

Модульная Сетевая Операционная Система Телекоммуникационного Оборудования –  
SDK CITOS

Руководство программиста. Часть 2

643.ТЛМБ.13001-01 33 01-2

Листов 132

## Содержание

<b>1</b>	<b>Введение</b>	<b>4</b>
<b>2</b>	<b>Функциональные характеристики</b>	<b>5</b>
2.1	Поддерживаемые протоколы . . . . .	5
2.1.1	Поддержка Port Control . . . . .	5
2.1.2	Поддержка QOS . . . . .	5
2.1.3	Поддержка L2 Switching . . . . .	5
2.1.4	Поддержка Protection . . . . .	6
2.1.5	Поддержка L3 & Routing . . . . .	6
2.1.6	Поддержка Security . . . . .	7
2.1.7	Поддержка OAM & Test . . . . .	7
2.1.8	Поддержка Management . . . . .	7
2.1.9	Поддержка Standard MIBs . . . . .	8
2.1.10	Поддержка Synchronization . . . . .	9
2.2	Интерфейс оператора конечного оборудования (интерфейс командной строки) . . . . .	9
<b>3</b>	<b>Генерация команд ICLI</b>	<b>65</b>
3.1	Поток генерации . . . . .	65
3.2	Дизайн команд . . . . .	65
3.2.1	Переменная « < > » . . . . .	66
3.2.2	Исключающее OR « » (Или) . . . . .	66
3.2.3	Обязательный «{ }» . . . . .	67
3.2.4	Необязательный «[ ]» . . . . .	67
3.2.5	Случайный Необязательный «[ ] [] ... [ ]» . . . . .	67
3.2.6	Случайное обязательное «{[ ] [] ... [ ]}*n» . . . . .	68
3.2.7	Повторить « ... » . . . . .	68
3.2.8	Цикл «( )*N» . . . . .	69
3.2.9	Ограничения . . . . .	69
3.3	Состав файла сценария . . . . .	70
3.3.1	Сегмент Module . . . . .	71
3.3.2	Сегмент Include . . . . .	72
3.3.3	Сегмент Function . . . . .	72
3.3.4	Сегмент Command . . . . .	73
3.3.5	Defining Constant String . . . . .	85
3.3.6	Пример сегмента команды . . . . .	86
3.4	Компиляция: Make File . . . . .	88
3.5	Руководство ICLI . . . . .	103
3.5.1	Повторяющиеся слова . . . . .	103
3.5.2	Множественное необязательное начало . . . . .	104
3.5.3	Уровни подрежима . . . . .	104
3.6	Клавиши быстрого доступа . . . . .	104
<b>4</b>	<b>Настройка Web-интерфейса</b>	<b>106</b>
4.1	Изменение названия продукта . . . . .	106
4.2	Изменение описания продукта . . . . .	106
4.3	Изменение логотипа продукта . . . . .	107
4.4	Изменение изображения коммутатора . . . . .	108

4.5	Изменение описания UPNP . . . . .	108
4.6	Изменение версии программного обеспечения . . . . .	109
4.7	Изменение цвета графического интерфейса . . . . .	110
4.8	Изменение IP-адреса по умолчанию . . . . .	115
4.9	Добавление новой ссылки на панель навигации . . . . .	116
4.10	Изменение расположения портов . . . . .	117
<b>5</b>	<b>Настройка JSON-RPC</b>	<b>119</b>
5.1	Начало работы с JSON-RPC . . . . .	119
5.1.1	Использование vson . . . . .	120
5.2	Спецификации интерфейса JSON . . . . .	121
5.2.1	Варианты использования . . . . .	121
5.2.2	Возможности . . . . .	122
5.3	Обратная и прямая совместимость . . . . .	129
5.3.1	Рекомендации по внедрению изменений . . . . .	129
5.3.2	Реализация совместимости . . . . .	130
5.4	Модели интеграции . . . . .	130
5.4.1	HTTP(s) как анализ сообщений . . . . .	130
5.4.2	Альтернативные системы анализа сообщений . . . . .	131

## **1 Введение**

Руководство программиста состоит из двух частей.

Часть 1 - содержит инструкции по установке, настройке SDK CITOS, а также базовый набор действий по сборке двоичного образа под целевую платформу.

Часть 2 - содержит описание функциональных характеристик SDK CITOS, решение задач конфигурирования под целевую платформу, требования к ресурсам.

## 2 Функциональные характеристики

В данном разделе перечисляются поддерживаемые программным обеспечением протоколы взаимодействия в телекоммуникационных сетях, а также поддерживаемый список команд в интерфейсе оператора конечного оборудования.

### 2.1 Поддерживаемые протоколы

Программное обеспечение поддерживает следующие протоколы взаимодействия в телекоммуникационных сетях на основе международных отраслевых стандартов:

#### 2.1.1 Поддержка Port Control

- Port Speed/Duplex Mode/Flow Ctrl;
- Port Frame Size (Jumbo frames);
- Port State (administrative status);
- Port Status (link monitoring);
- Port Statistics (MIB counters);
- DDMI;
- UDLD.

#### 2.1.2 Поддержка QoS

- Traffic Classes (8 active priorities);
- Port Default Priority;
- User Priority;
- Input priority mapping;
- QoS Control List (QCL Mode);
- Global Storm Control for UC, MC and BC;
- Random Early Discard (RED);
- Port policers;
- Queue policers;
- Global/VCAP (ACL) policers;
- Port egress shaper;
- Queue egress shapers;
- DiffServ (RFC2474) remarking;
- Tag remarking;
- Scheduler mode;
- Management of credit-based shaper;
- IEEE-802.1Qbv (TAS) Time-aware Scheduler;
- IEEE-802.1Qbu & 802.3br Frame Preemption;
- IEEE-802.1Qci ingress gating/policing/checking.

#### 2.1.3 Поддержка L2 Switching

- Auto MAC addr. Learning/Ageing;
- MAC Addresses - Static;
  
- Virtual LAN;
- Bidirectional VLAN translation;

- Unidirectional VLAN translation (ingress/egress);
- Private VLAN Static;
- Port Isolation Static;
- MAC-based VLAN;
- Protocol-based VLAN;
- IP subnet-based VLAN;
- VLAN Trunking;
- iPVLAN Trunking;
- GARP VLAN registration GVRP;
- Multiple Registration Protocol MRP;
- Multiple VLAN Registration Protocol MVRP;
- IEEE-802.1ad Provider Bridge (Native or Translated VLAN);
- MSTP;
- Rapid Spanning Tree RSTP, STP;
- Loop Guard;
- Link Aggregation Static;
- Link Aggregation LACP;
- AGGR/LACP user interface alignment with Industry standard;
- UNI LAG (LACP) 1:1 Active/Standby;
- LACP Revertive/Non-revertive;
- LACP loop free operation;
- BPDU Guard & Restricted Role;
- IGMPv2 snooping;
- IGMPv3 snooping;
- MLDv1 snooping;
- MLDv2 snooping;
- IGMP filtering profile;
- IPMC throttling, filtering, leave proxy;
- MVR;
- MVR profile;
- Voice VLAN;
- DHCP snooping;
- ARP inspection;
- Port Mirroring;
- Flow mirroring;
- Rmirror;
- DHCPv6 Shield.

#### 2.1.4 Поддержка Protection

- G.8032 - Ring protection;
- G.8032 - Ring protection v2;
- MRP (Media redundancy protocol);
- MRP (Media redundancy protocol) Interconnect;
- IEEE-802.1CB (FRER);

#### 2.1.5 Поддержка L3 & Routing

- DHCP option 82 relay;

- UPNP;
- HW based IPv4 L3 Static Routing with Linux Kernel integration;
- RFC2992 (ECMP) support for HW based L3 static routing;
- RFC 2453 RIPv2 dynamic routing;
- RFC 2328 OSPFv2 Dynamic routing;
- RFC 3101 The OSPF Not-So-Stubby Area (NSSA) Option;
- RFC 3137 OSPF Stub Router Advertisement;
- HW based IPv6 L3 Static Routing;
- RFC 2740/5340 OSPFv3 Dynamic Routing;
- RFC-1812 L3 checking (version, IHL, checksum, и т.д.).

#### 2.1.6 Поддержка Security

- Port-Based 802.1X;
- Single 802.1X;
- Multiple 802.1X;
- MAC-Based Authentication;
- VLAN Assignment;
- QoS Assignment;
- Guest VLAN;
- RADIUS Authentication and Authorization;
- RADIUS Accounting;
- MAC Address Limit;
- Persistent MAC learning;
- IP MAC binding;
- IP/MAC binding dynamic to static;
- TACACS+ Authentication and Authorization;
- TACACS+ Command Authorization;
- TACACS+ Accounting;
- Web & CLI Authentication;
- Authorization (15 user levels);
- ACLs for filtering/policing/port copy;
- IP source guard;
- Secure FTP Client;
- IP6 Source guard.

#### 2.1.7 Поддержка OAM & Test

- CFM: Continuity Check (ETH-CCM);
- CFM: IS-, OS-, PS-, SID-TLV;
- CFM: APS using ETH-CCM and ETH-APS;
- CFM: ERPS using ETH-CCM and ETH-RAPS;
- CMF: HW-accelerated OAM.

#### 2.1.8 Поддержка Management

- JSON-RPC;
- JSON-RPC Notifications;
- RFC 2131 DHCP Client;

- RFC 2131 DHCP Server;
- DHCP Server support for DHCP relay packets;
- DHCP per port;
- RFC 3315 DHCPv6 Client;
- RFC 3315 DHCPv6 Relay Agent;
- RFC 7610 DHCPv6-ShieldProtecting against Rogue DHCPv6 Servers;
- RFC 1035 DNS client, relay;
- IPv4/IPv6 Ping;
- IPv4/IPv6 Traceroute;
- HTTP Server;
- CLI - Console Port;
- CLI - Telnet;
- Port Description CLI;
- Management access filtering;
- HTTPS;
- SSHv2;
- IPv6 Management;
- IPv6 Ready Logo PHASE2 (host only);
- RFC4884 (ICMPv6);
- System Syslog;
- Software Upload via web;
- SNMPv1 / v2c / v3 Agent - Note: No SNMPv1 trap support;
- RMON (Group 1, 2, 3 & 9);
- RMON alarm and event(CLI,web);
- IEEE 802.1AB-2005 Link Layer Discovery LLDP;
- TIA 1057 LLDP-MED;
- sFlow;
- FTP Client;
- Configuration Download/Upload - Industrial Standard;
- Loop detection restore to default.

### 2.1.9 Поддержка Standard MIBs

- RFC 2674 VLAN MIB;
- IEEE 802.1Q Bridge MIB 2008;
- RFC 2819 RMON (Group 1, 2, 3 & 9);
- RFC 1213 MIB II;
- RFC 1215 TRAPS MIB;
- RFC 4188 Bridge MIB;
- RFC 4292 IP Forwarding Table MIB;
- RFC 4293 Management Information Base for the Internet Protocol (IP);
- RFC 5519 Multicast Group Membership Discovery MIB;
- RFC 4668 RADIUS auth. Client MIB;
- RFC 4670 RADIUS Accounting MIB;
- RFC 3635 Ethernet-like MIB;
- RFC 2863 Interface Group MIB using SMI v2;
- RFC 3636 802.3 MAU MIB;
- RFC 4133 Entity MIB version 3;
- RFC 4878 Link OAM MIB;

- RFC 3411 SNMP Management Frameworks;
- RFC 3414 User-based Security Model for SNMPv3;
- RFC 3415 View-based access Control Model for SNMP;
- RFC 2613 SMON - PortCopy;
- IEEE 802.1 MSTP MIB;
- IEEE 802.1AB LLDP-MIB (LLDP MIB included in a clause of the STD);
- IEEE 802.3ad (LACP MIB included in a clause of the STD);
- IEEE 802.1X (PAE MIB included in a clause of the STD);
- TIA 1057 LLDP-MED ( MIB is part of the STD).

#### 2.1.10 Поддержка Synchronization

- SyncE with SSM;
- SyncE Nomination For 2 Interfaces;
- 1588v2 PTP with two step clock;
- 1588v2 PTP with one step clock;
- Peer-to-peer transparent clock over Ethernet/IPv4;
- End-to-end transparent clock over Ethernet/IPv4;
- End-to-end transparent clock over Ethernet/IPv6;
- Boundary Clock;
- Redundant masters and multiple timing domains;
- PTP over IPv4;
- Unicast/Multicast;
- TC internal Master/Slave w. PDV filtering and no modulation or latency feedback from modems;
- TC internal Master/Slave w. reduced PDV filtering and modem provides feedback on modulation or latency;
- Combined SyncE & 1588;
- MSCC Timing BU servo algorithm integration;
- MSCC Timing BU DPLL API integration;
- G.8265.1 BMCA;
- ITU G.8263 filtering;
- PTP Profile;
- Clock Quality;
- G.781 compliant clock selection algorithm for the platform as a PTP slave;
- G.8275.1 BMCA;
- G.8275 Compliant Filter;
- PTP Time Interface;
- NTPv4 Client;
- IEEE802.1AS-2011/IEEE802.1AS rev D4.2.

## 2.2 Интерфейс оператора конечного оборудования (интерфейс командной строки)

Программное обеспечение содержит средства разработки и модули поддержки интерфейса командной строки конечного оборудования. В зависимости от типа используемых модулей общепромышленных протоколов программное обеспечение имеет возможность автоматического формирования исходного кода для поддержки следующего списка команд:

- aaa accounting { console | telnet | ssh } tacacs { [ commands <priv\_lvl> ] [ exec ] }

- aaa authentication login { console | telnet | ssh | http } { { local | radius | tacacs } [ { local | radius | tacacs } [ { local | radius | tacacs } ] ] }
- aaa authorization { console | telnet | ssh } tacacs commands <priv\_lvl> [ config-commands ]
- access management
- access management <access\_id> <access\_vid> <start\_addr> [ to <end\_addr> ] { [ web ] [ snmp ] [ telnet ] | all }
- access management <access\_id> <access\_vid> <start\_addr> [ to <end\_addr> ] { [ web ] [ snmp ] [ telnet ] | all }
- access-list action { permit | deny }
- access-list logging
- access-list mirror
- access-list policy <policy\_id>
- access-list port-state
- access-list rate-limiter <rate\_limiter\_id>
- access-list rate-limiter [ <rate\_limiter\_list> ] { pps <pps\_rate> | 10pps <pps10\_rate> | 100pps <pps100\_rate> | 25kbps <kpbs25\_rate> | 100kbps <kpbs100\_rate> }
- access-list shutdown
- access-list { redirect } interface { <port\_type> <port\_type\_id> | <port\_type> [ <port\_type\_list> ] }
- action { [ class ] [ cos ] [ dpl ] [ pcp ] [ dei ] [ dscp ] }
- action { [ pcp ] [ dei ] [ dscp ] }
- address <v\_ipv4\_addr> interface <port\_type> <ifc>
- admin-state { enable | disable }
- admin-state { enable | disable }
- admin-state { enable | disable }
- admin-state { enable | disable }
- admin-state { enable | disable }
- aggregation group <v\_uint> mode { [ active | on | passive ] }
- aggregation mode { [ smac ] [ dmac ] [ ip ] [ port ] }
- alarm <alarm\_name> <alarm\_expression>
- alarm suppress <alarm\_name>
- alarm-level <alarm\_level>
- alarm-time-absent <alarm\_time\_absent\_ms>
- alarm-time-present <alarm\_time\_present\_ms>
- aps <inst>
- aps <inst> clear
- aps <inst> exercise
- aps <inst> freeze
- aps <inst> lockout
- aps <inst> switch { force | manual { protect-to-working | working-to-protect } }
- area <area\_id> authentication [ message-digest ]
- area <area\_id> nssa [ translate type7 { candidate | never | always } | no-summary ]
- area <area\_id> range <ip\_address> <ip\_address\_mask> [ advertise [ cost <advertise\_cost> ] | not-advertise | cost <cost> ]
- area <area\_id> range <ipv6\_subnet> [ advertise [ cost <advertise\_cost> ] | not-advertise | cost <cost> ]
- area <area\_id> stub [ no-summary ]
- area <area\_id> stub [ no-summary ]

- area <area\_id> virtual-link <router\_id> [ hello-interval <hello\_interval> ] [ retransmit-interval <retransmit\_interval> ] [ dead-interval <dead\_interval> ]
- area <area\_id> virtual-link <router\_id> authentication [ null | message-digest ]
- area <area\_id> virtual-link <router\_id> authentication-key { unencrypted <unencrypted\_pwd> | encrypted <encrypted\_pwd> }
- area <area\_id> virtual-link <router\_id> message-digest-key <md\_key\_id> md5 { unencrypted <unencrypted\_pwd> | encrypted <encrypted\_pwd> }
- authentication basic username <username> [ password <password> ]
- banner [ motd | login | exec ] <banner>
- base-time seconds <seconds> [ nanoseconds <nanoseconds> ]
- block-due-to-oversize-enable
- broadcast <ip>
- cbs <cbs>
- cfm domain <md\_name>
- cfm interface-status-tlv { disable | enable }
- cfm organization-specific-tlv { disable | enable oui <oui> subtype <subtype> value <value> }
- cfm port-status-tlv { disable | enable }
- cfm sender-id-tlv { disable | chassis | management | chassis-management }
- cir <cir>
- clear access management statistics
- clear access-list ace statistics
- clear aps [ <inst\_list> ] statistics
- clear cfm meps [ domain <md\_name> ] [ service <ma\_name> ] [ mep-id <mepid> ] statistics
- clear dot1x statistics [ interface <port\_type> [ <v\_port\_type\_list> ] ]
- clear erps [ <inst\_list> ] statistics
- clear ip acd
- clear ip arp
- clear ip dhcp detailed statistics { server | client | snooping | relay | helper | all } [ interface <port\_type> [ <in\_port\_list> ] ]
- clear ip dhcp relay statistics
- clear ip dhcp server binding <ip>
- clear ip dhcp server binding type { automatic | manual | expired }
- clear ip dhcp server statistics
- clear ip dhcp snooping statistics [ interface <port\_type> [ <in\_port\_list> ] ]
- clear ip igmp snooping [ vlan <v\_vlan\_list> ] statistics
- clear ip ospf process
- clear ip rip process
- clear ip statistics
- clear ipv6 dhcp relay statistics [ interface vlan <vlan\_id> [ interface vlan <rel\_vlan\_id> ] ]
- clear ipv6 dhcp snooping statistics [ interface <port\_type> [ <in\_port\_list> ] ]
- clear ipv6 mld snooping [ vlan <v\_vlan\_list> ] statistics
- clear ipv6 neighbors
- clear ipv6 ospf process
- clear ipv6 statistics
- clear known-host-keys
- clear lacp statistics
- clear link-oam statistics [ interface <port\_type> [ <plist> ] ]

- clear lldp statistics { [ interface <port\_type> [ <plist> ] ] | global }
- clear logging [ informational ] [ notice ] [ warning ] [ error ]
- clear mac address-table
- clear media-redundancy [ <inst\_list> ] statistics
- clear mvr [ vlan <v\_vlan\_list> | name <mvr\_name> ] statistics
- clear network-clock clk-source <clk\_list>
- clear port-security dynamic [ { address <mac> [ vlan <vlan\_on\_mac> ] } | { interface <port\_type> [ <plist> ] [ vlan <vlan\_on\_interface> ] } | vlan <vlan> ]
- clear ptp <clockinst> servo
- clear sflow statistics { receiver [ <receiver\_index\_list> ] | samplers [ interface [ <samplers\_list> ] <port\_type> [ <v\_port\_type\_list> ] ] }
- clear spanning-tree { { statistics [ interface <port\_type> [ <v\_port\_type\_list> ] ] } | { detected-protocols [ interface <port\_type> [ <v\_port\_type\_list\_1> ] ] } }
- clear statistics [ interface ] <port\_type> [ <port\_list> ]
- clear system led status [ switch <switch\_list> ] { fatal | software | post | ztp | stack-firmware | all }
- clear tsn flow meter [ <index\_list> ] [ mark-red ]
- clear tsn frer [ <inst\_list> ] statistics
- clear tsn stream filter [ <index\_list> ] [ statistics | stream-blocked-due-to-oversize-frame ]
- clear tsn stream gate [ <index\_list> ] [ gate-closed-due-to-octets-exceeded | gate-closed-due-to-invalid-rx ]
- client-identifier { { fqdn | name } <identifier> | mac-address <mac> }
- client-name <host\_name>
- clock summer-time <word16> date [ <start\_month\_var> <start\_date\_var> <start\_year\_var> <start\_hour\_var> <end\_month\_var> <end\_date\_var> <end\_year\_var> <end\_hour\_var> [ <offset\_var> ] ]
- clock summer-time <word16> recurring [ <start\_week\_var> <start\_day\_var> <start\_month\_var> <start\_hour\_var> <end\_week\_var> <end\_day\_var> <end\_month\_var> <end\_hour\_var> [ <offset\_var> ] ]
- clock timezone <word\_var> <hour\_var> [ <minute\_var> [ <subtype\_var> ] ]
- clock timezone offset <offset\_var>
- close-due-to-invalid-rx-enable
- close-due-to-octets-exceeded-enable
- color-mode
- config-change
- configure terminal
- continuity-check
- continuity-check interval { 3.3ms | 10ms | 100ms | 1s | 10s | 1min | 10min }
- control-list index <index> gate-state { open | closed } time-interval <interval> { ms | us | ns } [ ipv <ipv> ] [ octet-max <octet\_max> ]
- control-list-length <gcl\_length>
- control-vlan <vid>
- control-vlan <vid> [ pcp <pcp> ]
- copy { startup-config | running-config | <source\_path> } { startup-config | running-config | <destination\_path> } [ syntax-check ] [ save-host-key ] [ ftp-active ]
- coupling-flag
- cycle-time <cycle\_time> { ms | us | ns }
- ddmi

- debug aaa cmd { console | telnet | ssh } <username> [ priv-lvl <priv\_lvl> ] [ config-commands ] <command>
- debug aaa login { console | telnet | ssh | http } <username> [ <password> ]
- debug aaa logout { console | telnet | ssh } <username>
- debug acl PortListStackable <port\_type> [ <plist> ]
- debug acl all
- debug acl capabilities
- debug acl delete ace <ace\_id> <ace\_user\_name>
- debug acl detailed ace <ace\_id> <ace\_user\_name> [ switch <switch\_list> ]
- debug acl ifmux
- debug acl rate-limiter boundary
- debug afi [ { single | multi } ] { { pause | resume | cancel | print } { all | <id> } [ force ] | list }
- debug api [ interface <port\_type> [ <port\_list> ] ] [ action <action> ] [ { ail | cil } ] [ { init | misc | port | counters | phy [ vml-pattern ] | vlan | pvlan | mac-table | acl | qos | aggr | stp | mirror | erps | eps | packet | fdma | ts | pts | wm | ipmc | stack | cmef | vxlat | oam | sgpio | l3 | afi | serdes | kr } ] [ full ] [ clear ]
- debug assert { appl | os | except | cap | asan <value> }
- debug auto-failover interface <port\_type> [ <v\_port\_type\_list> ] [ <v\_uint16> <v\_uint16\_1> { host | line } event { gpio | los | lof | linkstatus | none } { enable | disable } [ filter <v\_0\_to\_4> <v\_uint16\_2> ] [ gpio-pins <v\_uint16\_3> <v\_uint16\_4> ] ]
- debug board [ mac <mac> ] [ serial <serial> ]
- debug board tag { save | list | <name> [ <value> ] | remove <name\_to\_remove> }
- debug chip [ { 0 | 1 | all } ]
- debug clear erps history
- debug clear ip statistics
- debug clear media-redundancy history
- debug clear port registrations
- debug clear port-security rate-limit statistics
- debug clear vlan counter <vid>
- debug clock read <v\_uint> [ <v\_uint\_1> ]
- debug clock test config
- debug clock write <v\_uint> <v\_uint\_1>
- debug conf blocks [ clear ]
- debug conf change [ { enable | disable } ]
- debug conf flash [ { enable | disable } ]
- debug conf stack [ { enable | disable } ]
- debug critd leaf-test
- debug critd list [ <module\_name> ] [ detailed ]
- debug critd maxlock [ <module\_name> ] [ clear ]
- debug critd { semaphore | mutex | recursive-mutex } { init | enter | exit | delete }
- debug ddmi interface <port\_type> [ <port\_list> ] { temperature | voltage | bias | tx-power | rx-power } { none | low-warn | high-warn | low-alarm | high-alarm }
- debug dhcp server binding { ip | id | chaddr | name | lease | expired }
- debug dhcp server declined add <declined\_ip>
- debug dhcp server declined delete <declined\_ip>
- debug dhcp server statistics
- debug eee wakeup-time [ { rx | tx } { <v\_uint16> | clear } ]
- debug enable password

- debug file create <file\_name> <file\_size>
- debug file delete <file\_name>
- debug file ls [ <path> ]
- debug file restore
- debug file sync
- debug firmware bootloader <url\_file> [ save-host-key ] [ ftp-active ]
- debug firmware bootstrap <url\_file> [ size <size> ] [ force ] [ save-host-key ] [ ftp-active ]
- debug firmware bootstrap [ nand | nor | mmc ] [ size <size> ]
- debug firmware fis resize <name> <new\_size>
- debug firmware load <fis> <url\_file> [ save-host-key ] [ ftp-active ]
- debug firmware max-size [ <max\_size> ]
- debug firmware name [ <filename> ]
- debug firmware ramload <url\_file> [ save-host-key ] [ ftp-active ]
- debug frame tx [ interface <port\_type> [ <plist> ] ] [ { fph <fph> [ keep-paused ] } | { emix <emix> kbps <kbps> [ rate-type { line | data } ] [ keep-paused ] } ] [ packet-length <packet\_length> ] [ dmac <dmac> ] [ smac <smac> ] [ otag vid <ovid> [ tpid <otpid> ] [ pcp <opcp> ] [ dei <odei> ] ] [ itag vid <ivid> [ tpid <itpid> ] [ pcp <ipcp> ] [ dei <idei> ] ] [ etype <etype> ] [ masquerade ] [ data <data1> <data2> <data3> <data4> ] [ payload { random | fixed <fixed\_payload\_value> } ]
- debug frame tx-cnt <tx\_cnt> [ dmac-incr ] [ smac-incr ] [ sleep <sleep\_ms> ]
- debug frr { zebra | staticd | ospfd | ospf6d | ripd } [ hex ] cmd <cmds>
- debug gdbserver
- debug gvrp msti
- debug gvrp protocol-state interface <port\_type> [ <v\_port\_type\_list> ] vlan <v\_vlan\_list>
- debug gvrp statistic
- debug heap [ changes-only ] [ <module\_name> ]
- debug i2c read <bus\_no> <i2c\_addr> [ { count <count> } ]
- debug i2c scan <bus\_no> <start\_addr> <addr\_cnt>
- debug i2c write <bus\_no> <i2c\_addr> <data>
- debug icfg debug-trace <cnt> <line>
- debug icfg dir
- debug icfg error-trace <line>
- debug icfg file table
- debug icfg wipe-flash-fs-conf-block
- debug icfg wipe-specific-block { local | global } <block>
- debug icli banner
- debug icli cls [ <n> ]
- debug icli count
- debug icli loop <range> <line>
- debug icli port configuration
- debug icli session close name <username>
- debug icli struct size
- debug init-modules cmd { init | start | conf-def | icfg-loading-pre | icfg-loading-post | suspend-resume } [ isid <isid> ] [ flags <flag> ]
- debug interrupt [ source { [ <intr\_name> ] | [ all ] } ]
- debug interrupt monitor [ source <intr\_name> ]
- debug ip arp inspection entry interface <port\_type> <port\_type\_id> <vlan\_var>

- <mac\_var> <ipv4\_var>
- debug ip dhcp helper frame information
- debug ip dns capability
- debug ip global interface table change
- debug ip http restart
- debug ip http status
- debug ip lpm stat clear <vlan\_list>
- debug ip lpm stat ip <vlan\_list>
- debug ip lpm stat ipv6 <vlan\_list>
- debug ip netlink poll { link | mac | addr4 | addr6 | route4 | route6 | neighbor4 | neighbor6 }
- debug ip port add <chip\_port>
- debug ip source binding dhcp-snooping interface <port\_type> <in\_port\_type\_id> <vlan\_var> <ipv4\_var> <mac\_var>
- debug ip tx [ interface <port\_type> [ <plist> ] ] [ packet-length <packet\_length> ] [ dmac <dmac> ] [ smac <smac> ] [ otag vid <ovid> [ tpid <otpid> ] [ pcp <opcp> ] [ dei <odei> ] ] [ itag vid <ivid> [ tpid <itpid> ] [ pcp <ipcp> ] [ dei <idei> ] ] [ { sip4 <sip4> | sip6 <sip6> } ] [ { dip4 <dip4> | dip6 <dip6> } ] [ dscp <dscp> ] [ dont-fragment ] [ proto { tcp | udp | <ip\_proto> } ] [ sport <sport> ] [ dport <dport> ]
- debug ip vlan ipv4 changed <vlan\_list>
- debug ip vlan ipv4 created <vlan\_list>
- debug ip vlan ipv6 changed <vlan\_list>
- debug ip vlan ipv6 created <vlan\_list>
- debug ip vrrp vlan <vid> [ vrid0 <vrid0> ] [ vrid1 <vrid1> ]
- debug ip-source-guard capabilities
- debug ipmc api { add | del } { ipv4 <v\_ipv4\_ucast> <v\_ipv4\_mcast> | ipv6 <v\_ipv6\_ucast> <v\_ipv6\_mcast> } <v\_vid> <v\_pid>
- debug ipmc mvr thread { resume | suspend }
- debug ipmc packet thread { resume | suspend }
- debug ipmc snooping thread { resume | suspend }
- debug ipv6 dhcp-client capability
- debug irq [ <irq\_name> [ mask | unmask ] ]
- debug kr-aneg [ { all } ] [ { adv-1g } ] [ { adv-2g5 } ] [ { adv-5g } ] [ { adv-10g } ] [ { adv-25g } ] [ { fec-req } ] [ { train } ] [ { no-remote } ] [ { disable } ]
- debug kr-debug [ { enable | disable } ]
- debug kr-pd [ { enable | disable } ]
- debug kr-status [ { eq-hist } ] [ { ber-hist } ] [ { irq } ] [ { details } ]
- debug link-oam dying-gasp add [ port <port> ]
- debug link-oam dying-gasp del <id>
- debug link-oam dying-gasp del all
- debug link-oam dying-gasp-snmp-trap send
- debug link-oam link-event-flags [ { link-fault | dying-gasp | crit-event } ]
- debug link-oam retrieve-mib-variable-data { local | remote }
- debug lldp clear-entries-table
- debug logging erase
- debug logging flash [ category { debug | system | application } ] [ level { informational | notice | warning | error } ]
- debug logging test [ level { informational | notice | warning | error } ] [ target { ram | flash } ] [ repeat <repeat\_cnt> ]



- debug packet rx [ mtu <mtu> ]
- debug packet shaper [ max-kbps <max\_kbps> ]
- debug packet throttle [ xtr-qu <xtr\_qu> ] [ max-frms-per-sec <max\_frms\_per\_sec> ]
- debug pagemap [ process-id <pid> ] [ { perm <perm> } | details ] [ changes-only ] [ section <section> ]
- debug parsing
- debug pgfaults
- debug phy aqr mmd-dump-direct <v\_0\_to\_2> <v\_0\_to\_31>
- debug phy clock <clock\_port> { get | set type { serdes | copper | tclk | xtal | disable } | frequency { 25m | 125m | 3125m } } [ squelch <squelch> ] }
- debug phy cls-45 read page <v\_0\_to\_65535> addr <v\_0\_to\_65535\_1>
- debug phy cls-45 write page <v\_0\_to\_65535> addr <v\_0\_to\_65535\_1> val <v\_0\_to\_65535\_2>
- debug phy csr interface <port\_type> [ <v\_port\_type\_list> ] [ { read <v\_uint8> <v\_uint16> } | { write <v\_uint8\_1> <v\_uint16\_1> <v\_val> } ]
- debug phy do-page-chk [ enable | disable ]
- debug phy event { set { enable | disable } <v\_0\_to\_31> | get | poll }
- debug phy force-pass-through-speed { 1G | 100M | 10M }
- debug phy gpio <gpio\_no> get
- debug phy gpio <gpio\_no> mode { output | input | alternative }
- debug phy gpio <gpio\_no> set { high | low }
- debug phy i2c read <i2c\_mux> <i2c\_device\_addr> <i2c\_reg\_addr> <i2c\_data\_cnt>
- debug phy i2c write <i2c\_mux> <i2c\_device\_addr> <i2c\_reg\_addr> <data>
- debug phy i2c-reset
- debug phy i2c-slave-conf [ <i2c\_slave\_id> ] [ <prescale> ]
- debug phy ib-cterm <ib\_cterm\_value> <ib\_eq\_mode>
- debug phy indy-coma { enable | disable }
- debug phy lcppll-status-get
- debug phy led-set [ led-num <v\_0\_3> ] [ led-mode <v\_0\_22> ]
- debug phy loopback [ near | far | connector | mac-serdes-input | mac-serdes-facility | mac-serdes-equipment | media-serdes-input | media-serdes-facility | media-serdes-equipment | qsgmii-tbi | qsgmii-gmi | qsgmii-serdes ]
- debug phy ltc-freq [ { enable <v\_0\_to\_255> <v\_0\_to\_255\_1> | disable } ]
- debug phy macsec-sd6g-csr <port\_type> [ <v\_port\_type\_list> ] { read | write } { target <v\_target> address <v\_addr> } [ value <v\_value> ]
- debug phy mmd read <devad> <mmd\_reg\_addr>
- debug phy mmd write <devad> <mmd\_reg\_addr> <value>
- debug phy mmd-read-direct <v\_0\_to\_2> <v\_0\_to\_31> <v\_0\_to\_31\_1> <v\_0\_to\_65535>
- debug phy mmd-write-direct <v\_0\_to\_2> <v\_0\_to\_31> <v\_0\_to\_31\_1> <v\_0\_to\_65535> <v\_uint16>
- debug phy mode-set [ media-if <v\_0\_9> ] [ conf-mode <v\_0\_2> ] [ speed { 1G | 100M | 10M } ]
- debug phy neg-1g { status | conf { get | set { ref | client | disable } } }
- debug phy ob-post0 <value>
- debug phy page-check
- debug phy page-check status
- debug phy patch-settings-get

- debug phy qsgmii-aneg { enable | disable }
- debug phy read <addr\_list> [ <page> ] [ addr-sort ]
- debug phy regdump
- debug phy reset
- debug phy ring-resiliency { get | { enable | disable } }
- debug phy serdes1g-rcpll-status-get
- debug phy serdes6g-rcpll-status-get
- debug phy spi-daisy timestamp interface <port\_type> [ <v\_port\_type\_list> ] [ { output { enable | disable } } ] [ { input { enable | disable } } ]
- debug phy statistic
- debug phy tod-time interface <port\_type> [ <v\_port\_type\_list> ] [ { time-sec <v\_0\_to\_255> } ]
- debug phy ts-api action add <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_to\_5> <v\_0\_to\_3\_1> [ { ptp <v\_0\_to\_4> <v\_0\_to\_1> <v\_0\_to\_255> <v\_0\_to\_255\_1> [ DelayReqTs ] } ] | { y1731-oam <v\_0\_to\_1\_1> <v\_0\_to\_255\_2> <v\_0\_to\_255\_3> } | { ietf-oam <v\_0\_to\_1\_2> <v\_0\_to\_7> } | { generic <v\_0\_to\_5\_1> <v\_0\_0xffffffff> <v\_0\_0xffffffff\_1> <v\_0\_0xffffffff\_2> <v\_0\_0xffffffff\_3> <v\_0\_to\_1\_3> <v\_0\_0xff> } ]
- debug phy ts-api action del <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_to\_5>
- debug phy ts-api block init <port\_type> [ <v\_port\_type\_list> ] [ clk-freq <v\_0\_to\_4> ] [ clk-src <v\_0\_to\_5> ] [ rx-ts-pos <v\_0\_to\_1> ] [ tx-fifo-mode <v\_0\_to\_1\_1> ] [ tx-fifo-spi-conf ] [ tx-fifo-hi-clk-cycs <v\_uint8> ] [ tx-fifo-lo-clk-cycs <v\_uint8\_1> ] [ modify-frm <v\_0\_to\_1\_2> ]
- debug phy ts-api channel-map <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ flow <v\_0\_to\_7> ] [ channel-mask <v\_0\_to\_3\_1> ]
- debug phy ts-api clear <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ correction-field { ptp-action <v\_0\_to\_5> } { ptp-packet <v\_0\_to\_3\_1> } ]
- debug phy ts-api comm-conf ach <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_0xff> <v\_0\_0xffff> <v\_0\_0xffff\_1>
- debug phy ts-api comm-conf eth1 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ pbb-en | pbb-dis ] [ etype <v\_0\_0xffff> ] [ tpid <v\_0\_0xffff\_1> ]
- debug phy ts-api comm-conf eth2 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ etype <v\_0\_0xffff> ] [ tpid <v\_0\_0xffff\_1> ]
- debug phy ts-api comm-conf gen-ts <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_1> { ingress | egress } next-proto-offset <v\_0\_0xff> flow-offset <v\_0\_to\_31>
- debug phy ts-api comm-conf ip1 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ ipv4 | ipv6 ] <v\_1\_0xffff> <v\_0\_0xffff> <v\_1\_0xffff\_1> <v\_0\_0xffff\_1>
- debug phy ts-api comm-conf ip2 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ ipv4 | ipv6 ] <v\_1\_0xffff> <v\_0\_0xffff> <v\_1\_0xffff\_1> <v\_0\_0xffff\_1>
- debug phy ts-api comm-conf mpls <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ cw-en ]
- debug phy ts-api delay-asym interface <port\_type> [ <v\_port\_type\_list> ] [ asym <v\_minus\_32768\_to\_32767> ]
- debug phy ts-api fifo { clear | status }
- debug phy ts-api flow-conf eth1 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> {

