

Software Development Kit Release Notes SDK 6.4.4

February 26, 2015

Broadcom
Network Switching

Section 1: About This Document

These are the Release Notes for the Broadcom Network Switching Software Development Kit Release 6.4.4.

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.

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:

Table 1: Product Documentation

Document	Description
56XX-PG644-R	BCM API Reference Guide. This manual describes the theory of operations of the API and all existing BCM APIs for this release.
56XX-PG707-R	Stacking Software Guide This guide describes how to use the discovery and stacking applications provided in this release.
56XX-PG818-R	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.

Section 3: New in this Release

This section describes feature and device support that is introduced in this release.

SUMMARY OF NEW FEATURES

BCM56860 (TRIDENT 2 PLUS)

TRIDENT 2 PLUS

The BCM56860 family of switches is aimed specifically at the cloud DC market, ToR and spine. The switch offers up to 104 x 10GbE, up to 32 x 40 GbE or up to 8 x 100 GbE ports. The faster member of the family provides up to 1.28Tbps of Ethernet switching in oversub mode or 960G in line rate mode.

This release supports the following device configurations.

- BCM56860: up to 104 x 10GbE, up to 32 x 40 GbE or up to 8 x 100 GbE ports

Basic bring up of SDK on hardware is completed. Many features have achieved Beta status (Development and unit test on HW complete). SQA testing is in progress and there is on-going test case enhancement to fill coverage gaps and address new features.

NEW FEATURES ADDED

HiGig Over Ethernet

Trident 2 Plus introduces HiGig Over Ethernet (HGoE) mode. In this mode, any Ethernet port can now carry an HG2 protocol which can be used for a stacking application. The physical port no longer needs to be configured as a HiGig port in order to be connected to another physical HiGig port to form a system.

MPLS Enhancements

Added software support for a custom, non-standard Entropy label implementation introduced in BCM56860. The custom entropy label needs to be at the top of the label stack and must not be configured as a forwarding label. An incoming mpls packet with a label identified as entropy label will be popped. The existing RTAG-7 hashing schemes allow the entropy label to be configured for hash resolution among the ECMP paths. Set of APIs added in this release will allow the users to program and manage the entropy label identifiers.

CoE Preview

Added a new feature for Channelization over Ethernet (CoE) on Trident2Plus based on the HiGig Proxy mechanism. Also added new module management APIs that are needed to be configured to setup the feature.

Added new API signatures to configure EtherTypes, TPID sizes relevant to the feature. In addition there are new profiling schemes added that are needed to support setting per-port properties for the new subtended ports.

RioT Preview

This is a new feature to support Routing into and out of L2 tunnels. This feature enhances the capability to route the packets into the overlay network in single pass. Overlay network is set up with L2 tunnels (vxlan, l2gres ..etc..). As part of this feature, the capability to route in the virtual domain (VPN) is also added. Before RioT feature was introduced, we could only route the packet in underlay domain in single pass and for routing into the overlay domain, the packet had to make two passes in switch.

OAM Preview

Added SDK enhancement for device BCM56860 to support OAM standards - IEEE 802.1ag, Y.1731 (Down MEP) and MPLS LM/DM (RFC-6374), and enable configuration of CPU as OLP (Offload processor)

FP Enhancements

Added new field Qualifiers and field Action Lists to support OAM in TD2Plus.

Over Subscription Buffer Enhancements

In TD2Plus, The Priority mapping configuration of OBM is now synced with Priority Group mapping of MMU. If `spn_MMU_LOSSLESS` is not configured, then all the priority groups are Lossy (Flow Control is disabled). If `spn_MMU_LOSSLESS` is SET (1), then priorities which are mapped to Priority Group - 7 will have flow control enabled, (i.e.) the per port priority mappings using configuration variable `buf.map.pri.prigroup` will now also be applied to the OBM.

VP-VLAN membership updates

Support added for virtual-ports to be associated with STP-states, like for front panel ports. Support is added for GPPs (Subtended ports) to be added to VLANs and to be associated STP-states. Support added to bind VPN based virtual-ports to VFIs. Support for setting up filtering method on VPs/GPPs. Support for new GPP/VFI key types when hash based membership checks are involved.

VXLAN Enhancements

VXLAN Scaling:

- A VXLAN port on the access can be shared across VPNs for IoT.
- In this case, One VXLAN port (SVP) can be shared across multiple VFIs when there are multiple matches with different VFIs to point to the same SVP from Ingress VLAN XLATE, VLAN and VFP.
- Thus the hardware VP resource can be saved at the ingress.
- Egress VLAN translation on the access is not on VXLAN port when it is shared across VPNs. It should be on VFI or VFI + DVP/DVP group for shared VXLAN port.

VXLAN Initiation Proxy:

On BCM56850, outer L2 header encapsulation on VXLAN is determined by next hop which mapping to egress local port. Thus, BCM56850 only supports the outer L2 header encapsulation per egress local port. Now, on BCM56860, next hop can be mapped to egress DGPP (Destination Global Physical Port). Thus, BCM56860 has the ability to do outer L2 header encapsulation for low end chips.

Multiple Next Hops On the Same Port for VXLAN:

On BCM56850, outer L2 header encapsulation on VXLAN is determined by next hop which has a 1:1 mapping to egress local port. Thus, BCM56850 only supports point-to-point links between nodes in the Underlay Transport Network. Now BCM56860 can use extra module IDs as my module ID to map multiple DGPPs (Destination Global Physical Port) to one local egress physical port. At the same time, an egress L3 next hop has a 1:1 mapping to a DGPP. Thus an egress L3 next hop has a N:1 mapping to an egress local port indirectly. Thus, BCM56860 supports point-to-multipoint links between nodes in the Underlay Transport Network.

My Station Enhancement:

On BCM56850, my station TCAM size is 512. Thus, TD2 only supports 512 my station hit. Now BCM56860 has enhanced L2_ENTRY (288K) and L2_USER_ENTRY (512) for my station hit. Thus, BCM56860 supports more than 4K my station hit.

VRF Aware VXLAN Termination:

On BCM56850, VXLAN termination supports:

- VXLAN tunnel termination using {DIP} lookup
- VXLAN source virtual port assignment using {SIP} lookup
- VXLAN VFI assignment using {VNID} lookup

They are VRF unaware. The support of VRF aware VXLAN termination includes:

- VRF aware VXLAN tunnel termination using {VRF, DIP} lookup
- VRF aware VXLAN source virtual port assignment using {VRF, SIP} lookup
- VRF aware VXLAN VFI assignment using {VRF, VNID} lookup

VLAN is a reasonable approximation for VRF. So BCM56860 uses outer VLAN instead of VRF in the lookup key. This means:

- VRF aware VXLAN tunnel termination using { OVID, DIP} lookup
- VRF aware VXLAN source virtual port assignment using { OVID, SIP} lookup
- VRF aware VXLAN VFI assignment using { OVID, VNID} lookup

Customer should be aware of the mapping between VRF and outer VLAN and maintains such mappings.

VXLAN Bud Node External Proxy:

Using a given DGLP to terminate the VXLAN packets on the second pass, the bud node external proxy needs two passes. The VXLAN packets are routed to all network receivers during the first pass, and a L2 copy is sent to the customer ASIC over a HG port. The customer ASIC encapsulates a given DGLP in the HG header and sends the packets back to the switch over this HG port working under the second pass proxy mode. The VXLAN packets are terminated according to matching the DGLP in the HG header with the configured one on the switch during the second pass.

Flex Stats Enhancements

In TD2Plus, Loss measurement in OAM is achieved through Flex Counters. To Support the OAM requirements, new flag for stat group mode creation (BCM_STAT_GROUP_MODE_CAPABILITY_OAM) is defined. When passed, the flag instructs the API to create a stat group, suitable for OAM. Group with OAM capability is supported through custom group_mode_id create functions, for example

```
int bcm_stat_group_mode_id_create(int unit, uint32 flags, uint32 total_counters,
uint32 num_selectors, bcm_stat_group_mode_attr_selector_t *attr_selectors,
uint32 *mode_id);
```

Here, the "flags" field should be set to (at least) (BCM_STAT_GROUP_MODE_INGRESS | BCM_STAT_GROUP_MODE_CAPABILITY_OAM) .

100G

100 Gigabit port support was added to Trident 2 Plus.

LEGACY FEATURE STATUS

The following list of Legacy features is supported in this release at Beta quality.

Table 2: Trident 2 Plus Legacy Feature Status

Feature	Status	Note
Legacy		
ALPM	Beta	
BCM12C	Beta	
Cosq	Beta	
FCOE	Beta	
HIGIG Proxy	Beta	
KNET	Beta	
Legacy Field Processor	Beta	
L2	Beta	
L2 and IP Multicast	Beta	
L3	Beta	
L2GRE	Beta	
Link	Beta	
Mac-in-Mac and SPB	Beta	
MPLS	Beta	Now including enhancements- see above
Mirror	Beta	
NAT	Beta	
Over Subscription	Beta	
Packet Rx/Tx	Beta	
Port	Beta	
PROXY	Beta	
Rate	Beta	
Rtag7	Beta	
Resilient Hash	Beta	
SER	Preview	
Switch Control	Beta	
Stack	Beta	

Table 2: Trident 2 Plus Legacy Feature Status

Feature	Status	Note
STAT	Beta	
STG	Beta	
TIME	Beta	
Tunnel	Beta	
Trill	Beta	
Trunk	Beta	
Port Extender	Beta	
VLAN	Beta	
VPLAG	Beta	
VXLAN	Preview	
Legacy Warmboot	Preview	

FEATURES NOT YET SUPPORTED

The following features are not yet supported.

Table 3: Features Not Supported in this Release

Feature	Release with Support
OTN-COE	SDK 6.4.5
Dynamic scheduler change between SP & WDDR	SDK 6.4.5
LAG Failure Protection	SDK 6.4.5
RioT VPLS and Provider Backbone Bridging	SDK 6.4.6
Hierarchical ECMP	SDK 6.4.6
IP/EP Enhancements	SDK 6.4.5
Flexport	Phased starting in SDK 6.4.5
OTN- APS:	SDK 6.4.5
1:1 Protection for Multicast Traffic	
1+1 Protection for Multicast and Unicast Traffic	
RDROP	SDK 6.4.5
OTN- Table Size Increases	SDK 6.4.5
Unknown IPMC Default Route	SDK 6.4.5
L2 Enhancements:	SDK 6.4.5
IGMP Support in L2 Entry Table	
L2 MAC Learning Rate improvement	
Enhanced Operation of IPMCv4 L2	
L2 Multicast Loop up Based on IP address	

STATE OF BCM56860 PHY MICROCODE:

All Eagle PMD PHY Microcode is development complete including manufacturing considerations.

The following table identifies SerDes features supported in this release.

Table 4: BCM56860 SerDes Feature Status

Feature	Status	Note
Eagle: UCODE	D10E_04	
MAC/PHY Loopback		
10G-XFI	Passed	Traffic is CPU generated
40G-MLD	Passed	Traffic is CPU generated
100G-MLD	Passed	Traffic is CPU generated
PRBS (some poly, probably the highest)		
10G-XFI	Passed	Traffic is CPU generated; only through "phy diag xe prbs" commands.
40G-MLD	Passed	Traffic is CPU generated; only through "phy diag xe prbs" commands.
100G-MLD	Not supported	Change port structure to 3x40 to run PRBS.
Port to Port		
Forced Speeds		
10G-XFI	Passed	Traffic is CPU generated, Traffic Gen.
40G-MLD	Passed	Traffic is CPU generated, Traffic Gen.
100G-MLD	Passed	Traffic is CPU generated; 442/343/244 has been tested. Traffic Gen.
AutoNeg Speeds		
CL73	Passed	10G/40G/100G. Simple AN and linkup + CPU traffic. No stress test done.
EyeScan Types 1,2,3	Passed	

BCM56960 BETA RELEASE

BCM56960 family is industry's First High-Density 25/100 Gigabit Ethernet Switch for Cloud-Scale Networks. The switch offers 32 100GbE, 128 25GbE ports. This release contains enhanced Early Access bring up support for BCM56960 A0.

This release supports the following device configurations.

BCM56960 A0: 32x100G, 128x10G, 128x25G, 28x100G_4x106G, 32x40G, 64xd40G

STATUS OF SUPPORT:

Basic bring up of SDK on hardware is completed. Many features have achieved Beta status (Development and unit test on HW complete). SQA testing is starting and there is on-going test case enhancement to fill coverage gaps and address new features.

NEW FEATURES IN THIS RELEASE

Soft Error Recovery

Soft Error (SER) logging support for Tomahawk is added in this release as preview. SER detection, correction and logging is supported for Ingress, Egress pipeline and MMU memories and TCAMs (including IFP_TCAMs) . Error injection and Tr144 (SER diagnostic)test support is enabled for hash, index, overlay memories and TCAMs.

Error injection and Tr144 for registers and MMU memories are not supported in this release. Warmboot as related to SER is not supported in this release.

LED Processor

This release supports the LED processor to program LED interfaces for Tomahawk has been added. Sample LED microcode has been provided which was used to test all the LED processor functionality with multiple port configurations (128x10, 32x100 and 32x40).

Robust Hash

This release supports Robust Hash mode for VLAN translate, egress VLAN translate and MPLS tables. A config variable has been added to overwrite the default random seed which is used to configure the robust hash tables. Please refer API guide for additional information on configuring the config property.

NIV Enhancements

BCM56960 provides enhancements that support embedding NIV and Port Extender related encapsulation information as part of the extended view entry in L3_ENTRY table. NIV/Port Extender support in L3 extended view can be enabled/disabled via config property `embedded_nh_vp_support` in SDK. To enable the feature, configure `embedded_nh_vp_support` to 1, thereby enabling embedded nexthop (`l3_egress`) information in `bcm_l3_host_t` to be used for other types of GPORTs (e.g. `BCM_GPORT_NIV` as the destinations. The tradeoff is that on some of the devices the number of distinct destination MAC addresses or other parameters might be limited.

Algorithmic LPM

This feature supports ALPM for longest prefix match in preview mode. The release support 4 bank mode, which uses all the UFT table when configured in ALPM mode. The configuration mechanism is similar to that of Trident2, using config property, `alpm_enable=1`.

Visibility Packet Trace

This release supports internal device packet tracing feature to provide detailed information on how a specific packet is processed through the ingress pipeline. This feature allows the user to send a packet into the Ingress Packet Processing Pipeline that are then processed as if they were received on one of the front panel ports and log internal forwarding and packet processing states. Please refer to API for packet trace API documentation.

Note: For Trident, Trident2, and Triumph3, `bcm_switch_pkt_info_hash_get` API is used as a method to determine the egress port of various hashed operations such as ECMP and Trunking. For Tomahawk, Internal device packet tracing (Visibility) feature provides detailed hashing resolution information instead of `bcm_switch_pkt_info_hash_get` API.

Sflow

This release supports below enhancements to ingress sFlow feature in BCM56960: 1. sFlow MTP configuration support - New flag `BCM_MIRROR_PORT_SFLOW` for `bcm_mirror_port_dest_add()` API has been added to create sFlow mirroring destination to a port.

2. sFlow Mirror packet encapsulation - New flag `BCM_MIRROR_DEST_TUNNEL_SFLOW` for `bcm_mirror_destination_create()` API has been added to allocate a MTP destination with a sFlow tunnel encapsulation. Also, new fields `udp_src_port` and `udp_dst_port` has been added to `bcm_mirror_destination_t` structure to allow configurable UDP header for sFlow mirror encapsulation headers.

3. Configure second bank of ingress samplers (Flex samplers) controlled by IFP user policy - The sampling rate of the flex sampler per port can be set using the port control `bcmPortControlSampleFlexRate`. Option to send Flex sFlow samples to either CPU or to all mirror destinations created with `BCM_MIRROR_PORT_SFLOW` flag can be set using port control `bcmPortControlSampleFlexDest`. A new switch control (`bcmSwitchFlexRandomSeed`) to set the seed for the random number generator of Flex Sampler has been added.

4. Option to send Ingress sFlow samples to either CPU or to all mirror destinations created with `BCM_MIRROR_PORT_SFLOW` flag can be set using port control `bcmPortControlSampleIngressDest`.

Please refer API guide for additional information on configuring the enhanced sFlow feature for BCM56960.

Cut Through Forwarding (ASF)

Tomahawk supports two forwarding modes in the MMU namely ? Store-and-Forward (SAF) and Cut Through Forwarding (ASF). Cut Through mode enables a significantly reduced packet forwarding latency in the MMU as opposed to Store-and-Forward mechanism. Tomahawk MMU supports Multimode ASF, in which CT forwarding between ports of dissimilar speeds is supported. Three CT modes are supported (i.e.) Same-Speed CT, Slow-to-Fast CT and Fast-to-Slow CT. CT mode on a port can be configured through the 'port' command's 'asf' switch on the BCM diag shell or through the BCM API: `bcm_switch_control_port_set()` and the switch control `bcmSwitchAlternateStoreForward` with arg (0 = SAF, 1 = Same Speed, 2 = Slow-to-Fast, 3 = Fast-to-Slow). The configuration property `asf_mem_profile` can be used to limit the scope of ASF configuration (CT profile) at the unit level. `asf_mem_profile = 0` (NO CT) would allow no CT configuration on the ports, `asf_mem_profile = 1` (SIMILAR SPEED CT) would limit the range of the source port speed to the HiGig counterpart during Fast-to-Slow CT and `asf_mem_profile = 2` (EXTREME SPEED CT) would open up a wider range of source port speeds during dissimilar speed CT. The default configuration for CT profile is SIMILAR SPEED CT.

Latency Bypass

Tomahawk supports a configurable pipeline latency in its Ingress & Egress pipelines. It is possible to achieve reduced levels of port-to-port switching latency at the cost of pipeline features by bypassing selective stages in the Ingress/Egress pipelines. Three latency modes are supported : Normal Latency, Low (L2) Latency and Balanced (L2/L3) Latency. In Low (L2) Latency mode, only the L2 functions for an 802.1Q-2005 switch are supported, and has the least forwarding latency. In Balanced (L2/L3) mode, basic IPv4/IPv6 unicast and multicast routing with IFP are supported, and has a nominal latency. In normal latency mode, all pipeline functionality are enabled and has the highest forwarding latency of the three modes. Pipeline latency mode can be configured only during boot time, through the configuration property `switch_bypass_mode` (0 = Normal latency [Default], 1 = Balanced (L2/L3) latency and 2 = Low (L2) Latency).

IFP SUPPORT

IFP Legacy Support

Support for IFP Legacy mode on BCM56960 has been added in preview status.

In this release, we are targeted to provide the legacy IFP behavior that is supported on Trident2 without any API sequence changes. There are some known limitations that will be addressed in future SDK releases as listed below.

- WarmBoot is not supported for IFP. Warmboot support is planned once the compression modes are also supported.
- When the policers are configured in global mode, meter parameters will be set per pipe. This will cause rate limiting logic to apply to ports per pipe instead of globally. This issue is under review with the hardware design team

IFP Per Pipe Support

This release supports IFP configuration per pipe. The mode selection between global and per pipe will be controlled by mode API (`bcm_field_group_oper_mode_set`). The pipe is selected based on ingress port bit map (ports field in `bcm_field_group_config_t`) passed by the user during group creation. The port bit map passed by the user should be within a single pipe. Once per pipe mode is enabled, the rules will be installed only on that particular pipe.

IFP Compression - Beta

A number of packet header fields are large in width but their number space is sparse. Manual compression introduces a field compression capability to enable narrow rules. The compression tables available in BCM56960 can be programmed using the newly introduced manual compression field API.

IFP Preselection - Beta

IFP Pre-selector module provides flexible pre-qualification capability which helps to segregate the rules based on the packet characteristics by having the first level of qualification at the beginning of the field processor stage. In previous XGS implementations, the keys fed to the Filter processor were picked solely based on the ingress port number. The Preselector selection allows the user to pick the keys for the IFP based on a number of pipeline decisions and packet characteristics, leading to more compact and precise keys that suit the user's application.

It can be achieved by creating the Preselection entry ID and adding qualifiers to it, and then associating the Preselection ID to a group during creation.

Note:

Compared to the last SDK-6.4.4 EA8 release, the IFP Qualifier test status is better. Refer to Field Qualifiers ([page 458](#)) in (Status of BCM56960 IFP Qualifier Testing) ([page 458](#))

LEGACY FEATURE STATUS

The following list of Legacy features is supported in this release at Beta quality.

Table 5: Tomahawk Feature Status

Feature	Status	Note
Supported Legacy Feature		
Alpm legacy	Beta	
Cosq	Beta	
Cut Through Mode	Beta	
ECMP-Load Balancing	Beta	
Port extender	Beta	
Failover	Beta	
FCOE	Beta	
Flex stat	Beta	
Higig Proxy	Beta	
Field - Legacy	Beta	
IPMC Legacy	Beta	
L2	Beta	
L2GRE	Beta	
L3	Beta	
Link	Beta	
MIM	Beta	
Mirror	Beta	
MPLS	Beta	
Multicast	Beta	
NAT	Beta	
NIV	Beta	
Packet Rx/Tx	Beta	
PIMDIR	Beta	
Port	Beta	
Proxy	Beta	
QoS	Beta	
Oversubscription	Preview	
Rate	Beta	
Legacy Rtag 7 - flex hashing	Beta	
SER Correction	Beta	
Stack	Beta	
STAT	Beta	
STG	Beta	
Switch Controls	Beta	
Time	Beta	

Table 5: Tomahawk Feature Status

Feature	Status	Note
TRILL	Beta	
Trunk	Beta	
Tunnel	Beta	
Virtual/Subport	Beta	
VPLAG	Beta	
VLAN	Beta	
VXLAN	Beta	
wlan	Beta	
WRED	Beta	

FEATURES NOT YET SUPPORTED

The following features are not yet supported.

Table 6: Features Not Supported in this Release

Feature	Release with Support
SER Warmboot	SDK 6.4.5
IPMC Enhancements	SDK 6.4.5
VP Switch Routing	SDK 6.4.5
UDF enhancements for Hashing	SDK 6.4.5
MPLS LSR	SDK 6.4.5
General rate enhancements	SDK 6.4.5
Field Enhancement - Exact Match Group	SDK 6.4.5
Field Enhancement - Per Pipe Support for EFP	SDK 6.4.5
MMU Warmboot	SDK 6.4.6
Hash Spray	SDK 6.4.6
Visibility - Aggregate and PFC Monitors	SDK 6.4.6
Field Enhancement - Exact Match Qual/Action	SDK 6.4.6
Field Enhancement - IFP New Qualifiers	SDK 6.4.6
Field Enhancement - Range Checkers	SDK 6.4.6
DTCP	SDK 6.4.6

STATE OF BCM56960 PHY MICROCODE:

The PHY microcode for Falcon PMD delivered in this release is only meant for bringup and initial testing purposes. It is not guaranteed to withstand stress testing (such as overnight runs on every port). It is partially regressed to be reasonably stable for typical process and temperatures. Its adaptation may vary slightly between ports depending on channel variations.

All Eagle PMD PHY Microcode is development complete including manufacturing considerations.

The following table identifies features supported in this release

Table 7: BCM56960 SerDes Feature Status

Feature	Status	Note
Eagle: UCODE	D10E_04	
Falcon: UCODE	D109-1	
SerDes PM4x10	Forced Speeds	
10G-XFI	Passed (Gloop and p2p)	CPU generated, TR72, Test Generator
10G-dual	Passed (Gloop and p2p)	Traffic is CPU generated
10G-XAUI	Passed (Gloop)	Traffic is CPU generated
SGMII	Passed (Gloop and p2p)	Traffic is CPU generated
SerDes PM4x10	Auto Negotiated Speeds	
10G-KR (CL73)	Passed, P2P	Traffic is CPU generated
SerDes PM4x25	Forced Speeds	
10GBASE-XFI	Passed (Gloop, PRBS (port to port, Traffic(port to port)))	CPU generated, TR72, + Test Generator
4x25	Passed (Gloop, PRBS (port to port, Traffic(port to port)))	Traffic is CPU generated
40G-MLD	Passed (Gloop, PRBS (port to port, Traffic(port to port)))	CPU generated, TR72
D40G (Dual lane)	Passed (Gloop, PRBS (port to port, Traffic(port to port)))	CPU generated, TR72, and Traffic Gen.
100G-MLD	Passed (Gloop, PRBS (port to port, Traffic(port to port)))	CPU generated, TR72, and Traffic Gen.
Forced speed Link Training	Preliminary validation of 10G and 25G channels.	Not all channels available for testing. Link training convergence seen in short/medium copper traces and short cables.
106HG	Passed (Port to port)	CPU generated, TR72
21G/42G/53G (dual lane)	Passed (port to Port)	CPU generated traffic only.
11G/27G (single)	Passed	CPU generated traffic only.
SerDes PM4x25	Autoneg Speeds, CL73(BAM)	
10G KR	Passed	Traffic is CPU generated
25G KR/CR (BAM)	Passed	Traffic is CPU generated
50G KR2/CR2(BAM)	Passed	Traffic is CPU generated
100G KR4/CR4	Passed	Traffic is CPU generated
40G KR4/CR4	Passed	Traffic is CPU generated
40G KR2/CR2(BAM)	Passed	Traffic is CPU generated
FEC on forced speed and autoneg		
10G/25G/27G (single lane) forced speed	Passed	Traffic is CPU generated
20G/40G/50G/21G/42G/53G (dual lane) forced speed	Passed	Traffic is CPU generated
40G/42G/100G/106G (4 lane) forced speed	Passed	Traffic is CPU generated
All the applicable Autoneg speed on BCM56960	Passed	Traffic is CPU generated
EyeScan Types 1,2,3	Passed	

BCM56260 (SABER2)

SABER2

The BCM56260 family of switches is aimed specifically at the cost/power sensitive Carrier Ethernet Access (EAD/NID), Microwave backhaul, Compact PTN and Compact Cell-site-Router markets. The switch offers 12x2.5G + 2x10G SerDes with support for 10M/100M/1000M, 2.5G and 10G-XAUI Ethernet speeds on Front panel Viper SerDes and 10M/100M/1000M, 2.5G, 10G-XAUI and 10G Serial Ethernet speeds on Uplink Eagle SerDes ports. The switch provides 50Gbps of Ethernet switching.

This release contains beta quality support for the BCM56260 A0 device.

This release supports the following device SKUs in the following port configurations:

- BCM56460 (Config Option 1) : 24x1G + 4x10G
- BCM56460 (Config Option 2) : 6x10G + 4x1G
- BCM56260 (Config Option 1): 12x2.5G + 2x10G with 12.5G Loopback Port
- BCM56260 (Config Option 2): 3xF.XAUI + 2x10G + 2x1G with 2.5G Loopback Port

SABER2 SDK STATUS

Changes in this release generally are further reduction of test failures.

ADDITIONAL FEATURES SUPPORTED

New Features

MMU IPMC Protection

The feature allows user to enable an alternative next hop to every next hop that was replicated in the MMU for Multicast traffic so that it can provide similar failover mechanism of unicast traffic which is implemented in IP.

LEGACY FEATURE STATUS

The following list of Legacy features is supported in this release at Beta quality.

Table 8: Status of Saber 2 features (BCM APIs)

Features	Status	Note
Auth	Beta	
BCMI2C	Beta	
COE	Preview	
COSQ	Beta	
Failover	Beta	
Field Processor	Beta	
Hashing - RTAG7 - Flexible Hashing	Beta	
IPMC	Beta	

Table 8: Status of Saber 2 features (BCM APIs)

Features	Status	Note
IP Tunnel	Beta	
L2	Beta	
L2 Multicast	Beta	
L3	Beta	
Linkscan	Preview	
Mirroring	Beta	
MIM	Beta	
MPLS/ VPLS/VPWS	Beta	
NIV	Beta	
OAM (Legacy)	Beta	
QoS	Beta	
Packet Rx/Tx	Beta	
Port	Beta	
Port Extender	Beta	
Proxy	Beta	
Rate Control	Beta	
Service Meters	Beta	
SPB	Beta	
STACK	Beta	
STAT and Flex Stats	Beta	
STG	Beta	
Switch	Beta	
Time	Beta	
TR101/156/200	Beta	
Trunk	Beta	
Tunnel	Beta	
VLAN	Beta	
Warmboot	Beta	

FEATURES NOT YET SUPPORTED

The following features are not yet supported.

Table 9: Features Not Supported in this Release

Feature	Release with Support
MPLS LM/DM - New - SDK needs also Ukernel 4.2.1	SDK 6.4.5
LED Processor	SDK 6.4.5
Flex IO	SDK 6.4.5
SAT - New	SDK 6.4.5
Port Extender	SDK 6.4.6
NIV	SDK 6.4.7

STATE OF BCM56260 PHY MICROCODE:

All Eagle PMD PHY Microcode is development complete including manufacturing considerations. The Viper PHYs don't have microcode.

Table 10: BCM56260 SerDes Feature Status

Feature	Status	Note
SerDes PM4x10	Forced Speeds	
Eagle: UCODE	D10E_04	
10G-XFI	Passed (Gloop, P2P, TfcGen)	Tested with CPU generated and IXIA tfc.
1G	Passed (Gloop, P2P, TfcGen)	Tested with CPU generated and IXIA tfc.
Autoneg Speeds		
CL73	Passed (P2P)	1G/10G only Tested with CPU traffic.
Eyes, Types 1/2/3	Passed	
Viper-Xaui	Forced Speeds	
1G	Passed (Gloop, P2P, TfcGen)	Tested with CPU generated and IXIA tfc. Note the speed must come up as 1G.
100M	Passed (Gloop, P2P)	Tested with CPU generated. Speed change from 1G mode.
10M	Passed (Gloop, P2P)	Tested with CPU generated. Speed change from 1G mode.
100FX	Passed (Gloop, P2P)	Tested with CPU generated. Speed change from 1G mode.
2500M	Passed (Gloop, P2P)	Tested with CPU generated. Note the speed must come up as 2500M.
10G-XAUI	Passed (Gloop, P2P, TfcGen)	Tested with CPU generated and IXIA tfc. Note the speed must come up as 10G.
AutoNeg Fiber (1G)	Passed	Tested with CPU generated packets after autoneg.
AutoNeg SGMII (10M/100M/1G)	Passed	Tested with CPU generated packets after autoneg.
Viper-SGMII	Forced Speeds	
10M/100M/1G	Passed (Gloop, P2P, TfcGen)	Tested with CPU generated and IXIA tfc.

BCM88670-FAMILY BRING-UP RELEASE

This release is a bring-up version for the BCM88370-Family and BCM88670-Family product line, following previously released bring-up EA versions, supporting the following SKUs:

- BCM88375
- BCM88378
- BCM88675

Compared to 6.4.4-EA9 bring-up EA versions, the 6.4.4 release delivers better validation coverage and integrativity with other BCM devices in the same SDK.

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

VALIDATED FEATURES

Unless stated otherwise, features validation for this release covered legacy (BCM88660-compatible) modes, and not the new modes added for BCM88370-Family and BCM88670-Family devices.

BASIC DATA PATH, CONNECTIVITY AND TRAFFIC MANAGEMENT FEATURES

Access, SKUs, Interfaces, Core, CPU

- Register & Memory access including DMA.
- Supported SKUs: Qumran-MX 88375, 88378. Jericho 88675.
- Device core mode: dual-core symmetrical
- CPU RX/TX (packet DMA)
- Interfaces: OCB, DDR4@3.2GHz, GDDR5@2.4GHz
- Recycling interface

Network Interface

NIF and fabric PLL: both 156.25MHz. and 125MHz NIF includes:

- 10GE: KR/XFI, RXAUI, XAUI
- 25GE: KR
- 40GE: XLGE, XLGE-2
- 100GE: CAUI

Fabric (BCM8867x) includes :

- CL72
- All supported rates were validated
- Packet Cell Packing (PCP)

Forwarding

- Unicast
- Inbound & Outbound mirroring
- Multicast - Ingress, Egress using bitmap group & linked list
- FMQ scheduling and Fabric Multicast (BCM88675)
- TDM unicast, bypass mode
- LAG (Trunk) destinations

CosQ

- End-to-end scheduling
- Ingress and Egress compensation
- VOQ, ISQ ingress queue creation & ranges configuration - symmetric & asymmetric modes. Mapping by-destination and by-flow-id



- VSQ: type A-D creation (types E, F currently not validated)
- Traffic Class configuration
- WRED, Tail Drop, Fair-adaptive tail-drop
- Credit request profile configuration excluding credit worth.
- Credit watch dog
- Egress queuing: number of priorities per port: 1/2/8
- Undersubscribed CoE (mapping multiple ports, no Flow Control)

Connectivity (BCM88675)

- FE3200 connectivity (UC and MC traffic)
- FE1600 connectivity
- Coexist with Arad
- Fabric pipes mapping: Single pipe, Dual pipe as UC, MC, Triple pipe as UC, high priority MC, low priority MC

Counters

Counter Processor (CRPS):

- Validate sources: Ingress VSI, Ingress OAM, Egress OAM, Ingress VOQ Ingress VSQ, Egress PMF
- Validate format: Packets and Bytes, Packets, Bytes, Packets and Packets, Max VOQ size

SNMP counters: Internal (diagnostics) counters

PACKET-PROCESSING FEATURES AND SUPPORTED PROTOCOLS

Basic legacy features are supported

- Packet forwarding - ITMH, Force-FWD, L2, QinQ (P2P,MP), VPLS, MPLS
- Field Processor
- L3 features with the new Routing table (KAPS) - IPv4 RPF, IPv4 tunnel initiator / tunnel termination, IPv4 with MPLS tunnel.

Overlay (VXLAN/L2GRE)

- VXLAN/L2GRE basic scenarios (Unicast) were tested
- Routing over-overlay: Tested ROO-VXLAN unicast packet using Host format.

OAM/BFD

- OAM - Down/Up MEP accelerated and non-accelerated works
- Ethernet OAM CCM
- BHH MPLS
- BFD IPv4 single hop and multi hop
- BFD PWE VCCV Type1

Enhanced MPLS termination

MPLS termination enhanced is introduced, GAL label termination was validated.

KNOWN ISSUES

NETWORK INTERFACE

- Only the interfaces indicated above are validated. All network interface types will be validated for 6.4.5 Release. 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

BRING-UP GUIDELINES

REFERENCE DOCUMENTATION

Multiple documents describing relevant HW and SW aspects are available, including:

SW DOCUMENTATION

- 88660_88670-AG1xx: API compatibility between Arad+ and QumranMX/Jericho. This document summarizes the differences in the BCM SDK software interface between BCM88660 and BCM88670 devices.
- 88670-PG2xx: Traffic Manager Theory of Operation This document describes theory of operation and provides driver reference for the BCM88670 device series.
- 88670-PG1xx: BCM-API Packet Processing: Theory of Operation This document describes BCM-API for the BCM88670 device packet processing capabilities, and how to configure it for networking applications.
- 88660-AN100-R: SDK Compatibility Guidelines 6.3.x to 6.4.x

HW ARCHITECTURE SPECS:

- 88670-AG2xx: Jericho/Qumran-MX Traffic Management Architecture Guide. This document describes the BCM88670 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 BCM88670 traffic management architecture provides.
- 88670-AG3xx: BCM88670 Packet Processor Enhancements This document is intended for system architects and anyone else seeking an understanding of the features and capabilities that the BCM88670 Packet processing architecture provides.
- 88670-AG4xx: OAM Architecture for BCM88670 This document provides an overview of the OAM functions supported in BCM88670 and explains how each OAM function is supported. It is intended for use by system architects and anyone else seeking an understanding of the OAM features in the BCM88670 switching device.

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
- `diag reassembly_context` - display reassembly and port termination context
- `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.
- Eyescan not validated yet.

DRAM:

- `diag buffers` - display the buffers usage on the device (Dram and OCB)
- `TR 140` - bist test, shows if error occurred in dram, for more info see UM

CosQ:

- `diag cosq qpairs` - display qpairs related diagnostics.
- `diag cosq print_flow_and_up` - display packet flow info (scheduling scheme).
- `diag cosq local_to_sys` - translate local ports to system ports.
- `diag egr_congestion` - display egress congestion statistics.
- `diag egr_calendars` - display egress calendars info.
- `cosq conn egr/ing` - display cosq connection for egress/ingress.

Fabric:

- `fabric link <'all' / link_id>` - display fabric link status
- `fabric connectivity` - display fabric interface connectivity information
- `fabric reachability <modid>` - display reachable links to modid
- `fabric link_config` - display current used fabric link configuration
- `fabric mesh_topology` - internal diagnostic of `mesh_topology` block
- `fabric rx_fifo` - internal fabric rx fifo's status
- `fabric thresholds <threshold_type> [pipe=<0/1/2>] [link=<link_id>]` - display the configured FIFO thresholds Supported threshold types: RCI, LLFC
- `fabric queues` - display queues status

Packet DMA:

- "tx" shell command

PORT MAPPING

There are some differences in port indexing and mapping compared to BCM88660, required to support dual-core. Please read the driver reference for "System Port Handles" section in the UM (88670-PG2xx).

Main differences highlighted below:

- Core association must be defined upon initialization, the bolded attributes in the SOC property below are new compared to BCM88660:



- `ucode_port.port<logical-port-id>.<unit> = <Interface_name>.<channel_num>:core_<core-id>.<tm-port-id>`
- Modport: this is practically the only place where you need to use the tm-port-id and not the logical port. This is since the tm-port-id is the actual tm-port that will be stamped on FTMH, and the modid is the fap-id to be stamped on the FTMH. Modport is defined by `BCM_GPORT_MODPORT_SET(portmod_gport, module_id, local_tm_port) .`
- The two cores of a device have consecutive FAP IDs. The FAP ID that is set/get using APIs is the FAP ID of core 0, and the FAP ID of core 1 is the FAP ID of core 0 plus 1. When setting a MOD port, remember to use the FAP ID of the correct core.
- Although not mandatory, at this stage it is safer to use TM port ID equal to the logical port ID. This is since feature validation coverage for bring-up releases is partial. In future releases, as validation coverage will increase; this disclaimer will not be needed.

DRAM INTERFACE

It is recommended to start the bring-up in "OCB-only" mode.

This allows starting c up. SOC properties: set `bcm886xx_ocb_enable=1` and `ext_ram_present=0` to work in OCB-only mode.

- When working with DRAM, change `ext_ram_present=<2|3|41|42|6|8>` according to the number of DRAM interfaces - see details in the UM (88670-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.
- It is recommended to use the following sequence for the Dram Bring-Up: Initialize the driver with the following soc properties: `bist_enable_dram =0 ext_ram_present ==<2|3|41|42|6|8> ddr3_auto_tune=3`.
- This allows the driver to load without performing Shmoo (ddrphytune) or dram BIST (tr 140). Once the driver has loaded, try to run independently Shmoo for each interface and then dram BIST for each interface.
- Once this phase is completed, change soc the properties as follows: `bist_enable_dram =1 ddr3_auto_tune=2`
- Initialize the driver with these settings. This step may take up to 20 minutes because Shmoo sequence (DRAM PHY calibration) is performed. After completing initialization, run a longer BIST to verify that DRAM interfaces are clean.
- Next step is to save and restore the calibration parameters, as described 88670-PG2xx.
- For example, using the Diag-shell application the BCM shell command to display calibration parameters is `'config show combo28_tune'`.
- Prints verbosity can be increased per-module to control the amount of debug information. For example for the Dram using BCM Diag shell reference application, the commands are: `'debug soc ddr verb'` and `'debug soc dram verb'`

NETWORK AND FABRIC 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 below the correct swap-mapping is 0x0321.

E2E SCHEDULING

When allocating SE using `bcm_cosq_gport_add`, the root port of the scheduling scheme must be passed in as E2E PORT gport. This wasn't required in BCM8866x, but is mandatory when working in dual core BCM8867x/BCM8837x.

- For setting VOQ connectors, it is recommended to use the new API: `bcm_cosq_voq_connector_gport_add`
- The legacy API `bcm_cosq_gport_add` is also supported but only under the constraints below:
- Single core: no constraints
- Dual core: this API can only be used with "WITH_ID" flag set, i.e. explicit ID allocation. In this case, the core is specified in the gport input parameter. The gport generated using the legacy 'BCM_COSQ_GPORT_VOQ_CONNECTOR_SET' macro must be now generated using the 'BCM_COSQ_GPORT_VOQ_CONNECTOR_CORE_SET' macro.

SYMMETRIC QUEUING MODE

Typically, when working in dual-core mode, symmetric queuing mode is used. This mode assumes that VOQ allocation and attributes are identical per destination, regardless the source interface through which the packet entered the device (the interface defines the relevant core on the ingress device).

Ingress Configuration:

Symmetric mode is set using the SOC property

- `device_core_mode =SYMMETRIC`

In this mode, only VOQs on core-0 of the source FAP device must be explicitly set. The driver implicitly allocates VOQs on core-1 of the source FAP. All VOQ and VSQ attributes are applied to both cores unless explicitly directed otherwise.

In case ingress queuing is symmetric (default mode), the connection to both cores can be done automatically.

Egress Configuration:

Each consecutive range of 1K VOQ connectors can be split into two halves containing 512 VOQ-connector IDs each. First 512 IDs will be allocated mapped to a source-FAP VOQs on core-0 of the source device. Second 512 IDs will be reserved (i.e. customer application cannot use them when working with "WITH_ID" flag), and will be allocated by the driver implicitly and mapped to source-FAP VOQs on core-1 of the source device.

In other words, for any VOQ connector with id 'x' allocated for source-FAP core 0, additional VOQ connector with id 'x+512' will be implicitly allocated for source-FAP core 1.

Such symmetric VOQ connectors must be allocated from special range, which are marked per 1K using soc property:

```
dtm_flow_nof_remote_cores_region_<1-127>=2
```

This property marks to the SW that each connector in this 1K range represents in fact 2 connectors, one per core of the source-FAP device. In this range, connectors with id x where $(x\%1024 < 512)$ are legal and others are not.

Note:

last 1K region, identified as `dtm_flow_nof_remote_cores_region_128`, must be 1 (single-core mapping).

Once the regions are set, call `bcm_cosq_voq_connector_gport_add` with `nof_remote_core=2` to allocate VOQ connectors in two-core mode. When using `WITH_ID` flag, the provided VOQ-connector ID must comply to `(ConnectorID%1024<512)` rule.

PROGRAMMABLE ITMH

ITMH (Ingress Traffic Management Header) has two formats: a hard-coded format, used for example in BCM88660, and a programmable format, set by default in 6.4.4 for BCM88670/5.

With the programmable format, all the abilities of the Field Processor are available also to ITMH packets. In the reference application, the BCM88670/5 ITMH is implemented via the Field groups APIs. The user can modify the ITMH structure by modifying the reference application. However, both OLP (Off-Load Processor for MAC Table learning) and the OAM Processor are always using the default ITMH as detailed in 88670-TI100-R.

Set the new `itmh_programmable_mode_enable` SOC property (set by default in BCM88670/5) to work with the programmable format. Unset this SOC property to work with the ITMH format used by BCM88660.

Note: OAM was validated only using the hard-coded format (`itmh_programmable_mode_enable=0`).

MAC TABLE DEFAULT LEARNING MODE

The MAC table learning mode default value is changed from egress-independent (`BCM_L2_EGRESS_INDEPENDENT`) in 6.3.x to ingress-distributed (`BCM_L2_INGRESS_DIST`) in 6.4.x for BCM88650/60.

EXPOSING THE TRAP HW ID

In Rx Trap module, the ingress trap HW ID was not exposed. The `trap_id` parameter returned in `bcm_rx_trap_type_create` is an abstract SW ID. In 6.4, by default, the returned value is the HW ID instead.

The previous mode can be restored by resetting the `bcm886xx_rx_use_hw_trap_id` SOC property.

BOUNCE-BACK FILTER NEW IMPLEMENTATION

In the overlay application (VXLAN, L2GRE, Trill), bounce-back filter was introduced to filter core packets back to the core. In BCM88670 Filter is implemented using Orientation filter. CINTs applications were adjusted accordingly.

TRILL APPLICATION

In BCM88670, Trill application was tested only in device mode soc property: `trill_mode = 2`

OAM TRAP

For trapped OAM packets the OAM-LIF should be extracted from the FTMH.MCID-or-OutLIF (in case of Up MEPs) or PPH.InLIF (in case of Down MEPs). Direction of the endpoint may be deduced from the trap code.

FHEI.Trap-Code-Qualifier no longer contains information about the OAM-LIF.

BCM88950 (FE3200) BETA RELEASE

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

This is a beta release of BCM88950 driver. It adds validation coverage and maturity on top of the previous preview releases, including connectivity to the BCM88670-Family devices.

The following BCM88950 exclusive features were added compared to the 6.4.3 version:

Bug fixes and improvements compared to 6.4.3:

- New Falcon SerDes firmware version was integrated. Together with few improvements of port library, few SerDes issues were fixed (most of them relates to link training)
- For future compatibility, coexist of BCM88950/BCM88670 with next generation devices, the system reference clock should be configured to 1200MHz. The default values of the relevant soc property (`system_ref_core_clock`) changed to 1200MHz.
- Driver crash after warm reboot fixed.
- Default value of SerDes polarity changed to be 0 (to be aligned with other DNX devices). Relevant SoC properties: `"phy_rx_polarity_flip"` and `"phy_tx_polarity_flip"`.
- Few minor fixes of port PCS (Physical Coding Sublayer).
- Added support to run HW snake test ("tr 131") with different PCS (Physical Encoding Sub-layer). Note that few interrupts might occur (and will not affect the test).
- Few minor fixes of counters diagnostic.
- FE3200 CPU date cells feature fixed.
- "Linkscan MDIO" test (tr 60) fixed.
- Source Routed Cells feature fixed.
- BCM88950 standalone compilation fixed.
- SoC Property `"serdes_preemphasis"` format fixed (preTap(03:00), mainTap(10:04), and postTap(15:11)).

BCM88650 (ARAD), BCM88660 (ARAD+) RELEASE

This release is the continuation of 6.4.3 for the BCM88650-Family and is backward compatible with the 6.3.X features (except Warm-boot).

BCM82764 BRINGUP BETA 1 RELEASE

BCM82764 offers Dual 40GbE channels with Dual 40GbE XLPPI to XLAI/Quad 40GbE PMA reverse bit MUX to convert 2x20G to XLPPI (4x10G) /CAUI4 to CAIU4/Octal 10GbE SFI to XGFI port. Firmware version is D008.

FUNCTIONALITY SUPPORTED:

Link status, Speed and Interface configuration, port enable/disable, Polarity inversion, Power set/get, PRBS, Eyescan, PLL sequencer restart, PHY diagnostics, PHY dump and Rx PMD lock.

DRIVER LIMITATION:

- Speed change is only supported through config.bcm. Speed switch through flex port is not supported in this release.
- 10G Pass Through with BCM84793 compatible mode is not supported.
- Chip issues reset to all the ports who share the same PHY-ID (MDIO ADDR)
- TX training may not finish on some 10G lanes (line side)



BCM82381 AND BCM83073 - BRINGUP RELEASE

Driver support for 82381 and 82073 devices has been added in this release suitable for bringup.

- 100G CR4, 40G SR4/LR4/CR4/ , 10G SR/LR, 50G KR2, 25G KR and 20G KR2

FUNCTIONALITY SUPPORTED:

PHY Reset(Hard/soft), Link status, Speed and Interface configuration, port enable/disable, remote/digital loopback, Polarity inversion, PRBS, Tx squelch, TX set/get, Rx set/get, Auto negotiation, FEC enable/Disable, PHY dump, DSC, PLL sequencer restart, Eyescan(line and system), Force Tx Training, Interrupts, Link monitor(PCS/FC), Rx PMD lock and cross switch lane map, analog parameters (pre-emphasis, driver current , TAPS), Module controller operations on QSFP module, configuraton of IEEE FIFO skew offset, AN link qualifier

KNOWN LIMITATIONS:

- Observing errors when using PRBS 31 and PRBS 58 in 100G and 25G
- Re-timer mode not supported

BCM82780 BRINGUP RELEASE

BCM82780 is an octal-channel, 10GbE SFI-to-XFI PHY that incorporates an EDC. The device supports two full-duplex 40GbE ports. Firmware version is 1e.

Bringup level driver support for 82780 device has been added in this release.

Driver supports 10G and 40G SR4/LR4/CR4 speeds,

FUNCTIONALITY SUPPORTED:

- Firmware download, Link status, Speed and Interface configuration, Polarity Inversion, PRBS and Rx PMD Lock

KNOWN LIMITATIONS:

- 10G to 40G to 10G switching is ot tested, Tested 10G and 40G only by changing port map in config.bcm

THINGS TO NOTE

This section lists items that require special attention.

MINIMUM VXLAN VPN ID FOR TRIDENT2 PLUS AND TOMAHAWK

VPN ID 0x7000 is allowed when calling `bcm_vxlan_vpn_create` with `BCM_VXLAN_PORT_WITH_ID` on BCM56850. It is not allowed for BCM56860 and BCM56960 due to VPN ID 0x7000 is reserved on those devices.

SCACHE SIZE REQUIREMENTS FOR TOMAHAWK

In Tomahawk the field related software data structures is saved into the s-cache and software recovers and link them back after warm boot. The s-cache size needs to be increased to save all the data structures available in field module. Software depends only on the saved data for the recovery of IFP configurations after warm boot in tomahawk.

The size of the s-cache will change with the configurations used in IFP. Below is the memory requirement for 3 different configurations.

128 groups with 50 qualifiers per group, 24K entries, 5 actions per entry ~33 MB

128 groups with 50 qualifiers per group, 24K entries, 10 actions per entry ~37 MB

128 groups with 50 qualifiers per group, 24K entries, 15 actions per entry ~42 MB

SCACHE SIZE REQUIREMENTS FOR DEVICES IN LEGACY IFP SUPPORT

In warmboot design, the software data structures are saved into the sCache during warmboot sync. When the SDK boots in warmboot mode, it recovers the data structures from sCache and initializes the structures. The sCache size is calculated based on total size of various data structures, considering the max value of every field in the data structure.

In Saber2, the total scache required for all modules together is 573408 bytes

In Greyhound, the total scache required for all modules together is 380188 bytes

BCM54 API DEPRECATION

BCM54 APIs have not been enhanced or supported for newer devices since SDK-5.10.2. Legacy BCM54 APIs, supported in SDK-5.10.2 will be deprecated starting with SDK-6.3.5 release. Customers are encouraged to transition from BCM54 APIs to their equivalent BCM APIs. Please contact Broadcom application support for any help in the transition.

SPN_PHY_PORT_PRIMARY_AND_OFFSET

Setting of the config property `spn_phy_port_primary_and_offset` is absolutely required for the following PHYs. BCM54880E BCM54680E BCM54682E BCM54685E BCM54640E BCM542XX

IP ROUTE LOOKUP

If a packet destination IP lookup falls in between 2 route prefixes having a common prefix part and differing lengths, then the lower route prefix, which is the correct match, may not always be returned correctly.

BCM FIELD QUALIFIER TUNNEL TYPE

The enumeration type `bcm_field_TunnelType_t` has changed its values between SDK-6.2.x, SDK-6.3.0 and SDK-6.3.1. The implication is that legacy field qualifier support for `bcm_field_qualify_TunnelType` in XGS devices running SDK-6.3.1 and later has been broken. Features such as RPC between systems running SDK-6.2.x or SDK-6.3.0 on one and SDK-6.3.1 (or later) on another will not work properly for the BCM field qualifier tunnel type. This issue is being addressed in SDK-6.3.5, SDK-6.4.0, and later releases.

WARMBOOT: VALIDATED WARMBOOT UPGRADES.

Following warmboot upgrades have been validated in this release.

Table 11: Validated Warmboot upgrades

Software upgrade Supported	
6.4.3 to 6.4.4	Yes
6.3.11 to 6.4.4	Yes

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

- Trident2
- Trident+
- Triumph3
- Katana2
- Helix4
- Hurrican2
- Katana
- Raven
- Greyhound

Initial warmboot testing without upgrade support has also been conducted for:

- Tomahawk
- Saber2
- Trident2+

NEW SPINLOCK APIS

In the SDK release v6.3.8/SDK 6.4.1 a new lock mechanism -- spinlock was introduced to satisfy some requirements for protecting small critical sections more efficiently. The spinlock mechanism is applicable to the scenario in which the critical section to be protected only contains simple operations, such as inserting or removing nodes from a linked list, increasing or decreasing shared data. The data structure and interfaces of the spinlock are defined at SAL layer in SDK as follows,

```
typedef struct sal_spinlock_s {
    char spinlock_opaque_type;
} *sal_spinlock_t;

sal_spinlock_t sal_spinlock_create(char *desc);
int sal_spinlock_destroy(sal_spinlock_t lock);
int sal_spinlock_lock(sal_spinlock_t lock);
int sal_spinlock_unlock(sal_spinlock_t lock);
```

These primitives have been implemented for vxWorks and Linux. The changes are available in `src/sal/core/linux/sync.c` for Linux kernel mode, `src/sal/core/unix/sync.c` for Linux user mode and `src/sal/core/vxworks/sync.c` for vxWorks. Customers who use different OSes will need to make similar implementation in their OS specific SAL layer source files. If additional information is needed, please refer to the field alert document "Spinlock Application Note" or contact your Field Support staff.

BCM56440 FAMILY OF DEVICES

TCAMs in BCM56440 family of chips were not included in the MEMSCAN until this (SDK-6.4.4) release. Due to a customer request, the SER for this family of chips is enhanced in this release to include protection for the TCAMs. Given the way the parity enable controls are configured in different releases, upgrading from previous releases to this releases would generate false memory error reports on the TCAM memories.

As a remedy, users can disable parity or/and memscan on the TCAM memories using the relevant configuration properties which will be equivalent to what previous releases used to offer for these devices. Error recovery on the TCAMs can be enabled by doing a cold reset of the system. If SER on the TCAMs is desired right after the upgrade, users may need to patch the changes from SDK-55917 on to previous releases.

NEW DEVICES AND SYSTEMS

For any given SDK release, support for certain devices may be provided in Preview or Supported status. Devices in preview status are provided to allow early integration of the customer's application with the SDK APIs that support that device. This software has not been tested on the physical target device and should not be expected to fully function.

Devices in "Supported" status 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.

Table 12: Supported Switch Devices

Family	Devices	Description
BCM56340	BCM56345	Twister . 12xF.QSGMII + Flex[4x10] + 1GE
BCM56340	BCM56346	Twister . 7xF.QSGMII + Flex[4x10] + 1GE
BCM56240	BCM56245	Saber SKU- 2x (10GbE/4x 1GbE/4x 2.5GbE) + 2x 10GbE/12GbE/13GbE, IEEE 1588 enable
BCM56240	BCM56246	Saber SKU - 10x 1GbE/2.5GbE, IEEE 1588 enabled
BCM56640	BCM56543	Apollo2+ SKU -100GE/3xF.HG[42] + F.HG[127] + 1GE 415MHz
BCM56450	BCM56458 B0	8xGE + 2xF.XAUI
BCM56450	BCM56452 B0	24xGE + 4xF.XAUI
BCM56450	BCM56454 B0	8xGE + 2 x F.XAUI
BCM56260	BCM56260_A0	50 Gbps Integrated Lower Power Carrier Ethernet Access Switch
BCM56460	BCM56460_A0	24-Port 1GbE Multilayer Switch with Four 10GbE/HiGig? Uplink
BCM56860	BCM56860_A0	104x 10GbE/32x 40GbE/8x 100GbE Multilayer Switc
BCM56960	BCM56960 A0	32x100 GbE/64x40GbE/128x10 GbE Multilayer Switch
BCM53400	BCM53402 A0	8 x 1G/2.5G/5G/10G
	BCM53412 A0	8 x 1G/2.5G/5G/10G
	BCM53454 A0	20 x 1G/2.5G + 4 x 1G/2.5G/5G/10G
	BCM53455 A0	20 x 1G/2.5G + 4 x 1G/2.5G/5G/10G with embedded ARM A9 processor
	BCM53422 A0	8 x 1G + 2 x 1G/2.5G/5G/10G
	BCM53424 A0	4 x QSGMII + 8 x 1G + 4 x 1G/2.5G/5G/10G (option1)
	BCM53424 A0	4 x QSGMII + 8 x 1G + 2 x 10G + 2 x HiGigDuo[13] (option2)
	BCM53424 A0	2 x QSGMII + 16x1G + 4 x 1G/2.5G/5G/10G (option3)
	BCM53426 A0	20 x 1G + 4 x 1G/2.5G/5G/10G
BCM56060	BCM56062 A0	2 x QSGMII + 16 x 1G/2.5G + 4 x 1G/2.5G/5G/10G (option1)
	BCM56062 A0	2 x QSGMII + 8 x 1G/2.5G + 2 x XAUI + 4 x 1G/2.5G/5G/10G (option2)
	BCM56062 A0	2 x QSGMII + 16 x 1G/2.5G + 4 x 1G/2.5G/5G/10G (option3)

Table 13: Preview Switch Devices

Family	Devices	Description
BCM53400	BCM53456 A0	4 x QSGMII + 8 x 1G/2.5G + 4 x 1G/2.5G/5G/10G (option1)
	BCM53456 A0	4 x QSGMII + 8 x 1G/2.5G + 2 x 10G + 2 x HiGigDuo[13] (option2)
	BCM53456 A0	2 x QSGMII + 16 x 1G/2.5G + 4 x 1G/2.5G/5G/10G (option3)
	BCM53457 A0	4 x QSGMII + 8 x 1G/2.5G + 4 x 1G/2.5G/5G/10G with ARM A9 (option1)
	BCM53457 A0	4 x QSGMII + 8 x 1G/2.5G + 2 x 10G + 2 x HiGigDuo[13] with ARM A9 (option2)



Table 13: Preview Switch Devices

Family	Devices	Description
	BCM53457 A0	2 x QSGMII + 16 x 1G/2.5G + 4 x 1G/2.5G/5G/10G with ARM A9 (option3)
BCM56060	BCM56063 A0	4 x QSGMII, 4 x 1G/2.5G/5G/10G (XFI)
	BCM56064 A0	4 x QSGMII, 8 x 1GbE, 4 x 10GbE (option1)
	BCM56064 A0	4 x QSGMII, 8 x 1GbE, 2 x 10GbE, 2 x HiGigDuo[13] (option2)
	BCM56065 A0	12 x 1G/2.5G/5G/10G + 12 x 1G/2.5G

Table 14: PHYs

Device	Driver Family	Description
BCM82381_A2	82381	Dual 100GbE with CAUI4 to CAUI4 / Dual 40GbE XLPPi to XLAUI/ Octal 10GbE SFI to XFI port. Firmware version D00C.
BCM82073_A1	82073	CAUI4-to-KR4 NRZ Backplane PHY/Octal 25G Backplane Retimer with 2x2 Crosspoint

Table 15: Preview PHYS

Device	Driver Family	Description
BCM84858_A0	84858	Quad 10GBASE-T Transceiver. Firmware version 1.01.04 (Bringup)
BCM84858_B0	84858	Quad 10GBASE-T Transceiver. Firmware version 00.02.02 (Bringup)
BCM82764_A2	82764	BCM82764 Dual 40GbE channels with Dual 40GbE XLPPi to XLAUI / Quad 40GbE PMA reverse bit MUX to convert 2x20G to XLPPi(4x10G) /CAUI4 to CAUI4 / Octal 10GbE SFI to XFI port. Firmware version D008.
BCM82780_A0	82780	10G Octal SFI-XFI PHY with 10GbE/40GbE and IEEE 1588 Support (Basic Bringup)
BCM84868_B0	84868	10GBASE-T Transceiver (Basic Bringup)

SUMMARY OF BCM CHANGES

This section summarizes BCM API changes in this release. Complete documentation is available in the Network Switching Software Programmer's Guide 56XX-PG644-R. (See section 2 earlier in this document for availability).

BCM FUNDAMENTALS

BCM MODULES

A new module BCM_MODULE_OOB has been added in this release

The BCM APIs are classified into API groups called 'modules'. The following table lists the BCM modules and their string equivalents.

Table 16: BCM Module IDs

BCM Module ID	String Equivalent
BCM_MODULE_PORT	port
BCM_MODULE_L2	l2
BCM_MODULE_VLAN	vlan
BCM_MODULE_TRUNK	trunk
BCM_MODULE_COSQ	cosq
BCM_MODULE_MCAST	mcast
BCM_MODULE_LINKSCAN	linkscan
BCM_MODULE_STAT	stat
BCM_MODULE_VIRTUAL	virtual
BCM_MODULE_COMMON	common
BCM_MODULE_MIRROR	mirror
BCM_MODULE_L3	l3
BCM_MODULE_STACK	stack
BCM_MODULE_IPMC	ipmc
BCM_MODULE_STG	stg
BCM_MODULE_TX	tx
BCM_MODULE_L2GRE	l2gre
BCM_MODULE_AUTH	auth
BCM_MODULE_RX	rx
BCM_MODULE_FIELD	field
BCM_MODULE_TIME	time
BCM_MODULE_SUBPORT	subport
BCM_MODULE_MPLS	mpls
BCM_MODULE_FABRIC	fabric
BCM_MODULE_MIM	mim
BCM_MODULE_POLICER	policer
BCM_MODULE_OAM	oam
BCM_MODULE_FAILOVER	failover
BCM_MODULE_VSWITCH	vswitch
BCM_MODULE_WLAN	wlan
BCM_MODULE_QOS	qos
BCM_MODULE_MULTICAST	multicast
BCM_MODULE_TRILL	trill

Table 16: BCM Module IDs

BCM Module ID	String Equivalent
BCM_MODULE_IPFIX	ipfix
BCM_MODULE_NIV	niv
BCM_MODULE_CES	ces
BCM_MODULE_PTP	ptp
BCM_MODULE_BFD	bfd
BCM_MODULE_REGEX	regex
BCM_MODULE_VXLAN	vxlan
BCM_MODULE_EXTENDER	extender
BCM_MODULE_FCOE	fcoe
BCM_MODULE_UDF	udf
BCM_MODULE_SAT	sat
BCM_MODULE_OOB	oob (New!)

New Macros for Analyzing/Constructing GPORTs ([page 35](#)) shows the new macros that are designated for working with GPORTs.

Table 17: New Macros for Analyzing/Constructing GPORTs

GPORT Macros	Macro Description
BCM_GPORT_MCAST_SUBSCRIBER_QUEUE_GROUP_SYSQID_SET(g, port_id, qid)	Sets multicast subscriber group type, port id and queue value
BCM_GPORT_MCAST_SUBSCRIBER_QUEUE_GROUP_SYSQID_GET(g)	Get portid from the multicast subscriber gport
BCM_GPORT_IS_LOCAL_CPU_IEEE	Returns 1 if local CPU for IEEE, else 0
BCM_GPORT_IS_LOCAL_CPU_HIGIG	Returns 1 if local CPU for HIGIG, else 0
BCM_COSQ_GPORT_VOQ_CONNECTOR_CORE	CoSQ gport, subtype connector, corresponding core
BCM_COSQ_GPORT_COMPOSITE_SF2_CORE	CoSQ gport, subtype second sub flow, corresponding core
BCM_COSQ_GPORT_SCHED_CIR_CORE	CoSQ gport, subtype CIR scheduling element, corresponding core
BCM_COSQ_GPORT_SCHED_PIR_CORE	CoSQ gport, subtype PIR scheduling element, corresponding core
BCM_COSQ_GPORT_VOQ_CONNECTOR_CORE_SET(g, cid, core)	Sets voq connector type and value for specified core
BCM_COSQ_GPORT_VOQ_CONNECTOR_CORE_GET(g)	Gets core value for voq connector type
BCM_COSQ_GPORT_COMPOSITE_SF2_CORE_SET(g_sf2, g, core)	Sets composite second sub flow type and value for specified core
BCM_COSQ_GPORT_COMPOSITE_SF2_CORE_GET(g_sf2)	Gets core field out of composite second sub flow type
BCM_COSQ_GPORT_SCHED_CIR_CORE_SET(g_cir, g, core)	Sets cir scheduling element type and value for specified core
BCM_COSQ_GPORT_SCHED_CIR_CORE_GET(g_cir)	Gets core field out of cir scheduling element type
BCM_COSQ_GPORT_SCHED_PIR_CORE_SET(g_pir, g, core)	Sets pir scheduling element type and value for specified core
BCM_COSQ_GPORT_SCHED_PIR_CORE_GET(g_pir)	Gets core field out of pir scheduling element type

New data type `bcm_oob_fc_rx_intf_id_t` is introduced in this release.

Table 18: New Broadcom Network Switching Basic Types

Type	Description
bcm_oob_fc_rx_intf_id_t	The Out-of-Band flow control receiving interface id. In BCM56960 there are 4 receiving interfaces (0-3).
bcm_class_t	8 bytes of class value.

CLASS OF SERVICE QUEUE CONFIGURATION

New COSQ control types are added for `bcm_cosq_control_set/get()` APIs.

Table 19: CoSQ Control Type Values

Value	Description	Arg value
bcmCosqControlPriorityPropagationEnable	Set priority propagation	value: 0-disable, 1-enable
bcmCosqControlPrioritySelect	Set high/low port priority for propagation	BCM_COSQ_HIGH_PRIORITY or BCM_COSQ_LOW_PRIORITY
bcmCosqControlFlowControlErrWatchdog	Set Watchdog period.	Number of usec
bcmCosqControlFlowControlErrHandle	Select Fc indication that will be sent when watch-dog error is reported	value: 0-xon, 1-xoff

New COSQ range configuration types are added for `bcm_cosq_control_range_set/get()` APIs.

Table 20: Range configuration type.

bcmCosqOcbCommittedMulticast_1	A range of MC IDs that are guaranteed to use OCB buffers If available. Range number 1.
bcmCosqOcbCommittedMulticast_2	A range of MC IDs that are guaranteed to use OCB buffers If available. Range number 2.
bcmCosqOcbEligibleMulticast_1	A range of MC IDs that are allowed to use OCB buffers If available. Range number 1.
bcmCosqOcbEligibleMulticast_2	A range of MC IDs that are allowed to use OCB buffers If available. Range number 2.

New threshold flags are added in this release for `bcm_cosq_gport_threshold_set/get()` APIs with the new field of priority in `bcm_cosq_threshold_t`.

Table 21: New Threshold flags

BCM_COSQ_THRESHOLD_POOL0	Set threshold value for shared Pool 0 resource
BCM_COSQ_THRESHOLD_POOL1	Set threshold value for shared Pool 1 resource

```

/* flow control / admission settings */
typedef struct bcm_cosq_threshold_s {
    ...
    int priority;                /* priority */

```

```
} bcm_cosq_threshold_t;
```

New configuration flags of flow control endpoint are added in this release for `bcm_cosq_fc_path_add()` API.

Table 22: Flow Control Endpoint flags

BCM_COSQ_FC_BDB	FC triggered by BDB resource
BCM_COSQ_FC_MINI_DB	FC triggered by Mini-Multicast Dbuffs
BCM_COSQ_FC_FULL_DB	FC triggered by Full-Multicast Dbuffs
BCM_COSQ_FC_OCB_DB	FC triggered by OCB Dbuffs
BCM_COSQ_FC_IS_OCB_ONLY	Specifies that FC triggered by OCB-only Resource
BCM_COSQ_FC_MUB	FC receive or generate via Multi-Use Bit
BCM_COSQ_FC_MASK_POOL_INDICATION	Ignore Pool related threshold

VSQ GPORT signatures change with additional parameter `_is_ocb_only`.

Table 23: Changed VSQ GPORT handle MACROs: used to handle the different VSQ types,

BCM_COSQ_GPORT_VSQ_CT_SET(_gport, _core_id, _is_ocb_only, _category)	Corresponding to gports created with BCM_COSQ_VSQ_CT flags
BCM_COSQ_GPORT_VSQ_CTTC_SET(_gport, _core_id, _is_ocb_only, _category, _traffic_class)	Corresponding to gports created with BCM_COSQ_VSQ_CTTC flags
BCM_COSQ_GPORT_VSQ_CTCC_SET(_gport, _core_id, _is_ocb_only, _category, _connection_class)	Corresponding to gports created with BCM_COSQ_VSQ_CTCC flags

New `bcm_cosq_voq_connector_gport_add`, `cosq_slow_profile_set/get` APIs with the required data types has been introduced in this release.

bcm_cosq_voq_connector_gport_add

Allocate voq connectors

Syntax

```
#include <bcm/cosq.h>
int
bcm_cosq_voq_connector_gport_add(int unit, bcm_cosq_voq_connector_gport_t
*config, bcm_gport_t *gport);
```

Parameters

unit	(IN) Unit number.
config	(IN) Config parameters for voq connector creation.
gport	(IN/OUT) GPORT ID for voq connector that is allocated.

Description

Refer to `bcm_cosq_gport_add` API in the 56XX-PG644-RDS.

Returns

bcm_cosq_slow_profile_set

Set the slow factor per profile.

Syntax

```
#include <bcm/cosq.h>
int bcm_cosq_slow_profile_set(
    int unit,
    bcm_cosq_slow_level_t *slow_level,
    bcm_cosq_slow_profile_attributes_t *attr);
```

Parameters

unit	(IN) BCM device number
slow_level	(IN) configuring the slow level
attr	(IN) (for "_set") the max rate to be configured for a specific slow level

Description

Set the the rate for a certain slow profilelevel.

Returns

bcm_cosq_slow_profile_get

Get the slow factor per profile.

Syntax

```
#include <bcm/cosq.h>
int bcm_cosq_slow_profile_get(
    int unit,
    bcm_cosq_slow_level_t *slow_level,
    bcm_cosq_slow_profile_attributes_t *attr);
```

Parameters

unit	(IN) BCM device number
slow_level	(IN) configuring the slow level
attr	(OUT) (for "_get") the max rate configured in a specific slow level

Description

Get the the rate for a certain slow profilelevel.

Returns

bcm_cosq_pfc_class_mapping_t_init

Initialize a PFC class mapping structure.

Syntax

```
#include <bcm/cosq.h>
void bcm_cosq_pfc_class_mapping_t_init(bcm_cosq_pfc_class_mapping_t *mapping);
```

Parameters

mapping	(INOUT) Pointer to PFC class mapping structure to be initialized
---------	--

Description

Initialize a PFC class mapping structure.

Returns

None.

bcm_cosq_pfc_class_mapping_set

Set mapping of PFC class to COSQs and / or scheduling nodes for a port.

Syntax

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

```
int bcm_cosq_pfc_class_mapping_set(
    int unit,
    bcm_gport_t port,
    int array_count,
    bcm_cosq_pfc_class_mapping_t *mapping_array);
```

Parameters

unit	(IN) BCM device number
port	(IN) Logical port number
array_count	(IN) (for "_set") Number of mappings in mapping_array
mapping_array	(IN) (for "_set") Array of PFC class mappings

Description

Set mapping between PFC class and COSQs and / or scheduling nodes for a port. Multiple mappings for a list of PFC class can be set at one time. Each PFC class can be mapped to multiple COSQs and / or scheduling nodes identified by gport id. Not all PFC classes are needed. User only set a group of PFC classes that system has to respond to. For `_getec`, the number of PFC class mappings associated with the port is returned by `array_countec`. PFC class and associated COSq priority or scheduling node's gport ID are returned respectively by `class` and `gport_list` in `mapping_arrayec`. `gport_list` is ended by `BCM_GPORT_INVALIDec`, when retrieved gports does not full fill the list. If `array_max<=0` and `mapping_array==NULLec`, only `array_count` is returned.

```
#define BCM_COSQ_PFC_GPORT_COUNT    16

/* PFC class to COSq / scheduling node mapping structure. */
typedef struct bcm_cosq_pfc_class_mapping_s {
    int class; /* PFC class. */
    bcm_gport_t gport_list[BCM_COSQ_PFC_GPORT_COUNT]; /* mapped COSqs, -1
for no mapping. */
} bcm_cosq_pfc_class_mapping_t;
```

Returns

bcm_cosq_pfc_class_mapping_get

Get mapping of PFC class to COSQs and / or scheduling nodes for a port.

Syntax

```
#include <bcm/cosq.h>
int bcm_cosq_pfc_class_mapping_get(
    int unit,
    bcm_gport_t port,
```



```
int array_max,
int *array_count,
bcm_cosq_pfc_class_mapping_t *mapping_array);
```

Parameters

unit	(IN) BCM device number
port	(IN) Logical port number
array_max	(IN) (for "_get") Number of mappings allocated in mapping_array
array_count	(OUT) (for "_get") Number of mappings returned
mapping_array	(OUT) (for "_get") Array of PFC class mappings

Description

Get mapping between PFC class and COSQs and / or scheduling nodes for a port. Multiple mappings for a list of PFC class can be set at one time. Each PFC class can be mapped to multiple COSQs and / or scheduling nodes identified by gport id. Not all PFC classes are needed. User only set a group of PFC classes that system has to respond to. For _getec, the number of PFC class mappings associated with the port is returned by array_countec. PFC class and associated COSq priority or scheduling node's gport ID are returned respectively by class and gport_list in mapping_arrayec. gport_list is ended by BCM_GPORT_INVALIDec, when retrieved gports does not full fill the list. If array_max<=0 and mapping_array==NULLec, only array_count is returned.

```
#define BCM_COSQ_PFC_GPORT_COUNT    16

/* PFC class to COSq / scheduling node mapping structure. */
typedef struct bcm_cosq_pfc_class_mapping_s {
    int class; /* PFC class. */
    bcm_gport_t gport_list[BCM_COSQ_PFC_GPORT_COUNT]; /* mapped COSqs, -1
for no mapping. */
} bcm_cosq_pfc_class_mapping_t;
```

Returns

PORT EXTENSION MANAGEMENT

A new port extender flag, used in bcm_extender_port_t port extension data type, is added.

Table 24: BCM_EXTENDER_PORT flags

Name	Purpose
BCM_EXTENDER_PORT_INGRESS_WIDE	Creates encap id wide mode



FABRIC

New fabric APIs are introduced in this release as following.

bcm_fabric_bandwidth_core_profile_set

Set fabric bandwidth profile per core

Syntax

```
#include <bcm/fabric.h>
int
bcm_fabric_bandwidth_core_profile_set(int unit, int core, int profile_count,
    bcm_fabric_bandwidth_profile_t *profile_array);
```

Parameters

unit	(IN) Unit number.
core	(IN) Core number.
profile_count	(IN) Number of profile entries to be configured / retrieved
profile_array	(IN) (for "_set") profile entry. see table BCM_FABRIC_BANDWIDTH_PROFILE_table

Description

Only meaningful for Jericho (NOT for Arad). Set bandwidth profile entry. Each link has entries corresponding to all the rci levels. Thus table size is the (number of links) * (rci levels). For Jericho, this utility relates to the 'core level' credit generator and is meaningful both in 'shared' mode and in non-shared mode (where each core operates independently of the other). See READ_SCH_DVS_CREDIT_SHARE_CONFIGURATIONr/WRITE_SCH_DVS_CREDIT_SHARE_CONFIGURATIONr

Returns

BCM_E_XXX

bcm_fabric_bandwidth_core_profile_get

Retrieve fabric bandwidth profile per core

Syntax

```
#include <bcm/fabric.h>
int
bcm_fabric_bandwidth_core_profile_get(int unit, int core, int profile_count,
    bcm_fabric_bandwidth_profile_t *profile_array);
```

Parameters

unit	(IN) Unit number.
core	(IN) Core number.
profile_count	(IN) Number of profile entries to be configured / retrieved
profile_array	(IN/OUT) (for "_get") profile entry. see table BCM_FABRIC_BANDWIDTH_PROFILE_table

Description

Only meaningful for Jericho (NOT for Arad). Retrieve bandwidth profile entry. Each link has entries corresponding to all the rci levels. Thus table size is the (number of links) * (rci levels). For Jericho, this utility relates to the 'core level' credit generator and is meaningful both in 'shared' mode and in non-shared mode (where each core operates independently of the other). See READ_SCH_DVS_CREDIT_SHARE_CONFIGURATIONr/WRITE_SCH_DVS_CREDIT_SHARE_CONFIGURATIONr

For "_get" the num_links and rci are inputs, while the max_kbps is the output.

Returns

BCM_E_xxx

FIELD PROCESSOR

New Field Processor qualifiers are added as following:

Table 25: New Field Qualifiers

bcmFieldQualifyInVPortWide	Wide attribute of Ingress VPort
bcmFieldQualifyStageClass	Pseudo Compression Class Stage
bcmFieldQualifyStageClassExactMatch	Pseudo Compression Exact Match Class Stage
bcmFieldQualifyFcoeOxID	FCoE Originator Exchange ID.
bcmFieldQualifyFcoeRxID	FCoE Responder Exchange ID.
bcmFieldQualifyIpProtocolClass	IP Protocol ID Class.
bcmFieldQualifyEtherTypeClass	EtherType Class.
bcmFieldQualifyL4SrcPortClass	L4 Source Port Class.
bcmFieldQualifyL4DstPortClass	L4 Destination Port Class.
bcmFieldQualifySrcIpClass	IPv4 SIP Class.
bcmFieldQualifySrcIpClassMsbNibble	IPv4 SIP Class 4-bits.
bcmFieldQualifySrcIpClassLower	IPv4 SIP Class Lower 16-bits.
bcmFieldQualifySrcIpClassUpper	IPv4 SIP Class Upper 16-bits.
bcmFieldQualifySrcIp6Class	IPv6 SIP Class.
bcmFieldQualifySrcIp6ClassMsbNibble	IPv6 SIP Class 4-bits.
bcmFieldQualifySrcIp6ClassLower	Ipv6 SIP Class Lower 16-bits.
bcmFieldQualifySrcIp6ClassUpper	IPv6 SIP Class Upper 16-bits.
bcmFieldQualifyFcoeOxIDClass	FCoE Originator Exchange ID Class.
bcmFieldQualifyDstIpClass	IPv4 DIP Class.
bcmFieldQualifyDstIpClassMsbNibble	IPv4 DIP Class 4-bits.
bcmFieldQualifyDstIpClassLower	IPv4 DIP Class Lower 16-bits.
bcmFieldQualifyDstIpClassUpper	IPv4 DIP Class Upper 16-bits.
bcmFieldQualifyDstIp6Class	IPv6 DIP Class.
bcmFieldQualifyDstIp6ClassMsbNibble	Ipv6 DIP Class 4-bits.

