Release Notes
For
Switch Software Development Kit

SDK 6.5.5

TABLE OF CONTENTS

Section 1: About This Document

These are the Release Notes for the Broadcom Network Switching Software Development Kit Release 6.5.5.

This document provides a general description of the release and its new features. It also describes the chips supported by the release, BCM API additions or changes, resolved issues, and any relevant open issues. The reader should refer to prior release notes for 6.5.x, as only new features or issues are described in this version of the release notes.

Section 2: Product Documentation

The following documents are available through Broadcom's Customer Support Portal, http://support.broadcom.com. They are the primary source of information and should be referenced when using this release:

Document	*Description*
56XX-PG655-R	*BCM API Reference Guide* This manual describes the theory of operations of the API and all existing BCM APIs for this release.
56XX-PG819-R	*Network Switching Software Platform Guide* This guide describes the SDK source and Makefile structure, abstraction and porting layers, device specific interactions, and the platform/operating system specific features of the SDK. If this is your first time working with the SDK, start with this document.
56XX-PG707-R	*Stacking Guide* This guide describes how to use the discovery and stacking applications provided in this release.

Section 3: New Devices added to this release

For any given SDK release, support for certain devices may be provided in Preview or Supported status. Devices in "Supported Switch Devices" and "Supported PHYs" have completed the full QA process and are intended for use in production systems. It is expected that customers would integrate the version of the SDK which provides "Supported" status for their use on actual development or production systems.

Devices in "Preview Switch Devices" and "Preview PHYs" are provided to allow early integration of the customer's application with the SDK APIs that support that device. This software has not been fully tested on the physical target device and should not be expected to fully function.

Section 3.1: Supported Switch Devices

Family	*Devices*	*Description*
BCM56565	BCM56565 B0	24x10GbE + 24x1GbE+ 4x10GbE+ 2xHG[106] Ethernet Embedded Switch
	BCM56567 B0	48x 1 / 2.5 GE + 8x10G (uplink) + 2xHG[106]
	BCM56568 B0	24xMGig + 24 x2.5GE + 4xHG[42] + 4x25GE
BCM56560	BCM56560 B0	2xCAUI + 4x40GbE + 12x10GbE Ethernet Embedded Switch
BCM56965	BCM56965 A0	32x 100G/64x 40G/128x 25G
	BCM56966 A0	24x 100G/48x 40G
	BCM56967 A0	6x 100G + 48x 25G
	BCM56969 A0	64x 10G
BCM56270	BCM56270 A0	4x 2.5GbE Port Integrated Low-Power Carrier Ethernet Access Switch with 8x 10GbE Uplinks, and Stacking
	BCM56271 A0	4x 2.5GbE Port Integrated Low-Power Carrier Ethernet Access Switch 4x 10GbE + 4x 2.5GbE Uplinks, and Stacking
	BCM56272 A0	4x 1GbE Port Integrated Low-Power Carrier Ethernet Access Switch with 2x 10GbE Uplinks, and Stacking
	BCM53460 A0	4x 2.5GbE Port Integrated Low-Power Carrier Ethernet Access Switch with 8x 10GbE Uplinks, and Stacking
	BCM53461 A0	4x 1GbE Port Integrated Low-Power Carrier Ethernet Access Switch with 4x 10GbE + 4x 2.5GbE Uplinks, and Stacking
BCM53400	BCM53369 A0	12 x 10G + 12 x 1G/2.5G

Section 3.2: Preview Switch Devices

Family	*Devices*	*Description*
BCM56760	BCM56760 B0	6x40GbE+ 48x10GbE Ethernet Embedded Switch
	BCM56762 B0	24x10GE + 6x40GE with support for 25GE
	BCM56765 B0	24x10GE + 6x40GE with support for 25GE
	BCM56768 B0	48x10GE + 4x100GE
BCM88270	BCM88270 A0	28x2.5GE + 8x10GE

Section 3.3: Supported PHYs

Family	*Devices*	*Description*	*Switch qualified against*

Section 3.4: Preview PHYs

Family	*Devices*	*Description*	*Switch qualified against*
BCM82332	BCM82332	1x 100G/3x 40G/12x 10G Ethernet PHY	BCM56860
	BCM82793	100G Gearbox for 100GBASE CR4/SR4/LR4 & SR10	BCM56860
BCM84848	BCM84888	10G/5G/2.5G/1000/ 100BASE-T Transceiver	BCM56860

Section 4: New Features per Device

Section 4.1: BCM56565 (FIREBOLT5) B0 FAMILY GENERAL AVAILABILITY (GA) RELEASE

The BCM56565 family is a purpose built device for access and aggregation deployment in the enterprise network. The BCM56565 family offers 48 front panel ports with 24 ports of MGig that offer complete flexibility with support for 1/2.5/5/10G. In addition BCM56565 also offers 200Gbps of stacking bandwidth (HiGiG only) for the new enterprise fixed/stackable market.

Config supported in this release for BCM56568:

- 24x10GbE (MGig) + 24x1GbE+ 4x10GbE (Uplink) + 2xHG[106] (stacking)
- 24x10GbE + 24x1/2.5 GbE +1x100GbE/4x25GbE (Uplink) + 4xHG[42]
- 24x10GbE + 24x1GbE + 8x10GbE (Uplink) + 4xHG[42]
- 24x10GbE + 24x1/2.5GbE+8x10GbE (Uplink) + 4xHG[42]
- 24x10GbE + 24x1/2.5GbE +1x100GbE/4x25GbE (Uplink) + 2xHG[106]
- 24x10GbE + 24x1GbE + 8x10GbE (Uplink) + 2xHG[106]
- 24x10GbE + 24x1/2.5GbE + 8x10GbE (Uplink) + 2xHG[106]
- 24x10GbE + 24x1GbE +2x100GbE (Uplink)
- 24x10GbE + 24x1/2.5GbE + 2x100GbE (Uplink)

Config supported in this release for BCM56565:

- 24x10GbE+24x1/2.5GbE+1x100GbE/4x25GbE (Uplink)+4xHG[42] (stacking)
- 24x10GbE+24x1/2.5GbE+8x10GbE (Uplink)+4xHG[42] (stacking)
- 24x10GbE+24x1/2.5GbE+1x100GbE/4x25GbE (Uplink)+2xHG[106] (stacking)
- 24x10GbE+24x1/2.5GbE+8x10GbE (Uplink)+2xHG[106] (stacking)
- 24x10GbE+24x1+2x100GbE (Uplink)
- 24x10GbE+24x1/2.5GbE+2x100GbE (Uplink)
- 24x10GbE+24x1/2.5GbE+4x10GbE (Uplink) + 2xHG[106] (stacking)
- 24x10GbE+24x1/2.5GbE+2x100GbE (Uplink)

Config supported in this release for BCM56567:

- 48x1/2.5GbE+ 8x10GbE + 4xHG[42] (stacking)
- 48x1/2.5GbE+ 8x10GbE + 2xHG[106] (stacking)
- 48x1/2.5GbE + 2xHG[42] (stacking)

This release provides General Availability (GA) support for BCM56565 B0, BCM56567 B0, and BCM56568 B0 devices.

Section 4.1.1: Additional features supported

Section 4.1.1.1: B0 Features Added

The B0 revision chips in BCM56565 family supports all the features that the respective A0 revision chips supported.

In addition, these B0 revision chips will also support Round Robin port selection criteria as a load balancing scheme for Front Panel Trunk groups. A maximum of two (2) trunk groups can be configured to distribute traffic across the member ports in a round robin fashion. Weighted round robin load balancing among the member ports can be achieved by adding the same member port in the trunk group multiple times.

Section 4.1.1.2: IEEE CL91 25G/50G AN FEC Support

In this release CL91 support is added into 25G/50G AN and FEC is added into bcm_port_ability to support this HW feature. The Falcon core microcode level has been upgraded to D10B_0E in this release.

Section 4.2: BCM56760 (MAVERICK) B0 Family Beta support

The BCM56760 family is a class of high performance, nonblocking network switching devices supporting up to 720Gbps to address performance, capacity and service requirements of next generation data center and cloud computing environment. They feature multiple integrated TSC Serdes core with support for 100GE delivering high-bandwidth, glueless network connectivity.

This release supports the following port configurations:

For BCM56760:
- 48x10GbE + 6x40GbE (ToR configuration)
- 72x10GbE
- 4x50GbE + 48x10GbE + 1x10GbE
- 2x100GbE + 4x40GbE+ 8x25GbE + 12x10GbE + 2x10GbE

For BCM56765:
- 72x10GbE
- 2x100GbE + 48x10GbE
- 16x25GbE + 8x40GbE
- 48x10GbE + 6x40GbE
- 18x40GbE

This release provides Beta support for BCM56760 B0 and BCM56765 B0. BCM56762 B0 and BCM56768 B0 are supported as Preview only in this release.

Section 4.2.1: Additional Features Supported

Section 4.2.1.1: B0 Features Added

The B0 revision chips in BCM56760 family supports all the features that the respective A0 revision chips supported.

In addition, these B0 revision chips will also support Round Robin port selection criteria as a load balancing scheme for Front Panel Trunk groups. A maximum of two (2) trunk groups can be configured to distribute traffic across the member ports in a round robin fashion. Weighted round robin load balancing among the member ports can be achieved by adding the same member port in the trunk group multiple times.

Section 4.2.1.2: IEEE CL91 25G/50G AN FEC Support

In this release CL91 support is added into 25G/50G AN and FEC is added into bcm_port_ability to support this HW feature. The Falcon core microcode level has been upgraded to D10B_0E in this release.

Section 4.2.2: Known Limitations

All features are at GA level in this release. However, final MMU threshold settings for the device in cut-through mode are not yet integrated into the SDK. Broadcom is working to integrate these settings in a future SDK release.

Section 4.3: BCM56965 (TOMAHAWK+) FAMILY GENERAL AVAILABILITY (GA) RELEASE

The Broadcom® BCM56965 family is a class of high-performance, high-connectivity network switching devices supporting up to 32x 100GbE, 64x 40GbE, 128x 25GbE, or 128x 10GbE switch ports. The device family features a maximum of 32 integrated FalconCore, each with four integrated 25G SerDes transceivers and associated PCS for native support of XFI, 10GBASE-KR/CR/SR/ER/LR, 40GBASE-KR4/CR4/SR4/ER4/LR4, 25GbE, 50GbE, 100GBASE-KR4/CR4/SR4/ER4/LR4, and Broadcom's proprietary MLD-HiGig2™ up to HG[106]. The BCM56965 delivers high-bandwidth, glueless network connectivity up to 3.2 Tbps on a single chip.

This release supports the following port configurations: 32x100G, 128x10G, 128x25G, 28x100G, 4x106G, 32x40G, 64x40G

The BCM56965 A0, BCM56966 A0, BCM56967 A0, and BCM56969 A0 SKUs are supported with GA level quality in this release.

Section 4.3.1 New Features in this Release

Section 4.3.1.1: Default MMU setting change

Default MMU Setting is now lossy vs lossless in the earlier generations of switch chips. To specifically set the lossless mode please use the configuration variable as follows:

mmu_lossless=1 /* enable lossless - set to the lossless mode */
mmu_lossless=0 /* disable lossless - set to the lossy mode */

Section 4.3.1.2: PortID field changes

bcm_port_class_set()/bcm_port_class_get() APIs now expanded to handle 8-bit class_id field due to expanded portID field for the BCM56965 family.

Section 4.3.1.3: IEEE CL91 25G/50G AN FEC Support

In this release CL91 support is added into 25G/50G AN and FEC is added into bcm_port_ability to support this HW feature. The Falcon core microcode level has been upgraded to D10B_0E in this release.

For auto-negotiation, use the bcm api to advertise both speed and FEC related property: `bcm_port_ability_advert_set(unit, port, &port_ability);`

For the forced speed, following bcm api should be used to enable either CL74 or CL91. To enable CL91 use the following:

```
bcm_port_phy_control_set(unit, port,
BCM_PORT_PHY_CONTROL_FORWARD_ERROR_CORRECTION_CL91, 1);
```

To enable CL74 use the following:

```
bcm_port_phy_control_set(unit, port,
BCM_PORT_PHY_CONTROL_FORWARD_ERROR_CORRECTION, 1);
```

For details on these BCM APIs please consult the programming guide (56XX-PG6412-R).

A static config is added also to enable FEC during init time. FEC is off by default.
`serdes_fec_enable=0` no FEC
`serdes_fec_enable=1` enable CL74
`serdes_fec_enable=2` enable CL91

Section 4.3.1.4: SDK overall feature status for BCM56965

In addition to the new features mentioned previously, the BCM56965 family supports all BCM56960 features at GA quality.

Section 4.4: BCM56560 (APACHE) B0 GENERAL AVAILABILITY (GA) RELEASE

The BCM56560 family is a class of high-performance, nonblocking network switching devices supporting up to 700Gbps switch ports bandwidth. Each device family features multiple integrated TSC Serdes Core, each with four integrated 100G SerDes transceivers and associated PCS for native support of SGMII (10/100/1000BT), XFI, XAUI, 10GBASE-KR/CR/SR, 40GBASE-KR4/CR4, XLAUI, CAUI4, and Broadcom's proprietary HiGig2(TM) at different speeds. The BCM56560 delivers high-bandwidth, glueless network connectivity up to 700 Gbps on a single chip.

This release supports the following port configurations:

- 2xCAUI10 + 48x10GbE
- 2xCAUI + 48x10GbE + 2x10GbE
- 2xCAUI10 / HG[106] + 16x25GbE
- 6x40GbE + 24x10GbE + 2x10GbE
- 2xCAUI10 + 2xCAUI + 24x10GbE + 2x10GbE

Section 4.4.1: Additional Features Supported

Section 4.4.1.1: B0 Features Added

The B0 revision chips in BCM56560 family supports all the features that the respective A0 revision chips supported.

In addition, these B0 revision chips will also support Round Robin port selection criteria as a load balancing scheme for Front Panel Trunk groups. A maximum of two (2) trunk groups can be configured to distribute traffic across the member ports in a round robin fashion. Weighted round robin load balancing among the member ports can be achieved by adding the same member port in the trunk group multiple times.

Section 4.4.1.2: IEEE CL91 25G/50G AN FEC Support

In this release CL91 support is added into 25G/50G AN and FEC is added into bcm_port_ability to support this HW feature. The Falcon core microcode level has been upgraded to D10B_0E in this release.

Section 4.5: BCM56270 (METROLITE) A0 FAMILY GENERAL AVAILABILITY (GA) RELEASE

The Broadcom BCM56270 is purpose-built for next-generation Carrier Ethernet Edge and Mobile Backhaul platforms. It combines a feature-rich packet processing engine, integrated hierarchical traffic manager, address management, and a non-blocking switch fabric into a single 28 nm CMOS device. The BCM56270 supports an integrated packet buffer. The BCM56270 is capable of handling L2 switching, L3 routing, metro VPN tunneling, and access control lists (ACLs). The BCM56270 virtualized architecture enables seamless support for VPLS, VPWS, MPLS-TP, PBB, and PBB-TE tunneling. In addition, the BCM56270 supports internal processing of Ethernet OAM, BFD, and BHH to enable a complete host CPU offload for these functions.

Configs supported in this release:

- 8x1G/2.5G/10G + 4x1G/2.5G
- 4x1G/2.5G/10G + 8x1G/2.5G
- 2x1G/2.5G/10G + 4x1G/2.5G G.INT
- 4x1G/2.5G/10G + 4x1G/2.5G G.INT
- 8x1G
- 2x1G + 4x1G G.INT

No new bundled External PHY drivers are planned for this device. New External PHY drivers should be obtained from the Broadcom PHY team as standalone PHY drivers.

Section 4.5.1: Additional Features Supported

Section 4.5.1.1: L3 SAT

Layer 3 Service Activation Testing (L3SAT) is a validation of a Layer 3 Service for a customer by a service provider before activating the service. The customer may be another service provider, upstream or downstream, or an end-point customer.

The goal of such an exercise is to check that the Service is correctly configured and meets Service Level Agreement (SLA) objectives before taking the Service online.

BCM56270 has Layer 3 Service Activation Testing, including Layer 3 SAT Generation, L3 SAT Termination, and Layer 3 Loopback. It builds upon the existing Layer 2 SAT added in BCM56260.

Section 4.6: BCM88060 (MONTREAL2) A0 Beta support

The Broadcom BCM88060 is a second generation Fibre Channel (FC) and Fibre Channel-over-Ethernet (FCoE) mapper. The device enables a flexible chip to be deployed in front of a data center switches to connect to FC endpoints directly. This device provides the flexibility to support FC/FCoE/Ethernet universal port capability as needed. The device incorporates the FC mapping into and out of Ethernet (FCoE) in a physical layer device. As a universal port device, the FCoE capability can optionally connect to a host of switches via a standard Ethernet when the FC universal port is needed.

This release supports the following device firmware: BCM8806x version 1.0.5 (rc/firmware/BCM8806x) which should be copied to the SDK boot directory. This firmware binary (BCM8806x) must be present in the SDK boot directory for loading during initialization.

Section 4.6.1: Status of supported features

Bring up and testing of the following BCM88060 features have been completed in this SDK release. Some features have reached Beta quality as they have already entered our test regression phase. Table 1 shows the status of features in this release.

Table 1: SDK Feature Status

Feature	Status	Note
Ethernet Retimer mode - Forced	Beta	Supported speeds: 10G / 20G / 25G / 40G / 50G / 100G
Ethernet Retimer mode - Autoneg	Beta	Supported speeds: 10G / 20G / 25G / 40G / 50G / 100G
Ethernet Repeater mode - Forced	Bringup	Supported speeds: 10G
Ethernet Gearbox mode - Autoneg/Forced	Beta	Supported 40G : System side 2x20G , line side 4x10G
Ethernet EBE (Extended Buffered for Ethernet mode)	Beta	Supported speeds: 10G / 20G / 25G / 40G / 50G / 100G Supports different speeds for line and system side, but lane counts should be same.
Warmboot	Beta	Tested in Linux environment with Ethernet and FCoE.
Ethernet Flexport	Beta	Quad, Tri0, Tri1, dual, single port mode supported.
HiGig	Bringup	Supported 42G
Boot optimization	Beta	Tested boot and initialization timings with multiple MT2 setup.
Parallel Download with multiple MT2	Beta	Tested with chained MT2 setup.
FCoE - Fixed mode and AN mode with TTS	Beta	Supported speeds: 4G / 8G / 16G / 32G.
FCoE - Flexport	Bringup	System side flex supported.
FCoE - Statistics and Events	Beta	Tested inband statistics in FCoE setup.

handling		
Diagnostics	Beta	Register read/write, dsc dump, eye scan, PRBS supported. Phy diag supports multistage loopback. Logging supported.
Traffic tests	Beta	TR19, TR72 supported.

Section 4.6.2: Notes and Known Limitations

- Tested with BCM956960 & BCM56850 SVK on PowerPC with VxWorks and Linux.
- Ethernet Repeater AN mode is not yet supported.
- Ethernet Repeater Forced only 10G supported.
- 106G HG not supported.
- Reset of port required after PRBS tests.
- FCoE COS to priority API not supported.

Section 4.7: BCM53400 (GREYHOUND) Updates

Section 4.7.1: Additional SKU support

This release adds GA support for derivative SKU BCM53369. This SKU was Preview level in SDK 6.5.4.

Section 4.8: BCM88470 (Qumran-AX) Family Beta Release

This release is a beta version for the BCM88470-Family product line, following previously released 6.5.4 Beta version.

The subsequent sections describe the features validated for this release, known issues and bring-up guidelines.

It is extremely important to review “Backward compatible important notes” section before starting the integration of the new release.

Note: IPsec SW package is available as a separate add-on upon request.

Section 4.8.1: Backward compatible important notes

See BCM88670-Family “Backward compatible important notes” section.

Section 4.8.2: Validated Features

Basic data path, connectivity and Traffic Management features:

- Register & Memory access including DMA
- Supported SKUs:
  - Qumran-AX: 88470, 88470P, 88471,88473
  - Kalia: 88476, 88476P
- Device core mode: dual-core symmetrical
- Interfaces:
  - SRAM, DDR4@1.6GHz
  - CPU RX/TX (packet DMA)
  - NIF 156.25MHz and 125MHz
  - NIF:
    - 10GE: KR/XFI, RXAUI, XAUI
    - 40GE: XLGE
    - 100GE
    - ILKN
      - Only 12.5G.
      - Eagle and Falcon on same port isn't allowed.
    - QSGMII
- Forwarding:
  - Unicast
  - Multicast
  - mirroring
- CosQ:
  - End-to-end scheduling
  - Ingress and Egress compensation
  - VOQ ingress queue creation. Mapping by-destination and by-flow-id
  - Tail Drop
  - Credit watch dog
  - Egress queuing: number of priorities per port: 1/2/8
  - Undersubscribed CoE (mapping multiple ports, no Flow Control)
- Counters:
  - SNMP counters
  - Internal (diagnostics) counters
- Packet forwarding - ITMH, Force-FWD
- Fabric connectivity with FE device (Kalia only)
- Link Bonding

Packet Processing:

- Diagnostics
- Port
- STG
- L2-Forwarding
- L2-Learning
- VLAN-Translation
- VLAN
- VSWITCH
- L3-IPv4
- L3-IPv6
- L3-Interface-RIF
- L3-Egress-ARP
- L3-Egress-FEC
- L3-ECMP
- L2VPN
- L3VPN
- MPLS
- VPLS
- QOS
- VPWS
- IP-Tunnel
- Protection (Ingress, FEC, Egress, 1:1, 1+1)
- VXLAN
- EVPN
- PON general feature except subnet anti-spoofing
- PON MAC limit
- TRILL
- 802.1BR
- ROO
- EVPN
- MIM/SPB (except for ROO MIM/SPB MC see Qumran-B0 changes section)
- Egress-FRR additional MPLS label (in both regular MPLS case and EVPN case)
- Egress MPLS additional labels (up to 6 MPLS labels same solution as Jericho)

Section 4.8.3: Packet Processing New Features Roadmap Plan

Feature/Bug fix	SDK JIRA	Roadmap item	Documentation	Timeframe & Notes
Bridging Into PWE Tagged Mode Tunnel with Service Delimiting Tags	SDK-92490	Native-VLAN-Editing: Bridging into VPLS tagged mode tunnel	8847X-AG302-R document, section Deeper Encapsulation Command Stack, example Bridging Into PWE Tagged Mode Tunnel with	6.5.5 (Done)

			Service Delimiting Tags	
IP UC Routing Into VXLAN Tunnel	--	ROO VXLAN	8847X-AG302-R document, section Deeper Encapsulation Command Stack, IP UC Routing Into VXLAN Tunnel	6.5.4 (Done)
Segment Routing, IP UC Routing Into Six Deep MPLS Tunnel	--	Egress MPLS additional labels (porting from Jericho/QMX) (see timeframe note)	8847X-AG302-R document, section Deeper Encapsulation Command Stack, Segment Routing, IP UC Routing Into Six Deep MPLS Tunnel (L3VPN Into Deep MPLS Stack)	Not planned, instead, SDK offers the same solution as 88670/88370, for more information see “Egress MPLS Additional Labels” section in document 88670-PG11X-R, porting is planned in 6.5.5 (Done)
Native Ethernet Encapsulation Stage	--	Native-VLAN-Editing: Bridging into VPLS tagged mode tunnel	8847X-AG302-R document, section ETPP Native Ethernet Encapsulation and Native Ethernet Editing Stage	6.5.4 (Done)
Native Ethernet VLAN Editing Stage {VSI, Global-OutLIF}	--	Native-VLAN-Editing: Bridging into VPLS tagged mode tunnel	8847X-AG302-R document, section Native Ethernet VLAN Editing Stage	6.5.4 (Done)
Native Ethernet VLAN Editing Stage {VSI, Global-OutLIF.Profile.Domain)	--	--	8847X-AG302-R document, section Native Ethernet VLAN Editing Stage	Not planned
Native Ethernet VLAN Editing Stage default EEDB entry	--	Native-VLAN-Editing: Bridging into VPLS	8847X-AG302-R document, section Native Ethernet	6.5.4 (Done). Note: Only 2 fixed options are

		tagged mode tunnel	VLAN Editing Stage	available in SDK for default EEDB entry.
Three Ethernet VLAN Tags Parsing	SDK-89417	3 Ethernet VLAN tags parsing	8847X-AG302-R document, section Three Ethernet VLAN Tags Parsing	6.5.5 (Done)
Support for Additional VLAN Tag TPIDs	--	Support for additional VLAN tag TPIDs	8847X-AG302-R document, section Support for Additional VLAN Tag TPIDs	6.5.4 (Done)
MPLS Termination After IP-Tunnel Termination	--	--	8847X-AG302-R document, section, MPLS Termination After IP-Tunnel Termination	Not planned
Ingress VLAN Edits PCP-DEI to Include Meter Result Information	SDK-73606	Ingress VLAN considers meter results	8847X-AG302-R document, section, Ingress VLAN Edits PCP-DEI to Include Meter Result Information	6.5.4 (Done)
TPID transparent EVE	SDK-73610	VSWITCH	8847X-PG1XX-R document, section Basic VLAN Translation	6.5.4 (Done)
Imposition of Special Labels	--	MPLS imposition enhancements	8847X-AG302-R document, section MPLS Imposition Enhancements	6.5.4 (Done)
Independent TTL and EXP Inheritance Models	--	MPLS imposition enhancements	8847X-AG302-R document, section MPLS Imposition Enhancements	6.5.4 (Done)
ITPP Network Headers Termination (Forwarding Copy)	--	None	8847X-AG302-R document, section, Ingress Transmit Packet Processor Enhancements	6.5.4 (Done)

ITPP Fallback2bridge	--	--	8847X-AG302-R document, section, Ingress Transmit Packet Processor Enhancements	Not planned (see errata issue EID#15 in 8847X-ES101-R)
KAPS Database Hit Bit per Entry	SDK-75711	KAPS DB hit bit per entry	8847X-AG302-R document, section, KAPS Database Hit Bit per Entry	6.5.7
Routing enablers first my-mac	--	L3 (v4, v6)	Will be documented in the next UM version (88470-PG1XX-R)	6.5.4 (Done), note that the feature changes the default behavior of QAX compared to Jericho (see “Backward compatible important notes”)
L2CP functionality per OutLIF	SDK-75719	VSWITCH	Will be documented in the next UM version (88470-PG1XX-R)	6.5.6
OAM MEP-DB - Additional data may store Loss Measurement (LM)/Delay Measurement (DM) statistics	--	LM/DM	88470-AG100-R, section Database Enhancements 8867X_8847X-AG10X-R section Adding Delay/Loss Measurement to Accelerated MEP	6.5.4 (Done)
OAM 48B flexible Maintenance Association Identifier (MAID)	SDK-98807	MAID 48B (New)	88470-AG100-R, section Database Enhancements	6.5.7
OAM Unicast continuity check message (CCMs)	--	--	88470-AG100-R, section Database Enhancements	Not planned
BFD jitter	--	--	88470-AG100-R, section New features	Not planned

BFD Authentication	--	--	88470-AG100-R, section New features	Not planned
OAM/BFD Additional TLVs on CCM/BFD packets	--	--	88470-AG100-R, section New features	Not planned
OAM/BFD On-Demand TX Machine	SDK-73623	On demand delay measurement, loss measurement (new), BFD in demand mode (New)	88470-AG100-R, section New features	6.5.8
OAM Delay Measurement Statistics Enhancements	--	--	88470-AG100-R, section New features	Not planned, SDK supports only two-way delay.
OAM Configurable Opcodes Have Their Own Maintenance Domain (MD) Levels	--	--	88470-AG100-R, section Bug Fixes and Improvements	Not planned. SDK supports AIS the same as Jericho solution. LCK is not planned.
OAM Punted Packets Include the Source of Failure	--	Punt packets	88470-AG100-R, section Bug Fixes and Improvements	6.5.4 (Done)
OAM RFC 6374	--	--	88470-AG100-R, section Bug Fixes and Improvements	Not planned
OAM Configurable Transmission Rates for Y.1731 messages	--	--	88470-AG100-R, section Bug Fixes and Improvements	Not planned. SDK solution is the same as Jericho (Static values, not configured)
OAM Hierarchical-LM per MD-level (2 levels) – B0 only	SDK-75703	Hierarichical LM: Per MD-Level (2 counters) (New)	TBD	6.5.8

OAM Hierarchical-LM per LIF (2 levels) – B0 only	SDK-75703	Hierarichical LM: Per LIF (New)	TBD	6.5.8
OAM RDI automatic assertion for multipoint services	SDK-75726	Automatic RDI asseration	TBD	6.5.5 (Done)
OAM separate report mode for LM and DM	SDK-88515	LM/DM	TBD	6.5.7
OAM Downmep injection new mode (Egress PP)	SDK-88994	LM/DM, Legacy Y.1731 over MPLS-TP (GAL), over PWE (ACH)	TBD	6.5.6
BFD VCCV Type 3	--	Legacy BFD over IPv4, over IPv4 over LSP, over PWE (ACH)	--	6.5.4 (Done)
BFD your-discr=0 trap	SDK-75728	Legacy BFD over IPv4, over IPv4 over LSP, over PWE (ACH)	--	6.5.6
Validity Checks May Be Disabled per MEP Profile	--	--	88470-AG100-R, section Bug Fixes and Improvements	Not planned, SDK does not expose this functionality explicit, but per feature requirement (for example 48B MAID will disable MAID verification)
Increase Punt profiles to 16	--	Punt packets	88470-AG100-R, section Database Enhancements	6.5.4 (Done)
Loss Measurement (LM) and Synthetic Loss Measurement (SLM) Coexist – per LIF decision	SDK-75710	LM or SLM per LIF basis (New)	88470-AG100-R, section Bug Fixes and Improvements	6.5.7

Section 4.8.4: Major Bug fixes

Packet Processing:

- OAM classifier configuration caused packets on lif hierarchy where the OAM endpoint is not on the innermost lif to not be counted on the OAM endpoint's counter
- See BCM88670-Family major bug-fixes section

Section 4.8.5: Known issues

Network Interface:

- Only the interfaces indicated above are validated. If any interface that is currently not supported is blocking initial bring-up, please consult with Broadcom AE.
- Auto-training and Auto-negotiation protocols are not validated at this stage
- SAT is not functional
- Control message of Link Bonding is not supported
- Policer/Meter is not functional , depended on CGM block which its work is not completed

Packet Processing:

- OAM/BFD statistics is not functional
- OAM – delay measurement using 1DM is not functional.
- OAM LB is not functional
- BFD echo is not functional
- OAM Y.1711 is not functional
- BFD IPv4 Single-hop detect multiplier check is not functional
- KBP is not functional
- Hashing (Configured,Flexible) is not functional
- L2GRE is not functional
- ECN-PP is not functional
- PON subnet anti-spoofing
- ROO over Port extender (802.1BR) is not functional

Section 4.8.6: BCM88470-Family B0 revision support

- BCM88470-Family B0 revision introduces bug fixes. For full information see:

  8847X-ES10X-R document (BCM88470 Device Errata), summary list
- Following is the list of the expected bug fixes and enhancements that require SDK support and its timeframe

Feature/Bug fix	SDK JIRA	Documentation	Timeframe	Comments
SPB/MIM ROO MC	SDK-92756	TBD	6.5.7	

Section 4.9: BCM88670 Family GA Release

This release is for the BCM88370-Family and BCM88670-Family product lines.

In the continued SDK support, all features introduced in SDK 6.5.4 are also supported in SDK 6.5.5.

The subsequent sections describe the increment in available features compared to 6.5.4, major bug-fixes and known issues.

It is extremely important to review "Backward compatible important notes" section before starting the integration of the new release.

Section 4.9.1: Backward compatible important notes:

- SW compatibility guidelines 6.4.X to 6.5.X:

  Preserving SW interface of BCM88670 device between SDK versions was a key consideration in 6.5.x device driver design. The SW interface was modified in the following cases: Improved SW design resulting in a more intuitive API, efficiency, significantly modified or extended device functionality.

  To assist the migration process from existing application configuring BCM88670 devices over 6.4.x SDK versions (6.4.10, 6.4.11), to an application configuring BCM88670 devices over 6.5.x SDK versions, a dedicated document was created: "BCM88670 BCM SDK Compatibility Guidelines 6.4.x to 6.5.x". We highly recommend going over the document. In case you don't find the document in docsafe, please approach your AE engineer to get a copy of it.