- ingress | egress } <v\_0\_to\_7> [ enable | disable ] [ match-full | any-uc | any-mc ] [ address <v\_mac\_addr> ] [ match-src | match-dest | match-src-dest ] [ vlan-chk-en | vlan-chk-dis ] [ num-tag <v\_0\_to\_2> ] [ tag-rng-none | tag-rng-outer | tag-rng-inner ] [ tag1-type <v\_1\_to\_4> ] [ tag2-type <v\_1\_to\_4\_1> ] [ tag1-lower <v\_1\_to\_4094> ] [ tag1-upper <v\_1\_0xffff> ] [ tag2-lower <v\_1\_to\_4094\_1> ] [ tag2-upper <v\_1\_0xffff\_1> ]
- debug phy ts-api flow-conf eth2 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_to\_7> [ enable | disable ] [ match-full | any-uc | any-mc ] [ address <v\_mac\_addr> ] [ match-src | match-dest | match-src-dest ] [ vlan-chk-en | vlan-chk-dis ] [ num-tag <v\_0\_to\_2> ] [ tag-rng-none | tag-rng-outer | tag-rng-inner ] [ tag1-type <v\_1\_to\_4> ] [ tag2-type <v\_1\_to\_4\_1> ] [ tag1-lower <v\_1\_to\_4094> ] [ tag1-upper <v\_1\_0xffff> ] [ tag2-lower <v\_1\_to\_4094\_1> ] [ tag2-upper <v\_1\_0xffff\_1> ]
  - debug phy ts-api flow-conf gen-ts <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_1> { ingress | egress } <v\_0\_to\_7> [ enable | disable ] [ dupper <v\_0\_0xffffffff> mask <v\_0\_0xffffffff\_1> ] [ dupper-mid <v\_0\_0xffffffff\_2> mask <v\_0\_0xffffffff\_3> ] [ dlower-mid <v\_0\_0xffffffff\_4> mask <v\_0\_0xffffffff\_5> ] [ dlower <v\_0\_0xffffffff\_6> mask <v\_0\_0xffffffff\_7> ]
  - debug phy ts-api flow-conf ip1 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_to\_7> [ enable | disable ] [ match-src | match-dest | match-src-dest ] [ { ipv4 <v\_ipv4\_addr> <v\_ipv4\_netmask> } | { ipv6 <v\_ipv6\_addr> <v\_ipv6\_netmask> } ]
  - debug phy ts-api flow-conf ip2 <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_to\_7> [ enable | disable ] [ match-src | match-dest | match-src-dest ] [ { ipv4 <v\_ipv4\_addr> <v\_ipv4\_netmask> } | { ipv6 <v\_ipv6\_addr> <v\_ipv6\_netmask> } ]
  - debug phy ts-api flow-conf mpls <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_to\_7> [ enable | disable ] <v\_1\_to\_4> [ stk-ref-top | stk-ref-bottom ] [ stk-lvl-0 <v\_0\_0xffff> ] [ stk-lvl-1 <v\_0\_0xffff\_1> ] [ stk-lvl-2 <v\_0\_0xffff\_2> ] [ stk-lvl-3 <v\_0\_0xffff\_3> ]
  - debug phy ts-api init <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } <v\_0\_to\_14> <v\_0\_to\_7> <v\_0\_to\_7\_1> [ strict | non-strict ]
  - debug phy ts-api interrupt interface <port\_type> [ <v\_port\_type\_list> ] { get | set | clear } <v\_1\_to\_10>
  - debug phy ts-api latency interface <port\_type> [ <v\_port\_type\_list> ] { ingress | egress } [ latency <v\_0\_to\_65535> ]
  - debug phy ts-api mode-set <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_3> { ingress | egress } [ enable | disable ]
  - debug phy ts-api nphase interface <port\_type> [ <v\_port\_type\_list> ] [ nphase <v\_1\_to\_5> ]
  - debug phy ts-api path-delay interface <port\_type> [ <v\_port\_type\_list> ] [ delay <v\_0\_to\_4294967295> ]
  - debug phy ts-api signature interface <port\_type> [ <v\_port\_type\_list> ] [ sig-mask <v\_0\_0x7f> ]
  - debug phy ts-api statistics <port\_type> [ <v\_port\_type\_list> ] [ time <v\_0\_to\_60> ]
  - debug phy ts-api timestamp interface <port\_type> [ <v\_port\_type\_list> ] no-of-timestamps <v\_1\_to\_50>
  - debug phy ts-mepa class get { ing | egr } <v\_0\_to\_23> [ eth | ip ]
  - debug phy ts-mepa class set { ing | egr } <v\_0\_to\_23> encap { none | eth | ip } [ clock <v\_0\_to\_5> ]
  - debug phy ts-mepa class-eth set { ing | egr } <v\_0\_to\_23> [ pbb-en | pbb-dis ] [ etype <v\_0\_0xffff> ] [ tpid <v\_0\_0xffff\_1> ] { match-addr { none | src | dst | src-or-dst } } [

- ```

match-full | any-mac | any-multi | any-uni ] [ mac <v_mac_addr> ] [ vlan-chk-en | vlan-chk-dis
] [ tag-cnt <v_0_to_2> ] [ out-tag { rng-low <v_0_0xFFFF> rng-high <v_0_0xFFFF_1>
| val <v_0_0xFFFF_2> mask <v_0_0xFFFF_3> } ] [ in-tag { rng-low <v_0_0xFFFF_4>
rng-high <v_0_0xFFFF_5> | val <v_0_0xFFFF_6> mask <v_0_0xFFFF_7> } ]

```
- debug phy ts-mepa class-ip set { ing | egr } <v\_0\_to\_23> [ udp-src-port <v\_0\_0xFFFF> ] [ udp-dst-port <v\_0\_0xFFFF\_1> ] [ match-addr { none | src | dest | src-or-dst } ] [ ipv4 { <v\_ipv4\_addr> <v\_ipv4\_netmask> } | ipv6 <v\_ipv6\_addr> <v\_ipv6\_netmask> ]
  - debug phy ts-mepa clock get { ing | egr } <v\_0\_to\_5>
  - debug phy ts-mepa clock set { ing | egr } { clock <v\_0\_to\_5> } { en | dis } [ clk-type { bc1 | bc2 | tc1 | tc2 } ] [ delay-type { e2e | p2p } ] [ cf-update { en | dis } ] [ version-rng low <v\_0\_0xFF> high <v\_0\_0xFF\_1> ] [ minor-vers-rng low <v\_0\_0xFF\_2> high <v\_0\_0xFF\_3> ] [ domain { rng-low <v\_0\_0xFF\_4> rng-high <v\_0\_0xFF\_5> | val <v\_0\_0xFF\_6> mask <v\_0\_0xFF\_7> } ] [ sdo-id { rng-low <v\_0\_0xFF\_8> rng-high <v\_0\_0xFF\_9> | val <v\_0\_0xFF\_10> mask <v\_0\_0xFF\_11> } ]
  - debug phy ts-mepa fifo empty
  - debug phy ts-mepa port-mode { ena | dis }
  - debug phy ts-read <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_7> <v\_0\_0xffff>
  - debug phy ts-write <port\_type> [ <v\_port\_type\_list> ] <v\_0\_to\_7> <v\_0\_0xffff> <v\_0\_0xffffffff>
  - debug phy write <addr\_list> <value> [ <page> ]
  - debug phy-10g apc [ interface <port\_type> [ <port\_list> ] ] [ host | line ] [ auto | manual | freeze | reset | restart | undo-freeze | undo-reset ]
  - debug phy-10g apc-restart [ interface <port\_type> [ <v\_port\_type\_list> ] ] { host | line }
  - debug phy-10g ckout [ { enable | disable } ] [ { use-squelch-src-as-is | invert-squelch-src } ] [ { Squelch-src <Squelch\_src> } ] [ { full-rate-clk | div-by-2-clk } ] [ { clk-out-src <clk\_out\_src> } ]
  - debug phy-10g clause37 [ { host } ] [ { line } ] [ { aneg-enable | aneg-disable } ]
  - debug phy-10g clk-sel [ { line | host } ] [ { clk-sel-no <clk\_sel\_no> } ] [ { clk-sel-val <clk\_sel\_val> } ]
  - debug phy-10g conf { [ mode { one-giga [ lane-0 | lane-3 ] | { repeater <v\_0\_to\_8> } | lan | wan } ] [ host-intf { xaui | { rxau [ ddr-a | ddr-k | ddr-m ] } } ] [ direction { host | line } ] [ clk-src { { lref | href } } ] [ media-type { sr | sr2 | zr | dac | kr | sr-sc | sr2-sc | zr-sc | zr2-sc | dac-sc | kr-sc } ] [ sgmi-pass-thru { enable | disable } ] [ amp-tol { high | low } ] [ link-6g-distance { short-range | long-range } ] [ apc-line-ld-ctrl { ld-lev-ini <v\_0\_0xff> } ] [ apc-offs-ctrl { apc-eqz-offs-par-cfg <v\_0\_0xffffffff> } ] }
  - debug phy-10g failover { a | b | c | d | e | f | get }
  - debug phy-10g fc-buffer-reset
  - debug phy-10g fw-status
  - debug phy-10g gpio <gpio\_no> { [ mode { output | input | alternative } ] [ level { high | low } ] }
  - debug phy-10g gpio-mode interface <port\_type> [ <v\_port\_type\_list> ] <v\_uint16> [ mode <v\_uint8> ] [ port-no <v\_uint> ] [ led-mode <v\_uint\_1> [ blink-time-100ms | blink-time-50ms ] ] [ pgpio-no <v\_uint16\_1> ] [ internal-signal <v\_uint8\_1> ] [ source <v\_uint16\_2> ] [ channel-interrupt <v\_uint8\_2> ] [ port-gpio-interrupt [ set | unset ] ] [ aggr-interrupt <v\_uint\_2> ] [ gpio-input <v\_uint8\_3> ] [ use-as-interrupt-source [ true | false ] ]
  - debug phy-10g ib [ interface <port\_type> [ <port\_list> ] ] [ host | line ] [ auto [ offs [ <offs\_min> <offs\_max> ] | freeze ] [ gain [ <gain\_min> <gain\_max> ] | freeze ] [ gainadj [ <gainadj\_min> <gainadj\_max> ] | freeze ] [ l [ <l\_min> <l\_max> ] | freeze ]

- ```

] [ c [ <c_min> <c_max> ] | freeze ] [ agc [ <agc_min> <agc_max> ] | freeze ] [ dfe1
[ <dfe1_min> <dfe1_max> ] | freeze ] [ dfe2 [ <dfe2_min> <dfe2_max> ] | freeze ] [
dfe3 [ <dfe3_min> <dfe3_max> ] | freeze ] [ dfe4 [ <dfe4_min> <dfe4_max> ] | freeze
] ] [ manual [ offs <offs_value> ] [ gain <gain_value> ] [ gainadj <gainadj_value> ] [ l
<l_value> ] [ c <c_value> ] [ agc <agc_value> ] [ dfe1 <dfe1_value> ] [ dfe2 <dfe2_value>
] [ dfe3 <dfe3_value> ] [ dfe4 <dfe4_value> ] ] [ prbs <pattern> [ invert ] ] [ apc-bit-mask
<apc_bit_mask> ] [ freeze-bit-mask <freeze_bit_mask> ] [ ld <0-63> ]

```
- debug phy-10g interrupt interface <port\_type> [ <v\_port\_type\_list> ] event <v\_1\_to\_32> { enable | disable }
  - debug phy-10g interrupt interface <port\_type> [ <v\_port\_type\_list> ] extended-event <v\_1\_to\_14> { enable | disable }
  - debug phy-10g interrupt interface <port\_type> [ <v\_port\_type\_list> ] poll extended-event <v\_1\_to\_14>
  - debug phy-10g interrupt interface <port\_type> [ <v\_port\_type\_list> ] poll-event <v\_1\_to\_32>
  - debug phy-10g jitter-conf [ interface <port\_type> [ <port\_list> ] ] [ direction [ host | line ] ] [ levn <levn\_val> ] [ incr-levn [ yes | no ] ] [ vtail <vtail\_val> ]
  - debug phy-10g kr-autoneg-train [ { enable | disable } ] [ { line } ] [ { host } ] [ kr-ld-adv-abil [ { adv-1g } ] [ { adv-10g } ] [ { fec-abil } ] [ { fec-req } ] ] [ kr-training [ { enable | disable } ] ] [ tr-method [ cm1 <cm\_1> ] [ c0 <c\_0> ] [ cp1 <c\_1> ] ] [ ld-pre-init { init | preset } ] ]
  - debug phy-10g kr-conf [ cm1 <cm\_1> ] [ c0 <c\_0> ] [ cp1 <c\_1> ] [ ampl <amp\_val> ] [ { ps25 | ps35 | ps55 | ps70 | ps120 } ] [ en-ob | dis-ob ] [ ser-inv | ser-no-inv ] [ host | line ]
  - debug phy-10g lane-sync [ { enable | disable } ] [ { rx-macro <rx\_macro> } { rx-channel-id <rx\_chid> } ] [ { tx-macro <tx\_macro> } { tx-channel-id <tx\_chid> } ]
  - debug phy-10g loopback [ { a | b | c | d | e | f | g | h | j | k | h2 | h3 | h4 | h5 | h6 | l0 | l1 | l2 | l3 | l2c } { enable | disable } ]
  - debug phy-10g pcs1g-sticky poll
  - debug phy-10g pkt-gen interface <port\_type> [ <v\_port\_type\_list> ] { enable { ingress | egress } { { frames { ptp { <v\_uint8> <v\_uint8\_1> <v\_uint8\_2> } | stand } <v\_0\_0xffff> <v\_0\_0xff> <v\_0\_0xffffffff> src-mac <v\_mac\_addr> dest-mac <v\_mac\_addr\_1> | idle } } | disable }
  - debug phy-10g pkt-mon interface <port\_type> [ <v\_port\_type\_list> ] { enable [ { update | reset <v\_0\_to\_6> } ] | disable } [ timestamp ]
  - debug phy-10g power [ enable ]
  - debug phy-10g prbs-counters interface <port\_type> [ <v\_port\_type\_list> ] { host | line }
  - debug phy-10g prbs-gen interface <port\_type> [ <v\_port\_type\_list> ] { line | host } prbsn <v\_0\_to\_5> { enable | disable }
  - debug phy-10g prbs-generator interface <port\_type> [ <v\_port\_type\_list> ] { { serdes prbsn <v\_0\_to\_5> des-interface-width <v\_0\_to\_5\_1> [ input-invert ] } } | pcs } { line | host } [ enable | disable ]
  - debug phy-10g prbs-mon interface <port\_type> [ <v\_port\_type\_list> ] { line | host } prbsn <v\_0\_to\_5> max-bist-frames <v\_0\_to\_0xffff> error-states <v\_0\_to\_3> des-interface-width <v\_0\_to\_5\_1> [ input-invert ] no-of-errors <v\_0\_to\_7> bist-mode <v\_0\_to\_3\_1> { enable | disable }
  - debug phy-10g prbs-monitor interface <port\_type> [ <v\_port\_type\_list> ] { serdes { prbsn <v\_0\_to\_5> max-bist-frames <v\_0\_to\_0xffff> error-states <0-3> des-interface-width <v\_0\_to\_5\_1> [ input-invert ] no-of-errors <v\_0\_to\_7> bist-mode <v\_0\_to\_3\_1> } | pcs } { line | host } [ enable | disable ]

- debug phy-10g prbs-monitor-status interface <port\_type> [ <v\_port\_type\_list> ] { serdes | pcs } [ reset ] { line | host }
- debug phy-10g reset
- debug phy-10g rxckout [ { disable | rx-clock | tx-clock } ] [ { pcs-fault-squelch | no-pcs-fault-squelch } ] [ { lopc-squelch | no-lopc-squelch } ]
- debug phy-10g sckout [ { enable | disable } ] [ { frequency <frequency> } ] [ { use-squelch-src-as-is | invert-squelch-src } ] [ { Squelch-src <Squelch\_src> } ] [ { sckout-clkout-sel <sckout\_clkout\_sel> } ]
- debug phy-10g serdes-status [ interface <port\_type> [ <port\_list> ] ]
- debug phy-10g srefclk [ enable ] [ freq { freq125 | freq15625 | freq15552 } ]
- debug phy-10g status [ interface <port\_type> [ <plist> ] ]
- debug phy-10g status [ interface <port\_type> [ <plist> ] ]
- debug phy-10g synce [ { mode <mode> } [ clk-out <clk\_out> ] [ hitless <hitless> ] [ rclk-div <rclk\_div> ] [ sref-div <sref\_div> ] [ wref-div <wref\_div> ] ]
- debug phy-10g synce-clkout [ enable ]
- debug phy-10g txckout [ { disable | rx-clock | tx-clock } ]
- debug phy-10g vscope interface <port\_type> [ <v\_port\_type\_list> ] { { fast-scan | fast-scan-plus } { host | line } { enable error-threshold <v\_3\_to\_1000> | disable } | xy-scan { host | line } { { enable x-start <v\_0\_to\_127> y-start <v\_0\_to\_63> x-count <v\_0\_to\_127\_1> y-count <v\_0\_to\_63\_1> x-incr <v\_0\_to\_10> y-incr <v\_0\_to\_10\_1> ber <v\_0\_to\_64> error-threshold <v\_3\_to\_1000\_1> } | disable } }
- debug phy-10g xfp-clkout [ enable ]
- debug phy-10g { { read-i2c <address\_read> } | { write-i2c <address\_write> <value> } }
- debug phy-1g ext-interrupt interface <port\_type> [ <v\_port\_type\_list> ] { poll | set | clear } <v\_1\_to\_11>
- debug phy-1g interface <port\_type> [ <v\_port\_type\_list> ] read-tr <v\_0\_0x1fff>
- debug pic erase
- debug pic program [ reboot ]
- debug pic read <offset> <length>
- debug pic verify
- debug port link { enable | disable }
- debug port-security rate-limit { [ fill-level-min <fill\_level\_min> ] [ fill-level-max <fill\_level\_max> ] [ rate <rate> ] [ drop-age <drop\_age> ] }
- debug process cmd <cmd>
- debug process manager
- debug psfptest <cmd>
- debug ptp <clockinst> 802-1as clock-status
- debug ptp <clockinst> 802-1as port-status [ <port\_type> [ <v\_port\_type\_list> ] ]
- debug ptp <clockinst> accuracy <ptp\_accuracy>
- debug ptp <clockinst> class <ptp\_class>
- debug ptp <clockinst> path-trace
- debug ptp <clockinst> slave statistics [ enable | disable | clear ]
- debug ptp <clockinst> unicast-cancel <slave\_idx> { ann | sync | del }
- debug ptp <clockinst> variance <ptp\_variance>
- debug ptp <domain> tod-delta <sec\_msb> <sec\_lsb> <nanosec> [ neg ]
- debug ptp basic-servo hybrid { enable | disable }
- debug ptp egress-latency [ clear ]
- debug ptp ms-pdv log-level <loglevel>

- debug ptp pim statistics [ <port\_type> [ <v\_port\_type\_list> ] ] [ clear ]
- debug ptp pps-tod statistics [ clear ]
- debug ptp servo best-master <clockinst>
- debug ptp timers
- debug pvlan capability
- debug qos queue-shaper queue <queue> { cir <cir> [ cbs <cbs> ] } { [ eir <eir> [ ebs <ebs> ] ] } [ excess | credit ] [ rate-type { line | data } ]
- debug qos registrations [ clear ]
- debug qos shaper cir { <cir> [ cbs <cbs> ] } { [ eir <eir> [ ebs <ebs> ] ] } { [ rate-type { line | data } ] }
- debug qs [ mode <mode> ] [ <wm1> ] [ <wm2> ]
- debug resume
- debug serdes [ { dfe <h1> <h2> <h3> <h4> <h5> <dlev> | ctle <vga\_r> <vga\_c> <c> <gain> | txeq <tap\_dly> <tap\_adv> <amplitude> } ]
- debug sfp dump [ interface <port\_type> [ <port\_list> ] ]
- debug sfp read <i2c\_addr> <i2c\_bytes> <port\_no> <sfp\_reg\_addr\_0> [ <sfp\_reg\_addr\_1> ]
- debug sfp write <i2c\_addr> <port\_no> <sfp\_reg\_addr> <i2c\_data\_0> <i2c\_data\_1>
- debug sgpio conf group <group\_no> port-mask <port\_mask> bits-per-port <bit\_count>
- debug sgpio read [ group <group> ]
- debug sgpio write [ group <group> ] <data\_1> [ <data\_2> [ <data\_3> [ <data\_4> ] ] ]
- debug show aggregation
- debug show aps capabilities
- debug show aps history
- debug show aps rules
- debug show aps timers
- debug show capabilities
- debug show cfm capabilities
- debug show cfm rules
- debug show cfm timers
- debug show eee { { phy [ raw ] } | status } [ interface <port\_type> [ <port\_list> ] ]
- debug show erps capabilities
- debug show erps history
- debug show erps rules
- debug show erps timers
- debug show firmware mfi-info
- debug show frame tx-cnt
- debug show gvrp config
- debug show interface <port\_type> [ <port\_list> ] oper-state
- debug show interface <port\_type> [ <port\_list> ] state
- debug show ip capabilities
- debug show ip dhcp pool counter [ <pool\_name> ]
- debug show ip http secure-certificate
- debug show ip statistics [ interface vlan <vlan\_list> ]
- debug show irq
- debug show lacp
- debug show mac address-table { [ max-count <v\_uint> ] | [ stack ] }
- debug show macsec [ interface <port\_type> [ <v\_port\_type\_list> ] ]

- debug show media-redundancy capabilities
- debug show media-redundancy history
- debug show media-redundancy rules
- debug show media-redundancy state
- debug show media-redundancy timers
- debug show monitor capabilities reflect-port
- debug show mvrp msti-port-ring [ msti <instance> ]
- debug show mvrp state-machines interface <port\_type> [ <plist> ] vlan <vlist>
- debug show phy ts-api interface <port\_type> [ <v\_port\_type\_list> ] engine <v\_0\_to\_3> { ingress | egress } { eng-init | comm-conf flow <v\_0\_to\_7> { eth1 | eth2 | ip1 | ip2 | mpls | ach | gen-ts } | flow-conf flow <v\_0\_to\_7\_1> { eth1 | eth2 | ip1 | ip2 | mpls | gen-ts } | action { ptp | oam | generic } }
- debug show port capabilities
- debug show port registrations
- debug show port-security [ interface <port\_type> [ <plist> ] ]
- debug show port-security address [ interface <port\_type> [ <plist> ] ]
- debug show port-security rate-limit config
- debug show port-security rate-limit statistics
- debug show port-security ref-cnt [ interface <port\_type> [ <plist> ] ]
- debug show qos shapers
- debug show sgpio [ group <group> ] [ changes-only ]
- debug show snmp mib redefine
- debug show snmp prefix <subtree>
- debug show snmp users
- debug show timers
- debug show tsn frer capabilities
- debug show tsn frer rules
- debug show tsn status
- debug show tsn timers
- debug show vce registry
- debug show vcl ip-subnet [ <ipv4> ]
- debug show vcl mac [ address <mac\_addr> ]
- debug show vcl policy
- debug show vcl protocol [ eth2 { <etype> | arp | ip | ipx | at } ] [ snap { <oui> | rfc-1042 | snap-8021h } <pid> ] [ llc <dsap> <ssap> ]
- debug show vlan bulk-update
- debug show vlan config
- debug show vlan counter <vid>
- debug show vlan port-config interface <port\_type> [ <plist> ]
- debug show vlan tag <vid> interface <port\_type> [ <plist> ]
- debug slabinfo [ changes-only ]
- debug sleep <msecs>
- debug snmp netsnmp <token>
- debug snmp translate [ <subtree> ]
- debug snmp trap listen <trap\_name>
- debug snmp trap send
- debug spi-transfer <v\_1\_to\_3> <v\_0\_to\_1> <v\_uint> <v\_uint\_1> <v\_0\_to\_50> [ <v\_hexval> ]

- debug stp port <instance> <interface>
- debug stp pstate <instance> <interface>
- debug suspend
- debug sym query <pattern>
- debug sym read <pattern>
- debug sym write <pattern> <value>
- debug synce subject <v\_name> <v\_index> write <v\_value>
- debug synce-cpld read addr <v\_0\_0xFF>
- debug synce-cpld write addr <v\_0\_0xFF> value <v\_0\_0xFF\_1>
- debug system led post
- debug system led set { off | green | red | yellow }
- debug system led status test [ fatal | software | post | ztp | stack-firmware ]
- debug system shell [ <shell\_cmd> ]
- debug temperature monitor { set [ board <b\_temp> ] [ junction <j\_temp> ] | show }
- debug testing-script ifindex interface <port\_type> <port>
- debug testing-script ifindex usid <usid> uport <uport>
- debug testing-script ifindex vlan <vid>
- debug testing-script portlist-index <port\_type> [ <plist> ]
- debug testing-script portlist-stackable { [ <port\_type> [ <plist> ] ] [ CPU ] }
- debug testing-script synce-clock-type
- debug testing-script uport <port\_type> <port>
- debug testing-script vlanlist <vlanlist>
- debug thread load-monitor
- debug thread priority set <thread\_id> { below | normal | above | high | higher | highest } [ real-time ]
- debug thread { { status [ backtrace ] [ id { all | <tid> } ] } | context-switches }
- debug time get
- debug time set <d> <t>
- debug timer clear
- debug timer measure <id>
- debug timer test start timer-cnt <timer\_cnt> duration-ms <duration\_ms> period-ms <period\_ms> itr-cnt <itr\_cnt> prio { below-normal | default | above-normal | high | higher | highest | below-normal-rt | default-rt | above-normal-rt | high-rt | higher-rt | highest-rt }
- debug timer test status handle <handle>
- debug tod freq-adjust <v\_minus\_65535\_to\_65535>
- debug tod monitor { enable | disable }
- debug tod phy ts [ enable | disable ]
- debug trace configuration { read | write | erase }
- debug trace global level [ none | error | warning | info | debug | noise | racket ]
- debug trace hunt <target>
- debug trace module level [ { <module\_name> | \* } [ { <group\_name> | \* } [ none | error | warning | info | debug | noise | racket ] ] ]
- debug trace module ringbuffer <module\_name> <group\_name> { enable | disable }
- debug trace module usec <module\_name> <group\_name> { enable | disable }
- debug trace port [ enable | disable ]
- debug trace ringbuffer flush
- debug trace ringbuffer print
- debug trace ringbuffer { start | stop }

- debug trace vtss-basics [ fatal | error | warning | info | debug | noise | racket ]
- debug trace { revert | defaults }
- debug tsn frame-preemption [ enable { tx-on | tx-off } ] [ verify-disable { on | off } ] [ queue <queue> ] [ verify-time <verify\_time> ] [ frag-size <frag\_size> ]
- debug tsn frame-preemption add-frag-size <size>
- debug uboot printenv
- debug ufdma
- debug vcl address [ interface <port\_type> [ <v\_port\_type\_list> ] ] [ set { dmacdip | smacsip } ]
- debug vcl policy <policy\_no>
- debug vlan add <vid> interface <port\_type> [ <plist> ] { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror }
- debug vlan bulk-update { enable | disable }
- debug vlan del <vid> <slist> { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror }
- debug vlan empty
- debug vlan fillup <vid>
- debug vlan frame-type interface <port\_type> [ <plist> ] { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror } { all | tagged | untagged }
- debug vlan ingress-filter interface <port\_type> [ <plist> ] { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror } { enable | disable }
- debug vlan port-del interface <port\_type> [ <plist> ] { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror }
- debug vlan port-type interface <port\_type> [ <plist> ] { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror } { unaware | c-port | s-port | s-custom-port }
- debug vlan pvid interface <port\_type> [ <plist> ] { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror } <pvid>
- debug vlan tag <vid> interface <port\_type> [ <plist> ] { port | enable | disable }
- debug vlan tx-tag interface <port\_type> [ <plist> ] { static | dot1x | mvr | voice-vlan | mstp | erps | mrp | vcl | gvrp | rmirror } { tag-this <tag\_vid> | untag-this <untag\_vid> | tag-all | untag-all }
- debug vmstat [ changes-only ]
- debug voice vlan { lldp [ interface <port\_type> [ <port\_list> ] ] | capability }
- debug vtss-ifindex-getnext { port | llag | vlan } { first | <previous\_> }
- debug vtss-ifindex-iterate { all-type | port | llag | vlan } { first | <previous\_> } [ check-exist ] [ iterate-cnt <iterate\_cnt> ]
- debug wis atti interface <port\_type> [ <v\_port\_type\_list> ] [ { overhead <v\_0\_to\_1> } ]
- debug wis cons-act <port\_type> [ <v\_port\_type\_list> ] [ { enable <v\_0\_0x3> <v\_0\_0xf> <v\_0\_0x7ff> } ]
- debug wis counters interface <port\_type> [ <v\_port\_type\_list> ]
- debug wis defects interface <port\_type> [ <v\_port\_type\_list> ]
- debug wis event-force interface <port\_type> [ <v\_port\_type\_list> ] <v\_0\_0xffffffff> { enable | disable }
- debug wis force-conf interface <port\_type> [ <v\_port\_type\_list> ] [ { set <v\_0\_to\_3> <v\_0\_to\_3\_1> <v\_0\_to\_3\_2> } ]
- debug wis mode interface <port\_type> [ <v\_port\_type\_list> ] [ enable | disable ]
- debug wis perf-counters interface <port\_type> [ <v\_port\_type\_list> ]
- debug wis perf-thr interface <port\_type> [ <v\_port\_type\_list> ] [ { set <v\_0\_0xffffffff> } ]

- <v\_0\_0xffffffff\_1> <v\_0\_0xffffffff\_2> <v\_0\_0xffffffff\_3> <v\_0\_0xffffffff\_4> } ]
- debug wis prbs31-err-inj interface <port\_type> [ <v\_port\_type\_list> ] { single-erro | sat-erro }
- debug wis reset interface <port\_type> [ <v\_port\_type\_list> ]
- debug wis status interface <port\_type> [ <v\_port\_type\_list> ]
- debug wis test-mode interface <port\_type> [ <v\_port\_type\_list> ] [ loopback | no-loopback ] [ gen-dis | gen-sqr | gen-prbs31 | gen-mix ] [ ana-dis | ana-prbs31 | ana-mix ]
- debug wis test-status interface <port\_type> [ <v\_port\_type\_list> ]
- debug wis tx-oh-conf interface <port\_type> [ <v\_port\_type\_list> ] [ { set { all | sec-oh | line-oh | path-oh | d1-d3 | sec-ord | suc | res-sg | d4-d12 | line-ord | aps-rdil | sync | res-lg | c2pl | puc | ptcn | res-pg } { <v\_word> } } ]
- debug wis txtti interface <port\_type> [ <v\_port\_type\_list> ] [ set <v\_0\_to\_1> <v\_0\_to\_2> <v\_word> ]
- debug zl3036x dpll cfg <pllid>
- debug zl3036x dpll status <pllid>
- debug zl3036x ncofreq
- debug zl3036x pll status
- debug zl3036x read <v\_uint> <v\_uint\_1> <v\_uint\_2>
- debug zl3036x ref cfg <refid>
- debug zl3036x ref status <refid>
- debug zl3036x trace all <v\_uint\_1>
- debug zl3036x trace module <v\_uint\_1> <v\_uint\_2>
- debug zl3036x write <v\_uint> <v\_uint\_1> <v\_uint\_2>
- debug zl3077x dpll cfg <pllid>
- debug zl3077x dpll status <pllid>
- debug zl3077x pll status
- debug zl3077x print-mailbox { [ ref <v\_0\_to\_9> ] | [ dpll <v\_1\_to\_4> ] }
- debug zl3077x read <v\_uint> <v\_uint\_1> <v\_uint\_2>
- debug zl3077x ref cfg <refid>
- debug zl3077x ref status <refid>
- debug zl3077x trace module <v\_0\_to\_37> <v\_0\_to\_5>
- debug zl3077x write <v\_uint> <v\_uint\_1> <v\_uint\_2>
- debug zl3077x-spi read page <v\_0\_to\_14> addr <v\_0\_0xFFFF> cnt <v\_0\_to\_128>
- debug zl3077x-spi write page <v\_0\_to\_14> addr <v\_0\_0xFFFF> val <v\_0\_to\_255>
- debug zls30387 cgu <v\_0\_to\_2> server <v\_0\_to\_3> cmd <v\_0\_to\_1000> [ val <v\_minus\_32000\_to\_32000> ]
- default access-list rate-limiter [ <rate\_limiter\_list> ]
- default range <entry\_name>
- default-information originate
- default-information originate [ always ] [ metric <metric\_val> ] [ metric-type { 1 | 2 } ]
- default-metric <def\_metric>
- default-metric <metric\_val>
- default-router <ip> [ <ip1> [ <ip2> [ <ip3> ] ] ]
- delete <path>
- description <dscr>
- description <profile\_desc>
- dir
- direction { up | down }

- disable [ <new\_priv> ]
- distance <admin\_distance>
- distance <admin\_distance>
- distance <admin\_distance>
- dmac { <dmac\_addr> [ / <mask\_mac\_addr> ] | multicast | broadcast | any }
- dns-server <ip> [ <ip1> [ <ip2> [ <ip3> ] ] ]
- do <command>
- domain-name <domain\_name>
- dot1x authentication timer inactivity <v\_10\_to\_100000>
- dot1x authentication timer re-authenticate <v\_1\_to\_3600>
- dot1x feature { [ guest-vlan ] [ radius-qos ] [ radius-vlan ] }
- dot1x guest-vlan
- dot1x guest-vlan <value>
- dot1x guest-vlan supplicant
- dot1x initialize [ interface <port\_type> [ <plist> ] ]
- dot1x max-reauth-req <value>
- dot1x port-control { force-authorized | force-unauthorized | auto | single | multi | mac-based }
- dot1x radius-qos
- dot1x radius-vlan
- dot1x re-authenticate
- dot1x re-authentication
- dot1x system-auth-control
- dot1x timeout quiet-period <v\_10\_to\_1000000>
- dot1x timeout tx-period <v\_1\_to\_65535>
- drop-on-yellow
- duplex { half | full | { auto [ half | full ] } }
- ebs <ebs>
- editing
- egress interface <port\_type> [ <port\_list> ]
- eir <eir>
- enable
- enable [ <new\_priv> ]
- enable password [ level <priv> ] <password>
- enable secret { 0 | 5 } [ level <priv> ] <password>
- erps <inst>
- erps <inst> clear
- erps <inst> switch { force | manual } { port0-to-port1 | port1-to-port0 }
- etype <etype\_type>
- excessive-restart
- exec-banner
- exec-timeout <min> [ <sec> ]
- fec { auto | r-fec | rs-fec | none }
- firmware swap
- firmware upgrade <url\_file> [ save-host-key ] [ ftp-active ]
- flooding
- flow-meter id <id>
- flowcontrol { on | off }
- format { none | string <name> }

- format { string <format\_string> | integer <format\_integer> | primary-vid | icc <format\_icc\_string> | icc-cc <format\_icc\_cc\_string> }
- frame-length-check
- frer-vlan <vid>
- gate id <id>
- green-ethernet eee
- green-ethernet eee optimize-for-power
- green-ethernet eee urgent-queues [ <urgent\_queue\_range\_list> ]
- green-ethernet energy-detect
- green-ethernet fan { [ temp-on <temp\_on\_num> ] [ temp-max <temp\_max\_num> ] }
- green-ethernet led interval <v\_0\_to\_24> intensity <v\_0\_to\_100>
- green-ethernet led on-event { [ link-change <v\_0\_to\_65535> ] [ error ] }
- green-ethernet short-reach
- guard-time <guard\_time>
- gvrp
- gvrp join-request vlan <v\_vlan\_list>
- gvrp leave-request vlan <v\_vlan\_list>
- gvrp max-vlans <maxvlans>
- gvrp time { [ join-time <join\_time> ] [ leave-time <leave\_time> ] [ leave-all-time <leave\_all\_time> ] }
- hardware-address <mac>
- history size <history\_size>
- hold-off-time <hold\_off>
- hold-off-time <hold\_off>
- host <ip> <subnet\_mask>
- host <v\_ipv6\_ucast> [ <udp\_port> ] [ traps | informs ]
- host { <v\_ipv4\_ucast> | <v\_word> } [ <udp\_port> ] [ traps | informs ]
- hostname <hostname>
- informs retries <retries> timeout <timeout>
- ingress stream-id-list <stream\_list> inner-tag [ vid { <vid> [ / <vid\_mask> ] | any } ] [ pcp <pcp\_val> [ / <pcp\_mask> ] ] [ dei <dei\_val> ] [ { c-tag | s-tag } ]
- interconnection control-vlan <vid>
- interconnection id <id>
- interconnection interface <port\_type> <port>
- interconnection mode { link-check | ring-check }
- interconnection name <name>
- interconnection recovery-profile { 200ms | 500ms }
- interconnection role { mic | mim | none }
- interconnection sf-trigger { link | { mep domain <md\_name> service <ma\_name> mep-id <mepid> } }
- interface <port\_type> <port>
- interface <port\_type> [ <plist> ]
- interface llag <llag\_id>
- interface vlan <vlan\_list> area <area\_id>
- interface vlan <vlist>
- interface-status-tlv { disable | enable | defer }
- interface-status-tlv { disable | enable | defer }
- ip address <subnet>