Table 25: New Field Qualifiers

bcmFieldQualifyInVPortWide	Wide attribute of Ingress VPort
bcmFieldQualifyDstIp6ClassLower	IPv6 DIP Class Lower 16-bits.
bcmFieldQualifyDstIp6ClassUpper	IPv6 DIP Class Upper 16-bits.
bcmFieldQualifyFcoeRxDClass	FCoE Responder Exchange ID Class.
bcmFieldQualifyFibreChanSrcIdClass	Fibre Channel Src Id Class.
bcmFieldQualifyFibreChanSrcIdClassMsbNibble	Fibre Channel Src Id Class 4-bits.
bcmFieldQualifyFibreChanSrcIdClassLower	Fibre Channel Src Id Class Lower 16-bits.
bcmFieldQualifyFibreChanSrcIdClassUpper	Fibre Channel Src Id Class Upper 16-bits.
bcmFieldQualifyFibreChanDstIdClass	Fibre Channel Dst Id Class.
bcmFieldQualifyFibreChanDstIdClassMsbNibble	Fibre Channel Dst Id Class 4-bits.
bcmFieldQualifyFibreChanDstIdClassLower	Fibre Channel Dst Id Class Lower 16-bits.
bcmFieldQualifyFibreChanDstIdClassUpper	Fibre Channel Dst Id Class Upper 16-bits.
bcmFieldQualifyTcpClassZero	TCP Class from TCP_FN0 .
bcmFieldQualifyTosClassZero	TOS Class from TOS_FN0 .
bcmFieldQualifyTtlClassZero	TTL Class from TTL_FN0 .
bcmFieldQualifyTcpClassOne	TCP Class from TCP_FN1 .
bcmFieldQualifyTosClassOne	TOS Class from TOS_FN1 .
bcmFieldQualifyTtlClassOne	TTL Class from TTL_FN1 .
bcmFieldQualifyOuterVlanPriCfi	Outer VLAN Priority + CFI
bcmFieldQualifyEthernetOamInterfaceClassNiv	Ethernet OAM Class Id assigned for packet based on Source VP table.
bcmFieldQualifyEthernetOamInterfaceClassMim	Ethernet OAM Class Id assigned for packet based on Source VP table.
bcmFieldQualifyEthernetOamInterfaceClassVlan	Ethernet OAM Class Id assigned for packet based on Source VP table.
bcmFieldQualifyEthernetOamInterfaceClassVxlan	Ethernet OAM Class Id assigned for packet based on Source VP table.
bcmFieldQualifyEthernetOamClassVlanTranslateKeyFirst	Ethernet OAM Class Id assigned for packet based on 1st Lookup in Vlan_Xlate table.
bcmFieldQualifyEthernetOamClassVlanTranslateKeySecond	Ethernet OAM Class Id assigned for packet based on 2nd Lookup in Vlan_Xlate table.
bcmFieldQualifyEthernetOamInterfaceClassMpls	Ethernet OAM Class Id assigned for packet based on Source VP table.
bcmFieldQualifyEthernetOamClassVpn	Ethernet OAM Class Id assigned for packet based on VFI table.
bcmFieldQualifyMplsOamUpperLabelDataDrop	MPLS OAM Labels above the forwarding label.
bcmFieldQualifyMplsOamClassMplsSwitchLabel	MPLS OAM Class Id assigned for packet based on MPLS table.

New types of OLP header format are added for `bcm_field_olp_header_type_t` elements to allow the user to specify the OLP header type to be used.

Table 26: Field Qualify OLP header type Format

Formats	Purpose
bcmFieldOlpHeaderTypeBfdOam,	BFD OAM packet.
bcmFieldOlpHeaderTypeEthOamLm,	Ethernet OAM LM packet.
bcmFieldOlpHeaderTypeEthOamDm,	Ethernet OAM DM packet.
bcmFieldOlpHeaderTypeBhhOamCcm,	BHH OAM CCM packet.
bcmFieldOlpHeaderTypeBhhOamLm,	BHH OAM LM packet.
bcmFieldOlpHeaderTypeBhhOamDm,	BHH OAM DM packet.
bcmFieldOlpHeaderTypeBhhOamOthers,	BHH OAM other opcode packet.

Table 26: Field Qualify OLP header type Format

Formats	Purpose
bcmFieldOlpHeaderTypeRfc6374Dlm,	MPLS LM/DM DLM packet.
bcmFieldOlpHeaderTypeRfc6374Dm,	MPLS LM/DM DM packet.
bcmFieldOlpHeaderTypeRfc6374DlmPlusDm,	MPLS LM/DM DLM + DM packet.
bcmFieldOlpHeaderTypeRfc6374Ilm,	MPLS LM/DM ILM packet.
bcmFieldOlpHeaderTypeRfc6374IlmPlusDm,	MPLS LM/DM ILM + DM packet.
bcmFieldOlpHeaderTypeSat,	Service Activation Test.
bcmFieldOlpHeaderTypeOtherAch,	OAM packets with other ACH types.
bcmFieldOlpHeaderTypeCount	Always last, not a usable value.

New enumerations for `bcm_field_oam_domain_t` elements are added to allow the user to specify more OAM domains on which the MEP lookups have to be done.

Table 27: Field Qualify OAM domain type Format

Formats	Purpose
bcmFieldOamDomainMplsLmDmSectionPort	RFC 6374 Section - Port based loss/delay measurement.
bcmFieldOamDomainMplsLmDmSectionInnerVlan	RFC 6374 Section - Port + Inner vlan based loss/delay measurement.
bcmFieldOamDomainMplsLmDmSectionOuterVlan	RFC 6374 Section - Port + Outer vlan based loss/delay measurement.
bcmFieldOamDomainMplsLmDmSectionOuterPlusInnerVlan	RFC 6374 Section - Port + Outer vlan + Inner Vlan based loss/delay measurement.
bcmFieldOamDomainMplsLmDmPw	RFC 6374 Psuedowire based loss/delay measurement.
bcmFieldOamDomainMplsLmDmLspLabel	RFC 6374 LSP based loss/delay measurement.

New actions of field processor are added in this release as following.

Table 28: Field Actions

Action	Description	param0	param1
bcmFieldActionVpnSet	Set a new VPN value	param0: VPN Id.	n/a
bcmFieldActionL3DestRoutePublicValueSet	Set L3 Destination public lookup result.	L3 Destination public lookup result	n/a
bcmFieldActionL3SrcRoutePublicValueSet	Set L3 Source public lookup result.	L3 Source public lookup result	n/a
bcmFieldActionClassZero	Set Field Action Class Zero for Stage Class. Must be set using api <code>bcm_field_action_class_add</code>	Class Zero Value.	n/a
bcmFieldActionClassOne	Set Field Action Class One for Stage Class. Must be set using api <code>bcm_field_action_class_add</code>	Class One Value.	n/a
bcmFieldActionTrapReduced	Treats traffic hitting the entry according to the provided trap code without setting the trap qualifier	TRAP GPORT	n/a

Table 28: Field Actions

Action	Description	param0	param1
bcmFieldActionGroupClass Set	Set Class ID value to a ingress field group. It is used as a part of lookup key.	Class ID	n/a

A new field group for field processor is added in this release. Default Action resolution ID for groups is defined BCM_FIELD_GROUP_ACTION_RES_ID_DEFAULT.

Table 29: New Group config flags

Group config flag	Purpose
BCM_FIELD_GROUP_CREATE_IS_CASCADED	If set, the cascaded action value of this Field group is set in the bcmFieldQualifyCascadedKeyValue qualifier of the Field group 'group_ref'.
BCM_FIELD_GROUP_CREATE_HANDLE_ENTRIES_BY_KEY	If set, entries in this group is handled by key and not by ID. Entries with identical key cannot be added. Getting and deleting entries should be according to the key

A new field action_res_id is added in bcm_field_group_config_t data structure.

```

/*
 * Group configuration structure. Used to create a field group with
 * specific attributes.
 */
typedef struct bcm_field_group_config_s {
    ...
    int action_res_id;           /* Action resolution id for Group. */
} bcm_field_group_config_t;

```

A new type for Fabric header used for bcm_field_fabric_header_t is added in this release.

Table 30: New Fabric Header

Name	Purpose
bcmFieldFabricHeaderEthernetLearn	regular Ethernet, stacking FTMH with DSP Extension and PPH fabric header

New field APIs are introduced in this release.

bcm_field_entry_stat_extended_attach

Attach statistics with stat action entity to Field Processor entry.

Syntax

```

#include <bcm/field.h>
int bcm_field_entry_stat_attach(int unit, bcm_field_entry_t entry, int stat_id,
bcm_field_stat_action_t stat_action);

```

Parameters

<code>unit</code>	(IN) BCM device number.
<code>entry</code>	(IN) Field entry ID.
<code>stat_id</code>	(IN) Statistics entity ID.
<code>stat_action</code>	(IN) Statistics Action.

Description

The purpose of this API is to attach statistics collection entity with stat action to a field entry.

Table 31: *bcm_field_stat_action_t*: Field Stat Action

Name	Purpose
<code>bcmFieldStatActionIncrement</code>	Increment the stat.
<code>bcmFieldStatActionIncrementCancel</code>	Do not increment the stat.
<code>bcmFieldStatActionSample</code>	Sample the stat value to the packet.
<code>bcmFieldStatActionCount</code>	Always Last. Not a usable value.

Returns

bcm_field_entry_stat_extended_get

Get statistics entity with stat action attached to Field Processor entry.

Syntax

```
#include <bcm/field.h>
int bcm_field_entry_stat_extended_get(int unit, bcm_field_entry_t entry, int
*stat_id,
    bcm_field_stat_action_t stat_action);
```

Parameters

<code>unit</code>	(IN) BCM device number.
<code>entry</code>	(IN) Field entry ID.
<code>stat_id</code>	(OUT) Statistics entity ID.
<code>stat_action</code>	(OUT) Stat Action.

Description

The purpose of this API is to check if any statistics with stat action collection entity is attached to field entry.

Returns

bcm_field_oam_stat_action_add

Add Oam Stat Action to the entry.

Syntax

```
#include <bcm/field.h>
int bcm_field_oam_stat_action_add (int unit, bcm_field_entry_t entry,
    bcm_field_oam_stat_action_t *oam_stat_action);
```

Parameters

unit	(IN) BCM device number.
entry	(IN) Field entry ID.
oam_stat_action	(IN) Oam Stat Action.

Description

Adds the stat action to the field entry for the given Flex stat Object and Mode.

bcm_field_oam_stat_action_delete

Delete Oam Stat Action from entry.

Syntax

```
#include <bcm/field.h>
int bcm_field_oam_stat_action_delete (int unit, bcm_field_entry_t entry,
    bcm_field_oam_stat_action_t *oam_stat_action);
```

Parameters

unit	(IN) BCM device number.
entry	(IN) Field entry ID.
oam_stat_action	(IN) Oam Stat Action.

Description

Delete Stat Action for a given Flex Stat Object and Mode from the entry.

bcm_field_oam_stat_action_delete_all

Delete all the Oam Stat Actions from entry.

Syntax

```
#include <bcm/field.h>
int bcm_field_oam_stat_action_delete_all (int unit, bcm_field_entry_t entry);
```

Parameters

unit	(IN) BCM device number.
entry	(IN) Field entry ID.
oam_stat_action	(IN) Oam Stat Action.

Description

Delete all the Oam Stat Actions from the entry.

bcm_field_oam_stat_action_get

Get the Oam Stat action for a given Stat Object installed in the entry.

Syntax

```
#include <bcm/field.h>
int bcm_field_oam_stat_action_get(int unit, bcm_field_entry_t entry,
    bcm_field_oam_stat_action_t *oam_stat_action);
```

Parameters

unit	(IN) BCM device number.
entry	(IN) Field entry ID.
oam_stat_action	(INOUT) Oam Stat Action.

Description

Returns the Stat Action for a given Flex Stat Object and Mode from the entry. Stat object should be filled in the `oam_stat_action` structure and API will fill all the elements in the structure and return.

bcm_field_oam_stat_action_get_all

Get all the Oam Stat actions added to the entry.

Syntax

```
#include <bcm/field.h>
int bcm_field_oam_stat_action_get_all(int unit, bcm_field_entry_t entry,
    int oam_stat_max,
    bcm_field_oam_stat_action_t *oam_stat_action,
    int *oam_stat_count);
```

Parameters

<code>unit</code>	(IN) BCM device number.
<code>entry</code>	(IN) Field entry ID.
<code>oam_stat_max</code>	(IN) Maximum number of oam stat actions.
<code>oam_stat_action</code>	(OUT) Oam Stat Action.
<code>oam_stat_count</code>	(INOUT) Number of Oam Stat Actions present in entry.

Description

Returns all the Oam Stat Actions present in the entry. When `oam_stat_max` is zero, api returns the number of oam stat actions added to the entry, in the `oam_stat_action` element. When `oam_stat_max` is non-zero, API assumes corresponding memory has been allocated in the `oam_stat_action` structure and fills the oam stat actions added to the fp entry and `oam_stat_count` will return number of oam stat actions present in the fp entry.

bcm_field_oam_stat_action_t_init

Initialize Field Oam Stat Action Structure.

Syntax

```
#include <bcm/field.h>
void bcm_field_oam_stat_action_t_init(
    bcm_field_oam_stat_action_t *oam_action);
```

Parameters

<code>oam_action</code>	Oam Stat Action.
-------------------------	------------------

Description

Because types can expand in the future, functions such as this are provided to set up default states for structures. This should be called before any `bcm_field_oam_stat_action_t` is filled in, to set default states for the fields within it, or when reusing a `bcm_field_oam_stat_action_t` for a new purpose or similar.

bcm_field_qualify_XXX

Add a qualification to a field entry

Syntax

```
#include <bcm/field.h>
int bcm_field_qualify_InVPortWide(
    int unit,
    bcm_field_entry_t entry,
    uint64 data,
    uint64 mask);
int bcm_field_qualify_FcoeOxID(
    int unit,
    bcm_field_entry_t entry,
    uint16 data,
    uint16 mask);
```

```
int bcm_field_qualify_FcoeRxID(
    int unit,
    bcm_field_entry_t entry,
    uint16 data,
    uint16 mask);
int bcm_field_qualify_IpProtocolClass(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_EtherTypeClass(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_L4SrcPortClass(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_L4DstPortClass(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIpClass(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIpClassMsbNibble(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIpClassLower(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIpClassUpper(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIp6Class(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIp6ClassMsbNibble(
    int unit,
    bcm_field_entry_t entry,
```

```
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIp6ClassLower(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_SrcIp6ClassUpper(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
    bcm_class_t *mask);
int bcm_field_qualify_FcoeOxidClass(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_DstIpClass(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_DstIpClassMsbNibble(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_DstIpClassLower(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_DstIpClassUpper(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_DstIp6Class(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_DstIp6ClassMsbNibble(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_DstIp6ClassLower(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
```

```
int bcm_field_qualify_DstIp6ClassUpper(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FcoeRxIDClass(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanSrcIdClass(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanSrcIdClassMsbNibble(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanSrcIdClassLower(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanSrcIdClassUpper(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanDstIdClass(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanDstIdClassMsbNibble(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanDstIdClassLower(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_FibreChanDstIdClassUpper(  
    int unit,  
    bcm_field_entry_t entry,  
    bcm_class_t data,  
    bcm_class_t mask);  
int bcm_field_qualify_TcpClassZero(  
    int unit,  
    bcm_field_entry_t entry,
```

```
bcm_class_t data,
bcm_class_t mask);
int bcm_field_qualify_TosClassZero(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_TtlClassZero(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_TcpClassOne(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_TosClassOne(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_TtlClassOne(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t data,
    bcm_class_t mask);
int bcm_field_qualify_OuterVlanPriCfi(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_EthernetOamInterfaceClassNiv(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_EthernetOamInterfaceClassMim(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_EthernetOamInterfaceClassVlan(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_EthernetOamInterfaceClassVxlan(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_EthernetOamClassVlanTranslateKeyFirst(
```

```
        int unit,
        bcm_field_entry_t entry,
        uint8 data,
        uint8 mask);
int bcm_field_qualify_EthernetOamClassVlanTranslateKeySecond(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_EthernetOamInterfaceClassMpls(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_EthernetOamClassVpn(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_MplsOamUpperLabelDataDrop(
    int unit,
    bcm_field_entry_t entry,
    uint8 data,
    uint8 mask);
int bcm_field_qualify_MplsOamClassMplsSwitchLabel(
    int unit,
    bcm_field_entry_t entry,
    uint16 data,
    uint16 mask);
```

Parameters

unit	BCM device number
entry	Field entry ID
data	Data to match against
mask	Mask to choose which bits of data to match against

Description

Adds a qualification to a filter entry. Each qualification added makes the entry more specific and match fewer possible packets.

Returns

bcm_field_qualify_XXX_get

Get a qualification match criteria from a field entry

Syntax

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

int bcm_field_qualify_InVPortWide_get(
    int unit,
    bcm_field_entry_t entry,
    uint64 *data,
    uint64 *mask);

int bcm_field_qualify_FcoeOxID_get(
    int unit,
    bcm_field_entry_t entry,
    uint16 *data,
    uint16 *mask);

int bcm_field_qualify_FcoeRxID_get(
    int unit,
    bcm_field_entry_t entry,
    uint16 *data,
    uint16 *mask);

int bcm_field_qualify_IpProtocolClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);

int bcm_field_qualify_EtherTypeClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);

int bcm_field_qualify_L4SrcPortClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);

int bcm_field_qualify_L4DstPortClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
```



```
int bcm_field_qualify_SrcIpClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_SrcIpClassMsbNibble_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_SrcIpClassLower_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_SrcIpClassUpper_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_SrcIp6Class_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_SrcIp6ClassMsbNibble_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_SrcIp6ClassLower_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_SrcIp6ClassUpper_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FcoeOxidClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_DstIpClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_DstIpClassMsbNibble_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
```

```
bcm_class_t *mask);
int bcm_field_qualify_DstIpClassLower_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_DstIpClassUpper_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_DstIp6Class_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_DstIp6ClassMsbNibble_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_DstIp6ClassLower_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_DstIp6ClassUpper_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FcoeRxIDClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FibreChanSrcIdClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FibreChanSrcIdClassMsbNibble_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FibreChanSrcIdClassLower_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FibreChanSrcIdClassUpper_get(
    int unit,
```

```
bcm_field_entry_t entry,
bcm_class_t *data,
bcm_class_t *mask);
int bcm_field_qualify_FibreChanDstIdClass_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FibreChanDstIdClassMsbNibble_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FibreChanDstIdClassLower_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_FibreChanDstIdClassUpper_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_TcpClassZero_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_TosClassZero_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_TtlClassZero_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_TcpClassOne_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_TosClassOne_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
int bcm_field_qualify_TtlClassOne_get(
    int unit,
    bcm_field_entry_t entry,
    bcm_class_t *data,
    bcm_class_t *mask);
```

```
int bcm_field_qualify_OuterVlanPriCfi_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamInterfaceClassNiv_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamInterfaceClassMim_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamInterfaceClassVlan_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamInterfaceClassVxlan_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamClassVlanTranslateKeyFirst_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamClassVlanTranslateKeySecond_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamInterfaceClassMpls_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_EthernetOamClassVpn_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_MplsOamUpperLabelDataDrop_get(
    int unit,
    bcm_field_entry_t entry,
    uint8 *data,
    uint8 *mask);
int bcm_field_qualify_MplsOamClassMplsSwitchLabel_get(
    int unit,
    bcm_field_entry_t entry,
```

```
uint16 *data,
uint16 *mask);
```

Parameters

unit	BCM device number
entry	Field entry ID
data	Data to match against
mask	Mask to choose which bits of data to match against

Description

Get a match criteria for a specific qualifier from a field entry.

Returns

FIELD COMPRESSION

A number of packet header fields are large in width but their number space is sparse. Tomahawk introduces an extended field compression capability to enable narrow rules. Compressed forms of the following fields can be generated in Tomahawk:

Table 32: Field Compression

Compression	Qualifier	Stage	Action Supported
EtherType	bcmFieldQualifyEtherType	bcmFieldQualifyStageClass ExactMatch	bcmFieldActionClassZero
TTL	bcmFieldQualifyTtl	bcmFieldQualifyStageClass ExactMatch	bcmFieldActionClassZero, bcmFieldActionClassOne
TOS	bcmFieldQualifyTos	bcmFieldQualifyStageClass ExactMatch	bcmFieldActionClassZero, bcmFieldActionClassOne
TCP	bcmFieldQualifyTcpControl	bcmFieldQualifyStageClass ExactMatch	bcmFieldActionClassZero, bcmFieldActionClassOne
IP Protocol	bcmFieldQualifyIpProtocol	bcmFieldQualifyStageClass ExactMatch	bcmFieldActionClassZero
L4 SrcPort	bcmFieldQualifyL4SrcPort, bcmFieldQualifyFcoeOxID	bcmFieldQualifyStageClass ExactMatch	bcmFieldActionClassZero
L4 DstPort	bcmFieldQualifyL4DstPort, bcmFieldQualifyFcoeRxID	bcmFieldQualifyStageClass ExactMatch	bcmFieldActionClassZero
Src Compression	bcmFieldQualifySrcIp, bcmFieldQualifySrcIp6, bcmFieldQualifyFibreChanS rcId	bcmFieldQualifyStageClass	bcmFieldActionClassZero
Dst Compression	bcmFieldQualifyDstIp, bcmFieldQualifyDstIp6, bcmFieldQualifyFibreChanD stId	bcmFieldQualifyStageClass	bcmFieldActionClassZero

To add entry in compression stage, first group is created using api `bcm_field_group_config_create` with corresponding compression qualifier, stage and action added as mentioned in [Field Compression](#) ([page 61](#)) Then entry is created in main IFP using `bcm_field_qualify_xxxClassyyy` api to qualify incoming packet on compressed values.

bcm_field_class_info_t_init

Initialize a `bcm_field_class_info_t`.

Syntax

```
#include <bcm/field.h>
int bcm_field_class_info_t_init(bcm_field_class_info_t *class_info);
```

Parameters

`class_info` Pointer to the `bcm_field_class_info_t` to initialize.

Description

Initializes the Field Class Info structure to default values. This function should be used to initialize any `bcm_field_class_info_t` structure prior to filling it out or passing it to an API function. This ensures that subsequent API releases may add new structure members to the `bcm_field_class_info_t` structure, and `bcm_field_class_info_t_init` will initialize the new members to correct default values.

Action for class entries(entry that is a member of a compression stage) is added using api `bcm_field_action_class_add` ([page 62](#)) . Action parameters are passed in form of structure `bcm_field_class_info_t`.

```
/*
 * Field Class Info structure. Used to attach action
 * to field class entries.
 */
typedef struct bcm_field_class_info_s {
    bcm_field_action_t action; /* Field Class action. */
    bcm_class_t class; /* Field Class value. */
} bcm_field_class_info_t;
```

bcm_field_action_class_add

Add an action to a field class entry (an entry that is a member of a compression stage).

Syntax

```
#include <bcm/field.h>
int bcm_field_action_class_add(int unit,
bcm_field_entry_t entry,
bcm_field_class_info_t *class_info);
```

Parameters

<code>unit</code>	BCM device number
<code>entry</code>	Field entry ID
<code>class_info</code>	Pointer to <code>bcm_field_class_info_t</code> structure

Description

Adds an action to be performed when a packet matches the entry. This action add is applicable for entries present in compression stage.

Only two actions `bcmFieldActionClassZero` and `bcmFieldActionClassOne` are valid for class entries, followed by a class value which is of type `bcm_class_t`.

User must specify class zero or one actions with class value to be applied as action for packets matching class entry.

Returns

`bcm_field_action_class_get`

Get an action from a field class entry (an entry that is a member of a direct extraction group).

Syntax

```
#include <bcm/field.h>
int bcm_field_action_class_get(int unit,
bcm_field_entry_t entry,
bcm_field_class_info_t *class_info);
```

Parameters

<code>unit</code>	BCM device number
<code>entry</code>	Field entry ID
<code>class_info</code>	Pointer to <code>bcm_field_class_info_t</code> structure

Description

Gets an action that would be performed when a packet matches the entry. See `bcm_field_action_class_add` ([page 62](#)) for more details.

User must pass if action type to retrieve is bcmFieldActionClassZero or bcmFieldActionClassOne as these are two applicable actions for class entries.

This api will return class value associated to provided entry for provided class action. Class Value returned is of type bcm_class_t.

Returns

bcm_field_class_size_get

Get Class size for provided qualifier.

Syntax

```
#include <bcm/field.h>
int bcm_field_class_size_get(int unit,
    bcm_field_qualify_t entry,
    uint16 *class_size);
```

Parameters

unit	BCM device number
qual	Field Qualifier
class_size	Pointer to class size

Description

Get Class Size for provided qualifier. Class Size returned is in bits size.

For all qualifiers for which compression action is supported, this api will return class size for provided qualifier. Class Size returned can vary in size based on qualifier provided as input.

Returns

IP MULTICAST

A new field of `bcm_ipmc_addr_t` has been added for IPv4 Source subnet mask. `/* IPMC address type. */ typedef struct bcm_ipmc_addr_s { ... bcm_ip_t s_ip_mask; /* IPv4 Source subnet mask. */ bcm_if_t ing_intf; /* L3 interface associated with this Entry */ } bcm_ipmc_addr_t;`

A new IPMC flag `BCM_IPMC_L2` has been added in this release.

Table 33: New IPMC Flag

Name	Purpose
<code>BCM_IPMC_L2</code>	Define IPMC L2 domain lookup (VLAN/VFI) in L3 IPMC table.

LAYER 2 ADDRESS MANAGEMENT

A new L2 learn distribution event used with `bcm_l2_addr_distribute_t` and the `bcm_l2_addr_distribute_t_init()` routine has been added.

Table 34: L2 Learning events flags

Name	Purpose
<code>BCM_L2_ADDR_DIST_SET_NO_DISTRIBUTER</code>	Do not send events to any distributor

A new field `age_state` has been added in `bcm_l2_addr_t` data structure in order to record the age state of the L2 entry.

```
typedef struct bcm_l2_addr_s {
    ...
    int age_state; /* Age state of the entry */
} bcm_l2_addr_t;
```

A new flag used with `bcm_l2_egress_t` and L2 egress API has been added, which is used to configure the packet editing details as the packet is egressing.

Table 35: New BCM L2 Egress Flag

Name	Purpose
<code>BCM_L2_EGRESS_WIDE</code>	Set wide entry in egress.

A new L2 station flag is added for data structure `bcm_l2_station_t` in this release as following.

Table 36: New BCM L2 Station Flag

Name	Purpose
BCM_L2_STATION_UNDERLAY	Entry for processing UNDERLAY tunnel headers when RIOT is enabled. When RIOT is enabled, this flag must be set for creating tunnel mac addresses to distinguish them from routing mac addresses.

The following new L2 API has been introduced in this release.

bcm_l2_station_traverse

Iterates over all entries in the L2 station table and executes user callback function for each entry.

Syntax

```
#include <bcm/l2.h>
```

```
int bcm_l2_station_traverse(int unit, bcm_l2_station_traverse_cb trav_fn, void *user_data);
```

Parameters

<code>unit</code>	BCM device number
<code>trav_fn</code>	Call back function provided by API caller
<code>user_data</code>	Pointer to any data provided by API caller

Description

Iterates over all valid entries in the L2 station table and executes user provided callback function that defined as following:

```
typedef int (*bcm_l2_station_traverse_cb)(int unit, bcm_l2_station_t *info, void *user_data);
```

Returns

L2GRE MANAGEMENT

A new field `match_port_class` is added in `bcm_l2gre_vpn_config_t` structure for the local port vlan domain when configuring L2GRE vpn.

```
typedef struct bcm_l2gre_vpn_config_s {  
    ...  
    bcm_gport_t match_port_class;          /* local port vlan domain */  
}
```

```
} bcm_l2gre_vpn_config_t;
```

LAYER 3 MANAGEMENT

A new field `l3a_mtu_forwarding` is added in the L3 interface structure `bcm_l3_intf_t` for the configuration of Forwarding Layer MTU.

```
typedef struct bcm_l3_intf_s {
    ...
    int          l3a_mtu_forwarding; /* Forwarding Layer MTU */
    ...
    bcm_gport_t l3a_source_vp;      /* Source virtual port in overlay
                                     domain. SVP is used for deriving port
                                     properties in the egress device when
                                     packets are sent over HG flow. */
} bcm_l3_intf_t;
```

New fields `mpls_action` and `oam_global_context_id` are added in the L3 egress structure `bcm_l3_egress_t` for the passing of OAM global context id from ingress to egress XGS chip.

```
/*
 * L3 Egress Structure
 * Description of an L3 forwarding destination.
 */
typedef struct bcm_l3_egress_s {
    ...
    bcm_mpls_egress_action_t mpls_action; /* MPLS action. */
    ...
    uint16 oam_global_context_id; /* OAM global context id passed from
                                   ingress to egress XGS chip. */
} bcm_l3_egress_t;
```

Two new L3 ECMP flags have been added for `ecmp_group_flags` of `bcm_l3_egress_ecmp_t` data structure.

Table 37: New BCM L3 ECMP Flags

Name	Purpose
BCM_L3_ECMP_OVERLAY	If set, the ECMP group will be programmed at ECMP Level 1 in devices supporting Hierarchical ECMP feature.
BCM_L3_ECMP_UNDERLAY	If set, the ECMP group will be programmed at ECMP Level 2 in devices supporting Hierarchical ECMP feature.

A new L3 tunneling type is added in the enumerated `bcm_tunnel_type_t` type.

Table 38: New L3 Tunneling Type

Name	Purpose
bcmTunnelTypeL2EncapExternalCpu	External CPU L2 tunnel

MIRRORING

New flags for `bcm_mirror_pkt_header_updates_t` have been added in this release.

Table 39: BCM Mirror Packet Header Update Flags

Name	Description
<code>BCM_MIRROR_PKT_HEADER_UPDATE_COLOR</code>	Drop precedence.
<code>BCM_MIRROR_PKT_HEADER_UPDATE_PRIO</code>	Internal packet priority.
<code>BCM_MIRROR_PKT_HEADER_UPDATE_ECN_VALUE</code>	ECN capable and congestion encoding.
<code>BCM_MIRROR_PKT_HEADER_UPDATE_ECN_CNMCANCEL</code>	Ignore Congestion Point.
<code>BCM_MIRROR_PKT_HEADER_UPDATE_TRUNK_HAS_H_RESULT</code>	LAG load balancing key.
<code>BCM_MIRROR_PKT_HEADER_UPDATE_IN_PORT</code>	8b should be exposed?
<code>BCM_MIRROR_PKT_HEADER_UPDATE_VSQ</code>	selects STF (statistics flow) VSQ.
<code>BCM_MIRROR_PKT_HEADER_UPDATE_FABRIC_HEADER_EDITING</code>	changes to the fabric headers

MAC-IN-MAC MANAGEMENT

The new flags for MiM VPN config flag and MiM Port flag are added in this release respectively.

Table 40: New MAC-in-MAC VPN Config Flag

Name	Purpose
<code>BCM_MIM_VPN_SERVICE_ENCAP_INGRESS_WIDE</code>	create <code>service_encap_id</code> wide mode

Table 41: MAC-in-MAC Port Flags

Name	Purpose
<code>BCM_MIM_PORT_SERVICE_ENCAP_INGRESS_WIDE</code>	Used when the application creates <code>service_encap_id</code> wide mode missed packets.

MPLS MANAGEMENT

A new flag `BCM_MPLS_PORT_INGRESS_WIDE` is added as one of the MPLS port flags.

Table 42: New MPLS Port Flag

Name	Purpose
<code>BCM_MPLS_PORT_INGRESS_WIDE</code>	use Wide mode (additional data)

A new flag `BCM_MPLS_SWITCH_WIDE` is added as one of the MPLS Tunnel Switch flags.

Table 43: MPLS Tunnel Switch Flags

Name	Purpose
BCM_MPLS_SWITCH_WIDE	Create with wide mode
BCM_MPLS_SWITCH_LOOKUP_NONE	No source match done.
BCM_MPLS_SWITCH_EVPN_IML	Differentiate IML labels from ordinary labels.

Two fields are added in the `bcm_mpls_tunnel_switch_t` structure that is used to specify an entry in the MPLS label table.

```
typedef struct bcm_mpls_tunnel_switch_s {
    ...
    uint16 oam_global_context_id;    /* OAM global context id passed from
                                     ingress to egress XGS chip. */
    uint32 class_id;                /* Class ID */
} bcm_mpls_tunnel_switch_t;
```

New MPLS egress label flags are added with the `bcm_mpls_egress_label_t` structure that describes the MPLS label header values (label, EXP, TTL) and QOS options.

Table 44: MPLS Egress Label Flags

Name	Purpose
BCM_MPLS_EGRESS_LABEL_WIDE	Set wide entry in egress.
BCM_MPLS_EGRESS_LABEL_IML	Label is EVPN Implicit Multicast.

A new MPLS range action flag is added that is used in `bcm_mpls_range_action_t` data structure with `bcm_mpls_range_action_add` API.

Table 45: MPLS Range Actions

Name	Purpose
BCM_MPLS_RANGE_ACTION_EVPN_IML	Specify a range for EVPN IML labels.

New MPLS APIs are introduced in this release as following.

bcm_mpls_entropy_identifier_t_init

Initialize MPLS Entropy label identifier structure.

Syntax

```
#include <bcm/mpls.h>
void bcm_mpls_entropy_identifier_t_init(bcm_mpls_entropy_identifier_t *info);
```

Parameters

info Pointer to the struct to be initialized

Description

Initialize MPLS Entropy label identifier structure.

Returns

None.

bcm_mpls_entropy_identifier_add

Add an MPLS Entropy Label Identifier.

Syntax

```
#include <bcm/mpls.h>
int bcm_mpls_entropy_identifier_add(int unit, uint32 options,
                                   bcm_mpls_entropy_identifier_t *entropy_info);
```

Parameters

unit (IN) BCM device number
options (IN) MPLS_ENTROPY_LABEL_*
entropy_info (IN) Entropy label identifier information

Description

The bcm_mpls_entropy_identifier_t structure is used to specify an MPLS entropy label identifier.

Using this API, user can configure an entropy label identifier which will be used to detect the presence of an entropy label in the label stack. Currently, the only action supported is POP. User would have to ensure not to use any of the forwarding labels as entropy label because then it would qualify as MPLS forwarding label and not entropy label. Returns BCM_E_PARAM if the label or mask passed by user is invalid.

```
typedef struct bcm_mpls_entropy_identifier_s {
    bcm_mpls_label_t    label;      /* Incoming entropy label identifier */
    bcm_mpls_label_t    mask;      /* Entropy label mask */
    int                 pri;        /* Currently unused */
    uint32              flags;      /* Currently unused */
} bcm_mpls_entropy_identifier_t;
```

Table 46: MPLS Entropy Label Identifier Flags

Name	Purpose
BCM_MPLS_ENTROPY_LABEL_IDENTIFIER_TOS	Identifies entropy label, if present at the top of the label stack, based on a match of the upper label bits



Returns

bcm_mpls_entropy_identifier_get

Get information about an MPLS Entropy Label Identifier.

Syntax

```
#include <bcm/mpls.h>
int bcm_mpls_entropy_identifier_get(int unit, bcm_mpls_entropy_identifier_t
*entropy_info);
```

Parameters

unit	(IN) BCM device number
entropy_info	(IN/OUT) Entropy label identifier information

Description

On passing entropy label+mask+pri to this API, returns information of the other fields. Returns BCM_E_EMPTY if no entropy label is configured. If no valid label+mask is passed by the user, information about the only configured label is returned.

Returns

bcm_mpls_entropy_identifier_delete

Delete an MPLS Entropy Label Identifier.

Syntax

```
#include <bcm/mpls.h>
int bcm_mpls_entropy_identifier_delete(int unit, bcm_mpls_entropy_identifier_t
*entropy_info);
```

Parameters

<code>unit</code>	(IN) BCM device number
<code>entropy_info</code>	(IN) Entropy label identifier information

Description

Deletes MLPS entropy configuration corresponding to the passed label+mask+pri key. Returns `BCM_E_EMPTY` if no entropy label is configured. If label+mask passed by user is invalid, returns `BCM_E_PARAM`

Returns

bcm_mpls_entropy_identifier_delete_all

Delete all MPLS Entropy Label Identifiers.

Syntax

```
#include <bcm/mpls.h>
int bcm_mpls_entropy_identifier_delete_all(int unit);
```

Parameters

<code>unit</code>	(IN) BCM device number
-------------------	------------------------

Description

Deletes all MPLS Entropy label identifier entries.

Returns

bcm_mpls_entropy_identifier_traverse

Traverse all valid MPLS entropy label identifier entries and call the supplied callback routine.

Syntax

```
#include <bcm/mpls.h>
int bcm_mpls_entropy_identifier_traverse(int unit,
    bcm_mpls_entropy_identifier_traverse_cb cb,
    void *user_data);
```


Parameters

<code>unit</code>	(IN) BCM device number
<code>cb</code>	(IN) User callback function, called once per MPLS entry
<code>user_data</code>	(IN) Cookie

Description

Returns `BCM_E_EMPTY` if no entropy label is configured. Traverse all valid MPLS entropy label identifier entries and call the supplied callback routine. The callback function is defined as following:

```
typedef int (*bcm_mpls_entropy_identifier_traverse_cb)
(int unit, bcm_mpls_entropy_identifier_t *info, void *user_data);
```

Returns

bcm_mpls_range_action_t_init

Initialize the MPLS range action structure.

Syntax

```
#include <bcm/mps.h>
void bcm_mpls_range_action_t_init(bcm_mpls_range_action_t *info);
```

Parameters

<code>info</code>	Pointer to the struct to be initialized
-------------------	---

Description

Initialize the MPLS range action structure.

Returns

None.

OUT-OF-BAND HIGIG CHANNELIZED FLOW CONTROL (OOB HCFC) AND OUT-OF-BAND STATS (OOB STATS)

New OOB API module is new in this release for HiGig Channelized Flow Control and Stats.

Out-Of-Band (OOB) Flow Control messages are transmitted continuously via a dedicated OOB Interface. In Fabric Interface Chips (FIC) application, the TX interface provides lossless operation on the egress path while the RX interfaces supports PFC generation on the ingress path that protects both XGS device and the attached FIC from frame loss in this path. All OOB Flow Control API starts with `bcm_oob_fc_` prefix.

For configuration of OOB Flow Control API should be called in following order:-

OOB Flow Control Rx

1. Initialize the OOB FC Rx side configuration structure using `bcm_oob_fc_rx_config_t_init`
2. Get the existing configuration of given interface using `bcm_oob_fc_rx_config_get`
3. Do the required changes in the configuration fetched and then set the configuration using `bcm_oob_fc_rx_config_set`
4. Get the existing traffic class to priority mapping using `bcm_oob_fc_rx_port_tc_mapping_multi_get/bcm_oob_fc_rx_port_tc_mapping_get`
5. Do the required changes in the fetched traffic class to priority class mapping and set the new traffic class to priority mapping using `bcm_oob_fc_rx_port_tc_mapping_multi_set/bcm_oob_fc_rx_port_tc_mapping_set`

OOB Flow Control Tx

1. Initialize the OOB FC Tx side configuration structure using `bcm_oob_fc_tx_config_t_init`
2. Get the status of port for reporting congestion status using `bcm_port_control_get`
3. Enable/Disable the congestion state reporting using `bcm_port_control_set`
4. Get the existing OOB FC Rx configuration using `bcm_oob_fc_tx_config_get`
5. Do the required changes in the configuration fetched and then set the configuration using `bcm_oob_fc_tx_config_set`

Out-Of-Band Stats feature communicates internal device metrics to an external device at a higher rate. This statistics is communicated using the existing HCFC message format. All OOB Stats API starts with `bcm_oob_stats_` prefix.

For configuration of OOB Stats API should be called in following order:-

OOB Stats

1. Initialize the OOB Stats configuration structure using `bcm_oob_stats_config_t_init`
2. Get the existing configuration using `bcm_oob_stats_config_get`
3. Get the existing service pool list for those OOB Stats will be reported using `bcm_oob_stats_pool_mapping_get/bcm_oob_stats_pool_mapping_multi_get`
4. Do the necessary changes to the service pool list and program the new service pool list using `bcm_oob_stats_pool_mapping_set/bcm_oob_stats_pool_mapping_multi_set`
5. Get the existing Unicast queue list for those OOB Stats will be reported using `bcm_oob_stats_queue_mapping_get/bcm_oob_stats_queue_mapping_multi_get`
6. Do the necessary changes to the Unicast queue list and program the new Unicast queue list using `bcm_oob_stats_queue_mapping_set/bcm_oob_stats_queue_mapping_multi_set`

7. Do the required changes in the fetched OOB Stats configuration and the set the new configuration using `bcm_oob_stats_config_set`

Table 47: Configuration parameters for OOB Flow Control Tx

Parameter	Description
flags	Flags for enabling OOB Flow Control for different resources. BCM OOB Flow Control Tx Flags (page 75)
inter_pkt_gap	Inter Packet Gap in clock cycles.
gcs_id	Service Pool ID for Global congestion reporting
egr_mode	Egress congestion reporting mode (Note: Operate in only one mode) BCM OOB Flow Control Tx Egress Mode (page 75)

Table 48: BCM OOB Flow Control Tx Flags

Flags	Description
BCM_OOB_FC_TX_FCN_EN	Fast Congestion Notification enable flag
BCM_OOB_FC_TX_ING_EN	OOB Flow Control Ingress resource congestion notification enable flag
BCM_OOB_FC_TX_EGR_EN	OOB Flow Control Egress resource congestion notification enable flag
BCM_OOB_FC_TX_POOL_EN	OOB Flow Control service pool (ingress + egress) resource congestion notification enable flag

Table 49: BCM OOB Flow Control Tx Egress Mode

Flags	Description
bcmOOBFCTxReportUCOnly	OOB FC Tx only reports Unicast queues congestion state only
bcmOOBFCTxReportMCOOnly	OOB FC Tx only reports Multicast queues congestion state only
bcmOOBFCTxReportCos	OOB FC Tx reports COS queue group congestion states
bcmOOBFCTxReportUCAndMC	OOB FC Tx reports both Unicast and Multicast queues congestion states

Table 50: Information parameters for OOB Flow Control Tx

Parameter	Description
ing_base	Channel Base of the HCFC message that reports the Ingress (port, PG) Congestion States.
ucast_base	Channel Base of the HCFC message that reports the Egress Unicast Queue Congestion States.
mcast_base	Channel Base of the HCFC message that reports the Egress Multicast Queue Congestion States.
cos_base	Channel Base of the HCFC message that reports the Egress COS Queue Group Congestion States.
pool_base	Channel Base number to use when sending all service pool (ingress + egress) states in a single message.

Table 51: Configuration parameters for OOB Flow Control Rx

Parameter	Description
channel_base	Channel Base of the first HCFC message that is received on a given OOB FC Rx interface containing flow control information.
enable	Global control for enabling and disabling OOB FC Rx interface enable=1 (Enable) and enable=0 (Disable).



Table 52: BCM OOB Stats Configuration Structure

Parameter	Description
flags	Flags for OOB Stats Global Configuration BCM OOB Stats Flags (page 76)
inter_pkt_gap	Inter packet gap between OOB Stats message packets in clock cycles
config_id	Global configuration id.

Table 53: BCM OOB Stats Flags

Flags	Description
BCM_OOB_STATS_TIMESTAMP_EN	Enable flag for reporting Timestamp
BCM_OOB_STATS_POOL_EN	Enable flag for reporting instantaneous service pool stats
BCM_OOB_STATS_QUEUE_EN	Enable flag for reporting instantaneous Unicast queue stats

Table 54: BCM OOB Stats Resource Direction

Flags	Description
BCM_OOB_STATS_INGRESS	Ingress side resource
BCM_OOB_STATS_EGRESS	Egress side resource

bcm_oob_fc_tx_config_t_init

Initializing OOB Flow Control Tx global configuration structure

Syntax

```
#include <bcm/oob.h>
void bcm_oob_fc_tx_config_t_init(bcm_oob_fc_tx_config_t *config);
```

Parameters

config Pointer to OOB Flow Control Tx global configuration structure Configuration parameters for OOB Flow Control Tx ([page 75](#))

Description

Initialize the global configuration parameters for OOB Flow Control Tx

Returns

None.

bcm_oob_fc_tx_config_set

Set the OOB Flow Control Tx global configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_tx_config_set(int unit, bcm_oob_fc_tx_config_t *config);
```

Parameters

unit	(IN) BCM Device Number
config	(IN(set)) Pointer to OOB Flow Control Tx global configuration structure Configuration parameters for OOB Flow Control Tx (page 75)

Description

Configure the global configuration parameters for OOB Flow Control Tx

Returns

bcm_oob_fc_tx_config_get

Get the OOB Flow Control Tx global configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_tx_config_get(int unit, bcm_oob_fc_tx_config_t *config);
```

Parameters

unit	(IN) BCM Device Number
config	(OUT(get)) Pointer to OOB Flow Control Tx global configuration structure Configuration parameters for OOB Flow Control Tx (page 75)

Description

Retrieve the global configuration parameters for OOB Flow Control Tx

Returns

bcm_oob_fc_tx_info_t_init

Initializing OOB Flow Control Tx global information

Syntax

```
#include <bcm/oob.h>
void bcm_oob_fc_tx_info_t_init(bcm_oob_fc_tx_info_t *info);
```

Parameters

info (OUT) Pointer to OOB Flow Control Tx global information structure Information parameters for OOB Flow Control Tx ([page 75](#))

Description

Initialize the global information parameters for OOB Flow Control Tx

Returns

None.

bcm_oob_fc_tx_info_get

Retrieve the OOB Flow Control Tx global information

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_tx_info_get(int unit, bcm_oob_fc_tx_info_t *info);
```

Parameters

unit (IN) BCM Device Number
info (OUT) Pointer to OOB Flow Control Tx global information structure Information parameters for OOB Flow Control Tx ([page 75](#))

Description

Retrieve the global information parameters for OOB Flow Control Tx

Returns

bcm_oob_fc_rx_port_tc_mapping_multi_set

Multiple set of OOB Flow Control Rx interface side traffic class to priority mapping

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_rx_port_tc_mapping_multi_set(int unit, bcm_oob_fc_rx_intf_id_t
intf_id, bcm_gport_t gport, int array_count, uint32 *tc, uint32 *pri_bmp);
```

Parameters

<code>unit</code>	(IN) BCM Device Number
<code>intf_id</code>	(IN) OOB FC Rx Interface Number
<code>gport</code>	(IN) Modport type gport
<code>array_count</code>	(IN) Number of elements in Traffic Class array
<code>tc</code>	(IN) Pointer to Traffic Class array[0-7]
<code>pri_bmp</code>	(IN) Pointer to Priority Bitmap Array[0-7]

Description

Multiple set of traffic class to priority mapping for a given OOB FC Rx interface and modport gport.

Returns

bcm_oob_fc_rx_port_tc_mapping_set

Set OOB Flow Control Rx interface side traffic class to priority mapping

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_rx_port_tc_mapping_set(int unit, bcm_oob_fc_rx_intf_id_t
intf_id, bcm_gport_t gport, uint32 tc, uint32 pri_bmp);
```

Parameters

<code>unit</code>	(IN) BCM Device Number
<code>intf_id</code>	(IN) OOB FC Rx Interface Number
<code>gport</code>	(IN) Modport type gport
<code>tc</code>	(IN) Traffic Class array[0-7]
<code>pri_bmp</code>	(IN) Priority Bitmap Array[0-7]

Description

Set of traffic class to priority mapping for a given OOB FC Rx Interface and modport gport.

Returns

bcm_oob_fc_rx_port_tc_mapping_multi_get

Multiple get OOB Flow Control Rx interface side traffic class to priority mapping

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_rx_port_tc_mapping_multi_get(int unit, bcm_oob_fc_rx_intf_id_t
intf_id, bcm_gport_t gport, int array_max, uint32 *tc, uint32 *pri_bmp, int
*array_count);
```

Parameters

unit	(IN) BCM Device Number
intf_id	(IN) OOB FC Rx Interface Number
gport	(IN) Modport type gport
array_max	(IN) Maximum Length tc and pri_bmp array
tc	(IN) Pointer to Traffic Class array[0-7]
pri_bmp	(OUT) Priority Bitmap Array[0-7]
array_count	(OUT) Number of elements in Traffic Class array

Description

Multiple get traffic class to priority mapping for a given OOB FC Rx Interface and modport gport.

Returns

bcm_oob_fc_rx_port_tc_mapping_get

Get OOB Flow Control Rx interface side traffic class to priority mapping

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_rx_port_tc_mapping_get(int unit, bcm_oob_fc_rx_intf_id_t
intf_id, bcm_gport_t gport, uint32 tc, uint32 *pri_bmp);
```

Parameters

unit	BCM Device Number
intf_id	OOB FC Rx Interface Number
gport	Modport type gport
tc	Traffic Class array[0-7]
pri_bmp	Priority Bitmap Array[0-7]

Description

Get traffic class to priority mapping for a given OOB FC Rx Interface and modport port.



Returns

bcm_oob_fc_rx_config_t_init

Initializes the configuration for given OOB Flow Control Rx Interface.

Syntax

```
#include <bcm/oob.h>
void bcm_oob_fc_rx_config_t_init(bcm_oob_fc_rx_config_t *config);
```

Parameters

config	(OUT) Pointer to OOB FC Rx Interface configuration structure Configuration parameters for OOB Flow Control Rx (page 75)
--------	---

Description

Initializes the OOB Flow Control Rx configuration structure.

Returns

None.

bcm_oob_fc_rx_config_set

Set the configuration for given OOB Flow Control Rx Interface.

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_rx_config_set(int unit, bcm_oob_fc_rx_intf_id_t intf_id,
bcm_oob_fc_rx_config_t *config, int array_count, bcm_gport_t *gport_array);
```

Parameters

unit	(IN) BCM Device Number.
intf_id	(IN) OOB FC Rx Interface Number.
config	(IN) Pointer to OOB FC Rx Interface configuration structure Configuration parameters for OOB Flow Control Rx (page 75)
array_count	(IN) Number of gports in the gport array
gport_array	(IN) Pointer to array of modport type gport

Description

Set the configuration parameters and also control the OOB FC Rx interface for a given interface id. Also program the mapping of flow control byte number to system port number.

Returns

bcm_oob_fc_rx_config_get

Get the configuration for given OOB Flow Control Rx Interface.

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_rx_config_get(int unit, bcm_oob_fc_rx_intf_id_t intf_id,
bcm_oob_fc_rx_config_t *config, int array_max, bcm_gport_t *gport_array, int
*array_count);
```

Parameters

unit	(IN) BCM Device Number.
intf_id	(IN) OOB FC Rx Interface Number.
config	(OUT) Pointer to OOB FC Rx Interface configuration structure Configuration parameters for OOB Flow Control Rx (page 75)
array_max	(IN) Maximum length of gport array passed as an argument to the function.
gport_array	(OUT) Pointer to array of modport type gport
array_count	(OUT) Number of gports returned in the gport array

Description

Get the configuration parameters and also control the OOB FC Rx interface for a given interface id. Also program the mapping of flow control byte number to system port number.

Returns

bcm_oob_fc_rx_port_offset_get

Get the flow control byte number of a port on a given OOB Flow Control Rx interface.

Syntax

```
#include <bcm/oob.h>
int bcm_oob_fc_rx_port_offset_get(int unit, bcm_oob_fc_rx_intf_id_t intf_id,
bcm_gport_t gport, uint32 *offset);
```

Parameters

unit	(IN) BCM Device Number.
intf_id	(IN) OOB FC Rx Interface Number.
gport	(IN) Modport gport
offset	(OUT) Flow control byte number

Description

Get the flow control byte number of a port in HCFC message received on a given OOB Flow Control Rx interface.

Returns

bcm_oob_stats_config_t_init

Initialize the OOB Stats global configuration parameters.

Syntax

```
#include <bcm/oob.h>
void bcm_oob_stats_config_t_init(bcm_oob_stats_config_t *config);
```

Parameters

config	(OUT) Pointer to OOB Stats configuration structure. BCM OOB Stats Configuration Structure (page 76)
--------	---

Description

Initialize the OOB Stats global configuration structure parameters.

Returns

None.

bcm_oob_stats_config_set

Set the OOB Stats global configuration parameters

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_config_set(int unit, bcm_oob_stats_config_t *config);
```

Parameters

unit	(IN) BCM Device Number
config	(IN) Pointer to OOB Stats configuration structure. BCM OOB Stats Configuration Structure (page 76)

Description

Set the OOB Stats global configuration structure parameters.

Returns

bcm_oob_stats_config_get

Get the OOB Stats global configuration parameters

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_config_get(int unit, bcm_oob_stats_config_t *config);
```

Parameters

unit	(IN) BCM Device Number
config	(OUT) Pointer to OOB Stats configuration structure. BCM OOB Stats Configuration Structure (page 76)

Description

Get the OOB Stats global configuration structure parameters.

Returns

bcm_oob_stats_pool_mapping_multi_set

Multiple set of OOB Stats service pool list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_pool_mapping_multi_set(int unit, int array_count, int
*offset_array, uint8 *dir_array, bcm_service_pool_id_t *pool_array);
```

Parameters

<code>unit</code>	(IN) BCM Device Number
<code>array_count</code>	(IN) Number of elements in array
<code>offset_array</code>	(IN) Pointer to array containing the offset at which service pool need to be added in the service pool list
<code>dir_array</code>	(IN) Pointer to array containing the direction of service pool at the same index in the <code>pool_array</code> BCM OOB Stats Resource Direction (page 76)
<code>pool_array</code>	(IN) Pointer to array containing the service pool

Description

Multiple set of service pool id at a given index in the memory so that OOB Stats will be reported in the same order as the service pool ids are programmed in the memory. In BCM56960, Users can again elect to report the instantaneous size of a subset of service pool by writing the value 0xF into an entry in the `pool_array`.

Returns

`bcm_oob_stats_pool_mapping_multi_get`

Multiple get of OOB Stats service pool list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_pool_mapping_multi_get(int unit, int array_max, int
*offset_array, uint8 *dir_array, bcm_service_pool_id_t *pool_array, int
*array_count);
```

Parameters

<code>unit</code>	(IN) BCM Device Number
<code>array_max</code>	(IN) Maximum length of given array
<code>offset_array</code>	(IN) Pointer to array containing the offset at which service pool need to be added in the service pool list
<code>dir_array</code>	(OUT) Pointer to array containing the direction of service pool at the same index in the <code>pool_array</code> BCM OOB Stats Resource Direction (page 76)
<code>pool_array</code>	(OUT) Pointer to array containing the service pool
<code>array_count</code>	(OUT) Number of elements in array returned

Description

Multiple get of service pool id and their corresponding direction at a given index in the memory. OOB Stats will be reported in the same order as the service pool ids are programmed in the memory. In BCM56960, Value 0xF into an entry in the `pool_array` indicates the end of reporting round of instantaneous service pool stats.

Returns

bcm_oob_stats_pool_mapping_set

Set of OOB Stats service pool list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_pool_mapping_set(int unit, int offset, uint8 dir,
bcm_service_pool_id_t pool);
```

Parameters

unit	(IN) BCM Device Number
offset	(IN) Offset at which service pool need to be added in the service pool list
dir	(IN) Direction of service pool BCM OOB Stats Resource Direction (page 76)
pool	(IN) Service pool ID

Description

Set of service pool id at a given offset(index) in the memory so that OOB Stats will be reported in the same order as the service pool ids are programmed in the memory. In BCM56960, Users can again elect to report the instantaneous size of a subset of service pool by writing the value 0xF into an entry in the pool.

Returns

bcm_oob_stats_pool_mapping_get

Get of OOB Stats service pool list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_pool_mapping_get(int unit, int offset, uint8 *dir,
bcm_service_pool_id_t *pool);
```

Parameters

<code>unit</code>	(IN) BCM Device Number
<code>offset</code>	(IN) Offset at which service pool need to be read from the service pool list
<code>dir</code>	(OUT) Pointer to the direction of service pool at the same index in the pool BCM OOB Stats Resource Direction (page 76)
<code>pool</code>	(OUT) Pointer to the service pool

Description

Get of service pool id and their corresponding direction at a given offset(index) in the memory. OOB Stats will be reported in the same order as the service pool ids are programmed in the memory. In BCM56960, Value 0xF into an entry in the `pool_array` indicates the end of reporting round of instantaneous service pool stats.

Returns

bcm_oob_stats_queue_mapping_multi_set

Multiple set of OOB Stats Unicast queue list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_queue_mapping_multi_set(int unit, int array_count, int
*offset_array, bcm_gport_t *gport_array);
```

Parameters

<code>unit</code>	(IN) BCM Device Number
<code>array_count</code>	(IN) Number of elements in array
<code>offset_array</code>	(IN) Pointer to array containing the offset at which Unicast queue need to be added in the Unicast queue list
<code>gport_array</code>	(IN) Pointer to array containing the Unicast queue gport

Description

Multiple set of Unicast queue id at a given index in the memory so that OOB Stats will be reported in the same order as the Unicast queue ids are programmed in the memory. In BCM56960, Users can again elect to report the instantaneous size of a subset of Unicast queue by writing the value 0x7FF into an entry in the `gport_array`.

Returns

bcm_oob_stats_queue_mapping_multi_get

Multiple get of OOB Stats Unicast queue list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_queue_mapping_multi_get(int unit, int array_max, int
*offset_array, bcm_gport_t *gport_array, int *array_count);
```

Parameters

unit	(IN) BCM Device Number
array_max	(IN) Maximum length of given array
offset_array	(IN) Pointer to array containing the offset at which Unicast Queue need to be added in the Unicast queue list
gport_array	(OUT) Pointer to array containing the Unicast queue gport
array_count	(OUT) Number of elements in array returned

Description

Multiple get of Unicast queue at a given index in the memory. OOB Stats will be reported in the same order as the Unicast queue gport are programmed in the memory. In BCM56960, Value 0x7FF into an entry in the gport_array indicates the end of reporting round of instantaneous Unicast queue stats.

Returns

bcm_oob_stats_queue_mapping_set

Set of OOB Stats Unicast queue list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_queue_mapping_set(int unit, int offset, bcm_gport_t gport);
```


Parameters

<code>unit</code>	(IN) BCM Device Number
<code>offset</code>	(IN) Offset at which Unicast queue need to be added in the Unicast queue list
<code>gport</code>	(IN) Unicast queue gport

Description

Set of Unicast queue id at a given index in the memory so that OOB Stats will be reported in the same order as the Unicast queue ids are programmed in the memory. In BCM56960, Users can again elect to report the instantaneous size of a subset of Unicast queue by writing the value 0x7FF into an entry in the gport.

Returns

bcm_oob_stats_queue_mapping_get

Get of OOB Stats Unicast queue list configuration

Syntax

```
#include <bcm/oob.h>
int bcm_oob_stats_queue_mapping_get(int unit, int offset, bcm_gport_t *gport);
```

Parameters

<code>unit</code>	(IN) BCM Device Number
<code>offset</code>	(IN) Offset at which Unicast Queue need to be read from the Unicast queue list
<code>gport</code>	(OUT) Pointer to the Unicast queue gport

Description

Get of Unicast queue at a given index in the memory. OOB Stats will be reported in the same order as the Unicast queue gport are programmed in the memory. In BCM56960, Value 0x7FF into an entry in the `gport_array` indicates the end of reporting round of instantaneous Unicast queue stats.

Returns

OPERATIONS, ADMINISTRATION, AND MAINTENANCE

Two new OAM flags are added for OAM Group structure Flag and Endpoint2 structure flag respectively.

Table 55: New OAM Group Structure Flag Definition

Flag	Description
BCM_OAM_GROUP_GET_FAULTS_ONLY	bcm_oam_group_get fetches only faults and persistent_faults fields in bcm_oam_group_info_t structure.

New endpoint types are added accordingly in this release for all endpoints on a given port and VLAN.

Table 56: New OAM Endpoint Types

Endpoint type	Description
bcmOAMEndpointTypePwStatus	Protection coordination state protocol endpoint
bcmOAMEndpointTypeMplsLmDmSectionPort	RFC 6374 Section loss/delay measurement based on Port
bcmOAMEndpointTypeMplsLmDmSectionInnervlan	RFC 6374 Section loss/delay measurement based on Port and Inner vlan
bcmOAMEndpointTypeMplsLmDmSectionOuterVlan	RFC 6374 Section loss/delay measurement based on Port and Outer vlan
bcmOAMEndpointTypeMplsLmDmSectionOuterPlusInnerVlan	RFC 6374 Section loss/delay measurement based on Port and Outer vlan and Inner vlan
bcmOAMEndpointTypeMplsLmDmPw	RFC 6374 Section loss/delay measurement based on Pseudowire
bcmOAMEndpointTypeMplsLmDmLsp	RFC 6374 Section loss/delay measurement based on LSP
bcmOAMEndpointTypeBhhSection	BHH Section based on Port
bcmOAMEndpointTypeBhhSectionInnervlan	BHH section based on Port and Inner vlan
bcmOAMEndpointTypeBhhSectionOuterVlan	BHH section based on Port and Outer vlan
bcmOAMEndpointTypeBhhSectionOuterPlusInnerVlan	BHH section based on Port and Outer vlan and Inner vlan

New OAM events are added accordingly in this release.

Table 57: New OAM Event Types

Event type	Description
bcmOAMEventBHHCCMUnknownPeriod	Unknown Period has been detected on received BHH CCM PDU
bcmOAMEventBHHCCMUnknownPriority	Unknown Priority has been detected on received BHH CCM PDU
bcmOAMEventBHHCCMRdiClear	RDI has been cleared on received BHH CCM PDU
bcmOAMEventBHHCCMUnknownMegLevelClear	Unknown Meg Level defect has been cleared on received BHH CCM PDU
bcmOAMEventBHHCCMUnknownMegIdClear	Unknown Meg Id defect has been cleared on received BHH CCM PDU
bcmOAMEventBHHCCMUnknownMepIdClear	Unknown Mep Id defect has been cleared on received BHH CCM PDU
bcmOAMEventBHHCCMUnknownPeriodClear	Unknown Period defect has been cleared on received BHH CCM PDU
bcmOAMEventBHHCCMUnknownPriorityClear	Unknown Priority defect has been cleared on received BHH CCM PDU

New OAM opcode action formats are added accordingly in this release.

Table 58: OAM Opcode Action Formats

Action Format Type	Description
bcmOAMActionLowMdlProcessInInternalOamEngine	Process Low MDL packets in internal OAM/CCM engine.
bcmOAMActionL2StationMissProcessInInternalOamEngine	Process L2 station miss packets in internal CCM engine.
bcmOAMActionL2StationHitProcessInInternalOamEngine	Process L2 station hit packets in internal CCM engine.
bcmOAMActionMCProcessInInternalOamEngine	Process Multicast OAM packets in internal OAM engine.
bcmOAMActionSampleTimeStamp	Sample the time stamp value to DM packet.
bcmOAMActionSampleLmCounter	Sample the counter value to LM packet.

A new OAM control format is added accordingly in this release.

Table 59: New OAM Control Format

Control Format Type	Description
bcmOamControlUnknownAchCount	Count unknown ACH or Any VCCV packet with or without ACH.

Fields of `bcm_oam_endpoint_info_t` data structure are added for OAM endpoint info structure, which is equivalent to an 802.1ag Maintenance Endpoint (MEP).

```
typedef struct bcm_oam_endpoint_info_s {
    ...
    /* Out gport used for TX counting by BHH endpoints. */
    bcm_gport_t mpls_out_gport;
    ...
    /*
    * Offset in bytes from TOP of the MPLS label stack from where the
payload
    * starts for byte count computation.
    */
    uint32 lm_payload_offset;

    /*
value for
    * Offset of the Label from top of the stack which gives the EXP
    * deriving the COS value - valid values are 0/1/2.
    */
    uint32 lm_cos_offset;

    /* BYTE/PACKET. */
    bcm_oam_lm_counter_type_t lm_ctr_type;

    /* Valid values are 32/64. */
    bcm_oam_lm_counter_size_t lm_pdu_ctr_field_size;

    /* OAM priority map id. */
    uint32 pri_map_id;
} bcm_oam_endpoint_info_t;
```

New OAM Macros for OAM endpoint action operations are added in this release.

Table 60: OAM Endpoint Action Set Macros

Macro	Description
BCM_OAM_LOOKUP_TYPE_SET(_type, _lookup_type)	Set the specified lookup type in the collection
BCM_OAM_LOOKUP_TYPE_GET(_type, _lookup_type)	Get the status of the specified lookup type in the collection
BCM_OAM_LOOKUP_TYPE_CLEAR(_type, _lookup_type)	Clear the specified lookup type in the collection
BCM_OAM_LOOKUP_TYPE_CLEAR_ALL(_type)	Clear all lookup types in the collection
BCM_OAM_CONDITION_SET(_condition, _drop)	Set the specified drop condition in the collection
BCM_OAM_CONDITION_GET(_condition, _drop)	Get the status of the specified drop condition in the collection
BCM_OAM_CONDITION_CLEAR(_condition, _drop)	Clear the specified drop condition in the collection
BCM_OAM_CONDITION_CLEAR_ALL(_condition)	Clear all drop conditions in the collection

New OAM Drop condition control and OAM lookup types are added with `bcm_oam_conditions_t` and `bcm_oam_lookup_types_t` in this release.

Table 61: New OAM Drop condition control types

Control Types	Description
bcmOamConditionPFCDrop	PFC frames marked for drop
bcmOamConditionFlowControlDrop	VOQ Flow control packet marked for drop
bcmOamConditionMacsaEqualsMacdaDrop	Packet marked for drop by the DoS prevention logic due to MAC SA = MAC DA condition set via <code>bcmSwitchDosAttackMACSAEqualMACDA</code>
bcmOamConditionAutoMulticastTunnelControlPacketDrop	AMT(Auto Multicast Tunnel) control packet marked for drop due to AMT tunnel configuration being set to not to send AMT control packet to CPU
bcmOamConditionUnknownSubportErrorDrop	Packet marked for drop for coming from an unknown subrending port
bcmOamConditionSubportPktTagDrop	Packet with link layer tag marked for drop. This happens when <code>bcmPortControlSubportPktTagDrop</code> control is set to drop packets with LLTAG(TR101)
bcmOamConditionNoSubportPktTagDrop	Packet without link layer tag marked for drop. This happens when <code>bcmPortControlNoSubportPktTagDrop</code> control is set to drop packets with link layer tag(TR101)
bcmOamConditionIngressFilterDrop	Packet marked for drop due to port not being a member of VLAN. This happens when VLAN filtering is enabled by passing <code>BCM_PORT_VLAN_MEMBER_INGRESS</code> flags to <code>bcm_port_vlan_member_set</code> API
bcmOamConditionEgressFilterDrop	Packet marked for drop due to port not being a member of VLAN. This happens when VLAN filtering is enabled by passing <code>BCM_PORT_VLAN_MEMBER_EGRESS</code> flags to <code>bcm_port_vlan_member_set</code> API
bcmOamConditionInvalidTpidDrop	Packet marked for drop due to mismatch between packet VLAN TPID and configured VLAN TPID value (in VLAN profile)

Table 61: New OAM Drop condition control types

Control Types	Description
bcmOamConditionOamMetadataHeaderErrorDrop	Packet marked for drop due to OLP (OAM metadata) header error
bcmOamConditionFieldLookupDrop	Packet marked for drop due to VFP action
bcmOamConditionSATDrop	Packet marked for drop by SAT
bcmOamConditionVlanTranslateIngressMissDrop	Packet marked for drop due to vlan translate miss. This drop is configured via VLAN control bcmVlanTranslateIngressMissDrop
bcmOamConditionVlanTranslateEgressMissDrop	Packet marked for drop due to vlan translate miss. This drop is configured via VLAN control bcmVlanTranslateEgressMissDrop
bcmOamConditionMplsLabelLookupMissDrop	Packet marked for drop due to MPLS label lookup miss
bcmOamConditionMplsInvalidActionDrop	Packet marked for drop due to MPLS lookup result configured with invalid action
bcmOamConditionMplsInvalidPayloadDrop	Packet marked for drop by MPLS processing due to invalid payload
bcmOamConditionMplsTtlErrorDrop	Packet marked for drop due to MPLS TTL check fail
bcmOamConditionStgNotInForwardDrop	Packet marked for drop due to Spanning tree not being in FORWARDING_STATE
bcmOamConditionEgressTtlErrorDrop	Packet marked for drop due to TTL error
bcmOamConditionEgressStgBlockedDrop	Packet marked for drop due to Spanning tree in blocking state
bcmOamConditionEgressStgDisabledDrop	Packet marked for drop due to Spanning tree in disable state
bcmOamConditionEgressIPMCSameVlanPruneDrop	IPMC packet marked for drop due to bcmSwitchIpmcSameVlanPruning setting. When this is set, trying to do L2 switching for an IPMC packet results in packet drop
bcmOamConditionEgressMMUErrorDrop	Packet marked for drop due to MMU error - purge/age
bcmOamConditionEgressBadTdipDrop	Packet marked for drop due to bad tunnel destination IP in case of 6to4 tunnels and ISATAP tunnels - possible reasons are incorrect prefix for IPv6 DIP, IPv4 DIP/SIP is BC/Zero/loopback/Class-D , IPv4 SIP = DIP
bcmOamConditionEgressUnknownVlanDrop	Packet marked for drop due to unknown VLAN. This happens when bcmVlanDropUnknown vlan control is used to set dropping of unknown VLAN packets
bcmOamConditionEgressCompositeErrorDrop	Packet marked for drop due to parity error from memories or ISM lookup fail
bcmOamConditionEgressSplitHorizonDrop	Packet marked for drop due to split horizon check. This is done by enabling pruning for DVP network group via bcm_switch_network_group_config_set with BCM_SWITCH_NETWORK_GROUP_EGRESS_PRUNE_ENABLE
bcmOamConditionEgressSVPRemovalDrop	Packet marked for drop due to virtual port pruning - SVP = DVP and VP based pruning is enabled (DISABLE_VP_PRUNING is set to 0 in Source VP table)
bcmOamConditionEgressNonUCPruneDrop	Non-unicast packet marked for drop due to drop condition set in egress next hop via bcm_mpls_port_add API with BCM_MPLS_PORT_DROP flag
bcmOamConditionCount	Always Last. Not a usable value

```
typedef struct bcm_oam_conditions_s {
    SHR_BITDCL cbits[_SHR_BITDCLSIZE(bcmOamConditionCount)];
} bcm_oam_conditions_t;
```

Table 62: New OAM lookup types

Lookup Types	Description
bcmOamLookupTypePort	Port based Ethernet OAM
bcmOamLookupTypeInnerVlan	Inner Vlan based Ethernet OAM
bcmOamLookupTypeOuterVlan	Outer Vlan based Ethernet OAM
bcmOamLookupTypeOuterPlusInnerVlan	Outer+inner vlan based Ethernet OAM
bcmOamLookupTypeVirtualPort	VP based Ethernet OAM
bcmOamLookupTypeOuterVlanPassive	Outer Vlan based passive Ethernet OAM
bcmOamLookupTypeMplsSection	MPLS section OAM
bcmOamLookupTypeMplsLsp	MPLS LSP OAM
bcmOamLookupTypeMplsPw	MPLS Pseudo-wire OAM
bcmOamLookupTypeEgressPort	Egress Port based Ethernet OAM
bcmOamLookupTypeEgressInnerVlan	Egress inner VLAN based Ethernet OAM
bcmOamLookupTypeEgressOuterVlan	Egress Outer Vlan based Ethernet OAM
bcmOamLookupTypeEgressOuterPlusInnerVlan	Egress outer+inner VLAN based Eth-OAM
bcmOamLookupTypeEgressVirtualPort	Egress VP based Ethernet OAM
bcmOamLookupTypeEgressOuterVlanPassive	Egress outer Vlan based pass Eth-OAM
bcmOamLookupTypeEgressMplsSection	Egress MPLS section OAM
bcmOamLookupTypeEgressMplsLsp	Egress MPLS LSP OAM
bcmOamLookupTypeEgressMplsPw	Egress MPLS Pseudo-wire OAM
bcmOamLookupTypeCount	Always Last. Not a usable value

```
typedef struct bcm_oam_lookup_types_s {
    SHR_BITDCL_t bits[_SHR_BITDCLSIZE(bcmOamLookupTypeCount)]
} bcm_oam_lookup_types_t;
```

New MPLS LM DM opcode types are added.

Table 63: New MPLS LM DM opcode types

MPLS LM DM Types	Corresponding opcode	Description
bcmOamMplsLmDmOpcodeTypeLm	0	RFC 6374 Loss Measurement packet
bcmOamMplsLmDmOpcodeTypeDm	1	RFC6374 Delay Measurement Packet
bcmOamMplsLmDmOpcodeTypeIlm	2	RFC6374 Inferred Loss Measurement Packet
bcmOamMplsLmDmOpcodeTypeLmP lusDm	3	RFC 6374 LM+DM packet
bcmOamMplsLmDmOpcodeTypeIlmP lusDm	4	RFC 6374 ILM+DM packet

New OAM LM Counter types are added.

Table 64: New OAM LM counter types

OAM LM Counter Types	Description
bcmOamLmCounterTypePkt	Counts packets
bcmOamLmCounterTypeByte	Counts bytes

Table 64: New OAM LM counter types

OAM LM Counter Types	Description
bcmOamLmCounterTypeCount	Always the last value. Not usable.

New OAM LM Counter sizes are added.

Table 65: New OAM LM counter sizes

OAM LM Counter sizes	Description
bcmOamLmCounterSize32Bit	Counter size is 32 bit
bcmOamLmCounterSize64Bit	Counter size is 64 bit
bcmOamLmCounterSizeCount	Always the last value. Not usable.

API signature of `bcm_oam_performance_event_register` has changed where the parameter `event_data` is removed.

bcm_oam_performance_event_register

Register a handler for OAM performance events.

Syntax

```
#include <bcm/oam.h>
int bcm_oam_performance_event_register(
    int unit,
    bcm_oam_event_types_t event_types,
    bcm_oam_performance_event_cb cb,
    void *user_data);
```

Parameters

<code>unit</code>	BCM device number
<code>event_types</code>	(IN) A collection of event types for which the specified callback will be called (May only be one of the events defined to be performance events).
<code>cb</code>	(IN) Pointer to the callback function
<code>user_data</code>	(OUT) Pointer to arbitrary user data to be passed back to the callback

Description

Registers the specified callback to handle the performance event types specified in the collection.

Returns

New OAM APIs are introduced in this release.

bcm_oam_lookup_enable_set

Enable/Disable a drop condition for an OAM lookup type

Syntax

```
#include <bcm/oam.h>
int bcm_oam_lookup_enable_set(int unit, bcm_oam_lookup_type_t type,
bcm_oam_condition_t condition, int enable);
```

Parameters

unit	(IN) BCM device number
type	(IN) Type of lookup for which a particular drop condition is enabled/disabled.
condition	(IN) The drop condition that needs to be enabled/disabled.
enable	(IN) Enable/Disable.

Description

This API allows enabling or disabling a drop condition for an OAM lookup type.

Returns

bcm_oam_lookup_enable_multi_set

Enable/Disable multiple drop conditions for OAM lookup types.

Syntax

```
#include <bcm/oam.h>
int bcm_oam_lookup_enable_multi_set(int unit, bcm_oam_lookup_types_t types,
bcm_oam_conditions_t conditions, int enable);
```

Parameters

unit	(IN) BCM device number
types	(IN) Types of lookup for which drop conditions are enabled/disabled.
condition	(IN) The drop conditions that need to be enabled/disabled.
enable	(IN) Enable/Disable.

Description

This API allows enabling or disabling multiple drop conditions for OAM lookup types.

Returns

bcm_oam_lookup_enable_get

Get the drop conditions that are ignored for an OAM lookup type

Syntax

```
#include <bcm/oam.h>
int bcm_oam_lookup_enable_get(int unit, bcm_oam_lookup_type_t type,
bcm_oam_conditions_t *condition);
```

Parameters

unit	(IN) BCM device number
type	(IN) Type of lookup for which a particular drop condition is disabled.
conditions	(OUT) The drop conditions that are disabled.

Description

This API allows getting the disabled drop conditions for an OAM lookup type.

Returns

PACKET TRANSMIT AND RECEIVE

The default value for match rule which SDK uses to identify OLP packets is 255.

- `define BCM_DEFAULT_OLP_MATCH_RULE 255 /* Olp Match Rule (default) */`

New data types for OAM packet TX/RX are added accordingly in this release while the OAM counter max object is defined to 3.

- `define BCM_PKT_OAM_COUNTER_MAX 3`

Table 66: OAM Packet type Descriptions

Name	Description
<code>bcmPktRxOamTypeNone</code>	Not OAM packet.
<code>bcmPktRxOamTypeBfdOam</code>	BFD OAM packet.
<code>bcmPktRxOamTypeEthOamCcm</code>	Ethernet OAM CCM packet.
<code>bcmPktRxOamTypeEthOamLm</code>	Ethernet OAM LM packet.
<code>bcmPktRxOamTypeEthOamDm</code>	Ethernet OAM DM packet.
<code>bcmPktRxOamTypeEthOamOther</code>	Ethernet OAM other opcode packet.
<code>bcmPktRxOamTypeBhhOamCcm</code>	BHH OAM CCM packet.
<code>bcmPktRxOamTypeBhhOamLm</code>	BHH OAM LM packet.
<code>bcmPktRxOamTypeBhhOamDm</code>	BHH OAM DM packet.
<code>bcmPktRxOamTypeBhhOamOther</code>	BHH OAM other opcode packet.
<code>bcmPktRxOamTypeRfc6374Dlm</code>	MPLS LM/DM DLM packet.
<code>bcmPktRxOamTypeRfc6374Dm</code>	MPLS LM/DM DM packet.
<code>bcmPktRxOamTypeRfc6374DlmPlusDm</code>	MPLS LM/DM DLM + DM packet.
<code>bcmPktRxOamTypeRfc6374Ilm</code>	MPLS LM/DM ILM packet.
<code>bcmPktRxOamTypeRfc6374IlmPlusDm</code>	MPLS LM/DM ILM + DM packet.
<code>bcmPktRxOamTypeSat</code>	Service Activation Test.
<code>bcmPktRxOamTypeOtherAch</code>	OAM packets with other ACH types.
<code>bcmPktRxOamTypeCount</code>	Always last, not a usable value.

Table 67: OAM counter object ID type Description

Name	Description
<code>bcmOamCounterObjectNone</code>	Invalid counter object.

Table 67: OAM counter object ID type Description

Name	Description
bcmOamCounterObjectEndpointId	OAM counter object id.
bcmOamCounterObjectFlexStatId	OAM counter flex stats ID.

Table 68: OAM counter Structure Description

Field	Type	Description
counter_object	bcm_pkt_oam_counter_object_t	OAM counter object type. Refer pkt_oam_counter_object_table (page 98)
counter_object_id	uint32	Counter Object Id.
counter_offset	uint32	Offset from start of counter group.
counter_mode	bcm_pkt_oam_lm_counter_mode_t	Counter mode during Tx. Refer to pkt_oam_lm_counter_mode_table .
counter_value_upper	uint32	Upper 32 bit of OAM LM Counter in Rx Packet.
counter_value_lower	uint32	Lower 32 bit of OAM LM Counter in Rx Packet.
oam_lm_byte_count_offset	uint32	Offset to start byte counting.

New fields in `bcm_pkt_t` for OAM packet TX/RX are added with the corresponding flags in the continued flags2 of packet flag type.

Table 69: New fields in bcm_pkt_t Structure with Description

Field	Type	Description
oam_lm_replacement_offset	uint8	Replacement offset for LM counter in Bytes
rx_oam_pkt_type	bcm_pkt_rx_oam_type_t	OAM Pkt Type.
oam_counter	bcm_pkt_oam_counter_t	OAM counter array with BCM_PKT_OAM_COUNTER_MAX elements
oam_counter_size	uint32	Size of Counter array
_olp_hdr	uint8	OLP Header (network byte order) in 20 types.

Normally, when a packet is sent by the Host CPU to a OLP port, the values placed in the OLP header are derived by the SDK from `bcm_pkt_t` fields. In cases where the application instead wishes to specify its own OLP header, this field must be populated with the raw data in network byte order, and the flag `BCM_PKT_F2_OLP_READY` set.

Table 70: Packet flags2 Flags Descriptions

Flag	Description
BCM_PKT_F2_OAM_TX	Indicates OAM Packet Tx
BCM_PKT_F2_OLP_READY	Indicates OLP already constructed in field <code>_olp_hdr</code>

Table 70: Packet flags2 Flags Descriptions

Flag	Description
BCM_PKT_F2_LM_REPLACEMENT_OFFSET	This flag indicates that oam_lm_replacement_offset carries LM Counters replacement offset (in bytes) and oam_replacement_offset carries timestamp replacement offset (in bytes). This flag is used when LM and DM are carried in single packet.

New rx_reasons are added in this release.

Table 71: New Packet RX Reasons

Reason	Description
bcmRxReasonUnknownSubtendingPort	Unknwon subtending port packets to CPU
bcmRxReasonPacketTraceToCpu	Instrumentation copy packets to CPU

A new rx trap code is added in this release.

Table 72: New Rx Trap Code.

Trap Code	Description
bcmRxTrapOamBfdIpv6	Bfd over Ipv6 trap code
bcmRxTrapOamMipSnoop2ndPass	Snoop OAM packets with egress info (out port).
bcmRxTrapSat0	Sat trap code No 0
bcmRxTrapSat1	Sat trap code No 1
bcmRxTrapSat2	Sat trap code No 2

A new flag in the trap/snoop configuration structures defined in bcm_pkt_trap_flags.

Table 73: New BCM RX Trap Flags

Name	Purpose
BCM_RX_TRAP_UPDATE_COUNTER_2	update counter No.2
BCM_RX_TRAP_UPDATE_FORWARDING_HEADER	Override the Forwarding header
BCM_RX_TRAP_UPDATE_ENCAP_ID	Update Encap-id
BCM_RX_TRAP_DROP_AND_LEARN	Enable learning for dropped packets

Table 74: New BCM RX Snoop Flags

Name	Purpose
BCM_RX_SNOOP_UPDATE_ENCAP_ID	Update Encap-id

The bcm_rx_trap_config_t structure is used to configure device trap function with the below new fields and above flags. Similarly, the bcm_rx_snoop_config_t structure is used to configure device snoop function with the below new fields and above flags.