  Customers that migrate existing applications over 6.4.X releases that are prior to 6.4.10, are required to do two steps: first to go over the release notes of later 6.4.X releases up to 6.4.10 (including) and fit their application configuration accordingly and then start the migration process from 6.4.X to 6.5.X.

  ISSU is NOT supported from 6.4.X to 6.5.X.
- IPMC: The ipmc_vpn_lookup_enable SOC property isn't supported. This flag value will be assumed to be configured to 1 and the multicast entries TCAM<RIF,G,S> and LEM<RIF,G,S> aren't available.
- When adding a host entry for ROO to LEM, only even native ARP are allowed (HW limitation). The API now returns an error in case the native ARP is odd.
- Release is aligned with KBP-SDK 1.4.6 for external TCAM and KAPS.
- Scheduled warmboot is supported.
- Configuring the first or second ECMP hierarchy with the same polynomial as the LAG will fail with an error message.
- **Behavior of BCM_DNX_SUPPORT flag was changed.**
  Until version 6.5.4, using BCM_DNX_SUPPORT flag in Makefile (MAKE_LOCAL) compiled all DNX devices, both faps (88375_A0,88470_A0,88650_A0,88660_A0,88675_A0...) and fabric elements (88750_A0,88950_a0...).
  Start from 6.5.5, using BCM_DNX_SUPPORT flag will compile 88690_A0 (Jericho2) device family only.
  In case BCM_DNX_SUPPORT flag was used in Makefile (MAKE_LOCAL) prior to 6.5.5, in order to maintain same build behavior (i.e. compile all DNX devices), one need to remove BCM_DNX_SUPPORT from the Makefile, and add following 2 lines instead :

ALL_DPP_CHIPS = 1
ALL_DFE_CHIPS = 1

- Configuration example (config-sand.bcm) includes now also ILKN ports.
- In 6.5.3 and 6.5.4 SDKs, setting interface was immediately written to HW. However, this is a wrong behavior since it will cause HW combination of previous speed set with unmatched interface type.

  This was fixed to the following procedure:

  1) Save new interface type to software by bcm_port_interface_set.

  2) New speed will set HW with speed and with the saved interface type from step 1.

  Backward compatibility Implication: interface_type change will have no affect until speed_set is followed. If user was writing an application for SDK 6.5.3 or 6.5.4 that changes interface_type and not following by speed_set - in 6.5.5 and beyond he needs to follow with calling speed_set API.

Section 4.9.2: New Features

Basic data path, connectivity and Traffic Management features:

- ILKN non-consecutive lanes support
- LinkScan: The link status was only checked and determined by PHY status, which was not enough. Local/Remote fault checking has been added in Linkscan thread in this release.
- PHY MDIX: New API bcm_port_mdix_set is added to support external PHY MDIX configuration.

Packet Processing:

- UDP Tunnels support (from BCM88670-B0)
- Supporting OAM packets with OAM My-CFM MAC identical to L3 My-MAC
- Improved scaled OAM endpoint creation by dedicated calling sequence
- BFD endpoint period 0 will disable the transmission

High Availability:

- SER Interrupt: For some interrupts, the SER handling would do hard reset or software reset. In this release, an option is added to allow calling user callback instead of doing reset.
- SER Interrupt: In previous releases, if an interrupt occurred but not handled, no information was printed. In this release, interrupt info can be printed out for every interrupt routine.
- Exact Match Shadowing: SOC property exact_match_tables_shadow_enable supports a new value: 2 - Disabled for LEM, enabled for other exact match tables. Note that MAC table bulk operations are not supported when LEM shadow is enabled.

Diagnostics shell:

- PHY Diag: External phy and system interface type were not supported for command "phy diag <unit> dsc". In this release, new arguments are added to support external phy and interface. Command example, "phy diag <unit> dsc u=1 if=sys".

- PHY PRBS: PRBS for external PHY was not supported. In this release, this feature is added and new command is added. PHY unit, interface type and poly can be spcecified. Command example, "phy diag <port> prbs u=1 if=sys p=3".
- PHY Diag: Added PCS related commands:

"phy diag <port> pcs"

"phy diag <port> pcs topo"

"phy diag <port> pcs link"

"phy diag <port> pcs speed"

"phy diag <port> pcs aneg"

"phy diag <port> pcs tfc"

"phy diag <port> pcs antimers"

"phy diag <port> pcs state"

"phy diag <port> pcs debug"

Section 4.9.3: Major Bug fixes

The list below refers to major bugfixes, and does not provide a comprehensive coverage of various bugfixes on all levels.

Basic data path, connectivity and Traffic Management features:

- Fixes for credit request profiles:

1. The default application configures SCH to be SLOW (previously was LOW_DELAY).
2. The default application associates each port with its corresponding SLOW credit request profile (previously all ports associated with 10G_SLOW profile).
3. Introduce fine-tuned credit request thresholds in case of local switch (single device).

- Calling bcm_port_loopback_set(unit,port,BCM_PORT_LOOPBACK_PHY) on a disabled port of Falcon interface brought the port to be enabled. This issue is fixed.
- After the fix, ports status keeps no change during loopback setting.
- KBP: There was no re-transmission mechanism in KBP XPT layer in previous releases. Once transmission failed in XPT layer, it would return error to application layer. And KBP LUT initialization was not stable because of it. In this release, a re-transmission mechanism has been added into KBP XPT layer.
- EXT PHY: If external PHY AN result is 1G, system side was down. This caused traffic was blocked. This issue has been fixed in this release.
- EXT PHY: 40G ports could not be probed and loopback did not work. These isses are fixed. External PHY 40G mode works proprerly in this release.
- EXT PHY: The link status flipped when the external PHY worked on pcs repeater mode. This issue has been fixed in this release.

High Availability:

- Fixed incorrect caching settings for some talbes and registers

  Below tables are changed to be uncacheable:

  KAPS_TCM

  SCH_MEM_01F00000

  SCH_MEM_30000000

  ILKN_PMH_PORT_0_CPU_ACCESS

  ILKN_PMH_PORT_1_CPU_ACCESS

  ILKN_PML_PORT_0_CPU_ACCESS

  ILKN_PML_PORT_1_CPU_ACCESS

  Below tables are changed to be cacheable:

  PPDB_A_FEC_ENTRY_FORMAT_A

  PPDB_A_FEC_ENTRY_FORMAT_B

  PPDB_A_FEC_ENTRY_FORMAT_C

  PPDB_A_FEC_ENTRY_FORMAT_NULL

  PPDB_A_FEC_ENTRY_GENERAL

  PPDB_B_LARGE_EM_FORMAT_1

  PPDB_B_LARGE_EM_FORMAT_2

  PPDB_B_LARGE_EM_FORMAT_3_TYPE_0

  PPDB_B_LARGE_EM_FORMAT_3_TYPE_1

  PPDB_B_LARGE_EM_FORMAT_SLB_COUNTER

  PPDB_B_LARGE_EM_LEARN_FORMAT

  IPS_QPM_1_NO_SYS_RED
- bcm_petra_rx_init() might return error during parallel initialization. This issue is fixed.
- DRCx_PhyCdrAboveTh interrupt might be triggered during soft reset. This bug has been fixed.
- In previous releases, some cachable tables didn't initialize properly, led to SER recovery failed at the first time. In this release, this problem is fixed.

Packet Processing:

- OAM: standard MEP configuration can affect also default MEP configuration if both using same profile. Fix was provided to separate default-profile from other MEPs

- OAM: When updating lm_counter_base_id field of an up MEP using bcm_oam_endpoint_create, Both TX and RX counters get the same value instead of two consecutive values.
- MPLS: explicit push profile allocation using bcm_mpls_port_create was failing after all-zero push profile was allocated (ttl=exp=0, no CW)
- L2: Reduced the time performance of L2 traverse with delete operations when LEM shadowing was enabled
- OAM: when using LM-PCP endpoints counters allocated for ingress/egress LIF may not be correct: For Down MEPs, TX Packets with PCP n and RX packets with PCP (n+1) increment the same counter (for Up MEPs it was the other way around).
- OAM: Using bcm_oam_endpoint_action_set on endpoint of type MPLS / PWE and mpls_out_gport != gport can cause an error and crash
- MPLS: Memory leak issue has been resolved for PWE binded with MPLS entries.
- OAM packets transmitted to the passive side of down MEPs, with lower level of the MEP itself were not trapped
- OAM: first call to bcm_oam_delay_add() with REPORT_MODE flag set may result in other existing endpoints transmitting DMMs as well
- OAM: When using DMA events, event interface timeout was accidentally taken from soc property OAMP_FIFO_DMA_TIMEOUT instead of OAMP_FIFO_DMA_EVENT_INTERFACE_TIMEOUT
- Egress object (FEC) with l3 intf (RIF) larger of 4K are now available
- OAM: Setting SLM (calling bcm_oam_loss_add() with the SLR flag set), caused OAMP to reject the CCM packets. Therefore LOC events were not generated. SLM setting was fixed
- In case UDH is enabled in the system, OAM can support now also OAMP Generic Reply Messages for example LBM/LBR.
- OAM: When calling action set with a newly defined snoop for an Up MEP/MIP the snooped copy will arrive at it's destination with 2 sets of system headers. The outer set will state the destination of the snooped packet and the inner set will contain trapping information (source system port, in/outLIF, etc.) It is the user's responsibility to parse both system headers. If only one set is desired (with the trapping information) the user may use the trap code bcmRxTrapOamUpMEP3.
- MPLS: when using bcm_l3_egress_create api to create mpls push command (with mpls_action PUSH), delete and replace wasn't handling push profile allocation correctly.
- An issue where not all the FECs that were accessed returned an access indication using the bcm_l3_egress_get API was fix.

Section 4.9.4: Errata

The list below relates to major open bugs that are not resolved:

Basic data path, connectivity and Traffic Management features:

- PM4x10 WB isn’t supported.

Packet Processing:

- OAM UP-MEP default profiles (multiple egress default LIF-profiles) is not functional

Section 4.10: BCM88680 (Jericho+) Family GA Release

This release is a GA version for the BCM88680-Family product line, following previously released bring-up EA versions.

Section 4.10.1: Bring-up guidelines

1. Reference Documentation

   Multiple documents describing relevant HW and SW aspects are available, including:
2. SW Documentation
   - 88X7X-PG2xx: Traffic Manager Theory of Operation - This document describes theory of operation and provides driver reference for the BCM88670 device series.
   - 88680-PG1xx: BCM-API Packet Processing: Theory of Operation - This document describes BCM-API for the BCM88680 device packet processing capabilities, and how to configure it for networking applications.
   - BCM88670/BCM88680 Software Compatibility Guide - This document summarizes the differences in the BCM SDK software interface between BCM88670 and BCM88680 devices. This document is intended for customers using the BCM SDK to configure BCM88670 devices. The goal of this document is to assist the migration process from existing applications configuring BCM88670 devices to an application configuring BCM88680 devices.

3. HW Architecture specs:
   - 88670-AG2xx: Traffic Management Architecture - This document describes the BCM88670/6BCM88370 and BCM 88680 traffic management architecture and fabric adapter. It is intended for system architects and anyone else seeking an understanding of the features and capabilities that the BCM88680 traffic management architecture provides.
   - 8868X-AG1xx: BCM8868X Packet Processor Architecture Changes - This document is intended for system architects and anyone else seeking an understanding of the features and capabilities that the BCM8868x Packet processing architecture provides.

4. Diagnostics Shell

   The diagnostics commands below are useful for debugging. All the diagnostics are called from the BCM shell. If you type the diagnostic incorrectly, usage will be printed. The following diag shell commands are validated:
   - General

   diag count g - display a graphical representation of the device counters

   show counters: MIB counters

   clear counters
   - NIF/Ports:

   diag nif - display link status information.

diag port_db (diag port) - display ports information, mapping to PMs, channels etc.

- Packet DMA:

"tx" shell command

- PP:

diag pp

kbp kaps_show

diag dbal

5. Port Mapping

Core association must be defined upon initialization, using the new format of BCM8867x:

ucode_port.port<logical-port-id>.<unit> = <Interface_name>.<channel_num>:core_<core-id>.<tm- port-id>

6. DRAM Interface

It is recommended to start the bring-up in "SRAM-only" mode. Before trying to use DRAM. SOC properties: ext_ram_present=0 to work in SRAM-only mode.

- When working with DRAM, change ext_ram_present=<0|1|21|22|3> according to the number of DRAM interfaces - seed details in the UM (88x7x-PG2xx).
- When working with DRAM, please note that for each device, for the first time you need to run DRAM PHY calibration. Once calibration parameters are determined, they can be saved and restored upon later initializations to significantly reduce initialization time. See Driver Reference— DRAM PHY Tuning section in the UM for details.

7. Network Lane Swap and Polarity configuration

Network Interface

Lane swap and polarity - the direction of configuration for both RX and TX is from front panel to device. For example, for the swap configuration described below the correct swap-mapping is 0x0321.

Device lane0 -> front panel lane3

Device lane1 -> front panel lane0

Device lane2 -> front panel lane1

Device lane3 -> front panel lane2

8. Programmable ITMH

ITMH (Ingress Traffic Management Header) in 8867x format works.

Section 4.10.2: Validated Features

Basic data path, connectivity and Traffic Management features:

- Register & Memory access including DMA
- Interfaces:

- NIF:
  - 10GE: KR/XFI, RXAUI, XAUI
  - 40GE: XLGE
  - 100GE
  - ILKN
    - Only 12.5G.
    - Eagle and Falcon on same port isn't allowed.
  - QSGMII
- Forwarding:
  - Unicast
  - Multicast
  - mirroring

Packet Processing:
- L2
- L3
- MPLS
- STG
- VPLS
- VPWS
- VSWITCH, L2VPN
- ECMP (Configured-LB) only
- L3VPN
- QOS
- IP-Tunnels
- Protection
- MIM/SPB
- ROO
- VXLAN
- EVPN
- PON
- KBP NL12K
- Native-Link-layer
- MPLS push to swap
- OAM
- BFD

Section 4.10.3: New Features

Packet Processing:
- Added the ability to set a polynomial and a barrel shift for the second ECMP hierarchy in configured LB
- See BCM88670-Family New features section

Section 4.10.4: Known issues

Packet Processing:

- Flexible-LB is not functional
- OAM/BFD statistics is not functional
- BFD IPv4 Single-hop detect multiplier check is not functional
- ROO over Port extender (802.1BR) is not functional
- OAMP server is not functional
- SLM and LM per LIF base is not functional

Section 4.11: BCM88770 (FE3600) Release

The Broadcom BCM88770 (formerly named BCM88950) is the fourth generation in the Dune product line of Fabric Element (FE) devices.

This is a sustaining release for BCM88950 driver, with all major features supported.

4.11.1: Important Notes

- The default DCS thresholds were optimized: RCI, GCI, LLFC, Drop.

4.11.2: Major Bugfixes

- None

Section 4.12: BCM88270 (Qumran-UX) Release

This release is a preview version for the BCM88270-Family product line.

Section 4.13: BCM88660 (ARAD+), BCM88650 (ARAD) Release

This is a sustain release of BCM88660, BCM88650 driver, with all major features supported.

Section 4.13.1: Important notes

- See BCM88670-Family GA release section, important: SW compatibility guidelines 6.4.X to 6.5.X.

Section 4.13.2: Major Bugfixes

- None

Section 4.13.3: Errata

None

Section 4.14 New External PHY Support

Section 4.14.1 BCM82332 (Dino) Support

The BCM82332 is a low-power, low-latency PHY with optional gearbox, integrating CDR, adaptive equalizer, and preemphasis functions supporting 100 GbE/40 GbE/10 GbE/1GBASE-X applications. The BCM82332 is specifically designed to interoperate with devices using Flexport technology, including the BCM56860 Trident-II+. The BCM82332 has two main mission modes: retimer and gearbox. Retimer mode is primarily for 10 GbE, 40 GbE, and OLT4.10. In this mode, 12 lanes of 10 Gbps bidirectional data are retimed. 10 GbE, 40 GbE, and OTL4.10 protocols can be mixed in retimer mode. Gearbox mode adapts four lanes of 25 Gbps line-side data to 10 lanes of 10 Gbps system-side data (CAUI). Gearbox mode supports both 100 GbE and OTU4.

Features supported in this release and have been tested with the BCM56860 switch device:

Speeds/Modes
- 1G SR,KX,KR,CX,XFI,SFI
- 10G: SR, LR, ER, CR, CX, KR, XFI, SFI
- 40G: SR4, LR4, ER4, CR4, KR4, XLAUI, XLPPI
- 100G Gearbox: CAUI, VSR, CAC2C, CAC2M
- 100G Pass-through: SR10, LR10, CR10, KR10, CAUI, CAC2C, CAC2M, VSR

HIGIG
- 11G with KR, CR, XFI
- 42G with KR4, CR4, XLAUI
- 106G Gearbox: CAUI, KR4, CR4
- 106G Pass-through: CR10, KR10, CAUI

IEEE Modes
- 10G: KR, CR, XFI
- 40G: KR4, CR4
- 100G Gearbox: KR4,CR4,CAUI
- 100G Pass-through: CAUI,CR10,KR10

Broadcast firmware download

Diagnostics features such as PRBS, Eyescan, DSC, PCS link mon, BER on falcon lanes

Auto negotiation with 40G PT CR4, KR4, 100G PT CR10

Digital and Remote loopback.

FEC supported in 100G Gearbox mode.

Polarity, Power, Tx/Rx lane reset, Tx/Rx data path reset, Tx/Rx squelch.

PLL and AN interrupts.

Analog parameter configurations such as Tx FIR, driver current and DFE tap settings.

DFE mode configurations such as enabling/disabling of DFE, BR DFE, LP DFE.

GPIO pin configuration.

Flexport (Repeater and Retimer):

- 100G GB to 40G PT
- 100G GB to 10G PT
- 40G PT to 100G GB
- 10G PT to 100G GB
- 40G PT to 10G PT
- 10G PT to 40G PT
- 10G PT to 1G PT

Flexport (Repeater only):

- 100G PT to 40G PT
- 100G PT to 10G PT
- 40G PT to 100G PT
- 10G PT to 100G PT

Current device limitations in this release

- Warm boot is not yet supported.
- When Flexing from 100G to 10G, port counters cannot be cleared
- At 100G mode, ports are not coming up when system side lanes selected between 1 to 10 or 2 to 11.
- 100G PT per lane control on line side is not supported
- Digital loopback on line side is failing with 1G Falcon ports.
- 106G CR4 link training is not supported
- BER on Merlin lanes is not supported

Section 4.14.3: Interoperability testing for new PHY/switch combinations

The following switch and PHY combinations have been interoperability tested in the SDK 6.5.5 release. Below lists only the features that have been tested and supported in this release.

Section 4.14.3.1: BCM56160 and PHY BCM54292

Features supported:

- 10M/100M Forced Speed and Auto Negotiation modes
- 1G Auto-Negotiation mode only
- Loopback
  - PCS Loopback (10M)
  - PCS Loopback is not supported for 100M
  - PMD Loopback is not supported
- Diagnostics
  - PRBS, DSC dump, Eyescan and TX Squelch are not supported

Section 4.14.3.2: BCM56560/BCM56565/BCM56760 and PHY BCM84868

Features supported:

- 1G/2.5G/5G/10G Auto-Negotiation mode only

- 100M forced speed and Auto-Negotiation mode
- Loopback
    - PMD Loopback is not supported
    - PCS Loopback - not supported at 2.5G speed in this release
- Diagnostics
    - PRBS supported on system side only
    - DSC dump, TX Squelch and Eyescan are not supported

Section 4.14.3.3: BCM56560/BCM56565/BCM56760 and PHY BCM84888

Features supported:

- Bring-up support for 2.5G/5G Auto-Negotiation mode only
    - Limitation: ports take about 3 Seconds to link up
    - Loopback
        - PMD Loopback is not supported
        - PCS Loopback - not supported at 2.5G speed in this release.
- Diagnostics
    - PRBS supported on system side only
    - DSC dump, Line side PRBS, TX Squelch and Eyescan are not supported

Section 4.14.3.4: BCM56760 and PHY BCM54140

Features supported:

- 10M/100M/1G Copper
- 100M/1G Fiber
- 1G is supported in Auto-Negotiation mode only
- PCS Loopback
- Diagnostic
    - PRBS, DSC dump, Eyescan and TX Squelch are not supported
- Software support for EEE (Energy Efficient Ethernet) with BCM56760 switch is not supported in this release.
- Support for 1588 is not present in this release.

Section 4.14.3.5: BCM56760 and PHY BCM54380

Features supported:

- 10M/100M/1G Copper
    - 1G is supported in Auto-Negotiation mode only
- PCS Loopback
- Diagnostics - PRBS, DSC dump, Eyescan, TX Squelch not supported
- Software support for EEE (Energy Efficient Ethernet) with BCM56760 switch is not supported in this release.

Section 4.14.3.6: BCM88675 and PHY BCM84757

Features supported:

- 10G (Auto-Negotiation mode only), 1G
- PCS Loopback
- Diagnostics
    - Line side PRBS, DSC dump and Eyescan are not supported

Section 5: Things to note

This section lists items that require special attention that are new to this release. Please see prior 6.5.x release notes for a list of older items that should also be noted.

Section 5.1: ALERT ON FUTURE EXTERNAL PHY DRIVER SDK SUPPORT

Starting in CY3Q16, Broadcom will be providing only discrete drivers for new external PHY devices. Drivers for new external PHYs will not be integrated in the SDK. Broadcom will support standalone drivers for External PHYs which customers can integrate into their software stack. Legacy PHY driver code that has been tested against legacy switch devices will continue to be supported.

Section 5.2: End of maintenance for SDK 6.4.x

The SDK 6.4.11 release was the last official maintenance release under the SDK 6.4.x delivery stream. For new designs or software releases, customers should plan on moving to this or future 6.5.x releases.

Section 5.3: SBX Device Deprecation in 6.5.x

Starting with SDK 6.5.4, the following devices will no longer be built as part of the SDK releases going forward:

- BCM88020_A0, BCM88025_A0, BCM88030_A0, BCM88130_A0, BCM88230_A0

Section 5.4: Default config.bcm change from lossless to lossy

In this release, the default MMU setting has been changed from the standard lossless default to lossy in config.bcm for devices 56965, 56760, 56565, and 56560. Devices with GA support in prior SDK releases remain with the lossless default. This will be the direction going forward for all new devices, while legacy devices default will remain as is.

Section 5.5: Route Installation considerations on BCM56960 in ALPM mode

In BCM56960, due to the relatively small size of UFT and the design of sharing one UFT bucket between multiple pivots, the bcmSwitchObjectL3RouteV4RoutesMax, bcmSwitchObjectL3RouteV6Routes64bMax and bcmSwitchObjectL3RouteV6Routes128bMax in ALPM mode return the number of maximum possible routes that could be installed respectively. The actual number of routes that could be installed may be smaller than those objects.

Section 5.6: BCM56960 - Mix of Autoneg and Forced Speed Ports within a single Falcon Core is not supported

Due to a single PLL constraint inside the Falcon, a mix of both autoneg and forced speed on a single core is currently not supported. All the ports on a single core should be all in autoneg or forced speed mode. The first port of a core doing autoneg will disable the rest of the forced

speed ports on the same core. Once the first port on a core completes autoneg and the VCO is locked, the rest of ports on the same core can only do autoneg with the same VCO.

Section 5.7: Recent PHY interface library changes

The 100G interface for BCM56860, BCM86375, and BCM88560 devices required extensive changes to the interface library called "Portmod". Between SDK 6.5.2 and 6.5.3, BCM5686x operations for "Portmod" with BCM8279x family was verified and BCM8279x has achieved GA quality. For legacy 10G/40G devices, the "Portmod" changes required the shim layer to be created to enable the drivers to connect to the Falcon and Eagle cores. This work was completed in between SDK 6.5.2 and 6.5.3, thus making these legacy PHYs operational with the Portmod capable switches noted above.

Section 5.8: Internal loopback limitation on BCM848XX devices

For 84848, 84858, 84868, and 84888 PHY devices with AN enabled, internal loopback does not function unless port is cabled to a link partner to enable link up. Broadcom is working on a fix to resolve this issue in a future release.

Section 5.9: Engineering support statement for older XGS SDK releases

The following releases are now out of XGS engineering support since they are older than 12 months since release:

- SDK 6.4.x releases: 6.4.6, 6.4.5, 6.4.4, 6.4.3, 6.4.2, 6.4.1, 6.4.0
- All SDK 6.3.x and older releases

Service impacting defects will be reviewed and potentially addressed on these older releases. Support for non-service impacting defects will only be provided to customers running a SDK release that is less than 12 months old.

The following table shows the number of issues and improvements that have been added to our supported SDK releases by device over the past 12 months. While the table shows individual devices, many issues and improvements will apply to multiple products, e.g. BCM56850 and BCM56860, or all XGS products.

Table 2: Resolved issues and improvements per older XGS device family

Device	*Bugs resolved in the last 12 months*	*Improvements added in the past 12 months*
BCM56850 family	353	146
BCM56840_PLUS	60	10
BCM56340 family	72	25
BCM56640 family	120	35

BCM56450 family	191	56
BCM56150 family	22	2
BCM56440 family	55	18
BCM53440 family	61	26

Section 5.10: Warmboot Notes and Considerations

Section 5.10.1: Validated Warmboot upgrades

Following warmboot upgrades in the table below have been validated in this release.

Software Upgrade	*Supported*
6.4.11 to 6.5.5	Yes
6.5.4 to 6.5.5	Yes

Warmboot testing and issue resolution has focused on the following family of devices:

- BCM53400/BCM56060
- BCM56150
- BCM56220
- BCM56340
- BCM56440
- BCM56450
- BCM56640
- BCM56840
- BCM56850
- BCM56960
- BCM56260
- BCM56460
- BCM56965
- BCM56160
- BCM56760
- BCM56565
- BCM56560

Section 5.10.2: Determining warmboot scache requirements

The warmboot scache size requirements for a device for a particular release can be found by running the command "warmboot storage" at BCM prompt.

Section 5.10.3: Scache memory requirements for BCM56560, BCM56565, and BCM56760

To avoid warmboot failure due to out of memory, BCM56560, BCM56565 and BCM56760 devices require 4MB size of scache memory.

Section 5.10.4: BCM56820 upgrade recovery failures

For the BCM56820 family of devices, the dscp_map information was not synced correctly in 6.5.4 release. This can cause failures in recovery during Warmboot upgrade from 6.5.4 to 6.5.5. A patch from SDK-103830 should be applied to 6.5.4 to resolve this issue.

Section 5.10.5: Failures due to incorrect port and encap IDs when upgrading BCM56860 and BCM56960

Because the map between multicast next-hops maintained implicitly in SDK and L3 interface was not restored in the previous release, this info might be incorrect and cause issues after warm boot. This has been resolved in SDK-94122 and SDK-100058 in this release.

Section 5.10.6: BCM56960 downgrade considerations

Flex counters are created based on counter group mode. In SDK 6.5.5, 31 different group modes are supported while only 25 are supported in 6.4.11. During downgrade from 6.5.5 to 6.4.11, counter created with group mode id greater than 24 cannot be recovered since these are invalid group IDs in 6.4.11.

The method used to calculate the yellow/red cell limits was enhanced between 6.5.5 and 6.5.4. Customers may see an error during downgrade to 6.5.4 from 6.5.5 due to this change.

Section 5.10.7: External PHY driver considerations

During warmboot upgrade from SDK 6.5.4 to 6.5.5, For the 56860, 56565, 56760, and 56560 families of devices, the driver will not recognize the external PHY. The PHY can be recognized by reprobing the device using the `bcm_port_probe` API.

Section 5.10.8: Warmboot support for Embedded Applications

This section describes warmboot support and procedures for associated firmware supported in SDK 6.5.x. Below are the various embedded applications (FW) available and supported with SDK 6.5.x:

- BFD
- BHH LM/DM
- MPLS LM/DM
- ETH LM/DM
- Broadsync
- 1588/PTP

All applications support warmboot with the exception of 1588/PTP. Warmboot for the above apps follows the same process like any other module on SDK side. Eapps warmboot support is through Level-2 warmboot i.e. requires information to be stored in sCache.

SDK warmboot doesn't affect execution of embedded applications (except 1588/PTP firmware) on internal ARM cores and they keep running as is and maintain their state.

Warmboot procedures

- Users are expected to invoke "sc ControlSync=1" command in the BCM diag-shell to store the information in sCache
- The corresponding API is bcm_switch_control_set(int unit, bcm_switch_control_t type, int arg)
- Note: Typically, storing of information in sCache is required when there is a module or session or context creation on SDK side. Calling of the bcm_switch_control_set() API is left to the discretion of user/customer depending on their system requirements for example, after each session creation or module init or before the warmboot procedure initiation
- Free up the resources attached to the unit by calling below API equivalent of "exit clean" BCM shell command. i.e, "soc_shutdown (int unit)" Note: Broadsync warmboot procedure mandates usage of the above API. As part of API, SDK informs the embedded ARM core about the warmboot process initiation so that BroadSync FW waits for the SDK warmboot procedure to complete for further communication

Steps to follow during the SDK warmboot

Below is an example sequence of diag-shell commands executed (through reload.soc) as part of the SDK warmboot sequence for initializing SDK HOST and firmware communication

- MCSMsg INIT;
- MCSMsg 0;
- MCSMsg 1;
- "MCSMsg INIT" and "MCSMsg x" commands invokes the below APIs respectively:
- soc_cmic_uc_msg_start(int unit) /* Allocates Messaging resources/Mutexes */
- soc_cmic_uc_msg_uc_start(int unit, int uc) /* Starts Messaging with specific core.*/

Customers have to include the above calls in their SW.

Section 6: Summary of BCM API changes and enhancements

his section summarizes BCM API changes in this release. Complete documentation will be available in the Network Switching Software Programmer's Guide number 56XX-PG653-R.
For the full list of API support by Broadcom device, please reference the file SDK-6.5.x-Support-Matrix.xls in the sdk-all-6.5.x/RELDOCS directory in the release package.

Section 6.1: BFD

New BFD Tunnel type is added in this release.

Table 3: BFD Tunnel Types

bcm_bfd_tunnel_type_t	*Description*
bcmBFDTunnelTypeMplsPhp	Receive the packets IP+UDP+BFD, without mpls label

New BFD Event type is added in this release.

Table 4: BFD Event Types

Event type	*Description*
bcmBFDEventEndpoinUnExpectedMeg	Un-Expected Meg defect has been detected on an endpoint.
bcmBFDEventEndpoinUnExpectedMegClear	Un-Expected Meg defect has been cleared on an endpoint.

Section 6.2: Class of Service Queue Configuration

New cosq control types are added.

Table 5: CoSQ Control Type Values

Value	*Description*	*Arg value*
bcmCosqControlIngressPortDropTpid3	cosq ingress port drop global TPID 3.	
bcmCosqControlIngressPortDropTpid4	cosq ingress port drop global TPID 4.	
bcmCosqControlIngressPortDropIgnoreIpDscp	cosq ingress port drop global ignore IP DSCP.	
bcmCosqControlIngressPortDropIgnoreMplsExp	cosq ingress port drop ignore MPLS EXP.	
bcmCosqControlIngressPortDropIgnoreInnerTag	cosq ingress port drop ignore inner VLAN tag.	
bcmCosqControlIngressPortDropIgonreOuterTag	cosq ingress port drop ignore outer VLAN tag.	

New cosq threshold types are added.

Table 6: Threshold types

bcmCosqThresholdPacketDescriptorBuffers	consumed PDBs. Ingress Drop threshold, DP, at device level.
bcmCosqThresholdBundleDescriptorBuffers	consumed Bundle Descriptor Buffers. Ingress Drop threshold, DP, at device level.

New BST cosq IDs are added.