- ip address { { <address> <netmask> } | { dhcp [ fallback <fallback\_address> <fallback\_netmask> [ timeout <fallback\_timeout> ] ] [ client-id { <port\_type> <client\_id\_interface> | ascii <ascii\_str> | hex <hex\_str> } ] [ hostname <hostname> ] } }
- ip address { { <subnet> } | { dhcp [ fallback <fallback\_subnet> [ timeout <fallback\_timeout> ] ] [ client-id { <port\_type> <client\_id\_interface> | ascii <ascii\_str> | hex <hex\_str> } ] [ hostname <hostname> ] } }
- ip arp inspection
- ip arp inspection check-vlan
- ip arp inspection entry interface <port\_type> <in\_port\_type\_id> <vlan\_var> <mac\_var> <ipv4\_var>
- ip arp inspection logging { deny | permit | all }
- ip arp inspection translate [ interface <port\_type> <in\_port\_type\_id> <vlan\_var> <mac\_var> <ipv4\_var> ]
- ip arp inspection trust
- ip arp inspection vlan <in\_vlan\_list>
- ip arp inspection vlan <in\_vlan\_list> logging { deny | permit | all }
- ip dhcp excluded-address <low\_ip> [ <high\_ip> ]
- ip dhcp pool <pool\_name>
- ip dhcp relay
- ip dhcp relay information option
- ip dhcp relay information policy { drop | keep | replace }
- ip dhcp retry interface vlan <vlan\_id>
- ip dhcp server
- ip dhcp server
- ip dhcp snooping
- ip dhcp snooping trust
- ip dns proxy
- ip domain name { <v\_domain\_name> | dhcp [ ipv4 | ipv6 ] [ interface vlan <v\_vlan\_id\_dhcp> ] }
- ip helper-address <v\_ipv4\_ucast>
- ip http secure-certificate { upload <url\_file> [ pass-phrase <pass\_phrase> ] | delete | generate }
- ip http secure-redirect
- ip http secure-server
- ip igmp host-proxy [ leave-proxy ]
- ip igmp snooping
- ip igmp snooping
- ip igmp snooping compatibility { auto | v1 | v2 | v3 }
- ip igmp snooping filter <profile\_name>
- ip igmp snooping immediate-leave
- ip igmp snooping last-member-query-interval <ipmc\_lmqi>
- ip igmp snooping max-groups <throttling>
- ip igmp snooping mrouter
- ip igmp snooping priority <cos\_priority>
- ip igmp snooping querier { election | address <v\_ipv4\_ucast> }
- ip igmp snooping query-interval <ipmc\_qi>
- ip igmp snooping query-max-response-time <ipmc\_qri>

- ip igmp snooping robustness-variable <ipmc\_rv>
- ip igmp snooping unsolicited-report-interval <ipmc\_uri>
- ip igmp snooping vlan <vlan\_list>
- ip igmp ssm-range <v\_ipv4\_mcast> <ipv4\_prefix\_length>
- ip igmp unknown-flooding
- ip name-server [ <order> ] { <v\_ipv4\_ucast> | { <v\_ipv6\_ucast> [ interface vlan <v\_vlan\_id\_static> ] } | dhcp [ ipv4 | ipv6 ] [ interface vlan <v\_vlan\_id\_dhcp> ] }
- ip ospf authentication [ null | message-digest ]
- ip ospf authentication-key { unencrypted <unencrypted\_pwd> | encrypted <encrypted\_pwd> }
- ip ospf message-digest-key <md\_key\_id> md5 { unencrypted <unencrypted\_pwd> | encrypted <encrypted\_pwd> }
- ip ospf { priority <priority> | cost <cost> | hello-interval <hello\_interval> | retransmit-interval <retransmit\_interval> | dead-interval { <dead\_interval> | minimal hello-multiplier <fast\_hello\_packets> } }
- ip rip authentication key-chain <key\_chain\_str>
- ip rip authentication mode { text | md5 }
- ip rip authentication string { unencrypted <unencrypted\_pwd> | encrypted <encrypted\_pwd> }
- ip rip receive version { none | 1 [ 2 ] | 2 [ 1 ] }
- ip rip send version { 1 [ 2 ] | 2 [ 1 ] }
- ip rip split-horizon [ poisoned-reverse ]
- ip route <v\_ipv4\_addr> <v\_ipv4\_netmask> <v\_ipv4\_gw> [ distance <v\_distance> ]
- ip route <v\_ipv4\_subnet> <v\_ipv4\_gw> [ distance <v\_distance> ]
- ip routing
- ip source binding interface <port\_type> <in\_port\_type\_id> <vlan\_var> <ipv4\_var> <mac\_var>
- ip ssh
- ip verify source
- ip verify source
- ip verify source limit <cnt\_var>
- ip verify source translate
- ipmc profile
- ipmc profile <profile\_name>
- ipmc range <entry\_name> { <v\_ipv4\_mcast> [ <v\_ipv4\_mcast\_1> ] | <v\_ipv6\_mcast> [ <v\_ipv6\_mcast\_1> ] }
- ipv <ipv>
- ipv4 [ proto { <pr4> | tcp | udp | any } ] [ sip { <sip4> | any } ] [ dip { <dip4> | any } ] [ dscp { <dscp4> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } | any } ] [ fragment { yes | no | any } ] [ dport { <dp4> | any } ]
- ipv6 [ proto { <pr6> | tcp | udp | any } ] [ sip { <sip6> | any } ] [ dip { <dip6> | any } ] [ dscp { <dscp6> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } | any } ] [ dport { <dp6> | any } ]
- ipv6 address <subnet>
- ipv6 address { autoconfig | dhcp [ rapid-commit ] }
- ipv6 dhcp relay [ destination <v\_ipv6\_ucast> ] interface vlan <v\_vlan\_id>
- ipv6 dhcp snooping

- ipv6 dhcp snooping nh-unknown { drop | allow }
- ipv6 dhcp snooping trust
- ipv6 dhcp-client restart [ interface vlan <v\_vlan\_list> ]
- ipv6 mld host-proxy [ leave-proxy ]
- ipv6 mld snooping
- ipv6 mld snooping
- ipv6 mld snooping compatibility { auto | v1 | v2 }
- ipv6 mld snooping filter <profile\_name>
- ipv6 mld snooping immediate-leave
- ipv6 mld snooping last-member-query-interval <ipmc\_lmqi>
- ipv6 mld snooping max-groups <throttling>
- ipv6 mld snooping mrouter
- ipv6 mld snooping priority <cos\_priority>
- ipv6 mld snooping querier election
- ipv6 mld snooping query-interval <ipmc\_qi>
- ipv6 mld snooping query-max-response-time <ipmc\_qri>
- ipv6 mld snooping robustness-variable <ipmc\_rv>
- ipv6 mld snooping unsolicited-report-interval <ipmc\_uri>
- ipv6 mld snooping vlan <vlan\_list>
- ipv6 mld ssm-range <v\_ipv6\_mcast> <ipv6\_prefix\_length>
- ipv6 mld unknown-flooding
- ipv6 ospf { passive | priority <priority> | cost <cost> | hello-interval <hello\_interval> | retransmit-interval <retransmit\_interval> | transmit-delay <transmit\_delay> | dead-interval { <dead\_interval> } }
- ipv6 route <v\_ipv6\_subnet> <v\_ipv6\_ucast> [ interface vlan <v\_vlan\_id> ] [ distance <v\_distance> ]
- ipv6 source binding interface <port\_type> <port\_type\_id> [ vlan <vlan\_id> ] <ipv6\_ucast> <mac\_ucast>
- ipv6 verify source
- ipv6 verify source
- ipv6 verify source limit <max\_dynamic\_clients>
- ipv6 verify source translate
- json notification host <hname>
- json notification listen <notification> <host>
- key <key\_id> key-string { unencrypted <plaintext\_pwd> | encrypted <encrypted\_pwd> }
- key chain <keychain\_name>
- key { class | class-dpl | dscp | dscp-dpl }
- key { pcp | pcp-dei | dscp | dscp-pcp-dei }
- lacp
- lacp failover { revertive | non-revertive }
- lacp max-bundle <v\_uint>
- lacp port-priority <v\_1\_to\_65535>
- lacp system-priority <v\_1\_to\_65535>
- lacp timeout { fast | slow }
- lease { <day> [ <hour> [ <min> ] ] | infinite }
- length <length>
- line { <0 16> | console 0 | vty <0 15> }
- link-oam

- link-oam link-monitor frame { [ window <error\_window> ] [ threshold <error\_threshold> ] }
- link-oam link-monitor frame-seconds { [ window <error\_window> ] [ threshold <error\_threshold> ] }
- link-oam link-monitor supported
- link-oam link-monitor symbol-period { [ window <error\_window> ] [ threshold <error\_threshold> ] }
- link-oam mib-retrieval supported
- link-oam mode { active | passive }
- link-oam remote-loopback supported
- link-oam remote-loopback { start | stop } interface <port\_type> [ <v\_port\_type\_list> ]
- link-oam variable-retrieve { local-info | remote-info }
- llc <llc\_dsap> <llc\_ssap>
- lldp cdp-aware
- lldp holdtime <val>
- lldp med datum { wgs84 | nad83-navd88 | nad83-mlw }
- lldp med fast <v\_1\_to\_10>
- lldp med location-tlv altitude { meters | floors } <v\_word11>
- lldp med location-tlv civic-addr { { country <country> } | { state | county | city | district | block | street | leading-street-direction | trailing-street-suffix | street-suffix | house-no | house-no-suffix | landmark | additional-info | name | zip-code | building | apartment | floor | room-number | place-type | postal-community-name | p-o-box | additional-code } <v\_line> }
- lldp med location-tlv elin-addr <v\_word25>
- lldp med location-tlv latitude { north | south } <v\_word8>
- lldp med location-tlv longitude { west | east } <v\_word9>
- lldp med media-vlan policy-list <v\_range\_list>
- lldp med media-vlan-policy <policy\_index> { voice | voice-signaling | guest-voice-signaling | guest-voice | softphone-voice | video-conferencing | streaming-video | video-signaling } { untagged | tagged <v\_vlan\_id> [ l2-priority <v\_0\_to\_7> ] } [ dscp <v\_0\_to\_63> ]
- lldp med transmit-tlv [ capabilities ] [ location ] [ network-policy ] [ poe ]
- lldp med type { connectivity | end-point }
- lldp receive
- lldp reinit <val>
- lldp timer <val>
- lldp tlv-select { management-address | port-description | system-capabilities | system-description | system-name }
- lldp transmission-delay <val>
- lldp transmit
- lldp trap
- location <location>
- logging host { <ipv4\_addr> | <domain\_name> }
- logging level { informational | notice | warning | error }
- logging notification listen <name> level { informational | notice | warning | error } <node>
- logging on
- logout
- loop-protect
- loop-protect action { [ shutdown ] [ log ] }
- loop-protect shutdown-time <t>
- loop-protect transmit-time <t>

- loop-protect tx-mode
- mac address-table aging-time <v\_0\_10\_to\_1000000>
- mac address-table learning [ secure ]
- mac address-table learning vlan <vlan\_list>
- mac address-table static <v\_mac\_addr> vlan <v\_vlan\_id> [ interface <port\_type> [ <v\_port\_type\_list> ] ]
- map { { dscp { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } } } | { pcp <pcp\_num> [ dei <dei\_num> ] } } to { [ class <cosid> ] [ cos <cos> ] [ dpl <dpl> ] [ pcp <pcp> ] [ dei <dei> ] [ dscp <dscp> ] }
- map { { { dscp { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } } } | { class <cosid\_num> } [ dpl <dpl\_num> ] } to { [ pcp <pcp> ] [ dei <dei> ] [ dscp <dscp> ] }
- mark-red-enable
- max-metric router-lsa { [ on-startup <on\_startup\_interval> ] [ on-shutdown <on\_shutdown\_interval> ] [ administrative ] }
- max-sdu <sz>
- media-redundancy <inst>
- media-type { rj45 | sfp | dual | dac-1m | dac-2m | dac-3m | dac-5m }
- mep <mepid>
- mep <mepid>
- mode { 1-for-1 | bidirectional-1-plus-1 | unidirectional-1-plus-1 [ tx-aps ] }
- mode { generation | recovery }
- monitor session <session\_number> [ destination { interface <port\_type> [ <di\_list> ] | remote vlan <drvid> reflector-port <port\_type> <rportid> } | source { interface <port\_type> [ <si\_list> ] [ both | rx | tx ] | remote vlan <srvid> | vlan <source\_vlan\_list> | cpu [ both | rx | tx ] } ]
- monitor session <session\_number> inner egress-tag [ none | all ]
- more <path> [ save-host-key ] [ ftp-active ]
- motd-banner
- mrm priority <prio>
- mrm react-on-link-change
- mrp periodic
- mrp timers default
- mrp timers { [ join-time <jointime> ] [ leave-time <leavetime> ] [ leave-all-time <leavealltime> ] }
- mtu <max\_length>
- mvr
- mvr immediate-leave
- mvr name <mvr\_name> channel <profile\_name>
- mvr name <mvr\_name> frame priority <cos\_priority>
- mvr name <mvr\_name> frame tagged
- mvr name <mvr\_name> last-member-query-interval <ipmc\_lmqi>
- mvr name <mvr\_name> mode { dynamic | compatible }
- mvr name <mvr\_name> type { source | receiver }
- mvr name <mvr\_name> { election | igmp-address <v\_ipv4\_ucast> }
- mvr vlan <v\_vlan\_list> [ name <mvr\_name> ]
- mvr vlan <v\_vlan\_list> channel <profile\_name>

- mvr vlan <v\_vlan\_list> frame priority <cos\_priority>
- mvr vlan <v\_vlan\_list> frame tagged
- mvr vlan <v\_vlan\_list> last-member-query-interval <ipmc\_lmqi>
- mvr vlan <v\_vlan\_list> mode { dynamic | compatible }
- mvr vlan <v\_vlan\_list> type { source | receiver }
- mvr vlan <v\_vlan\_list> { election | igmp-address <v\_ipv4\_ucast> }
- mvrp
- mvrp mvrp managed vlan { all | none | [ add | remove | except ] <vlist> }
- name <name>
- name <vlan\_name>
- neighbor <ip\_addr>
- netbios-name-server <ip> [ <ip1> [ <ip2> [ <ip3> ] ] ]
- netbios-node-type { b-node | h-node | m-node | p-node }
- netbios-scope <netbios\_scope>
- network <ip> <subnet\_mask>
- network <ip\_address>
- network <ip\_address> <wildcard\_mask>
- network <ip\_address> <wildcard\_mask> area <area\_id>
- network-clock clk-source <clk\_list> nominate { clk-in | { ptp <ptp\_inst> } | { interface <port\_type> <port> } }
- network-clock clk-source <clk\_src> aneg-mode { master | slave | forced }
- network-clock clk-source <clk\_src> hold-timeout <v\_3\_to\_18>
- network-clock clk-source <clk\_src> priority <prio>
- network-clock clk-source <clk\_src> ssm-overwrite { prc | ssua | ssub | eec2 | eec1 | dnu | prs | stu | st2 | tnc | st3e | smc | prov | dus }
- network-clock input-source { 1544khz | 2048khz | 10mhz }
- network-clock option { eec1 | eec2 }
- network-clock output-source { 1544khz | 2048khz | 10mhz }
- network-clock selector { { manual clk-source <v\_uint> } | selected | nonrevertive | revertive | holdover | freerun }
- network-clock ssm-freerun { prc | ssua | ssub | eec2 | eec1 | dnu | inv | prs | stu | st2 | tnc | st3e | smc | prov | dus }
- network-clock ssm-holdover { prc | ssua | ssub | eec2 | eec1 | dnu | inv | prs | stu | st2 | tnc | st3e | smc | prov | dus }
- network-clock synchronization ssm
- network-clock wait-to-restore <wtr\_value>
- nis-domain-name <domain\_name>
- nis-server <ip> [ <ip1> [ <ip2> [ <ip3> ] ] ]
- no aaa accounting { console | telnet | ssh }
- no aaa authentication login { console | telnet | ssh | http }
- no aaa authorization { console | telnet | ssh }
- no access management
- no access management <access\_id\_list>
- no access-list ace <ace\_list>
- no access-list logging
- no access-list mirror
- no access-list policy
- no access-list port-state

- no access-list rate-limiter
- no access-list rate-limiter [ <rate\_limiter\_list> ]
- no access-list redirect
- no access-list shutdown
- no action
- no address <v\_ipv4\_addr>
- no aggregation group <v\_uint>
- no aggregation mode
- no alarm [ <alarm\_name> ]
- no alarm suppress <alarm\_name>
- no aps <inst> freeze
- no aps { <inst> | all }
- no area <area\_id> authentication
- no area <area\_id> nssa [ translate type7 { candidate | never | always } | no-summary ]
- no area <area\_id> range <ip\_address> <ip\_address\_mask> [ advertise | not-advertise | cost ]
- no area <area\_id> range <ipv6\_subnet> [ advertise | not-advertise | cost ]
- no area <area\_id> stub [ no-summary ]
- no area <area\_id> stub [ no-summary ]
- no area <area\_id> virtual-link <router\_id> [ hello-interval [ <hello\_interval> ] ] [ retransmit-interval [ <retransmit\_interval> ] ] [ dead-interval [ <dead\_interval> ] ]
- no area <area\_id> virtual-link <router\_id> authentication
- no area <area\_id> virtual-link <router\_id> authentication-key
- no area <area\_id> virtual-link <router\_id> message-digest-key <md\_key\_id>
- no authentication basic
- no banner [ motd | login | exec ]
- no base-time
- no block-due-to-oversize-enable
- no broadcast
- no cbs <cbs>
- no cfm domain { <md\_name> | all }
- no cir <cir>
- no client-identifier
- no client-name
- no clock summer-time
- no clock timezone
- no close-due-to-invalid-rx-enable
- no close-due-to-octets-exceeded-enable
- no color-mode
- no continuity-check
- no control-list index <index>
- no control-vlan
- no coupling-flag
- no cycle-time
- no ddmi
- o debug gdbserver
- no debug interrupt monitor [ source <intr\_name> ]
- no debug phy loopback [ near | far | connector | mac-serdes-input | mac-serdes-facility |

- mac-serdes-equipment | media-serdes-input | media-serdes-facility | media-serdes-equipment |  
qsgmii-tbi | qsgmii-gmi | qsgmii-serdes ]
- no debug ptp ms-pdv log-level
  - no debug trace hunt
  - no default-information originate
  - no default-information originate
  - no default-metric
  - no default-metric
  - no default-router
  - no description
  - no description
  - no distance
  - no distance
  - no distance
  - no dmac { <dmac\_addr> [ / <mask\_mac\_addr> ] | multicast | broadcast | any }
  - no dns-server
  - no domain-name
  - no dot1x authentication timer inactivity
  - no dot1x authentication timer re-authenticate
  - no dot1x feature { [ guest-vlan ] [ radius-qos ] [ radius-vlan ] }
  - no dot1x guest-vlan
  - no dot1x guest-vlan
  - no dot1x guest-vlan supplicant
  - no dot1x max-reauth-req
  - no dot1x port-control
  - no dot1x radius-qos
  - no dot1x radius-vlan
  - no dot1x re-authentication
  - no dot1x system-auth-control
  - no dot1x timeout quiet-period
  - no dot1x timeout tx-period
  - no drop-on-yellow
  - no duplex
  - no ebs <ebs>
  - no editing
  - no egress interface
  - no eir <eir>
  - no enable
  - no enable password [ level <priv> ]
  - no enable secret { [ 0 | 5 ] } [ level <priv> ]
  - no erps { <inst> | all }
  - no etype [ <0x600-0xffff> ]
  - no excessive-restart
  - no exec-banner
  - no exec-timeout
  - no fec
  - no flooding
  - no flow-meter

- no flowcontrol
- no frame-length-check
- no frer-vlan
- no gate
- no green-ethernet eee
- no green-ethernet eee optimize-for-power
- no green-ethernet eee urgent-queues [ <urgent\_queue\_range\_list> ]
- no green-ethernet energy-detect
- no green-ethernet fan [ temp-on [ <temp\_on\_num> ] ] [ temp-max [ <temp\_max\_num> ] ]
- no green-ethernet led interval <v\_0\_to\_24>
- no green-ethernet led on-event { [ link-change ] [ error ] }
- no green-ethernet short-reach
- no guard-time <guard\_time>
- no gvrp
- no gvrp max-vlans <maxvlans>
- no gvrp time { [ join-time <join\_time> ] [ leave-time <leave\_time> ] [ leave-all-time <leave\_all\_time> ] }
- no hardware-address
- no history size
- no hold-off-time <hold\_off>
- no hold-off-time <hold\_off>
- no host
- no hostname
- no informs
- no ingress stream-id-list
- no inner-tag
- no interconnection control-vlan
- no interconnection id
- no interconnection interface
- no interconnection mode
- no interconnection name
- no interconnection recovery-profile
- no interconnection role
- no interconnection sf-trigger
- no interface llag <llag\_id>
- no interface vlan <vlan\_list> area <area\_id>
- no interface vlan <vlist>
- no ip address
- no ip arp inspection
- no ip arp inspection check-vlan
- no ip arp inspection entry interface <port\_type> <in\_port\_type\_id> <vlan\_var> <mac\_var> <ipv4\_var>
- no ip arp inspection logging
- no ip arp inspection trust
- no ip arp inspection vlan <in\_vlan\_list>
- no ip arp inspection vlan <in\_vlan\_list> logging
- no ip dhcp excluded-address <low\_ip> [ <high\_ip> ]
- no ip dhcp pool <pool\_name>

- no ip dhcp relay
- no ip dhcp relay information option
- no ip dhcp relay information policy
- no ip dhcp server
- no ip dhcp server
- no ip dhcp snooping
- no ip dhcp snooping trust
- no ip dns proxy
- no ip domain name
- no ip helper-address
- no ip http secure-redirect
- no ip http secure-server
- no ip igmp host-proxy [ leave-proxy ]
- no ip igmp snooping
- no ip igmp snooping compatibility
- no ip igmp snooping filter
- no ip igmp snooping immediate-leave
- no ip igmp snooping last-member-query-interval
- no ip igmp snooping max-groups
- no ip igmp snooping mrouter
- no ip igmp snooping priority
- no ip igmp snooping querier { election | address }
- no ip igmp snooping query-interval
- no ip igmp snooping query-max-response-time
- no ip igmp snooping robustness-variable
- no ip igmp snooping unsolicited-report-interval
- no ip igmp snooping vlan [ <vlan\_list> ]
- no ip igmp ssm-range
- no ip igmp unknown-flooding
- no ip name-server [ <order> ]
- no ip ospf authentication
- no ip ospf authentication-key
- no ip ospf message-digest-key <md\_key\_id>
- no ip ospf { priority | cost | dead-interval | hello-interval | retransmit-interval }
- no ip rip authentication { mode | key-chain | string }
- no ip rip split-horizon [ poisoned-reverse ]
- no ip rip { send | receive } version
- no ip route <v\_ipv4\_addr> <v\_ipv4\_netmask> <v\_ipv4\_gw> [ distance <v\_distance> ]
- no ip route <v\_ipv4\_subnet> <v\_ipv4\_gw> [ distance <v\_distance> ]
- no ip routing
- no ip source binding interface <port\_type> <in\_port\_type\_id> <vlan\_var> <ipv4\_var>  
<mac\_var>
- no ip ssh
- no ip verify source
- no ip verify source
- no ip verify source limit
- no ipmc profile
- no ipmc profile <profile\_name>

- no ipmc range <entry\_name>
- no ipv
- no ipv4 [ proto { <pr4> | tcp | udp | any } ] [ sip { <sip4> | any } ] [ dip { <dip4> | any } ] [ dscp { <dscp4> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } | any } ] [ fragment { yes | no | any } ] [ dport { <dp4> | any } ]
- no ipv6 [ proto { <pr6> | tcp | udp | any } ] [ sip { <sip6> | any } ] [ dip { <dip6> | any } ] [ dscp { <dscp6> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } | any } ] [ dport { <dp6> | any } ]
- no ipv6 address [ <ipv6\_subnet> ]
- no ipv6 address { autoconfig | dhcp [ rapid-commit ] }
- no ipv6 dhcp relay [ { destination <ipv6\_ucast> interface vlan <v\_vlan\_id> } | { interface vlan <i\_vlan\_id> } ]
- no ipv6 dhcp snooping
- no ipv6 dhcp snooping trust
- no ipv6 mld host-proxy [ leave-proxy ]
- no ipv6 mld snooping
- no ipv6 mld snooping
- no ipv6 mld snooping compatibility
- no ipv6 mld snooping filter
- no ipv6 mld snooping immediate-leave
- no ipv6 mld snooping last-member-query-interval
- no ipv6 mld snooping max-groups
- no ipv6 mld snooping mrouter
- no ipv6 mld snooping priority
- no ipv6 mld snooping querier election
- no ipv6 mld snooping query-interval
- no ipv6 mld snooping query-max-response-time
- no ipv6 mld snooping robustness-variable
- no ipv6 mld snooping unsolicited-report-interval
- no ipv6 mld snooping vlan [ <vlan\_list> ]
- no ipv6 mld ssm-range
- no ipv6 mld unknown-flooding
- no ipv6 ospf { priority | cost | dead-interval | hello-interval | retransmit-interval | transmit-delay | passive }
- no ipv6 route <v\_ipv6\_subnet> <v\_ipv6\_ucast> [ interface vlan <v\_vlan\_id> ] [ distance <v\_distance> ]
- no ipv6 source binding interface <port\_type> <port\_type\_id> [ vlan <vlan\_id> ] <ipv6\_ucast> <mac\_ucast>
- no ipv6 verify source
- no ipv6 verify source
- no ipv6 verify source limit
- no json notification host <name>
- no json notification listen [ <notification> [ <host> ] ]
- no key <key\_id> key-string
- no key chain <key\_chain\_name>
- no lacp
- no lacp failover [ revertive | non-revertive ]

- no lacp max-bundle [ <uint> ]
- no lacp port-priority <v\_1\_to\_65535>
- no lacp system-priority <v\_1\_to\_65535>
- no lacp timeout { fast | slow }
- no lease
- no length
- no link-oam
- no link-oam link-monitor frame
- no link-oam link-monitor frame-seconds
- no link-oam link-monitor supported
- no link-oam link-monitor symbol-period
- no link-oam mib-retrieval supported
- no link-oam mode
- no link-oam remote-loopback supported
- no link-oam variable-retrieve
- no llc [ <0x0-0xff> [ <0x0-0xff> ] ]
- no lldp cdp-aware
- no lldp holdtime
- no lldp med datum
- no lldp med fast
- no lldp med location-tlv altitude
- no lldp med location-tlv civic-addr { country | state | county | city | district | block | street | leading-street-direction | trailing-street-suffix | street-suffix | house-no | house-no-suffix | landmark | additional-info | name | zip-code | building | apartment | floor | room-number | place-type | postal-community-name | p-o-box | additional-code }
- no lldp med location-tlv elin-addr
- no lldp med location-tlv latitude
- no lldp med location-tlv longitude
- no lldp med media-vlan policy-list [ <v\_range\_list> ]
- no lldp med media-vlan-policy <policies\_list>
- no lldp med transmit-tlv [ capabilities ] [ location ] [ network-policy ] [ poe ]
- no lldp med type
- no lldp receive
- no lldp reinit
- no lldp timer
- no lldp tlv-select { management-address | port-description | system-capabilities | system-description | system-name }
- no lldp transmission-delay
- no lldp transmit
- no lldp trap
- no location
- no logging host
- no logging notification listen [ <name> ]
- no logging on
- no loop-protect
- no loop-protect
- no loop-protect action
- no loop-protect shutdown-time

- no loop-protect transmit-time
- no loop-protect tx-mode
- no mac address-table aging-time
- no mac address-table aging-time <v\_0\_10\_to\_1000000>
- no mac address-table learning [ secure ]
- no mac address-table learning vlan <vlan\_list>
- no mac address-table static <v\_mac\_addr> vlan <v\_vlan\_id> [ interface <port\_type> [ <v\_port\_type\_list> ] ]
- no mark-red-enable
- no max-metric router-lsa [ on-startup ] [ on-shutdown ] [ administrative ]
- no max-sdu
- no media-redundancy { <inst> | all }
- no media-type
- no mep { <mepid> | all }
- no mode
- no monitor session <session\_number> [ destination { interface <port\_type> [ <di\_list> ] | remote } | source { interface <port\_type> [ <si\_list> ] [ both | rx | tx ] | remote | vlan <source\_vlan\_list> | cpu [ both | rx | tx ] } ]
- no motd-banner
- no mrm priority
- no mrm react-on-link-change
- no mrp periodic
- no mtu
- no mvr
- no mvr immediate-leave
- no mvr name <mvr\_name> channel
- no mvr name <mvr\_name> frame priority
- no mvr name <mvr\_name> frame tagged
- no mvr name <mvr\_name> last-member-query-interval
- no mvr name <mvr\_name> mode
- no mvr name <mvr\_name> type
- no mvr name <mvr\_name> { election | igmp-address }
- no mvr vlan <v\_vlan\_list>
- no mvr vlan <v\_vlan\_list> channel
- no mvr vlan <v\_vlan\_list> frame priority
- no mvr vlan <v\_vlan\_list> frame tagged
- no mvr vlan <v\_vlan\_list> last-member-query-interval
- no mvr vlan <v\_vlan\_list> mode
- no mvr vlan <v\_vlan\_list> type
- no mvr vlan <v\_vlan\_list> { election | igmp-address }
- no mvrp
- no name
- no neighbor <ip\_addr>
- no netbios-name-server
- no netbios-node-type
- no netbios-scope
- no network
- no network <ip\_address>

- no network <ip\_address> <wildcard\_mask>
- no network <ip\_address> <wildcard\_mask> [ area <area\_id> ]
- no network-clock clk-source <clk\_list> nominate
- no network-clock clk-source <clk\_src> aneg-mode
- no network-clock clk-source <clk\_src> hold-timeout
- no network-clock clk-source <clk\_src> priority
- no network-clock clk-source <clk\_src> ssm-overwrite
- no network-clock input-source
- no network-clock option
- no network-clock output-source
- no network-clock selector
- no network-clock ssm-freerun
- no network-clock ssm-holdover
- no network-clock synchronization ssm
- no network-clock wait-to-restore
- no nis-domain-name
- no nis-server
- no node-id
- no ntp
- no ntp server <index\_var>
- no ntp-server
- no offset-list <access\_list\_name> { in | out } <metric\_value> [ vlan <vlan\_id> ]
- no oui
- no outer-tag
- no passive-interface { default | vlan <vlan\_list> }
- no passive-interface { default | vlan <vlan\_list> }
- no platform phy mode
- no port-security
- no port-security aging
- no port-security aging time
- no port-security hold time
- no port-security mac-address { [ sticky ] [ <mac> [ vlan <vlan\_id> ] ] }
- no port-security maximum
- no port-security maximum-violation
- no port-security violation
- no port0 smac
- no port1 interface
- no port1 sf-trigger
- no port1 smac
- no port2 interface
- no port2 sf-trigger
- no priority
- no priority-flowcontrol prio [ <prio> ]
- no privilege <mode\_name> level <0-15> <cmd>
- no privilege level
- no prompt
- no protect-mep
- no protected-vlans

- no ptp <0-3> announce { interval | timeout }
- no ptp <clockinst>
- no ptp <clockinst> afi-announce
- no ptp <clockinst> afi-sync
- no ptp <clockinst> allow-faults
- no ptp <clockinst> allow-lost-resp
- no ptp <clockinst> clk
- no ptp <clockinst> delay-asymmetry
- no ptp <clockinst> delay-mechanism
- no ptp <clockinst> delay-req interval
- no ptp <clockinst> delay-thresh
- no ptp <clockinst> domain
- no ptp <clockinst> egress-latency
- no ptp <clockinst> gptp-interval
- no ptp <clockinst> gptp-to
- no ptp <clockinst> ho
- no ptp <clockinst> ingress-latency
- no ptp <clockinst> localpriority
- no ptp <clockinst> localpriority
- no ptp <clockinst> log
- no ptp <clockinst> mode { boundary | e2transparent | p2ptransparent | master | slave | bcfrentend }
- no ptp <clockinst> not-slave
- no ptp <clockinst> path-trace-enable
- no ptp <clockinst> priority1
- no ptp <clockinst> priority2
- no ptp <clockinst> servo ad
- no ptp <clockinst> servo ai
- no ptp <clockinst> servo ap
- no ptp <clockinst> servo displaystates
- no ptp <clockinst> servo gain
- no ptp <clockinst> sync-interval
- no ptp <clockinst> sync-rx-to
- no ptp <clockinst> two-step
- no ptp <clockinst> uni <idx>
- no ptp <clockinst> virtual-port accuracy
- no ptp <clockinst> virtual-port class
- no ptp <clockinst> virtual-port io-pin
- no ptp <clockinst> virtual-port local-priority
- no ptp <clockinst> virtual-port priority1
- no ptp <clockinst> virtual-port priority2
- no ptp <clockinst> virtual-port variance
- no ptp <clockinst> wireless mode interface <port\_type> [ <v\_port\_type\_list> ]
- no ptp <v\_0\_to\_3> compute-meanlinkdelay [ force ]
- no ptp <v\_0\_to\_3> compute-neighbor-rate-ratio [ force ]
- no ptp cmdls allow-faults <v\_1\_to\_255>
- no ptp cmdls allow-lost-resp <v\_0\_to\_10>
- no ptp cmdls compute-meanlinkdelay [ force ]

- no ptp cmdls compute-neighbor-rate-ratio [ force ]
- no ptp cmdls delay-asymmetry
- no ptp cmdls pdelay-thresh
- no ptp cmdls pdelayreq-interval
- no ptp cmdls pdelayreq-interval { <v\_minus\_7\_to\_5> | stop | default } [ force ]
- no ptp ext
- no ptp ho-spec
- no ptp io-pin <io\_pin>
- no ptp pps-delay
- no ptp pps-sync
- no ptp ref-clock
- no ptp rs422
- no ptp system-time
- no pvlan <pvlan\_list>
- no pvlan isolation
- no qos class
- no qos cos
- no qos cut-through queue <queue>
- no qos dei
- no qos dpl
- no qos dscp-classify
- no qos dscp-remark
- no qos dscp-translate
- no qos egress-map
- no qos ingress-map
- no qos map cos-dscp <cos> dpl <dpl>
- no qos map cos-tag cos <cos> dpl <dpl>
- no qos map dscp-classify { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } }
- no qos map dscp-cos { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } }
- no qos map dscp-egress-translation { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } } <dpl>
- no qos map dscp-ingress-translation { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } }
- no qos map egress <id>
- no qos map ingress <id>
- no qos map tag-cos pcp <pcp> dei <dei>
- no qos pcp
- no qos policer
- no qos qce <qce\_id\_range>
- no qos qce { [ addr ] [ key ] }
- no qos queue-policer queue <queue>
- no qos queue-shaper queue <queue>
- no qos shaper
- no qos storm { unicast | broadcast | unknown }
- no qos storm { unicast | multicast | broadcast }
- no qos tag-remark

- no qos trust dscp
- no qos trust tag
- no qos wred group <group> queue <queue> dpl <dpl>
- no qos wred-group
- no qos wrr
- no radius-server attribute 32
- no radius-server attribute 4
- no radius-server attribute 95
- no radius-server deadtime
- no radius-server host <host\_name> [ auth-port <auth\_port> ] [ acct-port <acct\_port> ]
- no radius-server key
- no radius-server retransmit
- no radius-server timeout
- no range <entry\_name>
- no recovery algorithm
- no recovery individual
- no recovery latent-error-detection [ difference <diff> ] [ period <period> ] [ paths <paths> ] [ reset-period <reset\_period> ]
- no recovery reset-timeout
- no recovery take-no-sequence
- no recovery terminate
- no recovery-profile
- no redistribute { static | connected | ospf }
- no redistribute { static | connected | rip }
- no redistribute { static | connected }
- no remote mep { <rmepid> | all }
- no reserved-only
- no revertive
- no rmon alarm <id>
- no rmon collection history <id>
- no rmon collection stats <id>
- no rmon event <id>
- no role
- no router access-list <access\_list\_name>
- no router access-list <access\_list\_name> { permit | deny } { any | <ipv4\_addr> <ipv4\_netmask> }
- no router ospf
- no router ospf6
- no router rip
- no router-id
- no rpl
- no service { <ma\_name> | all }
- no sflow [ <\_sampler\_idx\_list> ]
- no sflow agent-ip
- no sflow collector-address [ receiver <rcvr\_idx\_list> ]
- no sflow collector-port [ receiver <rcvr\_idx\_list> ]
- no sflow counter-poll-interval [ <\_sampler\_idx\_list> ]
- no sflow max-datagram-size [ receiver <rcvr\_idx\_list> ]

- no sflow max-sampling-size [ sampler < sampler\_idx\_list > ]
- no sflow timeout [ receiver < rcvr\_idx\_list > ]
- no shutdown
- no smac
- no smac { < mac\_addr > [ / < mask\_mac\_addr > ] | any }
- no snap [ { < 0x0-0xffff > | rfc-1042 | snap-8021h } [ < 0x0-0xffff > ] ]
- no snmp-server
- no snmp-server access < group\_name > model { v1 | v2c | v3 | any } level { auth | noauth | priv }
- no snmp-server community < v3\_comm > [ { ip-range < v\_ipv4\_addr > < v\_ipv4\_netmask > | ipv6-range < v\_ipv6\_subnet > } ]
- no snmp-server contact
- no snmp-server engine-id local
- no snmp-server host < conf\_name >
- no snmp-server location
- no snmp-server security-to-group model { v1 | v2c | v3 } name < security\_name >
- no snmp-server trap < source\_name > { [ id < filter\_id > ] [ < oid\_subtree > { include | exclude } ] }
- no snmp-server user < username > engine-id < engineID >
- no snmp-server view < view\_name > < oid\_subtree >
- no spanning-tree
- no spanning-tree auto-edge
- no spanning-tree auto-edge
- no spanning-tree bpdu-guard
- no spanning-tree bpdu-guard
- no spanning-tree edge
- no spanning-tree edge
- no spanning-tree edge bpdu-filter
- no spanning-tree edge bpdu-guard
- no spanning-tree link-type
- no spanning-tree link-type
- no spanning-tree mode
- no spanning-tree mst < instance > cost
- no spanning-tree mst < instance > cost
- no spanning-tree mst < instance > port-priority
- no spanning-tree mst < instance > port-priority
- no spanning-tree mst < instance > priority
- no spanning-tree mst < instance > vlan
- no spanning-tree mst forward-time
- no spanning-tree mst hello-time
- no spanning-tree mst max-age
- no spanning-tree mst max-hops
- no spanning-tree mst name
- no spanning-tree recovery interval
- no spanning-tree restricted-role
- no spanning-tree restricted-role
- no spanning-tree restricted-tcn
- no spanning-tree restricted-tcn