```
/* User-configurable, per-unit RX configuration. */
```

```
typedef struct bcm_rx_trap_config_s {
    ...
    /* Forwarding header position overridden value */
    bcm_rx_trap_forwarding_header_t forwarding_header;
    /* Encap-ID */
    bcm_if_t encap_id;
} bcm_rx_trap_config_t;

/* User-configurable, per-unit RX configuration. */
typedef struct bcm_rx_snoop_config_s {
    ...
    bcm_if_t encap_id;          /* Encap-ID */
} bcm_rx_snoop_config_t;
```

Table 75: Forwarding header position overridden value *bcm_rx_trap_forwarding_header_t*

Name	Description
bcmRxTrapForwardingHeaderPacketStart	Start of packet.
bcmRxTrapForwardingHeaderL2Header	Ethernet header.
bcmRxTrapForwardingHeaderFirstHeader	First header after Ethernet.
bcmRxTrapForwardingHeaderSecondHeader	Second header after Ethernet.
bcmRxTrapForwardingHeaderThirdHeader	Third header after Ethernet.
bcmRxTrapForwardingHeaderFourthHeader	Fourth header after Ethernet.
bcmRxTrapForwardingHeaderCount	Always Last. Not a usable value.

PORT CONFIGURATION

A new config field in `bcm_port_config_t` has been added for API `bcm_port_config_get()`.

Table 76: *bcm_port_config_t*

Field	BCM Type	Description
nif	bcm_pbmp_t	Bitmap or logical port list of all network ports

Two new fields in `bcm_port_interface_info_t` and `BCM_PORT_ADD_USE_PHY_PBMP` flag are added to allows for dynamic NIF port configuration changes with `bcm_port_add`, `bcm_port_get` and `bcm_port_remove` APIs.

Table 77: *bcm_port_interface_config_t*

Field	Type	Description
phy_pbmp	bcm_pbmp_t	bitmap of phy lanes the interface occupies
interface_id	uint32	should be indicated in case the id can't be derived from <code>phy_port</code> .

Table 78: Port add Flags

Flag	Meaning
BCM_PORT_ADD_USE_PHY_PBMP	The phy pots occupied by the interface are indicated by phy_pbmp in interface_info struct

A new port classification ID is added to aggregate a group of ports for further processing such as VLAN translation and field processing in bcm_port_class_set/get APIs.

The class types are shown in bcm_port_class_t:

New port PHY specific configurations are added for bcm_port_phy_control_set/get APIs.

Table 79: bcm_port_phy_control_t

BCM_PORT_PHY_CONTROL_INTR_MASK	PHY interrupt enable mask (a bit set to 1 means interrupt enabled)
BCM_PORT_PHY_CONTROL_INTR_STATUS	PHY interrupt control mask (write 1 to clear)
BCM_PORT_PHY_CONTROL_XSW_LANE_MAP	Cross-switch lane map. Bits[3:0] map Rx lane 0, bits [7:4] map Rx lane 1, etc. Default mapping is 0x76543210.
BCM_PORT_PHY_CONTROL_FIRMWARE_DFE_ENABLE	SerDes microcontroller mode to enable DFE
BCM_PORT_PHY_CONTROL_FIRMWARE_LP_DFE_ENABLE	SerDes microcontroller mode to enable low power DFE
BCM_PORT_PHY_CONTROL_FIRMWARE_BR_DFE_ENABLE	SerDes microcontroller mode to enable digital feedback equalization
BCM_PORT_PHY_CONTROL_GPIO_CONFIG	The 32-bit control value is divided into eight 4-bit configuration values. Bits [3:0] controls GPIO pin 0, bits [7:4] controls GPIO pin 1, etc The configuration value may be on of the following: BCM_PORT_PHY_CONTROL_GPIO_CONFIG_DISABLED (0) BCM_PORT_PHY_CONTROL_GPIO_CONFIG_OUTPUT (1) BCM_PORT_PHY_CONTROL_GPIO_CONFIG_INPUT (2) For example, in order to configure GPIO pins 0 and 1 as output and GPIO pins 2 and 3 as input, then the following value must be supplied: 0x00002211
BCM_PORT_PHY_CONTROL_GPIO_VALUE	The 32-bit control value is divided into eight 4-bit pin values. Bits [3:0] is the value of GPIO pin 0, bits [7:4] is the value of GPIO pin 1, etc. Pin value must be either 0 or 1. New pin values are only applied to pins configured as output pins. For example, in order to set output GPIO pins 0 and 1 to 0 and 1 respectively, then the following value must be supplied: 0x00000010 If input GPIO pins 2 and 3 are both 1, then the PHY control value returned by the API will be: 0x00001100

A new field vpn has been added in the bcm_port_match_info_t structure that describes the match criteria.

```
typedef struct bcm_port_match_info_s {
    ...
    bcm_vpn_t vpn; /* VPN ID */
} bcm_port_match_info_t;
```

The `bcm_port_phy_timesync_config_set/get` APIs with the `BCM_PORT_PHY_TIMESYNC_VALID_TIMER_ADJUSTMENT` and `BCM_PORT_PHY_TIMESYNC_VALID_PHY_1588_INBAND_CONTROL` has new flags while the original names also have slight changes by removing the TS word in the name. It's the similar change for `bcm_port_phy_timesync_inband_control_t`, which is renamed from `bcm_port_phy_timesync_inband_ts_control_t` and adds new field `bcm_port_phy_timesync_timer_mode_t`. The corresponding supported timer modes are also added. Two new fields have been added in `bcm_port_phy_timesync_config_t` data structure.

Table 80: Updated Port Phy timesync INBAND flags (Newly added and names changed)

BCM_PORT_PHY_TIMESYNC_INBAND_RESV0_ID_CHECK	Enable inband resv0 id check
<code>BCM_PORT_PHY_TIMESYNC_INBAND_SYNC_ENABLE</code>	Enable inband timestamping for Sync
<code>BCM_PORT_PHY_TIMESYNC_INBAND_DELAY_REQUEST_ENABLE</code>	Enable inband timestamping for Delay Request
<code>BCM_PORT_PHY_TIMESYNC_INBAND_PDELAY_REQUEST_ENABLE</code>	Enable inband timestamping for PDelay Request
<code>BCM_PORT_PHY_TIMESYNC_INBAND_PDELAY_RESPONSE_ENABLE</code>	Enable inband timestamping for PDelay Response
<code>BCM_PORT_PHY_TIMESYNC_INBAND_RESV0_ID_UPDATE</code>	Enables update of resv0 field in the ingress packet with the <code>resv0_id</code>
<code>BCM_PORT_PHY_TIMESYNC_INBAND_FOLLOW_UP_ASSIST</code>	Enable inband Follow_Up (T1) assist
<code>BCM_PORT_PHY_TIMESYNC_INBAND_DELAY_RESPONSE_ASSIST</code>	Enable inband Delay_Response (T3) assist
<code>BCM_PORT_PHY_TIMESYNC_INBAND_TIMER_MODE_SELECT</code>	Select 80- or 48-bit counter for inband PTP messages
<code>BCM_PORT_PHY_TIMESYNC_INBAND_CAP_SRC_PORT_CLK_ID</code>	By default, MAC DA is captured and saved in the memory/FIFO. To capture source port clock ID instead, this flag can be used. Only one of them can be selected at a time.
<code>BCM_PORT_PHY_TIMESYNC_INBAND_MATCH_VLAN_ID</code>	Enable the comparison of the VLAN ID field during delay response(T3) and follow up packet.
<code>BCM_PORT_PHY_TIMESYNC_INBAND_MATCH_SRC_PORT_NUM</code>	Enable the comparison of the source port number during delay response(T3) and follow up packet.
<code>BCM_PORT_PHY_TIMESYNC_INBAND_MATCH_MAC_ADDR</code>	Enable the comparison of the MAC address during delay response(T3) and follow up packet.
<code>BCM_PORT_PHY_TIMESYNC_INBAND_MATCH_IP_ADDR</code>	Enable the comparison of the IP address during delay response(T3) and follow up packet.

Table 81: `bcm_port_phy_timesync_inband_control_t`

flags	Inband Timestamping flags (BCM_PORT_PHY_TIMESYNC_INBAND_* flags)
<code>resv0_id</code>	resv0 ID
<code>timer_mode</code>	Timer mode

Table 82: TimeSync timer mode (bcm_port_phy_timesync_timer_mode_t)

Timer mode	Description
bcmPortPhyTimesyncTimerModeNone	No TimeSync timer
bcmPortPhyTimesyncTimerModeDefault	Use the default TimeSync timer
bcmPortPhyTimesyncTimerMode32bit	Use 32-bit TimeSync timer
bcmPortPhyTimesyncTimerMode48bit	Use 48-bit TimeSync timer
bcmPortPhyTimesyncTimerMode64bit	Use 64-bit TimeSync timer
bcmPortPhyTimesyncTimerMode80bit	Use 80-bit TimeSync timer

```

/* Timer adjustment type. */
typedef struct bcm_port_phy_timesync_timer_adjust_s {
    bcm_port_phy_timesync_timer_mode_t mode; /* Timer adjustment mode */
    int delta; /* Timer adjustment delta value */
} bcm_port_phy_timesync_timer_adjust_t;

```

Table 83: bcm_port_phy_timesync_config_t

Field	BCM Type	Description
timer_adjust	bcm_port_phy_timesync_timer_adjust_t	Direct timer adjustment
inband_control	bcm_port_phy_timesync_inband_control_t	Inband TS control

Below new GPORT types are added in this release.

Table 84: Port CONTROL CPU Port values

BCM_GPORT_LOCAL_CPU_IEEE	CPU port only for ethernet packets
BCM_GPORT_LOCAL_CPU_HIGIG	CPU port only for higig packets

The CPU port accepts both regular Ethernet packets and HiGig(2) packets at the same time, so it has has different CPU port number for Ethernet packets and Higig packets. BCM_GPORT_LOCAL_CPU_IEEE is used for ethernet packets, and BCM_GPORT_LOCAL_CPU_HIGIG is used for higig packets. For example: Enable IPv4 L3 forwarding for HiGig packets: `bcm_port_control_set(unit, BCM_GPORT_LOCAL_CPU_HIGIG, bcmPortControlIP4, enable);` Enable IPv4 L3 forwarding for Ethernet packets: `bcm_port_control_set(unit, BCM_GPORT_LOCAL_CPU_IEEE, bcmPortControlIP4, enable);`

New port control types have been added for `bcm_port_control_get/set` APIs with the following tables defining the values for some type enumerations as well as Port CONTROL SAMPLE DEST Flags.

The new feature types are shown as below:

Table 85: bcm_port_control_t

bcmPortControlOOBFCTxReportingEnable	OOB FC Tx Ingress congestion notification enable/disable per port.
bcmPortControlOOBFCTxReportEgrEnable	OOB FC Tx Egress congestion notification enable/disable per port.
bcmPortControlMplsEncapsulateExtendedLabel	Encapsulate additional MPLS tunnel on a given outlif
bcmPortControlSampleIngressDest	Set Ingress sFlow sample destination. See BCM_PORT_CONTROL_SAMPLE_DEST_flags (page 105) .
bcmPortControlSampleFlexDest	Set Flex sFlow sample destination. See BCM_PORT_CONTROL_SAMPLE_DEST_flags (page 105) .
bcmPortControlSampleFlexRate	Set Flex sFlow sampling rate per port. Every 1/value ingressing packets will be sampled enabled by FP action (0 indicate no sampling, 1 indicates sampling all packets). Range of values: 0x0-0x10000000.

Table 86: Port CONTROL SAMPLE DEST Flags

Flag	Meaning
BCM_PORT_CONTROL_SAMPLE_DEST_CPU	Copy packet samples to CPU
BCM_PORT_CONTROL_SAMPLE_DEST_MIRROR	Copy packet samples to all mirror destinations created with BCM_MIRROR_PORT_SFLOW flag

A new flag regarding the mapping of priority and DSCP is added for the APIs `bcm_port_dscp_map_set()` and `bcm_port_dscp_map_get()` .

Table 87: New BCM_PRIO_XXX flag

Flag	Meaning
BCM_PRIO_SECONDARY	Configure the In-PP-Port TC DP TOS index.

New Port interface types are added to configure the physical interface between the MAC and the PHY for the specified port in `bcm_port_interface_get` and `bcm_port_interface_set`.

Table 88: New Port Interface Values

BCM_PORT_IF_KR2	Backplane 2x10GbE 64B/66B interface
BCM_PORT_IF_CR2	Copper 2x10GbE 64B/66B interface
BCM_PORT_IF_SR2	Fiber 2x10GbE 64B/66B interface

A new encapsulation mode is added for `encap` field in `bcm_port_encap_config_t` used for `bcm_port_encap_config_set` and `bcm_port_encap_config_get` APIs while another new field `hgoe_reserved` is also added in `bcm_port_encap_config_t` data structure.

Table 89: Port Encapsulation Modes

Mode	Meaning
BCM_PORT_ENCAP_HIGIG_OVER_ETHERNET	Broadcom Proprietary Higig over Ethernet encapsulation.

Table 90: bcm_port_encap_config_t

Field	Type	Description
hgoe_reserved	uint16	Value for reserved field in HGoE header

New Port APIs are added in this release.

bcm_port_gport_add

Add a GPORT ID for the specified physical port.

Syntax

```
#include <bcm/port.h>
int bcm_port_gport_add(int unit, bcm_port_t local_port, bcm_gport_t modport)
```

Parameters

unit	BCM device number
local_port	Local front panel port number
modport	GPORT ID

Description

This API can be used to add a GPORT ID corresponding to a local port number. The supplied port number should return FALSE on a BCM_GPORT_IS_SET(port) test.

The GPORT ID returned in this API is a MODPORT type. Therefore this call will fail on devices without a module ID.

Returns

bcm_port_gport_get_all

Get all the GPORT ID for the specified physical port.

Syntax

```
#include <bcm/port.h>
int bcm_port_gport_get_all(int unit, bcm_port_t local_port, int size,
    bcm_gport_t *gport_array, int *count)
```

Parameters

unit	BCM device number
local_port	Local front panel port number
size	Max size of gport array
gport_array	GPORT array
count	GPORT count

Description

This API can be used to get all the GPORT IDs corresponding to a local port number. The supplied port number should return FALSE on a BCM_GPORT_IS_SET(port) test.

The GPORT IDs returned in this API are all MODPORT type. Therefore this call will fail on devices without a module ID.

Returns

bcm_port_gport_delete

Del the GPORT ID for the specified physical port.

Syntax

```
#include <bcm/port.h>
int bcm_port_gport_delete(int unit, bcm_port_t local_port, bcm_gport_t modport)
```

Parameters

<code>unit</code>	BCM device number
<code>local_port</code>	Local port number
<code>modport</code>	GPORT ID

Description

This API can be used to delete the GPORT ID corresponding to a local port number. The supplied port number should return FALSE on a `BCM_GPORT_IS_SET(port)` test.

The GPORT ID returned in this API is a MODPORT type. Therefore this call will fail on devices without a module ID.

Returns

bcm_port_gport_delete_all

Delete all the GPORT IDs for the specified physical port.

Syntax

```
#include <bcm/port.h>
int bcm_port_gport_delete_all(int unit, bcm_port_t local_port)
```

Parameters

<code>unit</code>	BCM device number
<code>local_port</code>	Local port number

Description

This API can be used to delete all the GPORT IDs corresponding to a local port number. The supplied port number should return FALSE on a `BCM_GPORT_IS_SET(port)` test.

Returns

bcm_port_wide_data_set

Configure additional data for a wide LIF.

Syntax

```
#include <bcm/port.h>
int bcm_port_wide_data_set(
    int unit,
    bcm_gport_t gport,
    uint32 flags,
    uint64 data);
```

Parameters

unit	(IN) Unit number.
gport	(IN) gport id (global lif type)
flags	(IN) BCM_PORT_WIDE_DATA_XXX
data	(INOUT) (for "_get") additional data for lif

Description

Configure additional data for a wide LIF.

Table 91: Port Wide data flags

Name	Description
BCM_PORT_WIDE_DATA_INGRESS	additional data for ingress lif
BCM_PORT_WIDE_DATA_EGRESS	additional data for egress lif

Returns

bcm_port_wide_data_get

Retrieve additional data for a wide LIF.

Syntax

```
#include <bcm/port.h>
int bcm_port_wide_data_get(
    int unit,
    bcm_gport_t gport,
    uint32 flags,
    uint64 *data);
```

Parameters

unit	(IN) Unit number.
gport	(IN) gport id (global lif type)
flags	(IN) BCM_PORT_WIDE_DATA_XXX
data	(INOUT) (for "_get") additional data for lif

Description

Configure/retrieve additional data for a wide LIF.

Returns

PRECISION TIME PROTOCOL

A new field `grandmaster_identity` is added for `bcm_ptp_foreign_master_entry_t` data structure which the Foreign Master Dataset contains information about remote peers that are sending PTP Announce packets and could be used as a master clock for synchronization.

```
typedef struct bcm_ptp_foreign_master_entry_s {
    ...
    bcm_ptp_clock_identity_t grandmaster_identity; /* Grandmaster Id field
of announce msg. */
```

```
} bcm_ptp_foreign_master_entry_t;
```

QUALITY OF SERVICE

New QoS Configuration Flags that can be used with the QoS APIs.

- BCM_QOS_MAP_OAM_INTPRI
- BCM_QOS_MAP_OAM_OUTER_VLAN_PCP
- BCM_QOS_MAP_OAM_INNER_VLAN_PCP
- BCM_QOS_MAP_OAM_MPLS_EXP

A new field `counter_offset` has been added in `bcm_qos_map_t` data structure, which is a structure that represents a mapping between packet attributes and internal attributes. It is used by the `bcm_qos_map_add` and the `bcm_qos_map_delete` APIs along with various configuration flags.

```
typedef struct bcm_qos_map_s {  
    ...  
    int counter_offset; /* Offset based on priority for indexing into  
                        the loss measurement counter table */  
} bcm_qos_map_t;
```

SERVICE ACTIVATION TEST (SAT)

The order of parameters in the API `bcm_sat_ctf_bin_limit_get` is updated, which is now represented as below.

bcm_sat_ctf_bin_limit_get

Get ctf bin limit configuration.

Syntax

```
#include <bcm/sat.h>  
int bcm_sat_ctf_bin_limit_get(  
    int unit,  
    int max_bins_count,  
    bcm_sat_ctf_bin_limit_t *bins,  
    int *bins_count);
```

Parameters

<code>unit</code>	(IN) BCM device number
<code>max_bins_count</code>	(IN) Number of elements in array parameter
<code>bins_count</code>	(OUT) Number of bins retrieved
<code>bins</code>	(OUT) Pointer to SAT bin limit structure array

Description

Get ctf bin limit configuration.

Returns

MULTI-DEVICE STACK CONTROL

A new stack API with the corresponding data type are added in this release.

bcm_stk_modid_config_t_init

Initialize a bcm_stk_modid_config_t structure.

Syntax

```
#include <bcm/stack.h>
void bcm_stk_modid_config_t_init(bcm_stk_modid_config_t *pconfig)
```

Parameters

pconfig (OUT) Pointer to bcm_stk_modid_config_t structure to initialize.

Description

Initializes a bcm_stk_modid_config_t structure to default values. This function should be used to initialize any bcm_stk_modid_config_t structure prior to filling it out and passing it to an API function. This ensures that subsequent API releases may add new structure members to the bcm_stk_modid_config_t structure, and bcm_stk_modid_config_t_init will initialize the new members to correct default values.

Table 92: bcm_stk_modid_type_t

<i>bcmStkModidTypePrimary</i>	<i>primary module IDs</i>
bcmStkModidTypeMultiNextHops	multiple next hops module IDs
bcmStkModidTypeCoe	COE module IDs

```
/* Structure for the configuration of the module. */
typedef struct bcm_stk_modid_config_s {
    ...
    bcm_stk_modid_type_t modid_type;
    int num_mods;
} bcm_stk_modid_config_t;
```



Returns

None.

STATISTICS

The BCM_STAT_GROUP_MODE_CAPABILITY_OAM is added for bcm_stat_group_mode_id_create API where in BCM56860, OAM LM is supported through Flex Counters, when BCM_STAT_GROUP_MODE_CAPABILITY_OAM flag is passed along with BCM_STAT_GROUP_MODE_INGRESS or BCM_STAT_GROUP_MODE_EGRESS, it instructs the API to create a stat group suitable for OAM.

New statistics objects are added regard Fixed Flexible Counter to allow configuring flex stats.

```
/* Ingress and Egress Statistics Accounting Objects */
typedef enum bcm_stat_object_e {
    ...
    bcmStatObjectEgrFieldStageEgress = 35, /* Egress FP Object */
    bcmStatObjectEgrMplsTunnelLabel = 36, /* Egress MPLS Tunnel Label */
    bcmStatObjectMaxValue = 37             /* This should be the maximum value */
} bcm_stat_object_t;
```

New packet types are added for bcm_custom_stat_trigger_t that is an enumeration of the packet types available for debug counters.

Table 93: New Custom Counter Packet Types

Name	Purpose
bcmDbgCntNonHGoE	Non HGoE Rx frames on an HGoE port
bcmDbgCntHGoE	HGoE RX frames

New Statistics APIs are added in this release.

bcm_stat_clear_single

Clear specific statistics for a port from the device

Syntax

```
#include <bcm/stat.h>
int bcm_stat_clear_single(int unit, bcm_port_t port,
    bcm_stat_val_t type);
```



Parameters

unit	(IN) Unit number.
port	(IN) Zero-based device or logical port number
type	(IN) SNMP statistics type defined in bcm_stat_val_t

Description

Set all counters to zero for a specific statistics type of the specified port.

Returns

bcm_stat_flex_counter_get

Get the specified statistic of a given counter

Syntax

```
#include <bcm/stat.h>
int
bcm_stat_flex_counter_get(
    int unit,
    uint32 stat_counter_id,
    bcm_stat_flex_stat_t stat,
    uint32 num_entries,
    uint32 *counter_indexes,
    bcm_stat_value_t *counter_values);
```

Parameters

unit	(IN) Unit number.
stat_counter_id (IN)	Stat counter Id
stat	(IN) Counter stat types(packet/byte)
num_entries	(IN) Number of counter Entries
counter_indexes (IN)	Pointer to counter indexes entries
counter_values (OUT)	Pointer to counter values

Description

This API will retrieve a set of counter statistic values for a given counter ID.

```
/* Flex statistics type */
```

```
typedef enum bcm_stat_flex_stat_e {
    bcmStatFlexStatPackets = 0, /* Packets statistics */
    bcmStatFlexStatBytes = 1,   /* Bytes statistics */
    bcmStatFlexStatCount = 2    /* Max value defined for this enum */
} bcm_stat_flex_stat_t;
```

Returns

BCM_E_XXX

bcm_stat_flex_counter_sync_get

Get the synced specified statistic of a given counter

Syntax

```
#include <bcm/stat.h>
int
bcm_stat_flex_counter_sync_get(
    int unit,
    uint32 stat_counter_id,
    bcm_stat_flex_stat_t stat,
    uint32 num_entries,
    uint32 *counter_indexes,
    bcm_stat_value_t *counter_values);
```

Parameters

unit	(IN) Unit number.
stat_counter_id (IN)	Stat counter Id
stat	(IN) Counter stat types (packet/byte)
num_entries	(IN) Number of counter Entries
counter_indexes (IN)	Pointer to counter indexes entries
counter_values (OUT)	Pointer to counter values

Description

This API will retrieve a set of counter statistic values for a given counter ID. Value returned is software accumulated counter synced with the hardware counter.

Returns

BCM_E_XXX

bcm_stat_flex_counter_set

Set counter value of a given counter

Syntax

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

```
int
bcm_stat_flex_counter_set(
    int unit,
    uint32 stat_counter_id,
    bcm_stat_flex_stat_t stat,
    uint32 num_entries,
    uint32 *counter_indexes,
    bcm_stat_value_t *counter_values);
```

Parameters

unit	(IN) Unit number.
stat_counter_id (IN)	Stat counter Id
stat	(IN) Counter stat types (packet/byte)
num_entries	(IN) Number of counter Entries
counter_indexes (IN)	Pointer to counter indexes entries
counter_values (OUT)	Pointer to counter values

Description

This API will set counter statistic values for a given counter ID.

Returns

BCM_E_XXX

bcm_stat_counter_config_set

Configures a counter engine.

Syntax

```
#include <bcm/stat.h>
int bcm_stat_counter_config_set(
    int unit,
    bcm_stat_counter_engine_t *engine,
    bcm_stat_counter_config_t *config)
```

Parameters

engine	(IN) Counter engine ID.
config	(INOUT) Counter configuration.

Description

```
/* Counter engine type */
typedef struct bcm_stat_counter_engine_s {
    uint32 flags;
    int engine_id;
} bcm_stat_counter_engine_t;
```

```

/* A range of counter pointers. */
typedef struct bcm_stat_counter_pointer_range_s {
    int start;
    int end;
} bcm_stat_counter_pointer_range_t;

/*
 * Define the counting source of a counter engine. That is, what block to
 * count, and what to count in a block
 */
typedef enum bcm_stat_counter_source_type_e {
    bcmStatCounterSourceIngressOam = 0,
    bcmStatCounterSourceEgressOam = 1,
    bcmStatCounterSourceIngressVsi = 2,
    bcmStatCounterSourceIngressInlif = 3,
    bcmStatCounterSourceIngressCnm = 4,
    bcmStatCounterSourceIngressField = 5,
    bcmStatCounterSourceEgressField = 6,
    bcmStatCounterSourceIngressLatency = 7,
    bcmStatCounterSourceIngressVoq = 8,
    bcmStatCounterSourceIngressVsqs = 9,
    bcmStatCounterSourceEgressTransmitVsi = 10,
    bcmStatCounterSourceEgressReceiveVsi = 11,
    bcmStatCounterSourceEgressTransmitOutlif = 12,
    bcmStatCounterSourceEgressReceiveOutlif = 13,
    bcmStatCounterSourceEgressTransmitQueue = 14,
    bcmStatCounterSourceEgressTransmitTmPort = 15,
    bcmStatCounterSourceEgressReceiveQueue = 16,
    bcmStatCounterSourceEgressReceiveTmPort = 17,
    bcmStatCounterSourceCount = 18 /* Must be last one. */
} bcm_stat_counter_source_type_t;

/* Counter Engine Source Configuration */
typedef struct bcm_stat_counter_source_s {
    bcm_core_t core_id; /* The counting source core */
    bcm_stat_counter_source_type_t engine_source; /* Counting source */
    bcm_stat_counter_pointer_range_t pointer_range; /* The range of counter
pointers. */
    int command_id; /* Defines what counting command to
read. */
    int num_voqs_per_entry; /* Defines how many consecutive VOQs are
counted in one counter set. Only
relevant for
bcm_stat_counter_engine_source_ingress_voq. */
} bcm_stat_counter_source_t;

/*
 * An entry counter set offset mapping, consists of color X {forward,
 * drop}
 */
typedef struct bcm_stat_counter_set_entry_s {
    bcm_color_t color; /* Counter color (Drop Precedence) */
    uint8 is_forward_not_drop; /* If set, the counter refers to Forward

```

```

packet
        counters. Otherwise, to dropped packet
        counters. */
    } bcm_stat_counter_set_entry_t;

    /* Mapping of an entry to a counter set offset */
    typedef struct bcm_stat_counter_set_entry_mapping_s {
        bcm_stat_counter_set_entry_t entry; /* The entry value, consists of color
X
        {forward, drop} */
        int offset;
    } bcm_stat_counter_set_entry_mapping_t;

#define BCM_STAT_COUNTER_MAPPING_MAX_SIZE 8 /* The maximum legal
size
        of a counter set */

    /* Counter configuration struct */
    typedef struct bcm_stat_counter_set_mapping_s {
        int counter_set_size; /* The size of the counter set, cannot
        an entry mapped to a value above it,
        will not be counted */
        int num_of_mapping_entries; /* The number of valid entries in
        entry_mapping array */
        /* An array of color X {forward, drop} mapped to count an offset in
the counter set */
        bcm_stat_counter_set_entry_mapping_t
entry_mapping[BCM_STAT_COUNTER_MAPPING_MAX_SIZE];
    } bcm_stat_counter_set_mapping_t;

    /* Defines the width of an entry, and what statistics to gather. */
    typedef enum bcm_stat_counter_format_type_e {
        bcmStatCounterFormatPacketsAndBytes = 0,
        bcmStatCounterFormatPackets = 1,
        bcmStatCounterFormatBytes = 2,
        bcmStatCounterFormatDoublePackets = 3, /* doubles that total number of
entries
        for a counter engine */
        bcmStatCounterFormatMaxVoqSize = 4,
        bcmStatCounterFormatIngressLatency = 5,
        bcmStatCounterFormatCount = 6 /* Must be last one. */
    } bcm_stat_counter_format_type_t;

    /* Counter configuration struct */
    typedef struct bcm_stat_counter_format_s {
        bcm_stat_counter_format_type_t format_type; /* Counting format: defines
the width of
        an entry, and what statistics to
        gather. */
        bcm_stat_counter_set_mapping_t counter_set_mapping; /* counter set
configuration: defines
        the size of a counter set, and what
        each offset inside a set counts. */

```

```

} bcm_stat_counter_format_t;

/* Counter configuration struct */
typedef struct bcm_stat_counter_config_s {
    bcm_stat_counter_format_t format;
    bcm_stat_counter_source_t source;
} bcm_stat_counter_config_t;

```

Table 94: Counter Config Flags

Flag	description
BCM_STAT_COUNTER_CLEAR_CONFIGURATION	Disables the counter engine counting clears it's configuration.
BCM_STAT_COUNTER_CACHE	Clears the counter cache.
BCM_STAT_COUNTER_SYMMETRICAL_MULTICORE_ENGINE	If this flag is specified than consecutive counter engines are configured for each core, each statistics gathering is aggregated for symmetrical sources. config->source.core_id must be BCM_CORE_ALL.

Returns

BCM_E_XXX

bcm_stat_counter_config_get

Retrieves a counter engine.

Syntax

```

#include <bcm/stat.h>

int bcm_stat_counter_config_get(
    int unit,
    bcm_stat_counter_engine_t *engine,
    bcm_stat_counter_config_t *config);

```

Parameters

engine (IN) Counter engine ID.
 config (INOUT) Counter configuration.

Description

Get counter engine configuration. See the detail description in bcm_stat_counter_config_set() above.

Returns

BCM_E_XXX

SWITCH CONTROL

A new event BCM_SWITCH_EVENT_RUNT_DETECT, registered for the callback function, is added.



Table 95: Switch Critical Events

Event	Description	arg1	arg2	arg3
BCM_SWITCH_EVE NT_RUNT_DETECT	Alarm was generated from HX4 chips	Port index	N/A	N/A

Below new switch control types are added in this release, while `bcm_switch_oam_stacking_support_t` is introduced for `bcmSwitchOamStackingSupport` of switch control in this release and `bcm_switch_control_key_t` and `bcm_switch_control_info_t` are also added.

```
typedef struct bcm_switch_control_key_s {
    bcm_switch_control_t type;
    int index;
} bcm_switch_control_key_t;

typedef struct bcm_switch_control_info_s {
    int arg;
} bcm_switch_control_info_t;
```

Table 96: `bcm_switch_oam_stacking_support_t`

Value	Description
<code>bcmSwitchOamStackingSupportNone</code>	Do not carry OAM endpoint context in place of VC label in stacking header
<code>bcmSwitchOamStackingSupportOamContextInStackingHeader</code>	Carry OAM endpoint context in place of VC label in stacking header

Table 97: Switch Type Values

Value	Description	Arg Value
<code>bcmSwitchOamStackingSupport</code>	Use VC label field to carry OAM endpoint context to egress XGS switch	Value used is of type <code>bcm_switch_oam_stacking_support_t</code>
<code>bcmSwitchHashOam</code>	Hash select for OAM tables	
<code>bcmSwitchHashOamDual</code>	Secondary hash select for OAM tables	
<code>bcmSwitchHashOamLeastFull</code>	Use least full bank in dual hash OAM tables	
<code>bcmSwitchHashDualMoveDepthOam</code>	Maximum moves in dual hash mode for OAM tables	
<code>bcmSwitchPortVlanTranslationAdvanced</code>	Enable Port Advanced default vlan Edit mode. 1 - enable, 0 - disable	0 / 1
<code>bcmSwitchForwardingLayerMtuFilter</code>	Enable/Disable Forwarding Layer filtering per header format	0 / 1
<code>bcmSwitchLinkLayerMtuFilter</code>	Enable/Disable Link Layer Filtering per header format	0 / 1
<code>bcmSwitchForwardingLayerMtuSize</code>	Mtu size for Forwarding Layer Filtering	
<code>bcmSwitchLinkLayerMtuSize</code>	MTU size for Link Layer Filtering	

Table 97: Switch Type Values

Value	Description	Arg Value
bcmSwitchMimBvidInsertionControl	Control BVID insertion mode. If set to 0, BVID is inserted into the MiM packet if the destination virtual port (bcm_mim_port_t object) is a Network port (legacy behavior). If set to 1, BVID is inserted into the MiM packet if the egress_tunnel_vlan field of the corresponding bcm_mim_port_t object contains a valid vlan id (1 .. 4095)	0/1
bcmSwitchL2CpuDeleteEvent	If enable, when SW initiates a DELETE operation to I2 table, SDK can get relative DELETE events from L2 Mode FIFO.	TRUE/FALSE
bcmSwitchL2LearnEvent	If enable, when HW does a LEARN operation to I2 table, SDK can get relative LEARN events from L2 Mode FIFO.	TRUE/FALSE
bcmSwitchL2CpuAddEvent	If enable, when SW initiates a INSERT operation to I2 table, SDK can get ADD events from L2 Mode FIFO.	TRUE/FALSE
bcmSwitchL2AgingEvent	If enable, when HW does a AGE operation to I2 table, SDK can get AGE events from L2 Mode FIFO.	TRUE/FALSE
bcmSwitchUdfHashEnable	Enable/disable UDF hash	<ul style="list-style-type: none"> BCM_HASH_FIELD0_ENABLE_UDFHASH - Enable/disable UDF hash on hash engine 0 BCM_HASH_FIELD1_ENABLE_UDFHASH - Enable/disable UDF hash on hash engine 1
bcmSwitchOlpMatchRule	Set Match rule value that identifies OLP packets when copied to cpu.	Range of values: 0x1 - 0x100
bcmSwitchMplsExtendedLabelTtl	MPLS Extended label - TTL value	0x00-0xFF
bcmSwitchMacroFlowHashOverlayMinOffset	Minimum offset value to be used in offset table for Overlay flow.	0-99
bcmSwitchMacroFlowHashOverlayMaxOffset	Maximum offset value to be used in offset table for Overlay flow.	0-99
bcmSwitchMacroFlowHashOverlayStrideOffset	Offset increment to be used for populating offset table for Overlay flow.	0-99
bcmSwitchMacroFlowEcmpHashOverlayConcatEnable	Enable hash concatenation for ecmp macro flow for Overlay flow.	TRUE/FALSE
bcmSwitchMacroFlowECMPOverlayHashSeed	Set Hash seed for ecmp macro flow in overlay layer. It doesn't support get, because the seed is scrambled.	Random seed value 0x1-0xFFFFFFFF
bcmSwitchECMPOverlayHashOffset	Hash offset for overlay ECMP. Hash chunks are arranged as 16-bit hash A0, 16-bit hash B0, 4-bit LBN, 16-bit DPORT, 8-bit LBID, 8-bit SW1 LBID, 16-bit hash A1, 16-bit hash B1, 64-bit concatenated hash B1_B0_A1_A0, 4-bit LBN, 16-bit DPORT, 8-bit LBID and 8-bit SW1 LBID. For a given arg, both of the msb and chunk are determined.	0-99, >100 concat mode

Table 97: Switch Type Values

Value	Description	Arg Value
bcmSwitchMacroFlowEcmpUnderlayHashConcatEnable	Enable hash concatenation for ecmp level 2 macro flow.	TRUE/FALSE
bcmSwitchMacroFlowECMPUnderlayHashSeed	Set Hash seed for ecmp level 2 macro flow. It doesn't support get, because the seed is scrambled.	Random seed value 0x1-0xFFFFFFFF
bcmSwitchECMPUnderlayVxlanHashOffset	BCM56960 Enhanced hash select for VxLAN ECMP Level 2 hashing.	0-99, >100 concat mode
bcmSwitchECMPUnderlayL2GreHashOffset	BCM56960 Enhanced hash select for L2Gre ECMP Level 2 hashing.	0-99, >100 concat mode
bcmSwitchECMPUnderlayTrillHashOffset	BCM56960 Enhanced hash select for TRILL ECMP hashing at Level 2.	0-99, >100 concat mode
bcmSwitchVirtualPortUnderlayDynamicHashOffset	BCM56960 Enhanced hash select for Virtual port Aggregation group at Level 2.	0-99, >100 concat mode
bcmSwitchECMPUnderlayHashSet0Offset	BCM56960 enhanced hash bits selection 0 for ECMP Level 2.	0-99, >100 concat mode
bcmSwitchSampleFlexRandomSeed	Set sFlow Flex RNG seed	

Below flags used for bcmSwitchTrunkHashPktHeaderSelect are not longer valid.

- BCM_HASH_HEADER_FORWARD - Forward header as starting header for hashing
- BCM_HASH_HEADER_TERMINATED - Last terminated header as starting header for hashing

New flags for Enhanced hash field config control are added as below.

Table 98: New enhanced hash config

Names	Purpose
BCM_HASH_FIELD_CONFIG_CRC32_ETH_LO	Lower 16-bit of Ethernet CRC32 (BCM56960 only)
BCM_HASH_FIELD_CONFIG_CRC32_ETH_HI	Higher 16-bit of Ethernet CRC32 (BCM56960 only)
BCM_HASH_FIELD_CONFIG_CRC32_KOOPMAN_LO	Lower 16-bit of Koopman CRC32 (BCM56960 only)
BCM_HASH_FIELD_CONFIG_CRC32_KOOPMAN_HI	Higher 16-bit of Koopman CRC32 (BCM56960 only)

A new switch packet information flag regarding bcm_switch_pkt_info_hash_get is added.

Table 99: New BCM Switch packet information flag

Name	Purpose
BCM_SWITCH_PKT_INFO_HASH_OVERLAY	Retrieve ECMP hash result for overlay layer.

A new hash control regarding L2, L3 and Multipath (ECMP/WCMP) is added with bcmSwitchHashControl for use of enhanced ECMP hashing for overlay layer.

- BCM_HASH_CONTROL_ECMP_OVERLAY_ENHANCE

New switch control APIs with index are added in addition to existing switch control APIs.

bcm_switch_control_indexed_set



Configure port-specific and device-wide operating modes. Device wide operating modes are configured on all ports, except the stack ports.

Syntax

```
#include <bcm/switch.h>
int bcm_switch_control_indexed_set(int unit, bcm_switch_key_t key,
bcm_switch_control_info_t info);
```

Parameters

unit	BCM device number
port	Device or logical port number
type	Switch control parameter (see Switch Type Values (page 120))
key	Switch control parameter that suggests the type and index
info	(for "_set") Argument of associated info
arg	(for "_set") Argument whose meaning is dependent on type

Description

These APIs set parameters related to general device operation such as controlling whether certain types of packets should be sent to the local CPU, or setting the priority to be used when such packets are forwarded to the CPU. The port specific operations affect only the indicated port (where possible) while the general APIs affect all ports. Not all operations can be carried out on individual ports on all devices.

Returns

bcm_switch_control_indexed_get

Retrieve port-specific and device-wide operating modes. Device wide operating modes are for all ports, except the stack ports.

Syntax

```
#include <bcm/switch.h>
int bcm_switch_control_indexed_get(int unit, bcm_switch_key_t key,
bcm_switch_control_info_t *info);
```

Parameters

unit	BCM device number
port	Device or logical port number
type	Switch control parameter (see Switch Type Values (page 120))
key	Switch control parameter that suggests the type and index
info	(for "_get", OUT) Argument of associated info
arg	(for "_get", OUT) Value whose meaning is dependent on type

Description

These APIs set parameters related to general device operation such as controlling whether certain types of packets should be sent to the local CPU, or setting the priority to be used when such packets are forwarded to the CPU. The port specific operations affect only the indicated port (where possible) while the general APIs affect all ports. Not all operations can be carried out on individual ports on all devices.

Returns

bcm_switch_lif_property_set/get

Set/Get the LIF property along with the desired property and the configuration values.

Syntax

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

int bcm_switch_lif_property_set(
    int unit,
    bcm_switch_lif_property_t lif_property,
    bcm_switch_lif_property_config_t *lif_config);

int bcm_switch_lif_property_get(
    int unit,
    bcm_switch_lif_property_t lif_property,
    bcm_switch_lif_property_config_t *lif_config);
```

Parameters

unit (IN) Unit number.

lif_property (IN) lif property property: bcmLifPropertyEEDBBankPhase | bcmLifPropertyOutGlobalLifRangeType | bcmLifPropertyEEDBBankExtension.

lif_config (INOUT) lif configuration value : key and value.

```
typedef struct bcm_switch_lif_property_config_s {
    uint32 key; /* property key : 0-47 in case of
                bcmLifPropertyOutGlobalLifRangeType or 0-23
in case of
                bcmLifPropertyEEDBBankPhase and
bcmLifPropertyEEDBBankExtension */
    uint32 value; /* property value */
} bcm_switch_lif_property_config_t;
```

Description

This API is used to Configure/retrieve the LIF property along with the desired property and the configuration values.

Returns

BCM_E_***

TRILL TRILL MANAGEMENT

A New trill port flag is added in this release.

Table 100: TRILL port flags

Name	Purpose
BCM_TRILL_PORT_INGRESS_WIDE	Create Trill port Wide mode

VXLAN MANAGEMENT

A new VxLan tunnel terminator flag is added in this release.

Table 101: New VxLan tunnel terminator flag

Name	Purpose
BCM_TUNNEL_TERM_WIDE	Use to create tunnel terminator wide mode.

A new field aux_data is added in bcm_tunnel_initiator_t data structure, which can be used for configuring tunnel initiator.

```
typedef struct bcm_tunnel_initiator_s {
    ...
    uint32 aux_data; /* Tunnel associated data. */
} bcm_tunnel_initiator_t;
```

A new field is added to `bcm_vxlan_vpn_config_t` structure that can be used for configuring VXLAN vpn.

```
typedef struct bcm_vxlan_vpn_config_s {
    ...
    bcm_gport_t match_port_class;          /* local port vlan domain */
} bcm_vxlan_vpn_config_t;
```

A new criteria is added to VXLAN port match criterions.

Table 102: VXLAN Port Match Criteria

Name	Purpose
BCM_VXLAN_PORT_MATCH_SHARE	Match shared logical port

New VxLan API is added in this release with its data type.

bcm_vxlan_port_traverse

Traverse all valid VXLAN port and call the supplied callback routine.

Syntax

```
#include <bcm/vxlan.h>
int bcm_vxlan_port_traverse(int unit,
    bcm_vxlan_port_traverse_cb cb,
    void *user_data);
```

Parameters

unit	(IN) BCM device number
cb	(IN) User callback function, called once per VXLAN port
user_data	(IN) Cookie

Description

Traverse all valid VXLAN port and call the supplied callback routine to get the VXLAN port information. The callback function is defined as following:

```
typedef int (*bcm_vxlan_port_traverse_cb)(int unit, bcm_vxlan_port_t *info,
void *user_data);
```

Returns

UDF RESOURCES MANAGEMENT

New UDF Data Structures and typs are listed below.

Table 103: New Option to bcm_udf_create

flags	Description
BCM_UDF_CREATE_O_UDFHASH	Hint to the API so the udf allocation is adjusted to be used by the UDF hashing entries.

Table 104: New Flags to control UDF hashing configuration

BCM_UDF_HASH_CONFIG_XXX	Description
BCM_UDF_HASH_CONFIG_ENGINE_0	UDF is used by hash engine 0.
BCM_UDF_HASH_CONFIG_ENGINE_1	UDF is used by hash engine 1.

New UDF APIs are added in this release.

bcm_udf_hash_config_t_init

Initialize the UDF hashing configuration structure

Syntax

```
#include <bcm/udf.h>
void
bcm_udf_hash_config_t_init(
    bcm_udf_hash_config_t *info);
```

Parameters

info (IN/OUT) Pointer to bcm_udf_hash_config_t structure

Description

Initialize the UDF hashing configuration structure

```
typedef struct bcm_udf_hash_config_s {
    uint32 flags; /* Reserved for future usage. Must be
                  zero today. */
    bcm_udf_id_t udf_id; /* UDF object id */
    int mask_length; /* Length of mask bytes. Must equal to
                     the width of the UDF object. */
    uint8 hash_mask[BCM_UDF_HASH_MAX_LENGTH]; /* Array of mask fields. Index
0 is MSB
of mask. */
} bcm_udf_hash_config_t;
```

Returns

BCM_E_XXX

bcm_udf_hash_config_add

Add UDF id into UDF hashing list

Syntax

```
#include <bcm/udf.h>
int
bcm_udf_hash_config_add(
    int unit,
    uint32 options,
    bcm_udf_hash_config_t *config);
```

Parameters

unit	(IN) Unit number.
options	(IN) See BCM_UDF_HASH_ADD_O_XXX
config	(IN) UDF hashing configuration info

Description

Add UDF id into UDF hashing list

Returns

bcm_udf_hash_config_delete

Delete UDF id from UDF hashing list

Syntax

```
#include <bcm/udf.h>
int
bcm_udf_hash_config_delete(
    int unit,
    bcm_udf_hash_config_t *config);
```

Parameters

unit	(IN) Unit number.
config	(IN) UDF hashing configuration info

Description

Delete UDF id from UDF hashing list

Returns

bcm_udf_hash_config_delete_all

Delete all UDF id from UDF hashing list

Syntax

```
#include <bcm/udf.h>
int
bcm_udf_hash_config_delete_all(
    int unit);
```

Parameters

unit (IN) Unit number.

Description

Delete all UDF id from UDF hashing list

Returns

bcm_udf_hash_config_get

Get UDF hashing configuration of a certain id

Syntax

```
#include <bcm/udf.h>
int
bcm_udf_hash_config_get(
    int unit,
    bcm_udf_hash_config_t *config);
```

Parameters

unit	(IN) Unit number.
config	(INOUT) UDF hashing configuration info

Description

Get UDF hashing configuration of a certain id

Returns

bcm_udf_hash_config_get_all

Get all added UDF ids from list

Syntax

```
#include <bcm/udf.h>
int
bcm_udf_hash_config_get_all(
    int unit,
    int max,
    bcm_udf_hash_config_t *udf_hash_config_list,
    int *actual);
```

Parameters

unit	(IN) Unit number.
max	(IN) Max count of UDF ids in the list.
udf_hash_config_list	(OUT) List of UDF hashing configuration
t	
actual	(OUT) Actual count of UDF ids

Description

Get all added UDF ids from list. If max = 0, it only gets the number of UDF ids in list.

Returns

VLAN MANAGEMENT

The new flag used by the VLAN gport API is added below.

Table 105: VLAN gport flags

Flag	Description
BCM_VLAN_GPORT_INGRESS_WIDE	vlan port eligible for additional data
BCM_VLAN_GPORT_VLAN_TRANSLATION	this flag is used for creating an AC in the EEDB that is referenced only by ESEM and no GLEM entries are allocated. must be used with _EGRESS_ONLY

The new flags used by the VLAN gport API are added in below table.

Table 106: New VLAN gport flags

Flag	Description
BCM_VLAN_GPORT_UNKNOWN_UCAST_DO_NOT_UPDATE	Do not update unknown unicast group. It is new defined to replace BCM_VLAN_GPORT_ADD_UNKNOWN_UCAST_DO_NOT_ADD
BCM_VLAN_GPORT_UNKNOWN_MCAST_DO_NOT_UPDATE	Do not update unknown multicast group. It is new defined to replace BCM_VLAN_GPORT_ADD_UNKNOWN_MCAST_DO_NOT_ADD
BCM_VLAN_GPORT_BCAST_DO_NOT_UPDATE	Do not update broadcast group. It is new defined to replace BCM_VLAN_GPORT_ADD_BCAST_DO_NOT_ADD
BCM_VLAN_PORT_INGRESS_WIDE	vlan port eligible for additional data
BCM_VLAN_PORT_VLAN_TRANSLATION	this flag is used for creating an AC in the EEDB that is referenced only by ESEM and no GLEM entries are allocated. must be used with _EGRESS_ONLY

The new key types for egress VLAN translation lookup are defined in bcm_vlan_translate_egress_key_t.

Table 107: bcm_vlan_translate_egress_key_t

VLAN Translate Egress Key Types	Description
bcmVlanTranslateEgressKeyInvalid	Invalid Key Type
bcmVlanTranslateEgressKeyVpn	Use VPN ID
bcmVlanTranslateEgressKeyVpnGport	Use VPN ID and Gport
bcmVlanTranslateEgressKeyVpnGportGroup	Use VPN ID and Gport group
bcmVlanTranslateEgressKeyVlanPort	Use VLAN ID and physical port
bcmVlanTranslateEgressKeyCount	VLAN translation egress key count

A new VLAN control port type is added in this release.

Table 108: New VLAN Control Port Type

Type	Argument Effect
bcmVlanPortTranslateEgressKey	Selects key to use for egress vlan xlate lookup

Three new fields have been added in the `bcm_vlan_control_vlan_t` data structure.

```
typedef struct {
    ...
    int          entropy_id;    /* aging profile ID */
    bcm_vpn_t    vpn;          /* VPN ID */
    bcm_vlan_t    egress_vlan; /* egress outer vlan ID encapsulation
*/
} bcm_vlan_control_vlan_t;
```

New field `class_id` has been added in `bcm_vlan_action_set_t` data structures.

```
typedef struct bcm_vlan_action_set_s {
    ...
    int action_id;                /* Action Set index */
    uint32 class_id;             /* Class ID */
    uint32 flags;                /* BCM_VLAN_ACTION_SET_xxx. */
} bcm_vlan_action_set_t;
```

New VLAN APIs are added in this release.

bcm_vlan_gport_extended_delete

Remove a virtual or physical port from the specified VLAN.

Syntax

```
#include <bcm/vlan.h>
int bcm_vlan_gport_extended_delete(int unit, bcm_vlan_t vlan,
    bcm_gport_t port, int flags);
```

Parameters

<code>unit</code>	BCM device number
<code>vlan</code>	VLAN ID
<code>port</code>	Virtual or physical port to be removed from the VLAN
<code>flags</code>	(IN) Control flags. See New VLAN gport flags (page 131)

Description

Removes the given port to the VLAN. For BCM56524 and BCM56634, the port can be either a WLAN virtual port or a regular physical port. For BCM56840, the port can be a layer 2 logical port or a regular physical port. For physical port, `flags` could be used to decide if the port should be removed from UUC/UMC/BC group.

Returns

Section 4: Test Statistics

HOW TO READ THE DATA

The below tables 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.

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:

Table 109:

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.

NOTE

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.

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's only counted once.

Table 110:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
golden	154	154	154
warmboot	368	318	294
auth	17	17	17
bfd	37	37	37
bhh	43	43	43
chip	9	9	9
cint	60	61	76
coe	543	543	510
cosq	405	371	310
custom	7	7	7
ea	108	108	108
eav	19	19	19
extender	33	33	12
fabric	7	7	7
failover	8	8	8
fcoe	37	37	37
field	886	821	760
higigproxy	129	129	129
infra	114	114	114
ipfix	17	17	17
ipmc	66	63	63
l2	252	247	238
l2gre	13	13	13
l3	331	276	262
l3.alpm.combined	71	67	67
l3.alpm.combined.64	58	55	55
l3.alpm.parallel	71	67	67
l3.alpm.parallel.64	58	55	55
link	26	26	26
mim	19	19	19
mirror	156	150	146
misc	16	16	16
mpls	201	130	128
multicast	23	17	17
niv	40	39	23
oam	187	187	187
pkt	44	44	44
port	298	262	262
proxy	23	23	23
ptp	115	115	115
qos	12	12	12
rate	21	21	21

Table 110:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
rtag7	32	32	32
rx	24	21	21
ser	54	53	52
stack	117	49	49
stat	214	203	203
stg	42	42	42
switch	189	189	189
time	16	16	16
tlvMsg	13	13	13
trill	40	40	40
trunk	180	170	177
tunnel	65	65	65
subport	30	30	33
vlan	217	207	207
vxlان	151	78	78
wlan	17	17	17
Test Suite Total	6503	5982	5761

TEST RESULTS

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

ALL DEVICES

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

Table 111:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
golden	0.6 %	0.9 %	1.1 %
warmboot	0.7 %	0.8 %	1.2 %
bcm.auth	0.5 %	0.4 %	0.4 %
bcm.bfd	0.0 %	0.0 %	0.0 %
bcm.bhh	1.4 %	1.4 %	1.6 %
bcm.chip	0.5 %	1.2 %	1.5 %
bcm.cint	0.0 %	0.0 %	1.3 %
bcm.coe	0.4 %	0.1 %	0.1 %
bcm.cosq	0.4 %	0.7 %	0.8 %
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 %	0.2 %	0.0 %
bcm.fabric	0.0 %	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %	0.2 %
bcm.field	0.7 %	0.9 %	1.7 %
bcm.higigproxy	0.0 %	0.0 %	0.5 %
bcm.infra	0.0 %	0.0 %	0.1 %
bcm.ipfix	0.6 %	0.6 %	0.5 %
bcm.ipmc	0.5 %	0.4 %	0.4 %
bcm.l2	0.7 %	0.8 %	1.0 %
bcm.l2gre	0.0 %	0.0 %	0.3 %
bcm.l3	0.4 %	0.7 %	1.0 %
bcm.l3.alpm.combined	0.0 %	0.8 %	0.0 %
bcm.l3.alpm.combined.64	0.0 %	0.0 %	0.0 %
bcm.l3.alpm.parallel	0.0 %	0.0 %	0.0 %
bcm.l3.alpm.parallel.64	0.0 %	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.9 %	2.0 %	2.2 %
bcm.misc	0.5 %	0.2 %	0.7 %
bcm.mpls	0.6 %	0.4 %	0.4 %
bcm.multicast	0.2 %	0.2 %	0.4 %
bcm.niv	0.0 %	0.3 %	0.3 %
bcm.oam	0.3 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	1.0 %	1.1 %	1.4 %
bcm.proxy	0.4 %	0.4 %	1.1 %
bcm.ptp	0.0 %	0.0 %	0.0 %

Table 111:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
bcm.qos	0.0 %	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %	0.0 %
bcm.rtag7	0.1 %	0.1 %	0.1 %
bcm.rx	0.0 %	0.2 %	0.1 %
bcm.ser	1.5 %	0.3 %	0.3 %
bcm.stack	0.1 %	0.1 %	0.2 %
bcm.stat	0.9 %	0.6 %	0.8 %
bcm.stg	0.2 %	0.2 %	0.5 %
bcm.switch	0.5 %	0.4 %	0.6 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.4 %	0.6 %	0.9 %
bcm.trunk	0.6 %	0.7 %	0.9 %
bcm.tunnel	0.0 %	0.0 %	0.3 %
bcm.subport	0.5 %	0.4 %	0.8 %
bcm.vlan	0.5 %	0.5 %	0.6 %
bcm.vxlan	0.0 %	0.1 %	0.0 %
bcm.wlan	1.0 %	0.8 %	0.8 %
Test Suite Total	0.5 %	0.6 %	0.8 %

TRIDENT2

Table 112:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
golden	0.1 %	0.1 %	0.1 %
warmboot	1.0 %	0.7 %	1.5 %
bcm.auth	0.0 %	0.0 %	0.0 %
bcm.bfd	0.0 %	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	1.3 %
bcm.cosq	0.2 %	1.1 %	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 %	2.2 %
bcm.field	0.4 %	0.5 %	0.6 %
bcm.higigproxy	0.0 %	0.0 %	0.6 %
bcm.infra	0.0 %	0.4 %	0.5 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.5 %
bcm.l2	1.4 %	0.0 %	0.1 %
bcm.l2gre	0.0 %	0.0 %	0.9 %
bcm.l3	0.6 %	0.6 %	1.6 %
bcm.l3.alpm.combined	0.0 %	0.0 %	0.0 %

Table 112:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
bcm.l3.alpm.combined.64	0.0 %	0.0 %	0.0 %
bcm.l3.alpm.parallel	0.0 %	0.0 %	0.0 %
bcm.l3.alpm.parallel.64	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	0.5 %	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %	1.1 %
bcm.niv	0.0 %	0.0 %	1.4 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	0.1 %	0.3 %	0.8 %
bcm.proxy	0.0 %	0.0 %	1.1 %
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	1.9 %	2.6 %	2.4 %
bcm.stack	0.0 %	0.0 %	0.5 %
bcm.stat	0.0 %	0.0 %	1.0 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	0.6 %	0.5 %	0.7 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	2.0 %
bcm.trunk	0.0 %	0.6 %	0.9 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	0.0 %
bcm.vlan	0.0 %	0.0 %	0.3 %
bcm.vxlan	0.1 %	0.2 %	0.4 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.3 %	0.4 %	0.5 %

TRIUMPH3

Table 113:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
golden	0.0 %	0.5 %	1.2 %
warmboot	0.3 %	0.2 %	1.4 %
bcm.auth	0.0 %	0.0 %	0.0 %
bcm.bfd	0.0 %	0.0 %	0.0 %
bcm.bhh	0.9 %	1.1 %	1.4 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	1.3 %
bcm.cosq	0.8 %	1.3 %	0.3 %



Table 113:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
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	0.5 %	0.9 %	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	0.7 %	1.3 %	1.4 %
bcm.higigproxy	0.0 %	0.0 %	0.8 %
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 %	0.5 %	1.8 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.2 %	0.7 %	0.6 %
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 %	1.9 %
bcm.mpls	0.3 %	0.6 %	0.7 %
bcm.multicast	0.0 %	0.0 %	0.0 %
bcm.niv	0.0 %	1.3 %	0.0 %
bcm.oam	0.0 %	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %	0.0 %
bcm.port	1.5 %	2.0 %	2.0 %
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 %	0.9 %
bcm.ser	0.0 %	0.0 %	0.0 %
bcm.stack	0.3 %	0.7 %	0.8 %
bcm.stat	0.3 %	1.3 %	1.1 %
bcm.stg	0.0 %	0.0 %	0.0 %
bcm.switch	1.3 %	0.4 %	0.4 %
bcm.time	0.0 %	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	3.2 %
bcm.trunk	0.6 %	0.6 %	0.6 %
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 %	3.2 %
Test Suite Total	0.5 %	0.6 %	0.6 %

KATANA2

Table 114:

	<i>sdk-6.4.4</i>	<i>sdk-6.4.3</i>	<i>sdk-6.4.2</i>
golden	0.2 %	0.0 %	0.2 %
warmboot	0.6 %	1.0 %	0.6 %
bcm.auth	0.0 %	0.0 %	0.0 %
bcm.bfd	0.0 %	0.0 %	0.0 %
bcm.bhh	0.0 %	0.0 %	1.4 %
bcm.chip	0.0 %	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %	1.3 %
bcm.coe	0.3 %	0.9 %	0.7 %
bcm.cosq	0.4 %	0.2 %	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	0.5 %	0.8 %	0.3 %
bcm.higigproxy	0.0 %	0.0 %	0.2 %
bcm.infra	0.0 %	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %	0.2 %
bcm.l2	0.4 %	0.0 %	0.4 %
bcm.l2gre	0.0 %	0.0 %	0.0 %
bcm.l3	0.3 %	0.8 %	1.1 %
bcm.link	0.0 %	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %	0.0 %
bcm.mirror	0.1 %	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %	0.8 %
bcm.mpls	1.9 %	1.9 %	1.8 %
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	1.8 %	2.4 %	2.4 %
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	3.1 %	1.9 %	1.9 %
bcm.stack	0.0 %	0.0 %	0.0 %
bcm.stat	1.9 %	0.5 %	0.5 %
bcm.stg	0.0 %	0.0 %	0.3 %
bcm.switch	0.0 %	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %	0.0 %

Table 114:

	sdk-6.4.4	sdk-6.4.3	sdk-6.4.2
bcm.tlvMsg	0.0 %	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %	0.0 %
bcm.trunk	0.4 %	0.4 %	0.4 %
bcm.tunnel	0.0 %	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %	1.0 %
bcm.vlan	0.0 %	0.0 %	0.3 %
bcm.vxlan	0.0 %	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %	0.0 %
Test Suite Total	0.6 %	0.6 %	0.6 %

GREYHOUND

Table 115:

	sdk-6.4.4	sdk-6.4.3
golden	0.0 %	0.0 %
warmboot	0.5 %	0.0 %
bcm.auth	0.0 %	0.0 %
bcm.chip	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %
bcm.coe	0.0 %	0.0 %
bcm.cosq	0.0 %	0.0 %
bcm.custom	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %
bcm.extender	0.0 %	0.0 %
bcm.fabric	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %
bcm.field	0.5 %	0.6 %
bcm.higigproxy	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %
bcm.ipmc	0.0 %	0.0 %
bcm.l2	0.4 %	0.0 %
bcm.l2gre	0.0 %	0.0 %
bcm.l3	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %
bcm.mpls	0.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %
bcm.niv	0.0 %	0.0 %
bcm.oam	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %
bcm.port	0.6 %	0.6 %
bcm.proxy	0.0 %	0.0 %

Table 115:

	sdk-6.4.4	sdk-6.4.3
bcm.ptp	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %
bcm.rtag7	0.0 %	0.0 %
bcm.rx	0.0 %	0.0 %
bcm.ser	1.9 %	0.8 %
bcm.stack	0.0 %	0.0 %
bcm.stat	0.0 %	0.0 %
bcm.stg	0.0 %	0.0 %
bcm.switch	0.0 %	0.0 %
bcm.time	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %
bcm.trill	0.0 %	0.0 %
bcm.trunk	0.0 %	0.0 %
bcm.tunnel	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %
bcm.vlan	0.9 %	1.0 %
bcm.vxlan	0.0 %	0.0 %
bcm.wlan	0.0 %	0.0 %
Test Suite Total	0.2 %	0.2 %

TOMAHAWK

Table 116:

	sdk-6.4.4	sdk-6.4.3
golden	0.0 %	2.9 %
bcm.auth	11.8 %	11.8 %
bcm.chip	0.0 %	0.0 %
bcm.cint	0.0 %	0.0 %
bcm.coe	0.0 %	0.0 %
bcm.cosq	0.1 %	1.6 %
bcm.custom	0.0 %	0.0 %
bcm.ea	0.0 %	0.0 %
bcm.eav	0.0 %	0.0 %
bcm.extender	0.0 %	6.1 %
bcm.fabric	0.0 %	0.0 %
bcm.failover	0.0 %	0.0 %
bcm.fcoe	0.0 %	0.0 %
bcm.field	2.5 %	8.7 %
bcm.higigproxy	0.0 %	0.0 %
bcm.infra	0.0 %	0.0 %
bcm.ipfix	0.0 %	0.0 %
bcm.ipmc	1.5 %	1.6 %
bcm.l2	1.2 %	0.0 %
bcm.l2gre	0.0 %	0.0 %
bcm.l3	0.6 %	0.0 %
bcm.l3.alpm.combined	0.0 %	1.5 %

Table 116:

	sdk-6.4.4	sdk-6.4.3
bcm.l3.alpm.combined.64	0.0 %	0.0 %
bcm.l3.alpm.parallel	0.0 %	0.0 %
bcm.l3.alpm.parallel.64	0.0 %	0.0 %
bcm.link	0.0 %	0.0 %
bcm.mim	0.0 %	0.0 %
bcm.mirror	0.0 %	0.0 %
bcm.misc	0.0 %	0.0 %
bcm.mpls	1.0 %	0.0 %
bcm.multicast	0.0 %	0.0 %
bcm.niv	0.0 %	2.6 %
bcm.oam	0.0 %	0.0 %
bcm.pkt	0.0 %	0.0 %
bcm.port	0.7 %	1.3 %
bcm.proxy	8.7 %	17.4 %
bcm.ptp	0.0 %	0.0 %
bcm.qos	0.0 %	0.0 %
bcm.rate	0.0 %	0.0 %
bcm.rtag7	3.1 %	3.1 %
bcm.rx	0.0 %	4.8 %
bcm.ser	1.9 %	7.5 %
bcm.stack	0.0 %	0.0 %
bcm.stat	0.0 %	2.0 %
bcm.stg	0.0 %	0.0 %
bcm.switch	0.8 %	1.6 %
bcm.time	0.0 %	0.0 %
bcm.tlvMsg	0.0 %	0.0 %
bcm.trill	0.0 %	2.5 %
bcm.trunk	0.0 %	2.3 %
bcm.tunnel	0.0 %	0.0 %
bcm.subport	0.0 %	0.0 %
bcm.vlan	0.0 %	1.0 %
bcm.vxlan	0.0 %	1.3 %
bcm.wlan	0.0 %	0.0 %
Test Suite Total	0.9 %	2.2 %

TRIDENT2+

Table 117:

	sdk-6.4.4
golden	0.6 %
bcm.auth	0.0 %
bcm.chip	0.0 %
bcm.cosq	0.7 %
bcm.custom	0.0 %
bcm.eav	0.0 %
bcm.extender	0.0 %
bcm.fabric	0.0 %

Table 117:

sdk-6.4.4	
bcm.failover	0.0 %
bcm.fcoe	0.0 %
bcm.field	2.5 %
bcm.higigproxy	0.0 %@c 2.3 %
bcm.ipfix	0.0 %
bcm.ipmc	0.0 %
bcm.l2	2.4 %
bcm.l2gre	0.0 %
bcm.l3	0.3 %
bcm.link	0.0 %
bcm.mim	0.0 %
bcm.mirror	0.0 %
bcm.misc	0.0 %
bcm.mpls	1.0 %
bcm.multicast	0.0 %
bcm.niv	0.0 %
bcm.oam	0.0 %
bcm.pkt	0.0 %
bcm.port	2.0 %
bcm.proxy	0.0 %
bcm.ptp	0.0 %
bcm.qos	0.0 %
bcm.rate	0.0 %
bcm.rtag7	0.0 %
bcm.rx	0.0 %
bcm.ser	5.6 %
bcm.stack	0.0 %
bcm.stat	0.5 %
bcm.stg	0.0 %
bcm.switch	0.5 %
bcm.time	0.0 %
bcm.trill	0.0 %
bcm.trunk	1.7 %
bcm.tunnel	0.0 %
bcm.subport	0.0 %
bcm.vlan	0.9 %
bcm.vxlan	4.0 %
bcm.wlan	0.0 %
Test Suite Total	1.0 %

SABER2

Table 118:

sdk-6.4.4	
golden	0.4 %
bcm.auth	0.0 %
bcm.chip	0.0 %

Table 118:

sdk-6.4.4	
bcm.coe	2.0 %
bcm.cosq	1.2 %
bcm.custom	0.0 %
bcm.eav	0.0 %
bcm.extender	0.0 %
bcm.fabric	0.0 %
bcm.failover	0.0 %
bcm.fcoe	0.0 %
bcm.field	0.8 %
bcm.higigproxy	0.0 %
bcm.ipfix	0.0 %
bcm.ipmc	0.0 %
bcm.l2	0.4 %
bcm.l2gre	0.0 %
bcm.l3	0.6 %
bcm.link	0.0 %
bcm.mim	0.0 %
bcm.mirror	0.6 %
bcm.misc	0.0 %
bcm.mpls	0.8 %
bcm.multicast	0.0 %
bcm.niv	0.0 %
bcm.oam	1.2 %
bcm.pkt	0.0 %
bcm.port	2.0 %
bcm.proxy	0.0 %
bcm.ptp	0.0 %
bcm.qos	0.0 %
bcm.rate	0.0 %
bcm.rtag7	0.0 %
bcm.rx	0.0 %
bcm.ser	24.1 %
bcm.stack	1.1 %
bcm.stat	0.8 %
bcm.stg	0.0 %
bcm.switch	0.5 %
bcm.time	0.0 %
bcm.trill	0.0 %
bcm.trunk	1.3 %
bcm.tunnel	0.5 %
bcm.subport	1.1 %
bcm.vlan	0.0 %
bcm.vxlan	0.0 %
bcm.wlan	0.0 %
Test Suite Total	1.0 %

STATIC CODE QUALITY ANALYSIS

Continued progress in whittling down static analysis defects per plan.