Table 7: BCM BST CoSQ IDs

flag	*Description*
bcmBstStatIdEgrPortPoolSharedUcast	Per port pool unicast
bcmBstStatIdEgrPortPoolSharedMcast	Per port pool multicast

New API bcm_cosq_classifier_id_get is defined.
New BCM_COSQ_INGRESS_PORT_DROP macros are added.

```
   #define BCM_COSQ_INGRESS_PORT_DROP_IP_MAP_KEY(dscp) ( ( (dscp) &
0xF) | 0x3000000 )
    #define BCM_COSQ_INGRESS_PORT_DROP_MPLS_MAP_KEY(exp) ( ( (exp) &
0x7) | 0x4000000 )
```

New BCM_COSQ_GPORT_SIZE flag is added.

Table 8: size flags

COSQ_GPORT_SIZE_PACKET_DESC	Configure packet description.

APIs for Section 6.2: Class of Service Queue Configuration

bcm_cosq_classifier_id_get

Gets the classifier id for the given profile.

SYNOPSIS

```
#include <bcm/cosq.h>
 int bcm_cosq_classifier_id_get(
     int unit,
     bcm_cosq_classifier_t *classifier,
     int array_count,
     bcm_cos_t *priority_array,
     bcm_cos_queue_t *cosq_array,
     int classifier_id);
```

PARAMETERS

unit	BCM device number
classifier	Classifier information
array_count	Number of elements allocated in priority_array and cosq_array
priority_array	Array of internal priorities
cosq_array	Array of CoS queues
classifier_id	(OUT) Classifier ID

DESCRIPTION

Get the classifier ID for a given profile if it is present else returns BCM_E_NOT_FOUND. User need to calls thebcm_cosq_classifier_id_get for the given profile before calling bcm_cosq_classifier_create. If bcm_cosq_classifier_id_get api returnsBCM_E_NOT_FOUND then only bcm_cosq_classifier_create is to be called to get the classifier_id, else the classifier id returned is to be used for mapping profiles and deleting profiles.

RETURNS

BCM_E_NONE

BCM_E_XXX

---

bcm_cosq_pfc_deadlock_recovery_event_register

bcm_cosq_pfc_deadlock_recovery_event_unregister

Register/Unregister pfc deadlock recovery callback in PFC deadlock feature

SYNOPSIS

```
#include <bcm/cosq.h>
 int bcm_cosq_pfc_deadlock_recovery_event_register(
     int unit,
     bcm_cosq_pfc_deadlock_recovery_event_cb_t callback,
     void *userdata);
 int bcm_cosq_pfc_deadlock_recovery_event_unregister(
     int unit,
     bcm_cosq_pfc_deadlock_recovery_event_cb_t callback,
     void *userdata);
```

PARAMETERS

unit	BCM device number
callback	(IN) callback function
userdata	(IN/OUT) user data

DESCRIPTION

Register/Unregister pfc deadlock recovery callback in PFC deadlock feature - Indicates if register more times,it will take the last one.

```
typedef int (*bcm_cosq_pfc_deadlock_recovery_event_cb_t)(
     int unit,
     bcm_port_t port,
```

```
            bcm_cos_queue_t cosq,
            bcm_cosq_pfc_deadlock_recovery_event_t recovery_state,
            void *userdata);
```

RETURNS

BCM_E_NONE

BCM_E_XXX

---

bcm_cosq_ingress_drop_control_frame_config_t_init

Initialize a CoSQ ingress drop control frame configuration structure.

SYNOPSIS

```
   #include <bcm/cosq.h>
    void
bcm_cosq_ingress_drop_control_frame_config_t_init(bcm_cosq_ingress_drop
_control_frame_config_t *config)
```

PARAMETERS

config	(IN/OUT) Control frame configuration

DESCRIPTION

Initialize bcm_cosq_ingress_drop_control_frame_config_t data structure.

RETURNS

None.

---

bcm_cosq_ingress_drop_control_frame_config_set

bcm_cosq_ingress_drop_control_frame_config_get

CoSQ ingress port drop properties to be recognized as control frames.

SYNOPSIS

```
   #include <bcm/cosq.h>
    int bcm_cosq_ingress_port_drop_control_frame_set(
        int unit,
        bcm_port_t port,
        uint32 flags,
        bcm_cosq_ingress_drop_control_frame_config_t
*control_frame_config)

    int bcm_cosq_ingress_port_drop_control_frame_get(
        int unit,
        bcm_gport_t port,
        uint32 flags,
        bcm_cosq_ingress_drop_control_frame_config_t
*control_frame_config)
```

PARAMETERS

unit	BCM device number
port	(IN) port
flags	(IN) flags
control_frame_config	(IN) (for "_set") properties to be recognized as control frames
control_frame_config	(OUT)(for "_get") properties recognized as control frames

DESCRIPTION

set/get cosq ingress port drop control frame properties.

Table 9: bcm_cosq_ingress_drop_control_frame_config_t

Name	*Purpose*
mac_da_val	MAC DA to recognize as Control Plane packet
mac_da_mask	MAC DA mask
ether_type_code	Eth type to recognize as Control Plane packet
ether_type_code_mask	Eth type code mask

RETURNS

BCM_E_NONE

BCM_E_XXX

bcm_cosq_ingress_port_drop_custom_ether_type_set

bcm_cosq_ingress_port_drop_custom_ether_type_get

configure cosq ingress port drop configurable Ether type to Ether type code.

SYNOPSIS

```
#include <bcm/cosq.h>
 int bcm_cosq_ingress_port_drop_custom_ether_type_set(
     int unit,
     bcm_port_t port,
     uint32 flags,
     uint32 ether_type_code,
     uint32 ether_type_val)

 int bcm_cosq_ingress_port_drop_custom_ether_type_get(
     int unit,
     bcm_port_t port,
     uint32 flags,
     uint32 ether_type_code,
```

```
uint32 *ether_type_val)
```

PARAMETERS

unit	BCM device number
port	(IN) port
flags	(IN) flags
ether_type_code	(IN) ether_type_code to be recognized by the configurable ether type.
control_frame_config	(IN) (for "_set") ether type to associate with the ether type code.
control_frame_config	(OUT)(for "_get") ether type to associate with the ether type code.

DESCRIPTION

set/get cosq ingress port drop configurable Ether type to Ether type code.

RETURNS

BCM_E_NONE

BCM_E_XXX

bcm_cosq_ingress_port_drop_default_priority_set

bcm_cosq_ingress_port_drop_default_priority_get

configure cosq ingress port drop default priority.

SYNOPSIS

```
#include <bcm/cosq.h>
 int bcm_cosq_ingress_port_drop_default_priority_set(
     int unit,
     bcm_port_t port,
     uint32 flags,
     uint32 default_priority)

 int bcm_cosq_ingress_port_drop_default_priority_get(
     int unit,
     bcm_port_t port,
     uint32 flags,
     uint32 *default_priority)
```

PARAMETERS

unit	BCM device number
port	(IN) port
flags	(IN) flags
default_priority	(IN) (for "_set") default priority.
default_priority	(OUT)(for "_get") default priority.

DESCRIPTION

set/get cosq ingress port drop default priority.

RETURNS

BCM_E_NONE

BCM_E_XXX

bcm_cosq_ingress_drop_flex_key_construct_id_t_init

Initialize a CoSQ ingress drop flex key construct id structure.

SYNOPSIS

```
#include <bcm/cosq.h>
 void
bcm_cosq_ingress_drop_flex_key_construct_id_t_init(bcm_cosq_ingress_dro
p_flex_key_construct_id_t *config)
```

PARAMETERS

config	(IN/OUT) Flex key construct id

DESCRIPTION

Initialize bcm_cosq_ingress_drop_flex_key_construct_id_t data structure.

RETURNS

None.

---

bcm_cosq_ingress_drop_flex_key_construct_t_init

Initialize a CoSQ ingress drop flex key construct structure.

SYNOPSIS

```
#include <bcm/cosq.h>
 void
bcm_cosq_ingress_drop_flex_key_construct_t_init(bcm_cosq_ingress_drop_f
lex_key_construct_t *config)
```

PARAMETERS

config	(IN/OUT) Flex key construct

DESCRIPTION

Initialize bcm_cosq_ingress_drop_flex_key_construct_t data structure.

RETURNS

None.

bcm_cosq_ingress_drop_flex_key_t_init

Initialize a CoSQ ingress drop flex key structure.

SYNOPSIS

```
#include <bcm/cosq.h>
 void
bcm_cosq_ingress_drop_flex_key_t_init(bcm_cosq_ingress_drop_flex_key_t
*config)
```

PARAMETERS

config	(IN/OUT) Flex key config

DESCRIPTION

Initialize bcm_cosq_ingress_drop_flex_key_t data structure.

RETURNS

None.

---

bcm_cosq_ingress_drop_flex_key_entry_t_init

Initialize a CoSQ ingress drop flex key entry structure.

SYNOPSIS

```
#include <bcm/cosq.h>
 void
bcm_cosq_ingress_drop_flex_key_entry_t_init(bcm_cosq_ingress_drop_flex_
key_entry_t *config)
```

PARAMETERS

config	(IN/OUT) Flex key entry

DESCRIPTION

Initialize bcm_cosq_ingress_drop_flex_key_entry_t data structure.

RETURNS

None.

---

bcm_cosq_ingress_port_drop_flex_key_construct_set

bcm_cosq_ingress_port_drop_flex_key_construct_get

set/get cosq ingress port drop flex key construct properties.

SYNOPSIS

```
#include <bcm/cosq.h>
 int bcm_cosq_ingress_port_drop_flex_key_construct_set(
     int unit,
     bcm_cosq_ingress_drop_flex_key_construct_id_t *key_id,
     uint32 flags,
     bcm_cosq_ingress_drop_flex_key_construct_t *flex_key_config)

 int bcm_cosq_ingress_port_drop_flex_key_construct_get(
     int unit,
     bcm_cosq_ingress_drop_flex_key_construct_id_t *key_id,
     uint32 flags,
     bcm_cosq_ingress_drop_flex_key_construct_t *flex_key_config)
```

PARAMETERS

unit	BCM device number
key_id	(IN) flex key unique information
flags	(IN) flags
flex_key_config	(IN) (for "_set") flex key information to construct the key.
flex_key_config	(OUT)(for "_get") flex key information to construct the key.

DESCRIPTION

set/get cosq ingress port drop flex key construct properties.

Table 10: COSQ_INGRESS_PORT_DROP_MAX_FLEX_KEY_FIELDS

Name	*Purpose*
COSQ_INGRESS_PORT_DROP_MAX_FLEX_KEY_FIELDS	max number of fields used to build a flex key.

key_id should be created from the following struct:

```
typedef struct bcm_cosq_ingress_drop_flex_key_construct_id_s {
     bcm_port_t port;   /* port */
 } bcm_cosq_ingress_drop_flex_key_construct_id_t;
```

flex key configuration should be created using the following struct:

```
typedef struct bcm_cosq_ingress_drop_flex_key_construct_s {
     uint32 array_size;                  /* number of elements in
the offset
                                            array */
     uint32
offset_array[BCM_COSQ_INGRESS_PORT_DROP_MAX_FLEX_KEY_FIELDS]; /* array
of offsets from which the flex
                                            key will be created */
 } bcm_cosq_ingress_drop_flex_key_construct_t;
```

RETURNS

BCM_E_NONE

BCM_E_XXX

bcm_cosq_ingress_port_drop_flex_key_entry_set

bcm_cosq_ingress_port_drop_flex_key_entry_get

set/get cosq ingress port drop flex key entry.

SYNOPSIS

```
#include <bcm/cosq.h>
 int bcm_cosq_ingress_port_drop_flex_key_entry_set(
     int unit,
     bcm_port_t port,
     uint32 flags,
     uint32 key_index,
     bcm_cosq_ingress_drop_flex_key_entry_t *flex_key_info)

 int bcm_cosq_ingress_port_drop_flex_key_entry_get(
     int unit,
     bcm_port_t port,
     uint32 flags,
     uint32 key_index,
     bcm_cosq_ingress_drop_flex_key_entry_t *flex_key_info)
```

PARAMETERS

unit	BCM device number
port	(IN) port
flags	(IN) flags
key_index	(IN) entry index
flex_key_info	(IN) (for "_set") flex key entry information.
flex_key_info	(OUT)(for "_get") flex key entry information.

DESCRIPTION

set/get cosq ingress port drop flex key entry. flex key should be created from the following struct:

```
typedef struct bcm_cosq_ingress_drop_flex_key_s {
     uint32 value;   /* flex key value */
     uint32 mask;    /* flex key mask */
 } bcm_cosq_ingress_drop_flex_key_t;
```

flex key entry info should be created using the following struct:

```
typedef struct bcm_cosq_ingress_drop_flex_key_entry_s {
     bcm_cosq_ingress_drop_flex_key_t ether_code; /* ether code
information for the flex key entry */
     uint32 num_key_fields;              /* number of elements in
the key_fields array */
     bcm_cosq_ingress_drop_flex_key_t
key_fields[BCM_COSQ_INGRESS_PORT_DROP_MAX_FLEX_KEY_FIELDS]; /* key
fields array to build the flex key entry */
     uint32 priority;                    /* priority to associate
with the flex key entry */
 } bcm_cosq_ingress_drop_flex_key_entry_t;
```

RETURNS

BCM_E_NONE

BCM_E_XXX

Section 6.3: Field Processor

New field control value is added.

Table 11: Field Control Values

Name	*Purpose*
bcmFieldControlLargeDirectLuKeyLength	Field processor large direct lookup key length. valid values are 14,15,16 or 17.

New fabric header is added.

Table 12: Fabric Headers

Name	*Purpose*
bcmFieldFabricHeaderEthernetlatency	regular Ethernet, stacking FTMH with OAM-TS-LATENCY header

APIs for Section 6.3: Field Processor

bcm_field_presel_create_stage_id

Create a preselection specification using a specific ID for a specific stage

SYNOPSIS

```
#include <bcm/field.h>

 int bcm_field_presel_create_stage_id(
     int unit,
     bcm_field_stage_t stage,
     bcm_field_presel_t presel_id);
```

PARAMETERS

unit	BCM device number

stage	Stage for the preselector
presel_id	presel ID

DESCRIPTION

Create a preselection specification for a preselector in a desired stage using a specific ID. The preselector indicates which packets are selected for this set.

To destroy the preselector, use bcm_field_presel_destroy API.

RETURNS

BCM_E_NONE	Operation completed successfully
BCM_E_INIT	BCM unit not initialized
BCM_E_EXISTS	Requested presel ID already exists
BCM_E_PARAM	*entry points to NULL
BCM_E_MEMORY	Memory allocation failure
BCM_E_RESOURCE	No unused entries available

bcm_field_group_ports_add

Add ports to the specified group

SYNOPSIS

```
#include <bcm/field.h>

int
bcm_field_group_ports_add(
    int unit,
    bcm_field_group_t group,
    bcm_pbmp_t pbmp);
```

PARAMETERS

unit	(IN) Unit number

group	(IN) Field Group ID
pbmp	(IN) Group Pbmp

DESCRIPTION

This API facilitates to add port/ports to an existing Per Port or Pbmp Field Group. The ports can be mentioned using pbmp parameter. Inports qualifier will be added by default to an existing Per Port or Pbmp Field Group. User has to update the qualifier for all required entries to add right ports they want.

RETURNS

int

---

bcm_field_group_ports_remove

Remove ports from the specified group

SYNOPSIS

```
#include <bcm/field.h>

int
bcm_field_group_ports_remove(
    int unit,
    bcm_field_group_t group,
    bcm_pbmp_t pbmp);
```

PARAMETERS

unit	(IN) Unit number
group	(IN) Field Group ID
pbmp	(IN) Group Pbmp

DESCRIPTION

This API facilitates to remove port/ports from an existing Per Port or Pbmp Field Group. The ports can be mentioned using pbmp parameter. Inports qualifier will be added by default to an existing Per Port or Pbmp Field Group. User has to update the qualifier for all required entries to remove right ports they want.

RETURNS

int

bcm_field_group_config_validate

To verify whether a group can be created with a given qset and mode.

SYNOPSIS

```
#include <bcm/field.h>

 int bcm_field_group_config_validate(
     int unit,
     bcm_field_group_config_t *group_config,
     bcm_field_group_mode_t *mode);
```

PARAMETERS

unit	(IN) Unit number
group_config	(IN) Field Group Config
mode	(OUT) Field Group Mode

DESCRIPTION

This API facilitates to verify whether a group can be created with a given qset and mode. If Single/Double/Triple is passed as mode, API will return SUCCESS when group can be created in that mode. Output parameter Mode will be updated with the corresponding mode. If Auto is passed as mode, API will return SUCCESS if group can be created in any of the standard modes like Single/Double/Triple. Output parameter Mode will be updated with the corresponding standard mode.

RETURNS

int

Section 6.4: Layer 3 Management

New L3 flags are added.

Table 13: BCM Layer 3 Flags2

Name	*Purpose*
BCM_L3_FLAGS2_SKIP_HIT_CLEAR	Skip hit clear when creating a FEC to enhance time performance.
BCM_L3_FLAGS2_ROUTE_NO_MOVE	A flag only to be used with BCM_L3_REPLACE flag for avoiding host entries move from the route table to the host table.
BCM_L3_FLAGS2_MPLS_PHP	Create an L3 egress object which is a member of an ECMP group. It should only be used in scenario where the destination is redirected from the L3 egress object to the ECMP group by IFP for mpls php action if not BOS.

Section 6.5: Link Bonding (LB)

New member variable is added in bcm_lb_modem_packet_config_t.

```
typedef struct bcm_lb_modem_packet_config_s {
     /* Link bonding format type */
     bcm_lb_format_type_t pkt_format;

     /* The destination MAC address in segment header */
     bcm_mac_t dst_mac;

     /* The source MAC address in segment header */
     bcm_mac_t src_mac;

     /* VLAN tag in segment header */
     uint16 vlan;
```

```
        /* If use_global_priority_map is set, then VLAN PCP is from global
        mapping. Otherwise, from VLAN in per modem configuration. */
        int use_global_priority_map;

} bcm_lb_modem_packet_config_t;
```

Section 6.6: MAC-in-MAC Management

New MAC-in-MAC VPN Config flags is added.

Table 14: MAC-in-MAC VPN Config Flags

Name	*Purpose*
BCM_MIM_VPN_EGRESS_SERVICE_TPID_REPLACE	SDTAG TPID control per Mac-in-Mac VPN, replaced TPID.

New MAC-in-MAC port flags are added.

Table 15: MAC-in-MAC Port Flags

Name	*Purpose*
BCM_MIM_PORT_EGRESS_SERVICE_PRI_REPLACE	Replace priority in SD-TAG on MIM egress.
BCM_MIM_PORT_EGRESS_SERVICE_TPID_REPLACE	Replace TPID in SD-TAG on MIM egress.
BCM_MIM_PORT_SHARE	Only to be used with BCM_MIM_PORT_TYPE_PEER flag for creating a share peer port.
BCM_MIM_PORT_PEER_NO_BTAG	Control whether add B-TAG for peer port traffic

New member variable egress_service_tpid is added in bcm_mim_vpn_config_t.

```
typedef struct bcm_mim_vpn_config_s {
     ....
     uint16 egress_service_tpid;     /* egress SD tag TPID */
```

```
        } bcm_mim_vpn_config_t;
```

Section 6.7: Mirroring

New Mirror destination flags are added.

Table 16: BCM Mirror Destination Flags

Name	*Description*
BCM_MIRROR_DEST_ID_SHARE	Share mirror destination id among multi ports
MIRROR_DEST_MTP_ADD	Add an MTP into a shared-id mirror destination
MIRROR_DEST_MTP_DELETE	Delete an MTP from a shared-id mirror destination
MIRROR_DEST_MTP_REPLACE	REPLACE an MTP of a shared-id mirror destination

Port Selection Criteria for mirroring.

```
   typedef enum bcm_mirror_psc_e {
        bcmMirrorPscNone = 0,       /* Invalid rtag. */
        bcmMirrorPscSrcMac = 1,     /* Source MAC address. */
        bcmMirrorPscDstMac = 2,     /* Destination MAC address. */
        bcmMirrorPscSrcDstMac = 3,  /* Source+dest MAC address. */
        bcmMirrorPscSrcIp = 4,      /* Source IP address. */
        bcmMirrorPscDstIp = 5,      /* Destination IP address. */
        bcmMirrorPscSrcDstIp = 6,   /* Source+dest IP address. */
        bcmMirrorPscCount = 7       /* Must be last. Not a usable value. */
    } bcm_mirror_psc_t;
```

New member variables are added in bcm_mirror_destination_t.

```
   struct bcm_mirror_destination_s {
        ...
        int stat_id2; /* stat_id2 for second counter pointer */
        ...
        bcm_mirror_psc_t rtag; /* specify RTAG HASH algorithm used for
shared-id mirror destination */
        uint8 df; /* Set the do not fragment bit of IP header in mirror
encapsulation */
    } bcm_mirror_destination_t;
```

Section 6.8: Network Interface Virtualization Management

New NIV egress object flag is added.

Table 17: NIV egress object flags

Name	Purpose
BCM_NIV_EGRESS_P_BIT_CLEAR	Clear the P bit for multicast packets

Section 6.9: OAM CSF (Client Signal Failure) objects

New OAM define is added.

Table 18: Miscellaneous OAM defines

Define Name	Description
BCM_OAM_MAX_NUM_TLVS_FOR_DM	Maximum number of DM TLVs

Table 19: OAM CSF Flag Definitions

Flag	Description
BCM_OAM_CSF_UPDATE	Update CSF settings for given endpoint

New CSF types defines are added.

```
    #define BCM_OAM_CSF_LOS         0x0        /* Loss of Signal. */
    #define BCM_OAM_CSF_FDI         0x1        /* Forward Defect
Indication. */
    #define BCM_OAM_CSF_RDI         0x2        /* Reverse Defect
Indication. */
    #define BCM_OAM_CSF_DCI         0x3        /* Defect Clear
Indication. */
```

New member variable is added in OAM endpoint info structure.

```
typedef struct bcm_oam_endpoint_info_s {
     ....
     bcm_oam_endpoint_t action_reference_id;
          /* Reference endpoint id whose actions
             will be used on the new created
             endpoint.
             Default value: BCM_OAM_ENDPOINT_INVALID. API call
             will set the parameter to
             BCM_OAM_ENDPOINT_INVALID */

 } bcm_oam_endpoint_info_t;
```

New OAM TLV object structure is defined.

```
/* OAM TLV object. */
 typedef struct bcm_oam_tlv_s {
     bcm_oam_tlv_pattern_type_t tlv_type;
     int tlv_length;                      /* Length of the value
field */
     uint32 four_byte_repeatable_pattern; /* used when Type = Data
(3) */
 } bcm_oam_tlv_t;
```

New OAM CSF objects and APIs are added.

```
/* OAM CSF (Client Signal Failure) object. */
 typedef struct bcm_oam_csf_s {
     uint32 flags;                       /* CSF flags. */
     uint32 type;                        /* CSF Type value in PDU.
*/
     bcm_oam_endpoint_t id;              /* Endpoint ID of Local
MEP. */
     int period;                         /* CSF Tx period in
milliseconds.
                                            Supported values are
1000 and 60000 */
     uint8 pkt_pri;                      /* Egress marking for
outgoing CSF
```

```
                                                    messages. */
        bcm_cos_t int_pri;                    /* Egress queuing for
outgoing CSF
                                                    messages */
        uint32
csf_tx_update_lm_counter_base_id[BCM_OAM_LM_COUNTER_MAX]; /* The base
index of the LM counter to
                                                    be updated by Tx CSF
packets */
        uint32 csf_tx_update_lm_counter_offset[BCM_OAM_LM_COUNTER_MAX];
/* Offset to the Base LM counter Id to
                                                    be incremented by Tx CSF
packets */
        uint8 csf_tx_update_lm_counter_size; /* Number of LM counters
to be
                                                    incremented by Tx CSF
packets */
    } bcm_oam_csf_t;
```

New OAM BHH endpoint/group fault flags are added.

Table 20: OAM BHH endpoint/group fault flags Definitions

Fault Flag	*Description*
BCM_OAM_ENDPOINT_BHH_FAULT_CSF_LOS	CSF Loss of Signal
BCM_OAM_ENDPOINT_BHH_FAULT_CSF_FDI	CSF Forward Defect Indication
BCM_OAM_ENDPOINT_BHH_FAULT_CSF_RDI	CSF Reverse Defect Indication

APIs for Section 6.9: OAM CSF (Client Signal Failure) objects

bcm_oam_csf_t_init

Initialize a bcm_oam_csf_t structure.

SYNOPSIS

```
#include <bcm/oam.h>
 void bcm_oam_csf_t_init(bcm_oam_csf_t *csf_ptr);
```

PARAMETERS

csf_ptr	(OUT) Pointer to OAM csf structure to initialize

DESCRIPTION

Initializes an OAM csf structure to default values. This function must be used to initialize any OAM csf structure before passing it to an API function.

bcm_oam_csf_add

Add an OAM csf object

SYNOPSIS

```
#include <bcm/oam.h>
 int
 bcm_oam_csf_add(
     int unit,
     bcm_oam_csf_t *csf_ptr);
```

PARAMETERS

unit	(IN) Unit number.
csf_ptr	(IN) The OAM csf object to add

DESCRIPTION

Add an OAM csf object.

RETURNS

BCM_E_NONE	Operation completed successfully
BCM_E_INIT	BCM unit's OAM subsystem not initialized
BCM_E_PARAM	Invalid parameter or NULL pointer parameter
BCM_E_XXX	

bcm_oam_csf_get

Get an OAM csf object

SYNOPSIS

```
#include <bcm/oam.h>
 int
 bcm_oam_csf_get(
     int unit,
     bcm_oam_csf_t *csf_ptr);
```

PARAMETERS

unit	(IN) Unit number.
csf_ptr	(OUT) The OAM csf object to get

DESCRIPTION

Get an OAM csf object.

RETURNS

BCM_E_NONE	Operation completed successfully
BCM_E_INIT	BCM unit's OAM subsystem not initialized
BCM_E_NOT_FOUND	Indicated entry does not exist or is not a looopback entry
BCM_E_PARAM	Invalid parameter or NULL pointer parameter

BCM_E_XXX	

bcm_oam_csf_delete

Delete an OAM csf object

SYNOPSIS

```
#include <bcm/oam.h>
 int
 bcm_oam_csf_delete(
     int unit,
     bcm_oam_csf_t *csf_ptr);
```

PARAMETERS

unit	(IN) Unit number.
csf_ptr	(IN) The OAM csf object to delete

DESCRIPTION

Delete an OAM csf object.

RETURNS

BCM_E_NONE	Operation completed successfully
BCM_E_INIT	BCM unit's OAM subsystem not initialized
BCM_E_NOT_FOUND	Indicated entry does not exist or is not a csf entry
BCM_E_PARAM	Invalid parameter or NULL pointer parameter
BCM_E_XXX	

Section 6.10: Port Configuration

New BCM_PORT_ADD_flag is added.

Table 21: Port add Flags

Flag	*Meaning*
BCM_PORT_ADD_TDM_PORT	Indicate a new TDM port.

New port class type is added.

Table 22: bcm_port_class_t

bcmPortClassVlanTranslateIngress	Class for ingress VLAN translation

New bcm_port_phy_control types are added.

Table 23: bcm_port_phy_control_t

BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE	configure the TX drivermode
BCM_PORT_PHY_CONTROL_FORCED_SPEED_LINE_SIDE	Set speed on line side of external PHY in case of EBE mode
BCM_PORT_PHY_CONTROL_EGRESS_FLOW_CONTROL_MODE	Supported flow modes are defined in in table BCM_PORT_PHY_CONTROL_FLOW_CONTROL_MODE_value
BCM_PORT_PHY_CONTROL_INGRESS_FLOW_CONTROL_MODE	Supported flow modes are defined in in table BCM_PORT_PHY_CONTROL_FLOW_CONTROL_MODE_value
BCM_PORT_PHY_CONTROL_PSM_COS_BMP	Set COS bit map for PSM (Packet Storage Modude) of an external PHY in EBE/EtherRAM mode.

New port phy control types are added.

Table 24: TX drivermode

Value	Meaning
BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_DEFAULT	default drivermode
BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_NOT_SUPPORTED	not supported drivermode
BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_HALF_AMP	half amp mode
BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_HALF_AMP_HI_IMPED	high impedence half amp mode

New flow control modes are defined.

Table 25: bcm_port_phy_control_flow_control_mode_t

bcmPortPhyControlFlowCtrlModeNone	No flow control mode (Relay mode).
bcmPortPhyControlFlowCtrlLegacyPauseEn	Lagacy Pause flow control mode.
bcmPortPhyControlFlowCtrlPfcPause	PFC flow control mode.
bcmPortPhyControlFlowCtrlLlfc	LLFC flow control mode.

Section 6.11: Precision Time Protocol

New enum type bcm_ptp_servo_lock_status_t is defined.

```
   typedef enum bcm_ptp_servo_lock_status_e {
        bcmPTPServoStatusUnlocked,    /* Servo in Unlocked State. */
        bcmPTPServoStatusFreqLocked,    /* Servo in Frequency locked State.
*/
        bcmPTPServoStatusPhaseLocked,    /* Servo in Phase Locked State. */
        bcmPTPServoStatusFreqPhaseLocked,    /* Servo in Phase and Frequency
Locked State. */
    } bcm_ptp_servo_lock_status_t;
```

New member variable is added in bcm_ptp_servo_status_t.

```
         typedef struct bcm_ptp_servo_status_s {
              ...
              bcm_ptp_servo_lock_status_t lock_status; /* Servo Locked Status
      */
          } bcm_ptp_servo_status_t;
```

New struct type bcm_ptp_servo_threshold_t is defined.

```
         /* PTP  Servo lock threshold structure. */
          typedef struct bcm_ptp_servo_threshold_s {
              int bcmPtpServoThresholdFreqLock, /* 5 (ppb) or value defined
      by customer */
              int bcmPtpServoThresholdFreqUnlock,  /* 10 (ppb) or value
      defined by customer */
              int bcmPtpServoThresholdPhaseLock,  /* 15 or 45 ns as per Class
      A/B */
              int bcmPtpServoThresholdPhaseUnlock, /* 20 or 50 ns  */
          } bcm_ptp_servo_threshold_t;
```

APIs for Section 6.11: Precision Time Protocol

bcm_ptp_servo_threshold_get

Get servo thresholds.

SYNOPSIS

```
#include <bcm/ptp.h>
 int bcm_ptp_servo_threshold_get (
     int unit,
     bcm_ptp_stack_id_t ptp_id,
     int clock_num,
     bcm_ptp_servo_threshold_t *threshold)
```

PARAMETERS

unit	(IN) Unit number.
stack_id	(IN) Stack identifier index.
clock_index	(IN) Output clock index.
threshold	(OUT) PTP servo threshold values

DESCRIPTION

Get servo error thresholds.

RETURNS

BCM_E_NONE	Operation completed successfully
BCM_E_xxx	Operation failed

bcm_ptp_servo_threshold_set

Set servo thresholds.

SYNOPSIS

```
#include <bcm/ptp.h>
 int bcm_ptp_servo_threshold_set (
     int unit,
     bcm_ptp_stack_id_t ptp_id,
     int clock_num,
     bcm_ptp_servo_threshold_t *threshold)
```

PARAMETERS

unit	(IN) Unit number.
stack_id	(IN) Stack identifier index.
clock_index	(IN) Output clock index.
threshold	(IN) PTP servo threshold values

DESCRIPTION

Set servo error thresholds.

RETURNS

BCM_E_NONE	Operation completed successfully
BCM_E_xxx	Operation failed

Section 6.12: Packet Transmit and Receive

New Rx trap codes are defined.

Table 26: Rx Trap Codes

Trap Code	*Description*
bcmRxTrapMplsTunnelTerminationTtl0	To be used upon independent configuration of MPLS tunnel termination TTL0 trap
bcmRxTrapMplsTunnelTerminationTtl1	To be used upon independent configuration of MPLS tunnel termination TTL1 trap
bcmRxTrapMplsForwardingTtl0	To be used upon independent configuration of MPLS forwarding TTL0 trap
bcmRxTrapMplsForwardingTtl1	To be used upon independent configuration of MPLS forwarding TTL1 trap

New member variables are added in bcm_rx_trap_config_t.

```
typedef struct bcm_rx_trap_config_s {
     ....
     int tunnel_termination_trap_strength;
     /* Strength of tunnel termination trap,
        strongest trap in processing holds. */

     int tunnel_termination_snoop_strength;
     /* Strength of tunnel termination trap,
     strongest snoop in processing holds. */
 } bcm_rx_trap_config_t;
```

New Rx trap flag is added.