- no spanning-tree transmit hold-count
- no speed
- no stream <id>
- no stream-id
- no stream-id <id>
- no svl fid { <fid\_list> | all }
- no switchport access vlan
- no switchport forbidden vlan
- no switchport hybrid acceptable-frame-type
- no switchport hybrid allowed vlan
- no switchport hybrid egress-tag
- no switchport hybrid ingress-filtering
- no switchport hybrid native vlan
- no switchport hybrid port-type
- no switchport mode
- no switchport trunk allowed vlan
- no switchport trunk native vlan
- no switchport trunk vlan tag native
- no switchport vlan ip-subnet <ipv4>
- no switchport vlan mac <mac\_addr> [ vlan <vlan\_id> ]
- no switchport vlan mapping
- no switchport vlan mapping <gid> <vlan\_list>
- no switchport vlan mapping <gid> { both | ingress | egress } <vid>
- no switchport vlan protocol group <grp\_id> [ vlan <vlan\_id> ]
- no switchport voice vlan discovery-protocol
- no switchport voice vlan mode
- no switchport voice vlan security
- no tacacs-server deadtime
- no tacacs-server host <host\_name> [ port <port> ]
- no tacacs-server key
- no tacacs-server timeout
- no temperature monitor
- no terminal editing
- no terminal exec-timeout
- no terminal history size
- no terminal length
- no terminal width
- no thermal-protect grp
- no thermal-protect grp <grp\_list>
- no timers basic
- no tsn flow meter <inst>
- no tsn frame-preemption
- no tsn frame-preemption ignore-lldp
- no tsn frame-preemption queue <queue>
- no tsn frame-preemption verify-disable
- no tsn frer { <inst> | all }
- no tsn ptp-check procedure
- no tsn ptp-check ptp-port

- no tsn ptp-check timeout
- no tsn stream filter <inst>
- no tsn stream gate <inst>
- no tsn tas always-guard-band
- no tsn tas base-time
- no tsn tas control-list index <index>
- no tsn tas control-list-length
- no tsn tas cycle-time
- no tsn tas cycle-time-extension
- no tsn tas gate-enabled
- no tsn tas gate-states
- no tsn tas max-sdu queue <queue>
- no udld port
- no udld { aggressive | enable }
- no upnp
- no upnp advertising-duration
- no upnp ip-addressing-mode
- no upnp static interface vlan
- no url
- no username <username>
- no uuid
- no vcl { dmacdip | smacsip }
- no vendor class-identifier <class\_id>
- no version
- no vlan protocol { { eth2 { <etype> | arp | ip | ipx | at } } | { snap { <oui> | rfc-1042 | snap-8021h } <pid> } | { llc <dsap> <ssap> } } [ group <word16> ]
- no vlan { { ethertype s-custom-port } | <vlan\_list> }
- no voice vlan
- no voice vlan aging-time
- no voice vlan class
- no voice vlan oui <oui>
- no voice vlan vid
- no wait-to-restore <wtr>
- no wait-to-restore <wtr>
- no web privilege group [ <group\_name> ] level
- no width
- no working-mep
- node-id <node\_id>
- ntp
- ntp server <index\_var> ip-address { <ipv4\_var> | <ipv6\_var> | <name\_var> }
- ntp server <index\_var> ip-address { <ipv4\_var> | <name\_var> }
- ntp-server <ip> [ <ip1> [ <ip2> [ <ip3> ] ] ]
- offset-list <access\_list\_name> { in | out } <metric\_value> [ vlan <vlan\_id> ]
- organization-specific-tlv { disable | defer }
- organization-specific-tlv { disable | defer }
- oui { default | siemens | custom <custom\_oui> }
- outer-tag [ vid { <vid> [ / <vid\_mask> ] | any } ] [ pcp <pcp\_val> [ / <pcp\_mask> ] ] [ dei <dei\_val> ] [ { c-tag | s-tag } ]

- passive-interface { default | vlan <vlan\_list> }
- password encrypted <encry\_password>
- password none
- password unencrypted <password>
- pcp <pcp>
- ping [ ip ] { <domain\_name> | <ip\_addr> } [ ttl <ttl\_value> ] [ repeat <count> ] [ { saddr <src\_addr> | sif { <port\_type> <src\_if> | vlan <vlan\_id> } } ] [ size <size> ] [ data <data\_value> ] [ { verbose | quiet } ]
- ping ipv6 { <domain\_name> | <ip\_addr> } [ repeat <count> ] [ saddr <src\_addr> ] [ sif { <port\_type> <src\_if> | vlan <vlan\_id> } ] [ size <size> ] [ data <data\_value> ] [ { verbose | quiet } ]
- platform debug { allow | deny }
- platform phy mode { wan | lg }
- port-security
- port-security aging
- port-security aging time <aging\_time>
- port-security hold time <hold\_time>
- port-security mac-address { [ sticky ] [ <mac> [ vlan <vlan\_id> ] ] }
- port-security maximum <limit>
- port-security maximum-violation <violate\_limit>
- port-security violation { protect | restrict | shutdown }
- port-status-tlv { disable | enable | defer }
- port-status-tlv { disable | enable | defer }
- port0 interface <port\_type> <port>
- port0 sf-trigger { link | { mep domain <md\_name> service <ma\_name> mep-id <mepid> } }
- port0 smac <mac>
- port1 interface <port\_type> <port>
- port1 interface <port\_type> <port>
- port1 sf-trigger { link | { mep domain <md\_name> service <ma\_name> mep-id <mepid> } }
- port1 sf-trigger { link | { mep domain <md\_name> service <ma\_name> mep-id <mepid> } }
- port1 smac <mac>
- port2 interface <port\_type> <port>
- port2 sf-trigger { link | { mep domain <md\_name> service <ma\_name> mep-id <mepid> } }
- preseedes <uint>
- preset classes <classes> [ color-aware ]
- preset classes <classes> [ color-aware ]
- priority { <prio> | any }
- priority-flowcontrol prio <prio>
- privilege <mode\_name> level <privilege> <cmd>
- privilege level <privileged\_level>
- prompt <prompt>
- protect interface <port\_type> <port>
- protect sf-trigger { link | { mep domain <md\_name> service <ma\_name> mep-id <mepid> } }

- protect-mep domain <md\_name> service <ma\_name> mep-id <mepid>
- protected-vlans <vlan\_list>
- ptp <clockinst> [ internal ]
- ptp <clockinst> afi-announce
- ptp <clockinst> afi-sync
- ptp <clockinst> allow-faults <allow\_faults>
- ptp <clockinst> allow-lost-resp <allow\_lost\_resp>
- ptp <clockinst> announce { [ interval { <interval> | stop | default } ] [ timeout <timeout> ] }
- ptp <clockinst> clk sync <threshold> ap <ap>
- ptp <clockinst> delay-asymmetry <delay\_asymmetry>
- ptp <clockinst> delay-mechanism { e2e | p2p | common-p2p }
- ptp <clockinst> delay-req interval { <interval> | stop | default }
- ptp <clockinst> delay-thresh <delay\_thresh>
- ptp <clockinst> domain <domain>
- ptp <clockinst> egress-latency <egress\_latency>
- ptp <clockinst> filter [ delay <delay> ] [ period <period> ] [ dist <dist> ]
- ptp <clockinst> filter-type { aci-default | aci-freq-xo | aci-phase-xo | aci-freq-tcxo | aci-phase-tcxo | aci-freq-ocxo-s3e | aci-phase-ocxo-s3e | aci-bc-partial-on-path-freq | aci-bc-partial-on-path-phase | aci-bc-full-on-path-freq | aci-bc-full-on-path-phase | aci-freq-accuracy-fdd | aci-freq-accuracy-xdsl | aci-elec-freq | aci-elec-phase | aci-phase-relaxed-c60w | aci-phase-relaxed-c150 | aci-phase-relaxed-c180 | aci-phase-relaxed-c240 | aci-phase-ocxo-s3e-r4-6-1 | aci-basic-phase | aci-basic-phase-low | basic }
- ptp <clockinst> gptp-interval { <interval> | stop | default }
- ptp <clockinst> gptp-to <gptp\_to>
- ptp <clockinst> ho [ filter <ho\_filter> ] [ adj-threshold <adj\_threshold> ]
- ptp <clockinst> ingress-latency <ingress\_latency>
- ptp <clockinst> local-clock { update | ratio <ratio> }
- ptp <clockinst> localpriority <localpriority>
- ptp <clockinst> localpriority <localpriority>
- ptp <clockinst> log <debug\_mode> [ log-to-file ] [ control ] [ max-time <max\_time> ]
- ptp <clockinst> log delete
- ptp <clockinst> mcast-dest { default | link-local }
- ptp <clockinst> mgtSettableLogAnnounceInterval { <interval> | stop | default }
- ptp <clockinst> mgtSettableLogGptpCapableMessageInterval { <interval> | stop | default }
- ptp <clockinst> mgtSettableLogPdelayReqInterval { <interval> | stop | default }
- ptp <clockinst> mgtSettableLogSyncInterval { <interval> | stop | default }
- ptp <clockinst> mode { boundary | e2etransparent | p2ptransparent | master | slave | bcffrontend } [ onestep | twostep ] [ ethernet | ethernet-mixed | ip4multi | ip4mixed | ip4unicast | oam | onepps | ip6mixed | ethip4ip6-combo ] [ oneway | twoway ] [ id <v\_clock\_id> ] [ vid <vid> [ <prio> ] ] [ mep <mep\_id> ] [ profile { ieee1588 | g8265.1 | g8275.1 | 802.1as } ] [ clock-domain <clock\_domain> ] [ dscp <dscp\_id> ]
- ptp <clockinst> not-slave
- ptp <clockinst> path-trace-enable
- ptp <clockinst> priority1 <priority1>
- ptp <clockinst> priority2 <priority2>
- ptp <clockinst> servo ad <ad>
- ptp <clockinst> servo ai <ai>

- ptp <clockinst> servo ap <ap>
- ptp <clockinst> servo displaystates
- ptp <clockinst> servo gain <gain>
- ptp <clockinst> slave-cfg [ stable-offset <stable\_offset> ] [ offset-ok <offset\_ok> ] [ offset-fail <offset\_fail> ]
- ptp <clockinst> statistics [ clear ]
- ptp <clockinst> sync-interval { <interval> | stop | default }
- ptp <clockinst> sync-rx-to <sync\_rx\_to>
- ptp <clockinst> time-property [ utc-offset <utc\_offset> ] [ valid ] [ leap-59 | leap-61 ] [ time-traceable ] [ freq-traceable ] [ ptp-timescale ] [ time-source <time\_source> ] [ leap-pending <date\_string> { leap-59 | leap-61 } ]
- ptp <clockinst> two-step [ true ]
- ptp <clockinst> two-step false
- ptp <clockinst> uni <idx> [ duration <duration> ] <ip>
- ptp <clockinst> useMgtSettableLogGptpCapableMessageInterval <usemgtSettableLogGptpCapableMessageInterval>
- ptp <clockinst> usemgtSettableLogAnnounceInterval <usemgtSettableLogAnnounceInterval>
- ptp <clockinst> usemgtSettableLogPdelayReqInterval <usemgtSettableLogPdelayReqInterval>
- ptp <clockinst> usemgtSettableLogSyncInterval <usemgtSettableLogSyncInterval>
- ptp <clockinst> virtual-port accuracy <ptp\_accuracy>
- ptp <clockinst> virtual-port class <ptp\_class>
- ptp <clockinst> virtual-port io-pin <ptp\_io\_pin>
- ptp <clockinst> virtual-port local-priority <local\_priority>
- ptp <clockinst> virtual-port priority1 <priority1>
- ptp <clockinst> virtual-port priority2 <priority2>
- ptp <clockinst> virtual-port variance <ptp\_variance>
- ptp <clockinst> wireless delay <base\_delay> [ <incr\_delay> ] interface <port\_type> [ <v\_port\_type\_list> ]
- ptp <clockinst> wireless mode interface <port\_type> [ <v\_port\_type\_list> ]
- ptp <clockinst> wireless pre-notification interface <port\_type> [ <v\_port\_type\_list> ]
- ptp <v\_0\_to\_3> compute-meanlinkdelay [ force ]
- ptp <v\_0\_to\_3> compute-neighbor-rate-ratio [ force ]
- ptp cal 1pps <cable\_latency>
- ptp cal p2p <port\_type> <ref\_port> <port\_type> <other\_port> <cable\_latency>
- ptp cal port <port\_type> <v\_port\_type\_id> [ mode { 10m-cu | 100m-cu | 1g-cu | 1g | 2g5 | 5g | 10g | all } ] reset
- ptp cal port <port\_type> <v\_port\_type\_id> offset <pps\_offset> cable-latency <cable\_latency>
- ptp cal port <port\_type> <v\_port\_type\_id> start [ synce ]
- ptp cal t-plane <port\_type> <v\_port\_type\_id> { ext | int }
- ptp cmls allow-faults <v\_1\_to\_255>
- ptp cmls allow-lost-resp <v\_0\_to\_10>
- ptp cmls compute-meanlinkdelay [ force ]
- ptp cmls compute-neighbor-rate-ratio [ force ]
- ptp cmls delay-asymmetry <v\_minus\_100000\_to\_100000>
- ptp cmls pdelay-thresh <v\_0\_to\_4000000000>
- ptp cmls pdelayreq-interval { <v\_minus\_7\_to\_5> | stop | default } [ force ]
- ptp cmls statistics [ clear ]

- ptp ext [ output | input | out-in ] [ ext <clockfreq> ] [ ltc | single | independent | common | auto ]
- ptp ho-spec [ cat1 <cat1> ] [ cat2 <cat2> ] [ cat3 <cat3> ]
- ptp io-pin <io\_pin> [ pps-output | waveform-output | load | save ] [ domain <domain> ] [ freq <freq> ] [ { interface <port\_type> <v\_port\_type\_id> } ]
- ptp pps-delay { { auto master-port interface <port\_type> <v\_port\_type\_id> } | { man cable-delay <cable\_delay> } }
- ptp pps-sync { main-auto | main-man | sub } [ pps-phase <pps\_phase> ] [ cable-asy <cable\_asy> ] [ ser-man | ser-auto ]
- ptp ref-clock { mhz125 | mhz156p25 | mhz250 }
- ptp rs422 baudrate <baudrate> [ parity { none | even | odd } ] [ wordlength <wordlength> ] [ stopbits <stopbits> ] [ flowctrl { none | rtscts } ]
- ptp rs422 { main-auto | main-man | sub | calib } [ pps-delay <pps\_delay> ] { ser [ proto { polyt | zda | rmc } ] | { pim interface <port\_type> <v\_port\_type\_id> } }
- ptp system-time { get | set }
- ptp tc-internal [ mode <mode> ]
- pvlan <pvlan\_list>
- pvlan isolation
- qos class <cosid>
- qos cos <cos>
- qos cut-through queue <queue>
- qos dei <dei>
- qos dpl <dpl>
- qos dscp-classify { zero | selected | any }
- qos dscp-remark { rewrite | remap | remap-dp }
- qos dscp-translate
- qos egress-map <id>
- qos ingress-map <id>
- qos map cos-dscp <cos> dpl <dpl> dscp { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } }
- qos map cos-tag cos <cos> dpl <dpl> pcp <pcp> dei <dei>
- qos map dscp-classify { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } }
- qos map dscp-cos { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } } cos <cos> dpl <dpl>
- qos map dscp-egress-translation { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } } <dpl> to { <dscp\_num\_tr> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } }
- qos map dscp-ingress-translation { <dscp\_num> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } } to { <dscp\_num\_tr> | { be | af11 | af12 | af13 | af21 | af22 | af23 | af31 | af32 | af33 | af41 | af42 | af43 | cs1 | cs2 | cs3 | cs4 | cs5 | cs6 | cs7 | ef | va } }
- qos map egress <map\_id>
- qos map ingress <map\_id>
- qos map tag-cos pcp <pcp> dei <dei> cos <cos> dpl <dpl>
- qos pcp <pcp>
- qos policer <rate> [ kbps | mbps | fps | kfps ] [ flowcontrol ]



- recovery individual
- recovery latent-error-detection [ difference <diff> ] [ period <period> ] [ paths <paths> ] [ reset-period <reset\_period> ]
- recovery reset-timeout <reset\_timeout>
- recovery take-no-sequence
- recovery terminate
- recovery-profile { 10ms | 30ms | 200ms | 500ms }
- redistribute { static | connected | ospf } [ metric <metric\_val> ]
- redistribute { static | connected | rip } [ metric <metric\_val> ] [ metric-type { 1 | 2 } ]
- redistribute { static | connected }
- reload { cold | warm | defaults [ keep-ip ] [ force ] }
- remote mep <rmepid>
- reserved-only
- revertive
- ring-id <ring\_id>
- ring-type { major | sub-ring [ virtual-channel ] | interconnected-sub-ring { connected-ring <connected\_ring\_inst> [ virtual-channel ] [ propagate-topology-change ] } }
- rmon alarm <id> { ifInOctets | ifInUcastPkts | ifInNUcastPkts | ifInDiscards | ifInErrors | ifInUnknownProtos | ifOutOctets | ifOutUcastPkts | ifOutNUcastPkts | ifOutDiscards | ifOutErrors | ifOutQLen } <ifIndex> <interval> { absolute | delta } rising-threshold <rising\_threshold> <rising\_event\_id> falling-threshold <falling\_threshold> <falling\_event\_id> { [ rising | falling | both ] }
- rmon collection history <id> [ buckets <buckets> ] [ interval <interval> ]
- rmon collection stats <id>
- rmon event <id> [ log ] [ trap [ <word127> ] ] { [ description <description> ] }
- role { mrc | mrm | mra }
- router access-list <access\_list\_name> { permit | deny } { any | <ipv4\_addr> <ipv4\_netmask> }
- router ospf
- router ospf6
- router rip
- router-id <router\_id>
- router-id <router\_id>
- rpl { owner | neighbor } { port0 | port1 }
- send { \* | <session\_list> | console 0 | vty <vty\_list> } <message>
- sender-id-tlv { disable | chassis | management | chassis-management | defer }
- service <ma\_name>
- sflow [ <sampler\_idx\_list> ]
- sflow agent-ip { ipv4 <v\_ipv4\_addr> | ipv6 <v\_ipv6\_addr> }
- sflow collector-address [ receiver <rcvr\_idx\_list> ] [ <ipv4\_var> | <ipv6\_var> | <domain\_name> ]
- sflow collector-port [ receiver <rcvr\_idx\_list> ] <collector\_port>
- sflow counter-poll-interval [ sampler <sampler\_idx\_list> ] [ <poll\_interval> ]
- sflow max-datagram-size [ receiver <rcvr\_idx\_list> ] <datagram\_size>
- sflow max-sampling-size [ sampler <sampler\_idx\_list> ] [ <max\_sampling\_size> ]
- sflow sampling-rate [ sampler <sampler\_idx\_list> ] [ <sampling\_rate> ]
- sflow timeout [ receiver <rcvr\_idx\_list> ] <timeout>
- show aaa

- show access management [ statistics | <access\_id\_list> ]
- show access-list [ interface [ <port\_type> [ <v\_port\_type\_list> ] ] ] [ rate-limiter [ <rate\_limiter\_list> ] ] [ ace statistics [ <ace\_list> ] ]
- show access-list ace-status [ static ] [ link-oam ] [ loop-protect ] [ dhcp ] [ dhcp6-snooping ] [ ptp ] [ upnp ] [ arp-inspection ] [ cfm ] [ aps ] [ erps ] [ mrp ] [ ipmc ] [ ip-source-guard ] [ ipv6-source-guard ] [ ip-mgmt ] [ ztp ] [ ip ] [ conflicts ] [ switch <switch\_list> ]
- show aggregation [ mode ]
- show alarm sources [ <filter> ]
- show alarm status [ <alarm\_name> ]
- show aps [ <inst\_list> ] { [ statistics ] | [ details ] }
- show cfm domains [ domain <md\_name> ] [ details ]
- show cfm errors
- show cfm meps [ domain <md\_name> ] [ service <ma\_name> ] [ mep-id <mepid> ] [ details ]
- show cfm services [ domain <md\_name> ] [ service <ma\_name> ] [ details ]
- show clock
- show clock detail
- show ddmi
- show dot1x statistics { eapol | radius | all } [ interface <port\_type> [ <v\_port\_type\_list> ] ]
- show dot1x status [ interface <port\_type> [ <v\_port\_type\_list> ] ] [ brief ]
- show env power
- show erps [ <inst\_list> ] [ statistics ] [ details ]
- show green-ethernet [ interface <port\_type> [ <port\_list> ] ]
- show green-ethernet eee [ interface <port\_type> [ <port\_list> ] ]
- show green-ethernet energy-detect [ interface <port\_type> [ <port\_list> ] ]
- show green-ethernet fan
- show green-ethernet short-reach [ interface <port\_type> [ <port\_list> ] ]
- show history
- show interface <port\_type> [ <in\_port\_list> ] switchport [ access | trunk | hybrid ]
- show interface <port\_type> [ <port\_list> ] capabilities
- show interface <port\_type> [ <port\_list> ] description
- show interface <port\_type> [ <port\_list> ] statistics [ { packets | bytes | errors | discards | filtered | dot3br | { priority [ <priority\_list> ] } | link-state-changes } ] [ { up | down } ]
- show interface <port\_type> [ <port\_list> ] status [ err-disable ] [ details [ clause-73 ] ]
- show interface <port\_type> [ <port\_list> ] transceiver
- show interface <port\_type> [ <port\_list> ] veriphy
- show interface vlan [ <vlist> ]
- show ip acd
- show ip arp
- show ip arp inspection [ interface <port\_type> [ <in\_port\_type\_list> ] | vlan <in\_vlan\_list> ]
- show ip arp inspection entry [ dhcp-snooping | static ] [ interface <port\_type> [ <in\_port\_type\_list> ] ]
- show ip dhcp detailed statistics { server | client | snooping | relay | normal-forward | combined } [ interface <port\_type> [ <in\_port\_list> ] ]
- show ip dhcp excluded-address
- show ip dhcp pool [ <pool\_name> ]

- show ip dhcp relay [ statistics ]
- show ip dhcp server
- show ip dhcp server binding <ip>
- show ip dhcp server binding [ state { allocated | committed | expired } ] [ type { automatic | manual | expired } ]
- show ip dhcp server declined-ip
- show ip dhcp server declined-ip <declined\_ip>
- show ip dhcp server statistics
- show ip dhcp snooping [ interface <port\_type> [ <in\_port\_list> ] ]
- show ip dhcp snooping [ statistics ] [ interface <port\_type> [ <in\_port\_list> ] ]
- show ip dhcp snooping table
- show ip domain
- show ip http
- show ip igmp snooping [ vlan <v\_vlan\_list> ] [ group-database [ interface <port\_type> [ <v\_port\_type\_list> ] ] [ sfm-information ] ] [ detail ]
- show ip igmp snooping mrouter [ detail ]
- show ip interface [ brief ]
- show ip name-server
- show ip neighbor
- show ip ospf
- show ip ospf database [ { router | network | summary | asbr-summary | external | nssa-external } [ link-state-id <link\_state\_id> ] ] [ adv-router <adv\_router\_id> | self-originate ]
- show ip ospf interface [ vlan <vlan\_list> | vlink <vlink\_list> ]
- show ip ospf neighbor [ detail ]
- show ip ospf route
- show ip rip [ database ]
- show ip route
- show ip source binding [ dhcp-snooping | static ] [ interface <port\_type> [ <in\_port\_type\_list> ] ]
- show ip ssh
- show ip statistics [ system ]
- show ip verify source [ interface <port\_type> [ <in\_port\_type\_list> ] ]
- show ipmc profile [ <profile\_name> ] [ detail ]
- show ipmc range [ <entry\_name> ]
- show ipv6 dhcp relay [ interface vlan <v\_vlan\_id> ]
- show ipv6 dhcp relay statistics [ interface vlan <vlan\_id> ]
- show ipv6 dhcp snooping [ interface <port\_type> [ <in\_port\_list> ] ]
- show ipv6 dhcp snooping statistics [ interface <port\_type> [ <in\_port\_list> ] ] [ zero-suppress ]
- show ipv6 dhcp snooping table [ all ]
- show ipv6 dhcp-client [ interface vlan <v\_vlan\_list> ]
- show ipv6 interface [ brief ]
- show ipv6 mld snooping [ vlan <v\_vlan\_list> ] [ group-database [ interface <port\_type> [ <v\_port\_type\_list> ] ] [ sfm-information ] ] [ detail ]
- show ipv6 mld snooping mrouter [ detail ]
- show ipv6 neighbor
- show ipv6 ospf
- show ipv6 ospf database [ { router | network | inter-prefix | inter-router | external | link | intra-

- ```

prefix } [ link-state-id <link_state_id> ] ] [ adv-router <adv_router_id> | self-originate ]

```
- show ipv6 ospf interface [ vlan <vlan\_list> ]
  - show ipv6 ospf neighbor [ detail ]
  - show ipv6 ospf route
  - show ipv6 route
  - show ipv6 source binding [ dhcpv6-snooping | static ] [ interface <port\_type> [ <port\_list> ] ]
  - show ipv6 statistics [ system ] [ interface vlan <vlan\_list> ]
  - show ipv6 verify source [ interface <port\_type> [ <port\_list> ] ]
  - show lacp { internal | statistics | system-id | neighbor } [ details ]
  - show licenses [ details ]
  - show line [ alive ]
  - show link-oam { [ status ] [ link-monitor ] [ statistics ] } [ interface <port\_type> [ <plist> ] ]
  - show lldp eee [ interface <port\_type> [ <v\_port\_type\_list> ] ]
  - show lldp med media-vlan-policy [ <v\_0\_to\_31> ]
  - show lldp med remote-device [ interface <port\_type> [ <port\_list> ] ]
  - show lldp neighbors [ interface <port\_type> [ <v\_port\_type\_list> ] ]
  - show lldp preempt [ interface <port\_type> [ <v\_port\_type\_list> ] ]
  - show lldp statistics [ interface <port\_type> [ <v\_port\_type\_list> ] ]
  - show logging <log\_id>
  - show logging [ informational ] [ notice ] [ warning ] [ error ]
  - show loop-protect [ interface <port\_type> [ <plist> ] ]
  - show mac address-table [ conf | static | aging-time | { { learning | count } [ interface <port\_type> [ <v\_port\_type\_list> ] | vlan <v\_vlan\_id\_2> ] } | { address <v\_mac\_addr> [ vlan <v\_vlan\_id> ] } | vlan <v\_vlan\_id\_1> | interface <port\_type> [ <v\_port\_type\_list\_1> ] ]
  - show media-redundancy [ <inst\_list> ] { status | statistics } [ details ]
  - show monitor [ session { <session\_number> | all | remote } ]
  - show mvrp status [ interface <port\_type> [ <plist> ] ] [ all | mvrp ]
  - show mvr [ vlan <v\_vlan\_list> | name <mvr\_name> ] [ group-database [ interface <port\_type> [ <v\_port\_type\_list> ] ] [ sfm-information ] ] [ detail ]
  - show network-clock
  - show network-clock clock-selection-config
  - show network-clock port-config
  - show network-clock port-status
  - show network-clock ptp-ports
  - show network-clock source-nomination-config
  - show network-clock station-clock-config
  - show network-clock synchronization
  - show ntp status
  - show platform debug
  - show platform phy [ interface <port\_type> [ <v\_port\_type\_list> ] ]
  - show platform phy id [ interface <port\_type> [ <v\_port\_type\_list> ] ]
  - show platform phy mode [ interface <port\_type> [ <v\_port\_type\_list> ] ]
  - show port-security [ interface <port\_type> [ <plist> ] ]
  - show port-security address [ interface <port\_type> [ <plist> ] ]
  - show privilege
  - show process list [ detail ]

- show process load
- show ptp <clockinst> filter-type
- show ptp <clockinst> local-clock
- show ptp <clockinst> slave-cfg
- show ptp <clockinst> slave-table-unicast
- show ptp <clockinst> virtual-port
- show ptp <clockinst> { default | current | parent | time-property | filter | servo | clk | ho | uni | master-table-unicast | slave | { { port-state | port-statistics | port-ds | wireless | foreign-master-record } [ interface <port\_type> [ <v\_port\_type\_list> ] ] } | log-mode }
- show ptp cal
- show ptp cmdls default-ds
- show ptp cmdls { port-state | port-ds | port-statistics } interface <port\_type> [ <v\_port\_type\_list> ]
- show ptp ext
- show ptp ms-pdv all-apr-statistics cgu <cgu\_id>
- show ptp ms-pdv apr cgu <cgu\_id>
- show ptp ms-pdv cgu <cgu\_id> server <server\_id> status
- show ptp ms-pdv cur-path-delays cgu <cgu\_id>
- show ptp ms-pdv path-statistics cgu <cgu\_id>
- show ptp ms-pdv psl-fcl-config cgu <cgu\_id>
- show ptp rs422
- show ptp rs422 baudrate
- show ptp servo mode-ref
- show ptp servo source
- show ptp system-time
- show pvlan [ <pvlan\_list> ]
- show pvlan isolation [ interface <port\_type> [ <plist> ] ]
- show qos [ { interface [ <port\_type> [ <port> ] ] } | wred | { maps [ dscp-cos ] [ dscp-ingress-translation ] [ dscp-classify ] [ cos-dscp ] [ dscp-egress-translation ] [ { ingress [ <ing\_id> ] } ] [ { egress [ <egr\_id> ] } ] } | storm | { qce [ <qce> ] } ]
- show radius-server [ statistics ]
- show rmon alarm [ <id\_list> ]
- show rmon event [ <id\_list> ]
- show rmon history [ <id\_list> ]
- show rmon statistics [ <id\_list> ]
- show running-config [ all-defaults ]
- show running-config feature <feature\_name> [ all-defaults ]
- show running-config interface <port\_type> [ <list> ] [ all-defaults ]
- show running-config interface vlan <list> [ all-defaults ]
- show running-config line { console | vty } <list> [ all-defaults ]
- show running-config vlan { [ <vlan\_list> ] } [ all-defaults ]
- show sflow
- show sflow statistics { receiver [ <rcvr\_idx\_list> ] | samplers [ interface [ <samplers\_list> ] <port\_type> [ <v\_port\_type\_list> ] ] }
- show snmp
- show snmp access [ <group\_name> [ { v1 | v2c | v3 | any } [ { auth | noauth | priv } ] ] ]
- show snmp community [ <community> ]
- show snmp host [ <conf\_name> ]

- show snmp mib context
- show snmp mib ifmib ifIndex [ port ] [ aggregation ] [ vlan ]
- show snmp security-to-group [ { v1 | v2c | v3 } [ <security\_name> ] ]
- show snmp trap [ <source\_name> ]
- show snmp user [ <username> [ <engineID> ] ]
- show snmp view [ <view\_name> [ <oid\_subtree> ] ]
- show spanning-tree [ summary | active | { interface <port\_type> [ <v\_port\_type\_list> ] } | { detailed [ interface <port\_type> [ <v\_port\_type\_list\_1> ] ] } | { mst [ configuration | { <instance> [ interface <port\_type> [ <v\_port\_type\_list\_2> ] ] } ] } ] ] ]
- show stream [ <index\_list> ] status
- show svl { [ fid [ <fid\_list> ] ] | [ vlan [ <vlan\_list> ] ] }
- show switchport forbidden [ { vlan <vlan\_list> } | { name <name> } ]
- show system cpu status
- show system led status [ switch <switch\_list> ]
- show tacacs-server
- show tech-support
- show temperature monitor
- show terminal
- show thermal-protect [ interface <port\_type> [ <port\_list> ] ]
- show tsn flow meter [ <index\_list> ] status
- show tsn frame-preemption status [ interface <port\_type> [ <port> ] ]
- show tsn frer [ <inst\_list> ] [ statistics ] [ details ]
- show tsn stream filter [ <index\_list> ] { statistics | status }
- show tsn stream gate [ <index\_list> ] status
- show tsn tas status [ interface <port\_type> [ <port> ] ]
- show uddl [ interface <port\_type> [ <plist> ] ]
- show upnp
- show user-privilege
- show users [ myself ]
- show version [ brief ]
- show vlan [ id <vlan\_list> | name <name> | brief ] [ all ]
- show vlan ip-subnet [ <ipv4> ]
- show vlan mac [ address <mac\_addr> ]
- show vlan protocol [ eth2 { <etype> | arp | ip | ipx | at } ] [ snap { <oui> | rfc-1042 | snap-8021h } <pid> ] [ llc <dsap> <ssap> ]
- show vlan status [ interface <port\_type> [ <plist> ] ] [ admin | all | combined | conflicts | erps | gvrp | mrp | mstp | mvr | nas | rmirror | vcl | voice-vlan ]
- show voice vlan [ oui [ <oui> ] | interface <port\_type> [ <port\_list> ] ]
- show web privilege group [ <group\_name> ] level
- shutdown
- smac <mac>
- smac { <mac\_addr> [ / <mask\_mac\_addr> ] | any }
- snap { <oui> | rfc-1042 | snap-8021h } <pid>
- snmp-server
- snmp-server access <group\_name> model { v1 | v2c | v3 | any } level { auth | noauth | priv } [ read <view\_name> ] [ write <write\_name> ]
- snmp-server community <v3\_comm> [ { ip-range <v\_ipv4\_addr> <v\_ipv4\_netmask> | ipv6-range <v\_ipv6\_subnet> } ] { <v3\_sec> | encrypted <v3\_sec\_enc> }

- snmp-server contact <v\_line255>
- snmp-server engine-id local <engineID>
- snmp-server host <conf\_name>
- snmp-server location <v\_line255>
- snmp-server security-to-group model { v1 | v2c | v3 } name <security\_name> group <group\_name>
- snmp-server trap <source\_name> [ id <filter\_id> ] [ <oid\_subtree> { include | exclude } ]
- snmp-server user <username> engine-id <engineID> [ { md5 { <md5\_passwd> | { encrypted <md5\_passwd\_encrypt> } } | sha { <sha\_passwd> | { encrypted <sha\_passwd\_encrypt> } } ] [ priv { des | aes } { <priv\_passwd> | { encrypted <priv\_passwd\_encrypt> } } ] ]
- snmp-server view <view\_name> <oid\_subtree> { include | exclude }
- spanning-tree
- spanning-tree aggregation
- spanning-tree auto-edge
- spanning-tree bpdu-guard
- spanning-tree edge
- spanning-tree edge bpdu-filter
- spanning-tree edge bpdu-guard
- spanning-tree link-type { point-to-point | shared | auto }
- spanning-tree mode { stp | rstp | mstp }
- spanning-tree mst <instance> cost { <cost> | auto }
- spanning-tree mst <instance> port-priority <prio>
- spanning-tree mst <instance> priority <prio>
- spanning-tree mst <instance> vlan <v\_vlan\_list>
- spanning-tree mst forward-time <fwdtime>
- spanning-tree mst hello-time <hellotime>
- spanning-tree mst max-age <maxage> [ forward-time <fwdtime> ]
- spanning-tree mst max-hops <maxhops>
- spanning-tree mst name <name> revision <v\_0\_to\_65535>
- spanning-tree recovery interval <interval>
- spanning-tree restricted-role
- spanning-tree restricted-tcn
- spanning-tree transmit hold-count <holdcount>
- speed { 10 | 100 | 1000 | 2500 | 5g | 10g | 25g | force-clause-73 | auto { [ 10 ] [ 100 ] [ 1000 ] [ 2500 ] [ 5g ] [ 10g ] { [ no-hdx ] | [ no-fdx ] } } }
- state { open | closed }
- stream <inst>
- stream-id <id>
- svl fid <fid> vlan <vlan\_list>
- switchport access vlan <pvid>
- switchport forbidden vlan { add | remove } <vlan\_list>
- switchport hybrid acceptable-frame-type { all | tagged | untagged }
- switchport hybrid allowed vlan { all | none | [ add | remove | except ] <vlan\_list> }
- switchport hybrid egress-tag { none | all [ except-native ] }
- switchport hybrid ingress-filtering
- switchport hybrid native vlan <pvid>
- switchport hybrid port-type { unaware | c-port | s-port | s-custom-port }
- switchport mode { access | trunk | hybrid }

