Rebooting slots (PP4X, PLC)

Command format:

snmpset -v2c -c <rw_community> <ipaddr> pp4RebootCommand.<board_id> u 1

where board_id takes values of 1-16 (PLC slots), 100 (master PP4X), 101 (slave PP4X).

Example:

```
snmpset -v2c -c private 192.168.0.1 pp4RebootCommand.15 u 1
```

The command is used for rebooting the 14-th slot.

ONT firmware

Downloading ONT firmware

To download the ONT firmware, use the *ltp8xONTFirmwaresDownload* group of parameters.

Parameter	Description
ltp8xONTFirmwaresDownloadPath	The name of the ONT FW file
ltp8xONTFirmwaresDownloadIPAddress	IP address of the server storing firmware file.
ltp8xONTFirmwaresDownloadProtocol	The protocol for access to http/tftp file.
ltp8xONTFirmwaresDownloadPort	A server port with firmware file.
ltp8xONTFirmwaresDownloadAction	Initiate update process

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTFirmwaresDownloadPath.0 s "<file_name>" ltp8xONTFirmwaresDownloadIPAddress.0 a <server_ip>
ltp8xONTFirmwaresDownloadProtocol.0 i <download_protocol>
ltp8xONTFirmwaresDownloadPort.0 u <server_port>
ltp8xONTFirmwaresDownloadAction.0 u 1

where:

- <server_ip> – ip address of tftp/http server storing firmware file;
- <download_protocol> – takes values "download"/"httpdownload" for update via tftp/http correspondingly.

Example:

```
snmpset -v2c -c private 192.168.0.1
ltp8xONTFirmwaresDownloadPath.0 s "ntp-rg-revb-3.24.3.87.fw.bin"
ltp8xONTFirmwaresDownloadIPAddress.0 a 192.168.0.55 ltp8xONTFirmwaresDownloadProtocol.0 i 2
ltp8xONTFirmwaresDownloadPort.0 u 8080
ltp8xONTFirmwaresDownloadAction.0 u 1
```

The command loads firmware file ntp-rg-revb-3.24.3.87.fw.bin from port 8080 of http server 192.168.0.55.

Acquiring the list of downloaded ONT firmware

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr>
ltp8xONTFirmwaresFilesName

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xONTFirmwaresFilesName
```

Deleting uploaded ONT firmware

Command format:

snmpset -v2c -c <rw_community> <ipaddr>
ltp8xONTFirmwaresFilesDelete.<file_id> u 1

where:

- <file_id> – ONT firmware file ID in the list (see section Acquiring the list of downloaded ONT firmware).

Example:

```
snmpset -v2c -c private 192.168.0.1 ltp8xONTFirmwaresFilesDelete.2 u 1
```

The command deletes firmware file with index 2.

Uploading/downloading ONT configuration

Downloading backup configuration

Command format:

Via TFTP:

snmpset -v2c -c <rw_community> <ipaddr>
cmdFileOperationPrivateCfg.0 s "<tftp_server_ip> <tftp_path> upload"

Via HTTP:

snmpset -v2c -c <rw_community> <ipaddr>
cmdFileOperationPrivateCfg.0 s "<http_sever_ip> <http_path> httpupload"

where:

- <tftp_path>, <http_path> – full path for uploading file on the server.

Example:

```
snmpset -v2c -c private 192.168.0.1 cmdFileOperationPrivateCfg.0 s
"192.168.0.55 new_config.cfg upload"
```

The command uploads the configuration via TFTP to the server 192.168.0.55 in a file named new_config.cfg.

Uploading ONT backup configuration

Command format:

Via TFTP:

snmpset -v2c -c <rw_community> <ipaddr>
cmdFileOperationPrivateCfg.0 s "<tftp_ip> <tftp_path> download"

Via HTTP:

snmpset -v2c -c <rw_community> <ipaddr>
cmdFileOperationPrivateCfg.0 s "<http_ip> <http_path> httpdownload"

where:

- <tftp_path>, <http_path> – full path for downloading file from the server.

Example:

```
snmpset -v2c -c private 192.168.0.1 cmdFileOperationPrivateCfg.0 s
"192.168.0.55:8080 config/new_config.cfg httpdownload"
snmpset -v2c -c private 192.168.0.1 cmdFileOperationPrivateCfg.0 s
"192.168.0.55:8080 config/new_config.cfg apply"
snmpset -v2c -c private 192.168.0.1 cmdFileOperationPrivateCfg.0 s
"192.168.0.55:8080 config/new_config.cfg confirm"
```

The command implements configuration upload via HTTP from the server 192.168.0.55, port 8080 from config/new_config.cfg.

OLT monitoring

Active alarms

Getting the number of active alarms

Command format:

snmpget -v2c -c <ro_community> <ipaddr> omsActiveAlarms.0

Example:

```
snmpget -v2c -c public 192.168.0.1 omsActiveAlarms.0
```

Getting the active alarms in trap forms:

Command format:

snmpset -v2c -c <rw_community> <ipaddr> omsActiveAlarms.0 u 1

Example:

```
snmpset -v2c -c private 192.168.0.1 omsActiveAlarms.0 u 1
```

The command sends request to display all the device active alarms by snmp-trap messages.

General information on chassis, PP4X, PLC8

General information on MA4000 is in *pp4System* and *pp4BoardState groups*.

Command format:

snmpget -v2c -c <ro_community> <ipaddr> <parameter_oid>.0

Example:

```
snmpget -v2c -c public 192.168.0.1 pp4SystemUnit1SerialNumber.0
pp4BoardFan3AbsoluteSpeed.0
```

The command displays the serial number of Unit1 and current speed of Fan3.

General information on PLC8 is in *ltp8xPLCBoardStateTable.*

Command format:

snmpget -v2c -c <ro_community> <ipaddr> <parameter_oid>.<slot>

Example:

```
snmpget -v2c -c public 192.168.0.1 ltp8xPLCBoardStateRAMFree.15
```

The command displays the amount of available RAM in bytes for the 14-th slot.

Power supply

Command format:

snmpget -v2c -c <ro_community> <ipaddr> pp4Feeder1Status.0 pp4Feeder1Active.0 pp4Feeder1Polarity.0 pp4Feeder1Current.0 pp4Feeder1Voltage.0 pp4Feeder2Status.0 pp4Feeder2Active.0 pp4Feeder2Polarity.0 pp4Feeder2Current.0 pp4Feeder2Voltage.0 pp4StationVoltage.0

Example:

```
snmpget -v2c -c public 192.168.0.1 pp4Feeder1Status.0 pp4Feeder1Active.0
pp4Feeder1Polarity.0 pp4Feeder1Current.0 pp4Feeder1Voltage.0
pp4Feeder2Status.0 pp4Feeder2Active.0 pp4Feeder2Polarity.0
pp4Feeder2Current.0 pp4Feeder2Voltage.0 pp4StationVoltage.0
```

State of PP4X ports and PON channels of PLC8

ifTable is used for displaying of port state.

Command format:

snmpget -v2c -c <ro_community> <ipaddr> ifOperStatus.<ifIndex>

Example:

```
snmpget -v2c -c public 192.168.0.1 ifOperStatus.63
```

The command displays the state of front-port 2/1.

For displaying of pon channels, *ltp8xPONChannelStateTable is used*.

Command format:

snmpget -v2c -c <ro_community> <ipaddr>
<parameter_oid>.<slot>.<pon_channel_id>

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xPONChannelONTCount.15.4
ltp8xPONChannelSFPVendor.15.4
ltp8xPONChannelSFPProductNumber.15.4
ltp8xPONChannelSFPRevision.15.4
```

The command displays the quantity of ONT and SFP data for the 3-rd channel on the 14-th slot.

MAC Table

MAC table of PP4X:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> pp4MacAddressEntryID

Example:

```
snmpwalk -v2c -c public 192.168.0.1 pp4MacAddressEntryID
```

The command displays indexes of PP4X MAC table.

Command format:

snmpget -v2c -c <ro_community> <ipaddr> pp4MacAddressVLAN.<entry_id>
pp4MacAddressAddress.<entry_id> pp4MacAddressPort.<entry_id>
pp4MacAddressType.<entry_id>

where <entry_id> – number of an entry in the table.

Example:

```
snmpget -v2c -c public 192.168.0.1 pp4MacAddressVLAN.3
pp4MacAddressAddress.3 pp4MacAddressPort.3 pp4MacAddressType.3
```

The command displays the 3-rd entry from PP4X MAC table.