Table 27: BCM RX Trap Flags

Name	*Purpose*
BCM_RX_TRAP_UPDATE_TUNNEL_TERMINATION_STRENGTH	update tunnel termination trap strengths

Rx MTU set flags are defined.

Table 28: MTU set flags

Mode	*Meaning*
BCM_RX_MTU_LIF	The input configuration is for LIF
BCM_RX_MTU_RIF	The input configuration is for RIF

New struct bcm_rx_mtu_config_t is defined.

```
typedef struct bcm_rx_mtu_config_s {
     uint32 flags;       /* BCM_RX_MTU_FLAG_XXX */
     bcm_if_t intf;      /* Interface  RIF */
     bcm_gport_t gport;  /* GPORT LIF */
     uint32 mtu;         /* MTU value to set for LIF/RIF */
 } bcm_rx_mtu_config_t;
```

New struct bcm_rx_cosq_mapping_t and APIs for Rx cosq mapping are added.

```
/* Rx cosq mapping. */
 typedef struct bcm_rx_cosq_mapping_s {
     uint32 flags;                   /* flags */
     int index;                      /* Index into COSQ mapping table
                                        (0 is highest match priority) */
     bcm_rx_reasons_t reasons;       /* packet reasons bitmap */
     bcm_rx_reasons_t reasons_mask;  /* mask for packet reasons bitmap */
     uint8 int_prio;                 /* internal priority value */
     uint8 int_prio_mask;            // mask for internal priority value
     uint32 packet_type;             /* packet type bitmap */
     uint32 packet_type_mask;        /* mask for packet type bitmap */
     bcm_cos_queue_t cosq;           /* cosq value */
 } bcm_rx_cosq_mapping_t;
```

APIs for Section 6.12: Packet Transmit and Receive

bcm_rx_cosq_mapping_t_init

Initialize a rx cosq mapping structure.

SYNOPSIS

```
   #include <bcm/rx.h>
    void bcm_rx_cosq_mapping_t_init(bcm_rx_cosq_mapping_t
*rx_cosq_mapping);
```

PARAMETERS

rx_cosq_ mapping	(OUT) Pointer to the rx cosq mapping structure.

DESCRIPTION

Initializes a rx cosq mapping structure to default values. This function should be used to initialize any rx cosq mapping structure prior to filling it out and passing it to an API function.

RETURNS

None.

bcm_rx_cosq_mapping_extended_set

Map packets to a CPU CoS queue with truncated packets or not.

SYNOPSIS

```
#include <bcm/rx.h>

 int bcm_rx_cosq_mapping_extended_set(
     int unit,
     uint32 options,
     bcm_rx_cosq_mapping_t *rx_cosq_mapping);
```

PARAMETERS

unit	(IN) Unit number.
options	(IN) Configuration options
rx_cosq_mapping	(IN) Rx cosq mapping structure

DESCRIPTION

Add Rx cosq mapping entry data.

RETURNS

BCM_E_NONE	Success
BCM_E_UNAVAIL	Feature unavailable
BCM_E_XXX	Error occurred

bcm_rx_cosq_mapping_extended_get

Get copy to Rx cosq mapping entry data.

SYNOPSIS

```
#include <bcm/rx.h>

int bcm_rx_cosq_mapping_extended_get(
    int unit,
    bcm_rx_cosq_mapping_t *rx_cosq_mapping);
```

PARAMETERS

unit	(IN) Unit number.
rx_cosq_mapping	(INOUT) Rx cosq mapping structure

DESCRIPTION

Get copy to Rx cosq mapping entry data.

RETURNS

BCM_E_NONE	Success
BCM_E_UNAVAIL	Feature unavailable
BCM_E_XXX	Error occurred

bcm_rx_mtu_set

Map LIF/RIF to to MTU value for MTU filtering

SYNOPSIS

```
#include <bcm/rx.h>

int bcm_rx_mtu_set(
  int unit,
  bcm_rx_mtu_config_t *config);
```

PARAMETERS

unit	(IN) Unit number.
config	(IN) Configuration MTU for LIF/RIF

DESCRIPTION

Map LIF/RIF to to MTU value for MTU filtering

RETURNS

BCM_E_NONE	Success
BCM_E_UNAVAIL	Feature unavailable
BCM_E_XXX	Error occurred

bcm_rx_mtu_get

Get the MTU value that was mapped to LIF for MTU filtering

SYNOPSIS

```
#include <bcm/rx.h>

int bcm_rx_mtu_get(
  int unit,
  bcm_rx_mtu_config_t *config);
```

PARAMETERS

unit	(IN) Unit number.
config	(IN) Configuration MTU for LIF/RIF

DESCRIPTION

Get the MTU value that was mapped to LIF/RIF for MTU filtering

RETURNS

BCM_E_NONE	Success
BCM_E_UNAVAIL	Feature unavailable
BCM_E_XXX	Error occurred

Section 6.13: Statistics

New enum values are added in bcm_stat_val_t.

```
typedef enum bcm_stat_val_e {
     snmpBcmTxE2ECCControlFrames,
     /* Total number of End to End Congestion
        Control(E2E-CC) Head of line(HOL)
        packets or DMVoQ flow control packets
        transmitted on each of the ports */
     snmpBcmE2EHOLDropPkts
     /* Total number of packets dropped due
        to E2EHOL (End to End Head of Line)
        drop status */
 } bcm_stat_val_t;
```

New counter config flag is added.

Table 29: Counter Config Flags

Flag	*Description*
BCM_STAT_COUNTER_MULTIPLE_SOURCES_PER_ENGINE	Connect 2 sources to one engine - for QAX OAM.

New member variables are added in bcm_stat structures.

```
/* A range of LIF IDs. */
 typedef struct bcm_stat_counter_lif_counting_range_s {
     ....
     int is_double_entry;
 } bcm_stat_counter_lif_counting_range_t;

 /* LIF Counting Source. */
 typedef struct bcm_stat_lif_counting_source_s {
     ....
    int stif_counter_id;
    uint32 *offset;
 } bcm_stat_lif_counting_source_t;
```

```
    /* stat counter value structure */
    typedef bcm_stat_counter_value_s {
        uint64 value; /* value of one statistic */
    } bcm_stat_counter_value_t;

    /* stat counter Input data in order to know which counter to read */
    typedef bcm_stat_counter_input_data_s{
        uint32 counter_source_id, /* counter source Id, (for example: global
lif, If the source is Lif) */
        bcm_stat_counter_source_type_t counter_source_type, /* type of source
to count */
        int command_id, /* command id */
        bcm_core_t core_id, /* core id to count from*/
        int nstat, /* size of statistics array to count */
    } bcm_stat_counter_input_data_t;

    /* stat counter output data */
    typedef bcm_stat_counter_output_data_s{
        bcm_stat_counter_value_t *value_arr; /* array with the values read
for each statistic */
    } bcm_stat_counter_output_data_t;
```

APIs for Section 6.13: Statistics

bcm_stat_counter_get

Get 64 bit counter values array for a multiple (or single if array size=1) statistic types belong to one source counter set.

SYNOPSIS

```
#include <bcm/stat.h>
 int bcm_stat_counter_get(
   int unit,
   uint32 flags,
   bcm_stat_counter_input_data_t* bcm_stat_counter_input_data,
   bcm_stat_counter_output_data_t* bcm_stat_counter_output_data);
```

PARAMETERS

unit	(IN) Unit number
flags	(IN) Flags
stat_counter_output_data	(OUT)
stat_counter_output_data	(OUT)

DESCRIPTION

Get 64 bit counter values array for a multiple (or single if array size=1) statistic types belong to one source counter set.

RETURNS

BCM_E_XXX

Section 6.14: Switch Control

New switch control types are added.

Table 30: Switch Type Values

Value	*Description*	*Arg Value*
bcmSwitchMacroFlowHigigTrunkHashMinOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Minumum offset value to be used in RTAG7 macroflow for Higig Trunk. The control set and getRTAG7_FLOW_BASED_HASH memory's SUB_SEL_HG_TRUNK andOFFSET_HG_TRUNK	0-99(BCM56860,BCM56960,BCM56450, BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchMacroFlowHigigTrunkHashMaxOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Maximum offset value to be used in RTAG7 macroflow for Higig Trunk. The control set and getRTAG7_FLOW_BASED_HASH memory's SUB_SEL_HG_TRUNK andOFFSET_HG_TRUNK	0-99(BCM56860,BCM56960,BCM56450, BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchMacroFlowHigigTrunkHashStrideOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Offset increment value to be used in RTAG7 macroflow for Higig Trunk. The control set and getRTAG7_FLOW_BASED_HASH memory's SUB_SEL_HG_TRUNK andOFFSET_HG_TRUNK	0-99(BCM56860,BCM56960,BCM56450, BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchMacroFlowTrunkHashMinOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Minumum offset value to be used in RTAG7 macroflow for Trunk. The control set and get RTAG7_FLOW_BASED_HASHmemory's SUB_SEL_TRUNK and OFFSET_TRUNK	0-99(BCM56860,BCM56960,BCM56450, BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchMacroFlowTrunkHashMaxOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Maximum offset value to be used in RTAG7 macroflow for Trunk. The control set and get RTAG7_FLOW_BASED_HASHme	0-99(BCM56860,BCM56960,BCM56450, BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)

	mory's SUB_SEL_TRUNK and OFFSET_TRUNK	
bcmSwitchMacroFlowTrunkHashStrideOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Offset increment value to be used in RTAG7 macroflow for Trunk. The control set and get RTAG7_FLOW_BASED_HASHmemory's SUB_SEL_TRUNK and OFFSET_TRUNK	0-99(BCM56860,BCM56960,BCM56450,BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchMacroFlowLoadBalanceEntropyHashMinOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Minumum offset value to be used in RTAG7 macroflow for Load Balance ID/Entropy. The control set and getRTAG7_FLOW_BASED_HASH memory's SUB_SEL_LBID_OR_ENTROPY_LABELand OFFSET_LBID_OR_ENTROPY_LABEL	0-99(BCM56860,BCM56960,BCM56450,BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchMacroFlowLoadBalanceEntropyHashMaxOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Maximum offset value to be used in RTAG7 macroflow for Load Balance ID/Entropy. The control set and getRTAG7_FLOW_BASED_HASH memory's SUB_SEL_LBID_OR_ENTROPY_LABELand OFFSET_LBID_OR_ENTROPY_LABEL	0-99(BCM56860,BCM56960,BCM56450,BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchMacroFlowLoadBalanceEntropyHashStrideOffset	BCM56860/BCM56960/BCM56450/BCM56640/BCM56340 Offset increment value to be used in RTAG7 macroflow for Load Balance ID/Entropy. The control set and getRTAG7_FLOW_BASED_HASH memory's SUB_SEL_LBID_OR_ENTROPY_LABELand OFFSET_LBID_OR_ENTROPY_LABEL	0-99(BCM56860,BCM56960,BCM56450,BCM56640,BCM56340),0-147 concat mode (BCM56860,BCM56960,BCM56450,BCM56640,BCM56340)
bcmSwitchUdpTunnelIPv4DstPort	UDP destination port for IPv4 encapsulation in UDP tunnel.	UDP destination port

bcmSwitchUdpTunnelIPv6DstPort	UDP destination port for IPv6 encapsulation in UDP tunnel.	UDP destination port
bcmSwitchUdpTunnelMplsDstPort	UDP destination port for MPLS encapsulation in UDP tunnel.	UDP destination port
bcmSwitchECMPSecondHierHash Offset	BCM88680: ECMP second hierarchy Hash key is barrel shifted by this value	
bcmSwitchECMPSecondHierHash Config	BCM88680: Selects hashing polynomial for the second hierarchy of the ECMP	
bcmSwitch3rdTagStampMode	Set the 3rd tag stamping mode, 0:Dsiable, 1: Enable.	TRUE/FALSE
bcmSwitchCrTransactionStart	If working in on demand CR, use this to start a new transaction.	TRUE/FALSE

New enum values are added in bcm_switch_pkt_trace_drop_reason_t.

```
typedef enum bcm_switch_pkt_trace_drop_reason_e {
     ....
     bcmSwitchPktTraceDropReasonStageLookup = 35,
     bcmSwitchPktTraceDropReasonStageIngress = 36,
     bcmSwitchPktTraceDropReasonCount = 37
 } bcm_switch_pkt_trace_drop_reason_t;
```

APIs for Section 6.14: Switch Control

bcm_switch_rov_get

An API to get the Recommended Operational Value for all FAPs and FEs by reading it from a register field and returning it as is.

SYNOPSIS

```
#include <bcm/switch.h>

 int bcm_petra_switch_rov_get(
     int unit,
     uint32 flags,
     uint32 *rov);
```

PARAMETERS

unit	(IN) BCM device number.
flags	(IN) Flags.
rov	(OUT) Recommend operational voltage.

DESCRIPTION

An API to get the Recommended Operational Value for all FAPs and FEs by reading it from a register field and returning it as is.

RETURNS

BCM_E_xxx

Section 6.15: UDF Resources Management

```
typedef struct bcm_udf_pkt_format_info_s {
     ....
     uint16 inner_protocol;          /* Inner protocol field in the
packet. Inner_ip should be set to BCM_PKT_FORMAT_IP_NONE to make this
take effect. IP-IN-IP type GRE should take original API sequence. */
     uint16 inner_protocol_mask;     /* Inner protocol mask */
 } bcm_udf_pkt_format_info_t;
```

Section 6.16: VLAN Management

Table 31: bcm_vlan_translate_key_t

bcmVlanTranslateKeyPortGroupDouble	Use Port-Group, O-VID[11:0] and I-VID[11:0]
bcmVlanTranslateKeyPortGroupOuter	Use Port-Group, O-VID[11:0]
bcmVlanTranslateKeyPortGroupInner	Use Port-Group, I-VID[11:0]
bcmVlanTranslateKeyPortGroupOuterTag	Use Port-Group, O-TAG[15:0]
bcmVlanTranslateKeyPortGroupInnerTag	Use Port-Group, I-TAG[15:0]
bcmVlanTranslateKeyPortGroupOuterPri	Use Port-Group, (VLAN-PRI, VLAN-CFI => O-TAG[15:12])

Section 7: Test Statistics

Section 7.1: How to read the Data

The tables below represent a spread of data gathered per-device, per-suite, per-release. The percentages represent the aggregate rate of failure for that suite when run against all variants of the family of devices.

The below data is not meant to be a precise indication of quality but instead serves as a guideline for improvements release-over-release. Additionally, although some cells show 0% failures, this does not necessarily mean the feature is supported in the device - tests are run to validate the appropriate SDK support even for unsupported features on older devices to ensure graceful handling of all API.

Finally, some devices have fewer columns listed if they were introduced recently.

Section 7.2: Overview

Each suite listed below is indicative of a specific module. Golden refers to a suite of tests that takes representation across multiple modules and serves as a sanity regression. Each suite contains tests of various types, loosely categorized as follows:

Test Categories	***Description***
Configuration Tests	Tests that verify that each API functions appropriately and can configure the device as expected.
Functionality Tests	Tests that further validate each of the API through functional use often requiring traffic to be run through the system.
Semantic Tests	Tests that ensure that the proper error handling mechanisms are working and users cannot crash the device through the API.

Section 7.2.1: Linux kernel versions used in this release

In SDK 6.5.5, the following Linux kernel versions were used in our development and regression cycles with these main CPUs:

- BCM9XLP208XMC (WRX): 3.10.59
- BCM958625XMC (RSX): 3.6.5
- BCM98548PPCXMC (GTO): 3.14
- BCM98548PPCXMC (XLR): 3.14

Please refer to the Broadcom Network Switching Software Platform Guide for more details about these CPUs.

Section 7.3: Total Tests

The below data represents the number of unique cases for each release. Note that although a particular test case will execute for each and every chip, it is only counted once.

Table 32: Tests per Module

	sdk-6.5.5	*sdk-6.5.4*	*sdk-6.5.3*
golden	153	154	154
warmboot	3004	2764	1242
auth	17	17	17
bfd	86	75	63
bhh	126	61	46
chip	9	9	9
coe	579	591	568
cosq	709	639	588
custom	7	7	7
ea	108	108	108
eav	19	19	19
extender	49	49	49
fabric	7	7	7
failover	10	10	10
fcoe	37	37	37
field	1435	1415	1404
higigproxy	129	129	129
infra	114	114	114
ipfix	17	17	17
ipmc	114	114	114
l2	337	337	337
l2gre	33	33	33
l3	525	522	506
l3.alpm	512	512	476
link	26	26	26
mim	46	46	46
mirror	173	173	173

misc	20	20	20
mpls	483	465	461
multicast	29	29	28
niv	66	65	65
oam	377	358	344
pkt	44	44	44
port	398	374	372
proxy	37	37	37
ptp	115	115	115
qos	13	13	13
rate	21	21	21
rtag7	43	43	43
rx	25	25	25
ser	157	157	157
stack	117	117	117
stat	392	386	386
stg	42	42	42
switch	197	197	197
time	33	33	33
tlvMsg	13	13	13
trill	47	47	47
trunk	232	223	223
tunnel	133	133	133
subport	31	31	31
vlan	240	240	239
vxlan	208	206	206
wlan	17	17	17
Test Suite Total	11911	11436	9728

Section 7.4: DVAPI Test Results

The tables below show percentages of failures for corresponding test suites per SDK release.

Section 7.4.1: All Devices

Note: This section represents aggregate results for all devices in the release.

	sdk-6.5.5	*sdk-6.5.4*	*sdk-6.5.3*
golden	0.1 %	0.1 %	0.2 %
warmboot	0.1 %	0.1 %	0.1 %
bcm.bfd	0.1 %	0.0 %	0.0 %
bcm.bhh	0.0 %	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	0.0 %
bcm.coe	0.1 %	0.2 %	0.1 %
bcm.cosq	0.1 %	0.2 %	0.1 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.2 %	0.2 %	0.2 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.5 %
bcm.ipmc	0.1 %	0.1 %	0.0 %
bcm.l2	0.1 %	0.1 %	0.1 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.1 %	0.1 %	0.1 %
bcm.l3.alpm	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.1 %	0.1 %	0.1 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.1 %	0.1 %	0.0 %

bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.1 %	0.1 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.4 %	0.4 %	0.5 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.1 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.1 %
bcm.stat	0.1 %	0.1 %	0.1 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.1 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.1 %	0.1 %	0.1 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.1 %	0.1 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.2 %	0.2 %	0.3 %

Section 7.4.2: Trident2

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***
golden	0.0 %	0.0 %	0.0 %

warmboot	0.1 %	0.1 %	0.1 %
bcm.bfd	0.0 %	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	0.0 %
bcm.cosq	0.0 %	0.0 %	0.0 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.1 %	0.1 %	0.1 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.1 %
bcm.l2	0.0 %	0.0 %	0.1 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.l3.alpm	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.0 %	0.0 %	0.1 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %

bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.0 %	0.0 %	0.0 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.1 %	0.1 %	0.1 %

Section 7.4.3: Triumph3

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***
golden	0.1 %	0.1 %	0.2 %
warmboot	0.0 %	0.0 %	0.1 %
bcm.bfd	0.0 %	0.0 %	0.0 %
bcm.bhh	0.0 %	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	0.0 %
bcm.cosq	0.0 %	0.0 %	0.0 %

bcm.custom	0.0 %	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.1 %	0.1 %	0.1 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	1.0 %	1.3 %	1.4 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %

bcm.stat	0.0 %	0.0 %	0.0 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.1 %	0.1 %	0.1 %

Section 7.4.4: Katana2

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***
golden	0.0 %	0.0 %	0.0 %
warmboot	0.1 %	0.0 %	0.1 %
bcm.bfd	0.0 %	0.0 %	0.0 %
bcm.bhh	0.0 %	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	0.0 %
bcm.coe	0.1 %	0.1 %	0.1 %
bcm.cosq	0.0 %	0.1 %	0.1 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %

bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.1 %	0.2 %	0.2 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.1 %	0.1 %	0.1 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.4 %	0.4 %	0.7 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.1 %	0.1 %	0.1 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %

bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.1 %	0.1 %	0.2 %

Section 7.4.5: Greyhound

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***
golden	0.0 %	0.0 %	0.3 %
warmboot	0.0 %	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	0.0 %
bcm.coe	0.0 %	0.0 %	0.0 %
bcm.cosq	0.0 %	0.0 %	0.0 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.0 %	0.0 %	0.0 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %

bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %	1.3 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.5 %	0.5 %	0.7 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.0 %	0.0 %	0.1 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.1 %	0.1 %	0.1 %

Section 7.4.6: Tomahawk

	sdk-6.5.5	*sdk-6.5.4*	*sdk-6.5.3*
golden	0.0 %	0.0 %	0.1 %
warmboot	0.1 %	0.1 %	0.1 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	0.0 %
bcm.coe	0.0 %	0.0 %	0.0 %
bcm.cosq	0.2 %	0.3 %	0.3 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.3 %	0.3 %	0.3 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.l3.alpm	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.1 %	0.1 %	0.4 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %

bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.0 %	0.1 %	0.1 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.0 %	0.0 %	0.0 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.2 %	0.2 %	0.2 %

Section 7.4.7: Trident2+

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***
golden	0.0 %	0.0 %	0.0 %
warmboot	0.0 %	0.1 %	0.1 %

bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cosq	0.0 %	0.0 %	0.0 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.0 %	0.1 %	0.1 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.l3.alpm	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.0 %	0.0 %	0.1 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %

bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.0 %	0.0 %	0.0 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.1 %	0.1 %	0.1 %

Section 7.4.8: Saber2

	sdk-6.5.5	*sdk-6.5.4*	*sdk-6.5.3*
golden	0.0 %	0.0 %	0.1 %
warmboot	0.0 %	0.0 %	0.1 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.coe	0.0 %	0.1 %	0.1 %
bcm.cosq	0.0 %	0.1 %	0.1 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.1 %	0.1 %	0.1 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %

bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.3 %	0.4 %	0.5 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.0 %	0.1 %	0.1 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %

Test Suite Total	0.1 %	0.1 %	0.1 %

Section 7.4.9: Hurricane3

	sdk-6.5.5	*sdk-6.5.4*	*sdk-6.5.3*
golden	0.0 %	0.0 %	0.0 %
warmboot	0.1 %	0.1 %	0.1 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cosq	0.0 %	0.0 %	0.0 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.0 %	0.0 %	0.0 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.2 %	0.2 %	0.5 %
bcm.proxy	0.0 %	0.0 %	0.0 %

bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.0 %	0.0 %	0.0 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.1 %	0.1 %	0.1 %

Section 7.4.10: Maverick

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***
golden	0.0 %	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.coe	0.0 %	0.0 %	0.0 %
bcm.cosq	0.2 %	0.0 %	0.0 %
bcm.custom	0.0 %	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %

bcm.fcoe	0.0 %	0.0 %	0.0 %
bcm.field	0.0 %	0.0 %	0.0 %
bcm.higigproxy	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.0 %
bcm.l2	0.0 %	0.0 %	0.0 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.1 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.0 %	0.0 %	0.1 %
bcm.proxy	0.0 %	0.0 %	0.0 %
bcm.ptp	0.0 %	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %	0.0 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	0.0 %	0.0 %	0.0 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %	0.0 %

bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.0 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.1 %	0.1 %	0.1 %

Section 7.4.11: Firebolt5

	sdk-6.5.5	*sdk-6.5.4*	*sdk-6.5.3*
golden	0.1 %	0.0 %	N/A
bcm.chip	0.0 %	0.0 %	N/A
bcm.coe	0.0 %	0.0 %	N/A
bcm.cosq	0.2 %	0.0 %	N/A
bcm.custom	0.0 %	0.0 %	N/A
bcm.eav	0.0 %	0.0 %	N/A
bcm.extender	0.0 %	0.0 %	N/A
bcm.fabric	0.0 %	0.0 %	N/A
bcm.failover	0.0 %	0.0 %	N/A
bcm.fcoe	0.0 %	0.0 %	N/A
bcm.field	0.1 %	0.0 %	N/A
bcm.higigproxy	0.0 %	0.0 %	N/A
bcm.ipfix	0.0 %	0.0 %	N/A
bcm.ipmc	0.0 %	0.0 %	N/A
bcm.l2	0.0 %	0.0 %	N/A
bcm.l2gre	0.0 %	0.0 %	N/A
bcm.l3	0.1 %	0.1 %	N/A
bcm.link	0.0 %	0.0 %	N/A
bcm.mim	0.0 %	0.0 %	N/A
bcm.mirror	0.0 %	0.0 %	N/A
bcm.misc	0.0 %	0.0 %	N/A
bcm.mpls	0.0 %	0.0 %	N/A
bcm.multicast	0.0 %	0.0 %	N/A

bcm.niv	0.0 %	0.0 %	N/A
bcm.oam	0.0 %	0.0 %	N/A
bcm.pkt	0.0 %	0.0 %	N/A
bcm.port	0.3 %	0.3 %	N/A
bcm.proxy	0.0 %	0.0 %	N/A
bcm.ptp	0.0 %	0.0 %	N/A
bcm.qos	0.0 %	0.0 %	N/A
bcm.rate	0.0 %	0.0 %	N/A
bcm.rtag7	0.0 %	0.0 %	N/A
bcm.rx	0.0 %	0.0 %	N/A
bcm.ser	0.0 %	0.0 %	N/A
bcm.stack	0.0 %	0.0 %	N/A
bcm.stat	0.0 %	0.0 %	N/A
bcm.stg	0.0 %	0.0 %	N/A
bcm.switch	0.0 %	0.0 %	N/A
bcm.time	0.0 %	0.0 %	N/A
bcm.trill	0.0 %	0.0 %	N/A
bcm.trunk	0.0 %	0.0 %	N/A
bcm.tunnel	0.0 %	0.0 %	N/A
bcm.subport	0.0 %	0.0 %	N/A
bcm.vlan	0.0 %	0.0 %	N/A
bcm.vxlan	0.0 %	0.0 %	N/A
bcm.wlan	0.0 %	0.0 %	N/A
Test Suite Total	0.1 %	0.2 %	N/A

Section 7.4.12: Apache

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***
golden	0.0 %	0.0 %	N/A
bcm.chip	0.0 %	0.0 %	N/A
bcm.coe	0.0 %	0.0 %	N/A
bcm.cosq	0.0 %	0.0 %	N/A

bcm.custom	0.0 %	0.0 %	N/A
bcm.eav	0.0 %	0.0 %	N/A
bcm.extender	0.0 %	0.0 %	N/A
bcm.fabric	0.0 %	0.0 %	N/A
bcm.failover	0.0 %	0.0 %	N/A
bcm.fcoe	0.0 %	0.0 %	N/A
bcm.field	0.0 %	0.0 %	N/A
bcm.higigproxy	0.0 %	0.0 %	N/A
bcm.ipfix	0.0 %	0.0 %	N/A
bcm.ipmc	0.0 %	0.0 %	N/A
bcm.l2	0.0 %	0.0 %	N/A
bcm.l2gre	0.0 %	0.0 %	N/A
bcm.l3	0.0 %	0.0 %	N/A
bcm.link	0.0 %	0.0 %	N/A
bcm.mim	0.0 %	0.0 %	N/A
bcm.mirror	0.0 %	0.0 %	N/A
bcm.misc	0.0 %	0.0 %	N/A
bcm.mpls	0.0 %	0.0 %	N/A
bcm.multicast	0.0 %	0.0 %	N/A
bcm.niv	0.0 %	0.0 %	N/A
bcm.oam	0.0 %	0.0 %	N/A
bcm.pkt	0.0 %	0.0 %	N/A
bcm.port	0.0 %	0.0 %	N/A
bcm.proxy	0.0 %	0.0 %	N/A
bcm.ptp	0.0 %	0.0 %	N/A
bcm.qos	0.0 %	0.0 %	N/A
bcm.rate	0.0 %	0.0 %	N/A
bcm.rtag7	0.0 %	0.0 %	N/A
bcm.rx	0.0 %	0.0 %	N/A
bcm.ser	0.0 %	0.0 %	N/A
bcm.stack	0.0 %	0.0 %	N/A
bcm.stat	0.0 %	0.0 %	N/A
bcm.stg	0.0 %	0.0 %	N/A

bcm.switch	0.0 %	0.0 %	N/A
bcm.time	0.0 %	0.0 %	N/A
bcm.trill	0.0 %	0.0 %	N/A
bcm.trunk	0.0 %	0.0 %	N/A
bcm.tunnel	0.0 %	0.0 %	N/A
bcm.subport	0.0 %	0.0 %	N/A
bcm.vlan	0.0 %	0.0 %	N/A
bcm.vxlan	0.0 %	0.0 %	N/A
bcm.wlan	0.0 %	0.0 %	N/A
Test Suite Total	0.1 %	0.1 %	N/A

Section 7.4.13: Tomahawk+

	sdk-6.5.5	*sdk-6.5.4*	*sdk-6.5.3*
golden	0.0 %	N/A	N/A
bcm.chip	0.0 %	N/A	N/A
bcm.coe	0.0 %	N/A	N/A
bcm.cosq	0.0 %	N/A	N/A
bcm.custom	0.0 %	N/A	N/A
bcm.eav	0.0 %	N/A	N/A
bcm.extender	0.0 %	N/A	N/A
bcm.fabric	0.0 %	N/A	N/A
bcm.failover	0.0 %	N/A	N/A
bcm.fcoe	0.0 %	N/A	N/A
bcm.field	0.1 %	N/A	N/A
bcm.higigproxy	0.0 %	N/A	N/A
bcm.ipfix	0.0 %	N/A	N/A
bcm.ipmc	0.0 %	N/A	N/A
bcm.l2	0.0 %	N/A	N/A
bcm.l2gre	0.0 %	N/A	N/A
bcm.l3	0.0 %	N/A	N/A
bcm.link	0.0 %	N/A	N/A
bcm.mim	0.0 %	N/A	N/A

bcm.mirror	0.0 %	N/A	N/A
bcm.misc	0.0 %	N/A	N/A
bcm.mpls	0.0 %	N/A	N/A
bcm.multicast	0.0 %	N/A	N/A
bcm.niv	0.0 %	N/A	N/A
bcm.oam	0.0 %	N/A	N/A
bcm.pkt	0.0 %	N/A	N/A
bcm.port	0.0 %	N/A	N/A
bcm.proxy	0.0 %	N/A	N/A
bcm.ptp	0.0 %	N/A	N/A
bcm.qos	0.0 %	N/A	N/A
bcm.rate	0.0 %	N/A	N/A
bcm.rtag7	0.0 %	N/A	N/A
bcm.rx	0.0 %	N/A	N/A
bcm.ser	0.0 %	N/A	N/A
bcm.stack	0.0 %	N/A	N/A
bcm.stat	0.0 %	N/A	N/A
bcm.stg	0.0 %	N/A	N/A
bcm.switch	0.0 %	N/A	N/A
bcm.time	0.0 %	N/A	N/A
bcm.trill	0.0 %	N/A	N/A
bcm.trunk	0.0 %	N/A	N/A
bcm.tunnel	0.0 %	N/A	N/A
bcm.subport	0.0 %	N/A	N/A
bcm.vlan	0.0 %	N/A	N/A
bcm.vxlan	0.0 %	N/A	N/A
bcm.wlan	0.0 %	N/A	N/A
Test Suite Total	0.2 %	N/A	N/A

Section 7.4.14: Metrolite

	sdk-6.5.5	***sdk-6.5.4***	***sdk-6.5.3***

golden	0.0 %	N/A	N/A
bcm.chip	0.0 %	N/A	N/A
bcm.coe	0.1 %	N/A	N/A
bcm.cosq	0.0 %	N/A	N/A
bcm.custom	0.0 %	N/A	N/A
bcm.eav	0.0 %	N/A	N/A
bcm.extender	0.0 %	N/A	N/A
bcm.fabric	0.0 %	N/A	N/A
bcm.failover	0.0 %	N/A	N/A
bcm.fcoe	0.0 %	N/A	N/A
bcm.field	0.0 %	N/A	N/A
bcm.higigproxy	0.0 %	N/A	N/A
bcm.ipfix	0.0 %	N/A	N/A
bcm.ipmc	0.0 %	N/A	N/A
bcm.l2	0.0 %	N/A	N/A
bcm.l2gre	0.0 %	N/A	N/A
bcm.l3	0.0 %	N/A	N/A
bcm.link	0.0 %	N/A	N/A
bcm.mim	0.0 %	N/A	N/A
bcm.mirror	0.0 %	N/A	N/A
bcm.misc	0.0 %	N/A	N/A
bcm.mpls	0.0 %	N/A	N/A
bcm.multicast	0.0 %	N/A	N/A
bcm.niv	0.0 %	N/A	N/A
bcm.oam	0.0 %	N/A	N/A
bcm.pkt	0.0 %	N/A	N/A
bcm.port	0.3 %	N/A	N/A
bcm.proxy	0.0 %	N/A	N/A
bcm.ptp	0.0 %	N/A	N/A
bcm.qos	0.0 %	N/A	N/A
bcm.rate	0.0 %	N/A	N/A
bcm.rtag7	0.0 %	N/A	N/A
bcm.rx	0.0 %	N/A	N/A

bcm.ser	0.0 %	N/A	N/A
bcm.stack	0.0 %	N/A	N/A
bcm.stat	0.0 %	N/A	N/A
bcm.stg	0.0 %	N/A	N/A
bcm.switch	0.0 %	N/A	N/A
bcm.time	0.0 %	N/A	N/A
bcm.trill	0.0 %	N/A	N/A
bcm.trunk	0.0 %	N/A	N/A
bcm.tunnel	0.0 %	N/A	N/A
bcm.subport	0.0 %	N/A	N/A
bcm.vlan	0.0 %	N/A	N/A
bcm.vxlan	0.0 %	N/A	N/A
bcm.wlan	0.0 %	N/A	N/A
Test Suite Total	0.2 %	N/A	N/A

Section 7.5: PHY Test Results

Starting with SDK 6.5.2, we have increased our focus and coverage on testing specific PHY and switch combinations in order to improve our quality in this area. The tables below represent specific results from our interoperability and regression testing for the release. We have been continually working to improve our results and coverage in ongoing releases.