- switchport trunk allowed vlan { all | none | [ add | remove | except ] <vlan\_list> }
- switchport trunk native vlan <pvid>
- switchport trunk vlan tag native
- switchport vlan ip-subnet [ id <1-128> ] <ipv4> vlan <vid>
- switchport vlan mac <mac\_addr> vlan <vid>
- switchport vlan mapping <gid>
- switchport vlan mapping <gid> <vlan\_list> <tvid>
- switchport vlan mapping <gid> { both | ingress | egress } <vid> <tvid>
- switchport vlan protocol group <grp\_id> vlan <vid>
- switchport voice vlan discovery-protocol { oui | lldp | both }
- switchport voice vlan mode { auto | force | disable }
- switchport voice vlan security
- tacacs-server deadtime <minutes>
- tacacs-server host <host\_name> [ port <port> ] [ timeout <seconds> ] [ key { [ unencrypted ] <unencrypted\_key> | encrypted <encrypted\_key> } ]
- tacacs-server key { [ unencrypted ] <unencrypted\_key> | encrypted <encrypted\_key> }
- tacacs-server timeout <seconds>
- access-list ace [ update ] <ace\_id> [ next { <ace\_id\_next> | last } ] [ ingress { switch <ingress\_switch\_id> | switchport { <1-53> | <ingress\_switch\_port\_list> } | interface { <port\_type> <ingress\_port\_id> | <port\_type> [ <ingress\_port\_list> ] } | any } ] [ policy <policy> [ policy-bitmask <policy\_bitmask> ] ] [ tag { tagged | untagged | any } ] [ vid { <vid> | any } ] [ tag-priority { <tag\_priority> | 0-1 | 2-3 | 4-5 | 6-7 | 0-3 | 4-7 | any } ] [ dmac-type { unicast | multicast | broadcast | any } ] [ frame-type { any | etype [ etype-value { <etype\_value> | any } ] } ] [ smac { <etype\_smac> | any } ] [ dmac { <etype\_dmac> | any } ] [ arp [ sip { <arp\_sip> | any } ] [ dip { <arp\_dip> | any } ] [ smac { <arp\_smac> | any } ] [ arp-opcode { arp | rarp | other | any } ] [ arp-flag [ arp-request { <arp\_flag\_request> | any } ] [ arp-smac { <arp\_flag\_smac> | any } ] [ arp-tmac { <arp\_flag\_tmac> | any } ] [ arp-len { <arp\_flag\_len> | any } ] [ arp-ip { <arp\_flag\_ip> | any } ] [ arp-ether { <arp\_flag\_ether> | any } ] ] | ipv4 [ sip { <sipv4> | any } ] [ dip { <dipv4> | any } ] [ ip-protocol { <ipv4\_protocol> | any } ] [ ip-flag [ ip-ttl { <ip\_flag\_ttl> | any } ] [ ip-options { <ip\_flag\_options> | any } ] [ ip-fragment { <ip\_flag\_fragment> | any } ] ] | ipv4-icmp [ sip { <sipv4\_icmp> | any } ] [ dip { <dipv4\_icmp> | any } ] [ icmp-type { <icmpv4\_type> | any } ] [ icmp-code { <icmpv4\_code> | any } ] [ ip-flag [ ip-ttl { <ip\_flag\_icmp\_ttl> | any } ] [ ip-options { <ip\_flag\_icmp\_options> | any } ] [ ip-fragment { <ip\_flag\_icmp\_fragment> | any } ] ] | ipv4-udp [ sip { <sipv4\_udp> | any } ] [ dip { <dipv4\_udp> | any } ] [ sport { <sportv4\_udp\_start> [ to <sportv4\_udp\_end> ] | any } ] [ dport { <dportv4\_udp\_start> [ to <dportv4\_udp\_end> ] | any } ] [ ip-flag [ ip-ttl { <ip\_flag\_udp\_ttl> | any } ] [ ip-options { <ip\_flag\_udp\_options> | any } ] [ ip-fragment { <ip\_flag\_udp\_fragment> | any } ] ] | ipv4-tcp [ sip { <sipv4\_tcp> | any } ] [ dip { <dipv4\_tcp> | any } ] [ sport { <sportv4\_tcp\_start> [ to <sportv4\_tcp\_end> ] | any } ] [ dport { <dportv4\_tcp\_start> [ to <dportv4\_tcp\_end> ] | any } ] [ ip-flag [ ip-ttl { <ip\_flag\_tcp\_ttl> | any } ] [ ip-options { <ip\_flag\_tcp\_options> | any } ] [ ip-fragment { <ip\_flag\_tcp\_fragment> | any } ] ] [ tcp-flag [ tcp-fin { <tcpv4\_flag\_fin> | any } ] [ tcp-syn { <tcpv4\_flag\_syn> | any } ] [ tcp-rst { <tcpv4\_flag\_rst> | any } ] [ tcp-psh { <tcpv4\_flag\_psh> | any } ] [ tcp-ack { <tcpv4\_flag\_ack> | any } ] [ tcp-urg { <tcpv4\_flag\_urg> | any } ] ] | ipv6 [ next-header { <next\_header> | any } ] [ sip { <sipv6> | sip-bitmask <sipv6\_bitmask> ] | any } ] [ hop-limit { <hop\_limit> | any } ] [ ipv6-icmp [ sip { <sipv6\_icmp> | sip-bitmask <sipv6\_bitmask\_icmp> ] | any } ] [ icmp-type { <icmpv6\_type>

- ```

| any } ] [ icmp-code { <icmpv6_code> | any } ] [ hop-limit { <hop_limit_icmp> | any }
] | ipv6-udp [ sip { <sipv6_udp> [ sip-bitmask <sipv6_bitmask_udp> ] | any } ] [ sport {
<sportv6_udp_start> [ to <sportv6_udp_end> ] | any } ] [ dport { <dportv6_udp_start>
[ to <dportv6_udp_end> ] | any } ] [ hop-limit { <hop_limit_udp> | any } ] | ipv6-tcp [ sip {
<sipv6_tcp> [ sip-bitmask <sipv6_bitmask_tcp> ] | any } ] [ sport { <sportv6_tcp_start> [
to <sportv6_tcp_end> ] | any } ] [ dport { <dportv6_tcp_start> [ to <dportv6_tcp_end>
] | any } ] [ hop-limit { <hop_limit_tcp> | any } ] [ tcp-flag [ tcp-fin { <tcpv6_flag_fin>
| any } ] [ tcp-syn { <tcpv6_flag_syn> | any } ] [ tcp-rst { <tcpv6_flag_rst> | any } ]
[ tcp-psh { <tcpv6_flag_psh> | any } ] [ tcp-ack { <tcpv6_flag_ack> | any } ] [ tcp-
urg { <tcpv6_flag_urg> | any } ] ] } ] [ action { permit | deny | filter { switchport
<filter_switch_port_list> | interface <port_type> [ <filter_port_list> ] } } ] [ rate-limiter
{ <rate_limiter_id> | disable } ] [ mirror [ disable ] ] [ logging [ disable ] ] [ shutdown [ disable
] ] [ lookup-second [ disable ] ] [ redirect { switchport { <1-53> | <redirect_switch_port_list>
} | interface { <port_type> <redirect_port_id> | <port_type> [ <redirect_port_list> ] }
| disable } ]

```
- temperature monitor { board | junction } { [ low <l\_temp> ] [ high <h\_temp> ] [ critical <c\_temp> ] [ hysteresis <h\_val> ] }
  - terminal editing
  - terminal exec-timeout <min> [ <sec> ]
  - terminal help
  - terminal history size <history\_size>
  - terminal length <lines>
  - terminal width <width>
  - thermal-protect grp <grp>
  - thermal-protect grp <grp\_list> temperature <new\_temp>
  - timers basic <update\_timer> <invalid\_timer> <garbage\_collection\_timer>
  - traceroute ip { <domain\_name> | <ip\_addr> } [ dscp <dscp> ] [ timeout <timeout> ] [ { saddr <src\_addr> | sif { <port\_type> <src\_if> | vlan <vlan\_id> } } ] [ probes <probes> ] [ firstttl <firstttl> ] [ maxttl <maxttl> ] [ icmp ] [ numeric ]
  - traceroute ipv6 { <domain\_name> | <ip\_addr> } [ dscp <dscp> ] [ timeout <timeout> ] [ saddr <src\_addr> ] [ sif { <port\_type> <src\_if> | vlan <vlan\_id> } ] [ probes <probes> ] [ maxttl <maxttl> ] [ numeric ]
  - tsn flow meter <inst>
  - tsn frame-preemption
  - tsn frame-preemption ignore-lldp
  - tsn frame-preemption queue <queue>
  - tsn frame-preemption verify-disable
  - tsn frer <inst>
  - tsn frer <inst> reset [ latent-error ]
  - tsn ptp-check procedure { none | ptp | wait }
  - tsn ptp-check ptp-port <ptp\_port\_val>
  - tsn ptp-check timeout <time\_val>
  - tsn stream filter <inst>
  - tsn stream gate <inst>
  - tsn tas always-guard-band
  - tsn tas base-time seconds <seconds> nanoseconds <nanoseconds>
  - tsn tas config-change
  - tsn tas control-list index <index> gate-state queue <queue> { open | closed } time-interval

- <interval> [ operation { set | set-hold | set-release } ]
- tsn tas control-list-length <length>
- tsn tas cycle-time <time> { ms | us | ns }
- tsn tas cycle-time-extension <extension>
- tsn tas gate-enabled
- tsn tas gate-states queue <queue> { open | closed }
- tsn tas max-sdu queue <queue> <sdu>
- type { port | vlan <vid> }
- uddl port [ aggressive ] [ message time-interval <v\_interval> ]
- uddl { aggressive | enable | message time-interval <v\_interval> }
- upnp
- upnp advertising-duration <v\_100\_to\_86400>
- upnp ip-addressing-mode { dynamic | static }
- upnp static interface vlan <v\_vlan\_id>
- url <url>
- username { default-administrator | <input\_username> } privilege <priv> password { unencrypted <unencyr\_password> | encrypted <encry\_password> | none }
- uuid <uuid>
- vcl { dmacdip | smacsip }
- vendor class-identifier <class\_id> specific-info <hexval>
- veriphy [ { interface <port\_type> [ <port\_list> ] } ]
- version { 1 | 2 }
- version { v1 [ { <v1\_comm> | encrypted <v1\_comm\_sec> } ] | v2 [ { <v2\_comm> | encrypted <v2\_comm\_sec> } ] | v3 engineID <v\_word10\_to\_64> [ <securityname> ] }
- version { v1 | v2 }
- vlan <vlist>
- vlan ethertype s-custom-port <etype>
- vlan protocol { { eth2 { <etype> | arp | ip | ipx | at } } | { snap { <oui> | rfc-1042 | snap-8021h } <pid> } | { llc <dsap> <ssap> } } group <grp\_id>
- vlan { inherit | <vid> }
- vlan { untagged | <vid> [ pcp <pcp> ] }
- voice vlan
- voice vlan aging-time <aging\_time>
- voice vlan class { <traffic\_class> | low | normal | medium | high }
- voice vlan oui <oui> [ description <description> ]
- voice vlan vid <vid>
- wait-to-restore <wtr>
- web privilege group <group\_name> level { [ configRoPriv <configRoPriv> ] [ configRwPriv <configRwPriv> ] [ statusRoPriv <statusRoPriv> ] [ statusRwPriv <statusRwPriv> ] }
- width <width>
- working interface <port\_type> <port>
- working sf-trigger { link | { mep domain <md\_name> service <ma\_name> mep-id <mepid> } }
- working-mep domain <md\_name> service <ma\_name> mep-id <mepid>

### 3 Генерация команд ICLI

В данном разделе описывается способ написания файлов сценария для автоматического создания файлов С и Н для автоматической регистрации команд. Перед использованием команды в промышленном интерфейсе командной строки (ICLI) требуется регистрация.

#### 3.1 Поток генерации

На рисунке 1 показан процесс генерации команд.

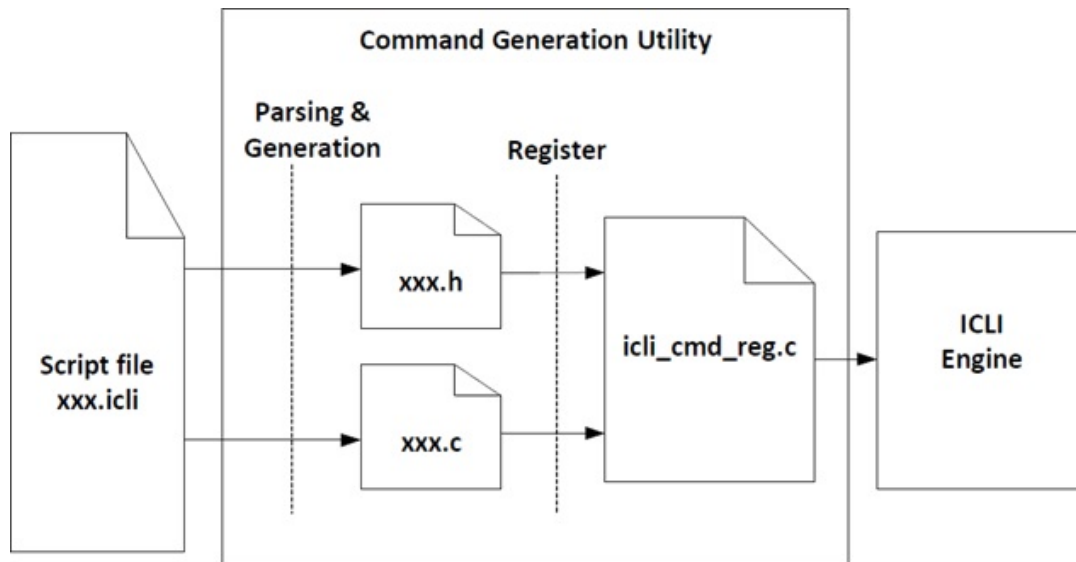


Рис. 1: Поток генерации команд

Алгоритм генерации команд выглядит следующим образом:

1. Написать файл сценария (*xxx.icli*).
2. С помощью утилиты генерации команд выполнить следующие действия:
  - Автоматически создать соответствующие файлы С и Н (*xxx\_icli.c* и *xxx\_icli.h*).
  - Автоматически зарегистрировать команды в механизме ICLI (*icli\_cmd\_reg.c*).

Преимущество написания файлов сценариев вместо написания кода на С заключается в следующем:

- Простота написания — файл сценария позволяет сосредоточиться на выполнении команд. Все остальные задачи и детали, такие как регистрация команд, синтаксический анализ, типы переменных и использование памяти, обрабатываются утилитой генерации команд.
- Сокращение усилий и затрат на написание кода — файл сценария позволяет писать часть исходного кода, связанную с реализацией. Остальной исходный код генерируется утилитой, чтобы свести к минимуму ошибки написания кода, усилия, несоответствия и затраты.
- Игнорировать детали реализации, такие как соглашения о написании кода.

#### 3.2 Дизайн команд

Команда состоит из нескольких слов с пробелом, используемым для разделения каждого слова. Слово может быть ключевым словом или переменной. Первое слово в команде должно быть ключевым словом и не может быть переменной.

Следующий синтаксис команды является первым шагом в разработке команды ICLI.

Синтаксис команды:

keyword word word ...

- Слово — слово в команде должно быть либо ключевым словом, либо переменной.
- Ключевое слово — ключевое слово в команде является постоянным словом.
- Переменная — переменная представляет собой тип переменной, заключенный в набор угловых скобок (< >), а тип переменной представляет тип пользовательского ввода. Механизм ICLl анализирует и проверяет, соответствует ли пользовательский ввод соответствующему типу переменной.

### Пример:

```
ip address <ipv4_addr> <ipv4_netmask>
```

Слова:

ip, address, <ipv4\_addr> и <ipv4\_netmask>

Ключевые слова:

ip и address

Переменные:

<ipv4\_addr> и <ipv4\_netmask>

Пользователем указываются строки в качестве входных данных, чтобы соответствовать типу:

<ipv4\_addr> и <ipv4\_netmask>.

Пример допустимой команды:

```
ip address 10.1.1.1 255.255.0.0.
```

где 10.1.1.1 - это <ipv4\_addr> и 255.255.0.0 - это <ipv4\_netmask>.

Данные элементы синтаксиса позволяют разработчикам модулей гибко и эффективно настраивать свои команды.

### 3.2.1 Переменная « < > »

Переменная должна изображаться заключенной в < >. Данное обозначение представляет собой тип переменной, которую механизм ICLl ожидает в качестве входных данных. Пользователь должен вводить строку, соответствующую типу переменной.

### 3.2.2 Исключающее OR «|» (Или)

Исключающее OR (или) делает обязательным выбор только одного из слов. Используется с обязательным { } или необязательным [ ].

### 3.2.3 Обязательный «{ }»

Одно из слов, заключенных в фигурные скобки { }, должно быть использовано. Отдельные слова внутри фигурных скобок должны быть разделены оператором исключения |. Например, в команде  $a \{ b / c \} d$  допустимые пользовательские вводы следующие:

- a b d
- a c d

Происходит исключение  $\{ b / c \} = b$  или  $c$ , с обязательным выбором одного из вариантов. Ниже приведен недопустимый пользовательский ввод:

- a d

Если { } не имеет внутри себя | (конструкции or), то это допустимо, но не имеет никакого смысла.

### 3.2.4 Необязательный «[ ]»

Пользователем допускается не использовать слова, заключенные в квадратные скобки [ ], в качестве входных данных. Отдельные слова должны быть разделены оператором исключения |. Например, команда  $a [b / c] d$  имеет следующие допустимые пользовательские вводы:

- a b d
- a c d
- a d

### 3.2.5 Случайный Необязательный «[ ] [ ] ... [ ]»

Необязательный ввод, в рамках одной команды, с опциональными вариантами, называется случайным необязательным. Ввод значений в таких случаях, не привязан к какому-то порядку следования слов. Например, команда  $a [b] [c] [d], [b] [c] [d]$  является случайной необязательной, а допустимые пользовательские вводы следующие:

- a
- a d
- a c b
- a d b
- a b c d
- a d c b

Использовать { } для привязки к порядку следования слов. Например, для команды  $a [b] [c] [d] [e], [b] [c] [d] [e]$  допустимы следующие пользовательские вводы:

- a
- a c
- a b c
- a b e d
- a c e d
- a b c e d

Ниже приведен недопустимый пользовательский ввод:

- a c b

### 3.2.6 Случайное обязательное «{ [ ] [ ] ... [ ] }\*n»

Случайный необязательный параметр допускает ввод без необязательных слов. Например, команда  $a [b] [c] [d]$ , допускает ввод команды из одного слова  $a$ , поскольку  $b$ ,  $c$  и  $d$  не являются обязательными. Если требуется случайный необязательный параметр хотя бы с одним словом в качестве входных данных, то синтаксис случайного обязательного должен обеспечить эту функцию. Синтаксис для случайного обязательного имеет следующие правила:

- Следует заключить случайный параметр обязательным как  $\{[\text{случайный необязательный}]\}$
- Следует убедиться, что  $*n$  следует за последней фигурной скобкой  $\}$  и между ними нет пробела, где  $n$  означает одну цифру.
- Следует использовать  $*n$ , чтобы указать, что в качестве входных данных должно быть предоставлено не менее  $n$  случайных слов.

Например, команда  $a \{[b] [c] [d]\}^*1$ ,  $\{[b] [c] [d]\}^*1$  случайно обязательная, и хотя бы одна из команд  $b$ ,  $c$  и  $d$  должна быть предоставлена в качестве входных данных. Ниже приведен допустимый синтаксис случайных обязательных команд:

- $a \{[b] [c] [d]\}^*1$
- $a \{[b] [c] [d]\}^*3$
- $a \{[b] [c] [d]\}^*9$
- $a \{b | \{[c] [d]\}^*1\}$
- $a \{[b] | \{[c] [d]\}^*1\}$
- $a \{b | \{[c] [d] \{[e] [f] [g]\}^*2\}\}^*1$

Следующие примеры являются недопустимым синтаксисом случайных обязательных команд:

- $a \{[b] [c] [d]\} *1$  (не допускается пробел между  $\}$  и  $*1$ )
- $a \{[b] [c] [d]\}^*10$  ( $*10$  больше одной цифры)
- $a \{[b] [c] [d]\}^*1a$  ( $*1a$  содержит не только цифры)

Эти команды имеют недопустимый синтаксис случайных обязательных команд, они по-прежнему являются допустимыми командами для ICL. В случаях с недопустимым синтаксисом для случайных обязательных  $*1$ ,  $*10$  и  $*1a$  рассматриваются как отдельные ключевые слова.

Так же, если  $n$  больше, чем общее количество  $m$  (вариантов случайных слов), то вместо  $n$  будет применено  $m$ . Например, команда  $\{[b] [c] [d]\}^*9$  допустима, но  $*9$  превышает количество вариантов  $\{[b] [c] [d]\}$  равное трём. Это означает что команда  $\{[b] [c] [d]\}^*9$  станет равной  $\{[b] [c] [d]\}^*3$ , и что все слова ( $b$ ,  $c$  и  $d$ ) должны будут быть использованы.

### 3.2.7 Повторить «...»

Повторы позволяют установить количество вводов в командах, имеющих  $[ | ]$  и/или  $\{ | \}$ . Повторы записываются троеточием  $\dots$ . Количество повторов равно количеству выборов в  $[ | ]$  и/или  $\{ | \}$ .

Примеры:

- $\{a|b|c\} \dots \}$  Равнозначно  $\{a|b|c\} \{a|b|c\} \{a|b|c\}$ .
- $\{[a|b|c] \dots \}$  Равнозначно  $\{[a|b|c] [[a|b|c] [[a|b|c]]]\}$ .
- $[[a|b|c] \dots ]$  Равнозначно  $[[a|b|c] [a|b|c] [a|b|c]]$ .
- $[[a|b|c] \dots ]$  Равнозначно  $[[a|b|c] [[a|b|c] [[a|b|c]]]]$ .

### 3.2.8 Цикл «( )\*N»

Данная подкоманда заключена в (), а  $N$  означает, что подкоманда может быть повторена  $1..N$  раз. Соответствующие переменные будут объявлены как массивы, поэтому для получения значений допускается использовать индекс массива. Следующий код является примером:

```
CMD_BEGIN
COMMAND = debug loop (uint <uint>)*3
...
CMD_VAR =
CMD_VAR =
CMD_VAR = b
CMD_VAR = value
...
CMD_END
```

Тогда переменные будут сгенерированы следующим образом:

```
BOOL b[3];
u32 value[3];
```

Таким образом, индекс массива используется для получения пользовательского ввода для каждого входа цикла. Количество циклов ( $N$ ) может быть константой, определяемой в коде следующим образом:

```
COMMAND = debug loop (uint <uint>)*_CONST_NUM_
```

Где

```
_CONST_NUM_
```

определяется в файле ICLI или другом заголовочном файле.

### 3.2.9 Ограничения

Синтаксический анализ механизма ICLI не допускает совпадения одних и тех же слов в обязательных { } или необязательных [ ]. В следующих примерах показано ограничение:

#### Пример 1

```
snmp client version <uint>
snmp server | client address <ipv4_ucast>
```

Ключевое слово *client* недопустимо перекрывается, так как *client* является обязательным { } во второй команде.

#### Пример 2

```
snmp client version <uint>
snmp [client] address <ipv4_ucast>
```

Ключевое слово *client* недопустимо перекрывается, так как *client* является необязательным [ ] во второй команде.

Следующие переработанные примеры позволяют избежать этой проблемы:

### Переработанный пример 1

```
snmp client { version <uint> | address <ipv4_ucast> }
snmp server address <ipv4_ucast>
```

### Переработанный пример 2

```
snmp client { version <uint> | address <ipv4_ucast> }snmp address <ipv4_ucast>
```

По возможности следует использовать один и тот же плоский префикс. Клиент snmp — это тот же плоский префикс в примерах.

Ниже приведен пример:

- a b x
- a c x
- a d x
- a b y
- a c y

Из-за ограничения следующий дизайн не допускается:

- a { b | c | d } x
- a { b | c } y

Вместо этого допускается следующая конструкция слева направо:

- a b { x | y }
- a c { x | y }
- a d x

## 3.3 Состав файла сценария

Расширение файла сценария — .icli. Каждый файл сценария состоит из сегментов, каждый из которых состоит из тегов. Для тега со значением необходимо использовать =, чтобы присвоить тегу значение (например, *TAG = TAG\_VALUE*). Для длинного *TAG\_VALUE* следует использовать \ для объединения со следующей строкой.

Внутри каждого сегмента также могут быть подсегменты. Чтобы прокомментировать строку внутри файла скрипта, используется !, # или // в начале строки.

В файле скрипта имеются четыре сегмента:

- Module
- Include
- Function
- Command

Сегмент Command имеет подсегменты, такие как подсегмент VARIABLE и подсегмент CODE. Ниже представлена сегментированная структура типичного файла сценария:

```

Module Segment | MODULE_IF_FLAG =
                | INCLUDE_BEGIN
Include Segment | ...
                | INCLUDE_END
                | FUNCTION_BEGIN
Function Segment | ...
                 | FUNCTION_END
                 | CMD_BEGIN
                 | ...
                 | COMMAND =
                 | FUNC_NAME =
                 | ...
Command Segment | ... }Variable Sub-Segment
                 | VARIABLE_END |
                 | CODE_BEGIN |
                 | ... } Code Sub-Segment
                 | CODE_END |
                 | CMD_END

```

### 3.3.1 Сегмент Module

Данный сегмент управляет настройками всего модуля. Единственным тегом для этого сегмента является *MODULE\_IF\_FLAG =*.

Значение этого тега помогает узнать, зарегистрированы ли все команды этого модуля или скомпилирован ли сгенерированный файл С/Н.

Значение тега добавляется к директиве *#if* в файле *icli\_cmd\_reg.c* и сгенерированных файлах С/Н. Следовательно, если значение тега не равно *TRUE*, команды не будут зарегистрированы и исходный код не будет скомпилирован.

#### Пример:

In *stp.icli*, the tag value is *B4CDS\_SW\_OPTION\_STP*.

```
MODULE_IF_FLAG = defined(B4CDS_SW_OPTION_STP)
```

```
...
```

In *icli\_cmd\_reg.c*, the auto-registration is enclosed by the tag value.

```
...
```

```
#if defined(B4CDS_SW_OPTION_STP)
```

```
&stp_icli_cmd_register,
```

```
#else
```

```
NULL,
```

```
#endif
```

```
...
```

In the generated C/H files, the whole context is enclosed by the tag value.

```
#if defined(B4CDS_SW_OPTION_STP)
```

```
...
```

```
...
```

```
...
```

```
#endif
```

### 3.3.2 Сегмент Include

Данный сегмент указывает включаемые файлы, необходимые для компиляции тела сгенерированного файла C.

Файл *icli\_api.h* автоматически включается генератором и не должен указываться здесь. Содержимое внутри сегмента вставляется точно так же, как и в сгенерированный файл C. Этот сегмент имеет следующие два тега:

- *INCLUDE\_BEGIN* — этот тег является необязательным. Он отмечает начало сегмента Include.
- *INCLUDE\_END* — этот тег является обязательным, если тег *INCLUDE\_BEGIN* существует. Этот тег отмечает конец сегмента Include.

Ниже приведена структура сегмента Include. Это содержимое должно быть вставлено как есть в сгенерированный файл C:

```
INCLUDE_BEGIN
#include <stdio.h>
# if STP_MODE_RSTP
#include "rstp.h"
#endif
INCLUDE_END
```

### 3.3.3 Сегмент Function

Сегмент Function используется, чтобы сделать следующее:

- Написать локальные функции, используя сегмент command.
- Объявить константы, макросы, типы данных или статические переменные для функций или тел команд.

Содержимое вставляется без изменений в сгенерированный файл C. Этот сегмент имеет следующие два тега:

- *FUNCTION\_BEGIN* — тег отмечает начало сегмента Function. Это необязательный тег.
- *FUNCTION\_END* — данный тег является обязательным, если существует тег *FUNCTION\_BEGIN*. Тег отмечает конец сегмента Function.

Ниже приведена структура сегмента Function. Данное содержимое должно быть вставлено без изменений в сгенерированный файл C:

```
FUNCTION_BEGIN
#define _IS_TRUE (x) ( (x) == TRUE)
static i32 svariable;
Static i32 _vid_get (void) {
return vlan_id;
}
BOOL vid_set (i32 vid) {
vlan_id = vid;
return TRUE;
}
FUNCTION_END
```

### 3.3.4 Сегмент Command

Данный сегмент определяет реализацию команды — один сегмент для одной команды. Реализация содержит привилегию, режим работы, свойство команды, справку и экземпляр выполнения.

Следующие перечисления представляют собой теги в сегменте Command:

- **CMD\_BEGIN**  
Обязательный тег. Он отмечает начало сегмента Command.
- **CMD\_END**  
Обязательный тег. Он отмечает конец сегмента Command.
- **DOC\_CMD\_DESC =**  
Необязательный тег. Необходимо использовать \ для объединения строк. Для каждого описания может использоваться несколько *DOC\_CMD\_DESC*.
- **DOC\_CMD\_DEFAULT =**  
Необязательный тег для обозначения значения по умолчанию для команды.
- **DOC\_CMD\_USAGE =**  
Необязательный тег. Используется \, чтобы объединить строки для рекомендаций по использованию команды. Для каждого описания может использоваться несколько *DOC\_CMD\_DESC*.
- **DOC\_CMD\_EXAMPLE =**  
Необязательный тег. Используется \ чтобы объединить строки для примеров команд. Для каждого описания может использоваться несколько *DOC\_CMD\_DESC*.
- **COMMAND =**  
Обязательный тег. Командная строка может быть определена как:
  - обязательное ({ })
  - необязательное ([ ])
  - или (|)
  - переменная (<>).
- **FUNC\_NAME =**  
Данный необязательный тег называет функцию выполнения команды. Утилита генерации команд ICLi использует значение тега для автоматического создания соответствующей статической функции в сгенерированном файле C. Эта статическая функция, функция выполнения команды, выполняется, когда пользователь вводит допустимую команду. Если *FUNC\_NAME* не определено, утилита генерации команд ICLi генерирует имя случайным образом.
- **FUNC\_REUSE =**  
Данный необязательный тег определяет имя повторно используемой функции выполнения другой команды. Тег *FUNC\_REUSE* повторно использует тело кода другой команды, где тело кода включает подсегмент *VARIABLE* и подсегмент *CODE*. Таким образом, когда команда повторно использует тело кода другой команды, команда может определить свои собственные свойства команды, такие как привилегия и строка справки. Нет необходимости писать подсегмент *VARIABLE* и *CODE*.

В следующем примере *cmd 2* повторно использует тело кода *cmd 1*. В то же время *cmd 2* определил свои собственные строки справки и другие свойства команды.

```

CMD_BEGIN
COMMAND= test-cmd 1
FUNC_NAME = _cmd_1_cb
HELP= Test command
HELP= Command 1

```

```

VARIABLE_BEGIN
...
VARIABLE_END
CODE_BEGIN
...
CODE_END
CMD_END
CMD_BEGIN
COMMAND= debug-cmd 2
FUNC_REUSE = _cmd_1_cb
HELP= Debug command
HELP= Command 2
CMD_END

```

• **PRIVILEGE =**

Данный обязательный тег определяет уровень привилегий команды *ICLI\_PRIVILEGE\_XX*, определенный в файле *icli\_porting.h*. Доступные уровни привилегий варьируются от *ICLI\_PRIVILEGE\_0* до *ICLI\_PRIVILEGE\_15*. Чем больше число, тем выше уровень привилегий команды. Команда может быть выполнена только сеансом с более высокими или равными привилегиями.

Например, если привилегия команды *ICLI\_PRIVILEGE\_7*, то сеансы с уровнем привилегии *ICLI\_PRIVILEGE\_0* *ICLI\_PRIVILEGE\_6* не могут получить доступ к команде. Сеансы с уровнем привилегий *ICLI\_PRIVILEGE\_7* *ICLI\_PRIVILEGE\_15* могут получить доступ к этой команде.

Таблица 1 — Описание свойств команд

Имя свойства	Характеристики
<i>ICLI_CMD_PROP_ENABLE</i>	Указывает, является ли команда исполняемой или нет. Когда пользователь предоставляет допустимую команду в качестве входных данных, механизм ICLI проверяет это свойство, чтобы решить, может ли выполняться соответствующая функция выполнения команды. Если свойство равно <i>ICLI_CMD_PROP_ENABLE</i> , то выполняется функция выполнения команды. <i>ICLI_CMD_PROP_ENABLE</i> и <i>ICLI_CMD_PROP_DISABLE</i> являются взаимоисключающими свойствами. Синтаксис: <i>#define ICLI_CMD_PROP_ENABLE 0x00</i>
<i>ICLI_CMD_PROP_DISABLE</i>	Указывает, является ли команда исполняемой или нет. Когда пользователь предоставляет допустимую команду в качестве входных данных, механизм ICLI проверяет это свойство, чтобы решить, может ли выполняться соответствующая функция выполнения команды.

Таблица 1 — Продолжение

Имя свойства	Характеристики
	<p>Если свойство равно <i>ICLI_CMD_PROP_DISABLE</i>, то функция выполнения команды не выполняется. <i>ICLI_CMD_PROP_ENABLE</i> и <i>ICLI_CMD_PROP_DISABLE</i> являются взаимоисключающими свойствами.</p> <p>Синтаксис:  <code>#define ICLI_CMD_PROP_DISABLE 0x01</code></p>
<i>ICLI_CMD_PROP_VISIBLE</i>	<p>Указывает, видна ли команда пользователям. Когда пользователь нажимает клавишу TAB или вводит ? для показа справки, механизм ICLI проверяет данное свойство, чтобы решить, следует ли отображать команду пользователю. Если свойство равно <i>ICLI_CMD_PROP_VISIBLE</i>, команда отображается. <i>ICLI_CMD_PROP_VISIBLE</i> и <i>ICLI_CMD_PROP_INVISIBLE</i> являются взаимоисключающими свойствами.</p> <p>Синтаксис:  <code>#define ICLI_CMD_PROP_VISIBLE 0x00</code></p>
<i>ICLI_CMD_PROP_INVISIBLE</i>	<p>Указывает, видна ли команда пользователям. Когда пользователь нажимает клавишу TAB или вводит ? для показа справки, механизм ICLI проверяет данное свойство, чтобы решить, следует ли отображать команду пользователю. Если свойство равно <i>ICLI_CMD_PROP_INVISIBLE</i>, команда не отображается для пользователя. <i>ICLI_CMD_PROP_VISIBLE</i> и <i>ICLI_CMD_PROP_INVISIBLE</i> являются взаимоисключающими свойствами.</p> <p>Синтаксис:  <code>#define ICLI_CMD_PROP_INVISIBLE 0x02</code></p>
<i>ICLI_CMD_PROP_GREP</i>	<p>Указывает, есть ли у команды функция grep для форматирования вывода. Если свойство равно <i>ICLI_CMD_PROP_GREP</i>, команда имеет функцию grep. <i>ICLI_CMD_PROP_GREP</i> и <i>ICLI_CMD_PROP_NOT_GREP</i> являются взаимоисключающими свойствами.</p> <p>Синтаксис:  <code>#define ICLI_CMD_PROP_GREP 0x04</code></p>
<i>ICLI_CMD_PROP_NOT_GREP</i>	<p>Указывает, есть ли у команды функция <i>grep</i> для форматирования вывода. Если свойство равно <i>ICLI_CMD_PROP_NOT_GREP</i>, команда не имеет функции <i>grep</i>.</p>

Таблица 1 — Продолжение

Имя свойства	Характеристики
	<p><i>ICLI_CMD_PROP_GREP</i> и <i>ICLI_CMD_PROP_NOT_GREP</i> являются взаимоисключающими свойствами.</p> <p>Синтаксис:  <code>#define ICLI_CMD_PROP_NOT_GREP 0x00</code></p>
<i>ICLI_CMD_PROP_LOOSELY</i>	<p>Указывает, что дополнительные слова разрешены после ввода полной допустимой команды. <i>ICLI_CMD_PROP_LOOSELY</i> и <i>ICLI_CMD_PROP_STRICTLY</i> являются взаимоисключающими свойствами.</p> <p>Синтаксис:  <code>#define ICLI_CMD_PROP_LOOSELY 0x08</code></p>
<i>ICLI_CMD_PROP_STRICTLY</i>	<p>Указывает, разрешает ли механизм ICLI дополнительные слова после ввода полной допустимой команды. Если свойство равно <i>ICLI_CMD_PROP_STRICTLY</i>, то команда, завершенная с дополнительной строкой, становится недействительной. <i>ICLI_CMD_PROP_LOOSELY</i> и <i>ICLI_CMD_PROP_STRICTLY</i> являются взаимоисключающими свойствами.</p> <p>Синтаксис:  <code>#define ICLI_CMD_PROP_STRICTLY 0x00</code></p>

**Пример**

```

COMMAND= ip dhcp snooping
...
PROPERTY= ICLI_CMD_PROP_LOOSELY
...
CMD_END

```

В данном случае командная строка `ip dhcp snooping` для включения `ip dhcp snooping` действительна по следующим причинам:

- *ip dhcp snooping* является допустимой командой.
- *ICLI\_CMD\_PROP\_LOOSELY* просит механизм ICLI игнорировать дополнительную строку, чтобы включить функцию *ip dhcp snooping*.

Но если *PROPERTY= ICLI\_CMD\_PROP\_STRICTLY*, то эта командная строка недействительна. Механизм ICLI по-прежнему анализирует строку, чтобы включить функцию *ip dhcp snooping*, но идентифицирует команду как недопустимую.

В случае, если значение тега игнорируется, значение по умолчанию равно 0, а следующее свойство является свойством по умолчанию.

**Примечание** – Значение *ICLI\_CMD\_PROP\_ENABLE* равно 0, и значение *ICLI\_CMD\_PROP\_VISIBLE* также равно 0. Если ожидаемое свой-

ство команды *ICLI\_CMD\_PROP\_ENABLE* | *ICLI\_CMD\_PROP\_VISIBLE*  
 | *ICLI\_CMD\_PROP\_GREP*, просто определите *PROPERTY* =  
*ICLI\_CMD\_PROP\_GREP*.

- **CMD\_MODE** =

Обязательный тег определяет командный режим *ICLI\_CMD\_MODE\_XXXX*, определенный в *icli\_porting.h*, в котором работает команда. Командный режим — это метод классификации команд. Команда видна и может быть выполнена только в рабочем режиме. Итак, разработчик модуля должен знать о разворачивании каждой из команд.

Команда может быть определена в нескольких командных режимах, если команда работает в нескольких командных режимах:

```
COMMAND= temperature { get | set <0-100> }
```

```
...
```

```
CMD_MODE= ICLI_CMD_MODE_EXEC
```

```
CMD_MODE= ICLI_CMD_MODE_GLOBAL_CONFIG
```

```
...
```

**Примечание:** — Существует два способа повторного использования команды: *FUNC\_NAME/FUNC\_REUSE* и *CMD\_MODE*. Следует использовать *CMD\_MODE*, если команда работает в разных командных режимах и все свойства команды одинаковы в разных командных режимах — одни и те же привилегии и одна и та же справка. *FUNC\_NAME/FUNC\_REUSE* используется для всех остальных случаев.

- **GOTO\_MODE** =

Необязательный тег определяет следующий командный режим после выполнения команды. Данный тег предназначен не только для выполнения команды, но и для разбора. Он автоматически генерируется тегом *SUB\_MODE*.

- **SUB\_MODE** =

Необязательный тег. Командный режим, *ICLI\_CMD\_MODE\_XXXX*, определенный в *icli\_porting.h*, который команда создаст и перейдет в него после выполнения. Если данный тег определен, следующие задачи выполняются автоматически. Другими словами, все необходимые задачи созданные в *SUB\_MODE* могут выполняться автоматически через этот тег.

- **IF\_FLAG** =

Необязательный тег позволяет условному флагу *#if* заключить команду. Значение тега отображается как постфикс директивы *#if*. Например, *IF\_FLAG = defined*, то значением тега является *#if defined*. Используется данный тег, чтобы определить команду условий, от которой зависит выполнение. Например, чтобы отключить эту команду, необходимо определить *IF\_FLAG = 0*, чтобы заключить команду в *#if 0*.

- **CMD\_VAR** =

Необязательный тег определяет переменную *c* для соответствующего слова командной строки. Тег позволяет однозначно сопоставлять один к одному каждое слово в командной строке.