Table 119:

	<i>Initial Reported Issues</i>	<i>Reported Issues SDK 6.4.1</i>	<i>Reported Issues SDK 6.4.2</i>	<i>Reported Issues SDK 6.4.3</i>	<i>Reported Issues SDK 6.4.4</i>
DNX	664	131	67	11	13
XGS	271	123	54	21	4
SBX	600	0	0	45	1
SerDes	147	74	21	16	18
Common	2827	200	60	40	14
Total	4509	528	202	127	50

Section 5: Resolved Issues for 6.4.4

The following issues are resolved in version 6.4.4 of the SDK.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-38281		56440_A0	bcm_cosq_gport_sched_get API is updated to return appropriate scheduler mode.
SDK-41912	564478	56440_A0	Fixed system reboot caused due to out of bound access to flex counter pool in bcm_port_stat_counter_set
SDK-43491		56640_A0 56440_A0 56850_A0 56440_A1 56440_B0	Fixed cpu queues not honoring weight in WRR mode on TD2/KT/TR3
SDK-43693		88650_A0 88640_A0	CL independent per flow proportional mode support: option to config CL from independent per flow type weight's as higher weight higher priority by set the flag BCM_COSQ_GPORT_SCHEDULER_CLASS_WFQ_MODE_INDEPENDENT_PROPORTIONAL in the gport_add. Note: The definition of the CL independent per flow mode is global for All the CLs and when proportional mode defined inverse mode cannot be used and the opposite
SDK-43716	591808	56440_A0	The stat value for GT16383 and GR 16383 can be read from API bcm_stat_ge_get
SDK-47451 SDK-52856	643625	56440_A0 56445_A0 56445_B0 56634_B0 56636_A0 56636_B0 56638_A0 56638_B0	In previous releases, UDF module support would not work for legacy chipsets which support eight chunks of four bytes each. This release extends UDF module support to work across legacy and latest devices supporting varying number of chunks and chunk size.
SDK-47624	650405	56850_A0	bcm_vxlan_stat_detach() API now detaches both ingress counter and egress counter.
SDK-48334	662397	56440_A0 56440_B0	Corrected retrieval of port and queue info in bcm_multicast_egress_subscriber_get.
SDK-49382	669106	56440_A0 56440_A1 56440_B0 56450_B0	Support has been added for the following commands. The "multicast show" command has been implemented to print the replication queue list for subscriber queue replication on Katana and Katana2. The LLS command has been implemented for Katana devices.
SDK-49460	682997	56850_A0 56850_A1 56850_A2	In earlier releases, we used external clock as default reference clock of TS_PLL and BS_PLL, if the external clock was not existent or was not OK, the system would print error message. Now we use internal clock as default reference clock. This has been resolved.
SDK-52406		56640_B0	In the earlier releases, when using "tcam prbs=1" command, an incorrect mapping to PRBS7 would occur, because Triumph3 does not support prbs=1. In this release, the incorrect mapping was fixed.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-54597	735093	56840_A0 56846_A0 56846_A1	If PFC-transmit was disabled while the XMAC was in the XOFF state, the XOFF state was retained and would send out XOFF frame when the PFC-transmit was re-enabled regardless of traffic. This has been fixed in this release.
SDK-55072	545556	56450_A0 56640_A0 56640_B0 56840_A0 56846_A0 56846_A1 56850_A0 56850_A1 56850_A2	In previous releases, the <code>MIRROR_CONTROL.M_ENABLE</code> field would not get set when a port initially configured as an XE port later changed to HiGig encapsulation. This has been fixed by setting <code>MIRROR_CONTROL.M_ENABLE</code> field to 1 during the transition.
SDK-55410	756377	All	The priority scoring mechanism for UDF packet type offsets has been documented. When the same type of UDF packet format is added using API <code>bcm_field_data_qualifier_ethertype_add</code> , it takes lower priority than when same type of packet format is added using API <code>bcm_field_data_qualifier_packet_format_add</code> .
SDK-55513	758422	56850_A0 All	In earlier releases, the SDK would not check if user mixed line-rate and oversubscription ports existed together within a TSC on Trident2. This has been resolved and the SDK can check in both flex port mode and non-flex port mode.
SDK-55675		56640_A0 56640_B0 All	BFD support over LAG : To run BFD session on LAG interfaces, application has to configure an independent asynchronous mode BFD session on every LAG member link in a LAG bundle. Asynchronous mode BFD sessions on every LAG member link is termed as "micro-BFD session". Dedicated MAC "01-00-5E-90-00-01" has to be used as next hop destination MAC in accordance with Standard RFC 7130.
SDK-55835	756688	88670_A0	In L2, the age can be now retrieved via a new field <code>age_state</code> to struct <code>bcm_l2_addr_t</code> . The Hit indication can be retrieved from the L2 flags (<code>BCM_L2_HIT</code> flag).
SDK-55917	764186	56440_A0 56440_A1 56440_B0	In earlier releases, The <code>L3_DEFIP</code> table was missed in S/W TCAM SER protection list, so the SER engine did not detect and correct corrupted entries in <code>L3_DEFIP</code> table. All the TCAM tables on KT have been added into S/W TCAM SER protection list in this release to address this issue

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-56197		88650_A0 88650_B0 88660_A0	<p>OAM: For BHH OAM endpoints over PWE, MPLS, the "Passive" side entry on which TX counters associated with the MEP are maintained can be extracted from the field <code>mpls_out_gport</code> in <code>bcm_oam_endpoint_create()</code> (by default this field is <code>BCM_GPORT_INVALID</code> in which case the LIFs are assumed to be symmetric). Passive side OAM traps also exist on this LIF at the egress. The outLIF can be set in the following way: 1. In case of MPLS tunnel, the actual outLIF can be taken from <code>label_array[*].tunnel_id</code> which will be returned from the API <code>bcm_mpls_tunnel_initiator_create()</code>. In case of VPLS, the outLif can be taken from the <code>encap_id</code> field of <code>bcm_mpls_port_add</code> api. 2.For BHH MPLS endpoints the outLIF should then be set as a tunnel gport using the macro <code>BCM_GPORT_TUNNEL_ID_SET()</code>. For PWE endpoints the outLIF should be set as a <code>MPLS_PORT_gport</code>. 3.The gport-subtype should be set. For example using the cint <code>mpls_tunnel_id_to_mpls_tunnel_gport()</code>.</p> <p>An example of this can be found in <code>oam_run_with_defaults_mpls_tp()</code>.</p>
SDK-56373	765537	56340_A0 56640_A0	On non-cache-coherent platform, DMA buffer needs to be invalidated before CPU access. However, this operation for all Triumph3 devices was missed in previous releases. This issue has been fixed in this release.
SDK-56690	773593	56850_A0	In previous releases, when enabled LLFC would conflict with PAUSE. This has been addressed by disabling PAUSE when LLFC is enabled, and enabling PAUSE when LLFC is disabled.
SDK-57009		56850_A0 56850_A1 56850_A2	In previous releases, <code>bcm_vxlan_stat_detach</code> took high execution time because redundant memory operation was executed. In this release, we remove memory read operation and use <code>soc_mem_write</code> instead of <code>soc_mem_write_range</code> conditionally to save time, then the execution time can be reduced a lot.
SDK-57125	779514	84848_A0	Support has been added for BCM84848 1G AutogrEEE with clear statistics feature.
SDK-57476		56850_A0 56850_A1 56850_A2	<p>In earlier releases <code>bcm_stat_group_create</code> could get stuck in loop for egress SVP counters under scaled set-up. The issue was due to macro <code>FLEX_COUNTER_DEFAULT_EGR_DVP_ATTRIBUTE_1_TABLE_POOL_NUMBER</code> not being defined correctly for TD2, which led to endless loop when the egress flex counter pool were exhausted. It was defined to 5 for all the chips include TD2 but actually it should be less than 4 for TD2 as TD2 only has 4 egress flex counter pools.</p> <p>The fix was to define separate macro for TD2.</p>

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-57494	783183	56850_A0 56850_A1 56850_A2	In previous releases, the performance of the API <code>bcm_l3_route_delete_all</code> did not match customers' requirements. In this release, it has been improved by 2.4 times for both ipv6 or ipv4 routes. This is achieved by leveraging DMA for bulk operations, skipping delete propagation and storing the index of each route entry.
SDK-57822	788335	56850_A0 56850_A1	Support has been added for a new API to traverse all valid VXLAN port and a callback routine to retrieve the vxlan port information. The callback function is defined as follows: <code>int (*bcm_vxlan_vpn_traverse_cb)(int unit, bcm_vxlan_vpn_config_t *info, void *user_data)</code>
SDK-58066	790849	56850_A0 56850_A1 56850_A2	In previous releases, a link down event on a HG[40] port with CL72 enabled triggered Eatabuf overflow. This issue has been fixed in this release.
SDK-58557	789635	56624_A0 56624_B1	For L3_ENTRY_ONLY and MPLS_ENTRY tables, parity errors on index subject to $((\text{entry index} \& 3) \neq 0)$ couldn't be recovered by the SDK. This was because the entry index of the dual hash table was not correct if the index & 3 was not previously 0. The code has been changed to put a correct entry index for <code>soc_ser_correction</code> .
SDK-58654		56960_A0	1. RTAG7_MPLS_L2_PAYLOAD_L3_HASH_FIELD_BMAP register is Rtag7 Hashing Field Selection Bitmap Register for MPLS terminated tunnels with L2 payloads. MPLS_L2_PAYLOAD_L3_BITMAP_A/B Hash field selection for MPLS terminated pkts with L2 payloads in RTAG7_MPLS_L2_PAYLOAD_L3_HASH_FIELD_BMAP register is now configurable in Tomahawk. 2. In addition RTAG7_HASH_CONTROL_2 registers MPLS_PAYLOAD_HASH_SELECT_A/B configuration is supported which selects inner L2 (0) or inner L3 (1) fields to drive to Hash buckets when MPLS packet is received and the tunnel is being terminated.
SDK-59769	804528	88650_A0 88650_B0 88650_B1 88660_A0	<code>arad_pp_frwrddfec_ecmp_hashing_global_info_get_unsafe()</code> will now return the <code>use_port_id</code> flag
SDK-59979 SDK-68669	811045	All	In earlier releases, parallel warpcore firmware loading was not supported. The feature was added in this release.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-60222 SDK-60187		88650_A0 88650_B0 88660_A0	added memory array caching: Some memories have multiple instances within a single block in the device. Internally, the SDK organizes such tables into arrays, so that each instance of a table can be accessed using an instance-id. The caching mechanism did not support caching of memory-arrays. Customer impact: Normally, memories can be cached to DRAM as part of soft-error (SER) protection mechanism, or in order to improve driver runtime performance. For memory arrays, caching was not available. For parity-protected configuration memories, caching is used to recover from a soft-error. As a result of this addition, memory arrays can now also be protected by this mechanism. list of relevant memories: EPNI_EEDB_BANK EPNI_DATA_FORMAT EPNI_LFEM_FIELD_SELECT_MAP IDR_MCDA_PRFSSEL IDR_MCDB_PRFSSEL IHB_FEM_BIT_SELECT IHB_FEM_MAP_INDEX_TABLE IHB_FEC_ENTRY IHB_TCAM_ACTION IHP_PARSER_PROGRAM IHP_VLAN_PORT_MEMBERSHIP_TABLE IHP_LIF_TABLE note for 6.3 branch: the list of cached tables is not filled by the fix in this JIRA, and will be filled soon.
SDK-60417	817061	56240_B0	As per the data sheet, Saber supports 2K Egress Flex Counter Table size, where as the support in SDK was only for 1K table size even though the device supported the same. Fix has been added to support 2k Egress Flex Counter Table size.
SDK-60443		88650_A0 88650_B0 88650_B1 88660_A0	VLAN Port: The API <code>bcm_vlan_port_find()</code> returns information regarding a VLAN-Port that was either supplied by a gport in the <code>vlan_port_id</code> field or by entering the fields that are essential for the match: <code>criteria</code> , <code>port</code> , <code>match_vlan</code> etc. The API, in the mode where <code>vlan_port_id</code> was supplied, was failing non-local gports for the device. After the fix, the API accepts non-local gports and returns all the locally available information. Some field values that aren't available on a remote device, such as protection fields, won't reflect the values in the device where the gport resides.
SDK-60978	811047	88650_B1	In previous releases, an rx reset or full reset with traffic running in ingress could cause persistent PFC XOFF and result in a traffic outage. This was due to the reset causing the pause mechanism not to configure properly and incorrect pause frames being transmitted. This has been solved by correcting the soft init sequence.
SDK-61030	822480	All	<code>soc_shutdown</code> now uses spinlocks for working in multi processor environment, and <code>dpc_disable</code> is called after disabling all the interrupts. This prevents intermittent crashes in the DPC thread during <code>soc_shutdown</code> .
SDK-61166	823916	56640_A0 56640_A1 56640_B0	12 lanes of 100G are used to check the prbs status including 2 unused lane which always returned failed even though the 10 active lanes prbs status were OK. Added the fix to use 10 active lanes for prbs status checking for 100G

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-61249		88650_A0 88660_A0 88670_A0	In TCAM memory, a mechanism for identifying and fixing SER errors in the TCAM exists, which handles an error upon SER occurrence, identifies the location of the error and according to TCAM bank owner, reproduces the TCAM entry from software database and rewrites it to TCAM memory. This mechanism is not supported for TCAM banks that are owned by VT or TT blocks. This is fixed.
SDK-61576		88660_A0	BFD: Allow BFD endpoints of type IP over MPLS by default for BCM88660.
SDK-61860	831533	88650_A0	Fixed a bug of access to non existing register fields in ARAD_A0.
SDK-61897	823047	56850_A0 56850_A1 56850_A2	Support has been added to allow the configuration of 20G oversubscription ports in a valid flex port application.
SDK-61902		56334_B0 56334_A0	In previous releases, linkscan would not enable the MAC which would result in traffic being blocked on the port. Logic has been added to check whether the MAC is enabled in the port update routine. If the MAC is disabled, the port update routine will enable the MAC.
SDK-62156	797863	56750_A0 56750_A2 56755_A0	In the previous release, the API <code>bcm_multicast_init</code> didn't clear L2MC table and L2 addr table. In this release, this issue has been addressed by clearing them in the API.
SDK-62199	827548	56634_B0	In previous releases, the protection group failed to work when the failover id was 0x18000001. This was because in the mpls failover vp group check, vp group 1 was not included as valid failover vp. This has been corrected by including vp 1 as valid failover group.
SDK-62229	825199	56640_B0	For Triumph3 EXTERNAL STAGE, L3Routable will be determined by considering MY_STATION_HIT, FORWARDING_TYPE & L3_IIF_VALID bits. To qualify a packet on L3Routable add qualifiers MyStationHit, ForwardingType & L3Routable to the group and configure an entry with the following values: L3Routable data=0x1 mask=0x1 ForwardingType data=0x2 (bcmFieldForwardingTypeL3) MyStationHit data=0x1 mask=0x1
SDK-62285	822442	56850_A0 56850_A1 56850_A2 All	In previous releases, the return values of <code>_bcm_vp_used_get</code> were not completely checked. This has been fixed by checking all the return values of <code>_bcm_vp_used_get</code> .
SDK-62414		88650_A0	<code>bcm_l3_egress_traverse()</code> will now fill the <code>bcm_l3_egress_t</code> passed to the callback correctly
SDK-62465		88670_A0	L3: Added support to carry EEI information of service label from FEC using <code>bcm_l3_egress_create</code> api with egress label properties.
SDK-62743	834109	56850_A0 56850_A1 56850_A2 All	In earlier releases, UDF hashing was not supported. This will now be supported by adding a set of APIs to enable/disable UDF hashing and configure UDF hashing properties.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-62751	825343	56440_A1 56440_B0 All	On KATANA devices, L2 MAC address learning would not work after Warmboot. This occurred because L0, L1 and L2 scheduler bit maps of MMU CosQ were not retrieved during Warmboot. Code has been added to retrieve L0, L1 and L2 scheduler bitmaps during Warmboot.
SDK-62757	838601	54282_A0 54282_C0 54282_C1 54382_B0 54382_C0 54382_C1	In previous releases, the API <code>bcm_port_phy_control_get</code> would return the wrong value for <code>BCM_PORT_PHY_CONTROL_EEE_AUTO_FIXED_LATENCY</code> . This has been corrected with the parameters <code>TRUE(1)</code> for fixed latency and <code>FALSE(0)</code> for variable latency.
SDK-62784	838583	56450_A0	Problem Statement: In earlier release, all L0 nodes were not usable, even when Linkphy was not enabled. In this release <code>s1_scheduler</code> and <code>s0_scheduler</code> are set to 0 when linkphy is not enabled, so that all L0 can be used.
SDK-62866	813901	56640_A0 56640_A1 56640_B0	When an IPFIX interrupt was generated, it was observed the interrupt was received after the disabling the interrupt. The overflow and timeout were not being checked checked and disabled if enabled. Logic has been added to check and disable the overflow and timeout if enabled before processing the IPFIX interrupt.
SDK-63079	837938	56960_A0	Added MBIST support (TR 147) for Tomahawk SDK.
SDK-63101	835936	56640_A0	After Warmboot, the entry 0 of table <code>ING_L3_NEXT_HOP_ATTRIBUTE_1</code> wouldn't successfully recover when parity error injected. This was caused by the valid bit map associated with this entry being cleared during Warmboot. This issue has been fixed and now the valid bit map of a entry in the software cache is not cleared when Warmboot is in progress.
SDK-63103	839828	56640_B0	The <code>MPLS_ENTRY</code> table did not recover after Warmboot. This would occur because the <code>MPLS_ENTRY</code> and <code>MPLS_ENTRY_EXTD</code> overlapped. In this release, the <code>MPLS_ENTRY</code> and <code>MPLS_ENTRY_EXTD</code> have been added to the overlapped memory list to solve the problem.
SDK-63107	841019	56440_A0 56440_A1 56440_B0	Removed Redundant code segment in function <code>_phy_xgxs16g_control_tx_driver_set()</code> .
SDK-63112	827173	All	Very long CINT scripts would exhaust memory resources. This was due to an internal function that declares and initializes variables would not release local objects immediately after their use. Though all objects were released at the end of the loop, during loop iterations there was increasing demand of RSS pages for the running process. A fix has been made to release these local objects as soon as their use is complete.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-63134		88660_A0	<p>Added support for string based MAID with format: 0x04 <MD Len> <5 bytes MD> 0x02 <MA Len> <6 bytes MA></p> <p>To use this format, set soc property <code>oam_maid_11_bytes_enable=1</code> and call <code>field_oam_advanced_features()</code> cint function.</p> <p>An example for this can be found in <code>oam_example</code> function in <code>cint_oam.c</code></p>
SDK-63143		88650_B1 88660_A0	<p>In earlier releases an error happened when the user get/set RIF in High VSI mode for some special services, which was caused by using an incorrect register in programming sequence . This has been resolved by modifying the register mapping between VSI and RIF, such that the user can set/get RIF correctly in High VSI mode.</p>
SDK-63395	843778	56340_A0	<p>An invalid peak temperature value was printed when "show temp" was performed. This issue was due to an incorrect initialization sequence for the temperature monitor. This has been corrected by rearranging the initialization sequence.</p>
SDK-63433		88650_A0 88660_A0	<p>In large exact match table, HW accelerated traverse operations were not functional when EM shadow is enabled. This is fixed.</p>
SDK-63539		56440_B0	<p>In the previous release the linkscan mode variables were incorrectly documented as <code>BCM_PORT_LINKSCAN_SW</code> <code>BCM_PORT_LINKSCAN_HW</code> <code>BCM_PORT_LINKSCAN_NONE</code> The documentation has now been corrected with the following variables respectively. <code>BCM_LINKSCAN_MODE_SW</code> <code>BCM_LINKSCAN_MODE_HW</code> <code>BCM_LINKSCAN_MODE_NONE</code></p>
SDK-63552	811881	88650_A0 88660_A0	<p>Allow ingress discard per In-LIF. The discard enable/disable is available via <code>bcm_port_discard_set()</code> for the specified In-LIF and the desired mode value. The mode value can now support <code>BCM_PORT_DISCARD_INGRESS</code>, while <code>BCM_PORT_DISCARD_NONE</code> removes the discard. Note: The available mode values effect both the In-LIF and the Out-LIF whenever a new mode value is set. <code>bcm_port_discard_get()</code> now supports the modified values for the mode field.</p>
SDK-63560		88670_A0	<p>OAM: New 88670 OAM diagnostics: 1) diag oam id: shows the OAM -ID of the last recieved OAM packet. 2) diag oam count oamp: Shows the number of recieved/transmitted packets by the OAMP. 3) diag oam prge: shows all loaded OAM PRGE programs. 4)diag oam prge last=1 shows the last run OAM PRGE program.</p>
SDK-63633	843722	56850_A2 56850_A0 56850_A1	<p>In earlier release, default setting of <code>bcmSwitchEntropyHashSet0Offset</code> was concatenated mode, and there was no way to modify it. This has been fixed by making default as non-concatenated mode, and letting <code>bcmSwitchEntropyHashSet0Offset</code> modify it.</p>

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-63663		54680_A0 54680E_B0	Traffic was not looped back during PHY Loopback. This has been corrected and now MII Control Register (internal loopback) is asserted for any device when PHY loopback is set.
SDK-63734	808883	All	Traverse with mask was not returning all the required entries. The traverse code was treating filtered out entries as if they are entered to the result buffer.
SDK-63755	832108	88660_A0	Added support for IPv6 SAV and IPv6 address compression. The result of IPv6 address compression can be qualified by PMF entries. The following two soc properties are introduced to enable/disable above function: 1.custom_feature_l3_source_bind 2.custom_feature_ip6_compression_enable
SDK-63853		56850_A2	In previous releases, bcm_switch_hash_entry_remove/ bcm_switch_hash_entry_destroy behaved the same. They both removed entry from the hardware tables and destroyed flex hash entry. The behaviors have been corrected according to API reference guide. bcm_switch_hash_entry_remove will only delete hardware table entry. Software table entry will be deleted by bcm_switch_hash_entry_destroy.
SDK-63891	847002	88650_B1 88660_A0	In Parser, some Parsing Macros were configured to mark packets with PLC/PFC 0x00 upon stopping following a parsing error, causing issues in subsequent blocks. Macros are now configured to fail-back to the calling Macro PLC/PFC values in case of a parsing error.
SDK-63900	846993	88650_B1 88660_A0	IPMC RP: Fixed flush mechanism for rendezvous point(rp) in ipmc-bidir database. Now 'bcm_ipmc_rp_get' can distinguish between different rp and return only configured interfaces according rp_id.
SDK-63928	845982	56640_A0 56640_A1 56640_B0	In earlier releases the MMU_WRED_CONFIG entry index was not right, which then caused ser to not correct the expected entry. Code has been changed to correct the entry index, ser can use cache restore to correct the MMU_WRED_CONFIG entry.
SDK-63979	846217	56840_A0	In previous releases, the UDF Group creation was order dependent. In this release the order of chunk allocation has been changed. When more than 4 chunks are required for a qualifier, UDF2 is given priority over UDF1.
SDK-63984	845383	56450_B1 56450_A0 56450_B0	On BCM5645x devices, support has been added for extended CLI port command with option 'detail' to print software and hardware port configuration in addition to basic port status.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-64003	848452	56640_A0 56640_A1 56640_B0 56643_A1	For Triumph3 devices, ESM was not working for IPv6 ACL Short and IPv6 ACL Full modes. To address this problem, the following issues have been fixed: 1. Number of searches required for IP6S_ONLY partition in LTR was programmed as 2 but should be 1 since only ACL partition needs to be searched. 2. Field mapper to key id entries for IP6SONLY & IP6FONLY tables were incorrectly programmed for number of Key responses expected. Only ACL response (KEY_GEN_RSP_TYPE_ACL) was expected. 3. 480bit partitions in SW were mapped to 640bit partition in external TCAM. To support 480bit, upper two 80bit portions of all 640bit entries in TCAM need to have a valid bit set along with data and mask set to 0. Valid bit was not set causing 640bit search to fail. All of these issues have been addressed in this release.
SDK-64043	849712	56540_A0 56545_A0 56640_A0	On a 56640, the DO_NOT_LEARN_MACSA bit would not be set when the bcm_l2_cache_set() API was called with the BCM_L2_CACHE_LEARN_DISABLE flag. Support has been added to handle the BCM_L2_CACHE_LEARN_DISABLE flag for the bcm_l2_cache_set() API.
SDK-64142 SDK-64914	849803	56450_A0 56450_B0	In previous releases, retrieving the index of the child node was based on the assumption that the child indexes were allocated contiguously. For nodes with WRR scheduling mode, this was not true. The SDK has been corrected to retrieve the index directly instead of the present base + offset method.
SDK-64144	849036	56640_A0	In earlier releases, when there was only ext_l2_fwd_table configured for triumph3, if external tcams were used to learn L2 address, the learning messages would be sent to L2_MOD_FIFO repeatedly. In this release, the same L2 address is reported to L2_MOD_FIFO only once if it is already in the external tcam.
SDK-64182	850934	56450_A0 56450_B1 56450_B0	In earlier releases, the xgxs16g1l driver did not support Hardware LOS solution. In this release, the SOC property "serdes_rx_los" has been implemented in Katana2. This will enable "signal_detect_en" and "invert_signal_detect" in register 0x8300.
SDK-64196	851179	56450_A0 56450_B0	The ING_L3_NEXT_HOP table extended queue info could not be cleared when extended queue was created in explicit extended queue mode. This issue has been corrected. In an mpls configuration ING_L3_NEXT_HOP, qtag and tag_type will change to 0 in ing_l3_next_hop memory when per_flow_queue_base is 0.
SDK-64222		56340_A0 56640_B0	Upon VCO change on warpcore block, all member ports of particular warpcore block should be hard reset. Earlier ports were soft reset. Changes are made to hard reset the warpcore member ports in this release.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-64270		88650_A0 88650_B0	memory error initiation mechanism fix: memory protection mechanism disabled for the following memories: IRR_SNOOP_MIRROR_DEST_TABLE IRR_SNOOP_MIRROR_TABLE_0 IRR_SNOOP_MIRROR_TABLE_1 IRE_NIF_PORT_TO_CTXT_BIT_MAP These tables are not protected by parity error detection mechanism by HW. Upon initiating error for these tables, incorrect memory error indication passed to the bcm driver and wrong memory entry correction take place. Therefore, BCM SDK disables the parity protection mechanism for them. Note, This fix applies only for memory error initiation mechanism. These memories are not causing parity error event.
SDK-64290	843249	56634_A0 56634_B0	In 56634 the "l3 l3table add vrf=0 ip=172.16.95.9 mac=00:04:DF:13:6E:99 Trunk=32" would process into a check that TRUNK num was restricted under 32 , so this command failed ; Now TRUNK num will not enter this inappropriate check and "l3 l3table add vrf=0 ip=172.16.95.9 mac=00:04:DF:13:6E:99 Trunk=32" is OK .
SDK-64335	852089	56524_A0	In previous releases, the <code>bcm_vlan_stat_get32()</code> API would return an incorrect value in a multi-unit system. In this release the global variable <code>_bcm_flex_stat_buff_toggle</code> has been converted to unit based in order to support multi-chip systems.
SDK-64352	816473	56450_A0 56450_B0	On Katana2, traffic would not function properly on ports at 100Mbps. This was due to the ports not being able to operate in independent channel mode while in asymmetric speed mode. This has been fixed by enabling Lane mode for ports in asymmetric speed mode at speeds less than or equal to 1G.
SDK-64388	851969	56850_A0	In a warpcore when the encapsulation of an xe port changed to h1gig, it caused the other three ports to change to 11G. This resulted in a reset to the core and link-down for the other three ports. This issue has been fixed. Now the config variable "port_max_speed" is used to overwrite the default HG speed. In order to reset the core, set the "port_max_speed" as 10G explicitly in the config.bcm. To switch HG at 11G, set 11G explicitly using <code>port_speed_set</code> API/command or remove "port_max_speed" from config.bcm.
SDK-64391		56850_A0 56850_A1 56850_A2	In previous releases, the element numbers of the local array <code>mask_entries</code> changed from 128 to 256 resulting in a stack overflow in TD2. This issue has been fixed in this release via allocating heap memory instead of stack memory for <code>mask_entries</code> .

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-64426		56450_B0	<p>When <code>single_wide</code> was used the reference count at index would be incremented. When <code>double_wide</code>, the reference count at index and index+1 was incremented. The presence of <code>ecmp</code> group was checked only at index and updating reference count at index.</p> <p>It is incorrect to update the count only at index when Katana2 is in <code>double_wide</code> mode. Index and Index+1 should be updated similar to <code>_bcm_xgs3_ecmp_group_add/</code> <code>_bcm_xgs3_ecmp_group_del</code>.</p> <p>In this release <code>BCM_XGS3_L3_MAX_ECMP_MODE(unit) ?</code> <code>_BCM_SINGLE_WIDE : BCM_DOUBLE_WIDE</code> have been added for updating reference count in functions wherever <code>single_wide</code> is being used and the flag is <code>BCM_L3_MULTIPATH (ecmp)</code>.</p>
SDK-64471	847082	56540_B0	In previous releases, the SDK would create two sets of DiffServ queues in the process of attaching a service queue. This has been fixed so that ports do not reuse indexes.
SDK-64501	851696	56640_B0	On TRUMPH3, when adding an L2 entry with group size greater than 63 an error was returned even though the supported number of groups was 1024. This was caused by the L2 table entry group incorrectly using 63(6 bits) instead of 1024(10bits). This has been corrected to use a size of 10bits for the group field of the L2 table entry.
SDK-64519		88660_A0	OAM: In case EVC filtering is enabled outgoing DM packets are corrupted.
SDK-64562	853905	56960_A0	<p>This is a query on whether SDK 6.4.3 supports preselection.</p> <p>SDK 6.4.3 will have only legacy FP support. It doesn't support multiple keygen profile for a logical table. This support is coming in phase-2.</p>
SDK-64650	854677	56450_B1 56450_A0 56450_B0	For 5645x devices, the ports that were not linked up before Warmboot, were not able to transmit post Warmboot even if the ports were later linked up. The root cause for the issue was support didn't exist to update the linkscan bitmaps for the ports that were linked up post Warmboot. Support for Katana2 linkscan pbmp and bmap update has been added.
SDK-64652		56340_A0 56547_A0 56548_A0	On an EHG port , the <code>ICONTROL_OPCODE_BITMAP</code> table entry was not set with the CPU port. This would cause ehg-higig2 control-opcode packets received on this port to not reach the CPU. This has been addressed so that the <code>ICONTROL_OPCODE_BITMAP</code> table entry is set with the CPU port when EHG is enabled on a port.
SDK-64660	854712	56850_A0	In the previous release, the API <code>bcm_xgs3_tunnel_terminator_delete</code> would incorrectly delete I3 ingress interfaces. In this release, this has been addressed by adding a check condition before deleting them.
SDK-64672	850967	All	Fixed software issue causing oversubscription buffer counters api to return <code>BCM_E_UNAVAIL</code> on TD2.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-64756	855331	56450_A0 56450_B0	In BCM5645x devices, disabled Auxiliary ports would affect the link status of the controlling port after performing a flex-IO operation. This was due to auxiliary ports not being properly reset before detaching them during a hot-swap. This has been fixed by enabling ports before detaching in order to ensure a proper reset.
SDK-64760		88660_A0	BFD: Support BFD over IPv6 classification.
SDK-64767	853890	56850_A2	In earlier releases, an invalid L0 node parent index value was chosen based on PIPE0 ports' configuration only. This has been fixed by also considering ports in PIPE1 when choosing an invalid parent index value.
SDK-64786		56450_B1 56450_A0 56450_B0	In previous releases, mirroring in a stack system was disabled for Katana2. In this release, the support has been added for Katana2.
SDK-64832		56850_A0 56850_A1 56850_A2	In the previous release, the API <code>bcm_esw_multicast_extender_encap_get</code> did not check whether the virtual port was used. This issue has been addressed by adding the relevant check condition.
SDK-64900	855758	88660_A0	DRCx_OPPDATAERRINT interrupts initialized for ARAD_PLUS device. Before this fix, access to uninitialized database cause a system crash upon event occurrence at ARAD_PLUS. This fix eliminates the crash and corrects the corrective action for this event.
SDK-64905	856320	56640_A0 56640_A1 56640_B0 56643_A0 56643_A1 56643_B0	A segmentation fault would be observed when the command <code>""phy control hg0 rxtune""</code> was executed. The crash was due to processing arguments of <code>rxtune</code> as <code>NULL</code> . This has been corrected by validating the arguments before processing to avoid the segmentation fault.
SDK-64911	856014	56638_A0	The API <code>bcm_oam_endpoint_get()</code> may return <code>BCM_E_PORT</code> when retrieving the OAM endpoint information. This has been fixed by correcting the port conversion in the <code>bcm_oam_endpoint_get</code> function for Triumph2 devices.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-64932	88650_A0 88660_A0		<p>Additional support is added to MTU Filtering. In Arad devices MTU filtering is performed twice per packet, the first in according to Forwarded-packet-size (hereinafter Forwarding Layer MTU Filter) and the second in according to Packet-Size on the TX wire (hereinafter Link Layer MTU Filter). Each of these filters compares the relevant packet size to one of four MTUs according to an MTU profile. For profile 0, the MTU is taken from the port configuration, and for profiles 1-3, the MTU is taken from the ForwardingLink Layer configured MTUs. The MTU profile is retrieved from a table that maps VSI to profile. However, the hardware presents a limitation, as in the Forwarding Layer the index to the table is the VRF and in the Link Layer the index to the same table is the VSI. The Forwarding Layer MTU filter is not supported in the driver. In addition, each of the filters can be enabled/disabled according to Forwarding Type, which is not supported either. Support in both these features is added with several constraints, that derive from the hardware limitation. The new abilities and constraints are as follows: 1. The user is able to enable/disable the ForwardingLink Layer MTU filtering according to Forwarding Type, using <code>bcm_switch_control_indexed_set</code> and <code>switch type</code> <code>bcmSwitchForwardingLayerMtuFilter</code> <code>bcmSwitchLinkLayerMtuFilter</code>. 2. The user is responsible to configure the MTUs for the ForwardingLink layer MTU filters per profile (except for profile 0 which is configured separately per port), using <code>bcm_switch_control_indexed_set</code> and <code>switch type</code> <code>bcmSwitchForwardingLayerMtuSize</code> <code>bcmSwitchLinkLayerMtuSize</code>. 3. When creating an L3 interface, the user will specify the MTUs for the Forwarding and Link Layer MTU filters. The MTUs must match the same profile (since both filters access the same table to retrieve the MTU profile). 4. In order to enable the Forwarding Layer MTU filter, the VRF should be set in the Ingress PMF to the MTU profile. 5. Since the MTU profile is passed, the entries 1-3 in the table that maps VSI to MTU profile must point to profiles 1-3 respectively. Hence the MTU profiles for VSIs 1-3 must not be the same. This feature is enabled using soc property <code>custom_feature_mtu_advanced_mode</code>. Example can be found at <code>src/examples/dpp/cint_field_mtu_filtering.c</code></p> <p>Important Note! When enabling this feature, it is mandatory to configure the MTU size according to the calling sequence above, otherwise <code>bcm_l3_intf_create</code> will fail.</p>
SDK-64934	855302 56640_A0		<p>In previous releases, a timeout would occur when draining cells on CPU port. This occurred because on 5664X devices when MMU drain was performed, MAC based drain was invoked for CPU ports though the CPU does not have MAC operations. In this release, the MAC drain is skipped for CPU ports.</p>

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-64974	854596	88650_A0 88650_B0 88650_B1 88650ACP_A0 88660_A0	When both RCPU and CPU ports exist, init sequence fails. Fixed
SDK-64984	856594	88650_A0	E2E interface BW get returns wrong value(rate). Fixed
SDK-65000		88650_A0 88660_A0 88670_A0	New field age_state in bcm_l2_addr_t. hit indication can be taken from the flags param (BCM_L2_HIT)
SDK-65004		88650_A0 88660_A0 88670_A0	BFD: local_min_rx is assumed to be in microseconds, as dictated by RFC-5880.
SDK-65030	856303	56640_A0 56640_A1 56640_B0	In this release the IBOD TDM recovery patch has been ported to the baseline.
SDK-65036	856284	56340_A0 56340M_A0 56342_A0 56342M_A0 56344_A0	ARP_RARP_TO_FP & ARP_VALIDATION_EN fields were not configured in ING_CONFIG register as the ARP packets were not reaching FP. This issue has been resolved.
SDK-65053	855408	88660_A0	Lag egress disable bad behavior: When calling arad_ports_lag_sys_port_remove_unsafe with BCM_TRUNK_MEMBER_EGRESS_DISABLE flag to set one port to be standby, the source system port was restored to its original state as if the port was removed from the lag. Now, when a port is egress disabled, its src_sys_port stays the system port of the lag and not being restored.
SDK-65085	853726	56340_A0 56340M_A0 56342_A0 56342M_A0 56344_A0 56640_A0 56640_A1 56640_B0	On TRIUMPH3 devices, when an l2_entry (single wide) existed and the same mac entry was added as an extended l2_entry (Double Wide) with the same lookup keys; the single wide l2_entry was not being removed. This occurred because when an extended l2 entry was added, the Lookup logic was not searching if a single wide l2 entry already existed. In this release, logic has been added to enable search on both single wide and double wide l2 entries to correct this issue.
SDK-65112		88950_a0	Clear errors for FE3200 registers dump.
SDK-65115	831796	88660_A0	Fixed bcm_stk_gport_sysport_get to work correctly with FAP ports not existing in the local FAP.
SDK-65138		88650_A0	PORT_ADD_DONT_PROBE wrong value causes overlapping with other flags, so if BCM_PORT_ADD_CONFIG_CHANNELIZED flag was set, then PORT_ADD_DONT_PROBE is also set causing for unexpected behavior. Fixed
SDK-65139	853495	88660_A0	fixed for 6.4.2 bug does not appear in 6.4.4
SDK-65156		88650_A0	If NIFport is removed and then instead added non CAUI or ILKN port (XLAUI for example) then the MLF lanes are stuck (causing for traffic freeze) - Fixed
SDK-65197		88650_A0	After ILKN port is removed diag nif doesn't work well. Fixed
SDK-65206	857140	56340_A0 84848_A0	Previously, the PHY84848 EEE Idle Threshold was not updated when variable latency was enabled. In this release the Idle Threshold can be set for variable latency.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65213	845505	56850_A0 56850_A1 56850_A2 All	In earlier releases, obsolete L0 nodes were not detached from root. This has been fixed by marking obsolete L0 nodes as unused.
SDK-65214	843334	84334_B1 84848_A0	In previous releases, the <code>macsec_dev_addr</code> was not correctly initialized in the <code>phy84334</code> driver. This has been addressed by fixing the <code>macsec_dev_addr</code> internal representation.
SDK-65220	854751	56640_A0 56640_A1 56640_B0	TPID reference count was not maintained properly during <code>mim</code> port add and delete calls. This has been fixed by correcting <code>tpid</code> reference count logic.
SDK-65228	856737	All	In earlier releases, annotation of <code>BCM_L2_TRAVERSE_MATCH_STATIC</code> was not accurate. This has been resolved by updating the annotation according to function realization.
SDK-65241		88650_A0	When master channel of Interlaken interface is removed rx-loss errors occur. Fixed
SDK-65252		88650_A0 88660_A0	MPLS termination indexed: When working in MPLS termination indexed mode (<code>mpls_termination_label_index_enable=1</code>), reserved labels (labels range value 0-15) termination caused on some cases errors.
SDK-65253		88660_A0	Removed the validity check that prevented adding both port and interface status TLV to an OAM endpoint.
SDK-65254		88660_A0	OAM: When creating accelerated Up Endpoints the restriction that the LSB of the <code>src_mac_address</code> must match the local port has been lifted. This applies to CCMs as well as DMM/Rs, LMM/Rs.
SDK-65264		88650_A0	When a port(channel) is added to an existing interface(channelized interface) lane speed is reset to maximum, instead it should be unchanged. - fixed
SDK-65267		88950_a0	Few issues with FE3200 standalone compilation were fixed. Note: In FE3200 only compilation do not use PTP in the feature list as it is not supported
SDK-65285	852274	All	Documentation has been updated to better explain failover port addition and the need to pass the failover id in order to correctly program the protection tables.
SDK-65296	852193	88650_A0	Calling BW set/get for E2E port may cause for a memory failure - Fixed
SDK-65323		88650_A0 88660_A0	Corrective action on corrupted <code>IPS_QDESC</code> entries now works properly for unused flows (as per <code>IPS_FLWID</code>). Formerly, entries related to unused flows could be confused with entries related to flow id '0'.
SDK-65325		88670_A0	VPLS: Split-horizon related issues: 1. Default split horizon configuration is filtering in group 1 - out group 1. Other packets will pass. 2. An error is generated when in 2b orientation mode, and user adds mpls port with <code>BCM_MPLS_PORT_NETWORK</code> flag set and network group id = 0.
SDK-65330		56643_B0	Support has been added for spinlock APIs when using a non-glibc library.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65333	841588	56850_A0 56850_A1 56850_A2	If a VLAN was modified using <code>bcm_vlan_control_vlan_set()</code> , the VLAN based L3 ingress counters were lost. The logic used to do the store and restore operation have been corrected.
SDK-65336	859226	56640_A0 56640_A1 56640_B0	An access port added to a VPWS VPN would fail when added after attaching the PWEs to the same VPWS VPN. Support has been added in the MPLS/VPWS port add sequence to let the access port be added after the network ports.
SDK-65356	831773	56634_A0 56634_B0 56636_A0 56636_B0 56638_A0 56638_B0 56639_A0	An MPLS scenario in which an egress object was created with an invalid MPLS label and then updated to a valid MPLS label, would result in a bad configuration due to a reserved entry being cleared. This scenario has fixed by not clearing the reserved entry.
SDK-65373		56960_A0	Problem: For EFP/VFP, for group expansion logic, instead of legacy methods, new Tomahawk related methods were called. Solution: Updated code to call legacy methods for EFP/VFP for group expansion and call new Tomahawk methods for IFP only.
SDK-65380		56450_B0	In earlier releases the user was not able to set <code>MPLS_RAL_EXPOSED_TO_CPU</code> and <code>MPLS_GAL_EXPOSED_TO_CPU</code> using switch control in Katana2. These switch controls trap MPLS RAL/GAL label packets to CPU when exposed. In this release we have enabled RAL/GAL label trap to CPU switch control for Katana2. COSQ reason code is updated along with it.
SDK-65381		88650_A0	Dynamic port - when channelized interface is added the <code>SerDes speed(bcm_prot_speed_set)</code> should be configured only for the first port since it's an interface attribute. Fixed
SDK-65383		88660_A0	In MACT management, aging profiles were managed automatically. Now it is possible to decide which aging profile is set to each vlan, <code>bcm_vlan_control_vlan_set</code> is the way to control that. <code>bcm_vlan_control_vlan_set</code> takes VSI and control struct as input. In the control struct parameter set the <code>entropy_id</code> to the aging profile number for this VSI (could be 1,2 or 3). <code>aging_cycles</code> is the number of cycles until MAC table entry is deleted. <code>aging_cycles</code> is configured per <code>entropy_id</code> . Each age cycle takes 1/6 of the time configured in <code>bcm_l2_age_timer_set</code> .
SDK-65387	788746	88650_A0	MPLS: Fixed a bug where removing a pwe port resulted in also removing mpls tunnel with the same label
SDK-65404		56960_A0	Updated <code>bcm_field_group_priority_set()</code> API for TH IFP Groups to program the relative group priority value in "IFP_LOGICAL_TABLE_ACTION_PRIORITY" table in hardware.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65429	859917	56960_A0	User configured tx preemphasis value was inappropriately deleted after serdes init. Fixed a bug in the preemphasis config so that the value will be kept.
SDK-65431		56960_A0	Added support for Ingress Field Processor(IFP) actions on Tomahawk. bcmFieldActionDynamicHgTrunkCancel bcmFieldActionTrunkLoadBalanceCancel bcmFieldActionEcmpLoadBalanceCancel
SDK-65438		88650_A0 88660_A0 88670_A0	In CRPS: the counter engines are initialized via SOC properties . In some Added new a API to dynamically configure a counter engine: bcm_stat_counter_config_set(int unit, bcm_stat_counter_engine_t *engine, bcm_stat_counter_config_t *config); bcm_stat_counter_config_get(int unit, bcm_stat_counter_engine_t* engine, bcm_stat_counter_config_t *config);
SDK-65442		56450_B1 56450_B0	When the packing feature was enabled on 5645X (Katana2) devices, multicast queues would not be created for HG interfaces. This has been addressed and the queues are now created properly .
SDK-65458		56850_A1 56850_A2	Support has been added for setting EGR_L3_INTF:IVID_PRESENT_ACTION and EGR_L3_INTF:IVID_ABSENT_ACTION together on an egress L3 interface.
SDK-65468		56634_A0 56634_B0	L2 clear by port would not clear the L2 entry on a 56634 with External TCAM. This has been resolved by checking that the key type is vfi instead of valid vfi value.
SDK-65475	859138	88950_a0	Fix error when compiling without WARM_BOOT_SUPPORT
SDK-65497	858870	88640_A0 88650_B0 88650_B1 88660_A0 88670_A0	In Load Balancing, BFD payloads over GAL were hashed, therefore distributed among LAG members, which disturbs BDF sessions. This is fixed.
SDK-65507	860390	88650_A0 88650_B0 88650_B1 88650ACP_A0 88660_A0	FC: LLFC can be enabled by setting fc_inband_intlkn_calender_llfc_mod e to 1 or 2. A problem occurs when enabling LLFC TX on ELK interface. LLFC will affect ELK related feature. LLFC TX will be disabled on ELK interface after the fix.
SDK-65524		88650_A0 88660_A0	the proper value to be restored are not min > max, but min=0 max=0xff. this was fixed, however to prevent packets from being sent via a port, it should be removed from the MC-id. bug was fixed on 6.4.2, bug wasn't found on 6.4.4

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65530	856190	All	pll configuration added for unused plls At system with multiple plls, there might be some plls that are not used and no input clock reference connected for them on board. For those plls, BCM driver needs to discard lock status otherwise driver failure will occur. This fix adds SoC property configuration for users to notify BCM driver about disabled plls. NIF and fabric plls disable configuration change: /* Reference clock frequency for the Network Interface SerDeses. * 0=125MHz, 1=156.25MHz, -1 = Disable. Default: 156.25MHz. */ #define spn_SERDES_NIF_CLK_FREQ "serdes_nif_clk_freq" /* Reference clock frequency for the Fabric SerDeses. * 0=125MHz, 1=156.25MHz, -1 = Disable. Default: 156.25MHz. */ #define spn_SERDES_FABRIC_CLK_FREQ "serdes_fabric_clk_freq"
SDK-65531	855514	88650_A0 88660_A0 88670_A0	TPID classification maps a TPID incoming combination to a TPID class either as part of bcm_port_tpid_set() in standard VLAN edit or using bcm_port_tpid_class_set() in Advanced VLAN edit. TPID classification configurations weren't applied for FCoE packets, resulting in default IVE & EVE operations. The fix enforces the TPID classification configuration to all supported packet types.
SDK-65536	851330	56452_B0 56450_A0 56450_B0 56450_B1 56455_A0	In previous releases, the TR 140 sanity test case was not running. This was caused by incorrect logical checks performed during the TR 140 test initialization. These checks have been corrected.
SDK-65538		88660_A0	When using external TCAM, "block clear" commands which were executed under traffic caused momentary traffic loss in some cases. "block clear" commands are now called at initialization process, so they won't be called at run time and will not cause traffic loss. This fix requires usage of KBP-SDK 1.3.3 and setting of the new device property KBP_DEVICE_PRE_CLEAR_ABS.
SDK-65545	857700	56450_A0 56450_B0 56450_B1	Previously, FP slice priority was not reported in fp show. In this release, fp show has been updated to display priority.
SDK-65550	860780	56850_A0 56850_A1 56850_A2	The API bcm_vxlan_stat_counter_set would not function properly when the statistics were cleared due to the offset requirement. This issue has been fixed by removing the offset requirement.
SDK-65553	856749	All	In previous releases, during Warmboot UDF priority was not properly recovered on Trident2. This would cause the next group creation to fail after Warmboot. This has been fixed and the tcam_arr priority is correctly restored.
SDK-65555	860583	All	In previous releases, mmap in Linux User BDE would truncate the physical address for base address above 4G boundary. In this release, the declaration of the physical address variable has been fixed to depend on compiler.
SDK-65618	859717	56450_B0	In earlier releases, dynamic change of port encapsulation was not handled in MMU for this device. As part of fix the support has been implemented

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65634		56850_A0 56850_A1 56850_A2	Previously, when an mpls port was assigned by trunk and a member removed from this trunk, some original port attributes were not cleared. This has been corrected so that when a port is removed from the trunk, the extra attributes will clear. When a port is added to the trunk, the extra attributes are attached.
SDK-65644	861345	56960_A0	Fixed bcm_l2_matched_traverse() API returning E_INTERNAL error for bcm56960 switch device.
SDK-65645	861318	56960_A0	In previous releases, L2 overflow was not supported in TH, In this release, L2 overflow is supported.
SDK-65650		88660_A0	In the previous release, the API bcm_port_control_set with bcmPortControlFloodUnknownUcastGroup returned BCM_E_RESOURCE when it was set more than 64K times. In this release, the issue has been addressed by adding reference count of dpp_am_template_l2_flooding_profile when destroying the VLAN gport.
SDK-65652	842044	56224_A0 56224_B0	System initialization would fail after second Warmboot on Raven devices. This was seen only when exit clean was used instead of init bcm for Warmboot. This was the result of a memory corruption due to incorrect access of entry parts in double/triple wide modes. The memory corruption has been fixed in double/triple wide mode entries. A second issue also occurred. Slice pbmp was incorrectly recovered for groups which didnt have any entries installed. During group construct in Warmboot, group pbmp was not being copied to slice pbmp. This has also been fixed in the release.
SDK-65660	859077	56224_A0 56224_B0	Peak Kbits per burst and Committed Kbits per burst recovery were not accurately reported due to an error used in computing the hardware values. The logic used for calculating these values has been corrected.
SDK-65703		88650_A0	When a channel belonging to a channelized NIF is removed, the whole interface is unmapped from the PS. Fixed
SDK-65717		88650_A0 88650_B0 88650_B1 88660_A0	On a CAUI port the API bcm_port_link_status_get would not return the current indication on the first call. This would require the API to be called twice. As a side-effect, the RX LOS thread might not lock up which would lead to traffic loss on RX. The RX LOS has been fixed to verify the link status and clear the bit. As a side-effect the RX LOS thread might not lock up, and would cause traffic loss on RX. The RX LOS is now fixed to read the link status twice to clear the sticky bit.
SDK-65742	862352	88670_A0	L3: Add FEC destination MPLS tunnel SWAP command (EEI), by calling bcm_l3_egress_create with mpls_action=BCM_MPLS_EGRESS_ACTION_SWAP and mpls_label=new_label value.
SDK-65744	861642	88950_a0	Add FE3200 support for TR60
SDK-65746		88650_A0 88650_B0 88650_B1	In SDK 6.4.2, couldn't set ethernet policer for large number ports. Now the issue is fixed.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65767	862516	All	In the previous release, drop counters coming from the MMU block were indexed incorrectly resulting in the show counters command showing counters for a port other than the specified port. The root cause of the issue was that the MMU block was incorrectly indexed by Source Port rather than the MMU port number. The indexing has been corrected in this release
SDK-65785	862465	56850_A0	In previous releases, the hit bit would not clear as expected because the flag BCM_IPMC_HIT_CLEAR in the API bcm_ipmc_add would get lost. This has been addressed by considering the flags when hardware resources are configured.
SDK-65802		56440_A0 56450_A0 56640_A0	Support has been added for two new flags. The BCM_OAM_GROUP_GET_FAULTS_ONLY flag has been added to bcm_oam_group_get API. This flag can be passed as part of flags field in bcm_oam_group_info_t structure, which tells bcm_oam_group_get API to fetch only faults and persistent_faults fields in the structure. The flag is cleared before return. The BCM_OAM_ENDPOINT2_GET_FAULTS_ONLY flag has been added to bcm_oam_endpoint_get API. This flag can be passed as part of flags2 field in bcm_oam_endpoint_info_t structure, which tells bcm_oam_endpoint_get API to fetch only faults and persistent_faults fields in the structure. The flag is cleared before return.
SDK-65826	860887	88650_B1 88660_A0	CINT QOS: There was a typo inside function qos_run in cint_qos.c that qos_map_id of ingress L3 interface was wrongly assigned "mpls_eg_map_id" instead of "mpls_in_map_id". This typo has been corrected in this release.
SDK-65827	861512	88670_A0	Support for bcmFieldIpFragNonOrFirst and bcmFieldIpFragNotFirst values for FP bcmFieldQualifyIpFrag qualifier was added. These values support qualification on IP packets which are first fragments or no fragments, or packets which are not first fragments.
SDK-65849	862957	88650_B0	TRILL MC - Fixed lookup key of Trill Multicast forwarding when working in non prune mode.
SDK-65858	863156	88650_B1 88660_A0	Support has been added to Arad for the API bcm_port_cable_diag(). This function retrieves port (phy) cable diag status.
SDK-65864	853809	88650_A0 88660_A0	In FCoE, Currently, when adding route with flag BCM_FCOE_LOCAL_ADDRESS (which is added to SEM), the key is built from 8 MSB of DFCID. Because of different VSAN can use the same DFCID the VSAN also added to the key, which means that the key to SEM is now VSAN+8 MSB of DFCID.
SDK-65873	860480	56334_B0	Fixed incorrect hardware table index applied in MPLS type of QoS map.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65874		88660_A0	<p>Egress Programmable Editor (PRGE) has constraint in resources such as the number of PRGE programs that can be loaded, the number of required LFEM and Program branch points. PRGE programs are initialized during device initialization. To enable more flexibility added some SOC properties that enables to control loading / unloading of particular programs as below:</p> <p>Unload PROG_OAM_EGRESS_SNOOP_WITH_OUTLIF_INFO - custom_feature_egress_snooping_advanced=0/1</p> <p>Unload PROG_EDITOR_PROG_OAM_LM_UP_TRAP - custom_feature_oam_lm_upmep_enable=0/1</p> <p>Unload PROG_EDITOR_PROG_OAM_LOOPBACK - oam_rcy_port=-1/<port val></p> <p>Unload PROG_EDITOR_PROG_BFD_ECHO - bfd_echo_enabled=0/1)</p> <p>Unload ARAD_EGR_PROG_EDITOR_PROG_REFLECTOR_ETH - RFC2544_reflector_mac_swap_port=0/<port val></p> <p>Unload PROG_EDITOR_PROG_OAM_DM_DOWN_TRAP - custom_feature_oam_dm_tod_msb_add_enable=0/1</p> <p>Unload PROG_EDITOR_PROG_OAM_DM_UP_RX_TRAP - custom_feature_oam_dm_upmep_rx_enable=0/1</p> <p>Unload PROG_EDITOR_PROG_OAM_CCM - custom_feature_oam_upmep_enable=0/1</p> <p>Unload PROG_EDITOR_PROG_END_TLV_MAID_INT - oam_maid_11_bytes_enable=0/1</p>
SDK-65897	857814	88650_B1 88660_A0	<p>VXLAN/L2GRE: BCM API bcm_tunnel_terminator_create set the learning data when lookup key type is SIP. API verify updated to include FEC length check.</p>
SDK-65913	862524	56440_A0 56445_A0	<p>Support has been added to Katana for the config variable module_64ports to enable Dual Modld.</p>
SDK-65916	863208	56847_A1 56846_A0	<p>In previous releases, memory test TR 50 would fail on EGR_VLAN_X and EGR_VLAN_Y due to an incorrect mask setting. This issue has been fixed by setting the correct port bitmap mask and parity control for EGR_VLAN_X and EGR_VLAN_Y.</p>
SDK-65924	863613	56450_A0 56450_B0	<p>Buffer corruption could result when a change in LLS hierarchy and link flap occurred on a port. This has been corrected by locking the LLS hierarchy change during the port flush and adding a protection mechanism.</p>

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-65953	863455	88650_B1	new rx_los API was added: rx_los_set_active_sleep_config(uint32_active_sleep_usec) . call it from cint, the same way as rx_los_set_config() . the value active_sleep_usec will be used whenever the state of the link is rx_los_state_no_signal_active.
SDK-65957	862881	88650_A0 88660_A0	QOS: color (DP) was always returned as 0 by bcm_qos_map_multi_get() API no matter what the mapping from ({pcp, dei} to {tc, dp}) was on ingress LIF direction. The root cause was color (DP) value was not retrieved from the right parameter. This release has fixed this issue.
SDK-65981	863743	56850_A2	In earlier releases, the whole route table could not be traversed chunk by chunk while calling l3 defip show and l3 ip6route show CLI commands. This has been fixed in this release by extending the usage of the two commands.
SDK-65985		88660_A0	In Arad+ devices, BFD packets are generated by the OAMP, which also creates the UDP and IPv4 headers. The source IP for IPv4 headers is retrieved from a pool of 16 addresses. The ability to extend the available source IPs is required. This is added. In order to enable this feature, a soc property must be enabled, and tunnel should be created with specified source IP and newly calculated IP header checksum. Calling sequence: 1. Enable soc property bfd_extended_ipv4_src_ip=1 2. Add to sequence of BFD endpoint creation, a tunnel creation as follows: bcm_tunnel_initiator_t tunnel; bcm_tunnel_initiator_t_init(&tunnel); tunnel.type = bcmTunnelTypeIpAnyIn4; tunnel.sip = 0x10203040; // desired source IP tunnel.aux_data = 0x11ae; // calculated IP header checksum tunnel.dip = 0; tunnel.l3_intf_id = link_layer_if; / / this is the interface ID that is usually specified in the BFD endpoint create bcm_tunnel_initiator_create(unit, &bfd_extended_l3_intf, &tunnel);.. bfd_endpoint_info.egress_if = bfd_extended_l3_intf.l3a_intf_id; // this is the interface ID that was returned.. bcm_bfd_endpoint_create(unit, &bfd_endpoint_info);
SDK-65991		88650_A0 88660_A0 88670_A0	Required changes in SDK in order to support KBP- SDK 1.3.3 for external TCAM are introduced.
SDK-66021		56850_A2	In earlier release, the API bcm_port_learn_modify() would return BCM_E_PORT when the parameter gport is correct, which was incorrect. Now it is fixed.
SDK-66069		88650_A0 88650_B0 88660_A0	IMPORTANT NOTE Changing application code is required. L3: flag BCM_BCM_L3_ENCAP_SPACE_OPTIMIZED was fixed to BCM_L3_ENCAP_SPACE_OPTIMIZED.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66081		88650_A0	When CAUI channel added to an existing interface, traffic may get lost since CLP is reset each time channel is added - Fixed
SDK-66110		56547_A0	During an ECMP add operation, the SDK improperly read VLAN_ID to configure L3_OIFs. This read would truncate the interface number and result in the wrong value for an L3 interface with indexes greater than 4K. This has been corrected by reading L3_OIF rather than VLAN_ID.
SDK-66113		88660_A0	OAM: For Y.1731 OAM over MPLS-TP, the MDL may be ignored as it is not really used. This behavior may be set with the soc property: "custom_feature_oam_over_mpls_ignore_md1=1". In this case, when calling bcm_oam_endpoint_create() for non accelerated endpoints (i.e. without the opcode_flag CCM_IN_HW) the field "level" is ignored and thus must be 0. For accelerated endpoints, "level" controls the MDL field of transmitted OAM PDUs.
SDK-66120		56450_A0 56450_B0	In previous releases, when SrcIp6High and DstIp6High were added to QSET in EFP, non-ipv6(L2, ipv4) egress traffic would not hit the EFP entry. This occurred because this particular QSET combination is not permitted on Katana2. In this release, group creation with this QSET combination will not be allowed.
SDK-66130		88660_A0	Delay based ECN marking now supports both ECT0 and ECT1
SDK-66148		88650_B1 88660_A0	ECN DM: ECN threshold configuration was fixed.
SDK-66155		56640_B0	Previously, the priority field of the L2 table entry would always be set for TRIUMP3 devices. This occurred because there wasn't a check in place and the Priority field was set unconditionally. A check has been added to set the priority field only if the user specifies it.
SDK-66156		56850_A0 56850_A1 56850_A2 56851_A0 56851_A1 56851_A2 56851P_A1 56851P_A2 56852_A0 56852_A1 56852_A2 56853_A0 56853_A1 56853_A2 56854_A0 56854_A2 56854_B0 56855_A0 56855_A2 All	In the previous release, GLP was computed based on port ID and module ID when port type was trunk. This would limit the maximum trunk ID to 127. In this release, the issue has been addressed by computing GLP based on TGID for trunk port.
SDK-66159		88650_A0	Added support for L2 encapsulation for external CPU, tunneling to CPU using Ethernet (.1q) encapsulation. In this format, the egress pipeline will remain the system-headers of the packet and encapsulate Ethernet header. Ethernet header will be constructed per Destination-CPU that will be retrieved from a Data-Entry (OutLIF). At the ingress pipeline, special mechanism will skip the Ethernet header. CINT example will be provided separately.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66180		All	In soft error recovery (SER) and shadowing for internal memories and registers, some cases were mishandled for blocks IHP, IHB, EPNI and OAMP. The corrective action for a parity error in one of the mentioned blocks would be handled as an exact match table parity error. This has been fixed and exact match tables parity errors are handled differently than other parity errors.
SDK-66189		88650_A0	bcm_cosq_control_get() with bcmCosqControlFlowSlowRate control didn't return the correct value. Fixed.
SDK-66209		88650_B0 88650_B1 88660_A0	OAM: Fixed a bug which prevented updating endpoints, i.e. calling bcm_oam_endpoint_create() for existing endpoints with the flag BCM_OAM_ENDPOINT_UPDATE.
SDK-66211		All	In L3, when calling API bcm_l3_egress_create() with flag BCM_L3_REPLACE set, for an EEP which was not previously allocated, lock release handling was faulty. This is fixed and lock is released properly in any case.
SDK-66251		56450_A0 56450_B0 All	On Katana2 chips, physical ports disabled during boot-up were not initialized for hardware linkscan. When enabled during Hot-swap, hardware linkscan would not work for newly formed flex ports as these ports were not initialized. This was corrected by initializing hardware linkscan for all physical ports during initialization.
SDK-66257		56640_A0 56640_A1 56640_B0 All	On BCM5664x chips, policers in committed mode showed incorrect attributes after Warmboot. If the user attempted to reset the values it may result in other policer attributes being overwritten. This issue has been resolved by correcting the recovery of policer parameters.
SDK-66279		88650_A0 88660_A0 88670_A0	Setting of TPIDs for a logical interface (MPLS_PORT, MIM_PORT) via bcm_port_tpid_set() is required in some applications for EVE as the packet TPID parsing doesn't include the native ETH. In case of a PWE gport, a TPID configuration for a LIF that wasn't previously configured, resulted in a wrong TPID profile configuration which may cause EVE to recognize packet with wrong tag format (for example: untagged Native Ethernet packet even if header is tagged). The PWE gport configuration via bcm_port_tpid_set() was fixed so that the configuration will have an impact regardless of the previous TPID configuration.
SDK-66311		88660_A0	BFD: OAMP may transmit BFD over IP frames with incorrect UDP source port.
SDK-66316		56847_A1 56849_A1 56840_A0 56841_A3 56841_B0 56842_A0 56843_B0 56844_A0 56845_A2 56845_B0 56846_A0 56846_A1	In earlier releases, a function argument was hard coded. This has been fixed in this release by taking unit number as argument.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66322		56850_A0	In previous releases, there was no support to get/set flexible counter values using counter ID. In this release, the feature has been supported by providing following APIs: bcm_stat_flex_counter_get bcm_stat_flex_counter_sync_get bcm_stat_flex_counter_set.
SDK-66336		All	Adding and removing ports many times (depends on how many ports were on init) will cause for bcm_port_add to fail due to excessive use of memory - fixed
SDK-66338		56850_A0 56850_A1 56850_A2 56851_A0 56851_A1 56851_A2 56851P_A1 56851P_A2 56852_A0 56852_A1 56852_A2 56853_A0 56853_A1 56853_A2 56854_A0 56854_A2 56854_B0 56855_A0 56855_A2 All	In earlier release, system might crash when adding alpm route with new VRF if TCAM entries were exhausted. This issue has been fixed.
SDK-66343		56450_A0 56450_B0	On Katana2, the API 'bcm_l2_ring_replace' would not perform the desired operation leading to the inability to perform fast flush for ring protection. Support has been added so that 'bcm_l2_ring_replace' works on these devices.
SDK-66385		88650_A0 88660_A0 88670_A0	IMPORTANT CHANGE: The first 4 Ingress Actions IDs are reserved for logical interfaces (MPLS_PORT, MIM_PORT) that parse TPIDs for the native ETH above it (BCM API: bcm_port_tpid_set). In Advanced VLAN editing, the operation and the TPID Profile that was associated with those IVE Action IDs weren't used properly, resulting in possible wrong IVE & EVE operations. Those Actions in AVT (advanced VLAN translation mode), are now performed as expected for IVE & EVE. In addition, the modification of those 4 IVE Actions IDs was blocked.
SDK-66400		56547_A0 56640_B0	This issue was specific to WLAN scenarios. The NLF port was deleted from the replication bitmap when any of the WLAN ports were deleted from the multicast group. This issue has been fixed and the WLAN port is deleted from the replication bitmap only when the last WLAN port in that multicast group has been deleted.
SDK-66414		88660_A0	BFD: BFD over IP packets transmitted by the OAMP may have an incorrect UDP dest port.
SDK-66419		All	Removing non channelized ports may cause for packet drop at EGQ for other non-channelized ports - Fixed
SDK-66423		88660_A0 88670_A0	New default learning mode is set in the init of the driver. The new mode is INGRESS DISTRIBUTED. EGRESS INDEPENDENT is no longer supported
SDK-66429		88650_B0 88650_B1 88660_A0	OAM: OAMB block can get corrupted causing incorrect OAM classification.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66471		88650_A0 88650_B0 88650_B1 88660_A0	Parity memory error correction with shadow: Upon parity error event SW tries to fix the corrupted entry with shadow saved in advance (if available). There are some memories at SCH block which shadowing correction sequence is not implemented for them. This fix will add the implementation for these memories. SW will fix corrupted entries with their shadow upon memory parity error event (if shadow available). The list of memories added to memory parity correction with shadow: . ARAD_INT_SCH_CSDTPARERROR . ARAD_INT_SCH_FIMPARERROR . ARAD_INT_SCH_PSDDPARERROR . ARAD_INT_SCH_PSDTPARERROR . ARAD_INT_SCH_SEMPARERROR . ARAD_INT_SCH_SIMPARERROR . ARAD_INT_SCH_TMCPARERROR
SDK-66480		88660_A0	OAM: Calls to <code>bcm_oam_endpoint_create()</code> for existing endpoints may fail to properly update SW databases, causing subsequent calls to <code>endpoint_destroy()</code> to fail.
SDK-66495		88660_A0	BFD: When calling <code>endpoint_create()</code> for an existing endpoint with the flag <code>BCM_BFD_ENDPOINT_UPDATE</code> the SW state is not properly saved, causing subsequent calls to <code>endpoint_destroy()</code> to fail.
SDK-66503		56850_A0	The accelerated linkscan algorithm was notifying the application of link up status only after the software linkscan interval. This has been fixed so after the link is up, the notification will occur on the next SW linkscan slow cycle.
SDK-66550		56850_A2	When each front port took more than 12 L1 nodes, the L1 nodes allocated by the internal LB port would be robbed by the front ports leading to an incorrect LLS structure. This has been corrected by using the gport structure to rebuild the cosq hierarchy when changing to ETS mode.
SDK-66571		88650_A0 88660_A0	VPLS: Support added for case where FEC destination is BLACK HOLE.
SDK-66586		88660_A0	In the previous release, the port extender feature was not enabled when the user only configured " <code>custom_feature_map_port_exter_enabled_<port>=1</code> " in soc property, which was incorrect. In this release, this issue has been addressed by updating setting of packet header when reading SOC configuration.
SDK-66614		88660_A0	New SOC property: <code>diag_l2_disable</code> default = 0 EGRESS_INDEPENDENT mode is going to be dropped and the new default mode requires some more configurations. There is a need for some I2 configurations after the init stage of the device. The new SOC property will allow disabling these configurations. The new I2 appl stage will configure learning messages.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66623		88650_A0 88660_A0	<p>We exposed the option to change the header offset.</p> <p>- In <code>bcm_rx_trap_config_t</code>: <code>bcm_rx_trap_forwarding_header_t</code> <code>forwarding_header</code>; /* Forwarding header position overridden value */</p> <p>- Where <code>bcm_rx_trap_forwarding_header_t</code> is: typedef enum <code>bcm_rx_trap_forwarding_header_e</code> { <code>bcmRxTrapForwardingHeaderPacketStart</code> = 0, /* Start of packet. */ <code>bcmRxTrapForwardingHeaderL2Header</code> = 1, /* Ethernet header. */ <code>bcmRxTrapForwardingHeaderFirstHeader</code> = 2, /* First header after Ethernet. */ <code>bcmRxTrapForwardingHeaderSecondHeader</code> = 3, /* Second header after Ethernet. */ <code>bcmRxTrapForwardingHeaderThirdHeader</code> = 4, /* Third header after Ethernet. */ <code>bcmRxTrapForwardingHeaderFourthHeader</code> = 5, /* Fourth header after Ethernet. */ <code>bcmRxTrapForwardingHeaderOamBfdPdu</code> = 6 * OAM or BFD PDU header. */ <code>bcmRxTrapForwardingHeaderCount</code> /* Always Last. Not a usable value. */ } <code>bcm_rx_trap_forwarding_header_t</code>;</p> <p>If the flag <code>BCM_RX_TRAP_UPDATE_FORWARDING_HEADER</code> is present, then the forwarding-header is done based on <code>bcm_rx_trap_forwarding_header_t</code>.</p>
SDK-66627		56440_A0 56445_A0	In this release, support has been added for DUAL MODID in MMU cosq for Katana devices.
SDK-66658		88650_B0 88650_B1 88660_A0 88670_A0	<p>New SOC property: <code>custom_feature_optimized_de_allocation</code> was added to support optimization of key allocation for direct extraction databases. This SOC property allows to change the order of key assignment of Direct Extraction: instead of Key A [179:80], ..Key D [179:80], Key A [79:0], .., Key D [79:0] - it will be Key A [79:0], Key A [179:80], .., Key D [79:0], Key D [179:80].</p>
SDK-66660	All		In Trident 2 BFD packet injection into ports belonging to Y-Pipe was not working due to incorrect queue-mapping. This issue has been fixed
SDK-66671		56850_A0 56850_A1 56850_A2 56851_A0 56851_A1 56851_A2 56851P_A1 56851P_A2 56852_A0 56852_A1 56852_A2 56853_A0 56853_A1 56853_A2 56854_A0 56854_A2 56854_B0 56855_A0 56855_A2 All	An interrupt storm would occur when the NAPI (polled) mode was enabled but Linux KNET base device had not been brought up. The software has been updated to display a warning message and disable the interrupt if NAPI mode is enabled and the base device has not been brought up.
SDK-66675		88660_A0	OAM: Fixing a bug in <code>bcm_oam_loopback_add()</code> for endpoints of type MPLS.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66691		56850_A0 56850_A1 56850_A2 56851_A0 56851_A1 56851_A2 56851P_A1 56851P_A2 56852_A0 56852_A1 56852_A2 56853_A0 56853_A1 56853_A2 56854_A0 56854_A2 56854_B0 56855_A0 56855_A2 All	In earlier releases, number of strict priority children nodes was updated incorrectly when detaching a queue from scheduling node . This has been fixed in this release by updating SP node count according to scheduling mode of the gport that being detached.
SDK-66702		56960_A0	In this release: 1. eyescan is now performed on the correct lane for all ports 2. eyescan now honors the lane option 3. The capability to print out lane mask has been added
SDK-66723		88660_A0	Fixed ILKN interface support for BCM88460
SDK-66731		88650_A0 88660_A0	MPLS: Encapsulation Optimized mode creation and deletion fixes. Note that BCM_MPLS_PORT2_LEARN_ENCAP flag can not be used in EGRESS_ONLY configuration.
SDK-66734		88650_A0 88660_A0	Important notes: the BCM_L2_CACHE_LOOKUP flag is not supported anymore. The usage of BCM_L2_CACHE_LOOKUP was incorrect in bcm_l2_cache_set: - to modify the destination in the bcm_l2_cache_addr_t addr structure, the user needs to set a valid destination in addr->group, addr->dest_trunk or addr->dest_port - otherwise, the destination is not set, and it is unnecessary to set BCM_L2_CACHE_LOOKUP flag to indicate an invalid destination.
SDK-66739		56850_A0 56850_A1 56850_A2 56851_A0 56851_A1 56851_A2 56851P_A1 56851P_A2 56852_A0 56852_A1 56852_A2 56853_A0 56853_A1 56853_A2 56854_A0 56854_A2 56854_B0 56855_A0 56855_A2	In previous releases, a print of soc_td2_sched_weight_get function was used "sched_weight_set". This has been fixed by using "sched_weight_get" instead of "sched_weight_set".
SDK-66746 SDK-67307		56850_A0	In the previous release, the soc_xgs3_l3_ecmp_hash used an incorrect mask and crc_val_shift, resulting in the hardware ECMP hash differing from the software based hash result. This issue has been addressed by ensuring the soc_xgs3_l3_ecmp_hash uses the correct mask and crc_val_shift for the ECMP hash.
SDK-66764		56850_A0 56850_A1 56850_A2	PORT_LOCK has been added to protect critical resource when accessing PHY APIs (soc_phyctrl_notify and soc_phyctrl_enable_set).