Section 7.5.1: SQA External PHY

Switch Device	*External Phy Device*	*Total Tests*	*% Fail*

56639_A0	phy8747_phy8728__10G	198	7.58%
56846_A0	Warpcore_40G/20G/10G	198	1.52%
56850_A2	phy84856_phy84858_10G	198	3.54%
56960_A0	100GFuria	198	1.01%
56960_A0	10GSesto	198	2.02%
56860_A1	phy84856_phy84858_10G	198	1.01%
56860_A0	10GQuadra28	198	0.00%
56860_A0	40GQuadra28	198	0.00%
56860_A0	100GSesto	198	4.55%
56867_A1	phy82764_10G	198	2.02%
56867_A1	phy82764_40G	198	1.52%
56960_B0	phy82864_100G	198	1.52%
56960_B0	phy82864_100G_alt	198	3.03%
56960_B0	phy82864_40GPt	198	0.51%
56960_B0	phy82864_40GPt_alt	198	2.02%
56960_B0	phy82864_10G	198	5.05%
56960_B0	phy82864_10G_alt	198	5.56%
56960_B0	phy82864_11G	198	2.53%
56960_B0	phy82864_11G_alt	198	3.03%
56960_B0	phy82864_42GPt	198	0.51%
56960_B0	phy82864_42GPt_alt	198	2.02%
56960_B0	phy82864_106G	198	1.01%
56960_B0	phy82864_106G_alt	198	2.02%
56960_B0	phy82864_40G2x20	198	4.55%
56960_B0	phy82864_40G2x20_alt	198	5.05%
56960_B0	phy82864_42G2x20	198	6.06%

56960_B0	phy82864_42G2x20_alt	198	3.03%
56960_B0	phy82864_40GMux	198	4.55%
56960_B0	phy82864_50G	198	1.01%
56867_A1	phy82332_100GPt	198	1.52%
56867_A1	phy82332_100G_gbox	198	13.13%
56867_A1	phy82332_106GPt	198	1.01%
56867_A1	phy82332_40G	198	4.04%
56867_A1	phy82332_42G	198	3.03%
56867_A1	phy82332_10G	198	5.56%
56867_A1	phy82332_11G	198	9.09%
56860_A1	phy82332_retimer_1G	198	1.52%
56860_A1	phy82332_retimer_10G	198	8.59%
56860_A1	phy82332_retimer_11G	198	9.60%
56860_A1	phy82332_retimer_40G	198	3.03%
56860_A1	phy82332_retimer_100G	198	13.13%
56765_A0	phy54140_1G_Copper	198	0.51%
56765_A0	phy54140_1G_Fiber	198	0.51%
56565_A0	phy84868_10G	198	0.51%
56160_B0	phy54292_1G	198	1.01%
56960_B0	phy82864_40GMux	147	5.44%

Section 7.5.2: Interop External PHY

Section 7.5.2.1: P2P Suite

Switch Device	*Port Macro*	*Total Tests*	*% Fail*

56560_A0	HG_G40_84328_42G	47	14.89%
56560_A0	XE_G40_84328_10G	132	8.33%
56560_A0	XE_G40_84328_40G	112	8.04%
56560_A0	XE_MT_84757_10G	49	2.04%
56860_A1	XE_QD28_82780_40G	133	3.01%
56860_A1	XE_QD28_82780_10G	167	1.20%
56860_A1	XE_SESTO_82764_40G_MUX	113	15.04%
56860_A1	CE_SESTO_82792_100G_343	160	17.50%
56860_A1	XE_SESTO_82764_40G_PT	196	3.06%
56860_A1	XE_SESTO_82764_10G_PT	180	1.11%
56860_A0	CE_DINO_82332_40G_10G_MIXEDMODE	270	0.00%
56860_A0	CE_DINO_82332_100G_PT_L1_10	192	0.00%
56860_A0	HG_DINO_82332_106G_GB	61	4.92%
56860_A0	CE_DINO_82332_100G_PT	252	0.00%
56860_A0	HG_DINO_82332_11G	90	15.56%
56860_A0	HG_DINO_82332_42G	84	11.90%
56860_A0	CE_DINO_82332_100G_PT_L2_11	192	0.00%
56860_A0	HG_DINO_82332_106G_PT	66	0.00%
56860_A0	XE_DINO_82332_10G	434	0.46%
56860_A0	CE_DINO_82332_100G_GB	267	10.11%
56860_A0	XE_DINO_82332_40G	143	10.49%
56860_A1	XE_G40_84328_10G	139	4.32%
56860_A1	XE_G40_84328_40G	113	2.65%
56860_A0	XE_G28_82322_10G	148	14.19%
56860_A0	XE_G28_82322_40G	122	1.64%
56860_A0	XE_54210	38	0.00%
56960_B0	HG_MADURA_82864_42G_PT	90	0.00%

56960_B0	CE_MADURA_82864_100G_ALT	216	0.00%
56960_B0	XE_MADURA_82864_10G	162	0.00%
56960_B0	HG_MADURA_82864_42G_DUAL_ALT	90	0.00%
56960_B0	XE_MADURA_82864_40G_PT_ALT	162	0.00%
56960_B0	XE_MADURA_82864_25G	78	5.13%
56960_B0	HG_MADURA_82864_42G_PT_ALT	90	0.00%
56960_B0	XE_MADURA_82864_50G	90	0.00%
56960_B0	XE_MADURA_82864_40G_MUX	162	0.00%
56960_B0	HG_MADURA_82864_11G	90	0.00%
56960_B0	HG_MADURA_82864_106G_ALT	108	1.85%
56960_B0	XE_MADURA_82864_50G_ALT	90	0.00%
56960_B0	HG_MADURA_82864_42G_MUX	90	0.00%
56960_B0	XE_MADURA_82864_40G_DUAL	138	3.62%
56960_B0	HG_MADURA_82864_42G_DUAL	66	12.12%
56960_B0	XE_MADURA_82864_25G_ALT	90	8.89%
56960_B0	HG_MADURA_82864_11G_ALT	90	0.00%
56960_B0	XE_MADURA_82864_10G_ALT	162	0.00%
56960_B0	XE_MADURA_82864_40G_DUAL_ALT	162	0.00%
56960_B0	HG_MADURA_82864_106G	68	17.65%
56960_B0	CE_MADURA_82864_100G	178	6.18%
56960_B0	XE_MADURA_82864_40G_PT	162	0.00%
56960_B1	CE_MADURA_82864_100G_RETIMER_ULL	173	2.89%
56960_B1	XE_MADURA_82864_25G_ALT_RETIMER	90	8.89%
56960_B1	XE_MADURA_82864_40G_DUAL_ALT_RETIMER	138	11.59%
56960_B1	XE_MADURA_82864_40G_PT_ALT_RETIMER	162	0.00%
56960_B1	XE_MADURA_82864_10G_RETIMER_ULL	162	0.00%

56960_B1	HG_MADURA_82864_42G_PT_ALT_RETIMER	90	0.00%
56960_B1	XE_MADURA_82864_10G_RETIMER	162	0.00%
56960_B1	HG_MADURA_82864_106G_ALT_RETIMER	48	0.00%
56960_B1	XE_MADURA_82864_50G_RETIMER	90	2.22%
56960_B1	HG_MADURA_82864_11G_ALT_RETIMER	90	0.00%
56960_B1	HG_MADURA_82864_106G_RETIMER	80	12.50%
56960_B1	CE_MADURA_82864_100G_ALT_RETIMER	216	0.00%
56960_B1	HG_MADURA_82864_11G_RETIMER	90	0.00%
56960_B1	XE_MADURA_82864_50G_ALT_RETIMER	90	1.11%
56960_B1	XE_MADURA_82864_25G_RETIMER	90	10.00%
56960_B1	HG_MADURA_82864_42G_PT_RETIMER	90	0.00%
56960_B1	HG_MADURA_82864_42G_DUAL_ALT_RETIMER	90	0.00%
56960_B1	XE_MADURA_82864_40G_MUX_RETIMER	162	0.00%
56960_B1	XE_MADURA_82864_40G_DUAL_RETIMER	138	5.07%
56960_B1	CE_MADURA_82864_100G_RETIMER	176	3.41%
56960_B1	XE_MADURA_82864_10G_ALT_RETIMER	162	0.00%
56960_B1	HG_MADURA_82864_42G_DUAL_RETIMER	66	12.12%
56960_B1	XE_MADURA_82864_40G_PT_RETIMER_ULL	153	0.00%
56960_B1	XE_MADURA_82864_40G_PT_RETIMER	162	0.00%
56960_A0	XE_MADURA_82864_40G_PT	93	0.0%

Section 7.5.2.2: Loopback Suite

Switch Device	*Port Macro*	*Total Tests*	*% Fail*

56560_A0	HG_G40_84328_42G	25	40.00%
56560_A0	XE_G40_84328_10G	10	0.00%
56560_A0	XE_G40_84328_40G	20	20.00%
56560_A0	XE_MT_84757_10G	4	0.00%
56860_A1	XE_QD28_82780_40G	33	15.15%
56860_A1	XE_QD28_82780_10G	55	16.36%
56860_A1	XE_SESTO_82764_40G_MUX	25	80.00%
56860_A1	CE_SESTO_82792_100G_343	47	76.60%
56860_A1	XE_SESTO_82764_40G_PT	30	0.00%
56860_A1	XE_SESTO_82764_10G_PT	30	0.00%
56860_A0	CE_DINO_82332_100G_PT_L1_10	16	0.00%
56860_A0	HG_DINO_82332_106G_GB	21	0.00%
56860_A0	CE_DINO_82332_100G_PT	56	0.00%
56860_A0	HG_DINO_82332_11G	30	0.00%
56860_A0	HG_DINO_82332_42G	30	33.33%
56860_A0	CE_DINO_82332_100G_PT_L2_11	16	0.00%
56860_A0	HG_DINO_82332_106G_PT	21	0.00%
56860_A0	XE_DINO_82332_10G	80	27.50%
56860_A0	CE_DINO_82332_100G_GB	63	39.68%
56860_A0	XE_DINO_82332_40G	35	0.00%
56860_A1	XE_G40_84328_10G	40	7.50%
56860_A1	XE_G40_84328_40G	32	12.50%
56860_A0	XE_G28_82322_10G	10	0.00%
56860_A0	XE_G28_82322_40G	8	0.00%
56860_A0	XE_54210	8	0.00%
56960_B0	HG_MADURA_82864_42G_PT	66	9.09%
56960_B0	CE_MADURA_82864_100G_ALT	88	1.14%

56960_B0	XE_MADURA_82864_10G	66	0.00%
56960_B0	HG_MADURA_82864_42G_DUAL_ALT	66	0.00%
56960_B0	XE_MADURA_82864_40G_PT_ALT	66	9.09%
56960_B0	XE_MADURA_82864_25G	22	40.91%
56960_B0	HG_MADURA_82864_42G_PT_ALT	66	9.09%
56960_B0	XE_MADURA_82864_50G	22	18.18%
56960_B0	XE_MADURA_82864_40G_MUX	66	7.58%
56960_B0	HG_MADURA_82864_11G	66	0.00%
56960_B0	HG_MADURA_82864_106G_ALT	88	0.00%
56960_B0	XE_MADURA_82864_50G_ALT	22	18.18%
56960_B0	HG_MADURA_82864_42G_MUX	66	7.58%
56960_B0	XE_MADURA_82864_40G_DUAL	66	24.24%
56960_B0	HG_MADURA_82864_42G_DUAL	66	25.76%
56960_B0	XE_MADURA_82864_25G_ALT	22	45.45%
56960_B0	HG_MADURA_82864_11G_ALT	66	0.00%
56960_B0	XE_MADURA_82864_10G_ALT	66	0.00%
56960_B0	XE_MADURA_82864_40G_DUAL_ALT	66	0.00%
56960_B0	HG_MADURA_82864_106G	88	43.18%
56960_B0	CE_MADURA_82864_100G	88	38.64%
56960_B0	XE_MADURA_82864_40G_PT	66	9.09%
56960_B1	CE_MADURA_82864_100G_RETIMER_ULL	77	45.45%
56960_B1	XE_MADURA_82864_25G_ALT_RETIMER	22	45.45%
56960_B1	XE_MADURA_82864_40G_DUAL_ALT_RETIMER	66	36.36%
56960_B1	XE_MADURA_82864_40G_PT_ALT_RETIMER	66	0.00%
56960_B1	XE_MADURA_82864_10G_RETIMER_ULL	66	27.27%
56960_B1	HG_MADURA_82864_42G_PT_ALT_RETIMER	66	0.00%
56960_B1	XE_MADURA_82864_10G_RETIMER	66	0.00%

56960_B1	XE_MADURA_82864_50G_RETIMER	22	9.09%
56960_B1	HG_MADURA_82864_11G_ALT_RETIMER	66	0.00%
56960_B1	HG_MADURA_82864_106G_RETIMER	88	31.82%
56960_B1	CE_MADURA_82864_100G_ALT_RETIMER	88	1.14%
56960_B1	HG_MADURA_82864_11G_RETIMER	66	0.00%
56960_B1	XE_MADURA_82864_50G_ALT_RETIMER	22	0.00%
56960_B1	XE_MADURA_82864_25G_RETIMER	22	45.45%
56960_B1	HG_MADURA_82864_42G_PT_RETIMER	66	0.00%
56960_B1	HG_MADURA_82864_42G_DUAL_ALT_RETIMER	66	0.00%
56960_B1	XE_MADURA_82864_40G_MUX_RETIMER	66	0.00%
56960_B1	XE_MADURA_82864_40G_DUAL_RETIMER	66	24.24%
56960_B1	CE_MADURA_82864_100G_RETIMER	88	21.59%
56960_B1	XE_MADURA_82864_10G_ALT_RETIMER	66	0.00%
56960_B1	HG_MADURA_82864_42G_DUAL_RETIMER	66	24.24%
56960_B1	XE_MADURA_82864_40G_PT_RETIMER_ULL	55	0.00%
56960_B1	XE_MADURA_82864_40G_PT_RETIMER	66	0.00%
56960_A0	XE_MADURA_82864_40G_PT	33	15.2%

Section 7.5.3: Interop Internal PHY

Section 7.5.3.1: P2P Suite

Switch Device	*Port Macro*	*Total Tests*	*% Fail*

56960_B1	HG_64XD53	108	0.00%
56960_B1	HG_32X106	72	0.00%
56960_B1	HG_128X27	48	6.25%
56960_B1	XE_32X100	136	2.94%
56960_B1	XE_64XD50	204	2.94%
56960_B1	XE_128X25	112	6.25%
56860_A1	HG_32X42	59	3.39%
56860_A1	XE_32X40	87	6.90%
56860_A1	HG_8x100_343	21	14.29%
56860_A1	CE_8X100_343_IEEE	59	8.47%
56860_A1	HG_104x10	36	0.00%
56860_A1	XE_104x10	86	2.33%
56860_A1	CE_8x100_442	59	8.47%
56260_A0	GE_VIPER_12x2P5	36	0.00%
56260_A0	EAGLE_XE_4x10	36	0.00%
56260_A0	GE_VIPER_24x1	39	0.00%
56260_A0	XE_VIPER_6x10	30	0.00%
56260_A0	EAGLE_GE_4x1	12	0.00%
56160_A0	SGMII_TSCE	36	0.00%
56160_A0	QSGMII_TSCE	90	2.22%
56160_A0	QSGMII_QTC	108	0.00%
56160_A0	SGMII_QTC	59	13.56%
56160_A0	SGMII_GPHY	48	0.00%
56160_A0	QSGMII_GPHY	48	0.00%
56560_A0	HG_72X11	33	18.18%
56560_A0	HG_16X27	282	0.00%
56560_A0	XE_72X10	117	0.00%

56560_A0	HG_18X42	64	25.00%
56560_A0	XE_18x40	188	4.26%
56560_A0	XE_16X25	311	8.04%
56565_A0	XE_MAKO	22	18.18%
56960_A0	XE_128X25	89	24.7%

Section 7.5.3.2: Loopback Suite

Switch Device	*Port Macro*	*Total Tests*	*% Fail*

56960_B1	HG_64XD53	33	0.00%
56960_B1	HG_32X106	22	0.00%
56960_B1	HG_128X27	22	0.00%
56960_B1	XE_32X100	22	0.00%
56960_B1	XE_64XD50	33	0.00%
56960_B1	XE_128X25	22	0.00%
56860_A1	HG_32X42	22	27.27%
56860_A1	XE_32X40	22	0.00%
56860_A1	HG_8x100_343	11	0.00%
56860_A1	CE_8X100_343_IEEE	11	0.00%
56860_A1	HG_104x10	11	0.00%
56860_A1	XE_104x10	22	0.00%
56860_A1	CE_8x100_442	11	0.00%
56260_A0	GE_VIPER_12x2P5	28	0.00%
56260_A0	EAGLE_XE_4x10	22	0.00%
56260_A0	GE_VIPER_24x1	30	0.00%
56260_A0	XE_VIPER_6x10	10	0.00%
56260_A0	EAGLE_GE_4x1	10	0.00%
56160_A0	SGMII_TSCE	20	0.00%
56160_A0	QSGMII_TSCE	8	0.00%
56160_A0	QSGMII_QTC	33	3.03%
56160_A0	SGMII_QTC	40	0.00%
56160_A0	SGMII_GPHY	20	5.00%
56160_A0	QSGMII_GPHY	20	0.00%
56560_A0	HG_72X11	15	66.67%
56560_A0	HG_16X27	20	0.00%
56560_A0	XE_72X10	10	0.00%

56560_A0	HG_18X42	10	100.00%
56560_A0	XE_18x40	20	40.00%
56560_A0	XE_16X25	35	40.00%
56565_A0	XE_MAKO	8	0.00%
56960_A0	XE_128X25	22	0.0%

Section 7.6: Static Code Analysis

Starting with SDK 6.5.4, we have upgraded our static analysis code tool to a version with many new checkers and will be working down the backlog of issues. Below shows the current baseline and progress in recent 6.5.x releases:

Table 33: Unresolved Static Code Analysis Issues

Line of Business	*New baseline as of 3/1/15*	*Open Issues SDK 6.5.5*	*Open Issues SDK 6.5.4*	*Open Issues SDK 6.5.3*	*Open Issues SDK 6.5.2*	*Open Issues SDK 6.5.1*	*Open Issues SDK 6.5.0*
DNX	120	124	107	104	61	75	125
XGS	90	23	37	86	16	27	23
SBX	40	*	*	34	1	0	48
SerDes	35	25	23	34	12	45	45
Common	45	32	42	42	11	16	46
Total	330	204	209	300	101	164	301

* As of SDK 6.5.4 we are no longer tracking SBX quality issues as part of our release process.

Section 8: Service Impacting Defects

A Service Impacting Defect (SID) is any defect (internal or external) that has high potential to severely disrupt network operations in a deployed system. This section lists the SIDs that have been identified since last release.

Table 34: Resolved Service Impacting Defects

Reference	*Chips*	*Affected Versions*	*Errata Synopsis*	*Details*
SDK-88820	56850_A0, 56850_A1, 56850_A2	Releases earlier than 6.5.3 and 6.4.10	In releases prior to SDK 6.5.3 and SDK 6.4.10, SDK didn't align LLS L1/L0 which is required by PFC feature.	SDK didn't allocate LLS L1/L0 nodes in multiples of 4. As a result, the PFC status of a port could be associated with L1/L0 nodes of another port. This could cause PFC message to be received on one port impact scheduling on another port if users created customized LLS tree and mapped PFC to L1/L0 schedule level.

Section 9: Resolved Issues for 6.5.5

Section 9.1: Resolved Improvements

The table below lists improvement JIRAs that were added to SDK 6.5.5:

Number	*CSP*	*Chips*	*Release Notes For 6.5.5*

PHY-2508	1062713	56760_A0	BCM82381 chip supports CAUI-C2C and CAUI-C2M interface for 100G. According to new portmod architecture in 100G it is configuring CAUI4 interface so it was failing, CAUI4 support is added.
SDK-73571		88470_A0	BFD packets with your discriminator=0 could be trapped to CPU. Step to configure: 1. Define user defined trap. 2. Create ep 0 with trap destination. Example could be found in cint_bfd.c
SDK-74154	911866	88650_B1 88660_A0 88670_A0	Soc property "counter_engine_source_stat" supported only ETPP counter mode configuration. Add option to configure also IRPP counter mode.
SDK-75726		88470_A0	#Test RDI generation from multiple remote peers: #Create a MEP ,create 3 RMEPs for this MEP. #Send packets as if from each of them and make sure they can all trigger the RDI flag on the MEP's CCMs
SDK-84011	972319	88660_A0	BFD: Add the functionality to define bfd period to be equal 0. In this case transmition will be disabled. This may be done by calling bcm_bfd_endpoint_create() with the UPDATE flag and the bfd_period field set to 0. Note: If BFD per endpoint statistics is enable, setting bfd_period to 0 will disrupt the statistics for the given endpoint.
SDK-86346	984404	AllChips	The release adds the feature of supporting BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE. We define the following drivermodes: BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_DEFAULT

	BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_NOT_SUPPORTED BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_HALF_AMP BCM_PORT_PHY_CONTROL_TX_FIR_DRIVERMODE_HALF_AMP_HI_IMPED.

SDK-87795 995212	56860_A0 56860_A1	A new field EGR_QOS_PROFILE_INDEX was added to table EGR_PORT to provide index for accessing EGR_PRI_CNG_MAP table on Trident2+. This enhancement, however, was not supported by API bcm_qos_port_map_set in previous releases. This has been fixed in this release.
SDK-88107 828888	88650_A0 88660_A0 88670_B0	After creating a MEP and assigning action to it, aditional MEPs may be created using the same action assignment. Additional MEPs' creation will take shorter time. Calling sequence: 1. Create first endpoint with bcm_oam_endpoint_create() and call bcm_oam_endpoint_action_set() as many times as need on the endpoint. 2. Create all subsequent endpoints with the action_reference_id set to the endpoint created in step 1. After this point then new endpoint will behave exactly as the reference endpoint (no further calls to endpoint_action_set() will be required. Notes: 1. Usage not recommended in Arad with soc property: oam_classifier_mode=1 2. Reference endpoints must be of the same type (MIP/MEP, Up/Down, Ethernet/MPLS/PWE) as the new endpoints.
SDK-88209 998990	88660_A0 88670_A0	For MPLS tunnel termination TTL0/1 traps independent configuration use the following trap ids: bcmRxTrapMplsTunnelTerminationTtl0 bcmRxTrapMplsTunnelTerminationTtl1 For MPLS forwarding TTL0/1 traps independent configuration use the following trap ids: bcmRxTrapMplsForwardingTtl0

	bcmRxTrapMplsForwardingTtl1 For IPv4/IPv6 tunnel termination TTL0/1 configuration, enable the flag BCM_RX_TRAP_UPDATE_TUNNEL_TERMINATION_STRENGTH and update the configuration structure fields: tunnel_termination_trap_strength tunnel_termination_snoop_strength

SDK-88949	56860_A0	In previous releases, there was no flag to control whether add B-TAG for peer port traffic. In this release, a flag of BCM_MIM_PORT_PEER_NO_BTAG is added to control whether add B-TAG for peer port traffic
SDK-89417	88470_A0	Up to 3 VLAN tags can be identified on BCM88470 and the above devices. The third tag information can be optionally configured to stamp in Ethernet qualifier by a new switch control: bcmSwitch3rdTagStampMode. This switch control was added in this JIRA.
SDK-90798	88470_A0 88670_B0	Outbound Mirroring: Add an example CINT for Jericho-B0 and Q-AX that utilizes the new flag BCM_VLAN_ACTION_SET_OUTER_VID_UNTAGGED.
SDK-91524 1018564	88950_a0	Added AVS driver function to fe3600
SDK-91726	88680_A0	In Mac-in-Mac point-to point application, ISID value with 2 most significant bits set (0xc00000 - 0xffffff) was not supported. All ISID values are now supported.
SDK-91728	88680_A0	MPLS PORT: Ingress to egress two pointers model supports PWE to MPLS linkage in EEDB. As a result, ingress multicast with single pointer can be now used for VPLS encap optimized scenario. (Pertinent only for Jericho plus)
SDK-91817 1018399	56850_A0 56850_A1 56850_A2	In previous release, SDK didn't support load balance mirror traffic among a set of ports when they were not in same trunk group. Now in this release, SDK can support load balance the mirror traffic among a set of ports (up to 8), and the mirror traffic outing from each port could own its unique RSPAN/ERSPAN encapsulation.
SDK-92490 1013905	88470_A0	In this release , tag/raw mode of PWE is available. Configure step: - Add bcm86xxx_roo_enable=1 to soc properity - legacy PWE and AC of vpls creating - create native eve ac for native eve with BCM_VLAN_PORT_CREATE_EGRESS_ONLY | BCM_VLAN_PORT_NATIVE flag

	- create the vlan translation class with adt legacy process.
	
	- call the api of bcm_port_match_add() with below parameter:
	
	port_match_info.flags = BCM_PORT_MATCH_NATIVE | BCM_PORT_MATCH_EGRESS_ONLY; indicate that it's for native VE
	
	port_match_info.match = BCM_PORT_MATCH_PORT_VPN; indicate that lookup fields are port and VPN
	
	port_match_info.vpn = vsi;
	
	port_match_info.port = PWE gport id;
	
	the relevant example is in examples/dpp/cint_pw_raw_tag_mode.c

SDK-92678	56260_A0	Support has been added to get Discarded bfd packet statistics from firmware.
SDK-93046 1025596	88660_A0	OAM: Enable transmitting CCMs with full 48 byte MAID from the OAMP. Calling sequence: • Set Soc property: custom_feature_maid48_enabled=1 • Define MAID as described in the 88670-PG1xx-RDS in section “Configuring the OAM application” with flag= BCM_OAM_GROUP_FLEXIBLE_MAID_48_BYTE and “name”=48 MAID bytes All endpoints set on the above group will have CCMs with the MAID as configured.
SDK-93715 1016243	56060_A0	This release will set the media type correctly for TSCE when issuing AN CL37 or AN SGMII mode.
SDK-95078	88670_A0	In previous releases, 2b ECC recovery of EM memory tables was not supported. In this release, support 2b ECC, CPU overwrite the HW data entry from shadow table.
SDK-95693 1034390	AllChips	In previous releases, if doing flexport, H/W updated FP_GLOBAL_MASK_TCAM.IPBM, but the data in cache was not updated. When memscan thread scanned this table, the difference was found and treated this as parity error. In this release, this issue has been fixed.
SDK-96161 1036544	88675_A0 88675_B0	In this release a new SOC property named FC_OOB_ILKN_PAD_SYNC_ON_DATA_PIN has been added to select ILKN OOB data Sync pin. There are two values for this property, 0 and 1. 0: Use pin FC_x_SYNC; 1: Use pin FC_x_STAT[1]; The default value is 0.
SDK-96496 1040334	56860_A0 56860_A1	With the given short term solution, BCM stacking application would work as is on HGoE port encapsulation. If customers are sending tx packet using CPU masquerade (bcm_tx() or atp_tx() calls) then there would be some limitations:

	1) Tx for HGoE port is limited to one packet at a time - if they make a call to Tx API they can only send 1 payload buffer within the packet. 2) Can only send to one HGoE port at a time. Packet has to be sent specifically to the HGoE port separately from non-HGoE ports. Customer has to send to non-HGoE ports in another call. 3) SOBMH packet to the HGoE port cannot be sent. Extent of changes: 1) No API changes - we used the current infrastructure for this solution. 2) One file was changed - should be very portable if necessary. NOTE: This short term solution is compile time disabled by default. To enable it users have to include this line in their make file before the compilation - CFGFLAGS += -DBCM_TX_HGOE_LIMITED_SOLUTION

SDK-96696	88470_A0 88670_A0	Added UDP tunnel support.
SDK-96707	88470_A0 88680_A0	In this release, EVPN FRR label is supported for ESI and ROO programs. The only change from previous release is that the ttl of the FRR label is taken from the previous existing label (as EX and BOS) and not propagated using ProgramVar.
SDK-96793 1038235	88660_A0	usage of bcm_trunk_set over an existing trunk with disabled members caused misrouted traffic for a short transition period
SDK-96890 1042144	88670_A0	redundant error message display was removed from bcm_vlan_port_create() api with BCM_VLAN_PORT_WITH_ID flag.
SDK-96989	88470_A0	Diagnose: diagnose command "diag buffer" can be used to display static configuration and dynamic status for buffer. Since some registers existing on old chips no longer exist on QAX, which cause asseration when typing the diagnose command on BCM SHELL. Buffer information can be displayed correctly after the fix.
SDK-97194 1044312	88670_A0	To skip hit indication clearing when creating a new FEC with the bcm_l3_egress_create for running time improvement the BCM_L3_FLAGS2_SKIP_HIT_CLEAR should be use.
SDK-97345 1044693	88670_B0	Jericho didn't support recovering port configuration during warm boot before. This caused warm boot failed on Jericho. From this release, port configuration is recovered from software DB during warm boot.
SDK-97426 945696	88670_A0	OAM: Support Up MEP loopback through the PRGE. • Customer will configure configure the Up loopback recycling ports for both cores, using the following soc properties (for example ports 30,31) ucode_port_30=RCY.2:core_0.30 ucode_port_31=RCY.3:core_1.30 tm_port_header_type_in_30=INJECTED_2_PP

	tm_port_header_type_in_31=INJECTED_2_PP tm_port_header_type_out_30=ETH tm_port_header_type_out_31=ETH oam_rcy_port_up_0=30 oam_rcy_port_up_1=31 • Customer should create an UP-MEP without any changes, according to the user manual PRGE will then respond to LBMs with LBRs. Note that the solution takes two passes through the pipeline (once for the trapped LBM and twice for the injected LBR)

SDK-97657 1045711	56640_A0 56640_A1 56640_B0	Enhancement: To support delete and create HSP hierarchy to get gports corresponding to nodes to be used later for scheduler attributes configuration.
SDK-97699 1046136	56334_A0 56334_B0	Fix the problem about LPM's class and priority are not reported in "l3 defip show" command.
SDK-97786	88375_A0 88470_A0 88650_A0 88660_A0 88675_A0 88690_A0 88750_A0 88950_a0	Behavior of BCM_DNX_SUPPORT flag used in Makefile (MAKE_LOCAL) was changed. Until version 6.5.4, using BCM_DNX_SUPPORT flag compiled all DNX devices, both faps (88375_A0,88470_A0,88650_A0,88660_A0,88675_A0...) and fabric elements (88750_A0,88950_a0...). Starting from 6.5.5, using BCM_DNX_SUPPORT flag will compile 88690_A0 (Jericho2) device family only. In case BCM_DNX_SUPPORT flag was used in Makefile (MAKE_LOCAL), to maintain same build behavior (i.e. compiled all DNX devices), one need to remove BCM_DNX_SUPPORT from the Makefile, and add following 2 lines instead : ALL_DPP_CHIPS = 1 ALL_DFE_CHIPS = 1
SDK-97814 1046560	AllChips	APIs bcm_multicast_egress_set and bcm_multicast_egress_subscriber_set can be used to add/update set of port based and queue based replications respectively.
SDK-98010 1047284	88670_A0 88670_B0	Add counting support (into counter processor) to API bcm_tunnel_initiator_create()
SDK-98249 1044353	88670_A0 88670_B0	Jericho supports SER for internal TCAM.
SDK-98285	56850_A0	In previous releases, I-SID could not be configured per a {DVP, VFI} pair flexibly for the port of type BCM_MIM_PORT_TYPE_PEER in the API bcm_mim_port_add. In this release, this has been fixed by extending the API bcm_mim_port_add with the flag BCM_MIM_PORT_EGRESS_TUNNEL_SERVICE_REPLACE.