**Пример**

```
COMMAND = a [ b | c ] d
```

```
CMD_VAR = v1
```

```
CMD_VAR = v2
```

```
CMD_VAR = v3
```

```
CMD_VAR = v4
```

где

- *v1* сопоставлено с *a*
- *v2* сопоставлено с *b*
- *v3* сопоставлено с *c*
- *v4* сопоставлено с *d*

Если нет необходимости в командных переменных *a* и *c*, следует игнорировать значения тегов, но теги все равно нужны для ведения переписки:

```
COMMAND = a [ b | c ] d
CMD_VAR =
CMD_VAR = v2
CMD_VAR =
CMD_VAR = v4
```

где

- *v2* сопоставлено с *b*
- *v4* сопоставлено с *d*

Переменная *C* будет автоматически объявлена в функции выполнения команды сгенерированного файла *C* в соответствии с типом слова в командной строке *COMMAND*. Для первого примера, поскольку *a*, *b*, *c* и *d* являются ключевыми словами, соответствующий тип *C* — *BOOL*. Объявления в функции выполнения команды сгенерированного файла *C* следующие:

```
BOOL v1 = FALSE;
BOOL v2 = FALSE;
BOOL v3 = FALSE;
BOOL v4 = FALSE;
```

Для второго примера объявления в функции выполнения команды сгенерированного файла *C* выглядят следующим образом:

```
BOOL v2 = FALSE;
BOOL v4 = FALSE;
```

Для синтаксиса повторить (...) переменные будут сгенерированы автоматически и сопоставлены с командными словами повторения. Например:

```
COMMAND = a { {b|c|d} ... }
CMD_VAR = a
CMD_VAR = b
CMD_VAR = c
CMD_VAR = d
```

Команда эквивалентна *a {b/c/d} {b/c/d} {b/c/d}*. Для второго *{b/c/d}* три *C*-переменные *b\_1*, *c\_1*, и *d\_1*, автоматически генерируются механизмом *ICLI* для сопоставления. Для третьего *{b/c/d}* *C*-переменные *b\_2*, *c\_2* и *d\_2*, автоматически генерируются механизмом *ICLI* для сопоставления.

Через *CMD\_VAR* можно узнать входное значение этой команды от пользователя и использовать эти переменные *C* в подсегменте *CODE*.

- **RUNTIME =**

Необязательный тег определяет обратный вызов для проверки во время выполнения соответствующего слова командной строки *COMMAND*, то есть это однозначное сопоставление с каждым словом командной строки. Правило сопоставления идентично правилу для *CMD\_VAR*.

Чтобы вызвать обратный вызов, используется один из следующих способов:

- нажать клавишу TAB
- ввести ?

Обратный вызов запрашивает текущую проверку, справку и диапазон значений времени выполнения. Ниже приведен прототип обратного вызова:

```

/*
ICLI_ASK_PRESENT      : ask if the word is present or not
ICLI_ASK_HELP        : ask help string
ICLI_ASK_RANGE       : ask integer range for signed or unsigned,
                      this works on variables for all signed and
                      unsigned integer or integer list
ICLI_ASK_PORT_RANGE  : ask port type and list for the port range,
                      this works on <port_type_id>, <port_type_list>,
                      <port_type>, <port_id>, <port_list>
ICLI_ASK_CWORD       : ask all possible customized words for <cword>
                      use 'NULL' for the end
ICLI_ASK_VCAP_VR    : ask range for vcap_vr
*/
typedef enum {
    ICLI_ASK_PRESENT,
    ICLI_ASK_HELP,
    ICLI_ASK_RANGE,
    ICLI_ASK_PORT_RANGE,
    ICLI_ASK_CWORD,
    ICLI_ASK_VCAP_VR,
    ICLI_ASK_URL,
    ICLI_ASK_NET_ADDR,
} icli_runtime_ask_t;

typedef union {
    BOOL                present;
    char                help[ICLI_HELP_MAX_LEN + 4];
    icli_range_t        range;
    icli_stack_port_range_t port_range;
    char                *cword[ICLI_CWORD_MAX_CNT];
    icli_ask_vcap_vr_t  vcap_vr;
    BOOL                url_not_support[ICLI_URL_PROTOCOL_MAX];
    BOOL                net_addr_not_support[ICLI_NET_ADDR_TYPE_MAX];
} icli_runtime_t;

/*

```

## INPUT

```

session_id      : session ID
ask             : what is asked at runtime

```

## OUTPUT

## Runtime

```

ICLI_ASK_PRESENT      : runtime.present
ICLI_ASK_HELP         : runtime.help
ICLI_ASK_RANGE        : runtime.range
ICLI_ASK_PORT_RANGE   : runtime.port_range
ICLI_ASK_CWORD        : runtime.cword
ICLI_ASK_VCAP_VR      : runtime.vcap_vr

```

## RETURN

## TRUE

ICLI engine will check the output value in runtime.

```

ICLI_ASK_PRESENT      : runtime.present == TRUE, enable the word
                       runtime.present == FALSE, disable the word

```

```

ICLI_ASK_HELP         : use runtime.help
ICLI_ASK_RANGE        : use runtime.range
ICLI_ASK_PORT_RANGE   : use runtime.port_range
ICLI_ASK_CWORD        : use runtime.cword
ICLI_ASK_VCAP_VR      : use runtime.vcap_vr

```

## FALSE

ICLI engine will ignore the output value in runtime.

```

ICLI_ASK_PRESENT      : the word is present
ICLI_ASK_HELP         : use original one in *.icli
ICLI_ASK_RANGE        : use original one in *.icli
ICLI_ASK_PORT_RANGE   : use system port range
ICLI_ASK_CWORD        : <cword> works as <word>
ICLI_ASK_VCAP_VR      : no range limit

```

```

*/

```

```

typedef BOOL (icli_runtime_cb_t)(
    IN u32                session_id,
    IN icli_runtime_ask_t ask,
    OUT icli_runtime_t    *runtime
);

```

Проверка используется во время выполнения в следующих случаях:

- Слово в команде включается или отключается во время выполнения. Например, соответствующий компонент включен или отключен.
- Диапазон значений изменяется во время выполнения.

Для каждого типа запроса он может работать только с определенными типами слов:

- *ICLI\_ASK\_PRESENT* — для ключевого слова и всех типов переменных.
- *ICLI\_ASK\_HELP* — для ключевого слова и всех типов переменных.
- *ICLI\_ASK\_VALUE* — только для типов переменных *<range\_lsit>*, *<int>*, *<uint>*, *<word>*, *<keyword>*, *<string>*, и *<line>*
- *ICLI\_ASK\_PORT\_RANGE* — для *<port\_type\_id>* и *<port\_type\_list>*.

- *ICLI\_ASK\_CWORD* — только для *<cword>*.
- *ICLI\_ASK\_VCAP\_VR* — только для *<vcap\_vr>*.

Например, протокол RSTP может быть включен или отключен во время выполнения, по этой причине для слова *rstp* предусмотрена проверка во время выполнения:

```

FUNCTION_BEGIN
static BOOL _rstp_runtime(
    IN u32 session_id,
    IN icli_runtime_ask_t ask,
    OUT icli_runtime_t *runtime
)
{
    switch ( ask ) {
    case ICLI_ASK_PRESENT:
        if ( rstp_enable() ) {
            runtime.present = TRUE;
        } else {
            runtime.present = FALSE;
        }
        return TRUE;
    default:
        break;
    }
    return FALSE;
}
FUNCTION_END
CMD_BEGIN
COMMAND = stp mode [ mstp | rstp | mrstp ]
...
RUNTIME =
RUNTIME =
RUNTIME =
RUNTIME = _rstp_runtime
RUNTIME =
...
CMD_END

```

Используя *ICLI\_ASK\_PRESENT* в первом ключевом слове команды, данная команда может быть *enabled+visible* и *disabled+invisible* во время выполнения. В следующем примере показано, что команда включена и видна, когда включен режим STP. В противном случае команда отключена и невидима.

```

FUNCTION_BEGIN
static BOOL _stp_runtime(
    IN u32 session_id,
    IN icli_runtime_ask_t ask,
    OUT icli_runtime_t *runtime
)

```

```

{
  switch ( ask ) {
  case ICLI_ASK_PRESENT:
    if ( stp_enable() ) {
      runtime.present = TRUE;
    } else {
      runtime.present = FALSE;
    }
    return TRUE;
  default:
    break;
  }
  return FALSE;
}
FUNCTION_END
CMD_BEGIN
COMMAND = stp mode [ mstp | rstp | mrstp ]
...
RUNTIME = _stp_runtime
RUNTIME =
RUNTIME =
RUNTIME =
RUNTIME =
...
CMD_END

```

- **HELP =**

Данный необязательный тег определяет строку справки для соответствующего слова командной строки. Строка справки появляется, когда пользователь нажимает ? чтобы получить полные описания следующих возможных командных слов.

Если строка справки длинная, следует использовать \ для объединения строк. Чтобы получить строку справки во время выполнения, необходимо выполнить команду *RUNTIME*.

### Пример

```

COMMAND = temperature { get | set <0-100> }
...
HELP =
HELP = get the current temperature in Celsius degree
HELP =
HELP = the range of degree is from 0 to 100
...

```

Ниже приведены пользовательские данные:

- *switch> temperature ?*
- *get*— получить текущую температуру в градусах Цельсия.
- *set*— установка текущей температуры в градусах Цельсия.
- *switch> temperature set ?*

- *<0-100>* диапазон от 0 до 100.
- *set* не имеет строки справки для отображения, потому что значение тега соответствующей *HELP* пусто

- **MODE\_VAR =**

Данный дополнительный тег определяет переменную *C* для переменной в команде входа в режим. Это не обязательно должно быть однозначное сопоставление *CMD\_VAR* с *COMMAND*, но механизм *ICLI* помогает искать переменную в команде входа в режим и сопоставлять ее. Переменная *C* автоматически объявляется в функции выполнения команды сгенерированного файла *C* в соответствии с типом переменной в команде входа в режим.

```
MODE_VAR = vid
```

Например механизм *ICLI* автоматически находит *vid* для *<1-4094>*. Таким образом, *VLAN ID* можно получить через переменную *C vid*. В этом примере используется *vid* для установки IP-адреса на правильном интерфейсе *VLAN*.

- **VARIABLE\_BEGIN/VARIABLE\_END**

Данные необязательные теги отмечают начало и конец подсегмента переменной. Подсегмент объявляет переменные, которые используются в следующем подсегменте *CODE*, и здесь можно инициализировать переменные *CMD\_VAR* и *MODE\_VAR*.

- **CODE\_BEGIN/CODE\_END**

Данные необязательные теги отмечают начало и конец подсегмента *CODE*. Данный подсегмент содержит тело кода, реализующее функцию команды. Существует переменная *C*, *session\_id*, которую можно использовать в этом сегменте. *session\_id* является входным параметром функции выполнения команды и идентифицирует текущую сессию *ICLI*. С другой стороны, *session\_id* необходим для использования *API*, экспортированных в *icli\_api.h*, поскольку это первый входной параметр *API* в *icli\_api.h*.

### Пример

```
icli_session_printf(session_id, "test");
```

- **NO\_FORM Tags**

Следующие теги являются необязательными. Использование данных тегов аналогично использованию: *DOC\_CMD\_DESC*, *DOC\_CMD\_DEFAULT*, *DOC\_CMD\_USAGE*, *DOC\_CMD\_EXAMPLE*, *VARIABLE\_BEGIN*, *VARIABLE\_END*, *CODE\_BEGIN* и *CODE\_END*.

- *NO\_FORM\_DOC\_CMD\_DESC*
- *NO\_FORM\_DOC\_CMD\_DEFAULT*
- *NO\_FORM\_DOC\_CMD\_USAGE*
- *NO\_FORM\_DOC\_CMD\_EXAMPLE*
- *NO\_FORM\_VARIABLE\_BEGIN*
- *NO\_FORM\_VARIABLE\_END*
- *NO\_FORM\_CODE\_BEGIN*
- *NO\_FORM\_CODE\_END*

Если в подсегменте *CODE* есть коды *NO\_FORM\_CODE\_BEGIN* и *NO\_FORM\_CODE\_END*, команда *no form* автоматически генерируется движком ICLI. В следующем примере механизм ICLI автоматически генерирует команду *no arp inspection*, а проверка дубликатов слов *arp inspection* используют те же самые свойства команды, которые были определены ранее:

```

CMD_BEGIN
COMMAND = arp inspection
...
HELP = ARP configuration
HELP = Arp Inspection configuration
...
VARIABLE_BEGIN
...
VARIABLE_END
CODE_BEGIN
/* enable ARP inspection */
...
CODE_END
NO_FORM_CODE_BEGIN
/* disable ARP inspection */
...
NO_FORM_CODE_END
CMD_END

```

#### • DEFAULT\_FORM Tags

Следующие теги являются необязательными. Использование этих тегов аналогично использованию: *DOC\_CMD\_DESC*, *DOC\_CMD\_DEFAULT*, *DOC\_CMD\_USAGE*, *DOC\_CMD\_EXAMPLE*, *VARIABLE\_BEGIN*, *VARIABLE\_END*, *CODE\_BEGIN*, и *CODE\_END*.

- *DEFAULT\_FORM\_DOC\_CMD\_DESC*
- *DEFAULT\_FORM\_DOC\_CMD\_DEFAULT*
- *DEFAULT\_FORM\_DOC\_CMD\_USAGE*
- *DEFAULT\_FORM\_DOC\_CMD\_EXAMPLE*
- *DEFAULT\_FORM\_VARIABLE\_BEGIN*
- *DEFAULT\_FORM\_VARIABLE\_END*
- *DEFAULT\_FORM\_CODE\_BEGIN*
- *DEFAULT\_FORM\_CODE\_END*

Если и только если в подсегменте *CODE* есть коды *DEFAULT\_FORM\_CODE\_END*, механизм ICLI автоматически генерирует команду *no form*. В следующем примере *default arp inspection* автоматически генерируется механизмом ICLI, а проверка дубликатов слов использует те же свойства команды, которые были определены ранее:

```

CMD_BEGIN
COMMAND = arp inspection
...

```

```

HELP = ARP configuration
HELP = Arp Inspection configuration
...
VARIABLE_BEGIN
...
VARIABLE_END
CODE_BEGIN
/* enable ARP inspection */
...
CODE_END
DEFAULT_FORM_CODE_BEGIN
/* reset ARP inspection to be default*/
...
DEFAULT_FORM_CODE_END
CMD_END

```

При развертывании команды *no/default* следует убедиться, что синтаксис нормальной команды и команды *no/default* идентичен.

Например, обычная команда проверки ARP — это *arp inspection*, и эта же команда *no form* будет *no arp inspection*. За исключением *no*, у них одинаковый синтаксис, поэтому он может использовать *no form*.

Обычная команда IGMP — это *ip igmp {V1/V2/V3}*, а *no form* вариант этой команды — *no ip igmp*. Команда *no form* не имеет синтаксиса *{V1/V2/V3}*, их синтаксис отличается, поэтому *no form* нельзя применить.

### 3.3.5 Defining Constant String

Константа может быть определена во всех сегментах файла сценария, за исключением следующих областей:

- Сегмент Include
- Сегмент Function
- Variable initial sub-segment
- Code sub-segment

Данные области вставляются с кодами точно так же, как сгенерированные файлы С и Н, поэтому механизм ICLI ни о чем не заботится в этих областях.

Другими словами, если за пределами этих областей определено значение I, и этот тег не зарезервирован механизмом ICLI, то значение I принимается как определение постоянной строки, будь то в верхнем или нижнем регистре. Когда на него ссылаются, префикс *##* необходим, чтобы указать, что следующая строка является именем постоянной строки. Данная функция упрощает повторное использование строки, например строки *HELP*.

В следующих примерах *SHOW\_HELP\_str* определяется в средней и последней строке, но его можно использовать в командах:

```

SHOW_HELP_str = show system information
COMMAND = show ip interface
...

```

```

HELP = ##SHOW_HELP_str
HELP =
HELP =
...
COMMAND = show mac address
...
HELP = ##SHOW_HELP_str
HELP =
HELP =
...

```

Независимо от того, где в скрипте определена константная строка, ее допустимо использовать во всей области скрипта:

### Пример 1

```

COMMAND = show ip interface
...
HELP = ##SHOW_HELP_str
HELP =
HELP =
...
SHOW_HELP_str = show system information
COMMAND = show mac address
...
HELP = ##SHOW_HELP_str
HELP =
HELP =
...

```

### Пример 2

```

COMMAND = show ip interface
...
HELP = ##SHOW_HELP_str
HELP =
HELP =
...
COMMAND = show mac address
...
HELP = ##SHOW_HELP_str
HELP =
HELP =
...
SHOW_HELP_str = show system information

```

#### 3.3.6 Пример сегмента команды

В следующем примере символ «!» используется для строчного комментария:

```

! Начало сегмента команды
CMD_BEGIN
! Описание команды
DOC_CMD_DESC = this command is used to set management ip interface.
! значение по умолчанию
DOC_CMD_DEFAULT = the default IP is 192.168.0.1/255.255.255.0
! описание использования
DOC_CMD_USAGE = when the default IP does not work on your network, \
use this command to modify the IP for your network.
! пример
DOC_CMD_EXAMPLE = if you wants to set ip and netmask to be 10.1.1.1/255.0.0.0, \
you can execute the command as follows.
DOC_CMD_EXAMPLE = Switch# ip address 10.1.1.1 255.0.0.0
! Командная строка
COMMAND = ip address <ipv4_ucast> <ipv4_netmask>
! Уровень привилегий команды
PRIVILEGE = ICLI_PRIVILEGE_8
! Свойство команды
PROPERTY = ICLI_CMD_PROP_ENABLE | ICLI_CMD_PROP_VISIBLE
! Переменная C для "ip"
CMD_VAR =
! Переменная C для "address"
CMD_VAR =
! Переменная C для "<ipv4_ucast>"
CMD_VAR = ip
! Переменная C для "<ipv4_netmask>"
CMD_VAR = netmask
! Строка справки для "ip"
HELP = ip interface
! Строка справки для " address"
HELP = ip address set
! Строка справки для "<ipv4_ucast>"
HELP = unicast IP address
! Строка справки для "<ipv4_netmask>"
HELP = netmask
! Команда работает в режиме интерфейса VLAN.
CMD_MODE = ICLI_CMD_MODE_INTERFACE_VLAN
! Переменная C для "<1-4094>"
MODE_VAR = vid
! Объявление переменных и инициализация
VARIABLE_BEGIN
char ip_str[20];
char netmask_str[20];
VARIABLE_END
! Тело выполнения команды
CODE_BEGIN
//translate IP and netmask to string format
icli_ipv4_to_str(ip, ip_str);

```

```
icli_ipv4_to_str(netmask, netmask_str);
ICLI_PRINTF("Set %s/%s on VLAN %d successfully\ n",
ip_str, netmask_str, vid);
CODE_END
\
CMD_END
```

### 3.4 Компиляция: Make File

Для того, чтобы автоматически сгенерировать и зарегистрировать команды ICLI, необходимо скопировать следующие две строки в make-файл компонента, *module\_x.in*:

```
# Built-in ICLI
$(eval $(call add_icli,$(foreach m, x_0 x_1,$(DIR_x_script)/$(m).icli)))
```

где

- *x\_0* и *x\_1* относятся к файлам сценариев ICLI *x\_0.icli* и *x\_1.icli*;
- *DIR\_x\_script* — это каталог, в котором хранятся файлы *x\_0.icli* и *x\_1.icli*.

В таблице 2 перечислены доступные типы переменных.

Таблица 2 — Типы переменных

Тип переменной	Тип C	Описание	Ввод
<i>a-b,c,d-e</i>	<i>u32</i>	Целое число в диапазоне. Диапазон разделен символом , и каждый блок между ними может быть одним десятичным целым числом или значением диапазона. Значение диапазона использует - для указания диапазона. В этом случае <i>a</i> , <i>b</i> , <i>c</i> , <i>d</i> и <i>e</i> — десятичные целые числа, $a < b$ и $d < e$ . Максимальное количество блоков диапазона равно 8. Предполагается, что -5-3,20,-1-0,6,15-119,-4 имеет 6 блоков диапазона. Таким образом, допустимое входное значение должно быть либо $(\geq -5 \ \&\& \leq -3)$ , $(\geq -1 \ \&\& \leq 0)$ , либо $(=6)$ , либо $(\geq 15 \ \&\& \leq 119)$ . Кроме цифровых чисел, <i>a</i> и <i>b</i> также могут быть константами. Если это константа, то она должна быть заключена в одинарные кавычки; например, $\langle 1-'ACL\_MAX\_CNT' \rangle$ или $\langle 'MIN\_NUM'-'MAX\_NUM' \rangle$ .	Допустимый: -4 0 6 17 118 Недопустимый: -6 -2 3 11 120

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>a b</i>	Если с отрицательным значением <i>icli_signed_range_t</i> * Если без отрицательных значений <i>icli_unsigned_range_t</i> *	Список диапазонов в диапазоне <i>a</i> и <i>b</i> , где $a < b$ , и максимальное количество блоков диапазона в списке равно 8. Пример: <-9 90>. <i>a</i> и <i>b</i> могут быть цифровыми числами или константами. Если они константы, то их нужно заключить в одинарные кавычки; например <1 'ACL_MAX_CNT'>.	Допустимый: -5 0,0,0,0 -5-3,-1- 0,6,10-90 1,3,4,6,7,8,5,2H Недопустимый: -30 -5-3,-1- 0,6,10-99 - 10,2,3,4,5,6,7, 8,9
<i>clock_id</i>	<i>icli_clock_id_t</i>	ID представляет собой массив шестнадцатеричных значений из 8 байтов.	Допустимый: 00:01:02:03:04:05 :06:07 0-1-2-3-4b-5-6-7 0-001-2-3:4:5: 006:7a Недопустимый: 0:1:2:3:4:5:6: 0-111-2-3-4-5-6- 7 0-1-2-3:4:5-6.7
<i>cword</i>	<i>char</i> *	Слова-константы, определенные во время выполнения.	Допустимый: Слова определенные во время выполнения. Недопустимый: Неопределенные слова
<i>date</i>	<i>icli_date_t</i>	Дата в виде гггг/мм/дд, гггг=1970-2037, мм=1-12, дд=1-31	Допустимый: 1970/09/25 2011/05/26 2037/12/31 Недопустимый: 1969/09/25 2011/050/26 203a/12/31
<i>domain_name</i>	<i>char</i> *	Доменное имя, соответствующее RFC1123: <domain> ::= <subdomain> <subdomain> ::= <label>   <subdomain> ".«label» <label> ::= <let-dig> [   <ldh-str> ] <let-dig>   <ldh-str> ::= <let-dig-hyp>   <let-dig-hyp> <ldh-str> <let-dig-hyp> ::= <let-dig>   " "; <let-dig> ::= <letter>   <digit> <letter> ::= A-Za-z <digit> ::= 0-9	Допустимый: abc.1.com 1.2.3.4 1.2-3 Недопустимый: abc.1. 1.2.3.4. 1.2-3-

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
		Поддерживаются $\langle domain\_name \rangle$ и $\langle domain\_name-m \rangle$ , где $m$ — максимальная длина слова, а $n$ — минимальная длина.	
<i>dpl</i>	<i>u8</i>	Сбросить уровень приоритета. Диапазон 0–3.	Допустимый: 0 1 2 3 Недопустимый: -0 -1 4 5
<i>dscp</i>	<i>u8</i>	Предоставляет определенные DSCP РНВ. Допустимые РНВ: af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, cs1, cs2, cs3, cs4, cs5, cs6, cs7, ef, va.	Допустимый: b be af13 cs6 e ef Недопустимый: be1 af1 c cs vb
<i>dword</i>	<i>char *</i>	Одно слово, в котором все символы представлены цифрами от 0 до 9. Поддерживаются $\langle dword \rangle$ и $\langle dwordn-m \rangle$ , где $m$ — максимальная длина слова, а $n$ — минимальная длина.	Допустимый: 0203 1235 10000 Недопустимый: f12345 -123 _iso34
<i>dwordn-m</i>	<i>char *</i>	То же, что и $\langle dword \rangle$ , но длина строки должна быть $\geq n$ и $\leq m$ . Например, $\langle dword4-10 \rangle$ .	Допустимый: 1234 00123 1234567890 Недопустимый: 123 fe80 12.34
<i>dwordm</i>	<i>char *</i>	То же, что и $\langle dword \rangle$ , но длина строки должна быть $\leq m$ . Например, $\langle dword5 \rangle$ .	Допустимый: 12345 0123 0 Недопустимый: 123456 fe80 -123

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>fword</i>	<i>char *</i>	Одно слово с плавающей запятой. Поддерживаются <i>&lt;fwordm&gt;</i> и <i>&lt;fwordm.n&gt;</i> , где <i>m</i> — максимальное количество цифр до точки, а <i>n</i> — максимальное количество цифр после точки.	Допустимый: 0.203 12.35 1000.0 Недопустимый: f2345 -123 100.00.0
<i>fwordm</i>	<i>char *</i>	То же, что и <i>&lt;fword&gt;</i> , но количество цифр перед точкой должно быть $\leq m$ . Например, <i>&lt;fword5&gt;</i> .	Допустимый: 12345.067456 0.7368 125.87 Недопустимый: 123456.123 fe80.12 12.34.56
<i>fwordm.n</i>	<i>char *</i>	То же, что и <i>&lt;fword&gt;</i> , но количество цифр до точки должно быть $\leq m$ , а количество цифр после точки должно быть $\leq n$ . Например, <i>&lt;fword4.5&gt;</i> .	Допустимый: 1234.00001 0.1 123.123 Недопустимый: 123.987654 123f.12345 12345.123
<i>hexval</i>	<i>icli_hexval_t</i>	Шестнадцатеричное значение начинается с 0x или 0X, а его максимальная длина составляет 128 байт. Поддерживаются <i>&lt;hexvalm&gt;</i> и <i>&lt;hexvaln-m&gt;</i> , где <i>m</i> — максимальная длина значения, а <i>n</i> — минимальная длина.	Допустимый: 0x012345678 0X567890abcdef Недопустимый: f2345 -123 100.00.0
<i>hexvaln-m</i>	<i>icli_hexval_t</i>	То же, что и <i>&lt;hexval&gt;</i> , но его длина должна соответствовать $\geq n$ и $\leq m$ . Например, <i>&lt;hexval2-3&gt;</i> .	Допустимый: 0x010 0x01020 0x010203 Недопустимый: 0x01ABCDEF 0xFF 0180c2
<i>ahexvalm</i>	<i>icli_hexval_t</i>	То же, что и <i>&lt;hexval&gt;</i> , но его длина должна соответствовать $\leq m$ . Например, <i>&lt;hexval3&gt;</i> .	Допустимый: 0x010203 0x01 0x01020 Недопустимый: 0x01ABCDEF 0x0180g2 0180c2
<i>hhmm</i>	<i>icli_time_t</i>	Время в чч:мм, где чч в диапазоне от 0 до 23, а мм в диапазоне от 0 до 59.	Пример: 00:00 05:05 23:15 Недопустимый: :00:00 05:05: 23::15 24:15:59

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>hostname</i>	<i>char *</i>	<p>&lt;host_name&gt; ::= &lt;let-dig&gt; [ [ &lt;ldh-str&gt; ] &lt;let-dig&gt; ]</p> <p>&lt;ldh-str&gt; ::= &lt;let-dig-hyp&gt;   &lt;let-dig-hyp&gt; &lt;ldh-str&gt;</p> <p>&lt;let-dig-hyp&gt; ::= &lt;let-dig&gt;   ”.”</p> <p>&lt;let-dig&gt; ::= &lt;letter&gt;   &lt;digit&gt;</p> <p>&lt;letter&gt; ::= A-Za-z</p> <p>&lt;digit&gt; ::= 0-9</p> <p>Максимальная длина по умолчанию составляет 63 байта. Поддерживаются &lt;host_namet&gt; и &lt;host_namen-m&gt;, где <i>m</i> — максимальная длина слова, а <i>n</i> — минимальная длина.</p>	<p>Допустимый:</p> <p>Abc0998</p> <p>9</p> <p>12-34</p> <p>67-gh</p> <p>Недопустимый:</p> <p>8@</p> <p>2.4</p> <p>6-8.com</p>
<i>int</i>	<i>i32</i>	32-битное целое число со знаком. Диапазон: -2147483648...2147483647.	<p>Допустимый:</p> <p>-100</p> <p>-0</p> <p>1234567890</p> <p>Недопустимый:</p> <p>-2147483649</p> <p>2147abc</p> <p>21474836470</p>
<i>int16</i>	<i>i16</i>	16-битное целое число со знаком. Диапазон составляет -32768...32767.	<p>Допустимый:</p> <p>-100</p> <p>-0</p> <p>23456</p> <p>Недопустимый:</p> <p>-2147449</p> <p>7abc</p> <p>14748364</p>
<i>int8</i>	<i>i8</i>	8-битное целое число со знаком. Диапазон составляет -128...127.	<p>Допустимый:</p> <p>-100</p> <p>-0</p> <p>123</p> <p>Недопустимый:</p> <p>-83649</p> <p>21bc</p> <p>128</p>
<i>ipv4_abc</i>	<i>b4cd_ip_t</i>	IPv4 IP-адрес в формате ddd.ddd.ddd.ddd, где d — десятичная цифра, а адрес должен относиться к классу A, B или C. Если он относится к классу D или E, то он недействителен.	<p>Допустимый:</p> <p>1.0.0.0</p> <p>10.254.255.0</p> <p>223.255.255.255</p> <p>Недопустимый:</p> <p>224.0.0.0</p> <p>240.255.255.254</p> <p>255.0.0.1</p>
<i>ipv4_addr</i>	<i>b4cd_ip_t</i>	Любой IP-адрес IPv4 в формате ddd.ddd.ddd.ddd, где d — десятичная цифра, а диапазон каждого ddd — от 0 до 255.	<p>Допустимый:</p> <p>0.0.0.0</p> <p>1.2.3.4</p> <p>255.255.255.255</p> <p>Недопустимый:</p> <p>0.0.0</p> <p>0.0.0.256</p> <p>1.2.3:4</p>

Таблица 2 — Продолжение

Тип переменной	Тип С	Описание	Ввод
<i>ipv4_mcast</i>	<i>b4cd_ip_t</i>	Многоадресный IP-адрес IPv4 в формате ddd.ddd.ddd.ddd, где d — десятичная цифра. Он находится в диапазоне от 224.0.0.0 до 239.255.255.255.	Допустимый: 224.0.0.0 230.254.255.0 239.255.255.255 Недопустимый: 0.0.1.2 223.255.255.254 240.0.0.1
<i>ipv4_netmask</i>	<i>b4cd_ip_t</i>	Сетевая маска IPv4 в формате ddd.ddd.ddd.ddd, где d — десятичная цифра.	Допустимый: 0.0.0.0 255.255.255.0 255.252.0.0 Недопустимый: 0.0.255.255 255.250.255.0 255.252.0.1
<i>ipv4_nmcast</i>	<i>b4cd_ip_t</i>	Немногоадресный IP-адрес IPv4 в формате ddd.ddd.ddd.ddd, где d — десятичная цифра, а IP-адрес — это адрес, не входящий в диапазон от 224.0.0.0 до 239.255.255.255.	Допустимый: 0.0.1.255 223.255.255.254 240.0.0.0 Недопустимый: 224.0.0.0 235.0.1.255 239.255.255.255
<i>ipv4_prefix</i>	<i>u32</i>	Длина префикса IPv4 в формате /n, когда n находится в диапазоне от 0 до 32.	Допустимый: /0 /32 /0010 Недопустимый: /-1 /33 /00101
<i>ipv4_subnet</i>	<i>icli_ipv4_subnet_t</i>	Адрес подсети IPv4 в 2 форматах, где d — десятичная цифра: -IP/сетевая маска- ddd.ddd.ddd.ddd/ddd.ddd.ddd.ddd -IP/длина префикса- ddd.ddd.ddd.ddd/dd, где IP — это IP-адрес индивидуальной рассылки, а длина префикса dd (последний dd) — от 0 до 32.	Допустимый: 10.1.1.1/ 255.0.0.0 10.1.1.1/0 10.1.1.1/32 223.255.255.254 /8 Недопустимый: 10.1.1.1/255.0.1.0 10.1.1.1/-1 10.1.1.1/33 224.1.1.1/8
<i>ipv4_ucast</i>	<i>b4cd_ip_t</i>	Одноадресный IP-адрес IPv4 в формате ddd.ddd.ddd.ddd, где d — десятичная цифра. Следующие 3 случая являются недопустимыми: 1. В диапазоне от 224.0.0.0 до 239.255.255.255 2. ddd.ddd.ddd.0 3. ddd.ddd.ddd.255	Допустимый: 0.0.1.2 223.255.255.254 240.0.0.1 Недопустимый: 0.0.1.255 224.0.0.1 240.0.0.0

Таблица 2 — Продолжение

Тип переменной	Тип С	Описание	Ввод
<i>ipv6_addr</i>	<i>b4cd_ipv6_t</i>	Любой IP-адрес IPv6 в формате hhhh:hhh:hhh:hhh:hhh:hhh:hhh:hhh, где h — шестнадцатеричная цифра. :: допускается использовать для пропуска некоторых шестнадцатеричных цифр, но используется только один раз.	Допустимый: :: ::1 0:0:0:0:0:0:0:0 1234::5678:90ab 1234:: ::5678:90ab Недопустимый: : :::1 0:0:0:0:0:0:0:0 1234::5678::90ab 1234:::5678:90ab 1234::5678:90ab:
<i>ipv6_mcast</i>	<i>b4cd_ipv6_t</i>	Многоадресный IP-адрес IPv6 в формате hhhh:hhh:hhh:hhh:hhh:hhh:hhh:hhh, где h — шестнадцатеричная цифра. :: допускается использовать для пропуска некоторых шестнадцатеричных цифр, но используется только один раз. (Первый hh должен быть 0xFF.)	Допустимый: FF00:: FF55::1 FF34::cd:82 FF81:5678::ab Недопустимый: :: FF0:: 3F55::1 1234::cd:90ab ::5678:90ab
<i>ipv6_netmask</i>	<i>b4cd_ipv6_t</i>	Сетевая маска IPv6 в формате hhhh:hhh:hhh:hhh:hhh:hhh:hhh:hhh, где h — шестнадцатеричная цифра. :: допускается использовать для пропуска некоторых шестнадцатеричных цифр, но используется только один раз.	Допустимый: FF00:: FFF0:: FFFF:FF00:: FFFF:FFFF:FC00 :: Недопустимый: ::FF00 FF10:: FFFF::FF00:: :FFFF:FFFF:FC00 ::
<i>Ipv6_prefix</i>	<i>u32</i>	Длина префикса IPv6 в формате /n, где nn находится в диапазоне от 0 до 128.	Допустимый: /0 /128 /0000101 Недопустимый: /-1 /129 /0001011

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>ipv6_subnet</i>	<i>icli_ipv6_subnet_t</i>	Адрес подсети IPv6, IP/маска сети или биты IP/маски.	Допустимый: ::/FF00:: ::1/FFFF:FF00: 0:0:0:0:0:0: 0/33 1234::5678:90ab /1 1234::/128 ::5678:90ab/F00 0:: Недопустимый: ::/FF00:: ::1/FFFF:FF00: 0:0:0:0:0:0:0 /133 1234::5678:90ab /200 1234::/228 ::5678:90ab/F00 0:F:
<i>ipv6_ucast</i>	<i>b4cd_ipv6_t</i>	Одноадресный IP-адрес IPv6 в формате hhhh:hhh:hhh:hhh:hhh:hhh:hhh:hhh:hhh, где h — шестнадцатеричная цифра. :: допускается использовать для пропуска некоторых шестнадцатеричных цифр, но используется только один раз. (Первый hh не должен быть 0xFF.)	Допустимый: :: FF00:: 3F55::1 1234::cd:90ab ::5678:90ab Недопустимый: FF00:: FF55::1 FF34::cd:82 FF81:5678::ab
<i>keyword</i>	<i>char *</i>	Одно слово, но должно начинаться с буквы (A-Z или a-z, а не числовое значение). Поддерживаются <i>&lt;keywordm&gt;</i> и <i>&lt;keywordn-m&gt;</i> , где <i>m</i> — максимальная длина слова, а <i>n</i> — минимальная длина.	Допустимый: ab2c3 f123 iso3456 Недопустимый: 12345 f 123 _iso34
<i>keywordn-m</i>	<i>char *</i>	То же, что и <i>&lt;keyword&gt;</i> , но длина строки должна быть $\geq n$ и $\leq m$ . Например, <i>&lt;keyword4-10&gt;</i> .	Допустимый: ab2c3 F123 iso34 Недопустимый: 1abc2 ab1 _ab1

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>keywordm</i>	<i>char *</i>	То же, что и <i>&lt;keyword&gt;</i> , но длина строки должна быть $\leq m$ . Например, <i>&lt;keyword5&gt;</i> .	Допустимый: ab2c3 f123 iso34 Недопустимый: 1abc2 abcde1 a bc2
<i>line</i>	<i>char *</i>	Строка может содержать несколько слов, разделенных пробелами. Поддерживаются <i>&lt;linem&gt;</i> и <i>&lt;linen-m&gt;</i> , где $m$ — максимальная длина слова, а $n$ — минимальная длина.	Допустимый: This is a book "This is a book" 123 45 "123 45" 12345 "12345" a 123 "a 123" Недопустимый: <enter>
<i>linen-m</i>	<i>char *</i>	То же, что и <i>&lt;line&gt;</i> , но длина строки должна быть $\geq n$ и $\leq m$ . Например, <i>&lt;line4-10&gt;</i> .	Допустимый: 12 3 123 45 "123 45" open book "open book" Недопустимый: <enter> "1 2 3 4 5" dog
<i>linem</i>	<i>char *</i>	То же, что и <i>&lt;line&gt;</i> , но длина строки должна быть $\leq m$ . Например, <i>&lt;line5&gt;</i> .	Допустимый: 123 5 "123 5" 12345 "12345" A dog "A dog" Недопустимый: <enter> "12345" "A dog"
<i>mac_addr</i>	<i>b4cd_mac_t</i>	Любой MAC-адрес Ethernet hh:hh:hh:hh:hh:hh, где h — шестнадцатеричная цифра.	Допустимый: 00:11:2a:bc:de:f9 0:1:0:46:5:3a Недопустимый: 00:11:2a:bc 0.1.0.46.5.3a
<i>mac_mcast</i>	<i>b4cd_mac_t</i>	Многоадресный MAC-адрес hh:hh:hh:hh:hh:hh, где первые два hh и 0x01 == 0x01	Допустимый: 03:11:2a:bc:45:fd b:1:0:46:5:3a Недопустимый: 02:11:2a:bc:de:f9 0:1:0:46:5:3a