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66770		56850_A0 56850_A1 56850_A2 All	<p>In previous releases, the user was unable to create a single FP group with 12 UDF chunks. When allocating UDF qualifier of 2bytes each, UDF1 was completely filled first before allocating chunks from UDF2. This group creation resulted in double wide mode.</p> <p>In this release, UDF1 has more flexible programming in FP. For smaller chunks less flexible UDF2 should be given more priority than UDF1 during UDF chunk allocation.</p>
SDK-66775		56640_A0	Traffic would stop flowing over the backup tunnel after the VC over primary tunnel was deleted and recreated. This would occur because the VPLS backup port was also being deleted when deleting the VPLS primary port. This has been corrected and now "bcm_mpls_port_delete" will only delete the port that is provided by user.
SDK-66783		All	In previous releases, non-flex ports were not properly handled by the PBM automatic linkscan update. When non-flex ports were re-enabled they didn't return to hardware linkscan as expected. This issue has been resolved by distinguishing the handling of flex ports and non-flex ports.
SDK-66784		56850_A0 56850_A1 56850_A2	In earlier releases, the API <code>bcm_cosq_bst_stat_clear</code> failed to clear BST related stat counters in Y-PIPE. This has been addressed by changing the API sequence to ensure both X-PIPE and Y-PIPE counters are properly cleared.
SDK-66787		56860_A0 56960_A0	MPLS functionality for label switch routers (LSRs) has been extended for BCM5686x devices by adding an option to forward packets to an ECMP group, which can be translated to a set of MPLS labels at the egress.
SDK-66790		56514_A0 56820_A0	The number of free L3 entries returned by the API <code>bcm_xgs3_l3_info()</code> was not correct on some devices. This has been resolved by distinguishing the different IPv4MC entry widths between various chips in the <code>L3_ENTRY</code> table.
SDK-66803		56850_A0 56850_A1 56850_A2	The next hop reference count was not being updated only for supported chipsets. This has been corrected by adding SOC checks to update the next hop reference count only for supported chipsets.
SDK-66831		All	An LED activity handling issue existed in which variables were improperly declared which could lead to a greater variable becoming negative. The variable type has been modified and they are now properly declared to eliminate this handling issue from occurring.
SDK-66832		All	ELK and statistics interface doesn't need a PS but they consume a PS - Fixed
SDK-66855		56850_A0 56850_A1 56850_A2 All	In earlier releases, the usage information of IP4 option profiles was stored in persistent storage, so Level 1 warm reboot could not be supported if IP4 option profiles have been configured explicitly by API. This has been documented in this release and applications must select Level 2 full recovery mode to recover the usage information of IP4 option profiles in that situation.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-66882		56450_A0 56450_B0	P_VECT_SPRI_7_4 is , P_NUM_SPRI are not reset to 0 when node is deleted , instead corresponding l2 index in the vector is reset.
SDK-66939		84793_A0	The phy control dump command did not correctly display drive current values. An incorrect register was being accessed, causing the Tx driver current to not function properly. This issue has been fixed by modifying the register access.
SDK-66950		56850_A0 56850_A1 56850_A2 All	The API documentation has been corrected to state that there is no fixed order when creating the two types of ports for MPLS failover.
SDK-66957 SDK-66998		All	If an update of the primary mpls port was made after deleting the failover mpls port, a BCM_E_NOT_FOUND error was returned. The has been addressed by fixing the handling of BCM_MPLS_PORT_REPLACE flag in the bcm_mpls_port_add API when adding ports to a VPWS VPN.
SDK-66960		56847_A1 56849_A1 56840_A0 56841_A3 56841_B0 56842_A0 56843_B0 56844_A0 56845_A2 56845_B0 56846_A0 56846_A1	When ALPM_ENABLE was defined on products which didn't support ALPM, a crash would occur when using "l3 alpm show all". This occurred because ALPM support was not properly determined. In this release, ALPM support is properly determined.
SDK-66962		88660_A0	In the previous release, arad_pp_lif_cos_profile_info_set_unsafe() only configured the lower 32 bits of IHP_COS_PROFILE_USE_L3r and IHP_COS_PROFILE_USE_L2r. In this release, this issue has been addressed by configuring all 64 bits of IHP_COS_PROFILE_USE_L3r and IHP_COS_PROFILE_USE_L2r.
SDK-66963		All	bcm_l3_egress_get is used to retrieve information about a FEC entry. There was a problem where a bcm_l3_egress_get for an odd FEC id would not return the failover_id even if the FEC is part of a protected pair. This fix makes sure that the failover_id is returned even for odd FEC ids.
SDK-66978		88650_B0 88650_A0 88660_A0	BFD: Remote Discriminator (remote_discr) field can now be updated on an existing endpoint using the API bcm_bfd_endpoint_create with the flag BCM_BFD_ENDPOINT_UPDATE.
SDK-66993		88650_A0 88670_A0	When CPU port is added it may exceed the range it was allocated for (according num_queues_pci) - Fixed to return error in such case.
SDK-67002		56340_A0 56340M_A0 All	In earlier releases the helix4 led ports were remapped to unused ports, which caused led ports to get wrong status. Code has been changed to skip unused ports on helix4.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-67015		56640_A0 56640_A1 56640_B0	A packet with the invalid IP header should be treated as a L2 packet. The previous SDK only treated the unknown type packets as L2 packets. The root cause happened when the SDK initialized the ESM_PKT_TYPE_ID. When the packet type was FWD_TYPE_L2_VLAN, the L2_PAYLOAD was only set to "1", so only the encapsulated packets with unknown ethertype were treated as L2 packets. This issue has been resolved. The SDK implements another rule (L2_PAYLOAD=0) when the packet type is FWD_TYPE_L2_VLAN. Now a packet with invalid IP header is treated as an L2 packet.
SDK-67020		88660_A0	In IPv4 Multicast, when working with GLOBAL-IPMC, (ipmc_vpn_lookup_enable = 0). In the look-up stage (FLP) two look-ups were performed with key <RIF,SIP,DIP>, one in the TCAM and one in the LEM. When both look-ups resulted in a hit, the TCAM result had precedence over the LEM result. This is wrong behavior. The look-up priorities are changed and now the LEM look-up has precedence over the TCAM look-up.
SDK-67021		88650_A0	Deleting and adding ports many times will cause of compensation error. Fixed
SDK-67050		88950_a0	Change in soc property --- serdes_preemphasis, containing: preTap(03:00), mainTap(10:04), and postTap(15:11) instead of --- serdes_preemphasis, containing: preTap(03:00), mainTap(09:04), and postTap(14:10)
SDK-67067		88650_B0 88650_A0 88660_A0	VPLS optimized mode: Added support for PWE PHP case where there is no MPLS tunnel in VPLS packet (i.e. DataoPWEoE). In this case FEC is pointing directly to LL (Link-layer). In these cases VLAN must be retrieved from the egress LL, this is done by bcm_l3_egress_create_vlan field.
SDK-67092		56850_A2	In previous releases, the soc_mem_clear routine only cleared X-pipe instance of a table which has access type 6. This has been addressed so that now tables that have access type 6 will clear the x-pipe instance, then clear the y-pipe instance, and the SBS control register will be restored to default value.
SDK-67107		56960_A0	For multi-link ports, using PRBS in per-port mode, but assuming per link results will create confusion. The solution is to use per link PRBS functions. This functionality has been designed but not added until now. It is now added.
SDK-67108		88660_A0 88670_A0	bcm_port_dscp_map_set() which is a global configuration was not implemented as BC88670-PG1XX-R. Change bcm_port_dscp_map_set() implementation as below to meet the 88670-PG1XX-R: 1. port = -1, do the global TOS to TC,DP mapping. We support 2 global mapping, Table index is according to prio & SECONDARY. 2. port != -1, do the port to profile mapping. Profile (table index) is according to prio & SECONDARY. The API is backward compatible when not using the secondary table index.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-67111		56960_A0	Fixed 2 issues 1. DstClassL3 and DstClassL2 are part of a group(both share the same secondary selectors), put the fix to program right secondary selectors values. 2. chunk offset adjustments for pass through bits in Level1 Bus.
SDK-67156		88650_A0	OAM: When creating multiple MEPs/MIPs on the same LIF OAM exact match tables may be configured incorrectly, causing some MEPs/MIPs to fail to trap packets.
SDK-67191		88660_A0	VPLS optimized mode: VPLS Optimized mode can not be configured on devices with ERSPAN or Routing Over VXLAN/Trill support (88660 only).
SDK-67200		56640_A1 56640_B0 56640_A0	All BFD packets were being redirected to UKERNEL. This has been fixed to redirect only the packets having a label on which BFD is enabled towards UKERNEL
SDK-67218		56850_A2	In earlier releases, SDK didn't reprogram OBM thresholds when port speed change. This has been fixed in this release by updating PGW OBM thresholds when port speed changes.
SDK-67237		56854_A2	A scheduling failure message would be displayed during system initialization of a BCM56854_A2 . The 20G sorting behavior during MMU TDM scheduling has been fixed.
SDK-67263		88660_A0	pop/swap commands arnt converted to swap anymore , cints fixed.
SDK-67308		56643_B0	The final ECMP path was generated by performing a module operation on the value of bits [4:0] from the hashing logic using the Count of ECMP member in the group. But in switch devices of Triumph3 family, they use bits [9:0] instead of bits [4:0]. The code has been corrected to use bits [9:0] to generate the final ECMP path for Triumph3 family.
SDK-67398		56850_A2	In earlier releases, existing IPv4 LPM routes were affected when <code>lpm_scaling_enable =1</code> and the user deleted IPv6 LPM routes. A change has been made to handle when range1 and range2 IPv4 routes are contiguous.
SDK-67431		56634_A0 56636_A0 56638_B0 56636_B0 56638_A0 56639_A0 56634_B0	Triumph2 FP Slice Mode was going to Quad mode with udf configured as Data2 and Data3 initialization was not done correctly. Solution: Data2 and Data3 are now initialized during qualifier init. Also udf bmap is set accordingly.
SDK-67440		88650_B1 88660_A0	OCB configuration failure: In 6.4.3 some changes were made to <code>bcm_cosq_gport_threshold_get</code> and set that made them to fail, this fix repairs these APIs <code>bcm_cosq_gport_threshold_get</code> and set takes now a VOQ connector as gport while configuring the OCB parameters.
SDK-67467		88650_A0	Previously, when a memory corruption event occurred in a memory that was part of an array of memories, the entry index was not calculated correctly and corrupted memory was not fixed. In this release, the array entry calculation is corrected and the corrupted entry of parity protected array memories fixed upon a SER error event.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-67492		56340_A0 56640_A1 56640_B0 56640_A0	Class-based station move was not working for the VP whose type was TRILL, L2GRE, Extender, NIV or Subport. This has been fixed.
SDK-67493		56046_B0	The CPU utilization for the Ranger+ counter thread was too high. This issue has been addressed by modifying the FP_COUNTER_TABLE for ranger+ to 2K entries in-line with the IFP table size.
SDK-67496		56850_A2	In previous releases, replication queues on Trident2 were not configured, which would result in all multicast traffic competing for the same queue. This has been addressed by assigning proper replication queues for each multicast COS.
SDK-67501		56640_A1 56640_B0 56640_A0 All	The physical to logical table for the port config BCM56643_4X10_4X42 had a value configured for a disabled port resulting in an assertion. This issue has been fixed so that the disabled port value is no longer in the table.
SDK-67506		88660_A0	In the previous release, upstream IPv6 compatible MC packets hit IP compatible MC program which has the higher priority than IP6 SAV program. In this release, the issue has been addressed by disabling IP compatible MC program because it's not necessary for PON application.
SDK-67535		88670_A0	New flag was added to allow disabling of events. e.g. disabling refresh events for vid=1: bcm_l2_addr_distribute_t dis; dis.flags = BCM_L2_ADDR_DIST_REFRESH_EVENT BCM_L2_ADDR_DIST_SET_NO_DISTRIBUTE_R; dis.vid = 1; bcm_l2_addr_msg_distribute_set(0, dis);
SDK-67556		56260_A0	This JIRA enables BroadSync firmware to Saber2.
SDK-67594		All	Applicable only to simulator, not applicable on real devices. PLI simulator is not updating the DMA DESC DONE status in the CMIC_IRQ_STAT register, instead, updating it in CMIC_DMA_STAT register - Fixed
SDK-67633		56640_A1 56640_B0 56640_A0	Modified the bcm_cosq_control_set/ bcm_cosq_control_get API to support the following control types: bcmCosqControlEgressUCQueueSharedLimitBytes bcmCosqControlEgressUCQueueMinLimitBytes bcmCosqControlEgressUCSharedDynamicEnable bcmCosqControlEgressUCQueueLimitEnable bcmCosqControlEgressMCQueueSharedLimitBytes bcmCosqControlEgressMCQueueMinLimitBytes bcmCosqControlEgressMCSharedDynamicEnable bcmCosqControlEgressMCQueueLimitEnable bcmCosqControlDropLimitAlpha
SDK-67634		56960_A0	In BCM56960, 16 bits are used to represent a port with trunk represented by bit 17(bit pos 15). In BCM56850, it was 15 bits, with trunk represented by bit 16 (bit pos 15). The fix is to set bit position 16 for trunk in BCM56960.
SDK-67635		All	Fix "diag nif" from BCM shell changing configuration of synce. When synce is enabled, "diag nif" will run without changing the configuration of synce, but won't show the serdeses rates.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-67636		88660_A0	In the previous release, <code>bcm_l3_source_bind_delete()</code> couldn't remove TT TCAM and FLP LEM entry separately for IP6 compression. In this release, the issue has been addressed by deleting TT TCAM entry and FLP LEM entry separately based on source binding flags.
SDK-67647		56640_B0	Added IpProtocol qualifier for ESM_SHORT_IPV6 mode
SDK-67673		56854_A2	Fixed MSI interrupt mode for non-iProc switch devices. This functionality was broken when MSI support was added for iProc-based devices.
SDK-67680		56640_A1 56640_B0 56640_A0	In previous releases, when network-side primary virtual port was deleted from the system in VPWS environment, destination virtual port of access-side virtual port was not modified. It pointed to an invalid virtual port. If other virtual ports were created and deleted repeatedly, it might cause some logic disorder. This issue was fixed in deletion procedure of VPWS virtual port by assigning correct value to DVP if access-side virtual port existed.
SDK-67681		56850_A1	In previous releases, when computing I3 table hash by CLI, error result was encountered intermittently. This has been resolved.
SDK-67691		56640_A1 56640_B0 56640_A0	In previous releases, the implementation of replacing next hop index in failover protection caused bad performance. All 64k entries would be read and compared with the entry which was expected to be replaced one by one even if the entry had never been used. This took too much CPU resource. Logic has been added to check whether the entry is valid and only the valid entry will be compared with the entry expected to be replaced. And the algorithm of for comparisons has been optimized as well.
SDK-67711		All	After removing a port and then calling for <code>bcm_port_vlan_priority_map_set</code> for another port will cause for a failure - Fixed
SDK-67718		88650_A0	When working with ILKN per channel counter mode, a port remove will cause for segmentation error - Fixed.
SDK-67731		56850_A2 56850_A0 56850_A1	In previous releases when a ser event error correction or clearing failed, it did not disable the specific parity control register. Now this issue has been fixed.
SDK-67732		56224_B0 56224_A0	Raven devices would not support the wild card option for the Vlan ID parameter when the API <code>vlan_translate_egress_delete()</code> was used. This had been inadvertently removed from the SDK. In this release, support for the Vlan ID wild card has been reimplemented for the API <code>vlan_translate_egress_delete()</code> .
SDK-67742		56340_A0 56342M_A0 56344_A0 56342_A0 56340M_A0	Added <code>l3_extended_host_entry</code> support in Helix_4 family chips.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-67779		88660_A0	In IP multicast, when adding a multicast entry via API <code>bcm_ipmc_add()</code> , the input VID was validated to be 0-4095. In case the ingress interface is not enabled for IP multicast, the vid stands for FID, which can be larger than 4095. Now the input VID is validated to be 0-4095 only in case when the ingress interface is enabled for IP multicast.
SDK-67781		88650_B1 88660_A0	User header is no longer removed when editing the network headers with ARP extension feature
SDK-67804		56850_A2 56850_A0 56850_A1 56640_A1 56640_B0 56640_A0	In early releases of the API Programming Guide, the document said that the port would permit untagged packets of X bytes and tagged packets of X + 4 bytes when the user used API <code>bcm_port_frame_max_set</code> to set ports maximum frame size to X. But after further investigation, we found that the port permitted X + 4 bytes untagged and tagged packets. The description has been corrected.
SDK-67830		56640_A0	In earlier releases, the API <code>bcm_cosq_port_sched_set()</code> returned an error when the port was a High-Speed Port and the ETS mode was enabled. This has been fixed.
SDK-67831		56640_A0	When the API <code>bcm_cosq_port_bandwidth_set()</code> was called for a high-speed port to configure the bandwidth distribution among CoS queues, the bandwidth value configured didn't take effect. This has been fixed.
SDK-67847		56850_A2 56850_A0 56850_A1 All	In previous releases, when using command 'mirror show', it did not show vp based mirror. This issue has been fixed.
SDK-67851		56248L_B0 56450_A0 56248L_A0	The bitmap for "MAC enabled ports" and the link bitmap were not being properly updated after Warmboot. This could result in a traffic hit during Warmboot. This issue has been corrected and the state is now properly recovered."
SDK-67855		56440_A0 56450_A0 56440_B0 56440_A1 56450_B1 56450_B0	PROBLEM: Bucket size setup on KT2 was half of the actual. ROOTCAUSE: Burst quantum was not calculated properly. SOLUTION: Adjusted burst quantum calculation to correct value.
SDK-67857		88650_A0 88660_A0	MPLS PORT: Creating FORWARDING_GROUP (FEC) via <code>bcm_mpls_port_add</code> , and then deleting it by calling <code>bcm_mpls_port_delete</code> was failing if no outlif was allocated.
SDK-67887		88650_A0 88660_A0 88670_A0	TPID for PWE-LIF: A PWE LIF is assigned TPIDs that form a TPID Profile associated with LIF and used for VLAN translation. The APIs <code>bcm_port_tpid_set()</code> are used <code>bcm_port_inner_tpid_set()</code> are used respectively to set the Outer and Inner TPIDs of the profile. The TPID profiles can use only 4 TPID values per device. The TPID configuration for a PWE LIF had failed when attempting to utilize the TPIDs per device table to the maximum. The calculation of the available TPIDs per device was fixed so that a PWE LIF configuration can make use of the last available TPID.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-68008		56850_A2 56850_A0 56850_A1 84328_A0 84328_B0	In previous releases, on TD2 with PHY 84328, the port would link down after changing encapsulation. This issue has been resolved.
SDK-68009		56450_A0 56455_A0 5645_A0 56452_B0 56457_B0 56450_B1 56456_A0 56450_B0 56456_B0	If port mirroring mode is set to both ingress and "true egress", the <code>bcm_mirror_port_get()</code> function skipped check for "true egress" mode (flag <code>BCM_MIRROR_PORT_EGRESS_TRUE</code>) and only returned <code>BCM_MIRROR_PORT_INGRESS</code> flag as set. This has been corrected
SDK-68012 SDK-68530		56340_A0 56340M_A0	In earlier releases, BCM INIT unconditionally did initialization for the management port even if the port was not a member of <code>spn_PBMP_VALID</code> . This could result in a system segment fault. The issue was fixed in this release.
SDK-68041		56456_B0	Issue: To change encap type to higid2 restriction of config variable <code>higid2_hdr_mode=1</code> is prerequisite in the current implementation. Solution: To change encap type to higid2 for any port capable of higid2 config variable <code>higid2_hdr_mode=1</code> restriction is removed.
SDK-68056		All 88660_A0	BFD: In the <code>bfd_endpoint_create()</code> API the fields <code>remote_flags</code> and <code>local_flags</code> support the P, F, and D flags (these are the Flags on the BFD control packet). P, F and D may not be set together (hitherto this restriction applied to the F and P flags as well).
SDK-68067		88650_B1 88660_A0	In Field module, the user can qualify on RxTrapCode qualifier. An issue with egress traps is fixed.
SDK-68072		88660_A0	In IPv4 multicast, when SOC property <code>ipmc_vpn_lookup_enable</code> is not set, an error was returned when trying to remove an entry, using the API <code>bcm_petra_ipmc_remove()</code> . This is fixed and using this API to remove IPv4 multicast entries returns a reliable return value.
SDK-68073		88660_A0	MPLS: Enable <code>bcm_mpls_port_add</code> replacement with <code>BCM_MPLS_PORT_INGRESS_ONLY</code> flag set.
SDK-68078		88650_B1 88650_B0 88650_A0 88640_A0 88660_A0 88670_A0	In Metering, calling <code>bcm_policer_create()</code> attached a meter instance to a bandwidth profile but didn't initialize the respective Policer Committed and Excess bucket levels. As a consequence, tokens previously stored in the buckets are used, effecting or even conflicting with the new configuration. The fix initializes the meter bucket levels to -1 (empty) upon creation.
SDK-68109		56540_B0 56540_A0	When programming ipfix mirror configuration registers for a specific port, the previous SDK releases did not use the port number in programming register. This has been corrected.
SDK-68130		56450_A0 56450_B1 56450_B0	On Katana2, the SDK was updating the RPEX fields for default route add when urpf was enabled which was incorrect since the RPEX and DEFAULTROUTE fields in L3 Table are not overlaid unlike other devices. Check has been added to not update the RPEX fields for KATANA2.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-68141		88650_A0 88650_B0 88650_B1 88660_A0 88670_A0 88750_A0 88750_B0 88950_a0	For future compatibility, coexist of BCM88650/ BCM88660/BCM88750 with next generation devices, the system reference clock should be configured to 1200MHz. Added driver support to system reference clock of 1200MHz and changing the default values of the relevant soc property (system_ref_core_clock) to 1200MHz.
SDK-68221	All		The behavior of <code>bcm_port_frame_max_set()</code> / <code>bcm_port_frame_max_get()</code> is inconsistent between XE and CE ports. A 4 byte padding is added internally by the SDK to account for the .1q tag for CE ports, however this is not done for XE ports. This release ensures consistent behavior across XE and CE ports, with respect to max frame size configuration and retrieval.
SDK-68261		56640_A0	An abnormal peak temperature was occasionally displayed when "show temp" was performed. The issue was due to an incorrect initialization sequence for temperature monitor. Rearranged the initialization sequence.
SDK-68286		56960_A0	Problem: Following commands did not work. <code>phy diag xe0 pcs phy diag xe0 pcs link phy diag xe0 pcs speed phy diag xe0 pcs aneg phy diag xe0 pcs antimers phy diag xe0 pcs state</code> The issue was in definition of following function. <code>int tscf_phy_pcs_info_dump(const phymod_phy_access_t* phy, uint32_t mode)</code> It should be <code>int tscf_phy_pcs_info_dump(const phymod_phy_access_t* phy, void *arg)</code> The tier1 function it was connected was wrong. Is should be connected following function. <code>int tefmod_diag_disp(PHYMOD_ST *pc, char* cmd_str)</code> After fixing above issue all the command described above works.
SDK-68333		88650_B0 88650_B1 88660_A0 88670_A0	In Metering, when two meters were applied in Serial Mode, the result color of the second meter was taken as the Output color even if the first meter didn't have sufficient credits of that color. In some scenarios, this might cause limits defined by the first meter to be completely ignored. This fix uses the PCD map to make sure credits are available for Output Color in both meters. If no credits are available in the first meter, Output Color will be changed to Red. As a consequence of this fix, PCD map will be enabled by default in both Serial and Parallel modes.
SDK-68336		88670_A0	Trill/VXLAN/L2GRE bounce-back filter: Bounce-back filter was adjusted to use orientation mechanism.
SDK-68342		56640_A0 56640_A1 56640_B0 All	In previous releases, SDK crashed when <code>L2_ENTRY</code> table was full, and <code>bcm_bfd_endpoint_create()</code> was called to create BFD session. This has been fixed.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-68351		88650_A0 88650_B0 88650_B1 88670_A0 All	In the previous release, SW and HW were out of sync after deleting and recreating the STG. In this release, this issue has been addressed by modifying HW STG initial status to forward in <code>bcm_stg_create_id</code> .
SDK-68356		56224_A0 56224_B0	Policer id retrieval logic was not proper when compact policer id and stat id's were pushed to scache file. Fixed the same.
SDK-68394		88650_A0	In DNX architecture, the PPH Learn-Extension is in general useful for Egress learning, transmitting the learning information to insert into the MAC table. However, when using Ingress learning, the user can use this Learn-Extension as raw data, e.g. to transmit information from Ingress PMF to Egress PMF (40 bits). The calling sequence is then: 1. Set the SOC property <code>bcm886xx_pph_learn_extension_disable</code> , so that no packet has a PPH learn extension by default. 2. For packets which should carry additional information to Egress, set the Field Processor action <code>bcmFieldActionFabricHeader = bcmFieldFabricHeaderEthernetLearn</code> to add such an extension, and set the Learn-information with the FP actions <code>bcmFieldActionLearnSrcPortNew</code> & <code>bcmFieldActionLearnInVPortNew</code>
SDK-68396		88660_A0	feature was delivered
SDK-68406		56440_A0 56440_A1 56440_B0 56450_A0 56450_B0 56450_B1	PROBLEM : Configured burst value in hardware was sometimes lesser than the API requested value. ROOTCAUSE: The burst calculation logic while calculating the hardware value was not adjusting the value correctly. SOLUTION : Corrected the burst adjustment logic to ensure the hardware configured value is at atleast as permissible as the API requested value.
SDK-68410		56040_A0 56340_A0	In the API <code>bcm_l2_cache_set()</code> , the sanity of the field "prio" was always checked even when the flag <code>BCM_L2_CACHE_SETPRI</code> was not set. This caused an error return. This has been fixed by adding the enabling check of <code>BCM_L2_CACHE_SETPRI</code> flag before the sanity check of the priority value.
SDK-68414		56960_A0	Schan access errors seen on BCM5696x series of devices when booting up a device with one or more port blocks that have been powered off in hardware. This issue is addressed by a) Bypassing via existing hardware controls, all port blocks that do not have atleast one member port specified in config.bcm as a <code>portmap_*</code> property. b) Ensuring software state is updated to reflect what port blocks are available for programming as per config.bcm.
SDK-68444		88650_A0 88660_A0	when a rop test to the kbp fails, a recovery sequence is activated in order to reset the ILKN and the KBP. there is an option to enable/disable this recovery sequence using the SoC property <code>custom_feature_kbp_recover_enable</code> . this feature is disabled by default.
SDK-68527		56850_A2	RangeCheck width was being initialized incorrectly as 32bits instead of 24bits. Fixed the same.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-68567		88660_A0 All	OAM: The RDI bit of transmitted CCMs can now be controlled manually. This may be achieved by using the flag <code>BCM_OAM_ENDPOINT_RDI_TX</code> and without using the flag <code>BCM_OAM_ENDPOINT2_RDI_FROM_RX_DISABLE</code> . Notes: 1. When adding an RMEP with the same ID as that of the local MEP, or ID (8K + that of the local MEP), if the flag <code>BCM_OAM_ENDPOINT2_RDI_ON_RX_RDI</code> is set then the RDI bit on outgoing packets will be updated according to incoming CCMs associated with one of those 2 RMEP entries. 2. When calling <code>endpoint_create()</code> for the local endpoint with the flag <code>*_ENDPOINT_REPLACE</code> set, the RDI bit will be set if and only if the flag <code>BCM_OAM_ENDPOINT_RDI_TX</code> has not been set. If the user does not wish to change the state of the RDI, the user must read this flag with the <code>endpoint_get()</code> API and set the flag accordingly in <code>endpoint_create()</code> .
SDK-68585		88660_A0	In Meter, when a meter was configured with a low rate for which the Exponent in the internal representation was zero, the SDK took the Mantissa value floor, even if this resulted in an inaccuracy. Software will now prefer a Mantissa value closest to the exact value (also consider the ceiling).
SDK-68586		88650_B1 88660_A0	In the previous release, once port is enabled as an extender port, if mirroring egress packets of this port, mirrored packets contained one byte that indicated the remote port, which was incorrect. In this release, the issue has been addressed by stripping the one byte header at ingress stage of mirror cycle. There is a limitation for mirroring plus dynamic switching. It doesn't supported setting mirror first and enabling CoE second. It supports enabling CoE first and setting mirror second.
SDK-68640		88660_A0	In Parser, some Parsing Macros were configured to mark packets with PLC/PFC 0x00 upon stopping following a parsing error, causing issues in subsequent blocks. Macros are now configured to fail-back to the calling Macro PLC/PFC values in case of a parsing error.
SDK-68646		All	Chips Affected: All Description: The <code>bcm_vlan_port_add()</code> API was accepting invalid pbmp(that is ports that are not present), to get added as vlan member. The fix validates the port bitmap and allows only valid bitmaps to get added to vlan member list.
SDK-68721		88660_A0	When Mirror-dest port is configured as extender-port, the port extended additional leading information was not added. Now PRGE port extender support SPAN traffic as well.
SDK-68723		56960_A0	There was an array overrun within a loop in oversub sorting logic. The problematic code has been removed with no impact to behavior.
SDK-68727		56960_A0	In this release, adapted 'show ring osc' command to account for differences between Trident2 and Tomahawk hardware.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-68733		56850_A0 56850_A1 56850_A2 All	In previous releases, the information of VPLESS failover port on VPWS application could not be replaced by calling <code>bcm_mpls_port_add()</code> . This has been fixed by supporting the usage of <code>BCM_MPLS_PORT_REPLACE</code> flag on VPLESS port.
SDK-68769		88650_A0 88650_B0 88650_B1	System header parsing in little endian system was fixed.
SDK-68772		56640_A0 56640_A1 56640_B0	In earlier releases, traffic couldn't hit <code>EXT_IP6_FWD_TABLE</code> when the three soc properties <code>ext_ip6_fwd_table</code> , <code>ext_ip4_fwd_table</code> and <code>ext_ip4_acl_table</code> were present in <code>config.bcm</code> . In this release, this bug was fixed.
SDK-68882		56960_A0	Added support for Following 4 qualifiers in Tomahawk IFP. <code>bcmFieldQualifyOuterVlanPri</code> <code>bcmFieldQualifyOuterVlanCfi</code> <code>bcmFieldQualifyInnerVlanPri</code> <code>bcmFieldQualifyInnerVlanCfi</code>
SDK-68919	All		Added grand master clock id to foreign master dataset entry
SDK-69025	All		When <code>entropy_id=0</code> and <code>aging_cycles=0</code> in <code>bcm_vlan_control_vlan_set</code> ignore these fields and do not update the aging profiles. In order to disable aging for a given VSI use <code>bcm_vlan_control_vlan_set</code> with <code>entropy_id=0</code> and <code>aging_cycles=-1</code>
SDK-69235		88650_B1 88660_A0	<code>BCM_L3_HIT_CLEAR</code> flag will now clear the FEC's hit indication once provided to <code>bcm_l3_egress_get()</code> . This flag was mistakenly treated in <code>bcm_l3_egress_create()</code> , although this doesn't make sense. This is now fixed.
SDK-69406		88650_B1 88660_A0	In FEC, when protection is used, the <code>failover_id</code> of the even FEC wasn't extracted correctly when this FEC record was requested by FEC index. This is now fixed.
SDK-69515		88660_A0	In Metering, when Sharing is enabled, there was an error in the way the minimum and maximum rates are calculated for multiple parameters that share Reverse-Exponent. The problem solved and now the minimum and maximum are calculated correctly
SDK-69654	All		Added grand master clock id to foreign master dataset entry.
SDK-69785		88670_A0	Fixed a small memory overwrite in VSQ attributes handling.
SDK-70142		88660_A0 88670_A0	Extender: Added a CINT (<code>cint_port_extender_cb_uc.c</code>) that demonstrates the configuration of a Controller Bridge for Uni-Cast traffic. Note: Some of the APIs are yet to be supported.

Table 120:

Number	CSP #	Chips	Release Notes For 6.4.4
SDK-70476		88670_A0	During init the arad_pp_eg_encap_range_info_set_unsafe() function reads value of register EPNI_EEDB_OUTLIF_ACCESS and traversing through ARAD_PP_EG_ENCAP_NOF_BANKS(). In arad the number of banks was 16 and register long is 32 bit In Jericho number of banks is 24 and the register width is 48 bit using a 32-bit register for setting the encap range for each bank in Jericho caused a memory override. this was fixed.

Section 6: Unresolved Issues for 6.4.4

The following issues are unresolved in version 6.4.4 of the SDK.

Table 121:

Number	CSP #	Chips	Release Notes
SDK-33686	389108	56634_A0	If a MiM virtual port has statistics enabled for it and if such MiM port is replaced using BCM API <code>bcm_mim_port_add()</code> along with flag <code>BCM_MIM_PORT_REPLACE</code> then the statistics of that MiM port might be lost.
SDK-44416		88640_A0	1. API is reading the wrong register from the device. 2. API is missing the parameter of <code>ResetLoad</code> , so this value cannot be configured.
SDK-45075		All	When an interrupt occurs on different blocks of the same type (e.g. multiple FMAC blocks), the count will be accumulated in the same counter. For example RX-LOS interrupt may occur on different FMAC blocks, and counted as same recurring event, although it is in fact a different interrupt. This may affect corrective action in case it is different for a recurring event, in the case if recurring-threshold for this event is crossed.
SDK-45366	611273	56440_A0	When the API <code>bcm_cosq_port_bandwidth_set(...)</code> is called on a particular port and COSq to enable egress rate limiting, sometimes the CLI command "show c" will show the incorrect dropping statistics on a irrelevant port.
SDK-46556	621213	88650_A0 88650_B0 88650_B1	<code>bcm_l2_cache_delete()</code> API does not delete <code>general_trap</code> entry configuration in HW
SDK-47366	642398	All	The implementation of SER (Soft Error Recovery mechanism) requires the SDK to perform periodical scanning of certain memories. The infrastructure for this scanning is provided by the optional <code>MEM_SCAN</code> feature (component) of the SDK. Since SER is a mandatory component, that can't be compiled out, <code>MEM_SCAN</code> becomes a mandatory component too as long as you are using a device, supported by SER.
SDK-47739	628786	56540_A0 56540_B0	For devices in the BCM56540 family, the CPU queues are allocated differently depending on the revision of the device (Ax vs. Bx). This force the developer to include revision specific code in the application.
SDK-48091	662661	56850_A0 56850_A1 56850_A2	For BCM56850 devices, when only a single GigE port is allocated to a TSC lane (the other 3 TSC lanes are not used), that port may be configured incorrectly resulting in port that appears functional but is not.
SDK-51978		88650_A0	In a device with channelized CPU ports, where some of the CPU ports are HigiG and some not, the WB wont preserve the HigiG indication correctly.
SDK-52383		88650_A0 88650_B0	Cud extension for Arad is not supported
SDK-54219	744517	88650_B1	In ARAD B1, VxLAN/L2GRE packet size of 236B-299B will be dropped by EPNI if the packet needs to be terminated. Workaround is introduced in FP CINT: <code>cint_field_ingress_large_termination.c</code>
SDK-54488		88650_A0	In Field Processor, when creating a cascaded field group, a failure might occur in key allocation in case the key ID is already determined. To be investigated.

Table 121:

Number	CSP #	Chips	Release Notes
SDK-54623		88660_A0	In Field Processor, when creating a field group, false error messages are printed when operation succeeds (return value indicates success). To be fixed.
SDK-56851	767727	88650_B1 88660_A0	The guarantee hardware mechanism only guaranteed the resource (BDs). The number of BDs used is not smaller than the used buffers, but there are three buffer types and not each buffer type can be used for any packet.

Table 121:

Number	CSP #	Chips	Release Notes
SDK-60570	88650	A0	<p>BCM user can chose to cache/shadow some tables in order to allow recovery from Parity/ECC-2bit errors, or as a means of improving driver runtime performance. Currently caching mechanism cannot deal with tables that have multiple access-formats. Each such a table represents a single memory in the HW, but is represented by multiple table-names in the SW reflecting the various formats in which the memory can be arranged. Current implementation allocates a separate shadowed space for each format. As a result, updates to the same entry using different formats are not captured properly. The list of affected tables: Block: EPNI Table name: Eedb Bank Available formats: EPNI_EEDB_BANK EPNI_DATA_FORMAT EPNI_MPLS_PUSH_FORMAT EPNI_MPLS_POP_FORMAT EPNI_MPLS_SWAP_FORMAT EPNI_IPV4_TUNNEL_FORMAT EPNI_OUT_RIF_FORMAT EPNI_TRILL_FORMAT EPNI_TRILL_FORMAT_B_0 EPNI_LINK_LAYER_OR_ARP_FORMAT EPNI_AC_FORMAT EPNI_AC_FORMAT_WITH_DATA</p> <p>Block: IHB Table name: FEC Entry Available formats: IHB_FEC_ENTRY IHB_FEC_ENTRY_GENERAL IHB_FEC_ENTRY_FORMAT_NULL IHB_FEC_ENTRY_FORMAT_A- IHB_FEC_ENTRY_FORMAT_C</p> <p>Block: IHP Table name: Lif Table Available formats: IHP_LIF_TABLE IHP_LIF_TABLE_AC_P2P_TO_AC IHP_LIF_TABLE_AC_P2P_TO_PWE IHP_LIF_TABLE_AC_P2P_TO_PBB IHP_LIF_TABLE_AC_MP IHP_LIF_TABLE_ISID_P2P IHP_LIF_TABLE_ISID_MP IHP_LIF_TABLE_IP_TT IHP_LIF_TABLE_TRILL IHP_LIF_TABLE_LABEL_PWE_P2P IHP_LIF_TABLE_LABEL_PWE_MP IHP_LIF_TABLE_LABEL_PROTOCOL_OR_LSP</p> <p>Block: IHP Table name: Large Em Format Available formats: IHP_MACT IHP_MACT_FORMAT_0_TYPE_0 IHP_MACT_FORMAT_0_TYPE_1 IHP_MACT_FORMAT_0_TYPE_2 IHP_MACT_FORMAT_1 IHP_MACT_FORMAT_2 IHP_MACT_FORMAT_3_TYPE_0 IHP_MACT_FORMAT_3_TYPE_1</p> <p>Block: IPS Table name: Qpm 1 Available formats: IPS_QPM_1 IPS_QPM_1_SYS_RED IPS_QPM_1_NO_SYS_RED</p> <p>Block: IPS Table name: Qpm 2 Available formats: IPS_QPM_2 IPS_QPM_2_SYS_RED IPS_QPM_2_NO_SYS_RED</p> <p>Block: IRR Table name: MCDB Available formats: IRR_MCDB IRR_MCDB_EGRESS_FORMAT_0- IRR_MCDB_EGRESS_FORMAT_7 IRR_MCDB_EGRESS_SPECIAL_FORMAT IRR_MCDB_EGRESS_TDM_FORMAT IRR_MCDB_EGRESS_TDM_FORMAT</p> <p>Block: OAMP Table name: Mep Db Available formats: OAMP_MEP_DB OAMP_MEP_DB_Y_1731_ON_MPLSTP OAMP_MEP_DB_Y_1731_ON_PWE OAMP_MEP_DB_BFD_ON_IPV4_ONE_HOP OAMP_MEP_DB_BFD_ON_IPV4_MULTI_HOP OAMP_MEP_DB_BFD_ON_MPLS OAMP_MEP_DB_BFD_ON_PWE OAMP_MEP_DB_RFC_6374_ON_MPLSTP OAMP_MEP_DB_LM_DB OAMP_MEP_DB_LM_STAT OAMP_MEP_DB_DM_STAT</p>



Table 121:

Number	CSP #	Chips	Release Notes
SDK-64289		56850_A0 56850_A1 56850_A2	The BCM56850 supports UDF in the VFP but requests for UDF rules may result in a "feature unavailable" error condition.
SDK-69531	887804	88650_A0 88660_A0 88670_A0	TPID: A TPID Profile consists of two device level TPID values. The bcm_port_tpid_xxx APIs are used to associate an object (port or logical LIF) with up to two TPIDs, thereby modifying the available per device 4 TPID profiles and the per device TPID values. TPID values and TPID Profiles can be removed or modified if no object is assigned with those values as may happen when TPIDs that were previously set are being modified or deleted. Currently, TPIDs and TPID Profiles that were set aren't being released when object TPIDs are modified or deleted, thus such manipulations cause both the TPIDs and TPID Profiles HW tables to be full earlier than expected.

Section 7: Device and Platform Support

The section describes all devices, platforms, and operating systems that are supported by this release.

SWITCH DEVICES

Table 122: Switch Devices

Family	Devices	Description
BCM5389	BCM5389 A0	8-Port GbE Switch with Integrated Serdes
	BCM5389 A1	8-Port GbE Switch with Integrated Serdes
BCM5396	BCM5396 A0	16-Port GbE Switch with Integrated Serdes
BCM53010	BCM53010 A0	5-Port Gigabit Ethernet Managed Switch integrated with single core ARM Cortex-A9 processor
	BCM53010 A2	
	BCM53011 A0	5-Port Gigabit Ethernet Managed Switch integrated with dual cores ARM Cortex-A9 processor
	BCM53011 A2	
BCM53012	BCM53012 A0	5-Port Gigabit Ethernet Managed Switch with one RGMII I/F integrated with dual cores ARM Cortex-A9 processor
	BCM53012 A2	
	BCM53017 A0	2-Port Gigabit Ethernet Managed Switch with one RGMII I/F integrated with dual cores ARM Cortex-A9 processor
	BCM53018 A0	5-Port Gigabit Ethernet Managed Switch with one RGMII I/F integrated with dual cores ARM Cortex-A9 processor
BCM53019	BCM53019 A0	5-Port Gigabit Ethernet Managed Switch integrated with dual cores ARM Cortex-A9 processor
	BCM58522 A0	5-Port Gigabit Ethernet Managed Switch integrated with 2 PHYs, ARM Cortex-A9 processor and macsec cores
	BCM58525 A0	5-Port Gigabit Ethernet Managed Switch integrated with 2 PHYs, SGMII I/F, ARM Cortex-A9 processor and macsec cores
	BCM58622 A0	8-Port Gigabit Ethernet Managed Switch integrated with 5 PHYs, ARM Cortex-A9 processor and macsec cores
BCM58623	BCM58623 A0	8-Port Gigabit Ethernet Managed Switch integrated with 5 PHYs, ARM Cortex-A9 processor and macsec cores
	BCM58625 A0	8-Port Gigabit Ethernet Managed Switch integrated with 5 PHYs, SGMII I/F, ARM Cortex-A9 processor and macsec cores
	BCM53101 A0	5-Port Fast Ethernet Managed Switch + 1 Fast Ethernet WAN port
	BCM53101 B0	
BCM53115	BCM53115 A0	5-Port GbE Managed Switch + 1 Gigabit WAN port with integrated serdes
	BCM53115 A1	
	BCM53115 B0	
	BCM53115 B1	
BCM53118	BCM53118 C0	
	BCM53118 A0	8-Port Gigabit Ethernet Switch
	BCM53118 B0	
	BCM53118 B1	
BCM53125	BCM53125 A0	5-Port Gigabit Ethernet Switch with 1 Gigabit WAN port and 8051 processor
	BCM53125 B0	
BCM53128	BCM53128 A0	8-Port Gigabit Ethernet Switch with embedded 8051 processor
	BCM53128 B0	
BCM53242	BCM53242 A0	Managed Switch with 24 FE Ports + 2 GbE Interface
	BCM53242 B0	



Table 122: Switch Devices

Family	Devices	Description
	BCM53242 B1	
	BCM53262 A0	Managed Switch with 24 FE Ports + 4 GbE Interface
	BCM53262 B0	
	BCM53262 B1	
BCM53280	BCM53282 A0	8-Port Fast Ethernet + 2-Port Gigabit Ethernet Multilayer Switch
	BCM53282 B0	
	BCM53282 B1	
	BCM53282 B2	
	BCM53283 A0	16-Port Fast Ethernet + 2-Port Gigabit Ethernet Multilayer Switch
	BCM53283 B0	
	BCM53283 B1	
	BCM53283 B2	
	BCM53284 A0	24-Port Fast Ethernet + 2-Port Gigabit Ethernet Multilayer Switch
	BCM53284 B0	
	BCM53284 B1	
	BCM53284 B2	
	BCM53286 A0	24-Port Fast Ethernet + 4-Port Gigabit Ethernet Multilayer Switch
	BCM53286 B0	
	BCM53286 B1	
	BCM53286 B2	
	BCM53288 A0	24-Port Fast Ethernet + 2-Port Gigabit Ethernet Multilayer Switch with one 2.5GbE Uplink Port
	BCM53288 B0	
	BCM53288 B1	
	BCM53288 B2	
BCM53300	BCM53300 A0	Managed 24-port L2 Switch
	BCM53300 A1	
	BCM53301 A0	Managed 16-port L2 Switch
	BCM53301 A1	
	BCM53302 A0	Managed 24-port L2 Switch
	BCM53302 A1	
BCM53310	BCM53312 A0	BCM53312 Integrated Multilayer Switch and CPU
	BCM53312 B0	
	BCM53313 A0	BCM53313 Integrated Multilayer Switch and CPU
	BCM53313 B0	
	BCM53314 A0	BCM53314 Integrated Multilayer Switch and CPU
	BCM53314 B0	
BCM53320	BCM53322 A0	BCM53322 Integrated Multilayer Switch and CPU
	BCM53323 A0	BCM53323 Integrated Multilayer Switch and CPU
	BCM53324 A0	BCM53324 Integrated Multilayer Switch and CPU
BCM53400	BCM53405 A0	16-port 10GbE Multilayer Ethernet Switch
	BCM53406 A0	12-port 10GbE plus 8-port 2.5GbE and 4-port 5GbE/2.5GbE Multilayer Ethernet Switch

Table 122: Switch Devices

Family	Devices	Description
	BCM53415 A0	16-port 10GbE Multilayer Ethernet Switch with integrated CPU
	BCM53416 A0	12-port 10GbE plus 8-port 2.5GbE and 4-port 5GbE/2.5GbE Ethernet Switch with integrated CPU
BCM53600	BCM53602 A0	8-Port Fast Ethernet + 3-Port Gigabit Ethernet Switch with one 1/2G-EPON ONU MAC/SerDes and embedded 600MHz MIPS32 74K processor
	BCM53603 A0	16-Port Fast Ethernet + 3-Port Gigabit Ethernet Switch with one 1/2G-EPON ONU MAC/SerDes and embedded 600MHz MIPS32 74K processor
	BCM53604 A0	24-Port Fast Ethernet + 3-Port Gigabit Ethernet Switch with one 1/2G-EPON ONU MAC/SerDes and embedded 600MHz MIPS32 74K processor
	BCM53606 A0	24-Port FE with S3MII interface + 3-Port Gigabit Ethernet Switch with one 1/2G-EPON ONU MAC/SerDes and embedded 600MHz MIPS32 74K processor
BCM89500	BCM89500 A0	4-Port Integrated Dedicated BRPHY + 3-Port Gigabit Ethernet Switch with embedded ARM processor
	BCM89500 B0	
BCM89500	BCM89501 A0	4-Port Integrated Dedicated BRPHY + 1-Port Integrated Dual-Mode BRPHY + 2-Port Gigabit Ethernet Switch with embedded ARM processor
	BCM89501 B0	
BCM89500	BCM89200 A0	1-Port Integrated Dedicated BRPHY + 1-Port Integrated Dual-Mode BRPHY + 2-Port Gigabit Ethernet Switch with embedded ARM processor
	BCM89200 B0	
BCM53710	BCM53714 A0	BCM56714 Integrated Multilayer Switch and CPU
	BCM53714 A1	
	BCM53714 A2	
	BCM53716 A0	BCM56716 Integrated Multilayer Switch and CPU
	BCM53716 A1	
	BCM53716 A2	
	BCM53718 A0	BCM56718 Integrated Multilayer Switch and CPU
	BCM53718 A1	
	BCM53718 A2	
BCM53720	BCM53724 A0	Managed 24-port L2 Switch with Integrated CPU
	BCM53724 B0	
	BCM53726 A0	Managed 24-port L2 Switch with Integrated CPU
	BCM53726 B0	
	BCM5675 A1	
	BCM5676 A0	4-Port, 96-Gbps Switch Fabric
	BCM5676 A1	
BCM56010	BCM56014 A0	24-Port Integrated Multilayer Switch and CPU
	BCM56014 A1	
	BCM56014 A2	
	BCM56018 A0	48-Port Integrated Multilayer Switch and CPU
	BCM56018 A1	
	BCM56018 A2	
	BCM56018 A1	48-Port Integrated Multilayer Switch and CPU
BCM56020	BCM56024 A0	24-Port Integrated Multilayer Switch and CPU

Table 122: Switch Devices

Family	Devices	Description
	BCM56024 B0	
	BCM56025 A0	24-Port Integrated L2 Switch and CPU
	BCM56025 B0	
	BCM56026 A0	24-Port Integrated L2 Switch and CPU
	BCM56026 B0	
BCM56060	BCM56060 A0	16-port 10GbE Multilayer Ethernet Switch with integrated CPU
	BCM56063 A0	16-port 1GbE plus 4-port 10GbE (XFI) Multilayer Switch with integrated CPU
	BCM56064 A0	24-port GbE plus 4-port 10GbE Multilayer Managed Switch with HiGi Uplinks and integrated CPU
BCM56100	BCM56100 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Multilayer Switch
	BCM56100 A1	
	BCM56101 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Multilayer Switch with One 10-Gigabit Ethernet/HiGig Port
	BCM56101 A1	
	BCM56102 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Multilayer Switch with Two 10-Gigabit Ethernet/HiGig Ports
	BCM56102 A1	
	BCM56105 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Layer 2 Switch
	BCM56105 A1	
	BCM56106 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Layer 2 Switch with One 10-Gigabit Ethernet/HiGig Port
	BCM56106 A1	
	BCM56107 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Layer 2 Switch with Two 10-Gigabit Ethernet/HiGig Ports
	BCM56107 A1	
BCM56110	BCM56110 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Multilayer Switch
	BCM56111 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Multilayer Switch with One 10-Gigabit Ethernet/HiGig Port
	BCM56112 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Multilayer Switch with Two 10-Gigabit Ethernet/HiGig Ports
	BCM56115 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Layer 2 Switch
	BCM56116 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Layer 2 Switch with One 10-Gigabit Ethernet/HiGig Port
	BCM56117 A0	24-Port Fast Ethernet and 2-Port Gigabit Ethernet Layer 2 Switch with Two 10-Gigabit Ethernet/HiGig Ports
BCM56130	BCM56132 A0	24-Port Fast Ethernet Multilayer Switch with Two 10-GbE/HiGig2 and Two 1G/2.5Gb Uplink Ports
	BCM56132 B0	
	BCM56132 B1	
	BCM56134 A0	24-Port Fast Ethernet Multilayer Switch with four 1G/2.5Gb Uplink Ports
	BCM56134 B0	
	BCM56134 B1	
BCM56140	BCM56140 A0	24-Port Gigabit Ethernet/6-Port SGMII GbE Multilayer switch with combination of two/four 1G/2.5/HiGig2 Uplink Ports
	BCM56142 A0	24-Port Gigabit Ethernet Multilayer switch with combination of two/four 1G/2.5/HiGig2 Uplink Ports
	BCM56143 A0	24-Port Gigabit Ethernet Multilayer switch with combination of two/four 1G/2.5/HiGig2 Uplink Ports

Table 122: Switch Devices

Family	Devices	Description
	BCM56144 A0	16-Port Gigabit Ethernet Multilayer switch with four 1G/2.5HG Uplink Ports
	BCM56146 A0	24-Port Fast-Ethernet Multilayer switch with four 2.5HG Uplink Ports
	BCM56147 A0	24-Port Fast-Ethernet Multilayer switch with combination of one/two/four 1G/2.5G/10/12/13HG Uplink Ports
BCM56150	BCM56150 A0	24-port GbE Managed Switch with 4-port 10 GbE uplinks, integrated CPU and 16 copper PHYs
	BCM56151 A0	24-port GbE Managed Switch with 4-port 10 GbE uplinks, integrated CPU (without PHYs)
	BCM56152 A0	24-port GbE plus 2-port GbE and 2-port 1GbE/13GbE uplinks Managed Switch, integrated CPU and 16 copper PHYs
	BCM53342 A0	8-port GbE Multilayer WebSmart Switch with Integrated CPU and Copper PHYs
	BCM53343 A0	16-port GbE plus 4-port GbE uplinks Multilayer WebSmart Switch with Integrated CPU and 16 Copper PHYs
	BCM53344 A0	24-port GbE plus 2-port GbE and 2-port 1GbE/13GbE uplinks WebSmart Switch, integrated CPU and 16 copper PHYs
	BCM53346 A0	24-port GbE Multilayer WebSmart Switch with 4-port 10 GbE uplinks, integrated CPU and 16 copper PHYs
	BCM53347 A0	24-port GbE Multilayer WebSmart Switch with 6xQSGMII + 4x1/10G
	BCM53393 A0	14-port GbE Multilayer Embedded Switch with integrated CPU (without PHY)
	BCM53394 A0	10-port GbE Multilayer Embedded Switch with 4-port 10 GbE uplinks, integrated CPU (without PHY)
BCM56210	BCM56212 A0	
	BCM56212 A1	
	BCM56212 A2	
	BCM56213 A0	
	BCM56213 A1	
	BCM56213 A2	
	BCM56214 A0	BCM56214 Integrated Multilayer Switch and CPU
	BCM56214 A1	
	BCM56214 A2	
	BCM56215 A0	
	BCM56215 A1	
	BCM56215 A2	
	BCM56216 A0	BCM56216 Integrated Multilayer Switch and CPU
	BCM56216 A1	
	BCM56216 A2	
	BCM56217 A0	
	BCM56217 A1	
	BCM56217 A2	
	BCM56218 A0	BCM56218 Integrated Multilayer Switch and CPU
	BCM56218 A1	
	BCM56218 A2	
	BCM56219 A0	BCM56219 Integrated Multilayer Switch and CPU
	BCM56219 A1	

Table 122: Switch Devices

Family	Devices	Description
	BCM56219 A2	
BCM56220	BCM56224 A0	24 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56224 B0	24 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56225 A0	24 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56225 B0	24 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56226 A0	16 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56226 B0	16 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56227 A0	16 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56227 B0	16 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56228 A0	8 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56228 B0	8 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56229 A0	8 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56229 B0	8 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56230 B1	12-Port GbE Multilayer Switch
	BCM56231 B1	6-Port GbE Multilayer Switch
	BCM56300 A0	24-Port Gigabit Ethernet Multilayer Switch
	BCM56300 A1	
	BCM56300 B0	
	BCM56300 B1	
BCM56300	BCM56301 A0	Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56301 A1	
	BCM56301 B0	
	BCM56301 B1	
	BCM56302 A0	24-Port Gigabit Ethernet Multilayer Switch with Two 10-Gigabit Ethernet/HiGig+ Ports
	BCM56302 A1	
	BCM56302 B0	
	BCM56302 B1	
	BCM56303 A0	24-Port Gigabit Ethernet Multilayer Switch with Three 10 Gigabit Ethernet/HiGig+ Ports
	BCM56303 A1	
	BCM56303 B0	
	BCM56303 B1	
	BCM56304 A0	24-Port Gigabit Ethernet Multilayer Switch with Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56304 A1	
	BCM56304 B0	
	BCM56304 B1	
BCM56305	BCM56305 A0	24-Port Gigabit Ethernet Multilayer Switch
	BCM56305 A1	
	BCM56305 B0	
	BCM56305 B1	
	BCM56306 A0	16 Port Gigabit Ethernet Switch

Table 122: Switch Devices

Family	Devices	Description
	BCM56306 A1	
	BCM56306 B0	
	BCM56306 B1	
	BCM56307 A0	24-Port GE L2 Switch with Two 10 GE/HiGig+ Ports
	BCM56307 A1	
	BCM56307 B0	
	BCM56307 B1	
	BCM56308 A0	24-Port GE L2 Switch with Three 10 GE/HiGig+ Ports
	BCM56308 A1	
	BCM56308 B0	
	BCM56308 B1	
	BCM56309 A0	24-Port GE L2 Switch with Four 10 GE/HiGig+ Ports
	BCM56309 A1	
	BCM56309 B0	
	BCM56309 B1	
BCM56310	BCM56310 A0	BCM56310 Series 24-Port GbE Multilayer Switch with Four 10-GbE/HiGig+ Uplink Ports
	BCM56311 A0	Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56312 A0	24-Port Gigabit Ethernet Multilayer Switch with Two 10-Gigabit Ethernet/HiGig+ Ports
	BCM56313 A0	24-Port Gigabit Ethernet Multilayer Switch with Three 10-Gigabit Ethernet/HiGig+ Ports
	BCM56314 A0	24-Port Gigabit Ethernet Multilayer Switch with Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56315 A0	BCM56310 Series 24-Port GbE Layer 2 Switch with Four 10-GbE/HiGig+ Uplink Ports
	BCM56316 A0	Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56317 A0	24-Port Gigabit Ethernet Layer 2 Switch with Two 10-Gigabit Ethernet/HiGig+ Ports
	BCM56318 A0	24-Port Gigabit Ethernet Layer 2 Switch with Three 10-Gigabit Ethernet/HiGig+ Ports
	BCM56319 A0	24-Port Gigabit Ethernet Layer 2 Switch with Four 10-Gigabit Ethernet/HiGig+ Ports
BCM56320	BCM56320 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56320 B0	
	BCM56320 B1	
	BCM56321 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56321 B0	
	BCM56321 B1	
BCM56330	BCM56331 A0	24-Port GbE Multilayer Switch with Four 2.5GbE Uplink Ports
	BCM56331 B0	
	BCM56331 B1	
	BCM56333 A0	16-Port GbE Multilayer Switch
	BCM56333 B0	
	BCM56333 B1	

Table 122: Switch Devices

Family	Devices	Description
	BCM56334 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56334 B0	
	BCM56334 B1	
	BCM56338 A0	8-Port GbE Multilayer Switch with two 10-GbE/HiGig2 Uplink Ports
	BCM56338 B0	
	BCM56338 B1	
BCM56340	BCM56040 A0	1xF.QSGMII + 3xF.HG[42] + 1GE
	BCM56041 A0	Ranger device, meant for embedded connectivity supports 1Ge (port 49), 2 X GE (iPROC), Flex 4x10G, 3 X 4 X 10G
	BCM56042 A0	12x2.5GE/1GE + 12x2.5GE/1GE + 1GE
	BCM56340 A0	12xF.QSGMII + Flex[4x10] + 2xHG[21] + 1GE, 12xF.QSGMII + 4xSGMII + 2xXFI + 2xHGd[21] + 1GE
	BCM56342 A0	7xF.QSGMII + Flex[4x10] + 2xHG[21] + 1GE
	BCM56344 A0	10xF.QSGMII + 3xFlex[4x10] + 1GE
BCM56340	BCM56547 A0	10xF.QSGMII + 3xF.HG[42] + 1GE, 12xF.QSGMII + 2xF.HG[42] + 1GE, 12xF.QSGMII + F.HG[42] + 2xHG[42] + 1GE
BCM56340	BCM56548 A0	7xF.QSGMII + 3xF.HG[42] + 1GE
BCM56440	BCM56440 A0	24-Port GbE Multilayer Switch with Four 10-GbE/Hig2 Uplink ports
	BCM56440 B0	
	BCM56441 A0	8-Port GbE Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
	BCM56441 B0	
	BCM56442 A0	16-Port GbE Multilayer Switch
	BCM56442 B0	
	BCM56443 A0	8-Port 2.5GbE Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
	BCM56443 B0	
	BCM56445 A0	24-Port GbE Multilayer Switch with Four 10-GbE/Hig2 Uplink ports pin compatible with BCM56334
	BCM56445 B0	
	BCM56446 A0	8-Port GbE Multilayer Switch with Two 10-GbE/Hig2 Uplink ports pin compatible with BCM56338
	BCM56447 A0	16-Port GbE Multilayer Switch pin compatible with BCM56333
	BCM56447 B0	
	BCM56448 A0	24-Port GbE Multilayer Switch with Four 1GbE/ One 2.5G Uplink ports
	BCM56448 B0	
BCM56450	BCM56248L B0	11xGE + 8x2.5G
BCM56450	BCM56450 A0	24-port GbE Multilayer Switch with 4-port 10 GbE uplinks, stacking, integrated CPU and Traffic Manager
BCM56450	BCM56450 B0	Katana2 Access 1 x XAUI + 8 x GE without L3 routing and MPLS features
BCM56450	BCM55450 B0	KT2 Access-8 FX + 2 F-HG
BCM56450	BCM56450 B1	24-port GbE Multilayer Switch with 4-port 10 GbE uplinks, stacking, integrated CPU and Traffic Manager
	BCM56455 A0	2 x 20GE (G.INT) + 2 x HG13
	BCM56455 B0	2 x 20GE (G.INT) + 2 x HG13
BCM56450	BCM56456 A0	24x GE + 4x F.XAUI
BCM56450	BCM56456 B0	24x GE + 4x F.XAUI
	BCM56456 B0	1 x XAUI + 8 x GE

Table 122: Switch Devices

Family	Devices	Description
BCM56456	BCM56456 B0	9xFXAUI + 1 x XAUI + 1x2.5GbE
BCM56456	BCM56458 B0	8xGE + 2xF.XAUI
BCM56500	BCM56500 A0	24-Port Gigabit Ethernet Multilayer Switch
	BCM56500 A1	
	BCM56500 B0	
	BCM56500 B1	
	BCM56500 B2	
	BCM56501 A0	Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56501 A1	
	BCM56501 B0	
	BCM56501 B1	
	BCM56501 B2	
	BCM56502 A0	24-Port GbE Multilayer Switch with Two 10-GbE/HiGig+ Ports
	BCM56502 A1	
	BCM56502 B0	
	BCM56502 B1	
	BCM56502 B2	
	BCM56503 A0	24-Port GbE Multilayer Switch with Three 10-GbE/HiGig+ Ports
	BCM56503 A1	
	BCM56503 B0	
	BCM56503 B1	
	BCM56503 B2	
	BCM56504 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig+ Ports
	BCM56504 A1	
	BCM56504 B0	
	BCM56504 B1	
	BCM56504 B2	
	BCM56505 A0	24-Port GbE Layer 2 Switch
	BCM56505 A1	
	BCM56505 B0	
	BCM56505 B1	
	BCM56505 B2	
	BCM56506 A0	Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56506 A1	
	BCM56506 B0	
	BCM56506 B1	
	BCM56506 B2	
	BCM56507 A0	24-Port GbE Layer 2 Switch with Two 10-GbE/HiGig+ Ports
	BCM56507 A1	
	BCM56507 B0	
	BCM56507 B1	

Table 122: Switch Devices

Family	Devices	Description
	BCM56507 B2	
	BCM56508 A0	24-Port GbE Layer 2 Switch with Three 10-GbE/HiGig+ Ports
	BCM56508 A1	
	BCM56508 B0	
	BCM56508 B1	
	BCM56508 B2	
	BCM56509 A0	24-Port GbE Layer 2 Switch with Four 10-GbE/HiGig+ Ports
	BCM56509 A1	
	BCM56509 B0	
	BCM56509 B1	
	BCM56509 B2	
BCM56510	BCM56510 A0	24-Port Gigabit Ethernet Multilayer Switch
	BCM56511 A0	Four-Port 10-GbE/HiGig+ Multilayer Switch
	BCM56512 A0	24-Port GbE Multilayer Switch With Two 10-GbE/HiGig+ Ports
	BCM56513 A0	24-Port GbE Multilayer Switch With Three 10-GbE/HiGig+ Ports
	BCM56514 A0	24-Port GbE Multilayer Switch With Four 10-GbE/HiGig+ Ports
BCM56520	BCM56520 A0	24-Port GbE Multilayer Switch
	BCM56520 B0	
	BCM56522 A0	24-Port GbE Multilayer Switch with Two 10-GbE/HiGig2 Uplink Ports
	BCM56522 B0	
	BCM56524 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56524 B0	
	BCM56526 A0	28-Port GbE Multilayer Switch with Six 10-GbE/HiGig2 Uplink Ports
	BCM56526 B0	
BCM56530	BCM56534 B0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56538 B0	48-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
BCM56540	BCM56540 A1	48xGE + 2xHG[42] + 2xHG[21] + 1GE, 48xGE + 4xXFI + 2xHG[42] + 1GE, 48xGE + 8xXFI + 1GE Multilayer Ethernet Switch
	BCM56540 B0	
	BCM56541 A1	28xGE + 2xHG[42] + 2xHG[21] + 1GE, 28xGE + 4xXFI + 2xHG[42] + 1GE, 28xGE + 8xXFI + 1GE Multilayer Ethernet Switch
	BCM56541 B0	
	BCM56542 A1	28xGE + 2xF.XAUI/2x10GE + 2xF.HG[42] + 2xF.HG[21] + 1GE, 28xGE + 8xGE/8x2.5GE + 2xHG[42] + 2xHG[21] + 1GE Multilayer Ethernet Switch
	BCM56544 A1	10xF.XAUI + 4xHG[21] + 1GE, 10xF.XAUI + 4xXFI, 10xF.XAUI + 2xHG[42], 4xXAUI + 12xXFI + 1GE Multilayer Ethernet Switch
	BCM56544 B0	10xF.XAUI + 4xHG[21] + 1GE, 10xF.XAUI + 4xXFI, 10xF.XAUI + 2xHG[42], 4xXAUI + 12xXFI + 1GE Multilayer Ethernet Switch
	BCM56545 A1	48xGE + 2xHG[42] + 2xHG[21] + 1GE, 48xGE + 4xXFI + 2xHG[42] + 1GE, 48xGE + 8xXFI + 1GE Multilayer Ethernet Switch
	BCM56545 B0	48xGE + 2xHG[42] + 2xHG[21] + 1GE, 48xGE + 4xXFI + 2xHG[42] + 1GE, 48xGE + 8xXFI + 1GE, 24xGE + 4xXAUI + 2xXFI + 2xHG[12] + 1GE Multilayer Ethernet Switch
	BCM56546 A1	28xGE + 2xHG[42] + 2xHG[21] + 1GE, 28xGE + 4xXFI + 2xHG[42] + 1GE, 28xGE + 8xXFI + 1GE Multilayer Ethernet Switch

Table 122: Switch Devices

Family	Devices	Description
	BCM56546 B0	28xGE + 2xHG[42] + 2xHG[21] + 1GE, 28xGE + 4xXFI + 2xHG[42] + 1GE, 28xGE + 8xXFI + 1GE Multilayer Ethernet Switch
BCM56580	BCM56580 A0	16 x 2.5 GbE + 4 x 10 GbE Ethernet Multilayer Switch
BCM56620	BCM56620 A0	
	BCM56620 A1	
	BCM56620 B0	
	BCM56620 B1	
	BCM56620 B2	
	BCM56624 A0	49 port 1-GbE Multilayer Ethernet Switch with 4 x 10-GbE/HiGig2 Uplink ports and External Table Expansion
	BCM56624 A1	
	BCM56624 B0	
	BCM56624 B1	
	BCM56624 B2	
	BCM56626 A0	25 port 1-GbE Multilayer Ethernet Switch with 6 x 10-GbE/HiGig2 Uplink ports and External Table Expansion
	BCM56626 A1	
	BCM56626 B0	
	BCM56626 B1	
	BCM56626 B2	
	BCM56628 A0	8 port 10-GbE/HiGig2 Multilayer Ethernet Switch with External Table Expansion
	BCM56628 A1	
	BCM56628 B0	
	BCM56628 B1	
	BCM56628 B2	
	BCM56629 B0	25 port 1-GbE Multilayer Ethernet Switch with 8 x 10-GbE/HiGig2 Uplink ports and External Table Expansion
	BCM56629 B1	
	BCM56629 B2	
BCM56630	BCM56630 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56630 B0	
	BCM56634 A0	48-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56634 B0	
	BCM56636 A0	24-Port GbE + 2-Port 10-GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56636 B0	
	BCM56638 A0	4-Port 10-GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56638 B0	
	BCM56639 A0	24-Port GbE + 4-Port 10-GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56639 B0	
BCM56640	BCM56044	Ranger+ SKU - 100G + 3xF.HG[42] + 1GE
BCM56640	BCM56045 B0	3xF.40GE + 3xF.HG[42] + 1GE
	BCM56046 B0	3xF.40GE + 2xF.HG[42] + 1GE

Table 122: Switch Devices

Family	Devices	Description
	BCM56640 A1	1x100GE + 1xHG[127], 1x100GE + 4xHG[32], 1x100GE + 8xHGd[16], 3xF.HG[42] + 1xHG[127], 3xF.HG[42] + 4xHG[32], 3xF.HG[42] + 8xHGd[16], 3xF.HG[42] + 3xF.HG[42] Multilayer Ethernet Switch
	BCM56640 B0	
	BCM56643 A1	48xGE + 4xXFI + 4xHG[42] + 1GE Multilayer Ethernet Switch
	BCM56643 B0	
	BCM56644 A1	48xGE + 2xHG[25] + 2xHG[25] + 1GE Multilayer Ethernet Switch
	BCM56644 B0	
	BCM56648 A1	48xGE + 2xHG[42] + 2xHG[21] + 1GE, 48xGE + 4xXFI + 2xHG[42] + 1GE, 48xGE + 8xXFI + 1GE Multilayer Ethernet Switch
	BCM56648 B0	
	BCM56649 A1	28xGE + 2xHG[42] + 2xHG[21] + 1GE, 28xGE + 4xXFI + 2xHG[42] + 1GE, 28xGE + 8xXFI + 1GE Multilayer Ethernet Switch
	BCM56649 B0	
BCM56640	BCM56545K	Triumph 3 SKU - 48-port GE switch + 4x10GE + 4xHG[42] / 40GE
BCM56640	BCM56546K	Triumph 3 SKU - 28-port GE switch + 4x10GE + 4xHG[42] / 40GE
BCM56680	BCM56680 A0	25 port 1-GbE/2.5GbE Multilayer Ethernet Switch with 4 x 10-GbE/HiGig2 Uplink ports
	BCM56680 A1	
	BCM56680 B0	
	BCM56680 B1	
	BCM56684 A0	24 port 1-GbE/2.5GbE Multilayer Ethernet Switch with 4 x 10-GbE/HiGig2 Uplink ports
	BCM56684 A1	
	BCM56684 B0	
	BCM56684 B1	
BCM56685	BCM56685 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56685 B0	
	BCM56689 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56689 B0	
BCM56700	BCM56700 A0	16-Port, 192-Gbps Lossless Switch Fabric
	BCM56701 A0	12-Port, 144-Gbps Lossless Switch Fabric
BCM56720	BCM56720 A0	16 Port, 16-Gbps HiGig2 Switch Fabric
	BCM56721 A0	12 Port, 16-Gbps HiGig2 Switch Fabric
BCM56725	BCM56725 A0	8 Port, 20-Gbps + 4 Port, 16-Gbps HiGig2 Switch Fabric
BCM56740	BCM56743 A0	480 Gbps Switch fabric
	BCM56743 A1	
	BCM56743 A2	
	BCM56743 A3	
	BCM56743 A4	
	BCM56743 B0	
	BCM56743 B1	
	BCM56745 A0	640 Gbps Switch fabric
	BCM56745 A1	
	BCM56745 A2	

Table 122: Switch Devices

Family	Devices	Description
	BCM56745 A3	
	BCM56745 A4	
	BCM56745 B0	
	BCM56745 B1	
BCM56740_PLUS	BCM56744 A0	480 Gbps Switch fabric
	BCM56744 A1	
	BCM56746 A0	640 Gbps Switch fabric
	BCM56746 A1	
BCM56800	BCM56800 A0	20-Port 10-Gigabit Ethernet Multilayer Switch
	BCM56801 A0	10-Port 10-Gigabit Ethernet and 8-Port HiGig2/10GbE Multilayer Switch
	BCM56802 A0	16-Port 10-GbE/HiGig2 Multilayer Switch
	BCM56803 A0	12 Port 10GE/HiGig2 Multilayer Switch
BCM56820	BCM56820 A0	24 x 10-GbE + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56820 B0	
	BCM56821 A0	12 x 10-GbE + 8 x HiGig2 + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56821 B0	
	BCM56822 A0	12 x 10-GbE + 4 x 20-Gbps HiGig2 + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56822 B0	
	BCM56823 A0	8 x 10-GbE + 4 x 20-Gbps HiGig2 + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56823 B0	
	BCM56825 B0	16 x 10-GbE + 8 x 20-Gbps HiGig2 + 1 x 1-GbE Multilayer Ethernet Switch
	BCM56740	
	BCM56743 A0	480 Gbps Switch fabric
	BCM56743 A1	
BCM56740	BCM56743 A2	
	BCM56743 A3	
	BCM56743 A4	
	BCM56743 B0	
	BCM56743 B1	
	BCM56745 A0	640 Gbps Switch fabric
	BCM56745 A1	
	BCM56745 A2	
	BCM56745 A3	
	BCM56745 A4	
	BCM56745 B0	
	BCM56745 B1	
	BCM56744 A0	480 Gbps Switch fabric
	BCM56744 A1	
	BCM56746 A0	640 Gbps Switch fabric
	BCM56746 A1	
BCM56840	BCM56841 A0	320 Gbps Ethernet Multilayer Switch
	BCM56841 A1	
	BCM56841 A2	

Table 122: Switch Devices

Family	Devices	Description
	BCM56841 A3	
	BCM56841 A4	
	BCM56841 B0	
	BCM56841 B1	
	BCM56843 A0	480 Gbps Ethernet Multilayer Switch
	BCM56843 A1	
	BCM56843 A2	
	BCM56843 A3	
	BCM56843 A4	
	BCM56843 B0	
	BCM56843 B1	
	BCM56845 A0	640 Gbps Ethernet Multilayer Switch
	BCM56845 A1	
	BCM56845 A2	
	BCM56845 A3	
	BCM56845 A4	
	BCM56845 B0	
	BCM56845 B1	
BCM56840_PLUS	BCM56842 A0	320 Gbps Ethernet Multilayer Switch
	BCM56842 A1	
	BCM56844 A0	480 Gbps Ethernet Multilayer Switch
	BCM56844 A1	
	BCM56846 A0	640 Gbps Ethernet Multilayer Switch
	BCM56846 A1	
BCM56846	BCM56831	Trident+ SKU - 24-port 10GE switch with 40GE support for embedded applications
BCM56846	BCM56835C	Trident+ SKU - (64 x 10 GbE) + (4 x 1 GbE)
BCM56846	BCM56847	Trident+ SKU - (64 x 10 GbE) + (4 x 1 GbE)
BCM56846	BCM56849	Trident+ SKU - (56 x 1GbE/2.5GbE) + (8 x 10GbE)
BCM56850	BCM56751P A1	1.28Tbps I/O, 960Gbps Core Ethernet Switch Fabric
BCM56850	BCM56751P A2	1.28Tbps I/O, 960Gbps Core Ethernet Switch Fabric
BCM56850	BCM56830 A1	960Gbps Ethernet Switch
BCM56850	BCM56830 A2	960Gbps Ethernet Switch
BCM56850	BCM56834	Trident2 SKU - High density 10G and 40G switch for embedded applications
BCM56850	BCM56838	Trident2 SKU - 72/320G Devices with 1.25/3.125/6.25G Serdes and 4 SFIs
BCM56850	BCM56850 A1	1.28Tbps I/O, 1Tbps Core Ethernet Switch
	BCM56852 A2	100x10G, 960Gbps Multilayer Switch
	BCM56854 A1	
BCM56851	BCM56751 A2	1.28Tbps I/O, 960Gbps Core Ethernet Switch Fabric
BCM88732	BCM88732 B2	Eight-Port 10 GbE or 2-Port 40 GbE MAC Aggregation Switch with 80 Gbps Uplink Capacity
BCM88020	BCM88020 A0	XGS Core (XCore/SBX) Fully Programmable Carrier Packet Processor with 24 GbE Ports, 2 10GbE Ports and 2 SPI Interfaces
	BCM88020 A1	
	BCM88020 A2	

Table 122: Switch Devices

Family	Devices	Description
BCM88025	BCM88025 A0	XGS Core (XCore/SBX) Fully Programmable Carrier Packet Processor with 24 GbE Ports, 2 10GbE Ports and 2 SPI Interfaces
BCM88030	BCM88030 A0	XGS Core (XCore/SBX) Scalable Switching 100 Gbps Fully Programmable Carrier Packet Processor
BCM88130	BCM88130 A0 BCM88130 A1	XGS Core (XCore/SBX) 630 Gbps Bandwidth Manager and Switching Engine
BME-3200	BME-3200 A0 BME-3200 B0	XGS Core (XCore/SBX) Fabric Bandwidth Manager with 32 SCI control ports and up to 40 SFI data ports
QE-2000	QE-2000 A1 QE-2000 A2 QE-2000 A3 QE-2000 A4	XGS Core (XCore/SBX) Fabric Queuing Engine with 49 SPI 4.2 subports
BCM88230	BCM88230 A0 BCM88230 B0	XGS Core (XCore/SBX) Fabric Queuing Engine with Integrated Traffic Management with 4 HiGig2 ports, 50Gbps
	BCM88235 A0 BCM88235 B0	XGS Core (XCore/SBX) Fabric Queuing Engine with Integrated Traffic Management with 4 HiGig2 ports, 80Gbps
	BCM88231 A0 BCM88231 B0	XGS Core (XCore/SBX) Traffic Manager with 4 HiGig2 ports, 50Gbps
	BCM88236 A0 BCM88236 B0	XGS Core (XCore/SBX) Traffic Manager with 4 HiGig2 ports, 80Gbps
BCM56930	BCM56931 A0 BCM56931 B0 BCM56936 A0 BCM56936 B0	XGS pass-through and standalone Traffic Manager, 4 HiGig2 ports, 50Gbps XGS pass-through and standalone Traffic Manager, 4 HiGig2 ports, 80Gbps
BCM88640	BCM88640 A0 BCM88640 B0	DNX 100G Flexible Packet Processor with Integrated Traffic Management
BCM88650	BCM88650 A0 BCM88650 B0 BCM88650 B1	DNX 200G Flexible Packet Processor with Integrated Traffic Management 200 GBps DNX Traffic Manager and Packet Processor
BCM88660	BCM88660 A0	DNX 200G Flexible Packet Processor with Integrated Traffic Management
BCM88750	BCM88750 A0 BCM88750 B0	DNX 1600 GBps Switch Fabric

Table 123: SER Supported Devices

Family	Devices
Trident	56841, 56842, 56843, 56844, 56845, 56846, 56850
Triumph	56640, 56643, 56644, 56648, 56649, 56540, 56541, 56542, 56544, 56545
Katana	All SKUs
Katana2	56450, 56455, 56456
Enduro2	All SKUs



Table 123: SER Supported Devices

Family	Devices
Hurricane2	56150, 56151, 53344, 53346, 53393, 53394
Helix4	56340, 56040, 56344, 56042, 56342

Table 124: Switch Devices that support Warm boot

Family	Devices	Description
BCM5675	BCM5675 A0	8-Port, 192-Gbps Switch Fabric
	BCM5675 A1	
	BCM5676 A0	4-Port, 96-Gbps Switch Fabric
	BCM5676 A1	
BCM56020	BCM56024 A0	24-Port Integrated Multilayer Switch and CPU
	BCM56024 B0	
	BCM56025 A0	24-Port Integrated L2 Switch and CPU
	BCM56025 B0	
BCM56026	BCM56026 A0	24-Port Integrated L2 Switch and CPU
	BCM56026 B0	
BCM56130	BCM56132 A0	24-Port Fast Ethernet Multilayer Switch with Two 10-GbE/HiGig2 and Two 1G/2.5Gb Uplink Ports
	BCM56132 B0	
	BCM56132 B1	
	BCM56134 A0	24-Port Fast Ethernet Multilayer Switch with four 1G/2.5Gb Uplink Ports
BCM56134	BCM56134 B0	
	BCM56134 B1	
BCM56142	BCM56142 A0	24-Port Fast Ethernet Multilayer Switch with four 1G/2.5Gb/Higig2/HG Lite Uplink Ports
BCM56150	BCM56150 A0	24-port GbE Managed Switch with 4-port 10 GbE uplinks, integrated CPU and 16 copper PHYs
	BCM56151 A0	24-port GbE Managed Switch with 4-port 10 GbE uplinks, integrated CPU (without PHYs)
	BCM56152 A0	24-port GbE plus 2-port GbE and 2-port 1GbE/13GbE uplinks Managed Switch, integrated CPU and 16 copper PHYs
	BCM53342 A0	8-port GbE Multilayer WebSmart Switch with Integrated CPU and Copper PHYs
	BCM53343 A0	16-port GbE plus 4-port GbE uplinks Multilayer WebSmart Switch with Integrated CPU and 16 Copper PHYs
	BCM53344 A0	24-port GbE plus 2-port GbE and 2-port 1GbE/13GbE uplinks WebSmart Switch, integrated CPU and 16 copper PHYs
	BCM53346 A0	24-port GbE Multilayer WebSmart Switch with 4-port 10 GbE uplinks, integrated CPU and 16 copper PHYs
	BCM53393 A0	14-port GbE Multilayer Embedded Switch with integrated CPU (without PHY)
	BCM53394 A0	10-port GbE Multilayer Embedded Switch with 4-port 10 GbE uplinks, integrated CPU (without PHY)
BCM56220	BCM56224 A0	24 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56224 B0	24 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56225 A0	24 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56225 B0	24 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56226 A0	16 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+



Table 124: Switch Devices that support Warm boot

Family	Devices	Description
	BCM56226 B0	16 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56227 A0	16 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56227 B0	16 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56228 A0	8 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56228 B0	8 GbE + 4 x 1 Gb/2.5 Gb, L3/L2+
	BCM56229 A0	8 GbE + 4 x 1 Gb/2.5 Gb, L2+
	BCM56229 B0	8 GbE + 4 x 1 Gb/2.5 Gb, L2+
BCM56230	BCM56230 B1	12-Port GbE Multilayer Switch
	BCM56231 B1	6-Port GbE Multilayer Switch
BCM56240	BCM56240 A0	2-Port 10GbE (OR 8 *2.5GbE) Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
BCM56240	BCM56240 B0	2-Port 10GbE (OR 8 *2.5GbE) Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
	BCM56241 A0	6-Port GbE Multilayer Switch with Two 2.5GbE Uplink ports
	BCM56242 A0	8-Port 2.5GbE Multilayer Switch with Two 2.5GbE Uplink ports
	BCM56243 A0	4-Port 2.5GbE Multilayer Switch
BCM56320	BCM56320 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56320 B0	
	BCM56320 B1	
	BCM56321 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56321 B0	
	BCM56321 B1	
BCM56330	BCM56331 A0	24-Port GbE Multilayer Switch with Four 2.5GbE Uplink Ports
	BCM56331 B0	
	BCM56331 B1	
	BCM56333 A0	16-Port GbE Multilayer Switch
	BCM56333 B0	
	BCM56333 B1	
	BCM56334 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56334 B0	
	BCM56334 B1	
	BCM56338 A0	8-Port GbE Multilayer Switch with two 10-GbE/HiGig2 Uplink Ports
	BCM56338 B0	
	BCM56338 B1	
BCM56340	BCM56340 A0	12xF.QSGMII + Flex[4x10] + 2xHG[21] + 1GE, 12xF.QSGMII + 4xSGMII + 2xXFI + 2xHGd[21] + 1GE
	BCM56342 A0	7xF.QSGMII + Flex[4x10] + 2xHG[21] + 1GE
	BCM56344 A0	10xF.QSGMII + 3xFlex[4x10] + 1GE
	BCM56040 A0	1xF.QSGMII + 3xF.HG[42] + 1GE
	BCM56041 A0	Ranger device, meant for embedded connectivity supports 1Ge (port 49), 2 X GE (iPROC), Flex 4x10G, 3 X 4 X 10G
	BCM560547 A0	10xF.QSGMII + 3xF.HG[42] + 1GE, 12xF.QSGMII + 2xF.HG[42] + 1GE, 12xF.QSGMII + F.HG[42] + 2xHG[42] + 1GE
	BCM560548 A0	7xF.QSGMII + 3xF.HG[42] + 1GE

Table 124: Switch Devices that support Warm boot

Family	Devices	Description
BCM56240	BCM56240 A0	2-Port 10GbE (OR 8 *2.5GbE) Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
BCM56240	BCM56240 B0	2-Port 10GbE (OR 8 *2.5GbE) Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
	BCM56241 A0	6-Port GbE Multilayer Switch with Two 2.5GbE Uplink ports
	BCM56242 A0	8-Port 2.5GbE Multilayer Switch with Two 2.5GbE Uplink ports
	BCM56243 A0	4-Port 2.5GbE Multilayer Switch
BCM56440	BCM55441 A0	24-Port GbE Multilayer Switch with Four 10-GbE/Hig2 Uplink ports
	BCM56440 A0	24-Port GbE Multilayer Switch with Four 10-GbE/Hig2 Uplink ports
	BCM56440 B0	24-Port GbE Multilayer Switch with Four 10-GbE/Hig2 Uplink ports
	BCM56441 A0	8-Port GbE Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
	BCM56442 A0	16-Port GbE Multilayer Switch
	BCM56443 A0	8-Port 2.5GbE Multilayer Switch with Two 10-GbE/Hig2 Uplink ports
	BCM56445 A0	24-Port GbE Multilayer Switch with Four 10-GbE/Hig2 Uplink ports pin compatible with BCM56334
	BCM56446 A0	8-Port GbE Multilayer Switch with Two 10-GbE/Hig2 Uplink ports pin compatible with BCM56338
	BCM56447 A0	16-Port GbE Multilayer Switch pin compatible with BCM56333
	BCM56448 A0	24-Port GbE Multilayer Switch with Four 1GbE/ One 2.5G Uplink ports
BCM56450	BCM56450 A0	24-port GbE Multilayer Switch with 4-port 10 GbE uplinks, stacking, integrated CPU and Traffic Manager
	BCM56455 A0	2 x 20GE (G.INT) + 2 x HG13
	BCM56456 B0	1 x XAUI + 8 x GE
BCM56500	BCM56500 A0	24-Port Gigabit Ethernet Multilayer Switch
	BCM56500 A1	
	BCM56500 B0	
	BCM56500 B1	
	BCM56500 B2	
	BCM56501 A0	Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56501 A1	
	BCM56501 B0	
	BCM56501 B1	
	BCM56501 B2	
	BCM56502 A0	24-Port GbE Multilayer Switch with Two 10-GbE/HiGig+ Ports
	BCM56502 A1	
	BCM56502 B0	
	BCM56502 B1	
	BCM56502 B2	
	BCM56503 A0	24-Port GbE Multilayer Switch with Three 10-GbE/HiGig+ Ports
	BCM56503 A1	
	BCM56503 B0	
	BCM56503 B1	
	BCM56503 B2	
	BCM56504 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig+ Ports

Table 124: Switch Devices that support Warm boot

Family	Devices	Description
	BCM56504 A1	
	BCM56504 B0	
	BCM56504 B1	
	BCM56504 B2	
	BCM56505 A0	24-Port GbE Layer 2 Switch
	BCM56505 A1	
	BCM56505 B0	
	BCM56505 B1	
	BCM56505 B2	
	BCM56506 A0	Four 10-Gigabit Ethernet/HiGig+ Ports
	BCM56506 A1	
	BCM56506 B0	
	BCM56506 B1	
	BCM56506 B2	
	BCM56507 A0	24-Port GbE Layer 2 Switch with Two 10-GbE/HiGig+ Ports
	BCM56507 A1	
	BCM56507 B0	
	BCM56507 B1	
	BCM56507 B2	
	BCM56508 A0	24-Port GbE Layer 2 Switch with Three 10-GbE/HiGig+ Ports
	BCM56508 A1	
	BCM56508 B0	
	BCM56508 B1	
	BCM56508 B2	
	BCM56509 A0	24-Port GbE Layer 2 Switch with Four 10-GbE/HiGig+ Ports
	BCM56509 A1	
	BCM56509 B0	
	BCM56509 B1	
	BCM56509 B2	
BCM56510	BCM56510 A0	24-Port Gigabit Ethernet Multilayer Switch
	BCM56511 A0	Four-Port 10-GbE/HiGig+ Multilayer Switch
	BCM56512 A0	24-Port GbE Multilayer Switch With Two 10-GbE/HiGig+ Ports
	BCM56513 A0	24-Port GbE Multilayer Switch With Three 10-GbE/HiGig+ Ports
	BCM56514 A0	24-Port GbE Multilayer Switch With Four 10-GbE/HiGig+ Ports
BCM56520	BCM56520 A0	24-Port GbE Multilayer Switch
	BCM56520 B0	
	BCM56522 A0	24-Port GbE Multilayer Switch with Two 10-GbE/HiGig2 Uplink Ports
	BCM56522 B0	
	BCM56524 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56524 B0	
	BCM56526 A0	28-Port GbE Multilayer Switch with Six 10-GbE/HiGig2 Uplink Ports

Table 124: Switch Devices that support Warm boot

Family	Devices	Description
	BCM56526 B0	
BCM56530	BCM56534 B0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56538 B0	48-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
BCM56620	BCM56620 A0	
	BCM56620 A1	
	BCM56620 B0	
	BCM56620 B1	
	BCM56624 A0	49 port 1-GbE Multilayer Ethernet Switch with 4 x 10-GbE/HiGig2 Uplink ports and External Table Expansion
	BCM56624 A1	
	BCM56624 B0	
	BCM56624 B1	
	BCM56624 B2	
	BCM56626 A0	25 port 1-GbE Multilayer Ethernet Switch with 6 x 10-GbE/HiGig2 Uplink ports and External Table Expansion
	BCM56626 A1	
	BCM56626 B0	
	BCM56626 B1	
	BCM56626 B2	
	BCM56628 A0	8 port 10-GbE/HiGig2 Multilayer Ethernet Switch with External Table Expansion
	BCM56628 A1	
	BCM56628 B0	
	BCM56628 B1	
	BCM56628 B2	
	BCM56629 B0	25 port 1-GbE Multilayer Ethernet Switch with 8 x 10-GbE/HiGig2 Uplink ports and External Table Expansion
	BCM56629 B1	
BCM56630	BCM56630 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56630 B0	
	BCM56634 A0	48-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56634 B0	
	BCM56636 A0	24-Port GbE + 2-Port 10-GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56636 B0	
	BCM56638 A0	4-Port 10-GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56638 B0	
	BCM56639 A0	24-Port GbE + 4-Port 10-GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56639 B0	
BCM56540	BCM56540 A0	48xGE + 2xHG[42] + 2xHG[21] + 1GE, 48xGE + 4xXFI + 2xHG[42] + 1GE, 48xGE + 8xXFI + 1GE Multilayer Ethernet Switch (Preview)
	BCM56540 A1	
BCM56540	BCM56541 A0	28xGE + 2xHG[42] + 2xHG[21] + 1GE, 28xGE + 4xXFI + 2xHG[42] + 1GE, 28xGE + 8xXFI + 1GE Multilayer Ethernet Switch (Preview)

Table 124: Switch Devices that support Warm boot

Family	Devices	Description
	BCM56541 A1	
BCM56540	BCM56542 A0	28xGE + 2xF.XAUI/2x10GE + 2xF.HG[42] + 2xF.HG[21] + 1GE, 28xGE + 8xGE/8x2.5GE + 2xHG[42] + 2xHG[21] + 1GE Multilayer Ethernet Switch (Preview)
	BCM56542 A1	
BCM56540	BCM56544 A0	10xF.XAUI + 4xHG[21] + 1GE, 10xF.XAUI + 4xXFI, 10xF.XAUI + 2xHG[42], 4xXAUI + 12xXFI + 1GE Multilayer Ethernet Switch (Preview)
	BCM56544 A1	
BCM56540	BCM56545 A0	48xGE + 2xHG[42] + 2xHG[21] + 1GE, 48xGE + 4xXFI + 2xHG[42] + 1GE, 48xGE + 8xXFI + 1GE Multilayer Ethernet Switch (Preview)
	BCM56545 A1	
BCM56540	BCM56546 A0	28xGE + 2xHG[42] + 2xHG[21] + 1GE, 28xGE + 4xXFI + 2xHG[42] + 1GE, 28xGE + 8xXFI + 1GE Multilayer Ethernet Switch (Preview)
BCM56640	BCM56640 A0	1x100GE + 1xHG[127], 1x100GE + 4xHG[32], 1x100GE + 8xHGd[16], 3xF.HG[42] + 1xHG[127], 3xF.HG[42] + 4xHG[32], 3xF.HG[42] + 8xHGd[16], 3xF.HG[42] + 3xF.HG[42] Multilayer Ethernet Switch (Preview)
	BCM56640 A1	
BCM56640	BCM56643 A0	48xGE + 4xXFI + 4xHG[42] + 1GE Multilayer Ethernet Switch (Preview)
	BCM56643 A1	
BCM56640	BCM56644 A0	48xGE + 2xHG[25] + 2xHG[25] + 1GE Multilayer Ethernet Switch (Preview)
	BCM56644 A1	
BCM56640	BCM56648 A0	48xGE + 2xHG[42] + 2xHG[21] + 1GE, 48xGE + 4xXFI + 2xHG[42] + 1GE, 48xGE + 8xXFI + 1GE Multilayer Ethernet Switch (Preview)
	BCM56648 A1	
BCM56640	BCM56649 A0	28xGE + 2xHG[42] + 2xHG[21] + 1GE, 28xGE + 4xXFI + 2xHG[42] + 1GE, 28xGE + 8xXFI + 1GE Multilayer Ethernet Switch (Preview)
BCM56680	BCM56680 A0	25 port 1-GbE/2.5GbE Multilayer Ethernet Switch with 4 x 10-GbE/HiGig2 Uplink ports
	BCM56680 A1	
	BCM56680 B0	
	BCM56680 B1	
	BCM56684 A0	24 port 1-GbE/2.5GbE Multilayer Ethernet Switch with 4 x 10-GbE/HiGig2 Uplink ports
	BCM56684 A1	
	BCM56684 B0	
	BCM56684 B1	
BCM56685	BCM56685 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56685 B0	
	BCM56689 A0	24-Port GbE Multilayer Switch with Four 10-GbE/HiGig2 Uplink Ports
	BCM56689 B0	
BCM56700	BCM56700 A0	16-Port, 192-Gbps Lossless Switch Fabric
	BCM56701 A0	12-Port, 144-Gbps Lossless Switch Fabric
BCM56720	BCM56720 A0	16 Port, 16-Gbps HiGig2 Switch Fabric
	BCM56721 A0	12 Port, 16-Gbps HiGig2 Switch Fabric
BCM56725	BCM56725 A0	8 Port, 20-Gbps + 4 Port, 16-Gbps HiGig2 Switch Fabric
BCM56800	BCM56800 A0	20-Port 10-Gigabit Ethernet Multilayer Switch
	BCM56801 A0	10-Port 10-Gigabit Ethernet and 8-Port HiGig2/10GbE Multilayer Switch
	BCM56802 A0	16-Port 10-GbE/HiGig2 Multilayer Switch

Table 124: Switch Devices that support Warm boot

Family	Devices	Description
BCM56820	BCM56803 A0	12 Port 10GE/HiGig2 Multilayer Switch
	BCM56820 A0	24 x 10-GbE + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56820 B0	
	BCM56821 A0	12 x 10-GbE + 8 x HiGig2 + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56821 B0	
	BCM56822 A0	12 x 10-GbE + 4 x 20-Gbps HiGig2 + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56822 B0	
	BCM56823 A0	8 x 10-GbE + 4 x 20-Gbps HiGig2 + 4 x 1-GbE Multilayer Ethernet Switch
	BCM56823 B0	
	BCM56825 B0	16 x 10-GbE + 8 x 20-Gbps HiGig2 + 1 x 1-GbE Multilayer Ethernet Switch
BCM56840	BCM56841 A0	320 Gbps Ethernet Multilayer Switch
	BCM56841 A1	
	BCM56841 A2	
	BCM56841 A3	
	BCM56841 A4	
	BCM56841 B0	
	BCM56841 B1	
	BCM56843 A0	480 Gbps Ethernet Multilayer Switch
	BCM56843 A1	
	BCM56843 A2	
BCM56840_PLUS	BCM56843 A3	
	BCM56843 A4	
	BCM56843 B0	
	BCM56843 B1	
	BCM56845 A0	640 Gbps Ethernet Multilayer Switch
	BCM56845 A1	
	BCM56845 A2	
	BCM56845 A3	
	BCM56845 A4	
	BCM56845 B0	
BCM56850	BCM56845 B1	
	BCM56842 A0	320 Gbps Ethernet Multilayer Switch
	BCM56842 A1	
	BCM56844 A0	480 Gbps Ethernet Multilayer Switch
	BCM56844 A1	
	BCM56846 A0	640 Gbps Ethernet Multilayer Switch
	BCM56846 A1	
	BCM56850 A0	1.28Tbps I/O, 1Tbps Core Ethernet Switch
	BCM56854 A0	1.28Tbps I/O, 1Tbps Core Ethernet Switch
	BCM56850 A1	1.28Tbps I/O, 1Tbps Core Ethernet Switch
BCM88640	BCM56854 A1	1.28Tbps I/O, 1Tbps Core Ethernet Switch
	BCM88640 A0	80GBps DNX Traffic manager + Packet processor

Table 124: Switch Devices that support Warm boot

Family	Devices	Description
	BCM88640 B0	
BCM88650	BCM88650 A0	200GBps DNX Traffic manager + Packet processor
	BCM88650 B0	
	BCM88650 B1	
BCM88660	BCM88660 A0	200GBps DNX Traffic manager + Packet processor
BCM88750	BCM88750 A0	1600GBps DNX Switch fabric
	BCM88750 B0	

Note: There is no warm boot support for External table expansion in BCM56620, BCM56630 and BCM56640 device family.