Switch PLC MAC table:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> ltp8xSwitchMacListMacAddressString.<slot>

Example:

```
snmpwalk -v2c -c public 192.168.0.1
ltp8xSwitchMacListMacAddressString.15
```

The command displays MAC table of the 14-th slot as MAC list.

Then, knowing the parameters of a certain entry, you may additionally aquire the interface and the type of it.

Command format:

snmpget -v2c -c <ro_community> <ipaddr> ltp8xSwitchMacListInterface.<slot>.<vid>.<dec_macaddress> ltp8xSwitchMacListStatic.<slot>.<vid>.<dec_macadress>

where:

- <dec_macadress> – MAC address in a decimal form.

Example:

```
snmpget -v2c -c public 192.168.0.1 ltp8xSwitchMacListInterface.
15.236.152.222.208.0.205.252 ltp8xSwitchMacListStatic.
15.236.152.222.208.0.205.252
```

The command requests interface and type of the entry from the 14-th slot having MAC 98:de:d0:00:cd:fc in 236 VLAN.

Multicast

pp4MulticastGroupsTable might be used for PP4X Multicast groups request.

To aquire list of groups and their entry_id, type the following command:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> pp4MulticastEntryID

Example:

```
snmpwalk -v2c -c public 192.168.0.1 pp4MulticastEntryID
```

Command format:

snmpget -v2c -c <ro_community> <ipaddr> pp4MulticastVLAN.<entry_id> pp4MulticastGroupAddress.<entry_id> pp4MulticastMemberPorts.<entry_id> pp4MulticastExpires.<entry_id>

where entry_id – number of entry in the table.

Example:

```
snmpget -v2c -c public 192.168.0.1 pp4MulticastVLAN.1
pp4MulticastGroupAddress.1 pp4MulticastMemberPorts.1
pp4MulticastExpires.1
```

The command requests data on multicast group having index 1 in the table.

PLC multicast groups: see section IGMP group table.

PPPoE sessions

Data on PPPoE sessions is available in the *ltp8xOLTPPPoESessionsTable table*.

To aquire a list of client MAC addresses of sessions, use the following command:

Command format:

snmpwalk -v2c -c <ro_community> <ipaddr> ltp8xOLTPPPoESessionsClientMac.<slot>.<gpon_port>.<ont_id>

Example:

```
snmpwalk -v2c -c public 192.168.0.1 ltp8xOLTPPPoESessionsClientMac.15.7.0
```

The command displays the table of client MAC addresses and corresponding GPON-PORT/ONT ID.

Knowing data on a certain entry in the table, you may request for the additional data (ONT GEM port, ID of PPPoE session, duration of PPPoE session, tim e of ONT port unblocking (time of ONT port unblocking does not equal to 0 when blocking due to the PPPoE packets limit exceeding. Value of the limit might be set in pppoe-ia profile):

Command format:

snmpget -v2c -c <ro_community> <ipaddr>
ltp8xOLTPPPoESessionsPort.<slot>.<channel>.<ont_id>.<client_mac>
ltp8xOLTPPPoESessionsSessionID.<slot>.<channel>.<ont_id>.<dec_client_mac>
ltp8xOLTPPPoESessionsDuration.<slot>.<channel>.<ont_id>.<client_mac>
ltp8xOLTPPPoESessionsUnblock.<slot>.<channel>.<ont_id>.<client_mac>
ltp8xOLTPPPoESessionsSerial.<slot>.<channel>.<ont_id>.<client_mac>

Example:

```
snmpget -v2c -c public 192.168.0.1
ltp8xOLTPPPoESessionsPort.15.7.0.168.249.75.90.189.124
ltp8xOLTPPPoESessionsSessionID.15.7.0.168.249.75.90.189.124
ltp8xOLTPPPoESessionsDuration.15.7.0.168.249.75.90.189.124
ltp8xOLTPPPoESessionsUnblock.15.7.0.168.249.75.90.189.124
ltp8xOLTPPPoESessionsSerial.15.7.0.168.249.75.90.189.124
```

The command displays data on PPPoE session of ONT 14/6/0 having MAC a8:f9:4b:5a:bd:7c.