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>mac_ucast</i>	<i>b4cd_mac_t</i>	Одноадресный MAC-адрес hh:hh:hh:hh:hh:hh, где первые два hh и 0x01 != 0x01	Допустимый: 02:11:2a:bc:de:f9 0:1:0:46:5:3a Недопустимый: 03:11:2a:bc:45:fd b:1:0:46:5:3a
<i>net_addr</i>	<i>icli_net_addr_t</i>	Тип переменной <i>&lt;net_addr&gt;</i> , который принимает имя хоста (если включена функция DNS), IPv4-адрес или IPv6-адрес (если включена функция IPv6).	Допустимый: 1.0.0.0 10.254.255.0 223.255.255.255 : ::1 0:0:0:0:0:0:0 1234::5678:90ab 1234:: ::5678:90ab Abc0998 9 12-34 67-gh Недопустимый: 0.0.0 0.0.0.256 1.2.3:4 : :::1 0:0:0:0:0:0:0 :0 1234::5678::90ab 1234:::5678:90ab 1234::5678:90ab: 8@ 2.4 6-8.com
<i>oui</i>	<i>icli_oui_t</i>	Уникальный идентификатор организации (OUI) — это 24-битное число, представляющее собой первые три байта MAC-адреса. (Должно быть одноадресным.)	Допустимый: 00-01-02 14:05:07 ba:09:fe Недопустимый: 01-01-02 14:05:07: Ba::fe
<i>pcp</i>	<i>icli_unsigned_range_t</i> *	Точка кода приоритета (PCP). Допустимые входные данные: конкретные (0, 1, 2, 3, 4, 5, 6, 7), диапазон (0–1, 2–3, 4–5, 6–7, 0–3, 4–7 или любой(0-7)).	Допустимый: 5 2-3 6-7 0-7 Недопустимый: 8 2-4 6-8 5-7

Таблица 2 — Продолжение

Тип переменной	Тип С	Описание	Ввод
<i>port_id</i>	<i>icli_switch_port_range_t</i>	Номер порта в формате switch_id/ID. ID (число) начинается с 1. Допустимый ввод зависит от настроек порта во время выполнения. Например, целевая платформа имеет 24 порта.	Допустимый: 1/1 1/15 Недопустимый: 1/0 1/25
<i>port_list</i>	<i>icli_stack_port_range_t*</i>	Список портов в формате списка switch_id/ID-list. Список разделен знаком ; Допустимый ввод зависит от настроек порта во время выполнения. Идентификационный номер начинается с 1.	Допустимый: 2/13 2/13,13; 1/1,3-6,23 1,2/1,3-6,23 1/4,6;2/9,17-23 Недопустимый: 3/1-24 1/0-4;2/1,3-25
<i>port_type</i>	<i>icli_port_type_t</i>	Необходимо указать типы портов в устройстве. Поддерживается пять типов портов: FastEthernet, Gigabit Ethernet (GbE), 2,5 GbE, 5 GbE и 10 GbE. Это зависит от настройки порта во время выполнения. Настройки порта можно изменить с помощью <i>icli_port_range_set()</i> . Например, что на устройстве есть порты 1 GbE и 10 GbE, но нет портов FastEthernet.	Допустимый: GbE 10 GbE Недопустимый: 2.5 GbE 5gi
<i>port_type_id</i>	<i>icli_switch_port_range_t</i>	Формат: port_type switch_id/port_id.	Допустимый: Gi 1/1 Fast 1/15 10gi 2/23 Недопустимый: gi 1/0 gi1/1,2 gi 1/2-5 tengi 2/23
<i>port_type_list</i>	<i>icli_stack_port_range_t*</i>	Формат: port_type switch_id/port_list.	Допустимый: gi 2/13 gi 2/13,13 10gi 1/1 fast 1/3-6,9,7 gi 1/5 2.5G 1/5 fast 1/7 gi 1/3 5G 1/4,6;2/ 9,17-23 gi 1/1,3,5 fast 1/2,4-8; 3/5,7 Недопустимый: gi 0/25 fast 2/13, tengi 1,2/ gi 1,2,3/1,3-6,2 tengi 1/4,6,2;2/9,17- 23gi 1/2-5 tengi 2/23

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>range_list</i>	<i>icli_range_t*</i>	Список диапазонов, максимальное количество блоков диапазонов в списке равно 8. Система допускает только инкрементный ввод.	Допустимый: -5 -5-3,-1- 0,6,10-99 1,2,3,4,5,6,7,8 Недопустимый: -5-30 -5-3,-7,6-10 1,1,1 1,2,3,4,5,6,7,8,9
<i>string</i>	<i>char *</i>	Строка может содержать несколько слов, разделенных пробелами, должна быть заключена в двойные кавычки “”. Поддерживаются <i>&lt;stringm&gt;</i> и <i>&lt;stringn-m&gt;</i> , где <i>m</i> — максимальная длина слова, а <i>n</i> — минимальная длина.	Допустимый: “123 45” “12345” “a 123” Недопустимый: "123 45" 123 45" 123 45
<i>stringn-m</i>	<i>char *</i>	То же самое с <i>&lt;string&gt;</i> , но длина строки должна быть $\geq n$ и $\leq m$ . Например, <i>&lt;string4-10&gt;</i> .	Допустимый: “123 45” “12345” “a 123” Недопустимый: "12345abcdef" 123 45" 123 45
<i>stringm</i>	<i>char *</i>	То же, что и <i>&lt;string&gt;</i> , но длина строки должна быть $\leq m$ . Например, <i>&lt;string5&gt;</i> .	Допустимый: “123 5” “12345” “a ” Недопустимый: "a12345" a1234" a1234
<i>switch_id</i>	<i>u32</i>	Идентификатор целевой платформы, который будет проверяться в соответствии с текущей конфигурацией целевой платформы. Например, что имеются целевые платформы 1, 3 и 9.	Допустимый: 1 3 9 Недопустимый: 0 2 8
<i>switch_list</i>	<i>icli_unsigned_range_t</i>	Список идентификаторов коммутаторов, которые будут проверены в соответствии с текущей конфигурацией коммутатора. Например, что имеются коммутаторы 1, 2, 3, 5, 8 и 9.	Допустимый: 2 2-3,9,8 1-3,5,5,8-9 Недопустимый: 4 2-5,8-9 1-3,5-9

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>time</i>	<i>icli_time_t</i>	Время в чч:мм:сс, чч=0...23, мм=0...59, сс=0-59.	Допустимый: 00:00:00 05:05:05 23:15:59 Недопустимый: :00:00 05:0a:05 23:15:59: 24:15:59
<i>uint</i>	<i>u32</i>	32-битное целое число без знака в диапазоне от 0 до 4294967295.	Допустимый: 0 100 4294967295 Недопустимый: -0 -1 10с 4294967296
<i>uint16</i>	<i>u16</i>	16-битное целое число без знака в диапазоне от 0 до 65535.	Допустимый: 0 100 29496 Недопустимый: -0 -1 10с 429496
<i>uint8</i>	<i>u8</i>	8-битное целое число без знака в диапазоне от 0 до 255.	Допустимый: 0 100 254 Недопустимый: -0 -1 10с 294
<i>url</i>	<i>char *</i>	<p>Унифицированный указатель ресурсов (URL) с именем файла. Это определенная строка символов, представляющая собой ссылку на ресурс. Если следующие специальные символы: пробел ! “ # \$ % &amp; ' ( ) * + , / : ; &lt; = &gt; ? @ [ ] ^ ` {   } должны содержаться во входной строке URL, они должны иметь процентное кодирование.</p> <p>Синтаксис:  <code>&lt;protocol&gt;://[&lt;username&gt;[:&lt;password&gt;]@]/&lt;host&gt;[:&lt;port&gt;][//&lt;path&gt;]</code></p> <p>Требования:  <code>&lt;protocol&gt;</code> — схема универсального идентификатора ресурса (URI). Во входной строке разрешены только строчные буквы, а ее максимальная длина составляет 31. Она должна учитывать tftp/ftp/http/https/file.</p>	Допустимый: http://user@abc.com tftp://1.2.3.4:123 ftp://b4cd.io:123/path https://user:passwd@host.com.tw:123/path file:///folder/path Недопустимый: httpx://user@abc.com tftp://1.2.3.4:123 ftp://b4cd.io:123path https://user:passwd@host.com.tw:123/path//

Таблица 2 — Продолжение

Тип переменной	Тип С	Описание	Ввод
		<p><i>&lt;username&gt;</i> — информация о пользователе (необязательно). Максимальная длина 63.</p> <p><i>&lt;пароль&gt;</i> — информация о пользователе (необязательно). Максимальная длина 63.</p> <p><i>&lt;host&gt;</i> — доменное имя или адрес IPv4. Максимальная длина 63.</p> <p><i>&lt;port&gt;</i> — номер порта (необязательно).</p> <p><i>&lt;path&gt;</i> — если указан путь, он должен быть разделен косой чертой (/). Максимальная длина 255.</p>	
<i>url_file</i>	<i>char *</i>	<p>Унифицированный указатель ресурсов (URL) с именем файла. Это определенная строка символов, которая представляет собой ссылку на ресурс. Если следующие специальные символы: пробел ! “ # \$ % &amp; ' ( ) * + , / : ; &lt; = &gt; ? @ [ ] ^ _ {   } должны содержаться во входной строке URL, они должны иметь процентное кодирование.</p> <p>Синтаксис:  <i>&lt;protocol&gt;</i>://[<i>&lt;username&gt;</i>[:<i>&lt;password&gt;</i>]/@]/<i>&lt;host&gt;</i>[:<i>&lt;port&gt;</i>][/<i>&lt;path&gt;</i>]/<i>&lt;file_name&gt;</i></p> <p>Требования:  <i>&lt;protocol&gt;</i> — схема универсального идентификатора ресурса (URI). Во входной строке разрешены только строчные буквы, а ее максимальная длина составляет 31. Она должна учитывать tftp/ftp/http/https/file.</p> <p><i>&lt;username&gt;</i> — информация о пользователе (необязательно). Максимальная длина 63.</p> <p><i>&lt;пароль&gt;</i> — информация о пользователе (необязательно). Максимальная длина 63.</p> <p><i>&lt;host&gt;</i> — доменное имя или адрес IPv4. Максимальная длина 63.</p> <p><i>&lt;port&gt;</i> — номер порта (необязательно).</p> <p><i>&lt;path&gt;</i> — если указан путь, он должен быть разделен косой чертой (/). Максимальная длина 255.</p> <p><i>&lt;file_name&gt;</i>: допустимое имя файла представляет собой текстовую строку, составленную из алфавита (A-Za-z), цифр (0-9), точки (.), дефиса (-), подчеркивания (_). Максимальная длина — 63 символа, дефис не должен быть первым символом. Имя файла, состоящее из символа точки не допускается.</p>	<p>Допустимый:  ftp://b4cd.io:123/path/file_name  https://user:passwd@host.com.tw:123/path/file_name  file:///folder/path</p> <p>Недопустимый:  httpx://user@abc.com  tftp://1.2.3.4:123  ftp://b4cd.io:123path  https://user:passwd@host.com.tw:123/path//</p>
<i>vlan_id</i>	<i>u32</i>	Идентификатор VLAN, целое число без знака в диапазоне от <i>min_vlan_id</i> до <i>max_vlan_id</i> , где <i>min_vlan_id</i> и <i>max_vlan_id</i> настраиваются во время выполнения.	<p>Допустимый:  1  9  24  4094</p>

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
		По умолчанию допустимым диапазоном является от 1 до 4095.	Недопустимый: -5 0 4095 10000
<i>vlan_list</i>	<i>icli_unsigned_range_t</i> *	Список идентификаторов VLAN — это список диапазонов блоков, которые должны находиться в диапазоне от <i>min_vlan_id</i> до <i>max_vlan_id</i> . По умолчанию допустимым диапазоном является от 1 до 4095.	Допустимый: 1100 1,1,200,300,55 1,20-30,90-2300 50,109,1010-4000 Недопустимый: -1100 1,200,30 1,20-30,9-2300 50,19,1010-4000
<i>vword</i>	<i>char</i> *	Одно слово, которое допускает 0-9a-zA-Z, но не разрешает использовать слово состоящее только из цифр 0-9. Поддерживаются <i>&lt;vwordm&gt;</i> и <i>&lt;vwordn-m&gt;</i> , где <i>m</i> — максимальная длина слова, а <i>n</i> — минимальная длина.	Допустимый: A123 123a avcd Недопустимый: 123 -123 abcd+def
<i>vwordn-m</i>	<i>char</i> *	То же, что и <i>&lt;vword&gt;</i> , но длина строки должна быть $\geq n$ и $\leq m$ . Например, <i>&lt;vword2-4&gt;</i> .	Допустимый: vln9 v99 Недопустимый: vlan9 v@11 123
<i>vwordm</i>	<i>char</i> *	То же, что и <i>&lt;vword&gt;</i> , но длина строки должна быть $\leq m$ . Например, <i>&lt;vword4&gt;</i> .	Допустимый: A123 123a avcd Недопустимый: a@1 ab c hello
<i>word</i>	<i>char</i> *	Одно слово без пробела внутри. Поддерживаются <i>&lt;wordm&gt;</i> и <i>&lt;wordn-m&gt;</i> , где <i>m</i> — максимальная длина слова, а <i>n</i> — минимальная длина.	Допустимый: 12345 ab2c3 _iso134*( Недопустимый: 123 45 ab 2c3 _iso i34*(
<i>wordn-m</i>	<i>char</i> *	То же, что и <i>&lt;word&gt;</i> , но длина строки должна быть $\geq n$ и $\leq m$ . Например, <i>&lt;word4-10&gt;</i> .	Допустимый: 12345 sentence _iso134*( Недопустимый: 123 ab 2c3 _iso i34*(

Таблица 2 — Продолжение

Тип переменной	Тип C	Описание	Ввод
<i>wordm</i>	<i>char *</i>	То же, что и <i>&lt;word&gt;</i> , но длина строки должна быть $\leq m$ . Например, <i>&lt;word5&gt;</i> .	Допустимый: 12345 ab  — Недопустимый: abc123 ab c12 h e l l o

### 3.5 Руководство ICLI

В следующих разделах описываются рекомендации ICLI.

#### 3.5.1 Повторяющиеся слова

Движок ICLI анализирует слева направо. Синтаксический анализ движком ICLI не допускает перекрытия одних и тех же слов в обязательных `{}` или необязательных `()`.

##### Пример 1 (недопустимый)

```
snmp client version <uint>
snmp server | client address <ipv4_ucast>
```

Ключевое слово *client* недопустимо перекрывается, так как *client* является обязательным `{}` во второй команде.

##### Пример 2 (недопустимый)

```
snmp client version <uint>
snmp [client] address <ipv4_ucast>
```

Ключевое слово *client* по-прежнему недопустимо перекрывается, потому что *client* является необязательным `[]` во второй команде.

В следующих примерах показаны предыдущие команды, переработанные как допустимые:

##### Пример 1 (допустимый)

```
snmp client version <uint> | address <ipv4_ucast>
snmp server address <ipv4_ucast>
```

##### Пример 2 (допустимый)

```
snmp client version <uint> | address <ipv4_ucast>
snmp address <ipv4_ucast>
```

Общее правило состоит в том, чтобы использовать один и тот же плоский префикс как можно чаще. snmp-клиент — это тот же плоский префикс в примерах.

### 3.5.2 Множественное необязательное начало

Если команда имеет следующую структуру, то ключевое слово *x* является множественным необязательным началом:

- *a [[x]]*
- *a [ b | [x] ]*

Для решения проблемы необходимо добавить ключевое слово перед *[x]*.

#### Пример:

- *a [ k [x] ]*
- *a [ b | k [x] ]*

### 3.5.3 Уровни подрежима

Механизм ICL1 поддерживает только один уровень для подрежима, поэтому создание нового подрежима в рамках существующего подрежима является недопустимым.

#### Допустимый пример

```
# configure terminal          ==> [Перейти в режим конфигурации]
(config)# vlan 2              ==> [Перейти в режим config-vlan-sub-mode]
(config-vlan)#                ==> [Режим config-vlan-sub-mode]
(config-vlan)# exit           ==> [Выйти из режима config-vlan-sub-mode]
(config)# inter gig 1/1       ==> [Перейти в режим config-port-sub-mode]
(config-if)#                  ==> [Режим config-port-sub-mode]
```

#### Недопустимый пример

```
# TestSubMode                ==> [Перейти к тестовому подрежиму]
(provision-test-sub-mode)# SecondLayerSubMode
                             ==> [Перейти ко второму слою тестового подрежима]
                             ==> Недопустимо! Это приведёт к непредсказуемому результату.
(config)# TestSubMode
                             ==> [Перейти в режим конфигурации тестового подрежима]
(config-test-sub-mode)# SecondLayerSubMode
                             ==> [Перейти к режиму конфигурации во втором
слое тестового подрежима]
                             ==> Недопустимо! Это приведёт к непредсказуемому результату.
```

Чтобы решить эту проблему, разделите «Недопустимый пример» на два подрежима:

```
(config) # Test1SubMode      ==> [Вход в config-test1-sub-mode]
(config-test1-sub-mode) #exit ==> [Выход из config-test1-sub-mode]
(config) # Test2SubMode      ==> [Вход в config-test2-sub-mode]
(config-test2-sub-mode) #exit ==> [Выход из config-test2-sub-mode]
```

### 3.6 Клавиши быстрого доступа

В таблице 3 перечислены доступные сочетания клавиш.

Таблица 3 — Сочетания клавиш

Функция	Сочетание клавиш	Описание
Строка		
	\$ (символ прокрутки строки)	Если длина команды превышает ширину окна, то «\$» будет использоваться для прокрутки ввода команды, но не для отображения следующей строки.
Курсор		
	Стрелка Влево	Назад на один символ влево.
	Стрелка Вправо	Вперед на один символ вправо
	HOME	Перейти к началу строки.
	Ctrl-a	
	END	
	Ctrl-e	Перейти в конец строки.
Удалить		
	Ctrl-n	Текущую строку
	DEL	Символ перед курсором.
	Ctrl-d	
	Backspace	Символ слева от курсора.
	Ctrl-h	
	Ctrl-u	Все символы от курсора до начала строки.
	Ctrl-x	
	Ctrl-k	Все символы от курсора до конца строки.
	Ctrl-w	
		Слово от курсора до начала слова.
Отображение страницы		
	Spacebar	Следующая страница.
	Все остальные клавиши	
	Enter	Следующая строка.
	Ctrl-c	Выход из большого количества выходных данных.
	q	
	g	
История		
	Стрелка Вверх	Предыдущая строка.
	Ctrl-p	
	Стрелка Вниз	Следующая строка.
Контекстная справка		
	TAB	В слове: автодополнение слова. После слова: перечислить все возможные варианты. Двойное нажатие: список всех возможных вариантов.
	?	Первый «?» перечисляет все возможные варианты с описанием справки. Второй непосредственный «?» перечисляет все возможные полные команды.
	Ctrl-q	Показать полный синтаксис команды.
Командный режим		
	Ctrl-z	Вернуться в режим exes напрямую.

## 4 Настройка Web-интерфейса

В данном разделе содержится подробное описание того, как настроить Web-интерфейс. В следующих пунктах обсуждаются различные сценарии настройки.

### 4.1 Изменение названия продукта

Название продукта отображается в верхнем баннере страницы (рисунок 2). Для изменения названия продукта, необходимо отредактировать файл *build/make/config.mk*. Где *config.mk* — это ссылка на файл *.mk* в каталоге *build/configs* на основе конфигурации, выбранной для этой конкретной платформы.

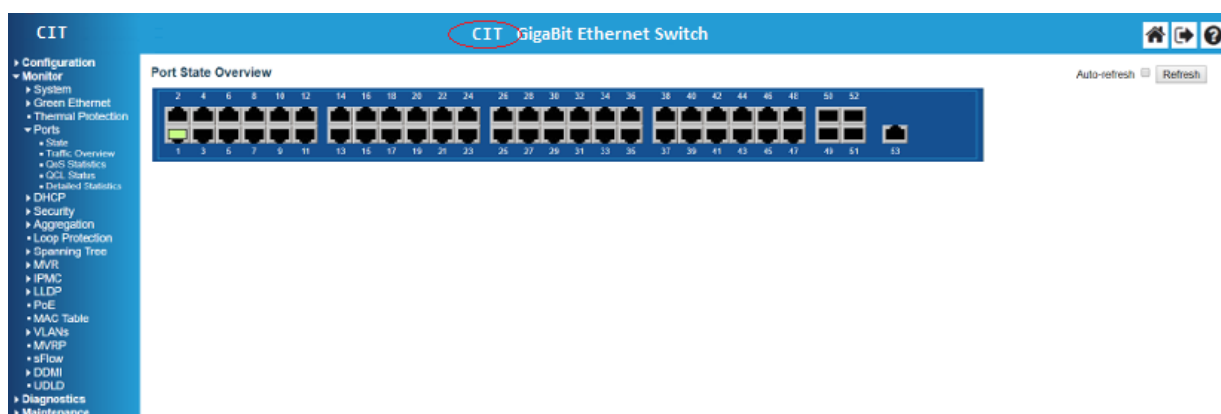


Рис. 2: Название продукта

Для изменения названия продукта необходимо добавить следующий текст:

```
Custom/SwitchName := <Product Name>
```

#### Пример:

```
Custom/SwitchName := Test_Switch
```

После изменения *config.mk* рекомендуется выполнить *make clean* перед *make*, для обеспечения правильной сборки с новыми настройками.

### 4.2 Изменение описания продукта

Описание продукта отображается в верхнем баннере рядом с названием продукта (рисунок 3). Для изменения описания продукта, необходимо изменить файл *build/make/config.mk*. *Config.mk* — это ссылка на файл *.mk* в каталоге *build/configs* на основе выбранной конфигурации.

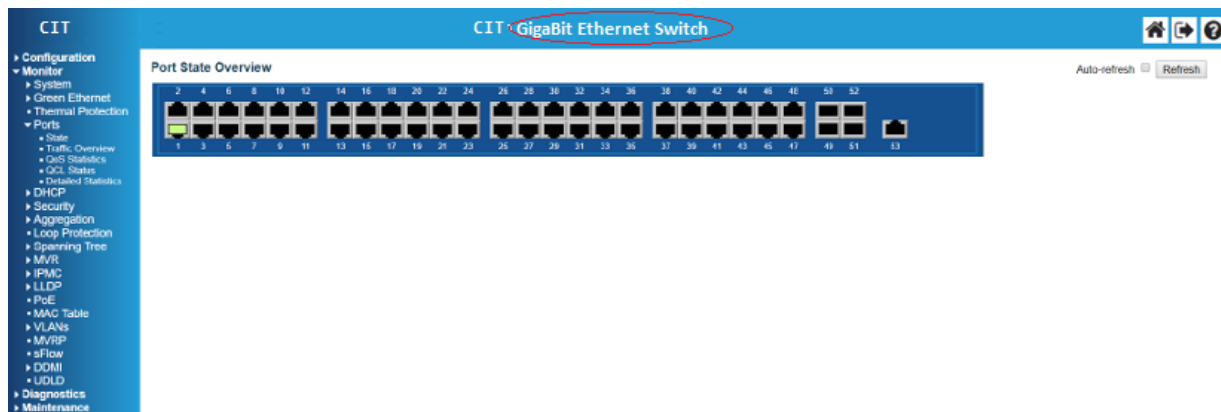


Рис. 3: Описание продукта

Для изменения описания продукта, необходимо добавить следующий текст:

```
Custom/SwitchDesc := <Product Description>
```

**Пример:**

```
Custom/SwitchDesc := 53 Port Gigabit Ethernet Switch
```

#### 4.3 Изменение логотипа продукта

Логотип продукта находится в верхнем левом углу баннера (рисунок 4).

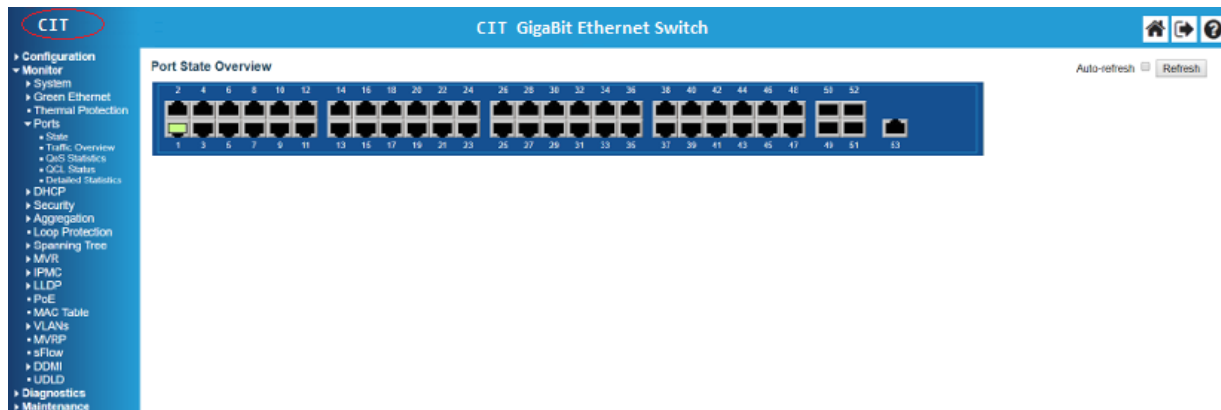


Рис. 4: Логотип продукта

Для изменения логотипа продукта, необходимо изменить следующие файлы:

- `b4cd_appl/web/html/logo.gif`
- `b4cd_appl/web/html/top.htm`

В файле `top.htm` необходимо найти следующую строку и изменить файл изображения логотипа продукта по желанию:

```

```

Если новое изображение не соответствует данному размеру, веб-браузер попытается подогнать изображение под размер. Побочным эффектом является снижение качества изображения. Поэтому рекомендуется использовать графический редактор для изменения размера изображения перед заменой `logo.gif`. Другой способ — указать атрибуты высоты и ширины изображения.

### Пример:

```

```

Данное действие может повлиять на макет и восприятие всей страницы. По данной причине может потребоваться изменение `b4cd_appl/web/html/index.htm`.

### Пример:

```
<frameset framespacing="0" border="0" frameborder="0" rows="54,*">
<frame name="top" noresize frameborder="0" marginwidth="0" \
marginheight="0" scrolling="no" border="no" src="top.htm" \
target="_self">
<frameset cols="166,*">
```

## 4.4 Изменение изображения коммутатора

Изображение целевой платформы (рисунок 5) отражает реальный продукт и обычно показывает название производителя и порты на передней панели.

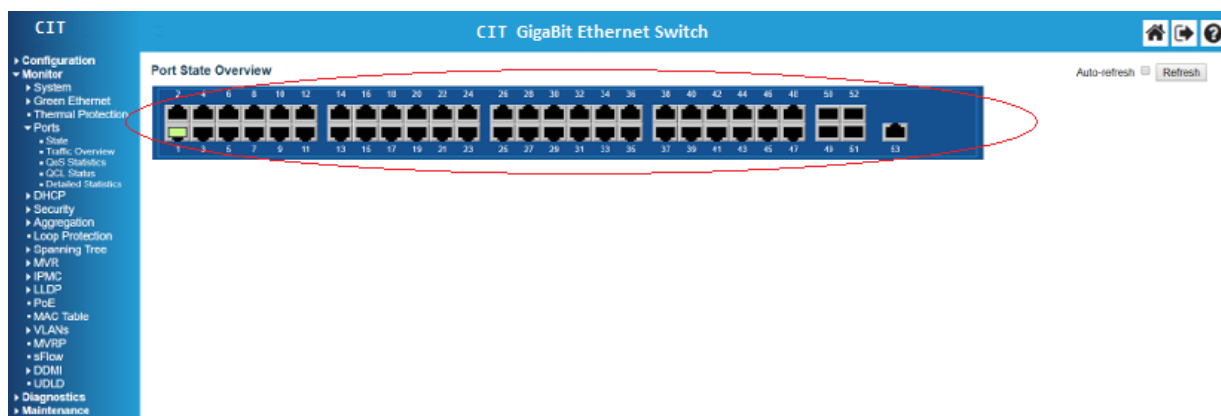


Рис. 5: Изображение коммутатора

Необходимо изменить соответствующий файл изображения: `switch_enzo.png`, `switch.png`, `switch_small.png` или `switch_48.png` в зависимости от используемой конфигурации. Рекомендуется использовать редактор изображений, чтобы настроить цвет и размер по желанию. Затем необходимо заменить файл изображения.

## 4.5 Изменение описания UPNP

Необходимо изменить `b4cd_appl/upnp/base/upnp_device.cxx`, заменив текст в следующем блоке кода на требуемый:

```

char g_xml_content[] =
    "<?xml version=\"1.0\"?>"
    "<root xmlns=urn:schemas-upnp-org:device-1-0\">"
    "<specVersion>"
    "<major>1</major>"
    "<minor>0</minor>"
    "</specVersion>"
    "<device>"
    "<deviceType>urn:schemas-upnp-org:device:Basic:1</deviceType>"
    "<friendlyName>B4CD</friendlyName>"
    "<manufacturer>B4CD Corp.</manufacturer>"
    "<manufacturerURL>http://b4cd.io</manufacturerURL>"
    "<modelDescription>Layer2+ Giga Stacking Switch SMBStaX</modelDescription>"
    "<modelName></modelName>"
    "<modelName></modelName>"
    "<serialNumber>%u</serialNumber>"
    "<UDN>%s</UDN>"
    "<serviceList>"
    "<service>"
    "<serviceType>urn:schemas-upnp-org:service:Layer4_Layer2:1</serviceType>"
    "<serviceId>urn:upnp-org:serviceId:Layer4_Layer21</serviceId>"
    "<controlURL></controlURL>"
    "<eventSubURL></eventSubURL>"
    "<SCPDURL></SCPDURL>"
    "</service>"
    "</serviceList>"
    "<presentationURL>http://%s:80</presentationURL>"
    "</device>"
    "</root>";

```

#### 4.6 Изменение версии программного обеспечения

Для просмотра информации о версии программного обеспечения, необходимо нажать «**Monitor**» > «**System**» > «**Information**». На информационной странице отображается информация о версии программного обеспечения (рисунок 6)

System Information	
<b>System</b>	
Contact	
Name	
Location	
<b>Hardware</b>	
MAC Address	00-01-c1-00-8d-c0
Chip ID	VSC7448
<b>Time</b>	
System Date	1970 01 01T00:37:54+00:00
System Uptime	0d 00:37:54
<b>Software</b>	
Software Version	dev-build by B4CD: 2018-01-19 1/12:00-18:00 Config: b4cd_1_00_00 SDK: 101/01-001
Software Date	2018-01-19T17:10:00+00:00
Code Revision	5199199
Acknowledgments	Details

Рис. 6: Информация о версии

Необходимо отредактировать файл *build/make/module\_main.in*, содержащий формат следующего блока кода. Далее необходимо найти следующий блок кода и при необходимости изменить значение версии – *\$1.o: RELEASE\_VERSION*:

```
version-$1.o: RELEASE_VERSION ?= $$ (BUILD_ID) Config:$1 Profile:$2$(if
$$ (MSCC_SDK_VERSION), SDK:$$ (MSCC_SDK_VERSION)-$$ (MSCC_SDK_FLAVOR))
```

Например, можно изменить вышеуказанную переменную как:

```
version-$1.o: RELEASE_VERSION ?= MyProd v1.00.
```

#### 4.7 Изменение цвета графического интерфейса

Домашняя страница (рисунок 7)(*index.htm*) содержит 3 фрейма HTML (набор фреймов), *top.htm*, *navbar.htm* и *top.htm*.

Цвет, шрифт и стиль в графическом интерфейсе описаны в следующих файлах CSS:

- *b4cd\_appl/web/html/lib/help.css*
- *b4cd\_appl/web/html/lib/menu.css*
- *b4cd\_appl/web/html/lib/normal.css*
- *b4cd\_appl/web/html/lib/top.css*

В следующем примере (рисунки 8 и 9) показано, как будет выглядеть графический интерфейс с разными шрифтами, цветами и стилями.

Для данного примера файл *top.css* имеет следующее содержимое:

```
diff --git a/vtss_appl/web/html/lib/top.css b/b4cd_appl/web/html/lib/top.css
index 4947ae9..34c38d0 100644
--- a/b4cds_appl/web/html/lib/top.css
+++ b/b4cd_appl/web/html/lib/top.css
@@ -34,7 +34,7 @@ div.helpbutton {
}
```

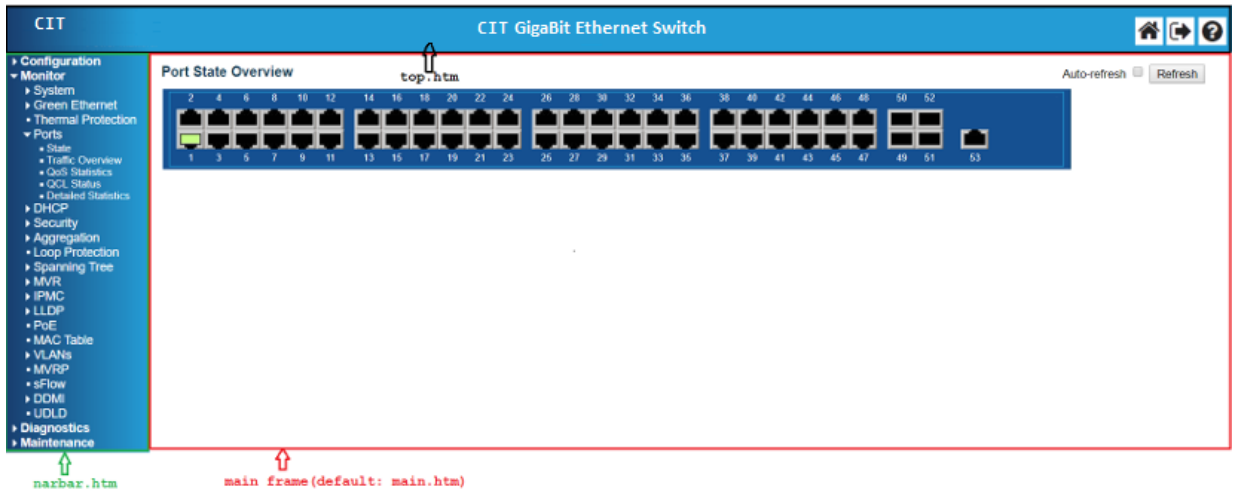


Рис. 7: Домашняя страница

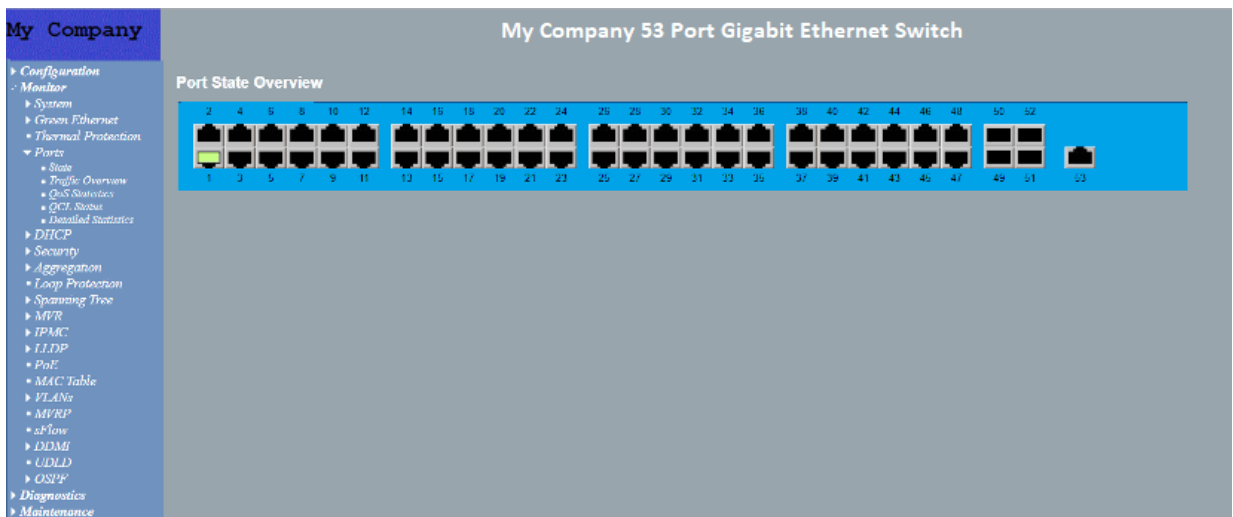


Рис. 8: Домашняя страница

```

body {
- background-color: #249CD5;
+ background-color: #98A5AC;
  color: white;
  font-family : Arial, Helvetica, sans-serif;
  margin: 0pt;

```

Файл *normal.css* для этого примера имеет следующее содержимое:

```

diff --git a/b4cd_appl/web/html/lib/normal.css b/b4cd_appl/web/html/lib/normal.css
index 6d92fea..3d97c40 100644
--- a/b4cd_appl/web/html/lib/normal.css
+++ b/b4cd_appl/web/html/lib/normal.css
@@ -33,7 +33,7 @@
body {

```

Port Configuration		My Company 53 Port Gigabit Ethernet Switch																
Port	Link	Speed		Adv Duplex		Adv speed					Flow Control			PFC		Maximum Frame Size		
		Current	Configured	Fdx	Hdx	10M	100M	1G	2.5G	5G	10G	Enable	Curr Rx	Curr Tx	Enable		Priority	
*			<>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
1	<span style="color: green;">●</span>	1Gtdx	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
2	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
3	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
4	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
5	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
6	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
7	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
8	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
9	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
10	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
11	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
12	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
13	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
14	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
15	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240
16	<span style="color: red;">●</span>	Down	Auto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0-7	10240

Рис. 9: Страница Port Configuration

```
font-family : Arial, Helvetica, sans-serif;
```

```
font-size : 10pt;
```

```
- background-color : white;
```

```
+ background-color : #B2BDC4;
```

```
}
```

```
body.content {
```

```
@@ -74,15 +74,16 @@ div.port_list {
```

```
h1 {
```

```
font-size : 12pt;
```

```
- color : #103851;
```

```
+ color : #FFFFFF; font-weight: bold;
```

```
}
```

```
h2 {
```

```
font-size : 11pt;
```

```
- color : #103851;
```

```
+ color : #FFFFFF; font-weight: bold;
```

```
}
```

```
td {
```

```
+ border: solid 1px #000000;
```

```
font-size : 10pt;
```

```
}
```

```
@@ -92,8 +93,13 @@ td.hdr {
```

```
}
```

```
td.hdr1 {
```

```
+ border: solid 1px #000000;
```

```
padding : 0ex 1ex 0ex 1ex;
```

```
font-weight : bold;
```

```
+ background-color : #B52C29;
```

```
+ padding-top: 10px;
```

```
+ padding-bottom: 10px;
```

```

+ color : #ffffe1;
}
td.crh {
@@ -107,8 +113,10 @@ td.cr {
}
td.cl {
+ border: solid 1px #000000;
padding : 0ex 1ex 0ex 1ex;
text-align : left;
+ background-color : #ffffe1;
}
td.dummy {
@@ -148,6 +156,7 @@ td.emphasize {
}
td.c {
+ border: solid 1px #000000;
text-align : center;
}
@@ -187,7+196,7 @@ td.linkUp {
.portlabel {
font-size : 8pt;
- color : #FFFFFF;
+ color : #000000;
text-align : center;
margin: 4px 0px 0px 0px;
}
@@ -202,26 +211,26 @@ table.config {
}
table thead tr, tr.config_header {
- background-color : #acdaf0;
- color: black;
+ background-color : #b52c29;
+ color: #ffffe1;
}
/* first and non-paired lines */
tr.config,tr.config_even {
- background-color : #FFFFFF;
+ background-color : #ffffe1;
}
tr.config_odd {
- background-color : #e1eaf0;
+ background-color : #ffffe1;
}
table.display {
- border: solid black 1px;
+ border: solid 1px #000000;
border-collapse: collapse;
}

```

```

tr.display_header {
- background-color : #acdaf0;
+ background-color : #b52c29;
color: black;
}
@@ -232,7 +241,7 @@ tr.display_separate_header{
/* first and non-paired lines */
tr.display, tr.display_even {
- background-color : #FFFFFF;
+ background-color : #ffffe1;
}
tr.display_odd {
@@ -240,14 +249,17 @@ tr.display_odd {
}
th {
- border: solid black 1px;
+ border: solid 1px #000000;
font-weight : bold;
font-size : 11pt;
padding : 0ex 1ex 0ex 1ex;
+ padding-top : 10px;
+ padding-bottom : 10px;
+ color : #ffffe1;
}
th.portno {
- border: solid black 1px;
+ border: solid 1px #000000;
font-weight : bold;
font-size : 9pt;
text-align : center;
@@ -256,14 +268,18 @@ th.portno {
/* TH-style - only the first cell is a header, next is typically an input element */
td.param_label {
+ border: solid 1px #000000;
font-weight : bold;
- background-color : #acdaf0;
+ background-color : #b52c29;
padding : 0ex 1ex 0ex 1ex;
+ padding-top : 10px;
+ padding-bottom : 10px;
+ color : #ffffe1;
}
/* Frame param leaders */
table.config td.param_label {
- border: solid black 1px;
+ border: solid 1px #000000;
height: 16pt;
}

```

```
@@ -271,6 +287,8 @@ table.config td.param_label {
td.displaytext {
font-weight : bold;
padding : 0ex 1ex 0ex 1ex;
+ padding-top : 10px;
+ padding-bottom : 10px;
}
.notice {
```

Файл *menu.css* для данного примера имеет следующее содержимое.

```
diff --git a/b4cd_appl/web/html/lib/menu.css b/b4cd_appl/web/html/lib/menu.css
index c7ba403..0bedaaf 100644
--- a/b4cd_appl/web/html/lib/menu.css
+++ b/b4cd_appl/web/html/lib/menu.css
@@ -28,7 +28,8 @@
*/
body {
- font-family: "Arial";
+ font-family: "fantasy";
+ font-style : italic;
margin: 0pt;
}
```

#### 4.8 Изменение IP-адреса по умолчанию

Архитектура системы конфигурации называется iCFG. В iCFG конфигурация по умолчанию содержится в одном текстовом файле.