PHYS

Table 125: PHYs

Device	Driver Family	Description
BCM5218	522x	10/100Base-TX/FX Octal-PHY(tm) Transceiver
BCM5220	522x	10/100BASE-TX/FX Mini-F(tm) Transceiver
BCM5221	522x	10/100BASE-TX/FX Mini-F(tm) Transceiver
BCM5226	522x	10/100 BASE- TX/FX Hex-PHY(tm) Transceiver
BCM5228	522x	10/100BASE-TX/FX Octal-F(tm) Transceiver
BCM5238	522x	10/100BASE-TX OCTAL-f(tm) Transceiver
BCM5248	522x	10/100BASE-TX Octal-F(tm) Transceiver
BCM52681E A1	54680	Octal 10/100 Ethernet Transceiver
BCM5401	5401	10/100/1000BASE-T Gigabit Copper Transceiver
BCM5402	5402	10/100/1000BASE-T Gigabit Copper Transceiver
BCM5404	5404	Quad-Port 10/100/1000BASE-T Gigabit Copper Transceiver
BCM5424	5424	Quad 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM5434	5424	Quad 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM5411	5411	10/100/1000BASE-T Gigabit Copper Transceiver
BCM5421	5421S	10/100/1000BASE-T Gigabit Copper Transceiver
BCM5421S	5421S	10/100/1000BASE-T Gigabit Copper Transceiver with SerDes
BCM5461	5464	10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM5464	5464	Quad-Port 10/100/1000BASE-T Gigabit Copper Transceiver
BCM5464R	5464	Quad-Port 10/100/1000BASE-T Gigabit Copper Transceiver
BCM5464S	5464	Quad-Port Gigabit Copper Transceiver with Copper/Fiber Media Interface
BCM5464SR	5464	Quad-Port Gigabit Copper Transceiver with Copper/Fiber Media Interface
BCM5466	5464	Quad-Port 10/100/1000BASE-T Gigabit Copper Transceiver
BCM5466R	5464	Quad-Port 10/100/1000BASE-T Gigabit Copper Transceiver
BCM5466S	5464	Quad-Port Gigabit Copper Transceiver with Copper/Fiber Media Interface
BCM5466SR	5464	Quad-Port Gigabit Copper Transceiver with Copper/Fiber Media Interface
BCM5482	5482	Dual-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM5488	5464	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54210SE_B0	54210	Single Copper/Fiber Gigabit Ethernet Transceiver. 1588 not yet supported
BCM54220SE_B0	54220	Dual Copper/Fiber Gigabit Ethernet Transceiver. 1588 not yet supported
BCM54240_C0	54280	Quad 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54240_C1	54280	Quad 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54280_A0	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54280_C0	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54280_C1	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54282_A0	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54282_C0	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54282_C1	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54285_C0	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54285_C1	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54290_A0	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (1588 feature is Bring-up)
BCM54292_A0	54280	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (1588 feature is Bring-up)
BCM54294_A0	54280	Quad 1000/100/10BASE-T Gigabit Ethernet Transceiver (1588 feature is Bring-up)



Table 125: PHYs

Device	Driver Family	Description
BCM54295SE_B0	54295	Octal Copper/Quad Fiber Gigabit Ethernet Transceiver -1588 not yet supported
BCM54296SE_B0	54296	Quad Copper/Fiber Gigabit Ethernet Transceiver - 1588 not yet supported
BCM54340_B0	54380	Quad 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54340_C0	54380	Quad 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54340_C1	54380	Quad 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54380_B0	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54380_C0	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54380_C1	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54382_B0	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54382_C0	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54382_C1	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54385_B0	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54385_C0	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54385_C1	54380	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver (Needs additional software component)
BCM54616_A0	54616	Single-Chip 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54640	54640	Quad-Port Gigabit Copper Transceiver with Copper/Fiber Media Interface
BCM54640E_A1	54640	Quad-Port Gigabit Copper Transceiver with Copper/Fiber Media Interface
BCM54640E_B0	54640	Quad-Port Gigabit Copper Transceiver with Copper/Fiber Media Interface
BCM54680_A0	54680	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54680E_A1	54680	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54680E_B0	54680	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54682E_A1	54682	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver with 2 Copper/Fiber Media Interface
BCM54682E_B0	54682	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver with 2 Copper/Fiber Media Interface
BCM54684_D0	54684	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54684E_B0	54682	10/100/1000 Octal (65nm) QSGMII-Copper/Fiber(2) with EEE
BCM54685	54682	Octal QSGMII to 10/100/1000BaseT or Fiber Ethernet Transceiver
BCM54685E_A1	54682	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver with Copper/Fiber Media Interface
BCM54810_A0	54880	BroadR-Reach Single-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54880_A0	54880	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver with BroadR-Reach support
BCM54880_B0	54880	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver with BroadR-Reach support
BCM54880E_A1	54680	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54880E_B0	54680	Octal-Port 10/100/1000BASE-T Gigabit Ethernet Transceiver
BCM54881_B0	54880	Octal 10/100Base/Tx Ethernet BroadReach Transceiver
BCM54942_A0	84728	Quad-Channel 10GbE XAUI-to-XFI PHY. Firmware version 0124
BCM54980_B2	54980	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM54980_C0	54980	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver



Table 125: PHYs

Device	Driver Family	Description
BCM54980_C1	54980	Octal 1000/100/10BASE-T Gigabit Ethernet Transceiver
BCM8040_A2	8040	Eight-Channel Multirate 1-Gbps - 3.2-Gbps Retimer/Switch
BCM8073_A0	8072	Dual-Channel Serial 10-GbE BASE-KR to XAUI Transceiver. Firmware version d502.
BCM8074_A0	8072	Quad-Channel Serial 10-GbE BASE-KR to XAUI Transceiver. Firmware version 010C.
BCM8704	8703	Serial 10-Gigabit Ethernet/Fibre Channel Transceiver with XAUI Interface
BCM8705	8705	Serial 10-Gigabit Ethernet/Fibre Channel Transceiver with WIS Layer and XAUI Interface
BCM8725	8705	Dual Serial 10-Gigabit Ethernet/Fibre Channel Transceiver with WIS Layer and XAUI Interface
BCM8726_A0	8706	Dual Serial 10-Gigabit Ethernet/Fibre Channel Transceiver with XAUI Interface
BCM8726_B1	8706	Dual Serial 10-Gigabit Ethernet/Fibre Channel Transceiver with XAUI(TM) Interface. Firmware version 0x0127
BCM8727_B0	8706	Dual Serial 10-Gigabit Ethernet/Fibre Channel Transceiver with XAUI Interface. Firmware version 0406.
BCM8727_C0	8706	Dual Serial 10-Gigabit Ethernet/Fibre Channel Transceiver with XAUI Interface. Firmware version 050D.
BCM84727_A0	84728	Dual SFI to XAUI with 1588 (Firmware version 0x124. Bring-up)
BCM8728_A0	8706	Dual-Channel 10-GbE SFI-to-XAUI(TM) Transceiver with EDC. Firmware version 0511. (Bring-up)
BCM8742	8706	Quad-Channel 10-GbE SFI-to-XAUI(TM) Transceiver. Firmware version 0511.
BCM8747_A0	8706	Quad-Channel 10-GbE SFI-to-XAUI(TM) Transceiver with EDC. Firmware version 0511.
BCM8750_A0	8750	Dual-Channel 10 GbE SFI-to-XFI PHY with EDC
BCM8752_A0	8750	Dual-Channel 10 GbE SFI-to-XFI PHY with EDC
BCM8754_A0	8750	Quad-Channel 10 GbE SFI-to-XFI PHY with EDC. Firmware version 0411.
BCM8481_B0	8481	10GBASE-T Transceiver (Firmware version B0 02.10)
BCM8481_C0	8481	10GBASE-T Transceiver (Firmware version C0 02.13)
BCM84164	BCM84740	Quad 10GBASE-KR-to-XFI or 40GBASE-KR4-to-XLAUI Transceiver Firmware version 0x128
BCM84168	BCM84740	Octal 10GBASE-KR-to-XFI or Dual 40GBASE-KR4-to-XLAUI Transceiver Firmware version 0x128
BCM84318_A0	84740	10.3 Gbps Octal Port CDR/Retimer with EDC. Firmware version D007
BCM82072	82072	CAUI4-TO-KR4 NRZ BACKPLANE PHY/OCTAL 25G BACKPLANE RETIMER.
BCM82322_B0	82328	12port Gallardo28 supporting 12x10G, 3x40G, 1x100G. Firmware version D
BCM82328_A0	82328	Dual 40 GbE/Octal 10 GbE QSFP+ XLPPI-to-XLAUI PHY. Firmware version 9 "(Bring-up)
BCM82328_B0	82328	Dual 40 GbE/Octal 10 GbE QSFP+ XLPPI-to-XLAUI PHY. Firmware version D
BCM82328F_B0	82328	Dual 40GbE/Octal 10GbE QSFP+ XLPPI-to-XLAUI PHY
BCM84328_A0	84328	Dual 40 GbE/Octal 10 GbE QSFP+ XLPPI-to-XLAUI PHY. Firmware version D026
BCM84328_B0	84328	Dual 40 GbE/Octal 10 GbE QSFP+ XLPPI-to-XLAUI PHY. Firmware version D026
BCM84333_B1	8481	Quad 10GBASE-T Transceiver. Firmware version 1.69 (Bring-up) (Needs additional software component)
BCM84334_B1	8481	Quad 10GBASE-T Transceiver. Firmware version 1.69 (Bring-up) (Needs additional software component)
BCM84336_B1	8481	Dual 10GBASE-T Transceiver. Firmware version 1.69 (Bring-up) (Needs additional software component)
BCM84793_A0	84793	100GbE/OTN 4x25/28G VSR28 to 10x10/11G CAUI Gearbox PHY. Firmware version 0xD009 (Bring-up - Mode-1 and Mode-3)
BCM84812_A0	8481	Dual 10GBASE-T Transceiver. Firmware version 2.13



Table 125: PHYs

Device	Driver Family	Description
BCM84821_A0	8481	10GBASE-T Transceiver. Firmware version 2.13 (Bring-up)
BCM84822_A0	8481	Dual 10GBASE-T Transceiver. Firmware version 3.02
BCM84823_A0	8481	Dual 10GBASE-T Transceiver. Firmware version 3.02
BCM84823_B0	8481	Dual 10GBASE-T Transceiver. Firmware version 4.02
BCM84823_B1	8481	Dual 10GBASE-T Transceiver. Firmware version 4.02
BCM84833_B1	8481	Dual 10GBASE-T Transceiver. Firmware version 1.69(Driver support for IEEE 1588 features are Bring-up)
BCM84834_B1	8481	Quad 10GBASE-T Transceiver. Firmware version 1.69(Driver support for IEEE 1588 features are Bring-up)
BCM84836_B1	8481	Dual 10GBASE-T Transceiver. Firmware version 1.69(Driver support for IEEE 1588 features are Bring-up)
BCM84844_A0	8481	Quad 10GBASE-T Transceiver. Firmware version 1.07.11(Driver support is Bring-up)
BCM84846_A0	8481	Dual 10GBASE-T Transceiver. Firmware version 1.07.11(Driver support is Bring-up)
BCM84848_A0	8481	Quad 10GBASE-T Transceiver. Firmware version 1.07.11(Driver support is Bring-up)
BCM84728 A0	84728	Dual-Channel 10 GbE SFI-to-XAUI LAN/WAN PHY with 1588. Firmware version 0124 (Driver support for IEEE 1588 features is Bring-up)
BCM84729_A0	84729	Dual-Channel SFI to XAUI with Macsec, 1588 (Firmware version 0x124. Driver support for IEEE 1588 features are Bring-up)
BCM84740 A0	84740	40 GbE PPI-to-XLAUI PHY with EDC. Firmware version D106.
BCM84741 B0	84756	40GbE XLPI-to-XLAUI/Quad 10G with IEEE MACsec/1588 Firmware version 0x0128 [Bring-up]
BCM84747_A0	84728	Quad SFI to XAUI with 1588 (Firmware version 0x124. Bring-up)
BCM84748_A0	84728	Quad SFI to XAUI with WAN/1588 (Firmware version 0x124. Bring-up)
BCM84749_A0	84749	Quad SFI to XAUI with Macsec, 1588 (Firmware version 0x124. Driver support for IEEE 1588 features are Bring-up)
BCM84752 A0	84740	Dual-Channel 10 GbE SFI-to-XFI PHY with EDC. Firmware version D105. (Bring-up)
BCM84753 A0	84740	Quad-Channel 10 GbE SFI-to-XFI PHY with EDC. Firmware version D105.
BCM84754 A0	84740	Quad-Channel 10 GbE SFI-to-XFI PHY with EDC. Firmware version D105.
BCM84756 A0	84756	Quad SGMII/XFI to SGMII/SFI Transceiver Firmware version D105. (Needs additional software component)
BCM84756 B0	84756	Quad SGMII/XFI to SGMII/SFI Transceiver Firmware version 0x0128(Needs additional software component)
BCM84756 C0	84756	Quad SGMII/XFI to SGMII/SFI Transceiver Firmware version 0x0128(Needs additional software component) [Bring-up]
BCM84758	84740	10GbE Quad SFI-XFI PHY with IEEE 1588 Firmware version 0x128
BCM84759 A0	84756	Quad SGMII/XFI to SGMII/SFI Transceiver Firmware version D105.
BCM84759 C0	84756	Quad SGMII/XFI to SGMII/SFI Transceiver Firmware version 0x0128. (Bring-up)
BCM84780_A0	84740	Octal-Channel 10 GbE SFI-to-XFI PHY with 1588. Firmware version 0x128 (Bring-up)
BCM84784_A0	84740	Dual 40GbE/Octal 10GbE QSFP+ XLPI-to-XLAUI PHY. Firmware version 0x125 (Bring-up)
BCM84764_A0	84728	Quad SFI to RXAUI with 1588 (Firmware version 0x124. Bring-up)
BCM84064 A0	84740	Quad 10G-KR-to-XFI or 40G-KR4-to-XLAUI Transceiver. Firmware version 0108.
BCM84074_A0	84728	Quad KR to XAUI (Firmware version 0x124. Bring-up)

Table 125: PHYs

Device	Driver Family	Description
BCM381_A2	82381	Dual 100GbE with CAUI4 to CAUI4 / Dual 40GbE XLPPPI to XLAUI/ Octal 10GbE SFI to XFI port. Firmware version D00E Driver supports 100G CR4, 40G SR4/LR4/CR4 and 10G SR/LR Functionality Supported: PHY Reset(Hard/soft), Link status, Speed and Interface configuration, port enable/disable, AN, Tx Training, remote/digital loopback, Polarity inversion, PRBS, Eyescan, DSC, PLL sequencer restart, Phy dump, FEC and Rx PMD lock. Driver Limitation : 1. Chip reset issues reset to all the ports that shares same PHY-ID(MDIO ADDR).
BCM82764_A2	82764	BCM82764 Dual 40GbE channels with Dual 40GbE XLPPPI to XLAUI / Quad 40GbE PMA reverse bit MUX to convert 2x20G to XLPPPI(4x10G) /CAUI4 to CAUI4 / Octal 10GbE SFI to XFI port. Firmware version D008. Driver supports 40G SR4/LR4/CR4/KR4 and 10G SR/LR/KR/CR Functionality Supported: Link status, Speed and Interface configuration, port enable/disable, Polarity inversion, Power set/get, PRBS, Eyescan, PLL sequencer restart, Phy dump, Rx PMD lock. remote/digital loopback set/get, PHY Reset(Hard/soft), DSC, Phy dump, Tx lane control set/get Tx set/Get, Rx set/Get and Tx Training Driver Limitation : 1. Speed change is only supported through config.bcm. Speed switch through flex port is not supported in this release. 2. Chip reset issues reset to all the ports those shares same PHY-ID(MDIO ADDR). 3. Tx Training may not finish on some 10G lanes(Line side).

OPERATING SYSTEMS

The SDK provides the SAL and BDE abstraction implementations necessary for running the SDK on the following operating systems. See the Platform Guide (56XX-PG818-R) for instructions on porting the SDK to another platform.

Table 126: Operating Systems

Operating System
VxWorks 5.5
VxWorks 6.2
VxWorks 6.4
VxWorks 6.5
VxWorks 6.6
Linux 2.6.21 User Mode
Linux 2.6.21 Kernel Resident Mode
Linux 2.6.25 User Mode
Linux 2.6.25 Kernel Resident Mode
Linux 2.6.27 User Mode
Linux 2.6.27 Kernel Resident Mode
Linux 2.6.35 User Mode
Linux 2.6.35 Kernel Resident Mode
Linux 3.6.5 User Mode
Linux 3.7.10 User Mode
POSIX Compliant (SAL ONLY)

CPU SUBSYSTEMS

Table 127: CPU Subsystems

CPU Subsystem	Description
BCM98245	CPCI 32-bit PPC with Motorola 8245 Processor
BCM98548XMC	XMC 32-bit PPC with Freescale 8548 Processor
BCM953003C	XMC 32-bit MIPS74Kc with BCM53003 Processor
BCM5300X	32-bit MIPS74Kc with BCM5300X Processor
BCM5301X	Integrated ARM Cortex-A9 CPU on BCM5301X Switch Devices
BCM5302X	Integrated ARM Cortex-A9 CPU on BCM5302X Switch Devices
BCM5621X	Integrated MIPS CPU on BCM5621X Switch Devices
BCM5622X	Integrated MIPS CPU on BCM5622X Switch Devices
BCM5331X	Integrated MIPS CPU on BCM5331X Switch Devices
BCM5360X	Integrated MIPS74Kc CPU on BCM5360X Switch Devices
BCM9XLP316LXMC	XMC with Broadcom XLP 316 processor that includes up to sixteen NXCPUs(4 cores)
BCM958625XMC(CPU)	XMC with BCM58625 processor (1.2 GHz ARMCortex.-A9 dual-core processor NXCPUs(4 cores) each operating at up to 2.0 GHz
BCM9XLP208XMC	XMC with Broadcom XLP 208 processor that includes up to eight NXCPUs(2 cores)
BCM9XLP2_XMC_A1	XMC with Broadcom XLP II 200 series multicore processor (MIPS64 Release-II ISA-compliant) with eight NXCPU. processing units, each operating at up to 2.0 GHz

CPU AND OPERATING SYSTEM COMBINATIONS

The following CPU and Operating System combinations are supported by the SDK (in addition to the above):

Table 128: CPU and Operating System Combinations

CPU Subsystem	Operating System	Description
BCM98245	VxWorks 6.2	BSP Provided
BCM98245	Linux 2.6.21	Available through WindRiver Linux 2.0
BCM5621X	VxWorks 6.4	BSP Provided
BCM5621X	Linux 2.6.21	Available through WindRiver Linux 2.0 bcm_ntswics
BCM5331X	VxWorks 6.4	BSP Provided
BCM5331X	Linux 2.6.21	Available through WindRiver Linux 2.0 bcm_ntswics
BCM98548XMC	VxWorks 6.5	BSP Provided
BCM98548XMC	Linux 2.6.27	Available through WindRiver Linux 3.0. Note: Additional patches for issues WIND00172598 and WIND00161649 are required. Contact your WindRiver support personnel for these patches and other WindRiver information.
BCM5300X	VxWorks 6.6	BSP Provided
BCM5300X	Linux 2.6.21	Available through WindRiver Linux 2.0
BCM5300X	Linux 2.6.27	Available through WindRiver Linux 3.x
BCM5301X	Linux 2.6.35	Available through Broadcom Customer Support Portal
BCM5302X	Linux 2.6.35	Available through Broadcom Customer Support Portal
BCM5360X	VxWorks 6.6	BSP Provided
BCM5360X	Linux 2.6.21	Available through WindRiver Linux 2.0
BCM5360X	Linux 2.6.27	Available through WindRiver Linux 3.x
BCM958625XMC br Integrated iPROC	Linux 3.6.5	Available through Broadcom Customer Support Portal
BCM9XLP208XMC br BCM9XLP2_XMC_A1 br BCM9XLP316LXMC	Linux 3.7.10	Available through Broadcom Customer Support Portal
Generic X86	Linux 2.6.25/2.6.27	

Section 8: Release Media

The Software Development Kit is released as a gzipped tar file on the Broadcom Customer Support Portal, <http://support.broadcom.com>. The Network Switching Software Platform Guide, also available on the Customer Support Portal, provides documentation on the various components, the source directory layout, how to build the release for various platforms, and how to customize and port the software to new platforms.

Section 9: Support

Questions, feedback, and/or suggestions should be sent to your Broadcom FAE.

Section 10: Firmware Compatibility Matrix

The following table shows compatibility between different versions of SDK and Firmware releases.

BCM53400 SDK COMPATIBILITY

Table 129:

SDK Version FW Versions	
6.4.3	4.1.0

BCM56150 SDK COMPATIBILITY

Table 130:

<i>SDK Version FW Versions</i>	
6.4.2 - 6.4.3	4.0.3 - 4.1.0
6.3.9 - 6.3.10	4.0.3 - 4.1.0

BCM56440 SDK COMPATIBILITY

Table 131:

SDK Version FW Versions	
6.4.x	4.0.0 - 4.1.0

* Patch required.

BCM56450 SDK COMPATIBILITY

Table 132:

SDK Version FW Versions	
6.4.x	4.0.0 - 4.1.0

BCM56640 SDK COMPATIBILITY

Table 133:

SDK Version FW Versions	
6.4.x	4.0.0 - 4.1.0

BCM56850 SDK COMPATIBILITY

Table 134:

SDK Version FW Versions	
6.4.x	4.0.0 - 4.1.0

BCM88650 SDK COMPATIBILITY

Table 135:

SDK Version FW Versions	
6.4.x	4.0.x

BCM88030 SDK COMPATIBILITY

Table 136:

SDK Version FW Versions	
6.4.x	4.0.x

BMACSEC SDK COMPATIBILITY MATRIX*Table 137:*

Switch SDK Release	BMACSEC SDK Release
5.10.2	3.1
5.10.3	3.2
6.0.1	3.3
5.10.4	3.4
6.0.2	3.4
6.2.0	3.5
5.11.0	3.6
6.2.1	3.7
6.2.2	3.8
6.2.3	3.8
5.11.1	3.9
6.2.4	3.9
6.2.5	3.10
6.2.6	3.11
6.2.7	3.12
6.2.8	3.13
6.2.9	3.14
6.3.0	4.0
6.3.1	4.1
6.3.2	4.2
6.3.3	4.3
6.3.4	4.4
6.4.0	4.5
6.3.5	4.6
6.3.6	4.7
6.3.7	4.8
6.4.1	4.8
6.3.8	4.9
6.3.9	4.10
6.4.2	4.10
6.4.3	4.11
6.4.4	4.12

Section 11: 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.

Table 138: EXTERNALLY LICENSED SOFTWARE COMPONENTS

Component	Origin	Location in source tree	License terms and conditions
EDITLINE	/afs/athena.mit.edu/contrib/sipb/src/editline	src/sal/appl/editline	See (EDITLINE License terms and conditions) (page 242)
ED Editor	USENET comp.sources.misc Volume 9, Issue 36	src/appl/diag/edline.c	See (ED Editor License terms and conditions) (page 244)
CINT	http://www.gnu.org/software/bison/	src/appl/cint/cint_parser.[ch]	See (CINT parser license terms and conditions) (page 245)
CES Driver	BATM Advanced Communications Ltd	src/soc/ces/nemo_driver/*. [ch], src/soc/ces/clsbuilder/*. [ch]	See (Circuit Emulation Service (CES) Driver terms and conditions) (page 246)
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	See (BIGDIGITS license terms and conditions) (page 247)
APIMODE	http://www.gnu.org/software/bison/	src/appl/diag/api/api_grammar.tab.[ch]	See (APIMODE parser license terms and conditions) (page 248)
VxWorks	Wind River Systems, Inc.	systems/vxworks	See (Wind River Systems license terms and conditions) (page 249)

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.crbcal.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.

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.Metzenthien) 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