SDK-98430 1049052	56260_A0 56260_B0	Added data tlv support for BHH DM messages.
SDK-98491 1039230	88670_A0	Added field stat_id2 in structure bcm_mirror_destination_t and corresponding code in bcm_petra_mirror_or_snoop_destination_create for giving the possibility to add different value according to the update flag. Added test in cintsMirrorDvapis for testing the functionality - AT_Cint_mirror_outbound_1_Count
SDK-98797 1051194	56960_B0 56960_B1	In this release, the following feature has been added: to enable or disable CL91 in AN, users have to invoke bcm_port_phy_control_set(unit, port, BCM_PORT_PHY_CONTROL_FORWARD_ERROR_CORRECTION_CL91, BCM_PORT_PHY_CONTROL_FEC_AUTO); for enabling OR bcm_port_phy_control_set(unit, port, BCM_PORT_PHY_CONTROL_FORWARD_ERROR_CORRECTION_CL91, BCM_PORT_PHY_CONTROL_FEC_OFF); for disabling After chip reset, the default value is CL91 enabled in AN.
SDK-98849 1050418	88670_A0 88670_B0	MPLS: In previous release, max reference threshold of mpls push profile was 64K. An issue occurred when more than 64K mpls tunnel eedb are created using same push profile, In this release, this issue has been addressed by enlarge the threshold to 128K.
SDK-99149 1020713	88375_A0 88375_B0 88470_A0 88670_A0 88670_B0	In Jericho\Qumran devices, in Egress PMF, the action bcmFieldActionDropPrecedence was using a destination resource that is shared with Egress Traps. This is changed, and now it modifies DP directly. When this action is used alongside Egress Traps, the configured DP for the Trap has the higher priority. The action bcmFieldActionDropPrecedence changes the Egress TM DP, and does not affect remarking. In order to configure TC and DP for remarking, the Egress PMF action bcmFieldActionQosMapIdNew should be used. Please find a CINT example in

	cint_field_egr_action_dp_tc_qos.c, which demonstrates the configuration of Egress PMF actions bcmFieldActionPrioIntNew, bcmFieldActionDropPrecedence and bcmFieldActionQosMapIdNew.

SDK-99187 1050290	56850_A0 56860_A0 56960_A0	In previous releases, ING_UNTAGGED_PHB which is pointed by TRUST_DOT1P_PTR from SOURCE_VP table for untagged packet's PHB mapping could not be configured in the API bcm_qos_map_add and bcm_qos_map_delete. This has been improved by using flag BCM_QOS_MAP_L2_UNTAGGED additionally to indicate the configuration of ING_UNTAGGED_PHB in this release.
SDK-99197 1034785	56340_A0 56547_A0 56548_A0 56640_A0 56640_A1 56640_B0	BFD packet transmission out of a remote module on stacking system is provided. RX packet reception from remote module and redirecting to local module where BFD session is maintained is still not supported and that has to addressed through another enhancement. Customer is expected to supply the remote modules port as TX port parameter in BFD endpoint information during BFD endpoint creation if transmission is expected go out of remote module.
SDK-99203 1051778	AllChips	Implemented the requested API for Triumph3 - bcm_cosq_pfc_class_mapping_set/bcm_cosq_pfc_class_mapping_get
SDK-99213 1048791	56450_A0 56450_B0 56450_B1	Support has been added to enable/disable subports, these subports are returned as gports from the bcm_subport_port_add API.
SDK-99216	88670_A0	In 6.5.3 and 6.5.4 SDKs, setting interface was immediately written to HW. However, this is a wrong behavior since it will cause HW combination of previous speed set with unmatched interface type. This was fixed to the following procedure: 1) Save new interface type to software by bcm_port_interface_set. 2) New speed will set HW with speed and with the saved interface type from step 1. Backward compatibility Implication: interface_type change will have no affect until speed_set is followed. If user was writing an application for SDK 6.5.3 or 6.5.4 that changes interface_type and not following with speed_set - in 6.5.5 and beyond he needs to

follow with calling speed_set API.

SDK-99432 1053660	56960_A0 56960_B0 56960_B1 AllChips	Warm boot support is implemented for external phy BCM82381 on TD2, 56960 platform.
SDK-99469	88470_A0 88670_A0	UDH: In order to support UDH in system with OAM enabled for Generic replay messages, program ARAD_EGR_PROG_EDITOR_PROG_OAM_DOWN MAP_RM_UDH was added that removes the UDH from packets going to the OAMP. This new program is selected by OAMP port Updated OAM upmap programs to add UDH, and OAM downmap programs to remove UDH.
SDK-99477 1015163	88650_A0 88660_A0 88750_A0	In previous releases, for SER interrupts requiring asic reset, the application driver just went do it. In this release, an option is given by using customer feature, supporting callback or do asic reset during SER handling.
SDK-99564 1045210	88670_A0 88670_B0	QOS: In previous releases,there was no method to do dscp to mpls tunnel exp mapping in ROO scenario. In this release, this feature has been supported by following steps: 1. Create l2 egress object with bcm_l2_egress_create(); 2. Create a qos map template, 3. Associate the l2 intf with qos map id with the api of bcm_qos_port_map_set()
SDK-99574 1054253	56620_B0	MTRO_PAR_ERR SER support implemented on Triumph.
SDK-99759 1054095	AllChips	From this release, a new flag BCM_VLAN_GPORT_ADD_EGRESS_L3_ONLY has been added for the API bcm_vlan_gport_add(), which can be used to program VLAN.PORT_BITMAP and EGR_VLAN.PORT_BITMAP separately. This flag is applied to physical port GPORT only.
SDK-99899 1046237	88670_A0	EVPN application has been enabled for

88670_B0	database_mode_10 termination.

SDK-100109	1055936	56960_A0 56960_B0 56960_B1	Issue: Redirection is not getting programmed for ports > 128 in Tomahawk. Fix: Redirection pbmp is programmed based on SOC_PBMP_WORD_MAX. Print PBMP as a parameter instead of splitting as multiple parameters.
SDK-100124	1054600	56160_A0 56160_B0	Enhance software based L2 replace operation to use DMA operation to write updated chunk back to Hardware instead of entry-by-entry operation.
SDK-100137	899317	88670_B0	call "bcm_field_control_set(unit, bcmFieldControlLargeDirectLuKeyLength, large_direct_lu_key_length);" to configure the key size for large direct lookup valid values are: 14/15/16/17
SDK-100152	1048924	56450_A0 56450_B0 56450_B1	Added API support for new switch controls to set minimum offset, maximum offset and stride offset increment values in RTAG7 macroflow based hash offset table for higig trunk, trunk and load balancing/entropy applications.
SDK-100172	1049052	56260_A0 56260_B0	Added data tlv support for BHH DM messages.
SDK-100185		88660_A0	Support for bcmBFDEventEndpointTimeout and bcmBFDEventEndpointTimein events has been added for BFD over IPv6 endpoints.
SDK-100399	1057490	56960_A0 56960_B0 56960_B1	In previous releases, "VLAN_PROTOCOL_DATA_INDEX" is always 0 on TH, In this releases,add the index allocation back for Tomahawk and Support profile management for VLAN_PROTOCOL_DATA table.
SDK-100481	1056128	56440_A0 56440_A1 56440_B0 56450_A0	Enhanced the parameter check and error message generated while validating port passed to redirect action handler.
SDK-100523	1050334	88670_A0	OAM: first call to bcm_oam_delay_add() with REPORT_MODE flag set may result in other existing endpoints transmitting DMMs as well.
SDK-100632	1058292	88375_A0 88375_B0	Packet can be forced switch from one port to another port after setting bcm_port_force_forward_set(). A problem occurs when port vlan COE feature is enable. There is another header(DMAC and SMAC) in

	front of output packet, same as original MAC header. A new program is added to handle this case.

SDK-100896	1059675	56840_A0	In previous releases, SER missed to protect some counter registers. (ING_NIV_RX_FRAMES_VLAN_TAGGED/ING_NIV_ RX_FRAMES_ERROR_DROP/ING_TRILL_RX_PKTS /ING_TRILL_RX_ACCESS_PORT_TRILL_PKTS_DIS CARDED/ING_TRILL_RX_NETWORK_PORT_NON_ TRILL_PKTS_DISCARDED) .This patch has fixed it.
SDK-100975	1056081	56340_A0 56340M_A0	Replied to the queries. Reusing L0 and L1 nodes of the unused AXP ports should be fine. Requested to raise an enhancement JIRA.
SDK-100982		88670_A0 88670_B0	SAT OAM: If user call bcm_oam_loopback_add() without tlv or the length of tlv less than 34 byte, the packet can't send out. And also bcm_oam_loopback_get() uses "read_no_clear" way to get rx counters and return E_INTERNAL base on Tx count < RX count, which isn't expected for API action. This issue affects the short loopback packet sending out and packet counters getting.After this fix, the short packet can send out and count correctly, which require the length of TLV longer than 34.
SDK-101029		56850_A0	In previous releases, there was no API to configure the "do not fragment" bit of IP header in mirror encapsulation. In this release, this has been fixed by proposing a new member df in bcm_mirror_destination_t.
SDK-101114	1059774	AllChips	-Which actions/scenarios may trigger the issue: IP-TOS marking with original DP - What is the impact when the issue occurs: In QAX, DSCP/EXP only can marking with Resolved DP(1) - What is the impact of the fix:

	DSCP/EXP only can marking with Resolved DP(1) or original DP (2)

SDK-101191		56242_A0 56440_A0	Added support for flags BCM_VLAN_CONTROL_VLAN_BROADCAST_GROUP_MASK, BCM_VLAN_CONTROL_VLAN_UNKNOWN_MULTICAST_GROUP_MASK, BCM_VLAN_CONTROL_VLAN_UNKNOWN_UNICAST_GROUP_MASK for L3_multicast group in bcm_vlan_control_vlan_selective_set()
SDK-101239	1050686	88675_B0	MIM SPB: When using custom_feature_mim_spb_enable SOC property, a flow of non-MAC in MAC packet transmitted to MIM port, caused dedicate MIM PRGE to fail for SPB bridge feature. This Issue is resolved by validating that the packet is MIM and has link layer present.
SDK-101263	1055838	88660_A0 88670_A0 88670_B0 AllChips	RX: For packets trapped to CPU, in previous version 'in_ac' value is not extracted to bcm_pkt_t structure. In this release, we extract it if the trap reason is 'bcmRxTrapUserDefine'.
SDK-101339	912876	56450_B0	Added BHH over MPLS-TP section support on KT2
SDK-101360	1061184	88470_A0	In the current release, By pass my mac feature could not work for port based pwe feature. In this release, these 2 feature could coexist.
SDK-101741		88470_A0 88670_A0 88670_B0	Supporting OAM packets with OAM MAC identical to L3 My-MAC.
SDK-101791	1047968	56868_A1	Added support for HG2 13G/16G speeds on Titan2+ (56868). 13G supported in both 2 lane and 4 lane modes; 16G support in 4 lane mode only.
SDK-101891	1059420	88670_B0 88675_B0	In previous version, If the port is 1-lane 25G, the portmod will return error. We check portmod module, phymod module and Serdes module, all of them support the port of 1-lane 25G, so I delete the judgement that the port of 1-lane 25G is unsupported.
SDK-102176		56850_A2 56960_A0	In earlier releases SDK there was no support for inner protocal in GRE packets for UDF hash. This has been

supported now.

SDK-102233	1065404	56960_A0 56960_B0 56960_B1	In previous releases, bcmPortControlTxEnable was not supported on Tomahawk. In this release, the feature has been implemented by dropping all packets destined to given port.
SDK-102258	1065477	56850_A0 56850_A1 56850_A2	SDK used S/W aging to simulate H/W aging on TD2/TD2+/TH. In previous releases, S/W aging did not clear hit bits of static entries. Now S/W aging clears hit bits of static entries as well as H/W aging. Both S/W aging and H/W aging do not delete static entries.
SDK-102496	1065404	56850_A0	In previous releases bcmPortControlTxEnable was not supported on Trident2. This has been fixed in this release.
SDK-102541	1047968	56868_A1	1) Update TD2P TDM to support 13G/16G 4-lane mode configuration.
SDK-103059	1069272	88470_A0	VPLS: In the bcm_mpls_port_get api previously if the failover_port_id is a FEC entry, the validity of FEC pointer will be checked. In this release, this check is removed when customer feature of "conn_to_np_enable" is enabled. This change enables user to store any value in learn data of mpls port, even if this value is not a valid FEC pointer.
SDK-103256		56860_A0	Before this JIRA, a legacy external PHY cannot use the following APIs to configure or retrieve the internal PHY's AN abilities because they were not implemented yet: - phy_portmod_dispatch_advert_ability_set - phy_portmod_dispatch_advert_ability_get - phy_portmod_dispatch_remote_ability_get - phy_portmod_dispatch_local_ability_get Also, phy_portmod_dispatch_an_set, which is used by the legacy external PHY to turn on AN for the internal PHY, needs to be fixed since it always overwrote internal PHY's advert ability with the hard-coded local ability.

			After the fix of this JIRA, legacy external PHY can set/get internal PHY's AN abilities, and turning AN on for the internal PHY will not change the advert ability to local ability unexpectedly.
SDK-104055	1072156	88670_A0 88670_B0	OAM:BFD endpoint creation improvement - more stability. BFD endpoint creation failure can cause inconsistency and in some cases inability to create more endpoints.

Section 9.2: Resolved Issues

The table below lists all defects resolved in SDK 6.5.5:

Number	***CSP***	***Chips***	***Release Notes For 6.5.5***

PHY-2454	1050653	56340_A0	BCM84848 supports 1G Native EEE
SDK-49757		56450_A0	Previously pause did not work for some 1G or 2.5G ports. Resolved by Making PG7 for lane0 ports and 30,33,36,39 ports and PG0 for remaining ports
SDK-54927	752923	56450_A0 56450_B0	In previous releases bcm_cosq_gport_delete API could return BCM_E_TIMEOUT during congestion scenarios. This issue has been fixed in API implementation by adjusting bandwidth and flush the packets completely.
SDK-63317	842876	56450_A0 56450_B0	When lanes split the EP to MMU credits setting is not managed properly, the TXLP_PORT_ENABLE is not set for the subports in the WC block. provided fix to set the TXLP_PORT_ENABLE once for every subport of the WC block additionally fix also includes the reset of EP-MMU credits when lanes are aggregated, EP to MMU Credits - TXLP_PORT_ENABLE- when reset for the WC ports , needs to be reset once per every WC subport instead of resetting for all the subports once.
SDK-65668	859710	AllChips	In previous releases, the link partner of a CE port might still be link up when the CE port RX fiber was broken. This issue has been fixed in this release.
SDK-76841	929938	56260_A0 56460	When creating UDF related to Lookup Stage, the BCM_FIELD_DATA_QUALIFIER_STAGE_LOOKUP flag needs to be set in bcm_field_data_qualifier_t structure, otherwise group creation might fail. In this release, the qualifier initialization for DW F2_2 in Lookup stage which was incorrectly pointing to Data2 has been corrected to point to Data1.
SDK-81669	954355	AllChips	'rcload' command was fixed to report the correct line and file name in case of a command failure. This JIRA fixes the issue so that the correct line and file name are printed. The fix is valid also for nested 'rcload' commands case.
SDK-83006	966671	88670_A0	When auto-negotiate is enabled, the command "port xxx sp=yyy" cannot configure the speed of port, and the issue will cause the auto-negotiate be disabled. The issue detailed above affects port command executing, ignoring speed configuration when auto-negotiate is set can solve the issue.

SDK-87481	994334	88670_A0 88950_a0	"phy diag sfixxx dsc" command get wrong clock information. It will shows COM_CLK value of 156.25M even if PLL is configure to another value. After fix this issue, the COM_CLK of fabric port will get correct value.
SDK-89624	1007597	56450_A0 56450_B0 56450_B1	For BCM5645x family of devices FCS error is observed on XE port under certain conditions if the port was mapped to external memory on the fly at run time. This issue has been fixed in this release.
SDK-91008	1013968	88660_A0	default for use_trunk_as_ingress_mc_destination soc property was changes in the following way: for BCM88650 the default is 0 for BCM88660 and BCM88670 the default is 1 this JIRA cause problems with stacking due to new default in BCM88670 and BCM88660. to prevent problems with stacking, use previous default - use_trunk_as_ingress_mc_destination=0 this default change meaning is that adding trunk destinations to mc group must be done in the new way (add trunk id gport to mc group) in relevant devices (Jericho and Arad Plus)
SDK-93388	1010786	56450_A0 56450_B0 56450_B1	DFC need to be reset before flush. Fix is clearing DFC before flush.
SDK-93789	1012774	88670_A0 88670_B0	When working with ILKN and link is down and up during heavy streaming - ILKN might get stuck. This commit solves this issue.
SDK-94704		88670_A0 88950_a0	ALUWP (Active-Link-Up-Watchdog-Period) value was updated to 253 instead of 2 in Jericho, Jericho plus, Kalia, QAX mesh.
SDK-94946	1031580	88670_A0	"diag nif" diagnostic now support ports with multiple ranges (relevant mainly for ILKN) for such ports, representation of the occupied serdeses will look like:

	00-23 0xce8f81 - bitmap represents active serdeses within the range on the right. ILKN_LANE_MAP soc property should be used according to active lanes. for example: config add ilkn_lane_map_lane4_il1.BCM88675=8 means mapping the 4th active lane to the 8th active lane (starting with 0). the ILKN active lanes should be described in SOC property ILKN_LANES. for example: ilkn_lanes_1=0xce8f81 here the 4th active lane is 10 and the 8th active lane is 18.

SDK-95588		88670_A0 88670_B0	add below line in config.bcm to enable the exactly match feature. custom_feature_gal_lookup_exactly=1 this feature will search mpls label database with really mpls label, if the packets has GAL as BOS, dune will search mpls database with mpls labels before GAL.
SDK-95881	1030707	88650_A0 88670_A0	TX: when sending a packet in 'TX', the CRC cannot be found at the FCS field in 'RX',it's next to the FCS field instead. Fixing it by putting the calculated CRC to correct field.
SDK-95925	1037745	88670_A0 88670_B0	In pm4x25, closing / opening MAC loopback should be while the MAC is in SOFT RESET state. Closing or opening the MAC while it is not in reset state might stuck the MAC. This fix puts the MAC into SOFT RESET before closing / opening the MAC loop and take it out of reset - afterwards.
SDK-95966		88670_A0	In OAMP-PE enable oam protection packet headers set together with UDH. Event and punt packets are separated.
SDK-96030		88670_A0	Guard changes to CL72/DFE with lane reset per Eagle/Falcon programmer guide.
SDK-96069		88660_A0 88670_A0	"tr oamp oam" test fixed. Now it tests creation and LOC tests of up to 1024 MEP pairs, including option for testing CCM period updates.
SDK-96175	1039617	88670_A0	portmod_port_enable_get API is fixed to handle all flags combinations (MAC, PHY, RX, TX). in both set/get APIs unsupported combinations will return error.
SDK-96180	1039749	88375_A0 88375_B0 88670_A0 88670_B0 88675_A0 88675_B0	ILKN 6/10 lanes over fabric support added
SDK-96270	1039987	56860_A0 56860_A1	This JIRA fixed the link-down issue for TD2+ with two ports connected using an external cable, one port with

	external PHY while the other with internal SERDES only, when issuing either one of the following two commands: (1) "BCM.0> port xe encap=higig2"; (2) "BCM.0> port hg encap=ieee".

SDK-96521	1039836	88470_A0 88640_A0 88670_A0 88670_B0 88675_A0 88675_B0 88680_A0 88690_A0	The COMMAND_PARITY_LATENCY field is only 3 bits, when we config ext_ram_cmd_par_latency=8 to this field, it will be failed. The problem has been fixed in this release.
SDK-96762	1039351	88470_A0 88670_A0 88670_B0 88675_A0 88675_B0	mbist sequence: memory bist sequence failed on the second run at some devices. Device reset needs to be done after each iteration. This issue fixed by updating memory bist sequence.
SDK-97196	1041457	88675_B0	Fix AVS module build error on 64-bit environment.
SDK-97285	1044210	88670_A0 88670_B0	"diag cosq voq id=<voq> detailed=1" fixed to work under per packet compensation mode
SDK-97427		88470_A0	"diag counters" assertion fixed.
SDK-97475		88470_A0	Diagnostic command "cosq conn egr" fixed to work on QAX.
SDK-97600	1045665	56860_A0 56860_A1	Modified values according to comments from Vamshi
SDK-97659	1043676	88670_A0	Added the missing soc property called stat_if_etpp_arad_compatible_mode to set the ETPP counter generation mode.
SDK-97681	1044009	56960_A0 56960_B0 56960_B1	Mutual exclusion added for protection of subwin structure.
SDK-97875	1046917	56260_A0 56260_B0	In case of SB2, name of pointer to disposition controls changed from other devices causing the crash.
SDK-97899	1046716	88375_A0 88375_B0 88650_A0 88650_B0 88650_B1 88660_A0	The deprecated internal diag command ssdump was removed.

	88670_A0 88675_B0

SDK-97901	1046714	88375_A0 88375_B0 88650_A0 88650_B0 88650_B1 88670_A0 88670_B0	in PMF, when using qualifiers inPort and outPort with input gport argument type of Sysport or Modport, the convert function to local port was using the device Modport instead of core argument. Now the core argument is being calculated from the gport modID and passed within the convert function. bcm_field_qualify_inPort can take arguments of type system gport and module gport.
SDK-97977	1033186	88950_a0	tr 131 fixed for FE3200 of unit number > 0.
SDK-97983	1020527	88660_A0 88670_A0 88670_B0	In presel_management advanced mode the presel-id defines the index in the program selection CAM table. Preselector could not be created in a specific stage. As a result it wasn't possible to assign the same preselector-id to different stages (Ingress-PMF, Egress-PMF, SLB). New APIs are introduced in order to support creating and defining preselectors with stage. In order to create preselector with specific stage the following API should be used: int bcm_field_presel_create_stage_id(int unit, bcm_field_stage_t stage, bcm_field_presel_t presel_id); Instead of using: bcm_field_presel_create_id API. In addition, when adding qualifiers for a preselector which is created with stage, the following flags should be added to indicate the stage for that presel-id: BCM_FIELD_QUALIFY_PRESEL_ADVANCED_MODE_STAGE_INGRESS BCM_FIELD_QUALIFY_PRESEL_ADVANCED_MOD

	E_STAGE_EGRESS BCM_FIELD_QUALIFY_PRESEL_ADVANCED_MODE_STAGE_HASH

SDK-97987		88660_A0 88670_A0	Creating an SLB program was resulting in overwritten and PMF ingress program field (first key of the LEM) in the PMF second pass.
SDK-98099		56960_A0	In previous releases, firmware could not be downloaded for external PHY when MDIO bus number is 6 or 7. In this release, the issue has been fixed.
SDK-98195	928456	88375_A0 88670_A0	Jericho can create 128K InLIFs when soc property "vlan_match_db_mode=FULL_DB". A problem occurred when customer created more than 64K InLIFs. The changes is to fix InLIF creation in SEM-A/SEM-B DBs, so 128K InLIFs can be created successfully.
SDK-98232	1047889	88470_A0	Maximum FAP id and maximum all-reachable FAP id in Kalia system are set to 511 by default. Can be decreased by API: - bcm_stk_module_max_set(unit, 0, max) for max FAP id. - bcm_stk_module_max_set(unit, BCM_STK_MODULE_MAX_ALL_REACHABLE, max) for max all-reachable FAP id. Note: max FAP id can be set to a number divisible by 32 and < 512.
SDK-98292	1047205	88375_A0 88375_B0	There was used a feature that sets the bcmPortControlPrbsMode and it's not available for the NIF ports. NIF ports always inject the PRBS in the PHY level.
SDK-98439	1048551	88660_A0	OAM: creating an accelerated endpoint with tx_gport set to a multicast destination with the OAM statistics enabled (soc property "oam_statistics_per_mep_enabled" set to 1) fails.
SDK-98493	1049453	88670_A0 88670_B0	Fixed an issue that occurred when the double capacity flag was configured with a value larger than 1 and caused the IN-RIFs to behave in an unexpected way (mostly block IPMC packets).
SDK-98502	1049815	56540_A0 56640_A0	In previous releases, the interface type of HG[40] port was set to XGMII. In this release, this interface type of HG[40] port has been set to KR4.
SDK-98513	1037022	88670_A0	An issue where not all the FECs that were accessed returned an access indication using the bcm_l3_egress_get API was fix.

	This issue was relevant only to multi core devices such as Jericho.

SDK-98534	1039379	88670_A0	Eth policer: When configured Aggregate eth policer, the output rate was much higher than expected. Wrong configuration of the eth mtr processor. configuration have been fixed, output rate is as expected.
SDK-98575	1033679	88660_A0	CLMAC may be shut down during Warmboot when Linkscan is on, which may cause traffic losses on CE ports. This has been fixed in this release.
SDK-98659	1049107	56340_A0	In previous releases, the second SerDes was caused link-flap when the first SerDes was disabled in the QG. In this release, the issue has been fixed.
SDK-98776	1048831	88470_A0	fixed bcm_cosq_gport_parent_get to work as expected for QAX
SDK-98786	1047761	AllChips	In previous release, missing update new ports into Oversub group memory table when the speed class of port does not change. In this release,Update new ports into Oversub group memory table after transition the lane of port, even if the speed class of port does not change.
SDK-98812		88670_A0 88670_B0 88675_A0 88675_B0	when using ILKN port with only part of quad, ILKN remap registers configuration according to the board swaps (soc property phy_rx_lane_map_quadX) is fixed.
SDK-98820	1045003	88660_A0	PON: A L3 source binding entry can be added by calling bcm_l3_source_bind_add(). A problem occurs when adding L3 source binding entry with forward group AC, source binding checking will be failed and traffic will be dropped because invalid to add a L3 source binding entry. The change is to add correct L3 source binding entry for forward group AC when custom_feature_l3_source_bind_use_fec_learn_data =1

SDK-98858	1049821	56860_A0 56860_A1	Fixing updation of HSP_SCHED_GLOBAL_CONFIG on flex operations.
SDK-98869	1049034	88470_A0	PON: Customer wants to use outlif[0-8192] on QAX. A problem occurs when create a LIF with outlif[0-8192], because these outlifs are reserved for rif interfaces. The changes are used to support allocate a LIF with outlif[0-8192] on QAX when rif_id_max=0 and custom_feature_pon_wihout_rif_support=1.
SDK-98884		88660_A0 88670_A0	MPLS: when using bcm_l3_egress_create api to create mpls push command (with mpls_action PUSH), delete and replace wasn't handling push profile allocation correctly.
SDK-98892	1046408	88670_A0	MPLS: In case of stack containing NULL label, upper layer protocol field cint example might not classify the next header correctly. cint_field_change_upper_layer_protocol_for_pop_mpls.c was fixed to support general case.
SDK-98951	1050752	56860_A0 56860_A1	There are 8 banks for L3_IPMC table on 56860 chip, but SDK treat it as 4 banks as 56850 chip. The issue has been fixed in this release.
SDK-98957		56850_A0 56850_A1 56850_A2	In previous releases, ALPM warmboot recovery would fail if the soc property num_ipv6_lpm_128b_entries was 4096. This has been fixed in this release.
SDK-98993	1049121	88660_A0 88670_A0	Meter Processor - fixing initial excess bucket level. When setting meter to committed mode, the excess bucket is invalid and no yellow credits should be given at all. The issue of wrong initial bucket level cause undesirable leakage of yellow credits. The fix initialize the bucket level with invalid value, which block the yellow credits.
SDK-99016	1050302	56260_A0 56260_B0	SDK performs TCAM encoding based on the fields KEY & MASK fields in FP_TCAM. For SB2/ML, KEY field which represents F1,F2,F3,Fixed fields but not F4. Due to this encoding/decoding was not getting applied to F4 field in TCAM. Enhanced SDK to perform encode/decode for F4 and DOUBLE_WIDE_MODE fields in FP TCAM.

SDK-99050	1051172	56450_A0 56450_B0 56450_B1 56452_B0 56455_A0 56455_B1 56456_A0 56456_B0 56457_B0 56458_B0	In previous releases, the API bcm_cosq_port_mapping_get would return BCM_E_NOT_FOUND, which was incorrect. This has been fixed in this release.
SDK-99056	1034448	88670_A0 88670_B0	When adding a host entry for ROO to LEM, only even native ARP are allowed (HW limitation). The API now returns an error in case the native ARP is odd.
SDK-99084	1049572	88660_A0 88670_A0	Fixed a check in src/soc/dpp/ARAD/arad_cnt.c in function arad_cnt_counters_set_unsafe() for the EGQ engines 16 and 17.
SDK-99085		88660_A0 88670_A0	Not allow EGRESS_FIELD on command 1 and not allow two different types from ERPP.
SDK-99122	1052203	56260_A0 56260_B0 56460_A0 56460_B0	Issue: SvpValid should enabled to qualify on programmed SrcMplsGport, whereas KTx devices were getting programmed wrongly in case of StageLookup. Solution: Qualifying SrcMplsGport need to program SvpValid also. For KTx devices, SrcMplsGport and SvpValid bits are not sequential will be programmed with two different offsets and widths. Instead programming SvpValid Setting last bit in SrcMplsGport(w.r.t width of SrcMplsGport) and ignored SvpValid. Both data and mask for SvpValid will be programmed w.r.t width of SvpValid.
SDK-99150		88670_A0 88670_B0	OAM:diagnostics "diag oam cl" returns wrong OAM1 key and payload when level argument equals: highest mep level configured + 1
SDK-99158	1051251	56860_A0	TD2+ 100G port uses 3 TSC cores. So TSC reset function is called 3 times. First calll has valid logical

	56860_A1	port number(> 0) but rest of two calls have negative logical port number. SOC_BLOCK_TYPE_BLKID_PORT macro didn't assume negative port number which eventually caused out of memory access. valid port number check should be added before the macro is used.