Имя файла конфигурации *icfg* по умолчанию можно найти в *build/make/module\_icfg.in*. Ключевое слово поиска — *ICFG\_DEFAULT\_CONFIG*. В зависимости от файла конфигурации используются разные файлы *icfg*. Для *build/config.mk* с включенным голосовым vlan (voice vlan) файл *icfg* по умолчанию — *b4cd\_appl/icfg/icfg-default-config.txt*. Пользователь имеет возможность редактировать элементы в строке *ip address* в следующем блоке кода:

```
! Default configuration file
! -----
!
! This file is read and applied immediately after
! the system configuration is reset to default.
! The file is read-only and cannot be modified.
vlan 1
name default
voice vlan oui 00-01-E3 description Siemens AG phones
voice vlan oui 00-03-6B description Cisco phones
voice vlan oui 00-0F-E2 description H3C phones
voice vlan oui 00-60-B9 description Philips and NEC AG phones
voice vlan oui 00-D0-1E description Pingtel phones
voice vlan oui 00-E0-75 description Polycom phones
voice vlan oui 00-E0-BB description 3Com phones
```

```
interface vlan 1
ip address 192.0.2.1 255.255.255.0
end
```

#### 4.9 Добавление новой ссылки на панель навигации

Для изменения панели навигации, необходимо отредактировать файл *b4cd\_appl/web/menu\_default.cxx*. Поскольку файл *navbar.htm* создается динамически на основе *menu\_default.cxx*, после изменения *menu\_default.cxx* следует выполнить команду *make* и будет создан новый файл *navbar.htm*.

Ниже приведен пример добавления новой ссылки «Hello\_World» (рисунок 10):

```
diff --git a/b4cd_appl/web/menu_default.cxx b/b4cd_appl/web/menu_default.cxx
index b981492..950bbce 100644
--- a/b4cd_appl/web/menu_default.cxx
+++ b/b4cd_appl/web/menu_default.cxx
@@ -54,6 +54,7 @@ int WEB_navbar(CYG_HTTPD_STATE* p, int &size) {
ITEM("Configuration");
ITEM(" System");
+ ITEM(" Hello_World");
#if defined(B4CD_SW_OPTION_POST)
ITEM(" POST,post_config.htm");
#endif
```

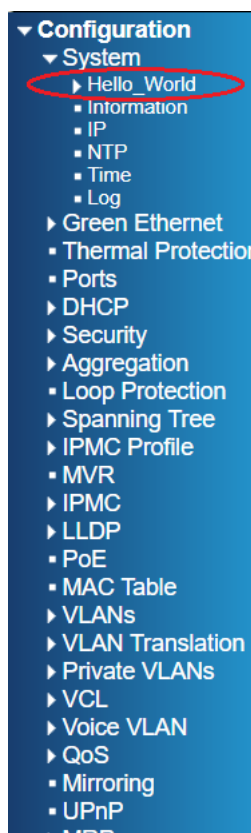


Рис. 10: Панель навигации

## 4.10 Изменение расположения портов

Информация о расположении портов (рисунок 11), находится в файле `b4cd_appl/port/port_web_xxx.c`, где `xxx` может быть `jr2`, `lu26` или `serval` в зависимости от используемой платформы.

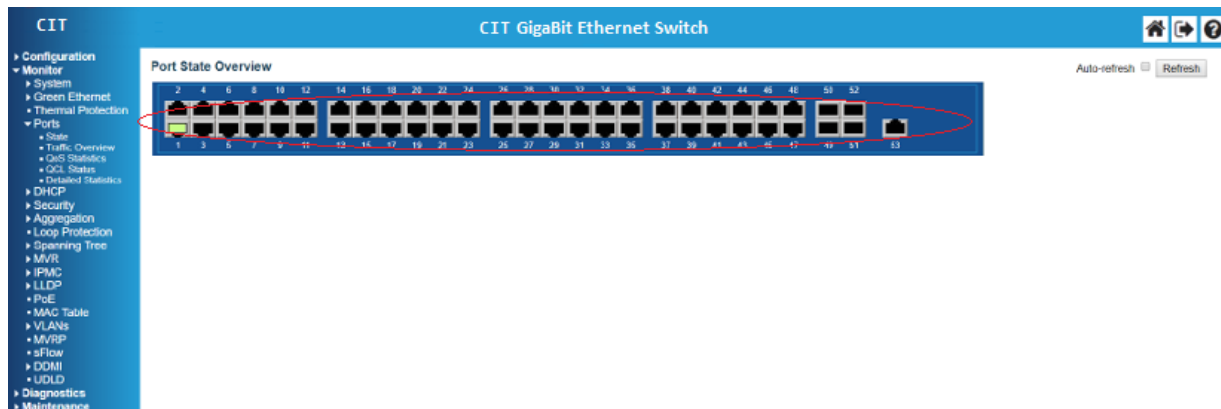


Рис. 11: Расположение портов

Расположение портов обрабатывается функцией `stat_portstate_switch_xxx()`, где `xxx` может быть `jr2`, `lu26` или `serval`.

С помощью данной функции настраивается расположение каждого значка порта. Нет необходимости изменять файл `html`. Следующие коды показывают возможность управления файлом изображения, смещая по оси X и Y и положением файла изображения для каждого порта:

```
for (iport = 0; iport < port_cnt; iport++) {
    if (port_custom_table[iport].cap & MEBA_PORT_CAP_SFP_DETECT) {
        if(lu26) {
            row = 0;
            col = 18 + (iport + 2 - mesa_port_cnt(nullptr)); // 2x SFP ports
            xoff = PORT_ICON_WIDTH + col * PORT_ICON_WIDTH + 20;
            yoff = PORT_ICON_HEIGHT * (2 - row);
        } else {
            row = 0;
            col = 10 + sfp_port;
            xoff = PORT_ICON_WIDTH + col * PORT_ICON_WIDTH + (12 * (col / 4 +
                (sfp_port++)));
            yoff = PORT_ICON_HEIGHT * 2;
        }
    } else {
        if(lu26) {
            /* Two columns */
            row = iport % 2;
            col = iport / 2;
            xoff = PORT_ICON_WIDTH + col * PORT_ICON_WIDTH + (12 * (col / 6));
            yoff = PORT_ICON_HEIGHT * (2 - row);
        } else {
            /* Single column */

```

```
row = 0;
col = iport;
xoff = PORT_ICON_WIDTH + col * PORT_ICON_WIDTH + (12 * (col / 4));
yoff = PORT_ICON_HEIGHT * 2;
}
}
```

## 5 Настройка JSON-RPC

JSON-RPC — это протокол, используемый для удаленных вызовов процедур. Сообщения, которыми обмениваются в JSON-RPC, представляют собой структуры данных в кодировке JSON. Протокол JSON-RPC имеет серверную и клиентскую части. Клиент инициирует связь, отправляя запрос на сервер, и сервер обрабатывает запрос, чтобы отправить ответ.

Спецификацию JSON-RPC см. на странице <http://json-rpc.org/>. Для получения информации об общей спецификации JSON см. <http://json.org>.

JSON-RPC предоставляет высокоуровневый интерфейс управления, который является функциональным эквивалентом ICLI или SNMP со следующими дополнительными свойствами:

- Удобный машинный интерфейс, который может быть легко интерпретирован людьми.
- Надежная связь, ориентированная на соединение, обеспечиваемая инкапсуляцией сообщений TCP и HTTP.
- Протокол RPC, совместимый с большинством языков программирования.
- Реализуется практически на любом языке и требует очень ограниченного объема памяти для программ и памяти для данных.

**Примечание** – JSON-RPC не является интерфейсом конечного пользователя (высокоуровневый машинно-дружественный интерфейс), предназначенным для взаимодействия с человеком. По этой причине целевая аудитория этого документа должна быть знакома с технологией JSON-RPC. Пользователям, еще не знакомым с JSON и JSON-RPC 1.0, рекомендуется ознакомиться с официальными стандартами.

### 5.1 Начало работы с JSON-RPC

Цель данного раздела — начало работы с JSON-RPC, убедившись, что программное обеспечение коммутатора поддерживает JSON-RPC и работает.

Для JSON-RPC требуется протокол разбора сообщений, который не является частью стандарта JSON-RPC, но должен предоставляться другими технологиями. Интеграция JSON-RPC по умолчанию выполняется с использованием протокола HTTP и существующего веб-сервера.

Приложение curl, доступное в большинстве систем Linux/Unix и Windows, можно использовать для отправки запроса JSON-RPC на устройство по протоколу HTTP. Приложение curl хорошо подходит для первоначальной отладки, но не следует полагаться на него при разработке приложений. Почти все языки программирования поддерживают либо встроенную, либо отдельную библиотеку JSON-RPC.

Ниже приведен пример использования curl:

```
curl http://admin:@target.ip.or.address/json_rpc -d \
  '{"method":"ip.status.interface.ipv4.get", \
   "params":[],"id":"0"}'
```

Пример результата:

```
{
  "id":"0",
  "error":null,
  "result":[
    {
      "key":{
```

```

        "VLAN 1",
        "10.99.10.12/24"
    },
    "val":{
        "broadcast":"10.99.10.255"
    }
}
]
}

```

### 5.1.1 Использование vson

Curl хорошо подходит для первоначальной отладки, но требует большого количества команд и не так удобен в использовании при изучении набора команд JSON-RPC. Для данных целей удобнее использовать консольное приложение vson, которое позволяет легко вызывать методы JSON-RPC и представляет результат в удобном для пользователя виде.

Команда для загрузки vson:

```
$ git clone https://github.com/vtss/json-rpc-util.git
```

Затем необходимо перейти в каталог json-rpc-util и выполнить следующую команду, чтобы отобразить все доступные параметры запуска:

```
$ ./vson -help
```

Кроме того, необходимо выполнить команду для загрузки спецификации json в *./bcd-json-rpc.spec*:

```
$ ./vson -d <target-IP-address> -c update-spec
```

Примерный результат выполнения запуска vson:

```

$ ./vson -d <target-IP-address> -c call port.status.get "Gi 1/1"
Calling port.status.get:
  Link: true
  Fdx: true
  Fiber: false
  Speed: speed1G
  SFPTypе: none
  SFPVendorName:
  SFPVendorPN:
  SFPVendorRev:
  LossOfSignal: true
  TxFault: true
  Present: false
  SFPVendorSN:

```

## 5.2 Спецификации интерфейса JSON

Спецификация интерфейса JSON используется для документирования методов, доступных в интерфейсе JSON.

Спецификация интерфейса JSON называется *json\_spec*. *json\_spec* — это документ в формате JSON с определенной схемой.

*json\_spec* обычно распространяется непосредственно на устройствах, и к нему можно получить доступ одним из следующих способов:

- Интернет — его можно загрузить с [http://address.or.ip.of.device/json\\_spec](http://address.or.ip.of.device/json_spec)
- JSON-RPC — ту же спецификацию можно также получить, вызвав команду `jsonRpc.status.introspection.generic.inventory.get`
- JSON-RPC (целевой) — вариант той же спецификации можно получить, вызвав команду `jsonRpc.status.introspection.specific.inventory.get` (Разница в том, что эта спецификация будет включать только атрибуты и методы, доступные на том устройстве, на котором она была создана)

### 5.2.1 Варианты использования

Ниже перечислены некоторые варианты использования, рассмотренные при разработке *json\_spec*:

- **Входные данные для автоматического создания клиентской библиотеки**

*json\_spec* включает всю информацию, необходимую для автоматического создания полной клиентской библиотеки. Автогенератор определяет язык, который будет использоваться для клиентской библиотеки, модели многопоточности, блокирующего/асинхронного взаимодействия и потенциальных сторонних библиотек. Процесс автоматического создания клиентской библиотеки представлен на рисунке 12.

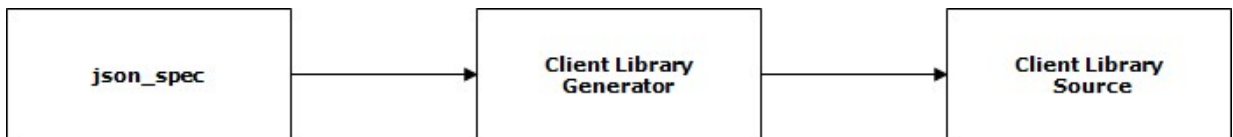


Рис. 12: Процесс автоматического создания клиентской библиотеки

- **Ввод в интерфейс генератора документации**

*json\_spec* включает все интерфейсы, необходимые для создания полного набора документации по интерфейсам. Генератор документации определяет целевой формат, стиль и другие детали. Процесс автоматического создания документа представлен на рисунке 13.

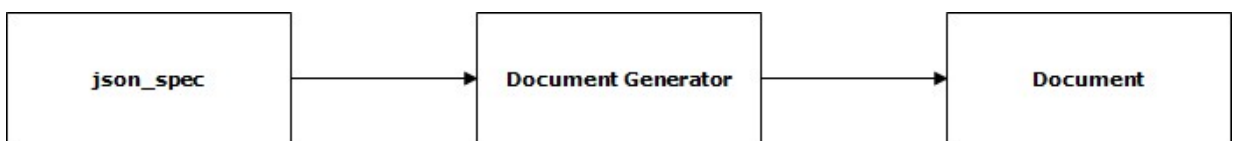


Рис. 13: Процесс автоматического создания документа

- **Автоматически заполняемый интерфейс управления**

*json\_spec* включает всю информацию, необходимую для создания полноценного интерфейса управления. Её можно использовать для создания интерфейса командной строки, веб-интерфейса или графического интерфейса ПК, который использует *json\_spec* для самоанализа. Цель состоит в том, чтобы избежать сохранения набора команд в Abstract Interface SW-Daemon (то есть клиентская часть json-rpc должна использовать *json\_spec* для определения поддерживаемых команд) при развитии набора команд в MSCC-Switch. Поток автозаполнения интерфейса управления представлен на рисунке 14.

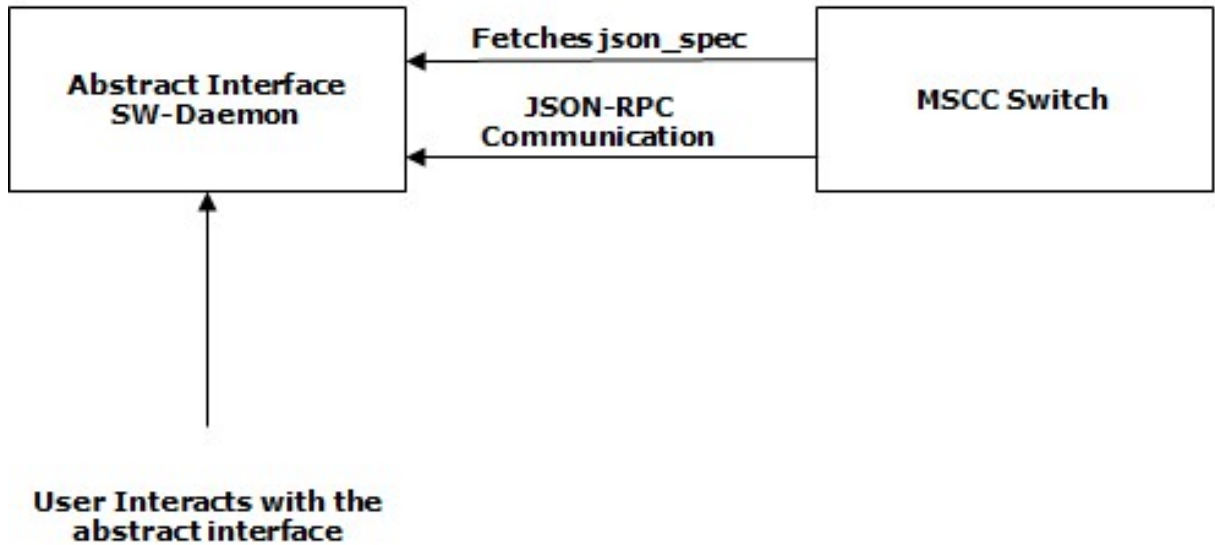


Рис. 14: Поток автозаполнения интерфейса управления

### 5.2.2 Возможности

Большая часть интерфейса JSON не зависит от используемой цели, но бывают случаи, когда это невозможно. Как правило, это связано с тем, что базовое оборудование предлагает различные возможности, которые необходимо предоставить пользователю. В таких случаях возможности используются для информирования о том, что доступно.

Возможность — это метод получения, который возвращает объект с набором логических значений. Данное логическое значение указывает, доступна ли данная возможность.

Атрибуты и методы, зависящие от возможности, предоставляют ссылку на возможность, от которой они зависят. Если пользователи интерфейса JSON-RPC используют только один целевой тип, может быть полезно создать версию *json\_spec* для конкретной цели.

- **Спецификация верхнего уровня**

*json\_spec* состоит из объекта, содержащего три списка: *types*, *groups* и *methods*:

```

{
  "types": [
    ...
  ],
  "groups": [

```

```

    ...
  ],
  "methods": [
    ...
  ]
}

```

— Список *types* представляет собой объединенный набор всех именованных типов, используемых всеми упомянутыми типами и методами.

— Список *groups* содержит объявления групп, на которые ссылаются другие группы или методы.

— Список *methods* — это набор доступных методов.

## • Спецификация типа

Спецификация типа используется для документирования семантики данного типа. Типы могут быть именованными или анонимными. Именованные типы включаются в список типов, а анонимные типы объявляются там, где они используются. Тип имеет базовую спецификацию и более сложную спецификацию, зависящую от поля класса. Ниже приведен пример базовой спецификации.

```

{
  ["type-name": "<name-of-type-goes-here>"],
  "class": "<TypeSpecification|Enum|Struct|Array|Tuple>",
  ["description": "<description-goes-here>"],
  "json-encoding-type": "<Number|String|Boolean|Array|Object|null>",
  ...
}

```

— *type-name* — имя типа. Это необязательно, поскольку не используется для анонимных типов, но экземпляры в списке *types* всегда будут включать имя. Когда другие типы или методы ссылаются на тип, они могут использовать это имя или включать тип непосредственно как анонимный тип.

— *class* используется для указания того, к какому семейству принадлежит данный тип. Поддерживаемые классы типов следующие:

- *TypeSpecification* — терминальный тип, то есть он не является комбинацией других предопределенных типов (например, IP-адрес, который закодирован как строка в JSON, но в программном интерфейсе должен находиться и быть представленным как выделенный тип).
- *Enum* — это тип, который используется для связывания строковых обозначений с числами. Интерфейс JSON использует только строковую нотацию, но программные интерфейсы могут выбрать внутреннее использование числа.
- *Struct* представляет собой совокупность других типов (именованных или анонимных).
- *Array* — это репликация одного типа в 0 или более экземплярах.
- *Tuple* — это отсортированная коллекция из нуля или более различных типов.
- *description* используется для документирования того, как определенные типы должны интерпретироваться и использоваться.
- *json-encoding-type* указывает, как данный тип кодируется в домене JSON.

**а) Спецификация типа**

Объект класса *TypeSpecification* использует ту же схему, что и базовый тип. Ниже приведены примеры объектов *TypeSpecification*.

**Пример IPv4:**

Это спецификация JSON типа *b4cd\_ipv4\_t*. В сообщениях JSON он будет закодирован как строка с использованием синтаксиса, определенного в поле описания. Клиент может сопоставить это с альтернативным типом, представляющим адрес IPv4.

```
{
    "type-name": "b4cd_ipv4_t",
    "class": "TypeSpecification",
    "description": "Ipv4 address encoded as ä.b.c.d", where
    a-d is a base-10 human readable integer in
    the range [0-255]",
    "json-encoding-type": "String"
}
```

**Пример IPv6:**

В следующем примере представлена спецификация JSON типа *b4cd\_ipv6\_t*.

```
{
    "type-name": "b4cd_ipv6_t",
    "class": "TypeSpecification",
    "description": "An Ipv6 address represented as human readable
    test as specified in RFC5952",
    "json-encoding-type": "String"
}
```

**б) Перечисление**

Объект класса *Enum* всегда имеет поле *elements*, представляющее собой список объектов, содержащих поля *name* и *values*.

```
{
    "type-name": "<name-of-type-goes-here>",
    "class": "Enum",
    "description": "<description-goes-here>",
    "json-encoding-type": "String",
    "elements": [
        "name": "<name-of-first-element>",
        "value": "<value-of-first-element>",
        "name": "<name-of-second-element>",
        "value": "<value-of-second-element>",
        ...
    ]
}
```

**Пример перечисления:**

Ниже приведен пример объекта *Enum*.

```
{
  "type-name": "b4cd_appl_ece_rule_t",
  "class": "Enum",
  "description": "The ECE rule type.",
  "encoding-type": "String",
  "elements": [
    "name": "both", "value": "0",
    "name": "rx", "value": "1",
    "name": "tx", "value": "2"
  ]
}
```

**с) Структура**

Тип объекта *Struct* имеет список *elements*, который всегда содержит набор дескрипторов элементов. Объект дескриптора элемента выглядит следующим образом:

```
{
  "name": "<name-of-first-element>",
  "type": "<type-of-first-element>",
  ["description": "<description-of-element>"],
  ["depends-on-capability":
    "<reference-to-dependency-it-depends-on>"]
}
```

Ниже перечислены значения каждого поля в *element-descriptor*:

- *name* — имя элемента в *struct*. Когда тип закодирован в JSON, имя является ключевой строкой, используемой в карте.
- *type* является либо ссылкой на тип в списке *types*, либо объявленным там анонимным типом.
- *description* — это необязательное значение, предоставляющее строку описания для данного элемента.
- *depends-on-capability* — необязательное значение, которое может указывать на возможность. Если этот атрибут присутствует в *element-descriptor*, то это означает, что данный элемент присутствует только в том случае, если возможность верна. Если атрибут отсутствует, то данный элемент всегда присутствует. Отсутствующий элемент (поскольку связанная с ним возможность является ложной для данной цели) игнорируется в операции установки и не включается в операцию получения.

В следующем примере показана структура *struct*, содержащая четыре элемента, два из которых присутствуют всегда, а два других зависят от возможностей.

```
b4cd_appl_evc_ece_conf_t in the json_spec.
{
  "type-name": "b4cd_appl_evc_ece_conf_t",
  "class": "Struct",
```

```

"description": "Example struct - not complete",
"encoding-type": "Object",
"elements": [
  {
    "name": "NextId",
    "type": "uint32_t",
    "description": "The ID of the next ECE."
  },
  {
    "name": "AdvLookup",
    "type": "b4cd_bool_t",
    "description": "Enable advanced ingress lookup.",
    "depends-on-capability":
      "HasEceAdvLookup@b4cd_appl_evc_capabilities_t"
  },
  {
    "name": "UniPortList",
    "type": "b4cd_port_list_stackable_t",
    "description": "List of UNI ports for the ECE."
  },
  {
    "name": "SrcMac",
    "type": "b4cd_mac_t",
    "description": "The source MAC address to match.",
    "depends-on-capability":
      "HasEceSrcMac@b4cd_appl_evc_capabilities_t"
  }
]
}

```

#### d) Массив

Объект типа *Array* имеет поле с именем *type*, которое указывает тип значений в массиве. Значение *type*, может быть ссылкой на именованный тип или непосредственно объявленным анонимным типом. Ниже приведен пример анонимного типа массива:

```

{
  "class": "Array",
  "encoding-type": "Array",
  "type": "b4cd_ifindex_t"
}

```

#### e) Кортеж

Объект типа *Tuple* похож на объект типа *Struct*. Отличие состоит в том, что поле имени в дескрипторе элемента является необязательным для массива (а не для кортежа), а *encoding-type* всегда является массивом. Ниже приведен пример дескриптора кортежа:

```

{
  "class": "Tuple",

```

```

"encoding-type": "Array",
"elements": [
  {
    "name": "ifindex",
    "type": "b4cd_ifindex_t"
  },
  {
    "name": "prio",
    "type": "int32_t"
  }
]
}

```

- **Спецификация группы**

Методы и группы могут быть частью групп. Группа может иметь описание, применимое ко всему, что хранится в этой группе. Для этого группы и методы могут идентифицировать группу, к которой они принадлежат. Такая группа является ссылкой на объект в списке групп. Объекты в списке групп имеют два обязательных поля — *group-name* и *description*, а также необязательное поле с *parent-group-name*:

```

{
  "group-name": "<name-of-group>",
  ["parent-group-name": "<name-of-parent-group>",]
  "description": "<description-of-group>"
}

```

Следующий код показывает полный пример:

```

{
  "group-name": "acl.config.global.ace",
  "description": "The ACL ACE configuration table."
},
{
  "group-name": "acl.config.global.ace.hardware",
  "parent-group-name": "acl.config.global.ace",
  "description": "Table of ACE hardware status."
}

```

- **Спецификация метода**

Метод определяется с использованием следующего синтаксиса:

```

{
  "method-name": "<name-of-method>",
  ["group-name": "<name-of-group>",]
  ["description": "<description-of-method>",]
  ["notification": "<name-of-notification>",]
  ["depends-on-capability": "<reference-to-dependency-it-depends-on>",]
  "params": [

```

```

    ...
  ],
  "results":[
    ...
  ]
}

```

Ниже перечислены значения синтаксиса:

- *method-name* — имя метода.
- *name-of-group* — ссылка на группу, к которой принадлежит данный метод. Описание группы должно наследоваться как описание метода.
- *description* — описание метода. Данное поле редко используется для методов, но вместо этого большинство методов документируются через группу.
- *depends-on-capability* — если этот атрибут присутствует, то это означает, что этот метод реализуется только для целей, где данная возможность оценивается как истинная. Если данный атрибут отсутствует, метод всегда реализуется (при условии, что модуль включен в цель).

Поля *params* и *results* представляют собой списки спецификаций аргументов со следующим синтаксисом:

```

{
  "name":"<name-of-argument>",
  ["description":"<description-of-argument>"],
  "type":"<name-of-type>"
}

```

### Пример:

```

{
  "method-name":"ip.statistics.interface.ipv4.get",
  "params":[
    {
      "name":"ifidx",
      "type":"b4cd_ifindex_t"
    }
  ],
  "results":[
    {
      "name":"stats",
      "description":"Interface statistics",
      "type":"b4cd_appl_ip_if_status_ip_stat_t"
    }
  ]
}

```

### 5.3 Обратная и прямая совместимость

В данном разделе описываются функции реализованные и поддерживаемые в *json\_spec*, но при необходимости данная спецификация может быть изменена в будущих выпусках. Если такие изменения имеют место, они должны быть обратно совместимыми. Обратная и прямая совместимость гарантирует, что клиент JSON-RPC продолжит взаимодействие с сервером при обновлении одного из них. Обратная совместимость необходима, когда сервер (программное обеспечение целевой платформы) обновляется и реализует более новую спецификацию JSON, в то время как клиент продолжает использовать более старую версию спецификации JSON. Прямая совместимость необходима, когда клиент обновляется и теперь поддерживает более новую версию спецификации JSON, чем та, которую поддерживает сервер.

Совместимость — это совместная работа сервера и клиента; сервер и клиент должны быть предназначены для достижения этой цели. В данном разделе описывается необходимая настройка на стороне сервера. Клиент должен быть правильно настроен для требуемой совместимости клиент-сервер.

#### 5.3.1 Рекомендации по внедрению изменений

Обеспечение 100% обратной и прямой совместимости для всего интерфейса JSON-RPC значительно усложняет исправление ошибок и внедрение новых функций.

Далее изложены рекомендации по внедрению изменений:

- **Основные доработки**

Переработка модуля может быть несовместима с предыдущими версиями. Данная проблема может произойти только при добавлении крупной новой функции. Такие изменения документируются в примечаниях к выпуску, а новый интерфейс модуля получает новое имя. В этом случае все методы должны быть переименованы. Данное действие гарантирует, что изменение будет замечено клиентом. В некоторых случаях в переходном выпуске могут быть доступны как новый, так и старый интерфейс.

- **Нормальная эволюция**

Модули развиваются с добавлением новых функций, которые добавляются путем расширения существующих структур новыми полями или методами. В таких случаях клиенты могут быть готовы к совместимости, игнорируя неизвестные поля. Сервер игнорирует попытку клиента настроить поле в *struct*, которую он не распознает. Если клиент попытается вызвать метод на сервере, который не реализован, он получит ошибку «*textitMethod not found*» *error code -32601*.

Совместимость обеспечивается, если клиент готов к таким возможностям. Такие изменения можно найти, изучив различия между предыдущей и текущей версией спецификации JSON. Проверить такие различия можно с помощью инструмента *b4cd-json-spec-diff*, который доступен вместе с инструментом *vsom*.

- **Исправления ошибок**

Ошибки можно исправить без изменения интерфейсов JSON-RPC, и совместимость не должна быть проблемой. Любые необходимые изменения классифицируются как часть нормальной эволюции.

### 5.3.2 Реализация совместимости

Далее описывается реализация для обеспечения совместимости, учитывая, что целевые системы не знают о будущих версиях или различных версиях клиентов, которые будут получать доступ к интерфейсу JSON-RPC:

- **Методы**

Целевая система реализует набор методов, соответствующих текущей версии спецификации JSON. Если клиент пытается вызвать несуществующий метод, сервер отвечает сообщением об ошибке «*Method not found*». Клиент должен распознавать неподдерживаемые ошибки метода и адаптироваться к ситуации.

Если клиент новее сервера может возникнуть проблема прямой совместимости. В большинстве случаев существующие методы сохраняются, а новые добавляются, если только не проводится серьезная переработка. Клиент должен правильно обрабатывать этот сценарий.

- **Структуры (объекты JSON)**

В реализации сервера JSON-RPC все поля в структурах считаются необязательными. Сервер игнорирует поле в struct, которое он не распознает. Если серверу не хватает назначений одного или нескольких полей в *set/add request*, он использует значения по умолчанию.

Для методов *set* значения по умолчанию предоставляются следующим образом:

- Выполнение запроса *get*.
- Перезапись значений, предоставленных *get* запросом, новыми значениями из *set* запроса.
- Вызов метода управления *C* для применения запроса. Данное исправление выполняется только на первом уровне. Если открыты вложенные структуры, этот процесс не применяется рекурсивно.

Для *add* или *set-in-write-only-context* процесс аналогичен, за исключением того, что вместо метода *get* используется внутренний метод *defaults*.

## 5.4 Модели интеграции

JSON-RPC — это только протокол для разрешения удаленных вызовов процедур. Он не определяет средства синтаксического анализа сообщений и предполагает, что это делается другими средствами. HTTP(s) — это наиболее часто используемый протокол обмена сообщениями, используемый вместе с JSON-RPC.

В этом разделе приведены рекомендации по использованию HTTP в качестве протокола обмена сообщениями. В нем также упоминаются альтернативы использованию HTTP(s) для обмена сообщениями.

### 5.4.1 HTTP(s) как анализ сообщений

Чтобы использовать сервер JSON-RPC с HTTP(s) в качестве протокола обмена сообщениями, отправьте запрос JSON-RPC на адрес */json\_rpc* и дождитесь ответа.

**Примечание:** — HTTP-клиент всегда должен использовать свойство *keep-alive*, чтобы разрешить повторное использование одного и того же TCP-соединения для ряда команд.

Запуск JSON-RPC через HTTP(s) — идеальное решение, если соблюдены все следующие

критерии. Обычно это происходит при использовании JSON-RPC для систем удаленного управления или других слабо связанных систем:

- Связь RPC должна проходить через соединение TCP/IP.
- RPC-связь должна иметь возможность проходить через IP-сеть, или что необходимы гарантии, предоставляемые TCP.
- Средства безопасности, предоставляемые HTTP(s), соответствуют критериям, требуемым для связи RPC.
- Несмотря на то, что HTTP использует базовую аутентификацию без какого-либо шифрования, в целевой системе можно включить HTTPS. Это позволит JSON-RPC работать через HTTPS и использовать атрибуты безопасности, предоставляемые HTTPS.

Запуск JSON-RPC через HTTP(s) не следует рассматривать как вариант, если требуется какое-либо из следующих свойств:

- Связь RPC должна работать, когда цель не имеет IP-адреса.
- Средства безопасности, предоставляемые HTTP(s), недостаточно хороши.
- Накладные расходы IP + TCP + HTTP неприемлемы.
- Трафик управления не должен проходить через обычное соединение Ethernet.

Если какие-либо из этих свойств необходимы, JSON-RPC через HTTP(s) использовать нельзя. В этом случае можно предоставить альтернативный протокол обмена сообщениями.

#### 5.4.2 Альтернативные системы анализа сообщений

Когда JSON-RPC через HTTP(s) не является оптимальным решением, можно интегрировать альтернативное средство обмена сообщениями с желаемыми характеристиками.

Для этого необходимо разработать альтернативное средство обмена сообщениями и интегрировать сервер JSON-RPC. Интеграция внутреннего сервера JSON-RPC должна быть простой, если обмен сообщениями налажен и работает.

Интеграцию JSON-RPC через HTTP(s) можно использовать в качестве примера при регистрации JSON-RPC поверх других средств. Эта интеграция находится в *b4cd\_appl/json\_rpc/json\_rpc\_platform.cxx* в функции *json\_http\_handler*. Интеграция внутреннего сервера JSON-RPC с новым средством обмена сообщениями выполняется путем вызова следующего метода C++:

```
namespace b4cd {
namespace expose {
namespace json {
vtss::json::Result::Code RootNode::process(str input,
                                           ostreamBuf &out);
} // namespace json
} // namespace expose
} // namespace b4cd
```

Методы должны вызываться для экземпляра *JSON\_RPC\_ROOT*, который является статической переменной в *b4cd\_appl/json\_rpc/json\_rpc\_platform.cxx*. Данный метод является потокобезопасным и может вызываться из многопоточного приложения без принятия каких-либо мер предосторожности.

Параметр *input* — это ссылка на входящий запрос JSON-RPC, который необходимо обработать. Ответ JSON-RPC записывается в поток буфера *out*.

Кандидатами на использование этого средства являются двухпроцессорные системы, в которых передача сообщений осуществляется по выделенным каналам, или тесно связанные системы, в которых требование функционального соединения ТСР/ІР невозможно (например, устройству не разрешено иметь настроенный ІР-адрес).