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.  */
```

CIRCUIT EMULATION SERVICE (CES) DRIVER TERMS AND CONDITIONS

The Circuit Emulation Services (CES) driver code provided herewith is provided by BATM Advanced Communications Ltd (BATM) and is subject to licensing agreement between BATM and Broadcom Corporation.

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.

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 APIMDOE to the `FEATURE_LIST` in SDK compilation flags.

See (CINT parser license terms and conditions) ([page 245](#)) for the Bison licence.

WIND RIVER SYSTEMS LICENSE TERMS AND CONDITIONS

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

Section 12: Resolved Issues for 6.4.3

The following issues are resolved in version 6.4.3 of the SDK.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-43749	589500	56440_A0	BCM_BFD_ENDPOINT_KEY_TYPE_USE_YOUR_DISC flag is provided to specify session identifier type as your_discriminator. (Instead of MPLS label)
SDK-48183	664095	All 56440_A0 56440_A1 56440_B0	When probe port was configured and GPORT_UMAC_CONTROL register was set for the entire block to which the port belongs, the entire block of ports were reset losing the pre-configuration. Now, it resets only the corresponding probe port.
SDK-48899	674806	56850_A0 56850_A1 56850_A2	bcmFieldQualifyIcmpTypeCode support is added in EFP for Trident2 device.
SDK-48913		88640_A0 88650_B0 88650_B1	In Rx Trap module, updating the counter ID via BCM_RX_TRAP_UPDATE_COUNTER was not working. This is fixed.
SDK-48986		88650_A0 88750_B0	Add support for bcm_switch_temperature_monitor_get () API. The API returns an array of temperatures.
SDK-49390	680811	56840_A0 56640_A0 56850_A0 56850_A1	EFP_TCAM entry KEY field width is increased in TD2 and TR3 as compared to TD/TR2. So, relevant changes are made in EFP recovery logic (warm boot) to support this change for TD2 and TR3 separately. Also, changes are made to recover SrcIp, DstIp qualifiers without Ip4 qualifier being part of the Group's QSET.
SDK-49647		56850_A0	EFP bcmFieldQualifyForwardingVlan/vrf/Vpn qualifier initialization has been fixed for Trident2 device.
SDK-49929		88650_A0 88650_B0 88650_B1 88660_A0	Cosq: Inband flow-control triggered by RX PORT FIFO is supported. 1) Added following flag to bcm_petra_cosq_fc_path_add/get/delete API for configuring generation flow-control association: BCM_COSQ_FC_ETH_PORT 2) Added below two new flags for bcm_petra_cosq_gport_threshold <set/get> to set flow-control threshold of RX PORT FIFO BCM_COSQ_THRESHOLD_ETH_PORT_LLFC - configure LLFC threshold BCM_COSQ_THRESHOLD_ETH_PORT_PFC - configure PFC threshold For a calling sequence example, refer to cint_arad_fc_inbnd_config_example.c.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-50261		88660_A0	<p>Added 48 bit stamping support for ARAD+ 1588 TC (transparent clock), whereas the default stamping is 32 bit. 48 bit stamping is enabled by setting <code>bcm88660_1588_48b_stamping_enable</code> SOC property to 1.</p> <p>In systems that contain more than 1 device, in addition to using Broadsync synchronization, the application should make sure that "EPNI_IEEE_1588_TS_MSB_COUNTER" (the upper 16 bit TOD register, increased by 1 every ~4 seconds) register is synced between all devices.</p>
SDK-50689		56640_A0 56640_A1 56640_B0	<p>Index return for TCAM search cli was not correct while searching data in L2 tcam partition. Rectified this behavior for tcam search cli and corrected index returned for searched data in L2 tcam partition.</p>
SDK-50945		56440_B0	<p>Port downsizer workaround failed when SDK is compiled with <code>DBCM_PORT_DEFAULT_DISABLE</code> flag.</p> <p>This was because the phy loopback was not able to be set to perform downsizer check since ports are in disabled state.</p> <p>Fixed by enabling port before loopback and resetting back to original state after performing downsizer check.</p>
SDK-53320	730039	56850_A0 56850_A1 56850_A2	<p>In earlier releases, when <code>alpm</code> is enabled, <code>num_ipv6_lpm_128b_entries</code> could only be set with 0, 1024, 2048, 3072 in combined mode, and its value could not be dynamically configured in both parallel and combine mode. Now, <code>num_ipv6_lpm_128b_entries</code> can be set with any value through 0 to 4096 in combine mode, and it can be dynamically configured via <code>bcmSwitchL3Max128BV6Entries</code> in both parallel and combine mode.</p>
SDK-54076		56640_B0	<p>Configuration of egress properties such as MTU, MAC address on a per L3 interface basis (on same VLAN) cannot be supported in conjunction with the functionality of strict URPF check and ICMP redirect in current architecture. A new configuration property, <code>"l3_intf_vlan_split_egress"</code> is introduced that allows the application to choose which of the functionality is active.</p>
SDK-55374	701952	56850_A2	<p>The BCM API <code>get()</code> funtion for <code>BCM_PORT_PHY_CONTROL_CL72</code> is fixed in this release.</p>
SDK-55650	754938	88670_A0	<p>MPLS PORT: Support use of two bits of orientation for PWE lifs.</p>
SDK-55955 SDK-48128	791050	56850_A0	<p>Issue :- Add Mask Support for <code>DstMplsGport</code>, <code>DstNivGport</code>, <code>SrcMulticastGroup</code>, <code>SrcMplsGport</code>, <code>SrcNivGport</code>, <code>DstMulticastGroup</code>.</p> <p>Fix:- Added SET/GET API for <code>DstMplsGport</code>, <code>DstNivGport</code>, <code>SrcMulticastGroup</code>, <code>SrcMplsGport</code>, <code>SrcNivGport</code>, <code>DstMulticastGroup</code></p>

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-55977		56820_A0	When both Port based mirroring and FP base mirroring is used, then mirror slot container is shared between port and FP leading to FP based MTP destination take precedence over Port mirroring . As a result of this if packet comes from mirrored port matching FP entry it will go to FP pointed MTP destination instead of Port MTP. To correct this behavior MirrorExclusive Switch Control is introduced, which once set will make mirror slot container between port and FP mutually exclusive. Mirror slot container can be shared with in Port Mirroring and FP mirroring but not across. Since Mirror Slot Containers are not shared , packet will go to both MTP as pointed by Port and FP destinations. By default this switch control is not set and has to be set explicitly to achieve exclusiveness between port and FP. This switch control introduces it's own limitation like if both slot container are allocated to Port mirroring then FP based mirroring cannot use third MTP destination as slot containers are not available , same will be true in vice versa case. Customer should be aware of this limitation and should use mirror slot container carefully in this mode.
SDK-55979		56820_A0	When both Port based mirroring and FP base mirroring is used, then mirror slot container is shared between port and FP leading to FP based MTP destination take precedence over Port mirroring . As a result of this if packet comes from mirrored port matching FP entry it will go to FP pointed MTP destination instead of Port MTP. To correct this behavior MirrorExclusive Switch Control is introduced, which once set will make mirror slot container between port and FP mutually exclusive. Mirror slot container can be shared with in Port Mirroring and FP mirroring but not across. Since Mirror Slot Containers are not shared , packet will go to both MTP as pointed by Port and FP destinations. By default this switch control is not set and has to be set explicitly to achieve exclusiveness between port and FP. This switch control introduces it's own limitation like if both slot container are allocated to Port mirroring then FP based mirroring cannot use third MTP destination as slot containers are not available , same will be true in vice versa case. Customer should be aware of this limitation and should use mirror slot container carefully in this mode.
SDK-56039	SDK-64106	766039 56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56843_B0 56841_A3 56846_A1 56841_B0	In previous release, customer found API <code>bcm_switch_control_port_set(unit, port, bcmSwitchPFCClass0Queue, 3)</code> could not work on TD+, but it worked on TD2. This inconsistent behavior made customer confused. A fix has been made now to make the API support legacy cosq mode.
SDK-56223	767450	All	Added KNET software link state control via new proc file interface. Examples: <code>echo eth4=down > /proc/bcm/knet/link</code> <code>echo eth4=up > /proc/bcm/knet/link</code>

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-56279	759897	88650_B0	In some traffic scenarios which includes few devices after device reboot the traffic full bandwidth was not restored. In order to solve this problem we improved our traffic disable procedure by disabling traffic from fabric MAC to FDR.
SDK-56382		56643_A0	Fixed the tx data array size to avoid the stack corruption in <code>_phy_wcmmod_speed-set</code> function
SDK-56754	771594	56334_B0 56850_A2	This problem and fix are applicable to the case where L3 ingress mode (bcmSwitchL3IngressMode switch control) is not being used. When there are multiple L3 interfaces on a given VLAN, all of them will be using the same VRF binding. Hence while deleting one of the L3 interfaces, SDK will not clear the VRF associated with the VLAN. Even in the case of the last L3 interface being deleted, SDK does not clear the VRF binding and expects the application software to necessarily set the VRF binding for the L3 interfaces that may be created subsequently.
SDK-56769		88650_A0 88670_A0	ERP port is no longer treated in Egress Scheduler.
SDK-56932	777407	56850_A0 56850_A1 56850_A2	Previously, it was found that after creating a multicast VXLAN logical port with a multicast egress object, if we call <code>bcm_l3_egress_traverse()</code> , the field port/trunk within the multicast egress object is zero. It is not proper. Now the VXLAN physical port/trunk index on which the multicast egress object is created can be retrieved by that field.
SDK-57033	766664	88660_A0 88670_A0	CoE and dynamic port features have been added. CoE introduces a kind of channelized port, which is mapped via master port plus CoE tag. ARAD has up to 256 channelized ports. Every channelized interface supports up to 64 channelized ports. Dynamic port support dynamic switching between CoE and regular mode. It allows to configure port extender mode at run time . Please see <code>cint_port_extender_dynamic_switching.c</code> for more details.
SDK-57102	779185	56850_A0 56850_A1 56850_A2	In earlier releases, If adding I3 host entry to HW failed, SDK should decrease the related reference count but this function did not work when it has a multipath flag. This has been resolved.
SDK-57475	782537	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In earlier releases, SDK could only configure the egress interface for each ECMP member. The related attributions like failover could not be installed. This has been resolved.
SDK-57700	787637	All	In previous releases, customer found egress flex counters would run out quickly in some circumstances like (4K VLANs + 1 VFI). In this release an improvement has been deployed. The user can now use a new soc property to denote which egress objects can share or exclude flexible counter pool.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-57770		56450_A0 56450_B0	BHH packets are redirected to Host CPU instead of BTE if user specifies to use his own application running in host CPU to process BHH packets.
SDK-58060		All	Updated Linux kernel modules to work with Linux kernel 3.10 and newer.
SDK-58469		88670_A0	The Jericho ITMH is implemented. The implementation is done via Field Processor APIs: it allows the user to define another ITMH header if needed. However, OAMP and OLP are configured to use the predefined Jericho ITMH.
SDK-58552		88650_A0 88650_B0 88650_B1 88660_A0	Added API support for: <code>bcm_port_mdix_get()</code> <code>bcm_port_mdix_set()</code> <code>bcm_port_mdix_status_get()</code> The APIs will work for supported external phys.
SDK-58616	677402	56850_A0 56850_A1 56850_A2	When ports state changes after a flexed configuration, the bit maps for counter thread did not change accordingly. In this release counter bitmaps will be automatically updated after "flexible" ports are flexibly configured. The user does not need to configure bitmaps or re-start counter thread by CLI commands.
SDK-58728		88750_A0	Procedures <code>bcm_fabric_bandwidth_profile_get()</code> / <code>bcm_fabric_bandwidth_profile_set()</code> now refer to 'max_kbps' element of 'profile_array' as implied from its name - Kilo bps (and not Mega bps)
SDK-58854	801789	56340_A0	Warpcore 2 and 3 of BCM56342 devices previously could not support 1G speed. SDK code is now enhanced to support 1G speed on Warpcore index 2 and 3 of BCM56342 device. New device config is supported for BCM56342: <code>spn_BCM56340_4X10 :HX4_1G_130: 7xF.QSGMII + Flex[4x10] + 2xFlex[21G/1G] + 1GE</code>
SDK-58964	797918	56850_A0 56850_A1	In earlier releases, when <code>alpm</code> is enabled, <code>num_ipv6_lpm_128b_entries</code> could only be set with 0, 1024, 2048, 3072 in combined mode. Now, <code>num_ipv6_lpm_128b_entries</code> can be set with any value through 0 to 4096 in combine mode.
SDK-59166	804804	56456_A0	SyncE input clock recovery is supported using switch controls "bcmSwitchSynchronousPortClockSource" and "bcmSwitchSynchronousPortClockSourceBkup". SyncE output using Serdes and Master LCPLL is supported in firmware, however it is not supported in SDK since it needs dedicated CPU to drive the SyncE output at 4KHz loop.
SDK-59313 SDK-58997	803175	56640_A0	Added support to clear XCON and Error CCM defects from software when all the RMEPs are deleted from that group due to a CCM Hardware engine limitation in Triumph3
SDK-59630		88670_A0	In Field Processor HW, the <code>bcmFieldQualifyRxTrapCode</code> qualifier is matched according to the exact HW trap ID. A new API <code>bcm_field_qualify_RxTrapCode32(unit, entry, data, mask)</code> is exposed to match trap-codes accordingly.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-59760	809847	88670_A0	In BCM APIs, the HW FEC Id space is managed by the user. Multiple HW constraints exist on how to handle this space: ECMP groups point only to consecutive FECs, some FECs are protected, etc. An example application is available from now on, describing how to handle the FEC IDs.
SDK-59842		56634_A0 56634_B0	When L3 ingress mode is set, added support in <code>bcm_mpls_tunnel_switch_get()</code> API to return <code>info->ingress_if</code> for BCM5644x and BCM5645x devices.
SDK-60108	812602	88650_A0 88650_B0 88650_B1 88660_A0	Local RBridge doesn't have InLif. Relevant error message added.
SDK-60237	813536	56850_A0 56850_A1 56850_A2	Issue :- Trident2 FP Hardware supports Vxlan Reserved Header Fields which was missing in SDK. Fix :- Added new Qualifier :- <code>bcm_field_qualify_VxlanHeaderBits8_31</code> <code>bcm_field_qualify_VxlanHeaderBits56_63</code> Added New Action list :- <code>bcmFieldActionVxlanHeaderBits8_31_Set</code> <code>bcmFieldActionVxlanHeaderBits56_63_Set</code>
SDK-60279		84740_A0 84780_A0	phy84740.c : Need to allow serdes link status check even in reverse mode (check added in <code>link_get</code> function)
SDK-60508	816530	56340_A0 56340M_A0	During multi hash move operation, the destination bucket can have different sized entries than the incoming entry. In earlier releases, the SDK did not care if the incoming entry could fit into destination bucket. Due to this, it could corrupt the existing entry in some cases. The fix is to add an infrastructure to check if incoming entry can be added into the destination bucket. if not, keep looking for next destination bucket until you get a free space or you have reached to max try counter.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-60596		88660_A0	<p>OAM: Support transmission/reception of CCMs with TLV where Type=2 (port status TLV) or 4 (interface status TLV). The TLV on transmitted CCMs can be set with the API <code>bcm_oam_endpoint_create()</code>. When setting Type=port status is desired, the flag <code>BCM_OAM_ENDPOINT_PORT_STATE_UPDATE</code> should be set in the flags field and the field <code>port_state</code> should be set to one of the <code>BCM_OAM_PORT_TLV_*</code> defines. When setting Type=interface status is desired, the flag <code>BCM_OAM_ENDPOINT_INTERFACE_STATE_UPDATE</code> should be set in the flags field and the <code>interface_field</code> should be set to one of the <code>BCM_OAM_INTERFACE_TLV_*</code> defines.</p> <p>The OAMP may also monitor the port/interface status of received CCMs. This may be done via the <code>bcm_oam_endpoint_get()</code> API for the remote MEP entry. The value of the latest port status will be returned at the field <code>port_state</code> and the value of the latest interface status will be returned at the field <code>interface_status</code>. Likewise for each port/interface value an interrupt may be triggered upon reception of the first CCM with such a value. This can be achieved using the API <code>bcm_oam_event_register()</code>.</p> <p>Note that due to a limitation at the OAMP a program at the egress editor must be used for CCMs destined to the OAMP. The program selection is based on an egress PMF rule. An example of such a rule may be found in <code>cint_oam_field_tlv.c</code>. For a more detailed explanation consult the user manual.</p>
SDK-60609	818931	56224_A0	<p>Problem: On 56224 devices, there would be false L2 DEL + ADD callbacks for entry at location x, if the adjacent entry at location x+1 is modified. Root cause: L2X thread which gives the callback, was using size of 16 bytes instead of device specific number of bytes for comparing. Fix: The change has been made to use device specific size instead of fixed size.</p>
SDK-60715	820099	56640_B0	<p>In this release added a new config <code>phy_xaui_active_lane_map</code> to specify the 100G active lane.</p>
SDK-60907	819128	All	<p>In L2 learning, <code>bcm_l2_addr_msg_distribute_get()</code> and <code>bcm_l2_addr_msg_distribute_set()</code> didn't support aging refresh, i.e. it wasn't possible to register for aging refresh events. Currently there is a new flag <code>BCM_L2_ADDR_DIST_REFRESH_EVENT</code> available for the above set and get functions. Setting this distribution flag will register the user to be notified on refresh events when aging counter is refreshed.</p>
SDK-60920	813693	88640_A0 88640_PCP 88650_A0 88650_B0 88650_B1 88650ACP_A0 88660_A0 88670_A0	<p>The 'snmpBcmTransmittedPkts1519to2047Octets' counted only 1519-1522. Fixed to counter correct length.</p>

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-60945	818527	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	A request was made to enable the SDK to report the reason why a packet was delivered CPU. In earlier releases there wasn't a mechanism to identify the receiving path of packet in current SDK. A new field "rx_path" was introduced in bcm_pkt_t to hold the receive path of packet. A new function dcb23_rx_switch_drop_get() was added for acquiring the "Switched Drop" bit in DCB after packet was received. Based on "Reason Code" and "Switched Drop" of DCB, the rx_path will be updated while rx parsing.
SDK-60987		56850_A0 56850_A1 56850_A2	In the previous release, the buffer peak value was not obtained correctly if BST is operated on PEAK mode. In this release, this issue has been addressed by correcting the internal API.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-61102	822950	All	<p>ARAD supports 32 VLAN ranges per port at most. To add multiple outer VLAN ranges on the same port, <code>bcm_vlan_translate_action_range_add</code> should be called as follows:</p> <pre>bcm_vlan_action_set_t action; bcm_vlan_action_set_t_init(&action); action.ot_outer = bcmVlanActionCompressed; action.new_outer_vlan = 1; bcm_vlan_translate_action_range_add(0, 15, 1, 2, BCM_VLAN_INVALID, BCM_VLAN_INVALID, &action); action.ot_outer = bcmVlanActionCompressed; action.new_outer_vlan = 4; bcm_vlan_translate_action_range_add(0, 15, 4, 5, BCM_VLAN_INVALID, BCM_VLAN_INVALID, &action);</pre> <p>To add multiple inner VLAN ranges on the same port, <code>bcm_vlan_translate_action_range_add</code> should be called as follows:</p> <pre>bcm_vlan_action_set_t action; bcm_vlan_action_set_t_init(&action); action.it_inner = bcmVlanActionCompressed; action.new_inner_vlan = 1; bcm_vlan_translate_action_range_add(0, 15, BCM_VLAN_INVALID, BCM_VLAN_INVALID, 1, 2, &action); action.it_inner = bcmVlanActionCompressed; action.new_inner_vlan = 4; bcm_vlan_translate_action_range_add(0, 15, BCM_VLAN_INVALID, BCM_VLAN_INVALID, 4, 5, &action);</pre> <p>To add multiple outer and inner VLAN ranges on the same port at the same time, <code>bcm_vlan_translate_action_range_add</code> should be called as follows:</p> <pre>bcm_vlan_action_set_t action; bcm_vlan_action_set_t_init(&action); action.dt_outer = bcmVlanActionCompressed; action.dt_inner = bcmVlanActionCompressed; action.new_outer_vlan = 1; action.new_inner_vlan = 4; bcm_vlan_translate_action_range_add(0, 15, 1, 2, 4, 5, &action); action.dt_outer = bcmVlanActionCompressed; action.dt_inner = bcmVlanActionCompressed; action.new_outer_vlan = 6; action.new_inner_vlan = 8; bcm_vlan_translate_action_range_add(0, 15, 6, 7, 8, 9, &action);</pre>
SDK-61167	787136	56640_A0 56545_A0 56640_A1 56640_B0 56545_A1 56545_B0	<p>On external PHY attached to the port, the PHY loopback has to be enabled for both internal and external PHY in order for IBOD to work.</p>



Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-61190	824318	56640_A0 56640_A1 56640_B0	SDK was not clearing REMOTE_FAULT_STATUS and LOCAL_FAULT_STATUS fields of CMAC_CLEAR_RX_LSS_STATUS register upon healthy link up for 100Gbps CE ports. Enabled the processing of CMAC_CLEAR_RX_LSS_STATUS for CE (100G) ports as well.
SDK-61227	823907	56850_A0 56850_A1 56850_A2	Description: Assert for wrong register width in SER handler makes SDK crash. Root cause: The root cause is using soc_reg32_get to read a 64b register(PGW_BOD_XLP0_ECC_STATUS) Solution: In this release the SDK has changed to use soc_reg_get to read the 64b register(PGW_BOD_XLP0_ECC_STATUS) .
SDK-61290	824389	56846_A0 56846_A1	In earlier releases, RxSwLos state machine could not be recovered during warmboot and led to traffic drop issue, This has been resolved.
SDK-61346	825676	56450_B0	In BCM56452, Software was not able to configure WC1 port as 20G Ethernet. This was because the DXAUI operating mode of the warpcore port was configured to support maximum of 10G. Handled this issue by re configuring the port to support maximum 20G Ethernet type traffic.
SDK-61400	825506	All 56450_A0 56450_B0	Adding MPLS entry with explicit queue mode was returning an error as it was always expecting a QOS map id. Fixed code to support both adding of MPLS entry with explicit queue or by specifying the base queue and a valid QOS map id.
SDK-61571	825912	88660_A0 88670_A0	In L2 learning, there was a constraint that didn't allow setting up ingress distribution learning and distribute it to a LAG port. The constraint was removed.
SDK-61625	816742	88650_A0 88650_B0 88660_A0	In Field Processor, when working with external tcam (KBP), API bcm_field_group_dump() returns incorrect values. This API was changed to support external TCAM correctly. In order to use this API with external TCAM, a new SOC property is introduced and should be set: custom_feature_unbound_uninstalled_external_tcam_entries_number=1. When set, the properties of the installed entries which reside on the external TCAM are kept in the SW state. This information is retrieved during the dump call. Note that the setting of this SOC property increases the memory allocation for Field Processor SW state.
SDK-61638	826721	88650_B1 88660_A0	In L2 module, when working in centralized mode, the LIF-valid bit entry was not received correctly on learn events (i.e., when the CPU was inserting learned entries via BCM SDK) for FEC destined entries (for example, FEC learning for PW tunnel). The LIF-valid bit is now set correctly on the learn events and matches the payload of the device learned entry.
SDK-61735	825191	88660_A0 88670_A0	In Ingress PMF, setting the action bcmFieldActionVportNew to an Out-LIF-ID is performed incorrectly, because the conversion from Gport to Out-LIF-ID is not supported. This is fixed.
SDK-61754	825213	56643_A1	Fixed a bug in hg42G speed set when cl72 is enabled.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-61769	827751	All	Ensure that KNET Tx buffers are flushed after optional Tx callbacks are done, otherwise we may transfer stale data, if system is not cache-coherent.
SDK-61833	817014	56720_A0 56743_A0 56745_A0 56746_A0 56750_A2	In earlier releases, creation of virtual port multicast groups with ID did not work correctly on fabric-only devices. Added support to handle various ipmc group types for fabric only devices and added warmboot support for the array that is used to store the ipmc group types.
SDK-61925	826420	56846_A1	In previous releases, if parity generation was disabled for TD+ and a specific field was cleared in the table MMU_CTR_UC_DROP_MEM, for example PKT_CNT, this may cause parity bit mismatch, then the table entry would be stuck on a value and not increase. In this release, this issue has been fixed by writing 0 to the whole entry when a specific field is going to be cleared.
SDK-61998		56640_A0 56640_A1 56640_B0	bcm_port_phy_set has a input parameter port, which can be a valid port number or a PHY MDIO address. Until 6.3.7, in bcm_port_phy_set API, the input parameter port is not checked for validity. In 6.3.8, port validity check is added and so, this API does not work when PHY MDIO address is passed via port parameter. This is again fixed in 6.3.10. Similar check is added to bcm_port_phy_get and bcm_port_phy_modify
SDK-62007	831748	56850_A0 56850_A1 56850_A2	Replace global SPL lock with dedicated mutex or spinlock wherever possible to reduce contention for the SPL lock.
SDK-62010	788734	88660_A0 88670_A0	Until now, the FTMH.DSP-Extension header was added to packets only in stacking systems, when the stacking-enable SOC property was set. However, e.g. for snooped and mirrored packets, users may want to add a DSP-Extension for the CPU to know what was the original packet's destination. This can be done from now on by setting ftmh_dsp_extension_add to '1'.
SDK-62023		88650_B0 88660_A0	Fixed a bug that caused unwanted bytes to be added to some packets in case user-header was used.
SDK-62031	831488	88650_B1	In L2 MAC table learning, when L3 routed learn mode was enabled, SA look up was opportunistic and thus SA unknown traps were not raised. Now SA look up is guaranteed and SA unknown traps are raised.
SDK-62041	831895	56640_A0 56640_A1 56640_B0	For any MPLS tunnel, in cases when failover is used, we usually create the failover port first, and pass its port_id as a parameter when creating the primary port. In previous releases, when we delete the failover port and add it again, the newly created failover port over writes the settings of the primary port. This is fixed as part of this JIRA.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-62070 SDK-60189		56340_A0 56850_A0	<p>In earlier releases, customers were unable to use warmboot as a tool for ISSU because SDK takes a while to warmboot and CPU packet tx and rx could not be started in a timely manner from application. This would cause protocol timers to expire and misbehave.</p> <p>In order to not starve the protocol timers and to allow applications to send and receive packets early in the initialization sequence and even during warm booting, new support has been added through <code>bcm_attach_early_txrx()</code> and <code>bcm_detach_late_txrx()</code> APIs.</p> <p>API <code>bcm_detach_late_txrx()</code> can be used to delay de-initializing the BCM TX and RX components until all the other BCM components are de-initialized. Then customers TX and RX threads have to be shut down after which <code>bcm_detach()</code> must be called as usual to de-initialize BCM TX and RX components and other control structures for a safe shutdown.</p> <p>API <code>bcm_attach_early_txrx()</code> can be used to attach the BCM unit and initialize the BCM TX and RX components. Customers, from their application code, can then start transmitting and receiving packets to/from CPU. Regular <code>bcm_attach()</code> API must still be invoked to allow complete and safe re-initialization so BCM APIs can be used for configuring the system.</p> <p>However, the CPU packet TX and RX procedures are limited by its nature during the warmboot process. In the absence of the required control data, link status and other integrity checks may be exempted during <code>bcm_tx()</code> when warm booting. Similarly processing of some packet metadata elements and converting the same to standard BCM types may not happen during packet RX.</p>
SDK-62092	824440	All	<p>In the earlier SDK, the <code>IESMIF_INTR</code> in <code>IP1_INTR_ENABLE</code> was enabled after SER had handled other errors which should be kept as disabled. When the status of <code>IESMIF_INTR</code> was set by accident, it made interrupt handler go to infinite loop by checking the status of <code>IESMIF_INTR</code>, and thus consumed CPU a lot. The issue has been fixed in this release.</p>
SDK-62097	831042	56340_A0 56450_A0 56640_A0	<p>Code to set <code>CLASS_ID</code> field in MPLS view was not present. Have added the same for supported platforms (<code>bcm56340_a0 56450_a0 56450_b0 56450_b1 56640_a0 56960_a0</code>)</p>
SDK-62168		88650_A0 88650_B0 88660_A0	<p>Resolve static analysis warnings</p>
SDK-62221	833933	56450_B0	<p>Problem: <code>bcm_l3_egress_create</code> was failing when we passed queues that is created using subport Fix: Now the validation is changed , so that validation is done with the actual port instead of <code>pp_port</code></p>

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-62248	821582	All	Fixed incorrect error handling in <code>sal_mutex_take</code> for Linux user mode (unix) SAL when <code>USE_POSIX_MUTEX_TIMEDLOCK</code> is enabled.
SDK-62256	795913	56640_A0 56640_A1 56640_B0	Fixed the behavior on Triumph3 to allow deletion of FDBs even if VP is not present
SDK-62258	822988	56850_A2	In earlier releases, <code>In</code> function, " <code>int soc_ser_log_load(int unit, void *location)</code> ", we inadvertently created a Mutex which needed to be destroyed but was not. This was uncovered by a customer static analysis tool. . This has been resolved.
SDK-62295	827961	56850_A0 56850_A1 56850_A2	In earlier releases, <code>bcm_vlan_stp_set/get()</code> didn't support virtual port. This has been fixed in this release by getting virtual port number from parameter <code><port></code> , then setting STP state to corresponding register. However, the behavior is different compared to physical port. For physical port, <code>port+STG</code> is atomic; while <code>port+vlan</code> is atomic for virtual port. User must be careful to handle the difference. Virtual port must be added to the vlan before calling <code>bcm_vlan_stp_set/get()</code> . Same STP state must be explicitly set to all vlans that comprise a STG.
SDK-62338	835376	88650_B1 88660_A0 88670_A0	Default In-LIF is used whenever the In-LIF lookup has no result. There was an inconsistency in the handling of this In-LIF in general and specifically the association of an In-LIF Profile by calling <code>bcm_port_class_set()</code> failed for the Default In-LIF. The Default In-LIF is now treated as any other Ingress only In-LIF including successful configuration of an In-LIF Profile.
SDK-62354	820140	88660_A0 88670_A0	Maximum isem vt/tt programs number fixed to be 16(arad)/32(jericho).
SDK-62384	827518	56820_A0 56820_B0	In earlier releases, 100M Fiber mode was not getting disabled for 1G port when enabling autonegotiation. In this release we now disable 100M Fiber mode before enabling autonegotiation.
SDK-62386	834601	56640_A0 56640_A1 56640_B0	1) The missing scheduler hierarchy occurs for nodes whose parent index is greater than 512. Due to an error in the code, when the node's parent index is greater than 512, we save the parent index in scache as 512 which, upon recovery, is treated as invalid index. This has been fixed by increasing the number of bits for parent index from 11 to 12. Since we can't modify the existing 11 bit parent id (no free space left in the first word), the 12th bit is saved in the next word. 2) Another issue seen was for multicast queues, the <code>hw_index</code> starts with an offset of 1024. However, during sync the maximum <code>hw_index</code> is considered as 1024 (0x400). This has been fixed by decrementing the <code>hw_index</code> by 1024 during sync for multicast queues. Upon recovery, increase the same by 1024. 3) The reason priority mapping was not being restored is that invalid <code>hw_cosq</code> is checked using 0xf instead of 0x8. This has been corrected by checking for invalid <code>hw_cosq</code> using 0x8 during recovery.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-62402		56850_A0 56850_A1 56850_A2	Coverity issues (memory leak) found by a customer using static analysis tools have been resolved.
SDK-62408		88650_A0 88650_B0 88660_A0	MPLS: Ability added to process (send and receive) MPLS traffic on certain port without an Ethernet header. For more details see PP User Manual.
SDK-62427		88650_A0 88650_B0 88650_B1 88660_A0	Apply HW recommendations for WC current. It is recommended for all customers using 6.3.x releases to integrate this fix.
SDK-62451		56640_B0	In Triumph3, <code>bcm_cosq_classifier_mapping_clear</code> was deleting the whole profile when called for a single port. After this fix the function clear only the port offsets. The complete profile will be deleted if its the last port in the profile
SDK-62454	835758	56846_A0	Added missing protection of critical section in implementation of BCM APIs for traffic shaping on BCM5684x and BCM5685x device families.
SDK-62463	SDK-59096 835815	56450_A0 56450_B0	In this release we have added SDK APIs to enable configuring Protection Status in Higig Header. "bcmSwitchRemoteProtectionEnable" can be used to Set "SET_PROT_STATUS" bit in <code>EGR_HG_HDR_PROT_STATUS_TX_CONTROL</code> register. While "bcmSwitchRemoteProtectionTrust" can be used to Set "USE_PROT_STATUS" bit is set in the <code>ING_CONFIG_64</code> register.
SDK-62466	834064	88660_A0 88670_A0	L3VPN: We introduce a new CINT demonstrates BCM886XX being PE-based L3VPN. In CINT following traffic scenarios: 1. Route Unicast traffic 2. mVPN application (multicast IPMC traffic) 3. For 88670_A0: we demonstrate how to configure hierarchical fec for L3VPN. See CINTs: <code>cint_l3vpn.c</code> , <code>cint_mvpn.c</code>)
SDK-62490		88650_A0	MPLS: <code>bcm_petra_mpls_port_get</code> fails for protected PWEs.
SDK-62501	834596	56850_A0 56850_A1 56850_A2	In previous releases, customer found that when a 40G port was changed to 4x10G, it would cause cosq BW shaping functionality is broken. The issue was caused by two reasons. 1 <code>SOC_INFO(unit) ->port_speed_max</code> was not changed to 10G after changing flex port. 2 On TD2, port queue weights should be used to calculate the burst size. Now this issue has been fixed.
SDK-62512	835671	56830_A0 56830_A1 56830_A2	In earlier releases, <code>MY_MODID</code> value was not set correctly in port entry of <code>cpu_hg_index</code> for BCM56831 when module-id was changed. This has been fixed in this release.
SDK-62524	834472	56640_A0 56640_A1 56640_B0	In earlier releasedJs, MacDrain Timeout error was experienced in a flex port configuration. The <code>mac_drain</code> operation is was being done for ports in disabled state causing in stale packets in the MMU. For disabled ports the <code>mac_drain</code> operation is now skipped as it is not required.
SDK-62591	824848	56636_B0	Schan timeout occurred while accessing un-initialized ESM through dump soc command. Corrected this behavior by skipping ESM registers if module is un-initialized in absence of external tcam config to avoid schan timeout error messages.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-62612	837279	56340_A0	Per customer request have added a new phy control option in existing <code>bcm_port_phy_control_set()</code> API which can freeze DFE values in serdes warpcore. New phy control option <code>BCM_PORT_PHY_CONTROL_DFE</code> is added in this fix.
SDK-62625	835156	56640_B0 56643_A0	When ACL 144 without L2 table was not configured, the keyoffset used for ACL was incorrect. Correct it to use ACL ONLY config when l2 table is not present
SDK-62632		88650_A0 88660_A0	OAM: To disable support for NTP in OAM DM messages, set SOC property: <code>oam_dm_ntp_enable=0</code>
SDK-62639		88650_A0 88650_B0 88660_A0	Diagnostics for IPv4 MC when working with external TCAM (KBP) returned wrong values for VRF != 0. This is fixed.
SDK-62641	831284	All	Fixed issue where PTP signaling deny messages from master could cause brief deadlock.
SDK-62642		88670_A0	Hierarchical FEC usage in Routing Over Vxlan. For more information see <code>cint_vxlan_roo.c</code>
SDK-62646		All	Fixed case where uninitialized memory could be used for a message to uKernel in a PTP fault case.
SDK-62652	835835	88660_A0 88670_A0	In Ingress PMF, it is possible to configure a NOP action for <code>bcmFieldActionPolicerLevel0</code> by setting <code>param0=0</code> . The same ability is added for <code>bcmFieldActionPolicerLevel1</code> .
SDK-62661		56960_A0	Continuous DMA SOC property added to enable continuous DMA mode for Packet I/O. The underlying Continuous DMA function is not supported in this release
SDK-62736		88660_A0	In Ingress Field Processor stage, in BCM88650/88660 devices, up to eight 160b keys can be built. However, one key has a different (called interleaved) TCAM resolution than the 7 others. A special mode allows to use this special key as long as the entries are limited to a single TCAM bank. To use this mode, create the Field group with flag <code>BCM_FIELD_GROUP_CREATE_SINGLE</code> .
SDK-62752	836938	All	During <code>bcm_ipmc_add</code> , the hit bit for that multicast entry was set by default in SDK. SDK was not servicing <code>BCM_L3_HIT_CLEAR</code> flag sent by user. Added code to not set hit bit if user sends <code>BCM_L3_HIT_CLEAR</code> flag.
SDK-62754	836778	88650_B1 88660_A0	VLAN filtering is used to discard packets based on packet's tag format. When advanced VLAN translation is enabled, VLAN filtering is set via <code>bcm_port_tpid_class_set()</code> . The function didn't support separate configuration for Ingress and Egress using the flags <code>BCM_PORT_TPID_CLASS_INGRESS_ONLY</code> and <code>BCM_PORT_TPID_CLASS_EGRESS_ONLY</code> . Instead, Ingress and egress VLAN filtering were set at the same time to similar values. Ingress and egress VLAN filtering can now be configured separately by calling <code>bcm_port_tpid_class_set()</code> with flag <code>BCM_PORT_TPID_CLASS_INGRESS_ONLY</code> or <code>BCM_PORT_TPID_CLASS_EGRESS_ONLY</code> after the fix.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-62765	836249	56450_B1 56450_A0 56450_B0	<p>In BCM5645x devices, DROP_PKT_R and DROP_BYTE_R counters were getting incremented even there was no actual drop.</p> <p>This was because, the software counter was reading the dequeue table incorrectly during the non dma flex counter sync operation.</p> <p>Fixed this issue by introducing a feature check to stop syncing flex counters when counter_toggled_read feature is not present.</p>
SDK-62774	837984	88650_B1 88660_A0	<p>In FCoE, API function bcm_fcoe_zone_delete() was not supported. The API function is now implemented and supported.</p>
SDK-62799	827592	88650_A0 88660_A0	<p>The soc property "customer_feature_always_map_result_dp_2_to_1" is used to map meter resolved DP to ingress DP and egress DP. It doesn't support mapping meter resolved DP to ingress DP only or egress DP only. Soc property "customer_feature_always_map_ingress_result_dp_2_to_1" has been added to map meter resolved DP to ingress DP only. Soc property "customer_feature_always_map_egress_result_dp_2_to_1" has been added to map meter resolved DP to egress DP only.</p>
SDK-62822		88660_A0	<p>Add MPLS tunnel gport information to SW database in order to support OAMoMPLS get functionality.</p>
SDK-62845		56850_A0	<p>In earlier releases, MTP port in IM_MTP_INDEX table could not be deleted after warm reboot for filter based mirroring. This has been fixed in this release.</p>
SDK-62854		56850_A2	<p>In earlier release, if there were configuration errors existed in non-RH mode, SDK could never predict or get the correct ECMP hash result in RH mode. This has been resolved.</p>

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-62864		88650_A0 88660_A0	<p>In Ingress PMF, each program may build up to 7 keys out of the potential 8 keys, where one key (first key in first PMF cycle) is unused due to a unique search method that is not supported by the driver. The support is now added, with the limitation of using a single TCAM bank for that key. In order to allow the use of that key, user must indicate using a single bank database once configuring a new field group, by setting the flag <code>BCM_FIELD_GROUP_CREATE_SINGLE</code> when creating field group.</p> <p>For example:</p> <pre>uint32 fg_flags = BCM_FIELD_GROUP_CREATE_WITH_MODE BCM_FIELD_GROUP_CREATE_WITH_ASET BCM_FIELD_GROUP_CREATE_SINGLE ; bcm_field_entry_t ent; bcm_mac_t damac1; /* mask the DA qualifier*/ uint32 i; BCM_FIELD_QSET_INIT(qset); BCM_FIELD_QSET_ADD(qset, bcmFieldQualifyStageIngress); BCM_FIELD_QSET_ADD(qset, bcmFieldQualifyDstMac); BCM_FIELD_ASET_INIT(aset); BCM_FIELD_ASET_ADD(aset, bcmFieldActionCodeDestSet); BCM_FIELD_ASET_ADD(aset, bcmFieldActionCodeSourceSet); bcm_field_group_config_t_init(&udh_group); udh_group.group=10; udh_group.priority=2; udh_group.mode=bcmFieldGroupModeAuto; udh_group.flags= fg_flags; udh_group.aset=aset; udh_group.qset=qset; res = bcm_field_group_config_create(unit, &udh_group); if (res != BCM_E_NONE) { printf("Error Field, bcm_field_group_config_create "); return res; }</pre>
SDK-62925	836745	56634_A0 56634_B0 56636_A0 56638_A0 56639_A0	<p>On 56634 devices with L2 tables on external memory, using cli command "l2 clear vlan" which internally calls the API <code>bcm_l2_addr_delete_by_vlan()</code> to delete MPLS VPN entries only, was not working. The entries were not being deleted. Though using the same command to clear vlan + mac based entries was working.</p> <p>The SDK uses software based deletion (clearing) of entries, which was not supported for the MPLS VPN entries.</p> <p>This has been addressed by using hardware based clearing of MPLS VPN based entries instead of using software based clearing..</p>

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-62946	839616	56846_A0	In previous releases, the possible value <code>_SOC_ACC_TYPE_PIPE_GROUP</code> of the 3rd parameter for the routine <code>soc_trident_populate_tcam_log()</code> was not considered and checked. This will cause SER actions for parity error in <code>FP_TCAM</code> table in TD+ X-pipe will not have log recording. In this release, the value <code>_SOC_ACC_TYPE_PIPE_GROUP</code> has been checked in the routine, and SER logging will work for <code>FP_TCAM</code> table in TD+ X-pipe.
SDK-62949		88660_A0	BFD default profile configuration result in allocation error.
SDK-62961		56440_A0 56450_B0 56740_A0	In the earlier release support for configuring the <code>EH_EXT_HDR_ENABLEf</code> in <code>EGR_PORT</code> was absent for KATANA2 and hence adding support for its configuration.
SDK-62979	803157	56640_A0 56640_A1 56640_B0	The following switch controls are not supported on BCM56640. updated the API Support Matrix with the same information. <code>bcmSwitchTrunkHashSet1UnicastOffset</code> <code>bcmSwitchTrunkHashSet1NonUnicastOffset</code> <code>bcmSwitchFabricTrunkHashSet1UnicastOffset</code> <code>bcmSwitchFabricTrunkHashSet1NonUnicastOffset</code> <code>bcmSwitchLoadBalanceHashSet1UnicastOffset</code> <code>bcmSwitchLoadBalanceHashSet1NonUnicastOffset</code> <code>bcmSwitchECMPHashSet1Offset</code>
SDK-62995 SDK-58494		All	Moved iProc and PCI initialization code from O/S-specific BDE modules into new shared BDE library in <code>\$SDK/systems/bde/shared</code> . The new library is designed for easy incorporation into BDE-equivalent customer code. For more information, please see <code>\$SDK/doc/bde-library.txt</code> .
SDK-63001		56850_A0	Added documentation for Enumerations for <code>"bcm_udf_layer_t layer"</code> used to choose UDF base offset
SDK-63025	840239	56450_B1 56440_A0 56440_A1 56440_B0 56450_A0 56450_B0 56640_A0 56640_A1 56750_A0 56750_A1 56750_A2 56850_A0 56850_A1 56850_A2	Problem Statement: Warmboot does not recover a TRUNK member configured with <code>BCM_TRUNK_MEMBER_EGRESS_DISABLE</code> . Resolution: The <code>rtag</code> is set to 0 if <code>BCM_TRUNK_MEMBER_EGRESS_DISABLE</code> is set. Warmboot reinitialization was returning without reading the trunk members if <code>rtag</code> is zero. The trunk is now recovered if the it has atleast one member port in it and even if <code>rtag</code> is zero.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-63060		88650_A0 88660_A0	<p>Exact match tables (similarly to other memories) may get corrupted due to the release of alpha particles. Entries in exact match tables are protected by a parity check.</p> <p>Soft error recovery (SER) was not supported for exact match tables. SER is now supported for entry corruption in all of the exact match tables. A new SOC property is supported: <code>exact_match_tables_shadow_enable</code> (set to '0' by default). When set, a shadow for all exact match tables is maintained in the SW. This property must be set in order to support the exact match SER functionality. Note that in some cases, when the corruption is in a MAC table entry, and may not be sourced to an entry which is kept in the SW shadow (i.e., a dynamic entry), the recovery time may be very long (dozens of seconds). In such case, we must align the shadow and HW static entries by comparing the shadow and HW databases and perform the following: 1. Add entries which reside in the shadow but missing from the HW database 2. Remove entries which reside in the HW database but are missing from the shadow</p> <p>Assumptions: - Only up to a single entry is corrupted at a given time - Corruption includes a single bit flip</p>
SDK-63079	837938	56960_A0	Added MBIST support (TR 147) for Tomahawk SDK.
SDK-63087 SDK-53093	838160	56240_B0 56242_A0 56242_B0	During warmboot Stat type was getting recovered incorrectly which caused entry deletion to fail . Corrected the same.
SDK-63102	841356	56440_B0	To address the issue of RDI being reset for CPU generated OAM TX packets, OAM group zero is now reserved in B0 device
SDK-63109	836449	56514_A0	When ClassTag is enabled, tag determination was not proper as HiGiG header was used. In case of ClassTag format, HiGiG header will not carry the tag info, so required info is obtained from DCB instead of HiGiG Header.
SDK-63120	841723	88660_A0	Support basic BFDIPv6 application. Multiple single hop sessions can be built between two BFD endpoints. BFDIPv6 packets are trapped to embedded ARM core via PMF rules. Checks for hop-limit, DIP and BFD version are done via PMF rules as well. <code>cint_bfd_ipv6.c</code> is a reference for BFDIPv6 application.
SDK-63130	841579	56850_A2	In earlier releases, <code>bcm_l3_host_delete_all</code> read entries from hw and was slow for large I3 table size. This has been improved by reading from DMA.
SDK-63142		88650_B1 88660_A0	ECN delay measurement adds a capability of signaling a congestion in the device on the ecn bits in ipv4 headers. The congestion is calculated by comparing the delay of the packet inside the device to a threshold configured by the user (in nano scale). Example can be found in <code>cint_field_ecn_dm.c</code> Note: this feature does not support WarmBoot and ISSU.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-63231	832200	56634_A0 56634_B0 56636_A0 56636_B0 56638_A0 56638_B0 56639_A0	The issue was that the customer could not use BCM_L3_COPY_TO_CPU flag in bcm_l3_egress_create API along with CPU interface index for L3 interface, as there was a check for a valid VLAN programmed in the L3 interface. The fix is to skip the VLAN check for the special case of CPU interface index (maximum index of EGR_L3_INTF table).
SDK-63233		56640_A0 56640_A1 56640_B0 56643_A0 56643_A1 56643_B0	Fixed logic for OAM endpoints warmboot recovery on TR3, for endpoints configured on Trunk ports
SDK-63250		56850_A0 56850_A1 56850_A2 56854_A0 56854_A2 56854_B0	In earlier releases, after creating and deleting udf ids over 65535 times, allocated udf ids wrapped back and increased from 0. Validation check did not take effect in this case. This has been fixed in this release by fixing the id allocation function.
SDK-63262		88670_A0	Support new BFD types for Jericho: bcmBFDTunnelTypePweRouterAlert, bcmBFDTunnelTypePweGal. See bfd_pwe_example() in cint_bfd.c or user manual for details.
SDK-63264	842499	56450_A0 56450_B0	In non pre-emptive kernels, thread trying to acquire spinlock would hog the CPU there by not allowing holding thread to release lock. Fix ensures that attempting thread yields (by using "select" call with non-zero timeout parameters each time) if it fails to acquire spinlock.
SDK-63265		56850_A0 56850_A1 56850_A2	Two possible ICMPTTypeCode combinations were missed in TD2 qualifier database, which caused group create to fail. Added the missing combinations.
SDK-63272		88650_A0	Added support for MAC prbs when the physical coding sublayer (PCS) is configured to 64_66 without FEC.
SDK-63284	838369	88650_A0	OAM: By default when a down MEP traps a LBM from an unknown unicast address or from a multicast address the packet is trapped to the CPU with trap code user_defined_38. This behavior may be changed using the API bcm_oam_endpoint_action_set() with the bcm_oam_action_type_t set to bcmOAMActionMcFwd and the opcode set to 3 (LBM).
SDK-63301	842644	56850_A0 56850_A1 56850_A2	In earlier releases, MMU_WRED_CONFIG_X_PIPE index of port based WRED was not right when WRED configured for all COS queues (cosq==1) with flag BCM_COSQ_DISCARD_PORT. This has been fixed in this release.
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.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-63344	843227	56450_B1 56450_A0 56450_B0	In BCM5645x devices, Configuring lanes with external PHY was failing when external phy driver supported a speed lower the maximum supported internal port speed. This was caused because maximum speed supported by a port was always set to internal speed supported by the port. Fixed issue by setting max speed of port as the lowest of internal and external phy driver speeds.
SDK-63348	843289	84848_A0	Added support for configuring fixed and variable latency for autogrEEEn for BCM84848 and BCM84334 drivers.
SDK-63350 SDK-66934	837895	56547_A0	The earlier SDK code didn't handle SER error event DLB_LAG_FLOWSET_TIMESTAMP_PAGE. This caused the SER error event of this memory being to be reported continuously. Code to clear the corrupted entry of DLB_LAG_FLOWSET_TIMESTAMP_PAGE has been added in the SER correction procedure to resolve this issue.
SDK-63363	838980	All	Problem definition : For MPLS TP LSP, Microcontroller was generating BFD packet skipping outer MPLS tunnel label and putting GAL label directly. This was due to the missing validation on SDK while retrieving the MPLS tunnel label from I3 egress interface. Soltuion :bcm_bfd_endpoint_create API returns error when failed to retrieve the MPLS tunnel label from L3 egress interface during encapsulation data creation.
SDK-63430		88640_A0 88650_B0 88660_A0	When receiving a source routed cell, copied payload data was incomplete Fixed.
SDK-63435		56850_A0 56850_A1 56850_A2	In earlier releases, BCM_VLAN_MPLS_DISABLE flag could not take effect on TD2 devices by calling bcm_vlan_control_vlan_set. This has been fixed in this release.
SDK-63437	843043	88650_A0	Bug fixed - Creating a MPLS tunnel with NULL label(label=0) explicitly returned error.
SDK-63484	843304	56850_A0 56850_A1 56850_A2	In earlier releases, bcm_xgs3_vlan_table_reload() should restore the value of prot_pkt_ctrl ref counts after warmboot, but this function did not work. This has been resolved.
SDK-63503		88650_A0	Reflector: benchmarking_methodology.c now runs in advanced VLAN editing mode.
SDK-63504		88650_A0 88660_A0 88670_A0	Added diagnostics command for OAM: diag oam Action_Key This diagnostic models the key generation for OAM action lookup in the classifier.
SDK-63522	823790	56850_A0 56850_A1 56850_A2	In earlier releases, ARL flag was cleared when doing multiple port operations simultaneously on Trident2. This has been resolved.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-63523		88660_A0	bcmVlanPortDoubleLookupEnable is used for PON port to search "Tunnel ID x Outer VLAN x Inner VLAN" first and "Tunnel ID x Outer VLAN second". An error occurs when trying to set bcmVlanPortDoubleLookupEnable on channelized PON ports. The issue detailed above affected lookup configuration of PON channelized port. ARAD supports bcmVlanPortDoubleLookupEnable for each channelized PON port after the fix.
SDK-63547		56450_B0	In BCM5645x devices, With the enhancement of Multiple split horizon groups, group id can be programmed for mim ports. However, this was missing for Access ports. Fixed this issue by enabling access ports to be programmed with network group id like peer and backbone ports.
SDK-63551		56845_A2 56850_A2	In earlier releases, TD didn't support setting MIN/SHARED Limit of UC/MC queues in <code>bcm_cosq_control_set()</code> , and TD2 failed to stop the traffic by setting MIN and SHARED limit of UC/MC queues to 0. This has been fixed in this release. Now both TD and TD2 can make port cosq stop traffic by setting MIN and SHARED limit of UC/MC queue to 0.
SDK-63564		88650_A0 88650_B0 88660_A0	Improvements for XLPORT Overrun/Underrun Workaround.
SDK-63582		56640_A0 56640_A1 56640_B0	Introduced faster look up of existing mpls entries to speed up API <code>bcm_mpls_port_add()</code> . Faster look up uses the hash table mechanism than linear search used today. During <code>bcm_mpls_port_add()</code> , SDK looked into all the label entries in h/w to check if there was already same existing entry present. This linear search was very slow and it slowed down <code>bcm_mpls_port_add()</code> . The solution was to add a hash mechanism to match for the existing entries. Hash will provide a faster look up. Using Hash mechanism has reduced the time taken by <code>bcm_mpls_port_add()</code> significantly (from 2300usecs to 600 usecs).
SDK-63589		88650_A0 88650_B0 88650_B1 88660_A0 88750_A0 88750_B0	Eyescan doesn't present output when external phy isn't presented. Fixed.
SDK-63596		56450_B1 56450_A0 56450_B0	In BCM5645x devices, <code>bcm_port_subsidiary_ports_get</code> function was not returning correct flex ports. This was because KT2 support was not provided for that function. Fixed this issue by provided support for the function <code>bcm_port_subsidiary_ports_get</code> and returning TRUE for controlling ports and FALSE for auxiliary/flex ports.
SDK-63603	845921	56640_A0 56640_A1 56640_B0	With previous releases, there might be a false link up notification when hardware linkscan is enabled on a port, even when link is down. This is now fixed by check the status of both cmic registers and phy registers before enabling hardware linkscan.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-63608		56640_A0 56640_A1 56640_B0	Problem: VFP stats for colors where not retrievable since the flex counter offset mode was getting wrongly programmed. Solution: Updated the correct flex counter mode to be passed while setting flex stat info and retrieve the correct mode after warmboot as well.
SDK-63618		56845_A2 56850_A2	Issue :- L3 Reference Count is not updated when Egress Objects are added through FP. Fix :- Updated FP code to update L3 Reference Count for actions "L3Switch" and "RedirectEgressNextHop"
SDK-63653		88650_A0 88650_B0 88650_B1 88660_A0	[Minor] Add lane option to eyescan diagnostics description message.
SDK-63676		88650_A0 88660_A0	BFD: when calling <code>endpoint_create()</code> for endpoints of type IP one hop, TTL must be 1 and for endpoints of type MPLS IP TTL must be 255.
SDK-63677	734372	88650_A0 88650_B0 88660_A0	MPLS: Added support for switch control <code>bcmSwitchMplsDefaultTtlCopy</code> , which enables set/get for the default TTL value in MPLS header in case of Uniform model.
SDK-63686	845847	56450_B1 56450_A0 56450_B0	Problem Statement: <code>bcm_cosq_port_sched_set()</code> failing with <code>BCM_E_TIMEOUT</code> in high traffic situation Resolution: Added a work around so that the node get enough bandwidth so that queue flush doesn't gets timed out
SDK-63706	845222	88650_B0 88650_B1 88660_A0	TDM bypass mode - MC group number enhanced to 64K. Before this fix only 16K multicast groups (1:16K-1) was allowed at TDM bypass mode. This fix enhances the number of multicast groups to 64K (1:64k-1) for TDM bypass mode.
SDK-63716	845844	56450_B1 56450_A0 56450_B0	When build flag <code>BCM_PORT_DEFAULT_DISABLE</code> was used for 5645x family of devices, the ports resulting from flex-IO operation did not initialize in disabled state. With this fix for 5645x family of devices the flexed ports will be initialized in disabled state when build flag <code>BCM_PORT_DEFAULT_DISABLE</code> is used.
SDK-63720	846795	56850_A2	Fixed the two issues below : 1) Inports qualifier not adding loopback ports to default entry 2) loopback secondary selector value added for tunnel and lookbacktype when not required
SDK-63727	846581	56640_A0 56640_A1 56640_B0	Issue: Calling <code>bcm_cosq_gport_bandwidth_set()</code> with argument cosq value greater than 31 returns an error. Fix: The numq value is changed to 64 to support maximum cosq limit of 64 in <code>bcm_cosq_gport_bandwidth_set()</code> .
SDK-63729		All	DMA swapping settings for communication with embedded applications (uKernel) are now auto-detected rather than set based on compile-time SDK options.
SDK-63746	847053	All	Enabled <code>XLMAC_MACSEC_CTRL.MACSEC_TX_CRC_CORRUPT_EN</code> to avoid the undesirable behaviors with which the downstream link partner still receives a packet with good FCS even though the packet is originally a bad one.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-63756		56640_A0 56640_B0	bcmSwitchModuleType switch control information was lost after warmboot which led to mis-programming of NONICAST_TRUNK_BLOCK_MASK in trunking. Initialized switch control to XGS3 by default to correct non unicast trunk member programming across modules to address this issue.
SDK-63760		88650_A0 88660_A0	OAM: The default destination for multicast DMM, LMM packets trapped by a MEP is the same as unicast DMM, LMM packets (such packets are legal according to Y,1731).
SDK-63774	845586	56340_A0 56340M_A0 56547_A0 56548_A0 56640_A0 56640_A1 56640_B0	Implemented MTU programming on TR3/HX4 through bcm_multicast_control_set/ bcm_multicast_control_get API.
SDK-63792		56850_A1 56850_A2	In earlier releases, INITIAL_PROT_NHI_TABLE entry could not be cleaned up after primary egress object replace. This issue has been fixed in this release.
SDK-63816	846855	56640_A0 56640_A1 56640_B0	Even though the service queues were configured as Strict Priority, the traffic pattern observed was Round Robin. This behavior was seen because the service queues did not have any limits set leading to queue starvation. The fix is to enable dynamic limit threshold which would limit buffer usage of the service queues and reducing probability of starvation.
SDK-63839	847494	88650_B0 88650_B1 88660_A0 88670_A0	Composite HR allocation (region type is 2) fails due to wrong mapping. Fixed.
SDK-63842	846697	88030_A0 88030_B0	Note.
SDK-63848	847213	56450_B1 56450_A0 56450_B0	A crash was seen in bcm_cosq_mapping_set during warmboot, because the SDK was releasing a lock which was never acquired. The code has been corrected to first get the lock and only then free it.
SDK-63887	817872	88660_A0	OAM: When creating accelerated Up Endpoints the restriction that the LSB of the src_mac address must match the local port has been lifted.
SDK-63918		88660_A0 88670_A0	For mirrored and snooped packets (sent to CPU), the user may want to retrieve the original destination. A new mode (not compatible with stacking systems) allows to stamp the destination of the original packet on the FTMH.DSP-Extension, by setting the ftmh_dsp_extension_add SOC property.
SDK-63921		88660_A0	In initialization we calculate a value to configure the register IDR_OCCUPIED_DBUFF_THRESHOLD field MMC_DBUFF_OCC_TH, but this configuration is never written to the register. The configuration is needed since if MMC buffers run out, UC buffers are used (if allowed by configuration). If this happens in the middle of the packet, packets are lost. Configurable threshold for buffer changes from MMC to UC when MMC occupied reaches threshold.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-63958		88650_A0 88650_B0 88650_B1 88660_A0	In the previous release, ARAD_PP_EG_QOS_IP_DSCP_NOF_BITS was defined incorrectly as 7, so the DSCP range was limited to 127. In this release, this issue has been addressed by defining ARAD_PP_EG_QOS_IP_DSCP_NOF_BITS from 7 to 8.
SDK-63961	844686	56150_A0 56340_A0 56340M_A0	Fixed inconsistent enabled status of Message-signaled interrupts among SDK drivers.
SDK-63986	837121	All	BFD: support accelerated BFD endpoints over LAG spanning over multiple device. For further details see user manual.
SDK-64009		88650_A0 88650_B0 88650_B1 88660_A0	Fixed static analysis items. (minor).
SDK-64016	843408	88660_A0	FORWARD_GROUP object in bcm_mpls_port_create, bcm_vlan_port_create was always created without id, that is the user couldn't provide the ID of the resource (FEC).
SDK-64017	840785	88660_A0	VPLS: fixed bcm_mpls_port_delete failure when called with forwarding group object.
SDK-64047	849780	56450_A0 56450_B0	Prevent occasional system crash caused by invalid access to an already freed semaphore. This could happen if the counter thread did not exit within the specified time and would typically happen if the counter thread runs at a very low priority.
SDK-64056		56640_B0	In previous releases, the memory index of ING_FLEX_CTR_COUNTER_TABLE [0 to 15] and EGR_FLEX_CTR_COUNTER_TABLE [0 - 7] was used as the entry index of these tables. The entry index was wrong because only the lower 12 bits of the memory index should be used as the entry index. This issue has been fixed in this release.
SDK-64129		88670_A0	For API bcm_l3_ingress_create, the resolution for the disable routing was according to UC / MC / MPLS. This new improvement gives better resolution for disable routing. Now it can be disabled by IPv4 UC / IPv4 MC / IPv6 UC / IPv6 MC / MPLS. The SDK support up to 16 combinations.
SDK-64137		88650_A0 88660_A0	In Field TCAM, the protection mechanism for parity errors is not enabled for Egress PMF. This is fixed.
SDK-64156		56450_A0	Problem definition On the ports supporting 8 priority groups (PG0-PG7) , SDK applies the PG limits to Priority group 0- PG0, This will cause incorrect loss less behavior as there is no PG min and headroom available in PG7 and SDK further configures the PG min on all unused PGs to a value of 1 on these ports. Solution : PG limits are set to PG7 for the ports supporting 8 priority groups(PG0-PG7) and removed the additional setting of PG MIN limit to 1 for all unused priority group for these ports.
SDK-64161		88660_A0	When deleting one of 4-size consecutive VOQ connectors group, the other group stop sending credits. The issue is fixed. Note that cold boot is required to get this fix. As a workaround bcm_cosq_gport_connection_set can be called again on egress.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-64202		88660_A0	MPLS PORT: <code>bcm_mpls_port</code> apis with flag <code>BCM_MPLS_PORT_EGRESS_ONLY</code> and <code>BCM_MPLS_PORT_FORWARD_GROUP</code> must be set with <code>vpn</code> parameter invalid. Error is generated if this parameter is valid.
SDK-64203		88660_A0	In L3, when SOC property <code>bcm886xx_l3_ingress_urpf_enable</code> was set, an assertion occurred in some cases due to incorrect usage of mapping from VSI to RIF. This is fixed now and VSI to RIF mapping is consistent. Note that this relevant only to ARAD-Plus device.
SDK-64212	849529	88030_A0 88030_B0	Note.
SDK-64254	850838	All	Improved <code>bcm_cosq_gport_connection_set</code> performance on MESH configuration.
SDK-64279	851155	88030_A0 88030_B0	Note.
SDK-64280	839118	88650_A0 88670_A0	Queue flow control accuracy improved by reducing burst size. Also there is an API to set burst size for a queue: <code>bcm_cosq_control_set(...)</code> <code>gport</code> should be of type <code>PORT_TC</code>
SDK-64298		88650_A0	In FCoE, when calling API <code>bcm_fcoe_route_add()</code> with flag <code>BCM_FCOE_DOMAIN_ROUTE</code> and prefix length = 8 the API returns error <code>BCM_E_INTERNAL</code> . The return value has been changed to <code>BCM_E_PARAM</code> in this case.
SDK-64328	851983	5600_A0 56450_B0	For the memory profile <code>egr_mpls_combo_map</code> , table count is 3. The memory tables are as follows, 1. <code>EGR_MPLS_PRI_MAPPING</code> 2. <code>EGR_MPLS_EXP_MAPPING_1</code> 3. <code>EGR_MPLS_EXP_MAPPING_2</code> We are passing the <code>entry_array</code> with its size as 2 to hold the entries for the above tables in function <code>soc_profile_mem_get</code> . Segmentation fault occurs when we access the third index in <code>entry_array</code> since table count is 3. When we increase the size of array to 3 in <code>_bcm_common_sw_dump</code> , "dump sw common" works fine.
SDK-64332		88650_A0 88660_A0	OAM: Support creating endpoints of type <code>bcmOAMEndpointTypeBHPw</code> .
SDK-64348	850276	88030_A0 88030_B0	Corrected the warm boot code that writes and restores data for the field module.
SDK-64389	852433	56450_A0 56450_B0	Issue reported was that if an attach of a node to a parent node fails due to resource exhaustion, the successive call to delete the node also fails. This was because upon failure, the SDK was not resetting the cos bitmap and child cosq of the parent node. A check for invalid <code>hw_index</code> is also added to prevent improper hardware access.
SDK-64410	848156	All	Fix VSI aging profile allocation.
SDK-64416	849492	88030_A0 88030_B0	Note.
SDK-64476		53406_A0	SDK driver for BCM534xx implemented to configure/initialize frequency synthesizers and timestampers and enable PTP feature.



Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-64515		88660_A0	In 1588, an increased resolution to the stamping is introduced (from 32b to 48b). The <code>bcm88660_1588_48b_stamping_enable</code> SOC property must be set (values: 1 - 48b stamping, 0 - 32b stamping. the default is 0).
SDK-64517	852696	All	Fixed regression failure that caused statically assigned PTP slaves to be sent packets with an incorrect local address.
SDK-64524	809817	88750_B0	In a multistage system resetting or isolating the FAP devices might cause drops over FE13. In order to apply the fix set the following soc property to 1 - <code>custom_feature_alrc_slow_link_down_fe13</code>
SDK-64537	852687	56547_A0	Idle threshold is now configurable for Gecko
SDK-64559		88660_A0 88670_A0	Fixed a bug in 88660 causing dropped ingress multicast traffic in scenarios with a high load of ingress multicast traffic. In 6.4.3 This is fixed In 6.3.11 This is fixed if using either <code>maintenance_default_override=1</code> or <code>maintenance_default_override_63x_ingr_mc_fifo_thresh=1</code>
SDK-64577	852417	All	OAM: creating more than 3 MIPs and then calling <code>bcm_oam_endpoint_action_set()</code> on each one fails due to lack of resources.
SDK-64587		88650_A0 88660_A0	OAM: For the following OpCodes, multicast packets are supported by default: AIS TST LCK APS CSF SLM
SDK-64622		88670_A0	MPLS: Add UHP support. This improvement enables allocating one In-Lif per MPLS label range given by the user. For more details see PP User Manual.
SDK-64661	852995	All	In earlier releases, there was dead lock between function <code>bcm_vlan_control_vlan_set</code> and <code>bcm_l3_egress_create</code> . The dead lock has been destroyed, so this JIRA is resolved.
SDK-64686		53400_A0	Added support for QOS mapping type <code>BCM_QOS_MAP_L2_VLAN_ETAG</code> for BCM53400 family.
SDK-64739		88670_A0	VLAN PORT: Support use of split horizon groups for AC lifs
SDK-64742	854717	88660_A0 All	OAM: classifier advanced mode 2 (soc property <code>oam_classifier_advanced_mode=2</code>) supports endpoint md-Level 0. Error message removed.
SDK-64763		88650_A0 88650_B0 88660_A0	OAM: when calling <code>bcm_oam_endpoint_create()</code> with a non accelerated Down endpoint of level 0, the id (returned as output) will be {1'b1,6'b0,16 bit LIF ID}. This fixes failure that is generated in case accelerated endpoint is added with same id.
SDK-64804		88660_A0	<code>bcm_vswitch_destroy_all</code> failed when mpls tunnel termination (<code>bcm_mpls_tunnel_switch_create</code> with action POP) entries exist.
SDK-64864		88660_A0	VPLS: fixed <code>bcm_mpls_port_delete</code> failure when called with forwarding group object.

Table 139:

Number	CSP #	Chips	Release Notes For 6.4.3
SDK-64872		88650_A0	When a channelized port is removed and instead added single NIF ports, the serdes isn't initialized correctly, causing for wrong lane rate - Fixed.
SDK-65431		56960_A0	Added support for Ingress Field Processor(IFP) actions on Tomahawk. bcmFieldActionDynamicHgTrunkCancel bcmFieldActionTrunkLoadBalanceCancel bcmFieldActionEcmpLoadBalanceCancel

Section 13: Resolved Issues for 6.4.2

The following issues are resolved in version 6.4.2 of the SDK.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-31854		All	The set and get versions of <code>bcm_rx_redirect_reasons</code> have been updated to return the correct error codes for wrong unit or wrong mode.
SDK-44471	599747	56544_A0	the lanes has to be single lane in boot up. The config property <code>portgroup_0=1</code> will configure the ports single lane.
SDK-45222	614537	56440_B0	<code>mis_con_mep_id</code> and length fields are added in endpoint info structure
SDK-45223	614521	56440_B0	<p>Problem: If user configured, source <code>mep_id</code> on both DUTs differently, than the BFD session was not able to be established properly intermittently. To debug this the user requested that the SW display/validate the Remote DUT <code>mep_id</code>, which was configured incorrectly.</p> <p>Solution: New fields are added in BFD endpoint info as <code>remote_mep_id</code> and <code>mis_con_mep_id</code> to achieve this. After enabling this feature, user needs to configure on both DUTs, <code>mep_id</code> and <code>remote_mep_id</code> and as well as length fields.</p> <p>If user mep configuration is incorrect, wrong mep id will be stored in <code>mis_con_mep_id</code>. From BCM shell, user has to invoke <code>bcm_bfd_endpoint_get()</code> API to display wrong mep id.</p>
SDK-45253		56643_B0	For flexible HG ports port mode register <code>xport</code> field should be set to 0 (Single mode). Setting it to zero removes the underflow issue. Fixed the same.
SDK-46712	634930	56840_A0	MC Prio2Cos values have been corrected to reset properly
SDK-47231	643758	56440_A0 56445_A0 56440_A1 56445_A1 56444_A1 56449_B0 56445_B0 56440_B0 56447_B0 56443_B0 56441_B0 56446_B0 56448_B0	In the previous release packet was not getting received while switching from IEEE mode to HIGIG mode. This was because the <code>STRICT_PREAMBLE</code> and <code>PROCESS_VARIABLE_PREAMBLE</code> in <code>XMAC_RX_CTRL</code> was not configured correctly for HG port. This issue has now been addressed by setting <code>STRICT_PREAMBLE=0</code> and <code>PROCESS_VARIABLE_PREAMBLE=0</code> in <code>XMAC_RX_CTRL</code> when port type is HG.
SDK-48892	652937	56850_A0	Fixed HSP port attach for Y pipe based ports.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-49048		88650_A0	Qpairs allocation of CPU ports is done from a specific qpairs range of 192-255. With this fix the qpairs allocation is done according to CPU channel. For each CPU port the following range of qpairs is allocated: [192+channel, 192+channel+nof_priorities-1] .
SDK-49169		56850_A0 56850_A1	fix code to display correctly ecmp info for host route.
SDK-49620	684645	56850_A0	Fixed addition of a member to an empty LAG, Hlgig trunk or ECMP group, when resilient hashing is enabled. This fix is applicable to BCM56850 device.
SDK-49630		56850_A0 56850_A1 56850_A2	The field actions contains the list of valid and invalid actions (Invalid actions are either removed actions or modified actions). Modified the code to scan the entire list instead of scanning till the first invalid action in the list. This allows the SDK to retain the valid actions when unwanted action is removed
SDK-49910		56640_A0 56640_A1 56640_B0	The issue is that the function <code>_bcm_tr3_cosq_node_get</code> expects a GPORT node to be passed. However, for the cpu node we don't pass a gport and this returns BCM_E_PORT. The fix is to get the sched index via <code>_bcm_tr3_cosq_index_resolve</code> in the alternate path.
SDK-50311	693316	56850_A0 56850_A1 56850_A2	In earlier releases, the mask in flex hash entry could not be configured with existing API. This has been resolved.
SDK-51769	691762	56840_A0	In earlier release, when HIGIG port speed was 40G and external phy was in reverse mode, the SOP, SOM and sequence alignment were not programmed correctly. This has been resolved.
SDK-52237	721240	All	Support to handle concurrent RPC requests is added . New Configuration variable "rpc_server_thread_count" is added. This variable holds the number of RPC server threads created.
SDK-52679	726763	56450_A0	Problem Statement: Previously we could not support movement for L0 node from one port to other port Solution: SDK provided a new API, <code>bcm_cosq_gport_reattach</code> which will allow user to move L0 node from one port to other

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-52710	729008	56224_B0	<p>On UNIMAC devices, calling <code>bcm_stat_init()</code> API was overwriting the maximum frame setting on port when configuration for Jumbo packet handling using <code>bcm_stat_jumbo</code>.</p> <p>This configuration will set the size used by counter collection to decide when to count over-sized packets. When call <code>bcm_stat_init()</code> the maximum frame size was being set to this jumbo size setting overwriting the existing max frame setting on a port.</p> <p>Removed this code for setting the max frame size during <code>bcm_stat_init</code>. The counter collection for over-sized packet has been fixed to take into account the oversize stats.</p>
SDK-53222 SDK-52915	733469	56450_A0	<p>In Katana2 the capability to display the IIs tree of the coe port was missing and It has been added</p>
SDK-53223	733490	56450_A0	<p>Shaper configuration was not reset when the nodes in scheduler tree were deleted or detached in the tree. Added code to reset the shaper configuration when the nodes in scheduler tree are deleted or detached.</p>
SDK-54341 SDK-57678		56640_A0 56344_A0	<p>When changing the parent of any L2 node from WDRR to SP, <code>BCM_E_TIMEOUT</code> is thrown, since dynamic change of SP<->WRR is not supported. The current way to resolve this is: Customer calls APIs to stop enqueue and drain the mmu queues before changing the scheduler mode.</p> <p>With this method, the customer is expected to call the following APIs to stop the enqueue and drain the mmu before detaching/attaching to a parent with different scheduler mode.</p> <pre>bcm_port_control_set(unit, port, bcmPortControlMmuTrafficEnable, 0); bcm_port_control_set(unit, port, bcmPortControlMmuDrain, 1); <Detach/attach/sched_set> bcm_port_control_set(unit, port, bcmPortControlMmuTrafficEnable, 1);</pre>
SDK-54758	750349	56450_A0	<p>Problem statement: <code>bcm_cos_mapping_set</code> is not working for flex ports Fix description: Added code so that flex ports are also used during <code>config_set</code> and <code>mapping_set</code> profiles, as a result <code>bcm_cos_mapping_set</code> is working appropriately</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-54796	751218	88650_A0 88650_B0 88650_B1	<p>When allocating a combination of 1-priority and more than 1-priority ports that cover the entire q-pairs range (256 q-pairs), 1 -priority port can't be allocated on the last q-pair. The driver allocates the qpairs in ports order, so if last port is 1-priority port error will be returned.</p> <p>To avoid this issue user can allocate one of the ports which aren't 1-priority to use the last qpair(s).</p> <p>The soc property is: otm_base_q_pair_<port_id>=<base_q_pair></p>
SDK-54889	750468	88030_A0	<p>Default route disappeared is caused by the bucket merge was not supported by taps driver. After we improve the taps error handling and support to merge the bucket in that handling, this issue could be resolved.</p>
SDK-54991	750024	All	<p>Added support for filter callbacks in the Linux KNET kernel driver.</p>
SDK-55162		88660_A0 88670_A0	<p>IP Routing-Over-Overlay (ROO) refers to a set of protocols/applications where the L2 forwarding to the Host/Next-Hop router is not accomplished by simple 802.1q bridging, but by L2-Overlay protocols (VXLAN, etc). BCM8866X supports ROO Host Unicast over VXLAN. See <code>cint_vxlan_roo.c</code> for cint example and Programmer's Reference Guide for more details.</p>
SDK-55180		All	<p>In non-interface map mode, <code>bcm_l3_intf_create()</code> and <code>bcm_vlan_control_vlan_set()</code> can overwrite vrf values set by each other. This is how these apis are designed to set up the parameters that they are overstepping each other. Added documentation to explain the behavior of APIs in case of VRF setting.</p>
SDK-55184		56850_A0	<p>Earlier SDK releases did not allow configuring MTU value for vxlan access ports. This release now supports setting/resetting MTU for vxlan access ports through <code>bcm_vxlan_port_add()</code> API.</p>
SDK-55454	758146	88650_A0	<p>In Counter Processor, one of the Ingress-PP counting mode is VSI. A support for counting packets by ingress VSI is added. The relevant soc property is <code>counter_engine_source <counter ID>=INGRESS_VSI</code>. When set, the default Counter-ID value for Ethernet packets is VSI. This value may be overridden by Ingress PMF (Field Processor APIs with <code>bcmFiedlActionStat0/1</code> actions) or by Rx-traps.</p>
SDK-55607	752009	56640_A0 56540_A0 56640_A1 56640_B0 56540_B0	<p>When OAM LM endpoint is deleted on Triumph3, feild group is also deleted if there are no more OAM LM endpoints present in that group. On deleting the field group, field group index was not set to invalid. This issue has been fixed.</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-55706	730810	56540_A0 56540_B0	Issue :- "fp show" and "listmem fp_tcam" are reporting incorrect entry values for IFP in Apollo2 Fix :- Updated "fp show" command to return correct entry count for IFP in Apollo2. "ListMem fp_tcam" cannot be fixed as the command reads from driver. Modifying the driver values is leading to other issues.
SDK-55886	760333	All	In previous releases, out-of-bounds access to an array of multicast group was allowed, which would result in crashing. This has been resolved by placing a proper check to ipmc id.
SDK-55929	759404	56620_B0	In earlier SDK releases, the SDK did not properly decode the dest modid and dest port in the Triumph and Triumph2 class of devices. The issue is fixed by appropriately accounting for the addressable port and module id range on the device.
SDK-56001	761280	All	soc_feature_timesync_support flag enables 1588/PTP protocol's transparent clock support. This JIRA enables PTP transparent clock support for Triumph 3.
SDK-56024		56850_A0	There's a bit in the VLAN_XLATE table called VLAN_ACTION_VALID, It must be enabled to process XLATE_DISABLE_VLAN_CHECKS for VXLAN virtual ports, but disable it for VXLAN access ports to drop packets at ingress. They have conflict. To solve the problem, a new flag has been added that allows the customer to control the bit, The new flag is BCM_VXLAN_PORT_ENABLE_VLAN_CHECKS.
SDK-56065 SDK-61361		88670_A0	In BCM88660 and below, any FEC has both a destination and a forwarding information (e.g., Out-LIF). The destination cannot be another FEC. In BCM88670, the destination of a FEC can be another FEC (Hierarchical FEC). There is an hardware limitations on this feature - a pointing FEC may not be accessed in the same bank as the pointed FEC (In hardware the FECs have been divided into banks). Due to this limitation the FECs must be allocated in such a way that a pointing FEC and a pointed FEC will never be in the same bank. A CASCADED flag has been created for any API that allocates FECs (bcm_l3_egress_create, bcm_vlan_port_create, bcm_mpls_port_create) to indicate that a FEC will be pointed by another FEC. If the CASCADED flag is not specified, the FEC can be used as a regular FEC or a pointing FEC. For more details see the Packet Processing User Manual.
SDK-56099	766312	56334_B0 56334_A0	Repaired EGR_VLAN table corruption during hash table entry balancing in Enduro devices.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-56203		88650_A0 88650_B0 88650_B1 88660_A0	<p>XLPORT Overrun/Underrun Workaround ----- -----: The Arad driver implements a sequence to recognize and recover the port from XLPORT Overrun/Underrun issue (see BCM88650 errata sheet). To activate the sequence during device init use the following soc property: <code>custom_feature_nif_recovery_enable=1</code> (default is disabled on 6.3.x, and enabled on 6.4.x).</p> <p>The sequence might perform several iterations when trying to recover the port. To limit number of iteration use the following SoC property: <code>custom_feature_nif_recovery_iter</code> (default is 3). Note that from lab experience the port is recover within single iteration.</p> <p>Limitations: 1. The SW WA works for XLP0 only. 2. The SW WA is called during init and isnt available for dynamic port.</p>
SDK-56317		56846_A0 56846_A1	<p>In previous releases, created multipaths more than max capacity could corrupt existing ECMP groups and return wrong value -1 if ECMP group size of TD+ configured to 256 as TD device. In this release, it returns <code>BCM_E_FULL (-6)</code> if creating ECMP multipaths more than max capacity.</p>
SDK-56424	767587	56850_A0	<p>Problem: Modified srtcm meter mode support was missing in Trident2 for Egress FieldProcessor Stage</p> <p>Solution: Added modified srtcm metering support in coldboot and warmboot for Egress Field Processor Stage in Trident2. This includes policer creation, attach and install the configs. It also includes recovery of the meter during warmboot.</p>
SDK-56462	770876	56640_A0 56640_A1 56640_B0	<p>Added the software support for 100G Remote loopback</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-56589	771496	56450_B0 56455_A0 56456_A0 56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56640_A0 56440_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56843_B0 56841_A3 56445_A0 56440_A1 56445_A1 56444_A1 56450_A0 56846_A1 56841_B0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0 56449_B0 56445_B0 56440_B0 56447_B0 56443_B0 56441_B0 56446_B0 56448_B0 56442_B0	Customer found the counter RDVLN did not work without configuring XMAC_RX_VLAN_TAG register. They also found there was no way to set this register in previous SDK release. Now an improvement has been implemented to set the register by bcm_port_tpid_set() & bcm_port_inner_tpid_set() APIs.
SDK-56619	771355	56846_A1	Issue: In earlier releases, no fault message was sent in single fiber connection for issue in phy_wc40_software_rx_los(). Root Cause: The rx_los_state was never set to RESET. So LOCAL_FAULT_ENABLE was never set as the RESET case was never hit. Fix: Allow the rx_los_state to set to RESET and subsequently move into the RESET case to set rx_los_state to START_TIMER.
SDK-56776	772447	56440_A0 56440_B0 56450_A0 56450_B0 56850_A0	Issue: When DK_MEM flag is enabled, we would access L2X views array during L2 initialization. We would use KEY_TYPE as index in L2X views array. KEY_TYPE field is 3 bits wide in KT/KT2 and 4 bits wide in TD2. Hence it is set to 7 in KT/KT2 and 15 in TD2. For KT,KT2 and TD2, KEY_TYPE is greater than maximum index of L2X views array and it may result in segmentation fault while accessing this array with index greater than maximum index. Fix: In soc_mem_entry_dump_common, for KT/KT2/TD2, if key_type is greater than maximum index of L2X views array, then key_type value is set to the respective L2X view array maximum index value.
SDK-56787	775384	All	In earlier releases, there is no way to use the API bcm_rate_bandwidth_set to set the rate to 0kbps speed. If the pass the speed as zero, the assumption was to disable metering. Now, to disable metering, kbits_sec parameter has to be passed as BCM_RATE_DISABLE and to set the rate to zero, kbits_sec parameter has to be passed as BCM_RATE_BLOCK

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-56916	775184	56340_A0	Problem: Regex match was not being reported if the pattern occurs beyond 2048 bytes for any given session. The reason was that the <code>AXP_SM_SIGNATURE_MATCH_CONTROL.MAX_BYTE_INSPECTED</code> is always being set to the default value of 2048. Solution: Set the above field/register to the payload depth provided through <code>bcm_regex_config_set</code> API. This will be set in sync with the <code>FT_CONFIG.MAX_BYTES</code> field.
SDK-57000	755118	56850_A0 56850_A1 56850_A2	Support has been added for VPLAG for VxLAN based VPs.
SDK-57024		88650_A0	Dynamic port provisioning feature added
SDK-57146	779181	56850_A0 56850_A1 56850_A2	Description: <code>MAP_TABLE</code> is not correctly indexed in <code>bcm_esw_switch_rcpu_encap_priority_map_set()</code> Root cause: It's a typo. Solution: The software now corrects the typo.
SDK-57171	780263	56850_A0 56850_A1 56850_A2	Originally, <code>VLAN_XLATE</code> entry was not cleared possibly when deleting a VP. It is fixed by setting <code>modid</code> or <code>trunk_id</code> to an invalid value (-1) in <code>_bcm_td2_vxlan_match_add</code> when creating a vxlan logical port on a physical port or a trunk.
SDK-57197 SDK-62313	776292	56620_B0	Description: A parity entry in <code>L3_DEFIP_DATA_ONLY</code> can't be cleared by SDK-6.3.6 Root cause: In the earlier software, the table <code>L3_DEFIP_DATA_ONLY</code> is not cached, so the SER engine returns not available, and hence the corrupted entry would not be cleared. Solution: Current software has the <code>SOC_MEM_FLAG_SER_CACHE_RESTORE</code> set to <code>L3_DEFIP_DATA_ONLY</code> , once an SER error was detected on an entry of <code>L3_DEFIP_DATA_ONLY</code> , the ser correction engine will clear/restore the whole <code>L3_DEFIP</code> entry.
SDK-57205	779120	56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56843_B0 56841_A3 56846_A1 56841_B0	In previous release, <code>bcm_stat_get/clear()</code> APIs could only handle software counters on CPU port although chip has hardware CPU counters. Now these APIs have been improved to support hardware CPU counters. This improvement only works on TD, TD2, TH chips now.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-57207	777630	56640_A0 56640_A1 56640_B0	Issue: Packets of size 64 to 75 bytes getting dropped for XE ports. Root Cause: The runt threshold value for XE ports was getting set as 76 instead of the correct value 64. Hence packets of size 64-75 bytes were getting dropped. Fix: For Triumph3 and Katana2, put explicit checks to ensure that runt threshold value is set to correct value, i.e. <code>RUNT_THRESHOLD_XE = 64</code> , <code>RUNT_THRESHOLD_GE = 64</code> and <code>RUNT_THRESHOLD_HG = 76</code> . Also optimized the function <code>mac_x_init</code> for multiple READ and WRITE for <code>XMAC_RX_CTRL</code> and <code>XMAC_TX_CTRL</code> . Added a single write common for all devices instead of multiple instances as was present previously.
SDK-57250		88660_A0	PON ONU tunnel-ID allocation: So far PON application reserved last 512 tunnel-id entries of PON port 7 by ARAD. An error occurs when trying to set tunnel ID profile mapping. Fixed mapping to allow up to 2016 tunnels-IDs (0-2015) for all PON ports. TX behavior has been changed since this improvement. PTCHs(packet termination control header) of packets sent from CPU should always have SSP(source system port).
SDK-57306	781729	88650_A0 88670_A0	Change <code>dram_crc_del_buffer_max_reclamis</code> description in Arad/Jericho UM TM. No code changes.
SDK-57324		56640_A0	<code>bcm_policer_group_create</code> API creates a group of policers which are mapped based on the input offset derived from the packet. This mapping is fixed and is not flexible. New API <code>bcm_policer_group_create_with_map</code> provides the flexibility to offset derived from the packet and policer flexibly. This mapping can be given as input through the structure <code>struct bcm_policer_map_t {int count; /* Number of entries in the map */ uint8 *offset_map; /* Array of policer offsets corresponding to the offset derived from the packet */}</code> Syntax of the new API: <code>int bcm_policer_group_create_with_map(int unit, bcm_policer_group_mode_t mode, bcm_policer_t *policer_id, int *num_policers, bcm_policer_map_t *offset_map);</code>
SDK-57344	780748	All 56548_A0 56546_A0 56545_A0 56544_A0 56542_A0 56541_A0 56540_A0 56545_A1 56540_B0 56541_B0 56546_B0 56544_B0 56547_A0 56545_B0 56542_B0	Fixed KNET DMA locking issue which could cause duplicate packets to be sent from the <code>bcm_tx()</code> API.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-57527		56640_A0 56850_A0	In case of TD2, CMIC free running clock was pointing to external clock even if no external clock is attached to unit. Fix is provided to address the issue so that internal reference clock is used if no external clock is present on reference pad.
SDK-57723	787163	56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56850_A0 56855_A0 56843_B0 56841_A3 56846_A1 56841_B0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In the previous release, the validity of niv_src_vif was not checked when calling _bcm_trident_mirror_niv_tunnel_set. In this release, the niv_src_vif is masked with _BCM_TD_MIRROR_NIV_SRC_VIF_MASK when calling _bcm_trident_mirror_niv_tunnel_set
SDK-57777		56440_B0 56450_B0	new config variable mmu_num_subscriber_queues and a macro "BCM_GPORT_UCAST_SUBSCRIBER_QUEUE_GROUP_SYSQID_SET" are defined to form the desired hierarchy tree of subscriber queues on front panel ports.
SDK-57823	789287	56450_A0 56450_B0	Fixed the issue of access to non-existent fields in ECC error registers resulting in assertion during ECC interrupt handling by making sure that only the valid fields in the ECC error register are accessed.
SDK-57834	778524	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	Added support for 64bit statistics counters for BFD endpoints.
SDK-57846	788171	56840_A0 56640_A0 56850_A0 56640_A1 56640_B0 56850_A1 56850_A2	In earlier release DMA buffer used to download WC firmware was not freed upon detach or exit clean. This has been resolved.
SDK-57853		88660_A0	Trill warmboot. Sw state trill alloc link list size was not correctly calculated at warmboot trill restore, causing incorrect size after warmboot
SDK-57859	789891	56450_A0 56450_B0	Problem : Old TDM-B was not having flex capability and was not matching to customer requirement i.e. 16x2.5G + 1G(WC0) + 10G(WC1) Fix : New TDM-B is provided to meet 16x2.5G + 1G(WC0) + 10G(WC1).

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-57905 SDK-55025	790243	56548_A0 56547_A0	<p>Firescout had a few static modes that could be used to limit 128-v6 and v4 routes from the same shared space. This way of configuration was not flexible as the modes were limited.</p> <p>This improvement removes all the static modes and provides a mechanism which can help users to configure v6/v6 routes flexibly. User can setup limit at route level granularity with the new approach. When the limit is set for 128-V6 routes, then rest of the memory is given to v4/64-v6 routes. user can also configure whether to share 128-v6 routes space with v4 routes.</p>
SDK-57983	789226	56850_A2	<p>DMA source mode was not set after warmboot. This is fixed</p>
SDK-58048		53125_A0 53018_A0 53125_B0	<p>Fixed segmentation fault encountered when SDK was operated with KNET on the Northstar and Starfighter devices of Robo family.</p>
SDK-58055	788661	56450_A0 56450_B0	<p>Issue: Access to iproc unimac(ethernet) registers is failing.</p> <p>Rootcause: Access failed in iproc linux since <code>request_mem_region</code> in linux kernel returns NULL. <code>request_mem_region</code> would return NULL when the memory range for specific driver is occupied already. This is because the ethernet driver in iProc was already using the memory specified for unimac registers.</p> <p>Fix: After commenting out the <code>request_mem_region</code> in linux kernel, we can now read/write those registers.</p>
SDK-58073 SDK-67926	790115	56334_B0 56334_A0	<p>Port's enable status would no longer influence <code>soc_phyctrl_loopback_set</code> setting. While setting the phy loopback mode, only the port is enabled would do the actions which the port needs to enable.</p>
SDK-58083		88650_A0 88660_A0	<p>VLAN edit:</p> <p><code>bcm_vlan_translate_action_create</code> () defines a VLAN edit action per LIF in a standard VLAN edit mode. The API can be provided with a physical port and inner/outer VLAN values. The API didn't fail call with VLAN out of range values (>4095). A validation was added.</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-58118	791236	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56850_A0 56855_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In previous release, there was no support to configure RTAG7 MPLS L2 PAYLOAD L3 HASH FI ELD_BMAP.MPLS_L2_PAYLOAD_L3_BITM AP_A/B, RTAG7_HASH_CONTROL.MPLS_PAYLOAD_ HASH_SELECT_A/B on Triumph3, RTAG7_HASH_CONTROL_2.MPLS_PAYLOA D_HASH_SELECT_A/B on Trident2. This has been resolved by providing and implementing bcmSwitchHashL3L2MPLSField0/1, bcmSwitchHashL2MPLSPayloadSelect0/1.
SDK-58163		56850_A2	In previous release, invoking <code>bcm_trunk_set</code> to a existing Trunk or HiGig Trunk would cause flow set table to be totally re-computed. This has been resolved in a manner that the flows assigned to staying members will now be kept without shuffling.
SDK-58247	787119	53018_A0	Fixed incorrect use of tx channel index of IPROC platform in KNET driver on BCM5301x platforms.
SDK-58270	790312	56150_A0	The entry status of dual hash search mechanism on Hurricane2 and Greyhound devices were incorrectly reported in the case of hash conflict. This has been fixed in this release.
SDK-58329	791806	56850_A0 56850_A1 56850_A2	In the previous release, there are some static analysis issues found by customer in the ALPM module. Now those issues have been fixed.
SDK-58485	796744	All 56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56850_A0 56855_A0 56843_B0 56841_A3 56846_A1 56841_B0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In the previous release, the internal priority could not be mapped correctly from the field DSCP in the IP packets with the API <code>bcm_qos_port_map_set</code> on TRIDENT2. In this release, this issue has been addressed by programming the field <code>TRUST_DSCP_PTR</code> in the table <code>PORT_TAB</code> .
SDK-58501	794058	56850_A0 56850_A1 56850_A2	In the previous release, <code>bcm_fcoe_route_find</code> didn't support the return of a <code>l3_intf_id</code> . In this release, it supports the return of the <code>l3_intf_id</code> .
SDK-58525	794780	88650_A0 88650_B0 88650_B1 88660_A0 88670_A0	The corrective action for interrupt <code>FCT_UNRCH_DEST_EVENT</code> doesn't work for unit#0 - Fixed.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-58550	797110	88750_B0	<p>FE13 Isolation is intended for a planned removal of an FE card without affecting traffic. Executing the current isolation sequence on an FE13 while under traffic will cause traffic loss. The reason behind it is that the FE13 isolation sequence will cause the FE3 RTP table to be updated before the FE1 RTP table. Therefore traffic will still go into the fabric plane, but the FE3 will not be able to forward it, and will be forced to drop the traffic.</p> <p>Workaround: Isolate an entire FE plane, by masking the relevant links in the FDT across all the FAPs in the system. Then execute the isolation sequence on the FE13 card. This option will stop all data cells from running through the fabric, allowing a safe extraction of a single FE card. Fabric plane redundancy is required in this case.</p> <p>In order to stop/start sending traffic over a FAP link use the following API:</p> <pre>bcm_fabric_link_control_set(unit, link, bcmFabricLinkTxTrafficDisable, disable);</pre> <p>The isolate sequence should disable to links (across all the FAPs) that connected to the relevant fabric plane.</p>
SDK-58641	799444	56640_A0 56640_B0	Added support in SDK software to use NL 11K series 20M external TCAM with the new 56545K SKU.
SDK-58642	799204	56850_A2	<p>Issue :- Trident2 :Action bcmFieldActionIngressGportSet in VFP configures FIELDS_ACTIONf which is not required. Fix :- Added relevant checks to not configure FIELDS_ACTIONf for bcmFieldActionIngressGportSet in Trident2.</p>
SDK-58653		56649_A0 56643_B0 56648_B0 56640_B0 56854_B0 56644_B0 56854_A0 56649_B0 56644_A1 56643_A1 56640_A1 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1 56640_A0 56643_A0 56644_A0 56641_A0 56642_A0 56850_A0 56855_A0 56645_A0 56648_A0 56850_A1 56855_A2 56852_A2 56853_A2 56854_A2 56851P_A2 56851_A2 56850_A2 56851_A1 56851P_A1	<p>In previous release, there was no support to configure RTAG7 MPLS L2 PAYLOAD L3 HASH FIELDS_BMAP.MPLS_L2_PAYLOAD_L3_BITMAP A/B, RTAG7 HASH CONTROL 2_64.MPLS PAYLOAD_HASH_SELECT A/B on KATANA2. This has been resolved by providing and implementing bcmSwitchHashL3L2MPLSField0/1, bcmSwitchHashL2MPLSPayloadSelect0/1.</p>
SDK-58698	799433	56150_A0 53344_A0	An unexpected timeout of draining packets happened when unplugging the cable from TSC ports of BCM53344 platform. This has been resolved with correct port disable sequence.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-58699	795957	56640_B0	Although flexible scaling feature was supported in TR3, due to a bug in TR3 feature list, flexible scaling feature for LPM was not enabled. Applications were not able to add 128b routes due to this. Enabled flexible scaling feature in TR3.
SDK-58725	799988	56450_A0 56450_B0	In the previous releases when more than one MPLS port share the same egress queue and if one of the MPLS port sharing the queue was removed then <code>ING_QUEUE_MAP</code> was deleted as no reference counter was maintained on <code>ING_QUEUE_MAP</code> . From this release added support for maintenance of reference counter on <code>ING_QUEUE_MAP</code> entries. <code>ING_QUEUE_MAP</code> entry is deleted only when all MPLS ports sharing the egress queue entry are deleted.
SDK-58749 SDK-63028	800086	56450_A0 56450_B0	1G speed is advertised with XE port or ≤ 10000 only and not on HIGIG port After flex-io operation on hg port, max-speed still remains as hi gig speed ($\geq 13g$) and port is not treated as normal ethernet (ge/xg port). Speed Capability decision was based on max-speed. Due to this max- hi-gig speed, changing speed to 1G port was not possible. Now ORed max-speed with <code>IS_XE_PORT()</code> and issue is resolved
SDK-58750	792732	88650_A0 88650_B0 88650_B1 88660_A0	Trill Adjacency check: 1. <code>encap_id</code> (Link-layer-outlif) now support full range 2. Fix information of <code>bcm_trill_port_get</code> for Adj. database.
SDK-58848 SDK-58983	801222	All	Fixed major lock release issues in SDK code.
SDK-58855	778638	56850_A1	The test in this JIRA involved incomplete AN mode for a given port and that port was flexported to another port mode. Sometime HW requires a clean up to clear the incomplete AN mode (<code>CRTSC_616</code>). This JIRA fix adds the fix to clean up the incomplete AN mode prior to flex port transition.
SDK-58856	795594	All	The CLI command "dump pcic" has been made independent of the BDE functions and instead it would now use the <code>cm_device_t</code> function vectors
SDK-58886	802162	All 56334_B0 56334_A0 56850_A0 56850_A1 56850_A2	In earlier releases, internal resource (<code>next_hop/vx_and_swap</code> objects etc.) could not be deleted if customer invoked <code>bcmx_trunk_destroy</code> first and then <code>bcmx_mpls_port_delete</code> . This has been fixed in this release.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-58890	781333	56440_A0 56440_A1	For Katana (and Saber) the counter <code>SOC_COUNTER_NON_DMA_COSQ_DROP_PKT</code> is implemented with a flex counter which counts packets that are discarded by the MMU. By default, this counts packets that are discarded by the MMU for a variety of reasons. One of them is oversized packets. When the MMU flushes a packet because it is oversized, in increments this flex counter. This is not the desired behavior. The fix is to set the bit <code>OP_CNT_CFG</code> in <code>THDO_DROP_CTR_CONFIG</code> which would ensure that counters will only count packets that have been dropped due to output thresholding checks, i.e. those within <code>op_drop_mask</code> .
SDK-58895		88650_B1 88660_A0	This table is static table. Therefore shadowing this memory should be allowed. This table removed from dynamic table list.
SDK-58901	799623	88650_A0 88660_A0	When L2 frame has an 802.3 format where the Ethertype is a Length field, the EXP/DSCP value does not match TC/DP to EXP/DSCP mapping table. This happens when packet is bridged and encapsulated by PWE or IP tunnels. The above functionality conflicts with ECN capabilities. In case ECN is not required in the system we introduce two new SOC properties to disable ECN functionality: <code>mpls_ecn_mode</code> value 0-disabled, <code>ip_ecn_mode</code> value 0-disabled. In case ECN is required in the system we suggest workaround in FP <code>cint_field_ecn_cni_extract.c</code>
SDK-58906		88670_A0	Jericho Global LIF: Support the Jericho LIF differentiation between a Global LIF and a Local LIF.
SDK-58913		56850_A0 56850_A1 56850_A2	In the previous release, the SVP multiple MTPs mirroring and DVP multiple MTPs mirroring were not supported. In this release, this issue has been addressed by adding code changes to support these two mirroring modes.
SDK-58919 SDK-58191 SDK-58894	799048	All	BCM diag shell commands did not work for a system in which the total number of devices <= device ID (i.e. when they are 'holes' in unit IDs, for example - 2 units, IDs '0' & '2').
SDK-58920	802188	56450_B0	Earlier MSI EP mapping was implemented within the BDE (that supported MSI interrupts). Customers who did not plan to use the Broadcom BDE then had to implement the MSI mapping in their BDE code. This issue has been fixed by adding the MSI EP mapping to SOC function ' <code>soc_pci_ep_config</code> '.
SDK-58958 SDK-48392	802230	All	Added check for XW ports used in helix4 while enabling Frame length check.
SDK-58969	797443	56850_A0 56850_A1 56850_A2	In previous releases, the macro definition <code>MAC_LSB_SHIFT</code> was different for big endian and little endian system. Actually it should be same in both endian systems. This has been fixed in this release.
SDK-58981	798049	56850_A0 56850_A1 56850_A2	In earlier releases SDK destination port was modified unexpectedly when aged through <code>bcm_l2_replace</code> . This has been resolved.



Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59047	802819	56450_B0	In BCM 5645X chips after flex port operation the packets will not egress if the port uses external memory. Resolved by adding the missing port reinit for mmu after flex port operation.
SDK-59048	803885	56450_B0	<p>Description: Unexpected Parity Error on LLS_ERROR register.</p> <p>The root cause of this issue is as following: When changing port loopback mode from LB=MAC or LB=PHY to LB=NONE, the operated port will link down after a while and in the time gap, traffic won't be stopped forwarding to the operated port until link down happens to it. Meanwhile, the mode changing sequences will modify the MMU associated to the operated port, this causes the MMU issue.</p> <p>Solution: When changing port loop back mode from LB=MAC or LB=PHY to LB=NONE, adding code to force the port to link down to prevent traffic being forwarded to the port which is being operated.</p>
SDK-59058	790799	56640_A0 56640_A1 56640_B0	This is an improvement to clear initialize the external TCAM (NL11K) that can be used in conjunction with 56640 chips. The entire TCAM database is cleared using a fast initialization sequence in order to keep the time consumed to a minimum.
SDK-59066	793397	56850_A0	In previous releases, the customer could experience a segment error when using stacking application sample code to configure symmetric HG stack links on TD2. This was caused by the SDK using an incorrect maximum number of fabric trunk groups. The issue has been fixed.
SDK-59074	800863	56846_A0	SER injection is accomplished by disabling parity, injecting error into hardware table, and then enabling parity again. In earlier SDK releases, soc_trident_pipe_select() which is used to set pipe-line on Trident was invoked after disabling parity. So the parity control register on Y-pipe was not configured and parity was not disabled on Y-pipe. In this release, soc_trident_pipe_select() has been placed before setting parity control register to disable parity.
SDK-59087	804148	56440_A0 56450_A0 56450_B0	In the previous release the STRICT_PREAMBLE and PROCESS_VARIABLE_PREAMBLE was not configured for speed less than 2.5G. This issue has now been addressed by setting STRICT_PREAMBLE=0 and PROCESS_VARIABLE_PREAMBLE=1 in XMAC_RX_CTRL for speed less than and equal to 2.5G