SDK-99183		88750_A0	Set default CLMAC_MODE.speed to 100G when input autoneg.data_rate==0. Fixed CRC error when AN is enabled.
SDK-99211		88670_A0	Priority order between LEM and KAPS public/private results were as following: 1. LEM Private result 2. KAPS result 3. LEM Public result which means that KAPS public result had higher priority than LEM public result.The new priority order between LEM and KAPS entries is as following: 1. LEM Private result 2. KAPS Private result 3. LEM Public result 4. KAPS Public result
SDK-99228	1052802	56860_A0 56860_A1	PG and Queue settings are set upon device init in legacy flex mode, so they don't need to be done again when activating those inactive ports
SDK-99239	1050913	88670_A0	Allow adding a multicast entry with InRIF larger than 4k for Jericho devices
SDK-99262	1052534	56765_A0 56765_B0	In this release, code changes have been made for Maverick SKU 56765 to handle package level lane swaps/polarity flips internally in SDK
SDK-99265		88660_A0	QoS: When DSCP preserving is enabled on a port (custom_feature_preserving_dscp_enabled=1), 1588 packets delay request packets transmitted out from this port get corrupted (16 bytes in the IP header are lost). With this change, the corruption will be fixed.
SDK-99266	1052711	88670_A0	Fixed an issue where global lifs would be consumed even if the local lif failed to allocate.
SDK-99313	1052743	56960_A0 56960_B0 56960_B1	In previous release for BCM5696x devices, it uses irrelevant bits in INSTR_BUFFER_ISW2_DW7 to generate packet drop reason. This release fixes this issue by using correct bits.
SDK-99357		88670_A0	take SIP into consideration when calculating the priority of the entry which needs to be inserted to TCAM
SDK-99363	1053439	AllChips	TX:when sending a packet by 'TX',the destination port will be set in the ITMH header.In previous release, we use the local port in ITMH header,that may lead to 'TX' fails if local port ID do not equal to system port ID.Now fixing it by using system port ID in the ITMH

header.

SDK-99409	1052838	88670_A0	PTP: In previous release, 1588 48b timestamping didn't work on CLPORT. In this release, this issue has been addressed by configuring 48b mode to CLPORT.
SDK-99447	1050961	88670_B0	PHYMOD: There was an issue that port with Furia PHY connected cannot be disabled, which led to removing port failure on Jericho. This has been fixed.
SDK-99467	1022582	56850_A0 56850_A1 56850_A2	In previous release, when deleting a L2 entry with BCM_L2_DELETE_NO_CALLBACKS flag, same MAC+VID was learnt at another port. For this case, SDK didn't issue a L2 notification. This issue has been fixed in this release.
SDK-99468	1052002	88670_B0	OAM: When calling action set with a newly defined snoop for an Up MEP/MIP the snooped copy will arrive at it's destination with 2 sets of system headers. The outer set will state the destination of the snooped packet and the inner set will contain trapping information (source system port, in/outLIF, etc.) It is the user's responsibility to parse both system headers. If only one set is desired (with the trapping information) the user may use the trap code bcmRxTrapOamUpMEP3.
SDK-99472		88650_A0 88660_A0 88675_A0	The table IPS_QSZ needs special SER handing.
SDK-99483		56845_B0 56846_A1	In previous releases, if customer repeated setting mac loopback on a XAUI port which using xmac, then xmac couldn't receive traffic. In this release, this issue has been fixed.
SDK-99496	1046255	88670_A0	When speculative parsing was used with BOS search on a IPv4/6oMPLS(1-3 labels)oEth packets the configured LB preformed hashing on unexpected parts of the headers. This issue was fixed on the expense of the L4 header support on the mention packets. In case the L4 header is needed and BOS use isn't necessary the custom_feature_speculative_L4_support should be set.

SDK-99516	1053879	88470_A0	MPLS: In previous release, if a port based PWE is configured as 1+1 protection, it could not be retrieved with the api of bcm_mpls_port_get() In this release, this type of PWE can be retrieved correctly.
SDK-99523	1052834	56260_A0 56260_B0	Problem MEPID mismatch was not displayed properly in show CLI. sdk oam endpoint show cli has been fixed.
SDK-99528	1053955	88670_A0	In Out-LIF management, a memory leak that may be caused by rare scenarios was found. This is fixed.
SDK-99532	1051951	56860_A0 56860_A1	In previous releases, there was a dead lock issue when bcm_mpls_tunnel_initiator_set and bcm_mpls_port_delete were called in different threads. In this release, the issue has been fixed.
SDK-99559		AllChips	In PMF, when allocating instructions for a key that exists in several programs, instructions of different sizes were allocated for different programs in rare scenarios. This caused the key structure to differ between the programs, that share the same TCAM database, which causes inconsistency in traffic processing. This is fixed.
SDK-99583	1054055	56960_A0 56960_B0 56960_B1	Issue: bcmFieldQualifyInnerVlanId overleaped instead programmed as bcmFieldQualifyInnerVlan while mapping ingress extractors. Fix: Corrected ingress extractor mapping as bcmFieldQualifyInnerVlanId instead of bcmFieldQualifyInnerVlan.
SDK-99613		88660_A0 88670_A0	Fixed an error in the diagnostic "diag alloc" which caused the last entry in a pool to not be displayed.
SDK-99618		56850_A0	In previous releases, NETWORK_PORT field in ING_DVP_TABLE / ING_DVP_2_TABLE was not set to 1 for MiM backbone VP on the devices which having this field. This has been fixed in this release.
SDK-99643	1038042	88650_A0 88650_B0 88650_B1 88660_A0 88670_A0 88670_B0	In field processor, when removing an entry from direct extraction DB, the FEM instruction is not cleaned properly. in addition, other programs can be affected by this operation. This is Fixed.
SDK-99655		88750_A0 88950_a0	Default values of RTP tables updated like the following: 1. TOTSF: each table entry set to '1'

	2. SLSCT: each table entry set to '0' 3. SCTINC: each table entry set to '0' 4. RCGLBT: each table entry set to be the bit of the same link (Entry 0 = 0x1, Entry 1 = 0x2 etc).

SDK-99743	1049200	88675_A0 88675_B0	Variables of type portmod_ucode_buf_t in pm4x10.c, pm4x25.c,...etc. These variables are STATIC per each PM file and are used to store the SERDES uCode. The above variables get memory allocation during system init and these memories are never freed during re-init or de-init. The fix is to add a segment of codes in the PM destroy APIs, which are called during system re-init or de-init, to free the static uCode buffer per PM file.
SDK-99746	1043361	88202_A0	The "port over priority" feature wasn't implemented before release 6.5.5. After this fix, backpressure from NPU works on a per port/traffic class in the ATMF.
SDK-99781	1037520, 1037422	56860_A0 56860_A1	In earlier releases when enable scaling LPM on 56850 and subsequent devices, the LPM management sw state could run into inconsistency on a very corner case and then further lead to other unexpected behaviors. This has been resolved.
SDK-99785	1054249	56860_A0 56860_A1	In previous releases, the 10M/100M advert ability was not properly configured in hardware. In this release, it has been fixed.
SDK-99819	1054884	56640_A0 56640_A1 56640_B0	The issue occurred when the same fabric modid was configured on a different port. This was because the OFFSET_MAP_TBL had already been configured for the first port. The SDK was changed to reprogram OFFSET_MAP_TBL for the second port.
SDK-99824		88470_A0 88670_B0 88675_B0	OAM: Outgoing packets weren't counted if the OAM endpoint was set on an outLIF which is not the inner-most outLIF. For example, if a packet is encapsulated with PWE and with MPLS and a MEP is set on the MPLS LIF, the packet wasn't counted.
SDK-99863	1054722	88670_A0	OAM: Setting SLM (calling bcm_oam_loss_add() with the SLR flag set), caused OAMP to reject the CCM packets. Therefore LOC events were not generated. SLM setting was fixed.
SDK-99877	1055333	56860_A0 56860_A1	Call soc_td2_port_traffic_egr_enable_set(), which sets THDU_OUTPUT_PORT_RX_ENABLE0_64r,

	THDU_OUTPUT_PORT_RX_ENABLE1_64r MMU_THDM_DB_PORTSP_RX_ENABLE0_64r, MMU_THDM_DB_PORTSP_RX_ENABLE1_64r MMU_THDM_MCQE_PORTSP_RX_ENABLE0_64r, MMU_THDM_MCQE_PORTSP_RX_ENABLE1_64r

SDK-99909	1051228	88670_A0 88670_B0	Which action cause this issue: IP Tunnel: We can update an existing IP tunnel. Commonly we get the encapsulation of tunnel entry and update some property of tunnel, then set again. Afterwards ttl-index and tos-index swapped. What does issue look like: TTL or TOS may not be expected after there are more tunnels; or tunnel update fail when doing more than twice; or tunnel clear fail if tunnel was updated dscp or ttl after creating What impact of this issue and how it is fixed in this release: Uniform IP tunnel should copy or map the TOS/DSCP from the forwarding packet to the tunnel encap correctly even after we updated the tunnel. After this release, we can get correct ttl-index and tos-index before updating tunnel, so copied TOS/DSCP is right even if tunnel updated.
SDK-100058	1053869	AllChips	In previous releases, route addition might fail after the warm boot. In this release, this has been addressed by using scache to record the map between next-hops maintained implicitly in SDK and L3 interfaces.
SDK-100107	1056100	56960_A0 56960_B0 56960_B1	In previous release, packet trace module sent out packet on unexpected port under dual mod mode. This release has fixed it by using the correct way to fetch port number.
SDK-100175	1055842	56340_A0	When the first hierarchy is created, TR3 moves to ETS mode. When this happens, the SDK was clearing the entire bitmap of L0 and L1 indices. The fix was to re-reserve the invalid index for each level.
SDK-100179	1053501	88670_B0	allow RIF larger than 4k for Jericho devices
SDK-100190	1055974	56260_A0 56260_B0	Problem: When the port external buffer is enabled dynamically, the MMU FIFO depth was not adapted to

	the new configuration. Solution: MMU FIFO depth is configured when external packet buffer is enabled.

SDK-100201 1054938	88470_A0	The cint - "cint_pmls_lsr.c" was configured for Jericho device only (multiple cores), now the cint can configure Qax devices also.
SDK-100204 1053212	88670_B0	Which actions/scenarios may trigger the issue: On startup or when adding new port. What is the impact when the issue occurs: TCG/HR shapers were less accurate on JERICHO than on ARAD What is the impact of the fix: TCG/HR shapers are more accurate on JERICHO now
SDK-100239 1054807	88670_A0 88670_B0	Which actions/scenarios may trigger the issue: while injecting to channelized interface and removing one of the channels What is the impact when the issue occurs: packets drops from the other channels What is the impact of the fix: when a port is removed from a channelized interface, it is disabled and all its resources are restored. Traffic on other ports on the same interface is not affected.
SDK-100263 1054553	88670_A0	In previous releases error was printed in Linkscan thread when an fabric port was added into SW Linkscan. This has been fixed in this release.
SDK-100272 1056912	56565_A0 56565_B0	With this release, user can enable software or hardware l2 aging in Apache family as per soc config spn_RUN_L2_SW_AGING.
SDK-100273 1056902	56460_A0 56460_B0	Problem: LMEP_1.INT_PRI is not supported/handled for UpMEP cases in saber2. Solution: Add required code changes to support LMEP_1.INT_PRI field for saber2
SDK-100283	88670_A0 88670_B0	SAT OAM: If user call bcm_oam_loopback_add() without explicitly having SOC property "sat_enable=1", then the API call causes an assertion failure.

	The issue affects the creating of OAM loopback and causes the assertion. A loopback can be created correctly after the fix through adding the SOC SAT enable checking.

SDK-100310	1056564	56224_A0 56224_B0	RX buffer pool was being cleaned up prematurely leading to a crash in a stacking setup. This has been corrected and RX pool will be cleaned up only after all units have been detached.
SDK-100320	1019032	AllChips	In previous releases, the same memory might be freed for several times due to incorrect use of break statement. In this release, this has been addressed by using goto statement to break out “switch-case”.
SDK-100333	1050503	56850_A0 56850_A1 56850_A2	In EFP, HW key will be constructed based on packet type. For slice mode 3, HW will form KEY4 for non-L3 packet and KEY1 for L3 packets. Qualifier IpTypeAny, SDK will install default key (KEY4) in HW, so it can match for only non-L3 packets only. This is a HW limitation. For more details please refer ECAP KEY generations in Section 13 : ECAP Key Fields in XGS V ContentAware Architecture Guide Do not Care (In Stage EFP, IpType will be matched based on packet types like NonIp, IPv4/Ipv6, FCoE, ARP etc., recommended to use respective IpTypes instead of IpTypeAny)
SDK-100344	1056787	56240_B0	Issue: Entry flag to set IPBM_OVERLAY is not getting recovered. Fix: Entry flag IPBM_OVERLAY will be set if entry part is 1 or 3 during entry recovery.
SDK-100357	1056890	56260_A0 56260_B0	Added qualifier data mask setting for IpInfo qualifier in VFP for KT series
SDK-100360	1055362	56860_A0 56860_A1	For Trident2plus, we make sure that indexes are at boundary of 4.
SDK-100400	1052403	88660_A0	L2 Flush machine pause logic is not working.Workaround this limitation set the current index to "0" when a pause is needed.
SDK-100412	1048956	88670_B0	OAM: When using DMA events, event interface timeout was accidentally taken from soc property OAMP_FIFO_DMA_TIMEOUT instead of OAMP_FIFO_DMA_EVENT_INTERFACE_TIMEOUT. This is now fixed.
SDK-100429	1049383	88660_A0	Fixed a metering issue of wrong behavior of Hierarchial policer in Serial mode.

SDK-100446	56850_A2	In previous release, when mirror was configured in non-flexible directed MTP mode, SDK call _bcm_esw_mirror_field_group_reload to recover mirror S/W state from FP policy table for FP mirroring during warmboot. But the invoke will fail when FP mirroring with ERSPAN since _bcm_esw_mirror_destination_find will return BCM_E_NOTFOUND for ERSPAN. Now in this release, the issue has been fixed.
SDK-100461 1056422	88650_A0 88660_A0	The macro SOC_IS_JERICHO_AND_BELOW() for !BCM_JERICHO_SUPPORT case was wrong defined before, it caused the Arad access the wrong memory. This issue is fixed by changing SOC_IS_JERICHO_AND_BELOW() from (0) to (SOC_IS_ARAD(unit) && !SOC_IS_QAX(unit) && !SOC_IS_JERICHO_PLUS(unit)).
SDK-100534 1049200	88675_A0 88675_B0	SOC: The shared resource allocated in bcm_common_tx_init() was not freed during unit detach, which was causing memory leak. This has been fixed.
SDK-100535 1057669	88375_A0 88650_A0 88660_A0 88675_A0	in VTT, unit numbers > 32 are allowed when BCM_MAX_NUM_UNITS is defined accordingly.
SDK-100612 1058455	56960_A0	Warm boot support was previously not available on TD2/56960-A0 for external PHY BCM82381. It has been implemented in this release.
SDK-100615 1057017	88670_A0 88670_B0	OAM: OAM packets transmitted to the passive side of down MEPs, with lower level of the MEP itself where not trapped.
SDK-100623	88670_A0 88670_B0	SAT OAM: If user call bcm_oam_loopback_add() with SOC " bcm886xx_rx_use_hw_trap_id=0", then the SAT can't receive the LBR packet. The issue affects loopback packet receiving. After this fix, the lb packet can send and receive correctly even with the SOC.
SDK-100624 1042446	88660_A0	L2: In CPU learning mode, learning events generated by hardware are parsed by SDK first, a proper GPORT is passed to the callback functions to indicate on which port or logical port this MAC address was learned. A problem occurs when a MAC is learned on a VLAN

	port which is created with the BCM_VLAN_PORT_FORWARD_GROUP flag, in which case SDK may fail to parse the GPORT for any learning events generated on this VLAN port. With this fix, the MAC learning callback will receive correct GPORT which indicates on which port or logical port this MAC was learned.

SDK-100649 1058626	56750_A0 56750_A1 56750_A2	D_TYPE offset correction for TT2 device is fixed.
SDK-100685 1058691	53416_A0	This release will fix the uC time-out problem of doing DSC dump at un-initialized lane for TSCE and TSCF.
SDK-100700 1053311	56850_A0 56850_A1 56850_A2	In previous releases, one more pipe_offset variable was not reset to zero after every bid loop in BST module, it would cause memory corruption. This has been fixed in this release.
SDK-100717	88470_A0	When setting [Tx->Mid] flow control in FE3200, using API: bcm_fabric_link_thresholds_pipe_set, [bcm_fabric_link_threshold_type_t = bcmFabricLinkTxAlmostFull]: Need to adjust sizes of Mid drop thresholds and fifo size, according to the warning message, if given. The relevant thresholds: bcmFabricLinkMidPrioDrop, bcmFabricLinkMidPrio1Drop, bcmFabricLinkMidPrio2Drop, bcmFabricLinkMidPrio3Drop, bcmFabricLinkMidFifoSize.
SDK-100721 1058832	88670_A0 88670_B0	Fixed a bug causing bcm_mirror_destination_get() to fail when the mirror/snoop destination is a trunk.
SDK-100733 990126	88660_A0	In this release, an issue with meter hierarchical mode (MEF 10.2) was fixed. The rate that was sent from each meter was wrong.
SDK-100745 1057736	56850_A0 56850_A1 56850_A2	In previous releases, after a NIV egress object was created with a SD-tag TPID configured, warm boot would fail during the retrieval of TPID. This has been fixed in this release.
SDK-100787 1045256	88650_A0 88650_B0 88670_A0 88670_B0 88675_A0 88675_B0	In L2-Cache, when calling bcm_l2_cache_set for more than one port with the same MAC addresses, followed by bcm_l2_cache_delete for one MAC address, caused an error in internal profile management, that led to inconsistency in HW configuration. This is fixed.
SDK-100816	88670_A0	Fixed EGQ NRDY Threshold configured by the driver

	according to new ASIC guidelines. ILKN 4 lanes latency should be as expected.

SDK-100824 1057385	56260_A0 56260_B0 56460_B0	Configure Ext DDR mode to 1 or 2 based on config
SDK-100832 1059382	56850_A0 56850_A1 56850_A2	In previous releases, warmboot may fail when trying to recover L3 egress objects from hardware. This has been fixed in this release.
SDK-100840 1055895	56760_A0 56760_B0	In this release, SOC_IS_xx check in soc_ledproc_config() has been corrected to cover for Apache so that LED1 gets programmed correctly
SDK-100893 1059657	56850_A0 56850_A1 56850_A2	In previous releases, FEC was not supported for HG[30]. In this release, the issue has been fixed.
SDK-100910 1058275	56860_A0 56860_A1	In previous releases, no error code would be returned if the API bcm_mpls_port_add was called for VPLS VP creation with a specified index which was occupied by another one. In this release, this has been fixed by returning the error code BCM_E_EXISTS.
SDK-100928 1058561	56460_A0 56460_B0	Support to monitor 1.8V and 3.3V supplies is added for BCM5626X and BCM5646X devices.
SDK-100937 1057315	56640_A0 56640_A1 56640_B0 56850_A0 56850_A1 56850_A2	Impact Chip: Triumph1/2/3 Impact Feature/Function: bcm_vlan_control_port_set Impact Module: MPLS In previous release, a function was added to validate MPLS port. However, this function is limited to TD2 platform by using define BCM_TRIDENT2_SUPPORT. Thus, Configuring MPLS using bcmVlanPortUseInnerPri on TR3 failed with BCM_E_PORT. Solution: delete the mpls port validation function, the MPLS port will be validated later as GPORT which is not limited to any platform.
SDK-100966 1059249	88660_A0 88670_A0	Fixed mirror APIs that previously in certain cases of failure, prevented mirror APIs to later work from other threads. APIs that were affected: bcm_mirror_port_dest_delete, bcm_mirror_port_dest_delete_all,

	bcm_mirror_port_vlan_dest_delete, bcm_mirror_port_vlan_dest_delete_all, bcm_mirror_port_destination_add, bcm_mirror_port_destination_get, bcm_mirror_port_vlan_destination_add, bcm_mirror_port_vlan_destination_get.

SDK-100979	88670_A0	Toggling the bcmSwitchL3UrpfMode switch of the global URPF leave the existing FECs in the old URPF state. This issue was fixed.
SDK-100987 1058844	AllChips	bcm_ptp_stack_destroy API is modified to wait for the pending management message transaction between SDK and firmware to complete. This will prevent the possible crash.
SDK-101030 1059961	56640_A0 56640_A1 56640_B0	Flag DLB_ECMP_FLOWSET_TIMESTAMP_PAGEm to be cleared during SER
SDK-101039 1056355	88670_A0	Fixed an error that caused memory leaks when working on more than one device and crash the driver if working on more than one main thread.
SDK-101041 1051545	88675_A0 88675_B0	CRPS bug fix: During Init device, Init DMA FIFO DB was made each time for all units. Fix: Init the DB per unit.
SDK-101106 1059943	56850_A0 56850_A1 56850_A2	Issue: Unable to recover FP mirror action gport parameter after Warmboot. Fix: Syncing the IngressMirror and EgressMirror action GPORTs in the scache and recovered during WB.
SDK-101166	56450_A0 56450_B0 56450_B1 56452_B0 56455_A0 56455_B1	This release addresses issues related to dynamic change in buffer settings for subports
SDK-101196	88670_A0 88670_B0	Added to jericho signals the following: SOC_PPC_FP_QUAL_ERPP_FHEI SOC_PPC_FP_QUAL_ERPP_FHEI_IPV4_TTL SOC_PPC_FP_QUAL_ERPP_FHEI_DSCP SOC_PPC_FP_QUAL_ERPP_FHEI_EXP
SDK-101208 1061045	56565_A0 56565_B0	There are two problems reported in this JIRA: (1) The default interface type of a 4-lane TSCE 42G HG2 port changes from XGMII to CR4 when SDK version changes from 6.5.3 to 6.5.4. (2) The default interface type of a 4-lane TSCF 42G HG2 port changes from KR4 to CR4 when SDK version changes from 6.5.3 to 6.5.4.

	After the fix submitted from this JIRA, both TSCE and TSCF have their default interface types changed back to XGMII and KR4.

SDK-101209 1059768	56850_A0 56850_A1 56850_A2	In previous releases, label 0 might be incorrectly pushed on packets. In this release, this has been addressed by returning invalid label for l3 egress next-hop without MPLS action.
SDK-101213 1061082	88470_A0	ILKN counters are set to "not collected" mode, which means ILKN counters are not collected by Counter Thread. ILKN counters are accessed directly.
SDK-101216 1061118	88670_A0 88670_B0	SCH header-compensation doesn't configures the right profile by calling bcm_cosq_gport_pkt_size_adjust_set() when adjust_info.source_info.source_type=bcmCosqPktSizeAdjustSourceScheduler. Result is not all the profiles can be set. Fixed.
SDK-101232 1056704	88670_A0 88670_B0	Updated the parameter valid check in bcm_petra_port_encap_map_get() and bcm_petra_port_encap_map_set() to allow 18 bit CUD for Jericho and above.
SDK-101260 1059959	88670_A0	MPLS: Memory leak issue has been resolved for PWE binded with MPLS entries.
SDK-101295 1061197	88470_A0	In previous version, some semaphores aren't given in pp_fp related functions, it will result resource leak. Modify the code to make sure the semaphore is released after it is taken.
SDK-101315 1060500	88675_A0	For 1+1 protection, M/U field in the NPH is equal to 0. A problem occurs when 1+1 protection is carried out based on multicast group. M/U field in NPH is equal to 1. A new program is added to handle this case. 2 LSB of Data-format entry should be set to NPH.sub_type to select this program.
SDK-101349 1052448	88670_B0	There was a memory leak while creating Outlif's . This was fixed.
SDK-101372 1061735	56850_A0 56850_A1 56850_A2	Issue 1. Unable to recover InPorts Qualifier during Warmboot if the qualifier PBMP is not part of group's PBMP. Fix: Restricting the ports to be configured in InPorts

	Qualifier, which are not part of Group's PBMP. Issue 2. Unable to recover Field Mirror action GPORTs. Fix: Fixed as a part of JIRA SDK-101106.

SDK-101387 1061741	56960_A0 56960_B0 56960_B1	Configuring Field Processor Group operational mode for EM stage also needs IFP to be configured with the same mode.
SDK-101452 1055459	56640_A0 56640_A1 56640_B0	In ring replace API, AUX L2 BULK control programming needs to be followed by L2 BULK CONTROL register programming. As this function was NOT lock protected, it can so happen that while one thread is updating AUX control fields, other thread can trigger l2 bulk operation. To make sequential programming between threads, all writes to Aux l2 bulk and l2 bulk control registers are protected through l2 mutex lock.
SDK-101461 1061950	56150_A0	In previous releases, the CLI command “edline” could not read file correctly and cause hang to sdk program. This has been fixed.
SDK-101497 1061995	56960_A0 56960_B0 56960_B1	Entry destroy in cold boot will disable the slice lookup if the last entry in the slice is removed. And during warmboot, we read groups based on slice control - lookup_enable and slice_enable. This was causing problem if the last entry field entry is removed. Updated Warmboot code to use only slice_enable and not lookup_enable slice_control to read slice information.
SDK-101529 1061696	56860_A0 56860_A1	Lower frequency SKUs were indexing port macros to see if they were flex capable, but the table being used was for high frequency SKU, which only checks the first port on the TSC-12. Created a table for lower frequency SKUs to properly index into the first port of a TSC-4 Port Macro rather than using the first port of the TSC-12 for all the TSC-4 in that TSC-12.
SDK-101566 1033679	88660_A0	CLMAC may be shut down during Warmboot when Linkscan is on, which may cause traffic losses on CE ports. This has been fixed in this release.
SDK-101583	88660_A0	QOS: When routing into a tunnel, SOC property custom_feature_preserving_dscp_enabled controls whether to preserve the DSCP field in the forwarding header based on per-port configuration. An issue occurred with ICMP packet, when ICMP packet is routed to a DSCP preserve enabled port, the PRGE program is messed up, IP header checksum is wrong when the packet is received.

	With this change, ICMP packet can be routed correctly with correct checksum.

SDK-101602	88670_A0	In L3 IPMC, when using the soc property ipmc_l2_ssm_mode = 2, the public lookups in the KAPS are now supported.
SDK-101621 1062048	88670_A0	bcmRxTrapEgTxMtuFilter is supported form device Jericho B0. But was enabled for creation in Jericho A0 and earlier. This was fixed
SDK-101629 1062313	88675_A0 88675_B0	While created MC RESERVED port profile even though same port, mac's and trap were user the port still got different profile. This is fixed now the port gets same profile.
SDK-101659 1062460	56746_A0	When bcm_port_duplex_set API is used, SDK will check PHY and MAC speed ability of port. If port speed ability which is configured by users is invalid, the API returns failure. In previous release, the API not only checked PHY speed ability of port, but also checked MAC speed ability of port, whether ports were in MAC loop-back or not. But now, for port in MAC loop-back, the API just needs to check MAC speed ability of port.
SDK-101721 1046758	88670_A0 88670_B0	Ingress compensation doesn't work as expected. Fixed.
SDK-101735 1061943	56860_A0 56860_A1	In previous releases, bcm_mpls_port_get might return BCM_E_PARAM after calling bcm_mpls_port_add with replace operation. In this release, this has been addressed by correct the update for tpid reference count.
SDK-101739 1053032	56850_A0 56850_A1 56850_A2	In previous releases, the ports set by BCM_VLAN_GPORT_ADD_INGRESS_ONLY in a given vlan would be deleted by the API bcm_vlan_gport_extended_delete with the flag BCM_VLAN_GPORT_ADD_EGRESS_L3_ONLY, which was incorrect. In this release, this issue has been addressed by supporting the flag BCM_VLAN_GPORT_ADD_EGRESS_L3_ONLY in the API.
SDK-101757 1060329	88670_B0	In Jericho Parser, an ITMH header is parsed according to Arad format, which differs from Jericho format in field offsets and fwd-dest-info encoding. when injecting ITMH Base in Jericho format, and the extension bit indication is set, parser does not recognize it and header size for ITMH Base Extension is not updated. Also, when injecting ITMH Base in Jericho format, for some values, the parser mistakenly identifies them as with an ITMH Base Extension, since it concluded based on Arad fwd-dest-info encoding.

	This is fixed. Now in Jericho devices ITMH is parsed according to Jericho format. In order to use Arad format for ITMH, SOC property itmh_arad_mode_enable must be set.

SDK-101777 1063290	AllChips	In previous releases,encapsulated HiGig2 frames are not decapsulated correctly,In this releases,it has been fixed.
SDK-101795 1059102	88670_A0	When adding entries to the LEM for BFD single hop logical table, MACT limit count may be affected in some cases (depends on system properties). Proper prefixes allocation for LEM logical tables was applied to avoid this case.
SDK-101824 1063369	56850_A0 56850_A1 56850_A2 56960_A0 56960_B0 56960_B1	In previous releases, reference count of profile for ING_PRI_CNG_MAP was not recovered correctly after warmboot. In this release, reference count of profile for ING_PRI_CNG_MAP will be recovered correctly after warmboot.
SDK-101886 1061473	88470_A0 88670_A0	OAM: Using bcm_oam_endpoint_action_set on endpoint of type MPLS / PWE and mpls_out_gport != gport can cause an error and crash
SDK-101889	88670_A0	A bug in kbp ipv6 remove was discovered and fixed. Additional bugs found and fixed in ipv4mc_traverse and ipv6mc_traverse. An issue with getting wrong values from arad_pp_dbal_kbp_buffer_to_entry_key is fixed.
SDK-101904 1058105	56850_A0 56850_A1 56850_A2	In the previous release, the default retry value of "rx_sync_retry" was not enough in some customers environment. This could cause KNET abort Rx prematurely and Rx stall if SDK API Rx mode was used. In this release, that retry times has been bumped enough to meet the customers environment requirement.
SDK-101961 1053260	88660_A0	In Arad devices, in Ingress PMF, when BCM_FIELD_GROUP_CREATE_SINGLE flag is set, the resource calculation attempts to optimize key allocation, and alters the cycle if needed. In that case, it does not update the resource bitmap, which causes incorrect resource allocation. This is fixed.
SDK-102005 1063960	88470_A0	1. Error accessing NBIH_HRF_RESET Register. Access fixed.

	2. Bug fixed in NIF scheduler configuration (NBIH_RX_REQ_PIPE_0_TDM_EN register configuration fixed).

SDK-102008 1063950	88660_A0	The BFD single hop ipv4 tunnel cannot be deleted on previous SDK. The condition has been updated for deleting BFD single hop ipv4 tunnel.
SDK-102052 1061258	56840_A0	In previous releases, the interface type of HG[40] port was set to XGMII. In this release, this interface type of HG[40] port has been set to KR4.
SDK-102066 1060791	56860_A0 56860_A1	Provided a soc API soc_uc_status which the user can use to query the running status of an uC. If the user has determined the BFD uC to have faulted, then the following sequence to recover the app : Call bcm_bfd_detach() API to clean up BFD configuration Shutdown the messaging system soc_cmic_uc_msg_stop() Reload the BFD firmware Re-establish the messaging system Invoke API bcm_bfd_init() to setup BFD again Re-establish BFD sessions
SDK-102075 1049209	56230_B1	Fix the continuously parity error reporting on L2X memory in bcm56230.
SDK-102092 1064603	88470_A0	For QAX the ext_ram_dbuff size can be only 4k, in previous verion, the defualt value of QAX will be set to 1024 if we don't config in config.bcm, it will result crash. In new version, The default value of ext_ram_dbuff size is set to 4096 in QAX, and it is set to 1024 for other chipset as previous verison.
SDK-102096 1060456	88670_A0 88670_B0 88675_A0 88675_B0 AllChips	Which actions/scenarios may trigger the issue: update tunnel DSCP more than twice with BCM_TUNNEL_INIT_IPV4_SET_DF flag. What is the impact when the issue occurs: tunnel update fail What is the impact of the fix: tunnel can update DSCP more times with BCM_TUNNEL_INIT_IPV4_SET_DF flag. without any

other impact.

SDK-102121 1062281	AllChips	In previous releases, VP-LAG gport couldn't be processed in bcm_port_tpid_set()/get()/add(). This has been fixed in this release.
SDK-102132 1064479	88660_A0 88670_A0	Two calls to bcm_l2_addr_freeze before calling bcm_l2_addr_thaw create a bug. Age enable state is saved in every call to bcm_l2_addr_freeze instead of only in the first time of cascading calls. The second save is always saving enable=false and overriding the first store.
SDK-102174 1064389	56850_A0 56850_A1 56850_A2	In previous releases, the error code - BCM_E_NOT_FOUND would be returned when bcm_ipmc_stat_attach was used for IPMC group with a valid VLAN. In this release, this has been fixed.
SDK-102179 1065123	56160_A0 56160_B0	Fixed the assertion failure issue that was observed over the device beyond unit 0 on the BCM56160 platform with multiple units.
SDK-102189 1062926	88670_B0	In previous release, the port was link flap when set force speed for the another port in the Sesto. In this release, the issue has been fixed.
SDK-102222 1064683	88470_A0	When we take semaphore we have to released it after using. In our functions we are using two labels "exit" and "exit_semaphore". Second label is releasing the semaphore. In a lot cases was used wrong label after taking the semaphore "exit" instead "exit_semaphore", I fixed it. There was also the opposite scenario when the label "exit_semaphore" was used without taken the semaphore, also fixed.
SDK-102250 1060198	88670_A0 88670_B0	Overflow won't happen with this cint anymore.
SDK-102251 1060466	56850_A0 56850_A1 56850_A2	With this change, BFD will no longer generate untagged packets if the l3 interface vlan matches with the port untagged vlan. Instead untagged packets will only be generated if the Tx port is part of the untagged bitmap of the l3 interface vlan.
SDK-102256 1056775	88670_B0	In External TCAM (ELK), when adding an ACL entry, it could have fail due to wrong soc implementation, which lead this call into the internal TCAM management. Issue is now fixed and the API is handle by the ELK.

	New external ACL semantic test added.

SDK-102272 1050932	88675_A0 88675_B0	CT VSQ: An FC- OOB Generation path for CT VSQ can be created by calling bcm_cosq_fc_path_add with flag bcmCosqFlowControlGeneration and source cosq value as "-1". Bad Parm occurs when created the FC- OOB Generation path for CT VSQ. The issue detailed above affects the created the FC-OOB Generation path for CT VSQ with the source cosq value as "-1". The FC path can be created correctly after the fix.
SDK-102287 1027616	88375_B0 88470_A0 88660_A0 88670_B0	There is no command to show all SER action behaviors. Now the command "list ser-action" is added to show this.
SDK-102288	88660_A0	VLAN port: When PON DSCP remarking feature is enabled and CPU managed centralized learning is used, the GPORT reported in the learning event is different before/after the DSCP remark feature is enabled. MAC address might fail to be learned by CPU. This change fixes the above issue so that the learn GPORT will be the same despite of the DSCP remark status.
SDK-102297 1065521	56547_A0 56847_A1	In previous releases, the remote ability of 40G ports is incorrect. In this release, this issue has been fixed.
SDK-102337 1059687	56960_A0 56960_B0 56960_B1	In the previous release, portmap regular expression as below. portmap_<port>=<physical port number>:<bandwidth in Gb>[:<bandwidth cap in Gb/i(inactive)/1/2/4(num lanes)>][:<i>] – bandwidth in Gb is the initial speed of a port. It can be selected from 1/10/11/20/21/25/27/40/42/50/53/100/106. – bandwidth cap in Gb is the maximum speed of a port.It can be selected from 1/10/11/20/21/25/27/40/42/50/53/100/106.

	– num lanes to indicate the number of lanes is required by the port. It can be selected from 1/2/4. When to set '1' as the value of the fourth parameter in portmap, '1' will be parsed as 1Gb bandwidth cap instead of a single lane. such as portmap_2=2:25:1 means that bandwidth is 25Gb and bandwidth cap is 1Gb. In this release, portmap regular expression is unchanged and following the left-to-right parsing rule. When the value of the fourth parameter is '1', parse '1' as 1Gb if it is greater than or equal to the value of bandwidth. Otherwise, parse '1' as a single lane. such as portmap_2=2:25:1 means that bandwidth is 25Gb and the port requires one lane. portmap_1=1:1:1 means that logical port number is '1', physical port number is '1', bandwidth is 1Gb and bandwidth cap is 1Gb.

SDK-102343	88670_A0 88950_a0	SER: Array memory protection was not right. A problem occurs when RTP array memory happens SER error. 1) The array index and index of this memory was calculated wrong. 2) The cache of array memory is not software cache. So the device always recover it from hardware. This is not right. 3) Some RTP monitor registers are an array register. But SDK does not care this. This caused the interrupt of these registers cannot be reported. The issues describe above affected RTP array SER protection. RTP array memory SER protection can work correctly after the fix.
SDK-102344	88670_B0 88950_a0	SER: The command "TR 153" does not test array index is not zero. The command "TR 153" does not cover random array index test. So some issue cannot be found. The issue describes above affected SER test. The command "TR 153" can cover random array index test after the fix.
SDK-102353	88670_A0	The following steps added to SW reset sequence: 1. FMAC is disabled before FDR reset. 2. FMAC is enabled after FDR is out of reset.
SDK-102373 1060862	AllChips	NIF: The release 6.5.3 introduced an issue that when setting IFG for CE / XL / XE ports, the maximum value will be limited to 31 bytes. The valid IFG range should be 8 bytes - 127 bytes. This has been fixed.
SDK-102394	88660_A0	OAM: "diag oam cl" diagnostics will show wrong action when the mdl argument is equal to the MEP level+1. This has been fixed.
SDK-102407 1066322	88470_A0	SyncE: In previous release, it would return error when calling API "bcm_switch_control_set" for SyncE. In this release, this issue has been addressed by

	configuring the register of SyncE PLL directly.

SDK-102428 1065543	88670_A0 88670_B0	Routing Over Overlay termination of ARP packets: ARPoEoIP tunnel packets were routed following tunnel termination, although they should have been bridged. ARPoEoIP tunnel packets are now bridged after tunnel termination.
SDK-102467 1063744	88670_A0	Crash occurs in schan mechanism when working with soc "pci_cmcs_num>1". DNX devices should not use multi cmc in schan mechanism.
SDK-102476 1066235	88470_A0	A bug in the OAM classifier configuration caused packets on lif hierarchy where the OAM endpoint is not on the innermost lif to not be counted on the OAM endpoint's counter. The action in this case would be of the default behavior for the innermost lif.
SDK-102481 1066385	88470_A0	OAM: when using LM-PCP endpoints counters allocated for ingress/egress LIF may not be correct: For Down MEPs, TX Packets with PCP n and RX packets with PCP (n+1) increment the same counter (for Up MEPs it was the other way around).
SDK-102485 1066659	88470_A0	PMF actions bcmFieldActionClassDestSet and bcmFieldActionClassSourceSet did not work. This is fixed
SDK-102499	88670_B0	Enabled the chicken bit for PWE LIF counter can't be supported if the packet has 4 outLIFs. EPNI_COUNTER_RESOLUTION_FIX_ENABLED is set to 1 in the code - SDK/src/soc/dpp/JER/jer_mgmt.c - function jer_mgmt_revision_fixes()
SDK-102521 1065439	56440_A0 56440_A1 56440_B0	Intr_enable should be reset after processing the interrupt so that it can wait for the next interrupt.
SDK-102524 1063141	88375_A0	L2 traverse with delete opeartion was taking more time than in 6.4.11. The reason was an unnecessary deletes of the SW shadow. The SW shadow was deleted more then once in one traverse delete operation. The fix is removing the redundant deletes.
SDK-102537 1060632	56060_A0	Fix the assertion failure while using 'l3 l3table hash" CLI on device without VRF feature.

SDK-102565 1066723	56860_A0 56860_A1	KNET module populates hard-coded 8021Q TPID. If RCPU encapsulation is used, hard-coded 8021Q TPID can be used. But if RCPU encapsulation is not used, hard-coded value should not be used in order to support S-TAG 0x88a8.
SDK-102587 1066867	88670_A0 88670_B0	When ILKN was set to non-default speed and user was trying to add the other ILKN port with a shared quad - the new port was changing the speed of the shared quad back to the default speed so the link of the existing ILKN port was falling down. The fixed code doesn't change the speed of a shared quad during port_attach.
SDK-102600 1051960	88670_A0 88670_B0	MPLS: explicit push profile allocation using bcm_mpls_port_create was failing after all-zero push profile was allocated (ttl=exp=0, no CW). This is now fixed.
SDK-102601	88670_A0	88675: Fixed a bug in vrrp where running in 64b mode would not set IPv6 unique mac prefix.
SDK-102622 1065884	88670_A0	Differentiate core for EGQ_PP_PPCT.COUNTER_COMPENSATION.
SDK-102628 1061090	56860_A0 56860_A1	In previous releases, the properties of L3_IIF would be updated by bcm_l3_intf_create when L3IngressInterfaceMapSet was set, which was incorrect. This has been fixed in this release.
SDK-102630	88660_A0 88670_A0	Which actions/scenarios may trigger the issue: When remarking TC to encapsulation MPLS or IP header with L2 forwarding, example MPLS L2VPN encapsulation or vXlan. - What is the impact when the issue occurs: Qos remark profile must be less than 8, otherwise mapping is invalid. - What is the impact of the fix: all 16 qos remark profile can be used for L2 forwarding

SDK-102694 1066553	56851_A0 56851_A1 56851_A2	In previous releases, if the physical port was not configured in config file, then the related logical port would be -1. We should check the logical port before using it during initialization. In this release, this issue has been fixed.
SDK-102722 1067638	56860_A0 56860_A1	Before the fix of this JIRA, any legacy external PHY using 84740 driver with PORTMOD failed to init because the SDK would enter an indefinite loop during the init process. The loop is as follows: - External PHY local ability API calls PORTMOD callback function to retrieve the internal PHY's local ability. - PORTMOD local ability API automatically assumed that its caller intended to retrieve the legacy external PHY's local ability, when a port is connected to a legacy external PHY. Thus PORTMOD local ability API calls external PHY local ability API. This JIRA fixes the PORTMOD local ability API so that it does not automatically assume that its caller intends to retrieve the legacy external PHY's local ability. Instead, it uses a macro to check whether its caller request the local ability of the internal PHY or the legacy external PHY and responds with the appropriate data.
SDK-102724 994334	88670_A0 88950_a0	The property 'serdes_fabric_clk_freq' couldn't be configured into fabric SERDES. The issue has been fixed in this release.
SDK-102836	88675_A0	A memory overwrite occurred when the bcm_ipmc_find API was used. This issue is fixed.
SDK-102854 1053503	56860_A0 56860_A1	In the previous SDK Release the bcm_port_phy_drv_name_get() API was returning incorrect PHY driver name on BCM56860 device. Implemented the functionality to get the PHY driver attached.
SDK-102859 1057703	56840_A0	In previous releases, there were SCHAN operation failure which was due to busyness of H/W. Now SDK will spend a much longer time to wait for SCHAN

operation completion.

SDK-102870 1065680	56340_A0	In the previous release, MSI interrupt does not work on the second PCIE interface when iproc integrated platform is configured in endpoint mode. This has been fixed.
SDK-102959	88470_A0 88670_A0	OAM: When updating lm_counter_base_id field of an up MEP using bcm_oam_endpoint_create, Both TX and RX counters get the same value instead of two consecutive values
SDK-103002 1068745	56960_A0 56960_B0 56960_B1	In previous releases, it was wrong to use maximum speed to initialize a port during Warmboot. Now, SDK gets current speed from MAC of a port, and use current speed to initialize port during Warmboot.
SDK-103044 1058022	88670_B0	OAM: standard MEP configuration can affect also default MEP configuration. This means that under certain circumstances traps created by the endpoint will remain.
SDK-103076 1041473	88670_A0 88670_B0	Fixed bug in port configuration for L2 encapsulation for remote CPU port. Added Encap ID override support for ingress snooping with non-gport value in Arad
SDK-103098 1069262	88670_A0 88670_B0	In the PMF access to TCAM, when a new action is being added, there are 3 action parameters that can be updated. With bcmFieldActionStat, the third parameter corresponds to Out_LIF and it was not being reflected to the actions record in 6.4.11 version. Now an update to this parameter is included.
SDK-103114	88675_A0	In this release the number of PRGE tm_profiles allowed to be loaded was fixed to properly enforce their maximal supported amount.
SDK-103221 928456	88670_A0	InAC: InAC can be created in SEM-B after SEM-A is full when vlan_match_db_mode=FULL_DB. A problem occured when destroying InAC which exists SEM-B. The destroying action was ignored by SDK. The issue described above affected InAC destroy, InAC in SEM-B can be removed correctly after the fix.
SDK-103228 1068905	88470_A0 88670_A0	OAM: bcm_oam_endpoint_get will return wrong mpls_out_gport in case endpoint is of type PWE and the lif used on creation for mpls_out_gport is an egress only lif
SDK-103277	88670_A0	In DMA learning, interrupts from the DMA are called

	even though there are no events waiting. Bad handling of this scenario caused the DMA to stay disabled after the handler is done. The fix is enabling the DMA interrupts in case of error as well.

SDK-103281 1067261	88660_A0 88670_A0 88670_B0	The command diag pp pkttm may cause overflow problem in 6.5.4. This has been fixed in 6.5.5.
SDK-103320 1069960	56850_A0 56850_A1 56850_A2	In previous releases, when a trunk group was set with no members, the dynamic size of the trunk group would be empty and the API bcm_trunk_member_add would return error after warmboot, which was incorrect. This has been fixed in this release.
SDK-103451 1067562	56850_A2	In previous releases, QGROUP was enabled by default for MMU lossy mode on TD2. This has been addressed by disabling QGROUP to use per Queue guarantee by default.
SDK-103463 1063334	88660_A0 88670_A0	BFD single hop supported with detect multiplier and learning routed packets functionality.
SDK-103633	88470_A0 88660_A0 88670_A0 88670_B0	In Jericho and above devices, default value for SOC property "exact_match_tables_shadow_enable" in config-sand.bcm reference is changed from value "1' to "2", meaning LEM shadow is disabled, while all other exact match tables shadow is enabled.
SDK-103643 1072434	88470_A0	MPLS PORT + OAM: In case OAM endpoint is configured on an MPLS PORT and MPLS PORT is called with REPLACE flag, OAM endpoint LM is not functional.
SDK-103652	88680_A0	In Jericho+ an ability to set a polynomial and a barrel shift for the second ECMP hierarchy in configured LB. Setting the polynomial using bcmSwitchECMPSecondHierHashConfig switch option and the barrel shift using the bcmSwitchECMPSecondHierHashOffset switch option.
SDK-103773	88670_A0	AT_Cint_field_test_action_protection uses TR152 within the test, whicdh does not support warmboot. The test is modified to return no_support in warmboot mode
SDK-103779	88670_A0	SAT: bcm_sat_gtf_stat_get can be used to get number of packets sent by GTF. Register for GTF counter will be cleared on read. And this register has fields for other functionality. A SW DB is implemented to store GTF counter value, in order to avoid GTF counter is cleared when accessing other field in the register.

	But Warmboot was not supported for the SW DB in previous release. Warmboot is supported for GTF counter after this fix.

SDK-103782	88660_A0 88670_A0	Interrupt storm nominal config buffer was not recovered after warm reboot. The problem is fixed by recovering the config after warm reboot.
SDK-103784	88670_A0	Adding ports and connectors after WB should work as expected.
SDK-103897	88670_A0	None
SDK-103948	88675_A0	In L3 IPv4MC, the diagnostic 'kbp kaps_show' now correctly parses the prefix length of the DIP(G) field in partially masked entries. Previously it was incorrectly reducing the length by 1 bit.
SDK-104014	88470_A0 88680_A0	88680: Fixed an issue that prevented the use of all EEDB and GLEM entries. Now the entire range of 112k entries can be used. Fixed an issue that prevented the use of all ISEM entries. Now all 128k entries can used, except when there's a hash collision.
SDK-104113	88470_A0	On QAX devices - APIs bcm_cosq_control_range_set/get don't support flags: bcmCosqThresholdDramMixDbuff bcmCosqOcbCommittedMulticast_1 bcmCosqOcbCommittedMulticast_2 bcmCosqOcbEligibleMulticast_1 bcmCosqOcbEligibleMulticast_2
SDK-104385	88470_A0 88680_A0	BFD over IPv4 trapping may for BFD over IPv4 multi hop, BFD over MPLS.
SDK-104473	88660_A0	Eth Policer: lock mechanism wasn't init after warm boot. Issue fixed
SDK-104490 1066675	88470_A0	Calling bcm_cosq_port_mapping_set() fails. Fixed.
SDK-104725	AllChips	VxWorks 6.5, GTO build configuration : Increased RAM_HIGH_ADRS in config.h by ~8MB (0x08000000 -> 0x08800000) in order to be able to build DNX image successfully. Without this change, VxWorks 6.5 GTO image for DNX fails with following error : vxWorks.st: 68791008(t) + 43112544(d) + 23372816(b) = 135276368 (2107216 over)

	Warning: text + data + bss is larger than 133169152 bytes! Booting this image with vxWorks bootROMs will fail unless you increase RAM_HIGH_ADRS in config.h

Section 10: Unresolved Issues for 6.5.5

The following open Urgent priority issues are unresolved in version 6.5.5 of the SDK.

Number	*CSP*	*Chips*	*Errata For 6.5.5*

PHY-2019	976013	56340_A0	SDK build fails when compiled with bmacsec for iproc devices
PHY-2561	1061119	AllChips	BCM82864 bcm_port_phy_control_get/set fails for pre-emphasis and pre-driver currrent
SDK-88792	1000311	56860_A0	TD2+: missing documentation for phy related configuratuions
SDK-91219	1014281	88670_A0 88670_B0 88950_a0	Using bcm_port_phy_control get for Falcon RX settings leaves uc stopped
SDK-100735	1058961	56860_A0 56860_A1	RRPKT and RRBYT when port speed command issued while in PHY loopback - Trident 2 Plus + BCM84328
SDK-101623	1048849	88670_A0	Intermittent failures during KBP init.
SDK-102793	1066930	88670_A0	Egress queue to CPU gets stuck after soft reset
SDK-103800		88670_A0	CLONE - ILKN SerDes can't be locked if enable CL72 at 6.4.11 while it's OK at 6.4.10
SDK-103803	1066867	88670_A0 88670_B0	When ILKN is set to non-default speed and user is trying to add the other ILKN port with a shared quad - the new port is changing the speed of the shared quad back to the default speed so the link of the existing ILKN port is falling down.
SDK-104543	1076009	88470_A0	[QAX]configure MEP on EEDB LL entry for section OAM
SDK-104806	1075344	56860_A0 56860_A1	Need BCM API support for inner vlan range (INNER_VLAN_RANGE_IDX) in double tagged interface

Section 11: Device and Platform Support

This section has been removed from the release notes. For the full list of Broadcom switch and PHY device support, please reference the file SDK-6.5.5-Device-Matrix.xlsx in RELDOCS directory in the release package.

Section 12: Compatibility

Section 12.1: Broadcom Task Engines (BTE) Firmware Compatibility Matrix

The following table shows compatibility between different versions of SDK and Firmware releases. Please refer to the appropriate Network Switching SDK Firmware release notes publication (56XX0_88XX0_FW-RNxxx-R) for the indicated version below.

	SDK-6.5.5	SDK-6.5.4	SDK-6.5.3	SDK-6.4.11	SDK-6.4.10	SDK-6.4.9
4.3.2	BCM56270 BCM56965 BCM88470					
4.3.1		BCM56760 BCM56560 BCM56565				
4.3.0			BCM56760 BCM56560 BCM56565			
4.2.7				BCM56230 BCM56260 BCM56460 BCM56445 BCM56450	BCM56260 BCM56460 BCM56445 BCM56450	
4.2.6						BCM56260 BCM56450 BCM56640 BCM88370 BCM88650 BCM88670

Section 12.2: BMACSEC SDK Compatibility Matrix

Switch SDK Release	*BMACSEC Release*

6.5.0	4.14
6.5.1	4.14
6.5.2	4.15
6.5.3	4.15
6.5.4	4.15
6.5.5	4.15

Section 12.3: PHY Firmware Compatibility Matrix

The following table identifies changes in PHY firmware for newer PHY devices.

PHY Core	6.5.1 Firmware Versions	6.5.2 Firmware Versions	6.5.3 Firmware Versions	6.5.4 Firmware Versions	6.5.5 Firmware Versions
BCM84861	00.00.10	00.00.10	00.00.10	01.00.00	01.00.00
BCM84864	00.00.10	00.00.10	00.00.10	01.00.00	01.00.00
BCM84868	00.00.10	00.00.10	00.00.10	01.00.00	01.00.00
BCM84858	01.02.06	01.02.10	01.03.02	01.03.02	01.03.02
BCM84856	01.02.06	01.02.10	01.02.10	01.03.02	01.03.02
BCM84744	0xD105(A0)/ 0x0128 (B0/C0)	0x0132 (B0/C0)	0x0132 (B0/C0)	0x0132 (B0/C0)	0x0132 (B0/C0)
BCM84757	0xD105(A0)/ 0x0128 (B0/C0)	0x0132 (B0/C0)	0x0132 (B0/C0)	0x0132 (B0/C0)	0x0132 (B0/C0)
BCM84328	R027	R027	R029	R029	R029
BCM82322/28 B1	0xF	0xF	0xF	0xF	0xF
BCM82780	0x22	0x23	0x23	0x23	0x23
BCM82752	0x22	0x23	0x23	0x23	0x23
BCM82758	0x22	0x23	0x23	0x23	0x23

BCM82764	D00F	D011	D013	D017	D017
BCM82790	D00F	D011	D013	D017	D017
BCM82792	D00F	D011	D013	D017	D017
BCM82796	D00F	D011	D013	D017	D017
BCM82381	D011	D011	D013	D013	D013
BCM82209	D011	D011	D013	D013	D013
BCM82073	D011	D011	D013	D013	D013
BCM82864	N/A	N/A	N/A	D00C	D00C
BCM82332	N/A	N/A	N/A	D006	D008
Eagle	D10F_03	D10F_05	D10F_OD	D10F_0D	D10F_0D
Falcon	D10A_06	D10A_07	D10B_07	D10B_0C	D10B_0E

Section 13: SDK Externally Licensed Software Components

SDK contains a number of third-party externally licensed software components. This appendix contains information regarding these components, the license for each of these components, and where these components are used in SDK.

Component	*Origin*	*Location in Source Tree*
EDITLINE	/afs/athena.mit.edu/contrib/sipb/src/editline	src/sal/appl/editline
LIBXML2	http://xmlsoft.org/downloads.html	src/shared/libxml
ED Editor	USENET comp.sources.misc Volume 9, Issue 36	src/appl/diag/edline.c
CINT	http://www.gnu.org/software/bison/	src/appl/cint/cint_parser.[ch]
BIGDIGITS	David Ireland, copyright (c) 2001-11 by D.I. Management Services Pty Limited <www.di-mgt.com.au>	src/soc/dpp/SAND/Utils/sand_u64.c
APIMODE	http://www.gnu.org/software/bison/	src/appl/diag/api/api_grammar.tab.[ch]
VxWorks	Wind River Systems, Inc.	systems/vxworks

Section 13.1: EDITLINE License terms and conditions

```
This package was obtained from the following location, and was modified
for purposes of inclusion into the SOC diagnostics shell.

Removed files:
        MANIFEST Make.os9 Makefile os9.h sysos9.c testit.c unix.h

Added files:
        sysvxworks.c Makefile

Changed functionality:
        Merged unix.h into editline.h
        M-P and M-N now behave like tcsh.
        list_history(count) routine displays history
        Commented out completion
        Changed rl_complete and rl_list_possib into caller-settable
                global functions
        Don't ring bell on TAB if word is already complete

------------------------------------------------------------------------
-
```

```
Index of /afs/athena.mit.edu/contrib/sipb/src/editline

[     ] Name                    Last modified     Size  Description
  ---------------------------------------------------------------------------

 [DIR]  Parent Directory       11-May-99 03:40      -
 [   ]  MANIFEST               07-Jul-97 11:20     1k
 [   ]  Make.os9               07-Jul-97 11:20     1k
 [   ]  Makefile               01-Sep-97 00:34     2k
 [   ]  complete.c             07-Jul-97 11:20     4k
 [   ]  editline.3             07-Jul-97 11:20     5k
 [   ]  editline.c             07-Jul-97 11:20    25k
 [   ]  editline.h             07-Jul-97 11:20     2k
 [   ]  os9.h                  07-Jul-97 11:20     1k
 [   ]  sysos9.c               07-Jul-97 11:20     1k
 [   ]  sysunix.c              07-Jul-97 11:20     3k
 [   ]  testit.c               07-Jul-97 11:20     1k
 [   ]  unix.h                 07-Jul-97 11:20     1k

  ---------------------------------------------------------------------------

$Revision: 1.7 $

This is a line-editing library.  It can be linked into almost any
program to provide command-line editing and recall.

It is call-compatible with the FSF readline library, but it is a
fraction of the size (and offers fewer features).  It does not use
standard I/O.  It is distributed under a "C News-like" copyright.

Configuration is done in the Makefile.  Type "make testit" to get
a small slow shell for testing.

This contains some changes since the posting to comp.sources.misc:
    -   Bugfix for completion on absolute pathnames.
    -   Better handling of M-n versus showing raw 8bit chars.
    -   Better signal handling.
    -   Now supports termios/termio/sgttyb ioctl's.
    -   Add M-m command to toggle how 8bit data is displayed.
The following changes, made since the last public release, come from
J.G. Vons <vons@cesar.crbca1.sinet.slb.com>:
    -   History-searching no longer redraws the line wrong
    -   Added ESC-ESC as synonym for ESC-?
    -   SIGQUIT (normally ^\) now sends a signal, not indicating EOF.
    -   Fixed some typo's and unclear wording in the manpage.
    -   Fixed completion when all entries shared a common prefix.
    -   Fixed some meta-char line-redrawing bugs.

Enjoy,
        Rich $alz
        <rsalz@osf.org>

 Copyright 1992,1993 Simmule Turner and Rich Salz.  All rights reserved.

 This software is not subject to any license of the American Telephone
 and Telegraph Company or of the Regents of the University of California.
```

```
 Permission is granted to anyone to use this software for any purpose on
 any computer system, and to alter it and redistribute it freely, subject
 to the following restrictions:
 1. The authors are not responsible for the consequences of use of this
    software, no matter how awful, even if they arise from flaws in it.
 2. The origin of this software must not be misrepresented, either by
    explicit claim or by omission.  Since few users ever read sources,
    credits must appear in the documentation.
 3. Altered versions must be plainly marked as such, and must not be
    misrepresented as being the original software.  Since few users
    ever read sources, credits must appear in the documentation.
 4. This notice may not be removed or altered.
```

Section 13.2: LIBXML2 - XML C parser terms and conditions

Package was obtained from http://xmlsoft.org/ and is used by diagnostics tool for miscellaneous input/output tasks

This README is part of SDK under src/shared/libxml and is as follows:

```
/*
 * $Id$
 *
 * $Copyright: (c) 2011 Broadcom Corporation
 * All Rights Reserved.$
 */

This package was obtained from http://xmlsoft.org/downloads.html
(ftp://xmlsoft.org/libxml2/libxml2-2.7.2.tar.gz)
and was modified for purposes of inclusion into the SOC diagnostics shell.

Only certain portion of package was included in SDK in 2 places:
    Under srs/shared/libxml
        chvalid.c, config.h, dict.c, encoding.c, entities.c, error.c
        globals.c, hash.c, libxml.h, list.c, Makefile, parser.c
        parserInternals.c, SAX2.c, threads.c, tree.c, uri.c, valid.c
        xmlIO.c, xmlmemory.c, xmlsave.c, xmlstring.c, xmlunicode.c
    Under include/shared/libxml
        catalog.h, chvalid.h, debugXML.h, dict.h, DOCBparser.h
        encoding.h, entities.h, globals.h, hash.h, HTMLparser.h
        HTMLtree.h, list.h, parser.h, parserInternals.h, pattern.h
        relaxng, SAX2.h, threads.h, tree.h, uri.h, valid.h, xinclude.h
        xlink.h, xmlautomata.h, xmlerror.h, xmlexports.h, xmlIO.h
        xmlmemory.h, xmlmodule.h, xmlregexp.h, xmlsave.h, xmlstring.h
        xmlunicode.h, xmlversion.h, xpath.h, xpathInternals.h, xpointer.h

No functionality was changed, but there were modifications to match SDK
requirements

Copyright

Except where otherwise noted in the source code (e.g. the files hash.c,
```

```
list.c and the trio files, which are covered by a similar licence but
with different Copyright notices) all the files are:

 Copyright (C) 1998-2003 Daniel Veillard.  All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is fur-
nished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FIT-
NESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  IN NO EVENT SHALL THE
DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CON-
NECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not
be used in advertising or otherwise to promote the sale, use or other deal-
ings in this Software without prior written authorization from him.
```

Section 13.3: ED Editor License terms and conditions

```
ed - standard editor
  ^^
   Authors: Brian Beattie, Kees Bot, and others

 Copyright 1987 Brian Beattie Rights Reserved.
 Permission to copy or distribute granted under the following conditions:
 1). No charge may be made other than reasonable charges for reproduction.
 2). This notice must remain intact.
 3). No further restrictions may be added.

 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  TurboC mods and cleanup 8/17/88 RAMontante.
  Further information (posting headers, etc.) at end of file.
 _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
 Modification log:
 25Aug92  (W.Metzenthen) Changed malloc() call to calloc() in makebitmap()
          to remove bugs under Linux. Changed a few '^' to the correct '~'.
          General tidying. Recognize Linux via the __linux__ symbol.
          Main change based upon suggestion by Wolfgang Thiel.
 07Sep99  Changed large amounts of stuff to simplify  --Curt McDowell
```

Section 13.4: CINT parser license terms and conditions

The C code for the CINT parser was generated by using GNU Bison parser generator from the file cint_grammar.y CINT is an optional diagnostic tool that can be included in your system by adding CINT to the FEATURE_LIST in SDK compilation flags.

Removed files:

None

Added files:

None

Changed functionality:

None

```
--------------------------------------------------------------------------------
/* A Bison parser, made by GNU Bison 2.4.1.  */

/* Skeleton implementation for Bison's Yacc-like parsers in C

      Copyright (C) 1984, 1989, 1990, 2000, 2001, 2002, 2003, 2004, 2005,
2006
   Free Software Foundation, Inc.

   This program is free software: you can redistribute it and/or modify
   it under the terms of the GNU General Public License as published by
   the Free Software Foundation, either version 3 of the License, or
   (at your option) any later version.

   This program is distributed in the hope that it will be useful,
   but WITHOUT ANY WARRANTY; without even the implied warranty of
   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
   GNU General Public License for more details.

   You should have received a copy of the GNU General Public License
   along with this program.  If not, see <http://www.gnu.org/licenses/>.  */

/* As a special exception, you may create a larger work that contains
   part or all of the Bison parser skeleton and distribute that work
   under terms of your choice, so long as that work isn't itself a
   parser generator using the skeleton or a modified version thereof
   as a parser skeleton.  Alternatively, if you modify or redistribute
   the parser skeleton itself, you may (at your option) remove this
   special exception, which will cause the skeleton and the resulting
   Bison output files to be licensed under the GNU General Public
   License without this special exception.

   This special exception was added by the Free Software Foundation in
   version 2.2 of Bison.  */

/* C LALR(1) parser skeleton written by Richard Stallman, by
   simplifying the original so-called "semantic" parser.  */
```

Section 13.5: BIGDIGITS license terms and conditions

```
Contains BIGDIGITS multiple-precision arithmetic code originally
written by David Ireland, copyright (c) 2001-11 by D.I. Management
Services Pty Limited <www.di-mgt.com.au>, and is used with
permission.

David Ireland and DI Management Services Pty Limited make no
representations concerning either the merchantability of this software
or the suitability of this software for any particular purpose. It is
provided "as is" without express or implied warranty of any kind. Our
liability will be limited exclusively to the refund of the money you
paid us for the software, namely nothing. By using the software you
expressly agree to such a waiver. If you do not agree to the terms, do
not use the software.
```

Section 13.6: APIMODE parser license terms and conditions

The C code for the APIMODE parser was generated by using GNU Bison parser generator from the file api_grammar.y APIMODE is an optional diagnostics shell interface that can be included in your system by adding APIMODE to the FEATURE_LIST in SDK compilation flags.

See "CINT parser license terms and conditions" for the Bison licence.

Section 13.7: Wind River Systems license terms and conditions

See WRS_LICENSE.pdf contained in each systems/vxworks subdirectory.