SDK-59101	790579	All	Fixed Linux kernel link errors appeared when Makefile did not include DNX chips.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59154		56850_A0 56850_A1 56850_A2	In previous releases, in <code>bcm_l3_route_add</code> API if parameter flags and L3 interface object consistency were not checked for ECMP and could lead to wrong reference count update. In current release, parameter flags and L3 interface object consistency are checked for ECMP.
SDK-59155		88670_A0	Added SOC properties for configuring core symmetrically, asymmetrically, and number of active cores.
SDK-59178	803680	88750_B0 88650_B1	This issue fixed at 6.3.7 and 6.4.2 versions. The fix is solving a problem which produced at SDK-58398 (masking all interrupts by default).
SDK-59194	804732	All	In the previous release, the left egress mirror could not work if one of the two egress mirrors was deleted when flexible mirroring destinations was enabled. In this release, this issue has been addressed by programming the register <code>MIRROR_SELECT</code> correctly when the egress mirror is deleted.
SDK-59198	800106	56846_A0 56840_A0	In previous release, "PORT" parameter was not retrieved correctly for registers per warpcore in the routines <code>_soc_trident_parity_reg_set</code> and <code>_soc_trident_parity_reg_get</code> on Trident+ devices. This would cause system crash when parity error occurred in these registers. The operation that calculates and gets the correct PORT parameter for registers per warpcore has been added for these two routines.
SDK-59211		88650_A0 88650_B0 88660_A0	MPLS: when using <code>bcm_mpls_port_get</code> field <code>vccv_type</code> was not filled (only value supported is <code>bcmMplsPortControlChannelTtl</code>). This is fixed.
SDK-59219		88670_A0	Support 2nd myMAC function in Jericho for ROO improvements.
SDK-59235		56640_A0 56640_A1 56640_B0	In previous releases, when trying to create an egress vlan entry on BCM5664X, for virtual ports, the API was giving an assertion, because it was trying to configure an invalid field in egress vlan xlate memory. This is fixed by correcting the field name.
SDK-59237		56334_A0 56304_B1 56150_A0	APIs required for supporting 1-step and 2-step timestamping for Hurricane2 devices has been added in this release.
SDK-59238		56450_A0	KT2 1588 Transparent timestamping support is added.
SDK-59250	770333	56640_A0 56640_B0	In earlier releases PFC on 100G port was not working correctly. When configuring HSP port on TR3 there is a 1 bit shift seen in cos mapping. Priority 2 is mapped to cos0, etc. This is because UC COS0 is tied to L0.1 and so on. This was fixed to assign L0 nodes such that first L0 node is 4X-1. HSP port consumes only 2 index of <code>FC_MAP_TBL</code> since it looks like the mapping in Tr3 would be similar to TD2 for HSP case. Thus the <code>hw_index</code> is changed to 236+8 for ce port 1.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59252		88650_B1 88660_A0	In FCoE module, when updating NPV port control, using API <code>bcm_port_control_set(unit, src_port, bcmPortControlFcoeNetworkPort, VALUE)</code> behavior is changed to be consistent with other devices flow. Now, to set port to source routing, VALUE should be set to '0'. For port to destination routing, VALUE should be set to '1'.
SDK-59263		56850_A0 56850_A1 56850_A2	Any read operations from CPU could result in the HW aging or L2_BULK triggered HW aging mechanism sometimes incorrectly deleting entries from the L2_ENTRY table. The SW triggered HW aging through L2_BULK operations was introduced to fix this issue. In order to protect L2_BULK operation from all read operations from CPU, not only will the SW aging thread grab the mutex locks for all the memory read operations needed before the L2_BULK operations to trigger the HW aging, but also has the highest priority in all SAL threads to void preemption by other SDK threads.
SDK-59266	806043	56846_A1	In previous releases the sizes of <code>l2_hitda_only</code> and <code>l2_hitsa_only</code> tables were 16K. The fix was to reduce <code>l2_hitda_only</code> and <code>l2_hitsa_only</code> tables to 8K, as each of them can contain the hit bits for 8 L2_entry and total L2_entry is 64K.
SDK-59271		88650_A0 88660_A0	add support for Trill Transparent Service. Enabled by soc property <code>trill_transparent_service</code> . Trill transparent service allows to carry up to one customer vlan in a trill FGL network This carried customer vlan should be ignored by Trill campus. Consequently: at transit Trill, for Trill Transparent Service in pruning mode, pruning won't involve the carried customer vlan. at egress Trill, for Trill Transparent Service, identify the local vsi won't involve the carried customer vlan. For more informations see <code>cint_trill_tts.c</code>
SDK-59286	804564	56629_B0 56624_A0	<code>BCM_COLOR_INNER_CFI</code> color setting for "bcm_port_tpid_add" API is not valid for TRX devices. TRX only allows OUTER_CFI to CNG mapping. But the API was not returning error. This has been fixed and will now reject <code>BCM_COLOR_INNER_CFI</code> color setting for "bcm_port_tpid_add" and return <code>BCM_E_UNAVAIL</code> .
SDK-59290	804355	56640_A0 56640_A1 56640_B0	IPv6 Packets were not hitting the external FP rule that matches SMAC and VRF for <code>_FP_EXT_ACL_L2_IPV6</code> mode. The issue is fixed by changing the offsets of qualifiers during external field qualifier initialization and by programming the correct offsets in LTR (Logical Table Registers) .
SDK-59307	804642	All 56450_A0 56450_B0	In the previous release the tr 140 test did not have an erring pattern print when BIST fails. The erring pattern print has been added now to locate the exact memory location it fails and the pattern used only when BIST fails.



Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59327		56340_A0	bcm_regex_match_check() was not taking account of chaining overhead while calculating regex DFA size, This issue has been fixed.
SDK-59337		88750_A0	serdes_qrtt_active soc property has 2 bugs: - it is unable to use it without .<unit number> - it has a bug if the quad number isn't divisible by 4 serdes_qrtt_active soc property is fully supported, and can be used without .<unit> and with any valid quad number
SDK-59352		56540_A0 56540_B0	On FB4 and TR3, the field name SUPPRESS_VXLT in FP_POLICY_TABLE was changed to HI_PRI_SUPPRESS_VXLT. And hence the SDK crashed due to accessing the invalid field during entry install. Fixed the issue by validating the field before accessing.
SDK-59358	805970	88650_A0 88660_A0	Fix to Arad ingress reset sequence
SDK-59371		56640_A0 56340_A0 56850_A2	Problem: No qualifier to match on class value assigned based on SourceGport in Egress Stage of FieldProcessor. Solution: A new qualifier (IngressInterfaceClassVPort) is implemented for matching Source GPort based class value in keys defined for bcmstageEgress. This can be used to match a value assigned to a packet using bcm_port_class_set based on ingress Gport / source Gport in Egress Stage of Field Processor.
SDK-59383	806046	56850_A0	In previous releases, soc_mem_generic_insert is running infinitely doing ser correction and soc_schan_op for the table L3_ENTRY_IPV4_UNICAST if there is parity error in this table. For some reason if the correction does not succeed even though soc_ser_sram_correction returns SOC_E_NONE, then soc_mem_generic_insert does SER correction infinitely. A schan retry limit has been introduced. Once this limit has been reached, ser correction and table operation retry will be terminated and the infinite loop will be avoided.
SDK-59405	806511	88030_B0	Added code to support SGMII ports.
SDK-59408 SDK-43524	803517	56840_A0	WC40 port stopped egressing pkts after link down/up event because of errata 2.6 mentioned in BCM56840-ES107. SW work around is provided to reset CL82 state machine when port goes down.
SDK-59414		56440_A0 56440_B0	On BCM56440, the default TSPLL reference has been changed from an external 12.8MHz reference to the internal TSPLL source. Users with an external TSPLL reference should set the configuration SOC property PTP_TS_PLL_FREF to the value of the external reference frequency in Hz. For a 12.8MHz oscillator, PTP_TS_PLL_FREF=12800000 should be used.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59417		56850_A0 56850_A1 56850_A2	In earlier release, when trying to clear the dest hit bit in L2 entry, L2 destination was unexpectedly replaced by <code>bcm_l2_replace()</code> API. This has been resolved.
SDK-59419		88660_A0	IHB 1B & 2B ECC mask enabled for all memories at Arad+.
SDK-59424		88670_A0	Routing over AC: Provided a CINT example for the application - <code>cint_route_over_ac.c</code> . The CINT should be uses only by the Jericho device.
SDK-59425		88650_A0	In TCAM management, TCAM banks may be accessed by different block 'owners': Vlan Translation (VT), Tunnel Termination (TT), Forwarding, OAM, Ingress-PMF-1st-cycle, Ingress-PMF-2nd-cycle, Egress-PMF. A single owner can be configured per bank so that two blocks are not accessing the bank at the same clock. This limitation was not prevented in the current Drivers for specifying owners (e.g. when TCAM bank owner is VTT, then it can be accessed by VT and TT, same for Ingress-PMF Cycle-0 and Cycle-1). This is fixed.
SDK-59455	806478	56450_A0 56450_B0	When L3 ingress mode is set, added support in <code>bcm_mpls_tunnel_switch_get()</code> API to return <code>info->ingress_if</code> for BCM5644x and BCM5645x devices.
SDK-59464	791410	88660_A0	Cosq: Two flags are added as below: 1) <code>BCM_COSQ_FC_PORT_OVER_PFC</code> for Mapping PFC source to port target. 2) <code>BCM_COSQ_FC_INTF_COSQ_PFC</code> for Mapping PFC source to relevant priority in all ports on same interface. For a calling sequence example, refer to <code>cint_arad_pfc_reception_mapping.c</code> .
SDK-59476		88660_A0	The avs read diagnostics shell command was broken for Arad+ (worked properly for Arad). This command is used to retrieve recommended operating voltage as define upon manufacturing.
SDK-59495		56340_A0	Improvement was requested to add FLOW end reason code in App-IQ engine generated Flow Tracker end report. Provided the support for the same.
SDK-59527	806930	56450_A0 56450_B0	Enabled <code>bcm_port_l3_mtu_set/_get</code> APIs for BCM5645x and BCM5644x devices.
SDK-59529	807602	56450_A0 56450_B0	Support has been added for CoE support gport in <code>bcm_port_tgid_set/get</code> API for BCM5645x devices.
SDK-59548		56640_A0 56450_A0 56640_A1 56640_B0 56450_B0	In earlier releases there was no SDK support for inner VLAN tag modification. Now fixed to add the support for inner VLAN tag modification through the usage of <code>bcm_qos_map_XXX()</code> API and <code>BCM_QOS_MAP_L2_INNER_TAG</code> flag to create profiles in <code>EGR_MPLS_EXP_MAPPING_2</code> table.
SDK-59553		88650_A0 88660_A0	OAM: <code>bcm_oam_loss_get()</code> returned incorrect values because <code>loss_farend</code> and <code>loss_nearend</code> were miscalculated.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59568		All	In earlier releases, there was no way to use the API <code>bcm_rate_bandwidth_set</code> to set the rate to 0kbps speed. If the speed was set to zero, the assumption was to disable metering. Now, to disable metering, <code>kbits_sec</code> parameter has to be passed as <code>BCM_RATE_DISABLE</code> and to set the rate to zero, <code>kbits_sec</code> parameter has to be passed as <code>BCM_RATE_BLOCK</code> .
SDK-59570		All	In earlier releases, there is no way to use the API <code>bcm_rate_bandwidth_set</code> to set the rate to 0kbps speed. If the pass the speed as zero, the assumption was to disable metering. Now, to disable metering, <code>kbits_sec</code> parameter has to be passed as <code>BCM_RATE_DISABLE</code> and to set the rate to zero, <code>kbits_sec</code> parameter has to be passed as <code>BCM_RATE_BLOCK</code> .
SDK-59573	806883	88030_B0	handle TMU error interrupt to avoid hash collision errors during hash insert for bcm88030
SDK-59590	798399	88650_B1	Cosq: CFC_EN bit in CFC_ENABLERS register is used to enable device-level flow control. This bit should be set after all other CFC settings are done. Otherwise, some errors may occur. After the fix, the bit will be set after all other CFC settings are finished.
SDK-59598	807290	56640_B0	added the <code>sw_rx_los</code> workaround for the 100G application
SDK-59617		56450_B0	Added support to use port class id upto 127 in <code>bcm_vlan_translate_egress_action_add()</code> API for BCM5645x devices.
SDK-59636		88650_A0 88660_A0 88670_A0	Add a new class, <code>bcmPortClassProgramEditorEgressPacketProcessing</code> , for the <code>bcm_port_class_get/set</code> API. Used to manage a programmable editor variable in the egress.
SDK-59669	808665	56450_A0 56450_B0	In the previous releases the TTL of L3 route label cannot be read back via <code>bcm_l3_egress_get</code> . This has been fixed to read TTL value of L3 route label for requested entry from <code>EGR_MPLS_VC_AND_SWAP_LABEL_TABLE</code> table and return to the user via <code>bcm_l3_egress_get</code> .
SDK-59681		88650_A0 88660_A0 88670_A0	In order to prevent redundant printouts upon <code>bcm.user/pcid</code> initialization, following need to be added to soc file : <code>debug bcm link</code> , <code>attach warn debug soc ddr warn debug soc common err debug sys verinet warn</code>
SDK-59686		88650_A0 88650_B0 88650_B1 88660_A0 88670_A0	In Ingress Field Processor, using qualifiers <code>bcmFieldQualifyIp4/Ip6/Mpls</code> indicates that <code>bcmFieldQualifyInnerSrcMac</code> and <code>bcmFieldQualifyInnerDstMac</code> qualifiers should be taken from third sub-header instead of the second. In practice, when <code>bcmFieldQualifyIp4/Ip6/Mpls</code> are used, then all qualifiers are taken from wrong header, instead of only inner Source/Destination Mac. This is fixed.
SDK-59690	804106	56340_A0	In this release added the phy control bcm shell command support for qsgmii serdes of Helix4

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59700		56850_A0 56850_A1 56850_A2	<p>Description: SER failed to correct entry of <code>ING_FLEX_CTR_COUNTER_TABLE_[0 to 7]</code> once a SER error occurred on it.</p> <p>Root cause: The reported entry index is the offset of the reported entry to the first entry of <code>ING_FLEX_CTR_COUNTER_TABLE_0</code>, but it should be the offset of the reported entry to the first entry of <code>ING_FLEX_CTR_COUNTER_TABLE_X</code> which the entry belongs to.</p> <p>Solution: The software now calculates the index again once we discover the reported entry is one of <code>ING_FLEX_CTR_COUNTER_TABLE_[0 - 7]</code>, the algorithm is using lower 12 bits of the index as the new index.</p>
SDK-59705	805285	All	<p>In L2 MAC table, when calling the API function <code>bcm_l2_addr_delete_by_trunk()</code>, the trunk ID was not masked properly when traversing the MACT, resulting in unintended deletion of entries. The masking of the trunk ID is now fixed.</p>
SDK-59710		88650_B1 88660_A0	<p>AC gport can be found via matching info. An error occurs when trying to get LIF gport via matching info for ingress only ACs. After the fix, VLAN gport can be found with <code>BCM_VLAN_PORT_CREATE_INGRESS_ONLY</code> flag.</p>
SDK-59761	808170	56450_A0 56455_A0 56450_B0	<p>Port command was not working properly in SDK for the Port advertisement field. The port advert mode was not getting set when done via CLI command.</p> <p>Root cause of this issue was that the action mask that enables the port advert field was altered to default this is port ability instead of port mode.</p> <p>Handled this issue by modifying the action mask that enables port advert field to set it by default to port mode.</p>
SDK-59764		88660_A0	<p>BFD: properly set the BFD rate given the parameter <code>bfd_period</code> in <code>bcm_bfd_endpoint_create()</code>.</p>
SDK-59765		88650_A0 88660_A0	<p>BFD: When using a high Detection Time for accelerated endpoints in <code>bcm_bfd_endpoint_create()</code>, the calculation may be wrong. Calculation is done by taking <code>local_min_rx * remote_detect_mult</code>. This value may not exceed 2000000 in MS.</p>
SDK-59823	808861	56850_A0 84328_A0 56850_A1 84328_B0 56850_A2	<p>It was reported that when there is an external PHY that doesn't support TX squelch, the squelching action would return "SOC_E_UNAVAIL", which would lead to the failures of some loopback related operations.</p> <p>The fixing was to use PHY notify event to squelch the TX on internal PHY instead of calling PHY control set to squelch the TX on external PHY when the external PHY is present.</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59833	804363	56850_A0 56850_A1 56850_A2	In previous releases, the conflicting MAC addresses couldn't be retrieved correctly due to a typo in <code>bcm_tr_dual_l2_conflict_get</code> . The issue had been fixed.
SDK-59839		88650_A0 88660_A0	L3: The API <code>bcm_l3_ingress_get()</code> retrieve some RIF related info that can also be set through the related API <code>bcm_l3_ingress_create()</code> . One of the retrieved fields is the VRF of the RIF. The VRF field wasn't updated by <code>bcm_l3_ingress_get()</code> although it was set by <code>bcm_l3_ingress_create()</code> . <code>bcm_l3_ingress_get()</code> was modified to retrieve the VRF field.
SDK-59853		88660_A0	In ARAD+, when using external TCAM for forwarding, RPF and forwarding searches were always performed serially on a single database, using SIP and DIP respectively in search keys. A new SOC property is added to enable the user to perform forwarding and RPF searches on duplicated databases in parallel (as in ARAD): <code>custom_feature_ext_rpf_fwd_parallel</code> - when set, forwarding and RPF searches are performed in parallel. When the SOC property is set, in case of IPv4/6 + RPF forwarding query, external ACL databases IDs are changed to 2 and 3. The actions sizes for ACL databases are also changed accordingly: The action size for ACL database 0 is 64 bits. The action size for ACL database 1 is 16 bits. The action size for ACL database 2 is 32 bits. The action size for ACL database 3 is 24 bits. All of the changes above apply only to ARAD+ devices.
SDK-59854	809432	56340_A0 56547_A0	<code>wcmmod_power_control()</code> was powering off global PLL block when one lane was disabled, which caused all 4 lanes to be powered down. Fixed to power down global PLL block when all 4 lanes are disabled.
SDK-59862	779893	88650_B1 88660_A0	Stacking systems are transmitting between the different systems the whole packet, including the internal headers (FTMH + extensions, PPH + extensions, User-Headers if present). In case the user-headers are present in the system, the second stacking system was adding again user-headers to the packet. This is fixed.
SDK-59864 SDK-61165 SDK-61022		88660_A0	In metering, the meter configuration is optimized for the best rate accuracy. However the configuration was not optimized to have the highest bucket update rate. This created some issues when the traffic is composed of bursts and the bucket size is small. This fix changes the calculation of the values to configure to have the best possible bucket update rate, without having any impact on rate accuracy.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59866		56450_A0 56450_B0	<p>Problem: Access to <code>ING_PHYSICAL_PORT_TABLE</code> in traffic was causing corruption and behaviour was unpredictable.</p> <p>Fix: 1) Added KT2-B1 support</p> <p>2) Fix for <code>ING_PHYSICAL_PORT_TABLE</code> corruption happening in hw Read and write to <code>ING_PHYSICAL_PORT_TABLE</code> access is protected by writing 1 to 0</p> <p><code>INQ_Q_BEGINr:ING_PHYSICAL_PORT_S</code> <code>BUS_WITH_PKT_DISABLEf</code></p>
SDK-59898	701984	88650_A0	<p>Disabling a port, using <code>bcm_port_enable_set()</code> fails since EGQ fail to drain. Fixed</p>
SDK-59917		56640_A0 56640_A1 56640_B0	<p><code>bcm_cosq_gport_scheduler_set()</code> API does not seem to take effect while setting the scheduler to WRR. Turns out, even though the API does set the register for enabling WRR initially, it gets overwritten later on. This has been rectified by changing code such that the register change is done towards the end of the API implementation.</p>
SDK-59934	811343	56850_A0	<p>Root cause: Once a ser error occurred on a entry of memory of <code>L3_DEFIPm</code>, the correction handle will invoke routine <code>soc_generic_ser_process_error</code> to correct it. This routine is a common one for all chips. But each chip inputs a chip based specific event not a BCM event. The chip based specific event varies on each chip. But in the routine, it only deals a SER error, so we need raise a PARITY error detected event. The existing code really has the code to raise a event but with a wrong type since it used the <code>err_type</code> directly. As a result, the event reported in this case would be different.</p> <p>After invoking the SER correction routine, a <code>SOC_SWITCH_EVENT_DATA_ERROR_LOG</code> should be reported if there is a valid log ID available. But the existing code did nothing.</p> <p>Solution: Correct the wrong reported type to <code>SOC_SWITCH_EVENT_DATA_ERROR_PARI</code> TY and adding code to report <code>SOC_SWITCH_EVENT_DATA_ERROR_LOG</code> if a valid log ID is available after invoking the SER correction routine can resolve the issue.</p>
SDK-59955	808817	88650_A0 88650_B0 88650_B1 88660_A0	<p>L2 MACT Flush: Added support for 2 fields flush in application-based-allocation mode using <code>bcm_l2_replace_match</code> api with valid <code>match_addr.dest</code> & <code>match_addr.encap_id</code> fields.</p>
SDK-59966		56640_B0 56850_A2	<p>Fixed an indexing issue in LPM algorithm that allocates space for the routes in <code>L3_DEFIP</code> TCAM. The issue was specific to the case of the TCAM being in paired mode and LPM scaling is enabled (config property <code>lpm_scaling_enable=1</code>).</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-59974	809811	56440_A0 56440_A1 56450_A0 56440_B0 56455_A0 56450_B0	In the previous release the priority given in <code>bcm_vlan_dtag_add</code> was not taken as the outer tag priority. ING_VLAN_TAG_ACTION_PROFILE table needed to program by setting <code>SIT_OPRI_ACTION</code> field to set the priority field.
SDK-60017	806360	88650_A0 88640_A0 88650_B0 88650_B1	OAM: Reduce running time of <code>bcm_oam_endpoint_create()</code> and <code>bcm_oam_endpoint_action_set()</code> by 50%-60%
SDK-60025	786758	88650_A0 88650ACP_A0 88650_B0 88650_B1 88660_A0	In Counter Processor, under Egress TM mode, the Counter Engine can differentiate between different Traffic Class, even if the port is set under one priority level. This was blocked in the Driver, and fixed now.
SDK-60029		88650_A0	Cell drops occurred on a system with single fabric pipe BCM88650 and dual fabric pipe BCM88750. Such a single fabric pipe BCM88650 should be configured as follow: 1. Set to single fabric pipe mode <code>is_dual_mode.BCM88650=0</code> 2. Adjust fabric cell format to dual pipe mode system <code>system_is_dual_mode_in_system.BCM88650=1</code>
SDK-60041	812854	56150_A0	Fix the packet loss issue during EEE mode on GE port with UNIMAC.
SDK-60042		88660_A0	Added support in BFD Remote Server. To add BFD endpoint to the OAMP server the flag <code>BCM_BFD_ENDPOINT_HW_ACCELERATION_SET</code> should be used in <code>bcm_bfd_endpoint_create</code> api. For usage example see <code>cint_bfd.c</code>
SDK-60048	803308	56624_B0 56624_A0 56624_A1 56514_A0 56334_A0	memscan support is added for TCAM error detection/correction on 56624, 56334 and 56514 chips.
SDK-60049		88650_A0 88650_B0 88650_B1 88660_A0	soc property length limitation cause segmentation fault at trill init. Change soc property max length to 128 characters and add test to prevent segmentation fault.
SDK-60056	812600	88650_B1 88660_A0 88670_A0	In Field Processor, the support of cascaded ingress-egress lookup was possible only for packets without Learn-Extension header. If <code>custom_feature_cascaded</code> egress learning SOC property is enabled, a Learn-Extension header is added to all the Ethernet packets, and cascaded ingress-egress lookup is supported in this case.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60070	811441	56450_A0 56450_B0	<p>Problem: Earlier interface mode retrieval method was based on PortType in legacy Xenia driver code (i.e. XE_TYPE ==> XGMII, GE ==> GMII etc). For WC core driver, interface mode was based on port speed. This was creating confusion</p> <p>Solution: Chaged interface mode retrieval method in Xenia driver. Now it will be based on port speed (i.e. 10G ==> XGMII, 1000/2500G == GMII , 100M == MII)</p> <p>This will make behaviour same for both xenia/unicore and warp-core driver and no confusion should appear now.</p>
SDK-60072	812645	88650_B1	<p>An HW bug is present in EGQ block when terminating more than 64 Bytes (see errata document for more information). A work-around was proposed in <code>srcexamplesdppoint_field_ingress_large_termination.c</code>, where 14 Bytes were terminated at ingress if the Forwarding-Header was located more than 32 bytes away from start of packet. In certain conditions (e.g., in stacking systems with multiple VLAN tags), removing 14 Bytes was not sufficient. It has been fixed to 28 Bytes.</p>
SDK-60073	794596	All 56850_A0 56850_A1 56850_A2	<p>Add two flags for extender usage: BCM_EXTENDER_PORT_DROP and BCM_EXTENDER_PORT_ID_ASSIGN_DISABLE.</p> <p>When BCM_EXTENDER_PORT_DROP is set, packets egress to the indicated extender port are dropped.</p> <p>When BCM_EXTENDER_PORT_ID_ASSIGN_DISABLE is set, matched packets are not forwarded to the extender port.</p>
SDK-60085	810017	56640_A0	<p>On Triumph_3, when parity error was induced by disabling CMIC_SER0_PROTECT_ADDR_RANGE_VALID and subsequently detected in SBUS DMA by Memscan, no action was being taken to recover from this error. This was because when memscan encounters DMA failure and falls back to individual memory read, data was being read from the cache and not from h/w and hence the parity error was not being caught.</p> <p>Corrected this to read from h/w which issues an Schan operation which then recovers from parity error.</p>
SDK-60094	810331	88660_A0	<p>Routing Over VXLAN in ARAD+: When <code>bcm_l2_egress_create</code> is called with different SMAC, the encapsulation of outer SMAC is always as the first allocated SMAC.</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60099		56850_A0 56850_A1 56850_A2	Issue: bcm_field_qualify_IpType is setting right h/w encoding for bcmFieldIpTypeNonIp but bcm_field_qualify_IpType_get is failing due to wrong h/w encoding value derived in internal get function. Fix: modified internal get/set functions to match h/w encoding for bcmFieldIpTypeNonIp.
SDK-60102		56450_A0 56440_A1 56450_B0 56440_A0 56455_A0 56440_B0	In the previous release the priority given in bcm_vlan_dtag_add was not taken as the outer tag priority. ING_VLAN_TAG_ACTION_PROFILE table needed to program by setting SIT_OPRI_ACTION field to set the priority field.
SDK-60109		56340_A0	_bcm_tr3_flex_ctr_pool_unmap was not releasing all reserved flex counters by _bcm_tr3_flex_ctr_pool_map. Now fixed _bcm_tr3_flex_ctr_pool_unmap to release all reserved flex counters for Appiq.
SDK-60110		88650_A0 88650_B0 88650_B1	For ISSU, in warmboot-engine module, an option is given internally to skip a variable when retrieving the external storage in a newer version. This feature had issues discovered in its first usage in 6.3.8: the Driver is still trying to access these variables in some part of the code, which may lead to errors in the variable offset computation. This is fixed.
SDK-60131		56334_B0 56334_A0	Problem : When you create a child and parent policer in a slice, it takes up 2 complete meter pools and reserves them completely for all parent and child policers created in that slice. And those meter pools will not be shared by any policers from other slices. Because of this, we are prematurely exhausting policer resources. Requirement here is to use same METER POOL for policers created in different slices. Since Level0 policers of a slice(i.e. group) cannot be shared by other slices, we need to expand level-1 meter creation logic to use same meter pool for different slices. Solution: Adding new variable to bcm_policer_config_t structure to save the meter pool id. User can retrieve the meter pool_id and create a policer with give meter pool id so that the meters are created in the same pool. But the field entries related to meters in same pool need to be mutually exclusive. Added required doc changes for above implementation.
SDK-60144	808055	88650_A0 88660_A0	added a lock in device reset so other threads won't access the device
SDK-60166	808650	56850_A0 56340_A0 56850_A1 56850_A2	In previous releases it was reported that bcm_vlan_translate_egress_action_add etc. APIs would return BCM_E_PORT error when passed a VLAN port type gport as an input parameter. This issue has been fixed by adding the VLAN type gport as a condition during gport type check so that not to return BCM_E_PORT directly.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60168	813722	All 56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In earlier releases, vlan port vp could not be destroyed when associated gport was a trunk and this trunk did not have members. This has been resolved.
SDK-60169		56850_A0 56850_A1 56850_A2	In the previous release, when two SVPs created on the same physical port were configured with two mirroring sessions pointing to different MTPs, the MTP of the first mirroring session would be overwritten by the second MTP, which was incorrect. In this release, this issue has been addressed by adding a new bookkeeping to ensure two MTP slots can be allocated in the case.
SDK-60170	813670	All	When the external phy is in repeater mode, then HW linkscan for those ports should scan the internal serdes link status instead of external phy's. In order to acheive this, the repeater flag was set for BCM84756. Linkscan will only scan the internal serdes link status if the flag is true.
SDK-60179	810107	56540_A0	64bit index space was not initialized in case of rpf mode and 128b entries disabled mode. Due to this applications were not able to add routes in this particular configuration. The fix is to initialized the 64 bit index space to right values.
SDK-60194	813706	56850_A0 56850_A1 56850_A2	In this release added support for BCM_PORT_PHY_CONTROL_RX_SIGNAL_DETECT APIs (get and set) to TSCMOD
SDK-60199	812394	56450_A0 56455_A0	In prior releases both OPRI_CFI_SELF and IPRI_CFI_SELF fields in EGR_VLAN_XLATE table were set to "1" if the priority is set to "-1" while adding entry to EGR_VLAN_XLATE table though the action configured includes only for outer tag. This impacted LSP EXP bit remapping not to work. Now fixed to set IPRI_CFI_SELF action includes configuration for inner tag and OPRI_CFI_SELF if the action includes configuration for outer tag.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60222 SDK-60187		88650_A0 88650_B0 88660_A0	added memory array caching: Some memories have multiple instances within a single block in the device. Internally, the SDK organizes such tables into arrays, so that each instance of a table can be accessed using an instance-id. The caching mechanism did not support caching of memory-arrays. Customer impact: Normally, memories can be cached to DRAM as part of soft-error (SER) protection mechanism, or in order to improve driver runtime performance. For memory arrays, caching was not available. For parity-protected configuration memories, caching is used to recover from a soft-error. As a result of this addition, memory arrays can now also be protected by this mechanism. list of relevant memories: EPNI_EEDB_BANK EPNI_DATA_FORMAT EPNI_LFEM_FIELD_SELECT_MAP IDR_MCDA_PRFSSEL IDR_MCDB_PRFSSEL IHB_FEM_BIT_SELECT IHB_FEM_MAP_INDEX_TABLE IHB_FEC_ENTRY IHB_TCAM_ACTION IHP_PARSER_PROGRAM IHP_VLAN_PORT_MEMBERSHIP_TABLE IHP_LIF_TABLE note for 6.3 branch: the list of cached tables is not filled by the fix in this JIRA, and will be filled soon.
SDK-60225		88650_A0	Reference only: Parity memories are added to the reference shadow list in order to be cached by default.
SDK-60233	815117	56640_A0 56640_A1 56640_B0	In earlier releases, the parity information structures for MMU_MTRO memories were incorrect. Hence the recovery logic fails to correct the error, and hence the error was being reported continuously. This is now fixed by correcting the parity structures.
SDK-60241 SDK-43891	814752	56640_B0	Previously, when we have an hash bucket full, and If we try to add an double wide L2 entry, it fails even if we pass an replace flag. Now support has been added to dynamically replace narrow and wide entries with each other, by passing an replace flag. In such cases, and existing entry will be deleted (two entries may be deleted if the newer entry is a wide entry and existing entries are narrow).
SDK-60243	814299	All	Diag Shell command was not able to qualify DstMultipath as parameter. Added case to handle DstMultipath when entered from diag shell.
SDK-60261	815126	88650_A0 88650_B0 88650_B1 88660_A0	fixed bug: EGQ_DSP_PTR_MAP wasn't correctly updated when removing LAG ports that had the BCM_TRUNK_MEMBER_INGRESS_DISABLE flag.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60262	815031	88650_B1 88660_A0	In TCAM memory, there is no mechanism for identifying and fixing SER errors in the TCAM. A new mechanism is added which handles an error upon SER occurrence, identifies the location of the error and according to TCAM bank owner, reproduces the TCAM entry from software database and rewrites it to TCAM memory.
SDK-60267	815566	88650_A0 88660_A0	In the Counter Processor module, during the background counter read from DMA (aka algorithmic read), the number of counters asked to be read could exceed the maximum size of the counter cache. This is fixed by adding a validation to the number of counters. This prevents the PCIE bus to crash.
SDK-60278	813007	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In previous releases, there was an issue for DLB HG trunks that traffic ingress from port located in Y-pipe would be getting dropped. The issue was fixed by configuring DLB_HGT_FLOWSET_Y table during DLB setting.
SDK-60280	815365	56450_A0 56450_B0	In bcm_port_tpid_set API port resolution logic for vp ports created on coe subport was not implemented. Now implemented port resolution logic for vp ports created over coe subports The fix has been updated for bcm_port_tpid_get and bcm_port_tpid_add APIs as well.
SDK-60301	815641	56340_A0	In previous releases when an warpcore block had both with both ieee and hlgig encapsulations, changing the encapsulation of xe port to hlgig caused all the other ports to change to 11G, which resulted in linkdown for all the ports. In such cases we needed to limit the maximum speed of xe ports to 10G which can be done by adding the config variable port_max_spd for that particular port . This issue has been resolved.
SDK-60302		56845_A2	PQ_XYZ registers were giving SCHAN read access time out errors with "dump soc" command. These registers aren't open anymore on TD/TD+. So they are all masked now in SDK .
SDK-60304		88650_A0	In mesh system, there was no gport type that addresses fap id 0. New gport type added: bcmCosqGportTypeGlobalFabricMeshDev0
SDK-60307 SDK-60336	814621	56456_B0 56450_A0 56455_A0 56456_A0 56450_B0	For BCM5645x devices, when config 10 (bcm5645x_config=10) was used, the existing TDM used was not fully flexible. So it caused issues while converting 10G port to 4x2.5GE port. The earlier TDM used for this config has been replaced with a fully flexible TDM to support conversion of 10G port to 4x2.5GE ports without any issue for BCM5645x devices using config 10.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60312	812350	88650_B1 88660_A0 88670_A0	OAM: For remote endpoints, a call to <code>bcm_oam_endpoint_get()</code> returns the flag <code>BCM_OAM_ENDPOINT_FAULT_CCM_TIMEOUT</code> in the field <code>faults</code> in cases there is a CCM timeout. Likewise for local endpoints, each of the flags <code>"BCM_OAM_ENDPOINT2_RDI_FROM_RX_DISABLE"</code> and <code>"BCM_OAM_ENDPOINT2_RDI_FROM_LOCAL_DISABLE"</code> will be on as outputs of <code>bcm_oam_endpoint_get()</code> if and only if these flags were on when calling <code>bcm_oam_endpoint_create()</code> for the respective endpoint.
SDK-60315	814659	56640_A0 56640_A1 56640_B0	Problem: <code>DstPort</code> qualifier used for qualifying cpu packets was getting qualified for transit multicast packets as well. Solution: Extended <code>DstGport</code> qualfier for all devices, where there is control to set the the type of packets that we want to qualify using <code>d_field/d_type</code> fields in <code>FP_TCAM</code> .
SDK-60321	806640	56456_B0 56450_A0 56455_A0 56456_A0 56450_B0	Added switch control <code>bcmSwitchTimesyncEgressTimestampingMode</code> to choose between 32bit or 48bit timestamping mode for 1588 packets. Also new flag <code>BCM_PORT_TIMESYNC_TIMESTAMP_CFUPDATE_ALL</code> is added to specify that all 1588 packets will be egress timestamped to to the <code>flags</code> field of <code>bcm_port_timesync_config_t</code> struct. Default mode is to timestamp only ingress timestamp packets.
SDK-60336 SDK-60307	813964	56450_A0 56450_B0	For BCM5645x devices, when config 10 (<code>bcm5645x_config=10</code>) was used, the existing TDM used was not fully flexible. So it caused issues while converting 10G port to 4x2.5GE port. The earlier TDM used for this config has been replaced with a fully flexible TDM to support conversion of 10G port to 4x2.5GE port without any issue for BCM5645x devices using config 10.
SDK-60337	816022	56456_B0 56450_A0 56456_A0 56450_B0	1588 protocol's host cpu based applications can use either one-step time-stamping or two-step timestamping mode. In case of,one step timestamping mode, PTP packets correction field is updated with packets transmits timestamp and in two step timestamping mode, PTP packet's egress time-stamp can be retrieved from MAC. This JIRA provides support for one-step and two-step timestamping for host CPU based PTP application through <code>bcm_tx()</code> API for Katana 2 chipset.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60353	807644	56846_A0 56840_A0 56850_A0	In previous release, bcm_port_dscp_map_mode_set/get(), bcm_vlan_priority_map_set/get(), bcm_port_dscp_map_mode_set/get() could only work for physical port. Now these APIs have been enhanced to support MPLS PORT on TD, TD+ & TD2. Please note that the GPORT DSCP & DOT1P profiles should be initialized first before being used in these APIs.
SDK-60356		88650_A0 88660_A0	QOS IPV4/MPLS: Added default 1:1 mapping from TC to TOS/EXP in the egress instead of just 0
SDK-60358		88670_A0	OAM: Implementation OAMP-DM trigger mechanism. Refer to user manual for full documentation.
SDK-60368	814626	56850_A0 56850_A1 56850_A2	When BCM_SWITCH_PKT_INFO_HASH_UDP_SOURCE_PORT flag is used, it is supposed to return the udp source port to be used in vxlan packet. So it should fit in 16 bits. Root cause: The underlying routine, compute_td2_rtag7_vxlan() computes the "raw" 20-bit value, which is never masked down to 16 bits for the UDP port option. Solution: bcm_switch_pkt_info_hash_get() is fixed to mask raw 20-bit entropy label value down to 16 bits and return the (64k) value properly.
SDK-60384		88660_A0	Decreasing E2E shaper of a non-active port causes packet drop of another active port within the same channelized interface - Fixed
SDK-60393		88650_A0 88660_A0	In Ingress Field Processor, when creating a Direct Table database, the first Direct table entry may be written to the wrong action table. Per TCAM bank, two action tables can be allocated to the Direct table Field group. This issue happens in case the second table (20b MSBs) is allocated, and no TCAM bank is yet allocated. This is fixed.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60394		88650_A0	<p>Credit scheduler memories are added to the reference-application cache list. Memories can be cached to DRAM as part of soft-error (SER) protection mechanism, or in order to improve driver runtime performance. Upon initialization, caching mechanism usually uses Table DMA to synchronize the HW with the shadowed memory cache. Since the standard table DMA mechanism is not applicable to scheduler tables, initialization for these tables uses non-DMA access. An alternative mechanism is currently being analyzed, and might be available in the future, to improve performance when caching these memories. For parity-protected configuration memories, caching is used to recover from a soft-error. Due to this fix, the following credit scheduler memories added to the cache list and can be protected by this mechanism: a. list of relevant memories:</p> <p>SCH_CIR_SHAPERS_STATIC_TABLE_CS ST SCH_CL_SCHEDULERS_CONFIGURATION _SCC SCH_CL_SCHEDULERS_TYPE_SCT SCH_DEVICE_RATE_MEMORY_DRM SCH_DUAL_SHAPER_MEMORY_DSM SCH_FLOW_DESCRIPTOR_MEMORY_STATI C_FDMS SCH_FLOW_GROUP_MEMORY_FGM SCH_FLOW_SUB_FLOW_FSF SCH_FLOW_TO_FIP_MAPPING_FFM SCH_FLOW_TO_QUEUE_MAPPING_FQM SCH_HR_SCHEDULER_CONFIGURATION_ SHC SCH_PIR_SHAPERS_STATIC_TABLE_PS ST SCH_PORT_SCHEDULER_WEIGHTS_PSW SCH_SCHEDULER_CREDIT_GENERATION_ CALENDAR_CAL SCH_SCHEDULER_ENABLE_MEMORY_SEM</p>
SDK-60399		56340_A0	<p>Resetting of logical engine was not detaching logical to physical engine mapping, which caused wrong free size calculation. Fixed it by marking logical to physical engine mapping as NULL after reset.</p>
SDK-60408		56450_A0 56450_B0	<p>Problem: Two consecutive IDLE slots are required in TDM to meet TCAM atomicity. One typo remained in TDM-A2 where instead of IDLE slot value as 63, value remained as 62. It could cause (theoretically) concern for TCAM atomicity in RTL design Also value of AUX_ARB_CONTROL_2r:TCAM_ATOMIC_W RITE_ENABLEf was 0 i.s.o. 1 Fix : Added one more check in tdm verification function for avoiding possibility of IDLE-SLOT=63 in middle of TDM. Also restricted Two consecutive IDLE slots to 63 i.s.o. 62 and 63 Made AUX_ARB_CONTROL_2r:TCAM_ATOMIC_W RITE_ENABLEf value as 1 in KT2:misc() routine</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60434	816207	56450_A0	Deletion of port in range of 120-169 was failing for the multicast group. Fixed the code for the port in range of 120-169 to delete the IPMC_VLAN replication pointer specific to the port for the multicast group, hence the deletion of port from replication list successful.
SDK-60436	816417	88030_A0 88030_B0	When allocate trie node, we need to confirm the valid field of bpm is between [0, skip_len+1] and mask out the higher bits which is invalid.
SDK-60439	814627	56456_B0 56450_A0 56456_A0	Broadcom switch hardware can recover SyncE clocks GE/XE ports. This JIRA adds support to recover SyncE clocks from GE/XE ports for Katana 2 (56450 A0 and 56450 B0) chips.
SDK-60440	817398	88030_A0 88030_B0	Changed the MACROs to ensure they are CALADAN3 specific checks and do not affect any other modules.
SDK-60446	815613	56850_A0 56850_A1 56850_A2	In earlier releases, displaying port bitmaps in the Diag Shell caused a segmentation fault. This has been resolved.
SDK-60464	815760	88650_B1	This memory is not accessible memory for CPU. The fix changes the print and provides correct information.
SDK-60470	817158	56850_A0	In earlier releases, only (<=4k) L3 interfaces per port could be deleted from a multicast group by calling bcm_multicast_egress_delete. This has been fixed in this release.
SDK-60477	813482	56640_B0	added the fix to allow tx transmit during linkdown
SDK-60482	817859	56850_A2	In earlier releases, SDK did not support changing vlan port match condition from vlan port vp to vplag vp when vlan port was added to vplag. This has been resolved.
SDK-60485		88670_A0	Failover: Added a CINT example for 1+1 protection creation in the new Jericho decoupled mode. The sequence can be seen in cint_vswitch_cross_connect_p2p.c upon creation of access side PWE and VLAN ports.
SDK-60486	817588	88650_B1 88660_A0	Reference code: BCM sdk enables some interrupts at the reference code for monitoring and corrective action performance. This fix adds some interrupt to this list in order to enable the monitoring and corrective action: ARAD_INT_EGQ_DBFECC_1BERRINT, // monitoring ARAD_INT_EGQ_DBFECC_2BERRINT, // monitoring ARAD_INT_EGQ_PARITYERRINT, // This interrupt indicates on parity error fault at EGQ parity protected memories. Enabling this interrupt makes parity error corrective action for these memories possible.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60489		88650_A0 88650_B0 88660_A0	Trill MC: In 6.4.X we moved to a new Trill sequence for multicast case. Bug was found when adding MACT L2 entries with multicast group and trill port information. When multicast group is not just a number but also has indication of multicast type (for example when we take <code>multicast_id</code> as a result of <code>bcm_multicast_create</code>) then error is invoked.
SDK-60491	818025	All	Egress True Mirroring destinations were not displayed in diag shell mirror show. Corrected this behavior and added flags to display true egress mirror destinations.
SDK-60494		All	Fixed BCM shell commands prefixed with unit ids indications such as '*' and '2-5:' for a system in which the total number of devices <= device ID [i.e. when they are 'holes' in unit IDs, for example - 2 units, IDs '0' & '2'] (these fixes were missed in SDK-58919).
SDK-60503	815053	56450_A0	The frame length check was getting enabled always for non Higig ports on BCM5645x devices when doing a flexIO hot-swap operation. This has been fixed by disabling frame length check by default after a flexIO hot-swap operation. The frame length check can be enabled after flexIO hot-swap operation by setting config variable <code>mac_length_check_enable</code> .
SDK-60508	816530	56340_A0 56340M_A0	During multi hash move operation, the destination bucket can have different sized entries than the incoming entry. In earlier releases, the SDK did not care if the incoming entry could fit into destination bucket. Due to this, it could corrupt the existing entry in some cases. The fix is to add an infrastructure to check if incoming entry can be added into the destination bucket. if not, keep looking for next destination bucket until you get a free space or you have reached to max try counter.
SDK-60515	793455	All	Fixed OTN (bypass TDM) in 88650/60 connected by Mesh.
SDK-60527		56846_A0 56850_A2	Problem: After warmboot upgrade, udf field entry was not getting recovered properly. Solution: Added a missing condition while recovering qualifier set for field entries having UDF.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60536		56455_A0	<p>In BCM5645x devices, Parity errors were observed during SDK initialization.</p> <p>This was because of parity errors in Channel interfaces were not handled and were also checked for interfaces that were not used.</p> <p>Fixed this issue by merging changes made SDK-35808 from master to 6.3.9 branch via SDK-61032 to modify CI error mask to handle parity errors and also to enable parity checks for only those interfaces supported in device.</p> <p>Also when 156.25Mhz reference clock for LCPLL is enabled in the configuration, system crashes</p> <p>This is because 156.25MHz refclk in LCPLL non-bypass mode is not supported in Katana2</p> <p>Fixed this issue by removing 156.25Mhz refclk initialization during soc reset.</p>
SDK-60543	811382	88660_A0	<p>Upon accessing link APIs when linkscan is active, the driver might deadlock from time to time on different links. This condition is fixed.</p>
SDK-60549		56150_A0 56151_A0	<p>BroadSync embedded application support added for BCM5615X</p>
SDK-60551		56640_A1	<p>In older releases, for 5664x devices, for Ports to come up as XE ports, the corresponding bits have to be set in the config property <code>pbmp_xport_xe</code>. Now, In addition to XE ports, all the 100G+ (CE) ports have to be added to this config property.</p>
SDK-60560	818014	88650_A0 88650_B0 88650_B1 88660_A0	<p>L2 Flush: Added support for <code>encap_id</code> masking in <code>bcm_l2_replace_match</code> api.</p>
SDK-60576	818773	56450_A0 56450_B0	<p>Problem Statement: Local port resolve is failing when port number is greater than 127 when trying to set the scheduling mode Resolution: Changed the code to use proper function to resolve local port resolve when the GPORT is of type <code>BCM_COSQ_GPORT_SUBSCRIBER</code></p>
SDK-60583	816905	56340_A0 56547_A0	<p>Parity errors in <code>VLAN_XLATE</code> table detected on packet lookups are reported to <code>ser_fifo</code> as table <code>RAW_ENTRY_TABLE</code> with physical addresses. In previous releases, the index of parity error entry was not obtained correctly in software. In this release this bug has been fixed to enable memory decoding routine to work for <code>RAW_ENTRY_TABLE</code> in <code>Helix_4</code>.</p>
SDK-60590		88650_A0 88650_B1 88660_A0	<p>This JIRA fixes segmentation fault which occurred occasionally at <code>deinit/init</code> sequences. This happened because of improper implementation of interrupt <code>deinit</code> sequence. At this case interrupt asserted during SW interrupt resources deallocation. This fix makes sure that no interrupt will asserted during interrupt application <code>deinit</code>.</p>
SDK-60593	819011	84793_A0	<p>Added per lane eye scan support. Modified <code>enzo</code> (<code>phy84793</code>) driver to return RX sequence done for the specified lane.</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60595		88650_A0 88660_A0	<p>OAM: For calls to <code>bcm_oam_endpoint_create()</code> with the <code>opcode_flags</code> <code>CCM_OAM_OPCODE_CCM_IN_HW</code> set (accelerated endpoints) calling sequence has been changed: For Up-MEPs the field <code>tx_gport</code> must be set to <code>BCM_GPORT_INVALID</code>. The gport on which OAM PDUs will be transmitted will be determined through the pipeline, as dictated by the protocol for up-MEPs. For Down MEPs the <code>tx_gport</code> field must be set to a valid system port. For Ethernet OAM, two endpoints using the same least significant byte in the <code>src_mac_address</code> field must use the same <code>tx_gport</code>.</p> <p>BFD: Fixed a bug in <code>bcm_bfd_endpoint_create()</code> for accelerated endpoints (calls in which the flags <code>BCM_OAM_OPCODE_CCM_IN_HW</code> is set) causing BFD frames to be sent to an incorrect gport when multiple endpoints exist on different gports.</p>
SDK-60630	798896	88660_A0	Enabling this feature with SOC property "custom_feature_lpm_custom_lookup=1" (will disable Fiber Channel application).
SDK-60657	818481	88660_A0 88670_A0	OAM: Support OAMP server. For complete documentation consult the user manual, or for an example see <code>oamp_server_example()</code> in <code>cint_oam_acceleration_advanced_features.c</code> . Currently only CCM support is available. Note that down MEP CCM transmission may also be defined on LAG gports.
SDK-60670		56850_A0 56850_A1 56850_A2	Support has been removed for API to retrieve member port for DLB of HG Trunk.
SDK-60672	816228	88650_A0	Info log should not flood the console with log message. Now change the log severity from verbose to debug and info to verbose can fix this issue.
SDK-60692	819679	56640_A0 56540_A0	<p>Problem : This is due to the issue that when we detach a counter from field entry, field module will reduce the <code>field_specific</code> reference count but it was not updating the stat module specific reference count . Due to this stat destroy was throwing error saying reference count is still not zero.</p> <p>Solution : Field entry detach routine of counter was updated to call relevant function to reduce stat related reference count.</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60710	794588	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In the previous release, the API BCM_PORT_EXTENDER_TYPE_NONE was not implemented. In this release, this issue has been addressed by implementing the API.
SDK-60711	817525	All	Handle LinkScan in case of port removal and insertion. Not relevant for most customers
SDK-60719 SDK-52048	818562	56640_A0	Fixed following issues related to OAM warmboot recovery: - recovery tries to access an invalid field - memory pointer getting freed was not set to NULL. - CCM period was not being recovered properly - ma index re-allocation was being attempted while trying to recover endpoint on same vlan port but different level, which resulted in failure. - scache allocation was not correct during upgrade from older version.
SDK-60729	818782	88660_A0 88670_A0	The bcmFieldQualifyVlanFormat is exposed at Egress Field Processor stage for both key and preselection. Its encoding is different, according to the internal Ethernet-Tag-Format (5 bits) signal.
SDK-60750		88660_A0	In L2 MAC table, aging period was not configurable per VSI. The number of meta-cycles of an entry not being observed before it is deleted were hard-coded. The number of aging meta-cycles before entry aging can now be configured per VSI, by using the API function bcm_vlan_control_vlan_set() and setting the field aging_cycles in the input structure bcm_vlan_control_vlan_t. Valid range for aging_cycles: 0-6, 0 is default. The user may configure up to 4 different aging cycle values for different VSI's.
SDK-60756	816538	56340_A0	Loopback ports were not being to valid ports bitmap for helix_4 in previous releases, and hence any operations on loopback ports were failing. This is now fixed by adding the loopback ports to valid ports bitmap.
SDK-60769	816767	56850_A0 56850_A1 56850_A2	In the previous release, the TPID did not enable VLAN GPORT. In this release, this issue has been addressed by correcting the related APIs.
SDK-60784	818772	56334_A0	In the previous SDK, the oam cli command could not accept MiM gport for port parameter when creating endpoints. This has been changed to allow to create endpoints on MiM gport.
SDK-60797	820536	56850_A0 56850_A1 56850_A2	Fixed bug in interrupt handler for CMICm-based devices which would inadvertently enable hardware interrupts in polled-IRQ mode thus causing a system hang.
SDK-60843		56450_A0 56340_A0	Enabled internal termination on the gigabit serdes interface for HX4 & KT2 when system is operating in eHost mode.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60846	809695	56440_A0 56440_B0	For BCM 5644x devices there was an issue with parity errors coming from SVM_POLICY_TABLE and SVM_MACROFLOW_INDEX_TABLE entry. This has been fixed.
SDK-60853	819575	56540_A0 56540_B0	True Egress frames of size greater than 8K were not getting egressed. Reason for this was that MTU Size programmed in PASSTHRU_NLF_MTU_CHECK register was set to 8K leading to drop seen. This default MTU Size on loopback port has been increased to maximum value which is around 16K by default . This is done to be in sync with other MTU register which currently hold default value of 16K.
SDK-60860		56844_A0	Add support for SKU 56742
SDK-60890		56845_B0 56845_A2 56840_A0 56850_A0 56850_A1 56850_A2	In previous release, bcm_qos_port_map_set() did not set TRUST_DSCP_V4/V6 on physical ports but set it on virtual port. The different behavior in the same API made customer confused, now this issue has been fixed.
SDK-60900	820836	88650_B1 88660_A0	In Forwarding stage (FLP), the Mac-in-Mac and TRILL-Multicast FLP programs were allocated statically at init with the same program ID. This prevented to allow both applications to run simultaneously. The Mac-in-Mac FLP program is now allocated dynamically.
SDK-60905	821319	56450_A0 56450_B0	In the previous release the validity check for the nodes in LLS tree was missing, this caused a segmentation error if any invalid node was getting accessed while flushing it. The check has been added in this release to avoid any segmentation error in attempt of accessing invalid nodes in the LLS tree.
SDK-60909		88650_A0	Minor coverity issues resolution involving the removal of unreachable code, and uninitialized variables.
SDK-60927	819752	88030_B0	BCM_CALADAN3_SUPPORT was being used without SOC_IS_CALADAN3(unit) check thus affect ARAD or other images being built with the above #ifdefine.
SDK-60955		All	Configuration variables which are parsed before soc_attach() is called, no longer require that the unit number is appended. For example, polled_irq_mode=1 now works as expected, whereas before this change it had to be specified as polled_irq_mode.<unit>=1.
SDK-60956		88650_A0	Diag command "diag cosq" without any parameter crashes. The command fixed to return usage description.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-60958		88650_B1 88660_A0	Corrective action for IPS QDESC parity error: A parity error in the IPS QDESC dynamic memory, corresponding to an unused VoQ, while credit watchdog range includes this unused VoQ, would cause a parity interrupt that is never fixed. Eventually the interrupt handler would detect interrupt storm and mask IPS parity interrupts. This fix recovers for a SER on unused VoQ by writing (zero-value) to the corrupted entry. Please note that for entries corresponding to active VoQ, SER condition is resolved once the ASIC updates the appropriate entry (no change from current behavior for active VoQs)
SDK-60964	820443	56640_A0 56340_A0	In previous releases, when parity errors are detected in VLAN_XLATE_1 or VLAN_XLATE_2 tables in Triumph_3 or Helix_4, the table name was not resolved in SER correction routine. This would cause VLAN_XLATE_1 or VLAN_XLATE_2 tables not to be corrected correctly. In this release, this bug has been fixed. Table name will be returned correctly and table entry will be corrected once parity error is detected.
SDK-60971		56845_B0 56845_A2	In earlier release, the base_ptr field in ECMP group table was not configured correctly on TD+ during defragment. This has been resolved.
SDK-60988	821704	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In earlier releases, deleting a secondary L3 Interface unbinded the VRF from the primary one. This has been resolved.
SDK-61031	823863	All	Host endian-ness was not correctly handled in SDK side as-well-as in Ukernel side. Due to that where byte swapping was not required (on Little Endian eHost) but was happening causing all reverse data, string messages. Ukernel Side fix: Corrected endianness issue and added some print messages for debugging purpose. Now Bytes will be swapped only when host side(ehost) endian-ness and own(ukernel) endian-ness differs. SDK Side fix: 1)Removed version string display in init phase as endian-ness of host side is not known in init-phase i.e. after system-info reply only, ukernel recognize and sends correct version-info string 2)Removed work-around of reverse version-string on LE host 3)Now sends 0 for LE host and 1 for BE host as system info reply

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-61032	815490	56440_A0 56440_A1 56450_A0 56440_B0 56450_B0	<p>In Katana & Katana2 devices, DDR Phytune command was causing high CPU utilization and parity errors during SDK init.</p> <p>This was because of parity errors in Channel interfaces were not handled and were also checked for interfaces that were not used.</p> <p>Fixed this issue by modifying CI error mask to handle parity errors and also enabled parity checks for only those interfaces supported in device</p>
SDK-61063		56340_A0	<p>There was memory leak inside <code>bcm_regex_match_set()</code> API in earlier release which was causing all system memory to be consumed when <code>bcm_regex_match_set()</code> was called repeatedly for ApplQ regex signature configuration, Fixed this API to avoid memory leak by freeing allocated memory when not in use.</p>
SDK-61073	813684	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	<p>In the previous release, BCM_L3_TRILL_ONLY/ BCM_L3_VXLAN_ONLY/ BCM_L3_L2GRE_ONLY were not saved. In this release, these flags are saved now.</p>
SDK-61093		88650_A0 88650_B0 88650_B1 88660_A0	<p>Fix <code>stat_if_pkt_size</code> documentation in config files.</p>
SDK-61094	822899	56640_B0	<p>Description: EGR_VLAN_XLATE table can't be recovered when the flow continue hit the entry</p> <p>Root cause: When the flow hits a corrupted EGR_VLAN_XLATE entry continuously, the EGR_VLAN_XLATE SER ERROR interrupt should be raised and then SER handler would be invoked to correct it. Unfortunately, there is another SER ERROR interrupt being raised meanwhile which notified SOP CELL error and packet drop happened. Every corrupted EGR_VLAN_XLATE entry hit will trigger this interrupt until the EGR_VLAN_XLATE ser error has been corrected. So the interrupt handler may go into a dead loop of handling the continuously SOP CELL error and packet drop interrupt since there is no opportunity to correct the corrupted EGR_VLAN_XLATE entry.</p> <p>Solution: Current software has changed the code to ensure handling EGR_VLAN_XLATE entry correction before the SOP CELL error notification.</p>
SDK-61096		56340_A0	<p>In earlier releases, there was no memory limit on the allocation for regex expansion which causes memory corruption in other applications as well. Boundary has now been set for not more than 100MB.</p>

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-61144		56063 56064 56062 56060 53400_A0 53406_A0 53402_A0	Apply the software workaround as suggested for 10G TSCPLL for the case that strap status of <code>lcp111_refclk_sel</code> equals to 0 on Greyhound platofrms.
SDK-61171	823889	56340_A0	Issue :- <code>bcm_ipmc_add()</code> calling failure after SDK upgrade to 6.3.7 from 6.3.4. Fix :- Issue was due to Code changes that were added to verify if entry is not present and REPLACE flag is passed, the API should return <code>BCM_E_NOT_FOUND</code> . Reverted the code change.
SDK-61199	819962	All 56850_A0 56850_A1 56850_A2	Added support for transmitting ETAG packets out of the extender port without modifying the ETAG.
SDK-61244	824828	88660_A0 88670_A0	In Field processor, when creating a new entry for direct extraction Database, the entry ID is taken from a range of Direct extraction pull, which comes after the TCAM ID pull. Thus, this range is offset by the number of TCAM entries (Internal and external). This maximum number of TCAM entries was defined incorrectly. It is fixed.
SDK-61251	814934	88660_A0	If there are 2 task to call <code>bcm_l2_learn_limit_set()</code> , the limit number in ARM wasn't same with the number at SDK at sometimes. Now add a lock to fix the issue.
SDK-61261		88660_A0 88670_A0	BFD: Adding BFD echo support. For detailed information refer to the User Manual.
SDK-61267		88650_B0 88660_A0	In FCoE VSAN assignment, when mode was updated to "VSAN from VSI" before enabling the NPV mode the VSAN assignment is not updated correctly. This is fixed.
SDK-61270		88650_B0 88650_B1 88660_A0	In FCoE, VSAN mode update did not updated the NPV programs in the FLP. Now, it is updated when NPV is enabled.
SDK-61276		88660_A0	OAM: In the APIs <code>bcm_oam_loss/delay_add()</code> , Fixed a bug concerning the field <code>peer_da_mac_address</code> . Under certain circumstances MSBs of the DA address of outgoing LMMs/DMMs wasnt properly set.
SDK-61281	811786	56456_B0 56640_A0 56450_A0 56640_A1 56640_B0 56455_A0 56456_A0 56450_B0	Fix contains changes for <code>bcm_port_timesync_config_get()</code> API which has a minor variable initialization error causing breaking of API functionality. Also syntax for deleting timesync configuration using <code>bcm_port_timesync_config_set()</code> API is documented.
SDK-61282		All	Diagnostic shell now prints timestamp with PTP events.
SDK-61301	813742	88650_A0 88660_A0	In LAG and ECMP, Load Balancing is not working according to native Ethernet for VxLAN packets. This is fixed.
SDK-61316		88650_A0 88650_B0 88660_A0	In FCoE application, key construction for LPM lookups in forwarding stage (FLP) when working with VSAN from VSI mode was built wrong. key construction fixed to be consistent with entries added to DB.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-61317		88650_A0 88650_B0 88660_A0	OAM: When using <code>bcm_oam_action_set</code> , deletion and creation of new endpoint was failing after a few iterations.
SDK-61323		88650_A0	BFD: For BFD endpoints of type other than <code>bcmBFD TunnelTypeUdp</code> , <code>bcmBFD TunnelTypeMpls</code> , if the flag <code>BCM_BFD_ENDPOINT_IN_HW</code> is not set (endpoint is not accelerated to the OAMP) then the field <code>local_discr</code> is not used. Thus this field must be zero when <code>bcm_bfd_endpoint_create()</code> is called in such cases. Likewise in all cases where the field <code>local_discr</code> is required, upon calling <code>bcm_bfd_endpoint_create()</code> correctly verify that the MSBs are consistent with prior endpoints added (The MSBs act as a global value per device). For further detail consult the user manual.
SDK-61373		88650_A0	BFD: Enable configuration of different BFD types (IP/MPS/MPLS-TP) on same device.
SDK-61381	825267	All	RX LOS application is used to detect, monitor, and manage RX loss of signal (LoS) indications from the BCM88650 and BCM88750 SerDes interfaces. Function <code>rx_los_unit_detach</code> used to detach unit from application and mask all the relevant to interrupts. Added a function (<code>rx_los_sw_unit_detach</code>) which allows to perform SW detach without any access to device (skip interrupts masking)
SDK-61406 SDK-61530		56440_A0	A scheduling change failed after warmboot if dynamic sched change algorithm was enabled. The missing warmboot data is now constructed and able to change sched mode after warmboot .
SDK-61442		88650_A0 88650_B0 88650_B1 88660_A0	Counter thread need to count for all ILKN channels when counting in per channel mode. Fixed.
SDK-61477	810833	88650_B1 88660_A0	As part of SER interrupt handling, the SDK can attempt to write to the corrupted memory. This is not allowed for dynamic memories, i.e. memories that are maintained by the device and not by CPU configuration. The interrupt handling logic did not account for this limitation. For most memories, trying to override a dynamic memory is protected by HW and would fail with error message. Some dynamic memories are not protected, writing to these memories could result in unpredictable device state. The fix does allow caching of 2 static memories which listed at dynamic list by mistake: <code>OAMP_RMEP_DB</code> <code>OLP_DSP_EVENT_ROUTE</code>
SDK-61484		56514_A0	During fp re-installation removed action items from software were not getting deleted in hardware. Reason for this behavior was during re-install action items marked as invalid were not skipped . Added a check to skip invalid action items from re-installation.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-61486	826612	88650_B1 88660_A0	cint_vswitch_vpls_roo.c is fixed to configure correct Link Layer (instead of Dst MAC 0)
SDK-61503		88650_A0	OAM: 88650_A0 does not support 1588 timestamp format. Validity check is added in bcm_oam_endpoint_create().
SDK-61505	826710	56334_B0 56334_A0	Add bcm_switch_pkt_info_hash_get API support for Enduro.
SDK-61507		88660_A0	BFD: For BFD over IP packets generated by the OAMP the UDP checksum should be set to 0 (none) as opposed to 0xffff.
SDK-61526		88750_A0 88750_B0	BCM88750 standalone compilation might fail due to linkage errors. Fixed.
SDK-61527		88660_A0	MPLS PORT: In case of ingress only configuration, learning was not configured properly.
SDK-61545	827380	56440_B0 56240_B0	In the earlier release, the "vlan translate action add" was not working when "policer" was not specified. This was because the default value of Policer was being set to "None" which is not its default value. The issue has been fixed by setting the default value to 0 in the current release.
SDK-61613		88660_A0	OAM InLIF profile can now be mapped to 4 default OAM trap profile. Each such trap profile can define a default endpoint to trap OAMoEth packets arriving on an InLIF with the suitable profile. Call sequence: 1. bcm_port_control_set - to map an InLIF profile to a OAM-Trap-profile (In simple InLIF profile mode, this sets the InLIF profile bits) 2. bcm_oam_endpoint_create with: - id = BCM_OAM_ENDPOINT_DEFAULT_INGRESS 0/1/2/3 (depending on the OAM-Trap-profile chosen in the previous step) - level = packets with MDL<=level will be trapped by that endpoint A default Up-MEP can also be created using: 1. bcm_oam_endpoint_create with: - id = BCM_OAM_ENDPOINT_DEFAULT_EGRESSO - level = packets with MDL<=level will be trapped by that endpoint - flags = BCM_OAM_ENDPOINT_UP_FACING for example sequence see cint_oam.c
SDK-61616		88670_A0	In BCM88670, QOS support for ROO application. Add support for PCP DEI profile in ROO application, using a new PCP DEI profile table. PCP DEI profile table: for a PCP DEI profile get the indication if the PCP DEI is mapped from DSCP or mapped from TC DP.
SDK-61619		88650_A0 88660_A0 88650_B0	For external Tcam (KBP) connection, use this soc property (ext_ilkn_reverse) to indicate the numbering mode of the lanes in KBP. default value for the system is 1 (reversed).
SDK-61635	828430	88030_A0 88030_B0	Note.
SDK-61636	828428	88030_A0 54880_B0	Fixed

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-61641	827595	56850_A0 56850_A1 56850_A2	In earlier releases, port control bcmPortControlSerdesReset was effective only once, after a power-on-reset. That's because the code did not reset TSC, only took it out of reset. This has been fixed in this release by resetting TSC first then taking it out of reset.
SDK-61652	826180	56640_A0 56640_A1 56640_B0	Problem Statement: The switch was hanging during exit clean with multiple VLAN_XLATE entries. Resolution: The hang was due to an uninitialized variable that was causing infinite loop. The variable is now initialized so that it picks appropriate value.
SDK-61659	828631	56450_A0 56450_B0	In the earlier release, a memory corruption issue was seen due to lesser memory allocation which did not accommodate the PP ports. The issue has been fixed by increasing the allocation size to include the PP ports in the current release.
SDK-61663		88650_A0	OAM: For Y.1731 over MPLS, allow classification of DMMs/DMRs with the field Version on the CFM header set to 1 (this is the value defined by Y.1731).
SDK-61672		88670_A0	MPLS: In case OAM is not enabled on the device, Router Alert is not set as special label. It can be trapped using the lif table as a non-special label.
SDK-61727	796172	All 56850_A0 56850_A1 56850_A2	In the previous release, when updating the extender port id which had been added into a VPLAG, the field SOURCE_VP in the table VLAN_XLATE would be mistakenly modified from the VP of the VPLAG to the VP of the extender port id, which was incorrect. In this release, this issue has been addressed by updating the field SOURCE_VP in the table VLAN_XLATE with the VP of the VPLAG.
SDK-61813	825520	56440_A0 56440_B0	In the earlier version, entry pointed by EGR_VLAN_CONTROL_1r and TPID reference count was not handled correctly. The fix correctly handles the problem and the issue is not seen.
SDK-61818	830940	56846_A0 56640_A0 56850_A0	In the previous release, the API bcm_multicast_egress_set couldn't delete the given L3 INTF from a multicast group with a specific sequence. In this release, this issue has been addressed by correcting the internal function of comparing the software status and hardware resources.
SDK-61823		88650_A0	Upon SER event, the driver collects information identifying the affected table entry. This information was not sampled correctly for IQM Parity-protected tables, and for OLP ECC-protected tables. As a result, prior to this fix parity errors in any IQM table, as well as ECC errors in any OLP table, were not handled by the appropriate corrective-action. However, the SER event detection and reporting functioned properly.
SDK-61855	831077	56834_A2	In previous releases, the L2_entry table size was fixed to 160K. The modification was to allow user to set the L2_entry table size to the capacity under 160K.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-61869		88660_A0 88670_A0	When creating OAM default endpoint in the egress, use the ID defined as <code>BCM_OAM_ENDPOINT_DEFAULT_EGRESS0</code> for BCM88660 <code>BCM_OAM_ENDPOINT_DEFAULT_EGRESS0/1/2/3</code> for BCM88670 and above.
SDK-61883		88660_A0	OAM: New oam classifier mode is available. It enables adding up to 8 MEPs and up to 2 MIPs on each lif, in any direction (up/down), but reduces available amount of endpoints to 8K. To enable the new mode set soc property <code>oam_classifier_advanced_mode</code> to value 2.
SDK-61901		88650_B1	Added "init rx_loss" to init commands usage.
SDK-61915		88660_A0	OAM: Incorrect mp-profile allocation at the OAM classifier may cause incorrect behavior of endpoints set on different directions of the same lif.
SDK-61934		88660_A0	OAM: Fixed bug causing <code>bcm_oam_loss_add/delete()</code> , <code>bcm_oam_delay_add/delete()</code> to misbehave. Error may be returned after any several calls to add APIs, each for different endpoints.
SDK-61938		56640_A0 56640_A1 56640_B0	Internal phy driver was not able to attach to few XTPORT block ports of triumph3 devices. A change made for a separate issue was blocking PLL reset for few block of the ports. This has been resolved
SDK-61949	830803	56850_A0 56850_A1 56850_A2	Previously, due to sharing of the fundamental interrupter handler, <code>bcmlINTR</code> would be woken up on each received knet interrupter even though there wasn't any one interrupter that needed to be processed by <code>bcmlINTR</code> . This reduced the system performance. This has been resolved by identifying the interrupter's owner, the fundamental interrupter handler won't wake up <code>bcmlINTR</code> thread if there isn't <code>bcmlINTR</code> -related interrupter.
SDK-62000	815947	All	Fixed usage of allocated-but-not-initialized memory by PTP module.
SDK-62002		All	Fixed enabling using the RPT link-integrity bitmap with <code>IreTdmMask</code> , only when calling <code>bcm_fabric_tdm_direct_routing_set()</code> , and not on init. In register <code>FDT_LINK_BITMAP_CONFIGURATIONr</code> field <code>IRE_TDM_MASK_MODEf</code> , The default mode is 2 - which means using RTP reachable bitmap. When calling <code>bcm_fabric_tdm_direct_routing_set()</code> The mode changes to 3 - which means using RTP link-integrity bitmap.
SDK-62018	826377	88650_B1 88660_A0	A bug caused extra bytes to be added to RSPAN packets in case a user header was used.
SDK-62022		88660_A0	MPLS PORT: Push-Profile allocations are mismanaged in PWE ingress only mode.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-62025 SDK-59407	832310	56850_A0 56850_A1 56850_A2	In earlier releases, when multiple trunk member ports were in failed state, deleting trunk member threw error. This has been fixed in this release by just skipping hardware configurations and returning ok, when multiple trunk member ports are in failed state. Fix a bug that when multiple trunk member ports are in failed state, deleting trunk member will throw error.
SDK-62032		88650_A0 88660_A0 88670_A0	In LAG and ECMP, Load Balancing is not working according to native IP for VxLAN packets. This is fixed.
SDK-62062	829166	88650_A0 88650_B0 88650_B1 88650ACP_A0 88660_A0 All	MPLS PORT: Flag BCM_MPLS_EGRESS_LABEL_TTL_DECREMENT is no longer mandatory for bcm_mpls_port_add api with action PHP.
SDK-62063		88650_A0 88650_B0 88650_B1 88660_A0	Port protocol based vlan classification. Configuration of vlan per port and trill/mpls ether_type, using the api bcm_vlan_port_protocol_action_add was causing a new user defined ether_type protocol, instead of being identified as trill or mpls ether_types.
SDK-62102	832410	53284_A0	Fixed the issue on ROBO chips thin which vlan information was not restored to default STG when removing the vlan from other STG.
SDK-62106	832908	56450_A0 56450_B0	In earlier releases INNER_TPID_ENABLE field in SYSTEM_CONFIG_TABLE was not configured during initialization for Katana2. This issue has been resolved now by setting INNER_TPID_ENABLE to 1 during port initialization.
SDK-62124		88750_A0 88950_a0	FE have limitations on its local modid values: BCM88750: 0-127 BCM88950: 0-143 But the limitations for FE13 local modid values are different, so FE13 the limitations have been changed to: BCM88750: 0-63 BCM88950: 0-71
SDK-62126		88650_B1 88660_A0	In L3, when using internal LPM routing table, adding and updating routes may cause memory overruns: Fixed possible memory overrun in DMA buffers or in adjacent memory. The memory overrun was up to 240 bytes after the buffer for route adds.
SDK-62209	834157	88650_B1 88660_A0	L3: L3 egress can be created by calling bcm_l3_egress_create. When creating L3 egress with voq_mapping_mode=DIRECT and egress.port (system-port) is bigger than 4K, API return ARAD_PP_FRWRD_FEC_DEST_VAL_OUT_OF_RANGE_ERR. L3 egress can be created correctly after the fix.
SDK-62222	826150	All	In the previous release, the BCM SHELL command "mc show" showed some interfaces in the given group which they didn't belong to, which was incorrect. In this release, this issue has been addressed by correcting the internal function of comparing the software status and hardware resources.



Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-62227		88650_A0	BFD: For incoming packets classified as BFD IP-multi-hop (according to the UDP dest-port), packet's destination IP address is compared to the endpoint's source IP address, which is configured in the field <code>src_ip_addr</code> in the API <code>bcm_bfd_endpoint_create()</code> . Up to 16 different source IP addresses are available per device.
SDK-62296		56440_A0 56850_A0	PTP subsystem now supports stack deletion, firmware reload, and restart.
SDK-62333		88660_A0	BFD: InLIF profile can now be mapped to 4 default OAM trap profile. Each such trap profile can define a default endpoint to trap BFD packets arriving on an InLIF with the suitable profile. Call sequence: 1. <code>bcm_port_control_set</code> -to map an InLIF profile to a OAM-Trap-profile (In simple InLIF profile mode, this sets the InLIF profile bits) 2. <code>bcm_bfd_endpoint_create</code> with: - <code>id = BCM_BFD_ENDPOINT_DEFAULT0/1/2/3</code> (depending on the OAM-Trap-profile chosen in the previous step) - <code>remote_gport = port</code> to trap to (defaults to the CPU port) for example sequence see <code>cint_bfd.c</code>
SDK-62336		88660_A0	OAM: When a MEP is configured on a LIF after a MIP is already configured on it, device might not handle MIP packets as expected.
SDK-62355		88650_A0 88650_B0 88650_B1 88650ACP_A0 88660_A0 88670_A0	MPLS PORT: Enable replacing <code>encap_id</code> in case PWE is protected
SDK-62364		88650_A0 88660_A0 88670_A0	Direct ingress queues to rate class mapping, and using an internal mechanism to handle the mapping is mutually exclusive. Blocked using direct ingress queues to rate class mapping after using the internally handled mechanism is blocked. relevant API: <code>bcm_cosq_profile_mapping_set(int unit, bcm_gport_t gport_to_map, bcm_cos_queue_t cosq, uint32 flags, bcm_switch_profile_mapping_t *profile_mapping);</code> MACROS: <code>BCM_GPORT_PROFILE_GET/SET,</code> <code>BCM_GPORT_IS_PROFILE;</code>
SDK-62367		88650_A0	OAM: Added "diag oam exact_match LIF=N" : display O-EM 1 entry by LIF
SDK-62379	833677	All	Default threshold for traffic classes is changed with respect to number of service pools used, if only 1 SP is used, only SP 0 will be configured for all dp's, otherwise, dp's 0-3 will be configured for SP 0 and dp's 4-7 will be configured for SP 1
SDK-62421		88660_A0	OAM: Inconsistent behavior might be seen on LIFs on which MIPs and MEPs were created and action set was used to modify destination.
SDK-62432		88660_A0	OAM: Calls to <code>bcm_oam_loss/delay_delete()</code> may interfere with the DA address of other available LMM/DMMs transmitted by the OAMP.

Table 140:

Number	CSP #	Chips	Release Notes For 6.4.2
SDK-62437	836260	88750_A0	BCM88750 load failed when soc property <code>backplane_serdes_encoding</code> is not explicitly defined. Fixed. In such a case BCM88750 encoding will be set to "KR_FEC",
SDK-62492		88660_A0	OAM: when creating accelerated endpoints of type <code>bcmOAMEndpointTypeBHHMPLS</code> , the MEP DB entry may not be properly updated, causing subsequent calls to <code>bcm_oam_loss/delay_add()</code> to fail.
SDK-62533	835794	56450_B1 56450_A0 56450_B0	Issue: Field <code>InPorts</code> Qualifier for max supported ports on KT2 device was failing to recover during level-1 Warmboot. Root Cause: During recovery, the group is constructed for all the ports in the device by retrieving from an internal macro which is updated by a port structure. However, during init time, the flex ports were not part of this macro due to which, the group pbmp and hardware configured pbmp value mismatches and returns an internal error. Fix: During recovery, the group is constructed for all the supported ports in the device by retrieving the flex ports explicitly.
SDK-62548		88660_A0	OAM: When a MIP is deleted, if no other MIP is defined on the same LIF, the remaining MEPs might not act as expected.
SDK-62550		88650_A0 88650_B0 88660_A0	For IPv4 MC, key construction is not performed correctly when using external TCAM (KBP). The in-RIF field is 12 bits, but for the KBP extra 4 bits are needed. A new instruction added to the key in order to use extra 4 zero bits.
SDK-62721		88650_A0 88660_A0	OAM: calls to <code>bcm_oam_endpoint_create</code> with the flags <code>BCM_OAM_ENDPOINT_REPLACE</code> and <code>BCM_OAM_ENDPOINT_REMOTE</code> may fail due to uninitialized values.
SDK-62817		88650_A0 88660_A0	OAM: Remove duplication, unused code from <code>cint_oam.c</code>
SDK-62820		88660_A0	OAM Classifier behavior is inconsistent after adding and removing endpoints on same lif.
SDK-63048		88650_A0	The name of <code>cint_oam_y1731.c</code> has been changed to <code>cint_oam_y1731.c</code> . This file provides examples of creating Y.1731 OAM endpoints over MPLS-TP, PWE.
SDK-63822	847793	88650_A0 88650_B0 88660_A0	<code>bcm_cosq_gport_stat_get()</code> : Fix counter processor statistics gathering for egress queues when a queue bundle size is different from 8. Input parameter - <code>cosq</code> is now used as a traffic class parameter only, not as a queue bundle offset.

Section 14: Resolved Issues for 6.4.1

The following issues are resolved in version 6.4.1 of the SDK.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-32461		56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56746_A0 56745_A0 56744_A0 56743_A0	<p>Problem: WRED thresholds were not taking effect because of hardware issue.</p> <p>Solution: Implemented workaround in software to get WRED memories into stable state.</p> <p>This workaround does below thinks to put WRED memories in stable state.</p> <ol style="list-style-type: none"> 1. Selects 4 Ethernet ports (one extended queue port from X,Y pipeline and one regular port from X, Y pipeline. 2. Configures all 4 ports in MAC loopback. 3. Disables CRC re-calculation on all 4 egress ports. 4. Enables CRC checks and configures <code>ING_PRI_CNG_MAP</code> table to mark incoming traffic with red color. 5. Add's l2 mac address in l2 table to switch the packets to all 4 ports. 6. Prepares 8 multi cell unicast SOBMH packets, configured DMA descriptor's and starts DMA engine. <p>These SOBMH packets will be loopbacked with bad CRC since we disabled CRC re-calculation on egress ports and switches to all 4 egress ports. Since these packets have CRC errors MMU will drop the packets after receiving EOP and stabilizes the WRED memories.</p>
SDK-34523		56820_A0 56820_B0	In previous SDK, customer found a crash on 56820 when the SDK was handling a MMU parity error. The root cause of this crash was the SDK visited an unavailable memory. Now this issue has been resolved.
SDK-36232	460304	All 56850_A0	In previous release, A L2 multicast with flag <code>BCM_MULTICAST_WITH_ID</code> and <code>Group_ID</code> was created by <code>bcm_multicast_create</code> , but the HW index in <code>Group_ID</code> was already occupied by other multicast group, than the existing entry could be overwrite and return <code>BCM_E_NONE</code> . In this release, it will return <code>BCM_E_EXISTS</code> and won't overwrite the existing entry.
SDK-38881		All	<code>bcm_port_priority_color_set</code> is modified to set color as none when color param to API is <code>bcmColorPreserve</code> .
SDK-41357	469082	56842_A0	There is an issue with the h/w logic related to the parity generation and checking for the <code>PORT_CBL_TABLE</code> memory. In this release occasional spurious reports of a parity error in <code>PORT_CBL_TABLE</code> has been fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-42031		88650_A0	Error indication was added to prevent AC P2P to PWE or to Mac-In-Mac inner vlan editing (VID-2 in LIF table) which is not supported by HW.
SDK-42289	565794	88650_A0	Static forwarding (i.e. <code>bcm_port_force_forward_set</code> API) can be used both in TM and PP modes. Some fixes are done to enable it also in TM mode.
SDK-42527		88650_A0	SDK-42527: Support TR 90 and TR91 for the ARAD.
SDK-42957	580600	88025_A0	Support for Down MEP on VPWS/VPLS attachment circuits is fixed.
SDK-42987	580192	56850_A0	Legacy method to add route entries to LPM table may trigger re-shuffling logic which could lead to massive HW entry movement. In the worst case, the memory read operation times of LPM tables could be very high and eventually cause bad route convergence time. A request was received to reduce the totaled time costs for bulk route add operation. This was achieved by enabling soc memory cache for route add/delete operation to reduce memory read time costs. Can be turned on/off by switch control <code>bcmSwitchL3RouteCache</code> . Read HIT* bits could be wrong during caching time.
SDK-44506	593957	56842_A0	Added a new soc property (<code>L3_DISABLE_ADD_TO_ARL</code>) to restrict applications to create L2 interface entry and L3 interface entry separately. SDK uses this property to recover the association between L2 entries and L3 entries during warmboot. Currently, during warmboot, SDK associates L3 entries with L2 entries assuming that they were created with <code>BCM_L3_ADD_TO_ARL</code> flag during <code>l3_intf_create</code> . Later, when L3 interface is deleted, SDK deletes the L2 entry also.
SDK-44591		56840_A0 56640_A0 56640_A1 56640_B0	Current implementation is not in-line with the issue. Function: <pre>wcmmod_esm_serdes_control_get(int unit, int lane, soc_phy_control_t type, uint32 *value)</pre> <pre>case SOC_PHY_CONTROL_DUMP: rv = wcmmod_uc_status_dump (unit, port, NULL); break;</pre> 'value' variable is not being used in this call.
SDK-44736		56850_A1	The 3-lane TSC configuration is now transcribed properly to TSC 31.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-44989		88660_A0	Supporting OAMP protection packets in 88660. To enable this feature, call <code>bcm_rx_trap_type_create()</code> with the flag <code>WITH_ID</code> , <code>trap_type</code> <code>bcmRxTrapOampProtection</code> and a trap id in the range 0x400 0x4ff, followed by <code>bcm_rx_trap_set()</code> with the trap id created in the above API, and a <code>bcm_rx_trap_config_t</code> with the field <code>dest_port</code> set to the destination of the protection packets. All other fields should remain blank (an example of this is found in <code>cint_oam.c</code>). Whenever an OAM event occurs, a protection packet of size 71 bytes will be sent to the destination selected above. The format of the Protection packet at the CPU will be FTMHoPPHoFHEI. The FHEI.CPU-TRAP-CODE field will be set to the LSB of the trap id selected in <code>bcm_rx_trap_type_create()</code> . The size of the protection packet will always be 71 bytes where the OAM events will be on the bottommost part of the packet.
SDK-45246		56840_A0	Implemented "bcmFieldActionL3ChangeMacDa" and "bcmFieldActionL3ChangeVlan" actions for TD2 device, TR3 and KATANAx devices. The actions expect the egress-object (l3 next hop index) which should be already created by L3 module and the ID should not be associated with any other modules. The actions are to replace the destination MAC and VLAN on the matched incoming packet with the MAC and VLAN associated with the given next hop index.
SDK-45535		88650_B0	OAM endpoint: It is now possible to create an OAM endpoint over LAG. This is done by adding the endpoint on the LAG port and separately configuring the mac address of the endpoint on each one of the LAG ports using <code>bcm_l2_station_add</code> . An example can be found in <code>cint_oam_over_endpoint.c</code> . In addition CINT includes a cleanup function, and an option to set VLAN-Ports lifs over lag without defining an OAM endpoint.
SDK-46635	625709	56640_A0 56640_A1 56640_B0	Added a new SOC property "ext_tcam_request_response_latency" and a new "tcam latency" bcm shell command for TR3 with external TCAM. Customers can execute the bcm command to calculate the latency and then specify the SOC property using the latency value in config file.
SDK-46641	633505	88650_A0 88650_B0	When running 802.3 llc packets, the ethernet field is used as length. There was a bug that if the length was set to 0, the packet would have been parsed as a trill packet.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-46757	636270	56643_A0	Triumph_3 has a Unified Forwarding Table and hash selection for L2/L3/Vlan/Mpls tables has to be programmed differently. HASH_CONTROL register does not have a L3_HASH_SELECT field. The command "l3 l3table hash" is trying to access this non-existent field resulting in an assertion failure. Support for "l3 l3table hash" and "l3 l3table ip6hash" commands have been implemented.
SDK-46833		56440_B0	The fix for this issue checks the PLL's current divider setting in register 0x8050, and then use this as the forced value instead of always forcing the PLL to the same frequency for PRBS test. Additionally, the asymmetric mode for the 40nm B0 core has been disabled for the PRBS function to work.
SDK-47665	650917	56854_A2 56854_B0 56854_A0	1G configs should be supported in latest release.
SDK-47824	636400	56846_A0 56845_B0 56845_A2 56844_A0 56842_A0	In previous releases, the updating on NONUCAST_TRUNK_BLOCK_MASK table took a fair amount of time to complete on Trident+ which has more NONUCAST_TRUNK_BLOCK_MASK table entries with the plain read API soc_read_mem(). In this release the performance has been improved by changing the update method such that it now uses a single DMA operation soc_read_mem_range() which gives a good amount of speedup to applicable trunk APIs.
SDK-47983	661534	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2	New cosq type bcmCosqControlEgressPortPoolYellowLimitBytes / bcmCosqControlEgressPortPoolRedLimitBytes have been added for configuring yellow/red limits. Added one service pool type bcmCosqServicePoolPortColorAware and bcm_cosq_service_pool_set/get APIs for per port per service pool color aware enable.
SDK-48016	661903	56840_A0	In the previous release,static multicast L2 entries were getting flushed with bcm_l2_addr_delete_by_port() API.This has been resolved.
SDK-48101	689094	56845_B0 56845_A2 84740_A0 84784_A0	Support for 40G repeater mode PRBS for the system side was missing for BCM84740. Support for the same is added and tested in this release.
SDK-48140		88650_A0	TRILL BEHAVIOR CHANGE. According to trill fgl rfc, at ingress trill fgl, native outer and inner tpids must have value 0x893b. So far in Trill application, native Ethernet tpids set outer tpid = 0x8100 and native inner tpid = 0x893b. New implementation is now aligned to trill fgl rfc. At ingress trill fgl, both native inner and outer tpids have value 0x893b. Settings are done using VLAN-editing and work for both normal vlan translation and Advanced modes.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-48296		88650_A0	When working with external TCAM, a master-key is sent from BCM886XX to KBP device with all the necessary fields for the forwarding and external ACL lookups. A diagnostic has been built to show the order of the fields in the master-key and master-result: BCM> kbp print master
SDK-48404	654018	56845_B0	For BCM56845m phy_wc40_ability_remote_get was unable to get the correct ability when the link partner did not enable CL73. Corrected the ability to obtain remote ability when the link partner doesn't enable CL73 on TD+/WC40.
SDK-48577 SDK-47609		56640_A0 56643_A0 56340_A0 56640_A1 56643_A1 56640_B0 56643_B0 56540_B0 56045_B0 56040_A0 56547_A0	Added cpu based UFT mem sweep to detect and fix parity errors. Fixed issues found with graceful lookup error handling.
SDK-48774 SDK-56539	672146	88650_A0 88650_B1 88660_A0	IMPORTANT CHANGE (MIRROR SEQUENCE): RSPAN Mirroring: ingress and egress settings have been decoupled. Mirroring into RSPAN is now done in the following way: 1. Create L3 interface (this has not been changed). 2. Set a bcm_tunnel_initiator_t object with type=bcm_TunnelTypeRspan, vlan, tpid, pkt_pri configured as desired and call bcm_tunnel_initiator_create(). This allocates entries in the EEDB. 3. Set a bcm_mirror_destination_t with the flag BCM_MIRROR_DEST_TUNNEL_WITH_ENCAP_ID set and with the encap_id field set to the tunnel_id returned from bcm_tunnel_initiator_create() using the macro BCM_GPORT_TUNNEL_ID_GET(). Getting information on the RSPAN tunnel may be done by setting a bcm_l3_intf_t object with the field l3a_tunnel_idx set to the tunnel_id returned from bcm_tunnel_initiator_create(), via the macro BCM_GPORT_TUNNEL_ID_GET(). Destroying the tunnel (freeing the EEDB entries) may be done with the API bcm_tunnel_initiator_clear(), with the l3a_tunnel_idx field set as in the get() API. Destroying the mirror not been changed. For an example see mirror_with_rspan_example() in cint_mirror_erspan.c
SDK-49047		88650_B0 88650_B1 88660_A0	1588 packets were stamped while received/send from/to ports in which 1588 was disabled. This is fixed. ISSU perspective: The fix supports ISSU if all the ports supporting 1588 are disabled before ISSU, and enabled after ISSU.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-49202		56640_A0 56640_A1 56640_B0	SOC properties ext_tcam_tx_driver_current, ext_tcam_tx_postcursor_tap, ext_tcam_tx_main_tap have been added to describe driver current, postcursor tap and main tap for NL11K serdes TX direction parameters, and SOC property ext_tcam_rx_gain has been added to describe RX gain for Serdes RX direction parameter.
SDK-49205		56640_A0 56640_A1 56640_B0	Support for ESM interrupt was added in CMIC level interrupt handler. Once the ESM fatal errors are detected, the new-implemented "esm recovery" thread will be woken to restore ESM.
SDK-49249		88650_A0 88650_B0 88650_B1 88660_A0	DPOE application: Added an application example to show how to classify L2/L3/L4 header field and mapping frames to PON LIFs. More information is provided in cint_pon_dml_fec_app.c.
SDK-49543	663298	88650_A0 88650_B0 88660_A0	Fixed ARAD ports Leds in Negev chassis (updated the Led microprocessor program to match recent changes in \$SDK software)
SDK-49694		56640_B0 56850_A1 56850_A2	Please use lpm_scaling_enable=1 in config to have the ability to add 64bv6 entries in paired tcam. When this config is enabled, V4,64B V6 entries can be added in the unreserved paired tcam. If lpm_ipv6_128b_reserved=0, then no tcam space is reserved for 128B V6 entries and complete paired TCAM can be used for 128BV6, V4, and 64B V6 entries. Please note that each entry of 64B V6 entry in the paired TCAM uses 2 indexes of L3_DEFIP view where as in unpaired TCAM ,it uses only 1 entry
SDK-50216	693383	56850_A0	In previous release, per VLAN VP replication was automatically enabled when a Gport adds to the VLAN. In this release, support has been added to control VP replication by bcm_vlan_control_vlan_set.
SDK-50389	695476	2000_A1	QE2000 fix when updating QOS parameters for a given multicast queue. The unicast queue configuration was incorrectly being updated when egress independent flow control is enabled.
SDK-50431		88660_A0	ERSPAN on XGS MAC extender system is now supported

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-50591		88650_A0 88650_B0 88660_A0	TRILL: For TRILL UC and MC egress RBridges, trill packets are classified to inLIFs whose IDs are always 0 by PORT VLAN Domain X Outer VLAN. A problem occurs when doing same interface filter. The inLIF ID and outLIF ID of trill packets of UC and MC egress RBridges are all 0. Packets are all dropped even in case it shouldn't be. A new program is added to classified trill packets into valid inLIF IDs. It's enabled by adding a soc property "custom_feature_trill_designated_vlan_inlif_<port>=<lif_id>". Once feature is enabled, soc properties should be enabled for all TRILL ports.
SDK-50755		88650_A0 88650_B0 88650_B1 88660_A0	Diag: "diag cosq qpair e2e ps=x" can be used to display e2e port scheduler model. Improvement in 1) adding new diagnose "diag cosq qpair e2e ps=x" 2) using "diag cosq qpair egq" instead of "diag cosq egq"
SDK-50760		88650_A0 88650_B0 88650_B1 88660_A0	Diag: "diag cosq print_flow_and_up" can be used to display scheduler model with given voq. Improvement in 1) adding port scheduler level between HR level and Port level 2) using "voq connector" in the diag command instead of "flow"
SDK-50899	634474	56845_B0	Updated Documentation for WRED Flags
SDK-51038	683239	88640_A0	Petra-B 88640 Ingress mirroring : Ingress mirror can't mirror the original packet for the Petra hardware limitation, the workaround for it is to configure the mirror port as RAW in config.bcm astm_port_header_type_2.BCM88640 = RAW. See cint_petra_mirror_tests.c for more information.
SDK-51292	708102	56640_A0	The HG capable dynamic ports Indexing offset was not accounted for on the SC/QM queues which led to wrong indexing for the dynamic ports. This has been fixed in the offset.
SDK-51352	708790	56846_A0 56840_A0 56846_A1	In previous versions, the routine _soc_trident_mem_parity_control() returned directly after configuring parity control for X-pipe and left Y-pipe parity control register un-configured for dual pipe IPIPE/EPIPE memories. In this version, _soc_trident_mem_parity_control() has been modified to configure both X-pipe and Y-pipe parity control.
SDK-51353		56643_B0	The new support for the below port configuration has been implemented. Device =56643 Frequency (MHz)= 450 Option = 4 GbE Port Group (XC[12:0]) = 36 x GbE+1 x GbE High Speed Port Gr 1 (WC[2:0])= 4 x XFI High Speed Port Gr 2 (WC[6:3])= 2 x HG[42] + x F.H [42]" AXP Port Guaranteed Bandwidth = 5G

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-51360	692893	56840_A0	When any module in SDK uses new stat APIs, the running STAT version changes from LEGACY to NEW. Currently VLAN Field Processor implementation works only with old stat APIs for older devices where advanced flex counters are not available. Hence, required support is added in VLAN Field Processor implementation to use new stat APIs for older devices that switched to NEW stat version.
SDK-51380		56440_A0 56440_A1 56440_B0	Enabled proper debug prints when API <code>bcm_policer_group_create()</code> fails.
SDK-51392 SDK-51964	710405	All	The BSL improvement has fixed this issue.
SDK-51464		88650_A0 88750_B0 88660_A0 88750_A0 88650_B0 88650_B1	<p>Source-routed data cells, generated by CPU, can be transmitted and received by Fabric Element (FE) and FAP devices (over fabric interface). These cells are routed according to the specific path information they carry, while disregarding the fabric reachability information. These messages are used mainly for debug and diagnostics purposes, but can be also used for CPU-to-CPU messaging.</p> <p>The previous SDK versions supported this features using SoC APIs. Instead, new BCM APIs created:</p> <pre>typedef struct bcm_fabric_route_s{ uint32 pipe_id; /* Origin fabric pipe */ uint32 number_of_hops; /* corresponds to the number of routing hops (number traversed links) */ int* hop_ids; /* traversed links */} bcm_fabric_route_t; int bcm_fabric_route_tx(int unit, uint32 flags, bcm_fabric_route_t * route, uint32 data_in_size, /* input payload size */ uint32 *data_in /* input payload buffer */); int bcm_fabric_route_rx(int unit, uint32 flags, uint32 data_out_max_size, /* maximal size of the payload buffer */ uint32 *data_out, / * output payload buffer */ uint32 *data_out_size /* actual output payload size */);</pre>
SDK-51494		88650_B0	Fix <code>cint_mpls_lsr.c</code> function <code>mpls_add_php_entry</code> . next protocol flag was overwritten by <code>BCM_MPLS_SWITCH_TTL_DECREMENT</code> flag.
SDK-51525	677768	88030_A0	<p>There was a bug in the C3 model and the XML based test framework wherein any changes to the configuration files (files of the type <code>g3p1_<xyz>.cfg.lrp</code>) did not take effect until the second run of the tests. This is because the models read in the existing configuration files first and the assembler updated them later.</p> <p>As of this release of the MDE, this has been fixed.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-51570		56850_A0 56850_A1	In previous release, NIV VP class-id setting was not supported by <code>bcm_port_class_set/get</code> API. In this release, support was added for setting NIV VP class-id by <code>bcm_port_class_set/get</code> API.
SDK-51601	708490	88030_B0 88030_A0	Egress filter issue
SDK-51617	710438	56450_A0	Issue was happening due to incorrect buffer length calculation. Function <code>_soc_mem_array_sbusdma_write()</code> is modified to use <code>chunk_entries</code> to write buffer with correct length.
SDK-51625		88650_A0 88650_B0 88650_B1	For debug reasons, an HW register is used to store the SW version used at init, and during ISSU.
SDK-51648	713425	56340M_A0 56640_A0 56340_A0 56640_A1 56640_B0	Added in the support for different freq. <code>QG_PLL</code> and <code>WC_PLL</code> for chipsets which have the H/W capability.
SDK-51658		88650_A0 88650_B0 88650_B1 88660_A0	Support the following APIs to replace properties without replacing Out-LIF discard indication: 1. <code>bcm_l3_egress_create</code> 2. <code>bcm_mirror_destination_tunnel_create</code> 3. <code>bcm_mpls_tunnel_initiator_create</code> 4. <code>bcm_tunnel_initiator_create</code> .
SDK-51707	715469	All	Optimized the ipmc performance if change 32K ipmc group from one ipmc index to another.
SDK-51725		56624_B0	SER support has been added for the following memories as part of this fix: <code>MMU_WRED_CFG_CELL</code> <code>MMU_WRED_THD_0_CELL</code> <code>MMU_WRED_THD_1_CELL</code> <code>MMU_WRED_CFG_PACKET</code> <code>MMU_WRED_THD_0_PACKET</code> <code>MMU_WRED_THD_1_PACKET</code> <code>MMU_WRED_PORT_CFG_CELL</code> <code>MMU_WRED_PORT_THD_0_CELL</code> <code>MMU_WRED_PORT_THD_1_CELL</code> <code>MMU_WRED_PORT_CFG_PACKET</code> <code>MMU_WRED_PORT_THD_0_PACKET</code> <code>MMU_WRED_PORT_THD_1_PACKET</code>
SDK-51810		88650_B1 88660_A0	Fixed three errors related to <code>bcm_vlan_port_find</code> : 1. When calling the API on an unprotected port, the <code>failover_port_id</code> field will be 1 instead of 0. 2. Any information related to 1+1 protection (<code>ingress_failover_id</code> , <code>failover_port_id</code>) was not filled when calling the API. <code>ingress_failover_id</code> and <code>failover_port_id</code> will now be filled when calling the API. 3. Added missing validations to function parameters.
SDK-51828	716994	56440_A0 56850_A0	Added new API <code>bcm_stat_flex_pool_info_multi_get</code> to retrieve the usage of flex counters in a pool
SDK-51906	699612	56450_A0	Due to flex operation issue on Cfg#12 with TDM-A2, used TDM-A3. Also corrected total slots required for TDM-A3 and removed one warning wrong comment.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-51936		56850_A2	For cl36 PRBS bus width must be set to 80 bits instead of 66 bits.
SDK-51997		88660_A0	<p>In BCM88660, in Field Processor, a new feature for field comparison is added.</p> <p>In HW, the comparison is performed on Key D in the second cycle of the PMF. It compares the two halves of the key (80 LSB bits and 80 MSB bits) and writes the result to the 5 MSB bits of the key (bits 159:155). Each bit of the result indicates a match of 20 bits, such that the 4 LSB bits correspond to 20 bits of the key, and the 5th MSB bit indicates match of the full key (80 bits).</p> <p>HW limitation: In order to act upon match the field group in the MSB must be Direct Extraction, and the 32 bit key is taken from the MSB bits (159:128). Direct extraction can filter up to 4 bits, thus only the 4 LSB bits of the compare result are used (it covers the full key).</p> <p>In SW, the sequence to enable the new compare feature is as follows: 1. Add a field group (max 80 bits each) with <code>bcm_field_group_config_create()</code> and set <code>BCM_FIELD_GROUP_CREATE_IS_EQUAL</code> flag in group. This field group will use 80 LSB bits of the key. 2. Add another field group (mode = Direct Extraction) using <code>bcm_field_group_config_create()</code> and set <code>BCM_FIELD_GROUP_CREATE_IS_EQUAL</code> flag in group. This Field Group must also add the qualifier <code>bcmFieldQualifyIsEqualValue</code> to its QSET. This field group will use 80 MSB bits of the key. * <code>bcmFieldQualifyIsEqualValue</code> qualifier is not properly part of the key (no HW instructions are allocated for it). It indicates that this key is written with the compare result. * The max size of this Field Group is 80 bits (although it is a Direct Extraction Field group, limited to 32 bits in general). The comparison is performed on the entire 80 bits and the Direct Extraction key is taken from the 32 MSB bits. * Note: the 5 MSB bits are overridden by compare result.</p> <p>The compare can be used in parallel to <code>bcmFieldQualifyCascadedKeyValue</code> qualifier.</p> <p>A new cint is added for example: <code>cint_field_dir_ext_compare_result.c</code></p>
SDK-52072	716983	88660_A0	<p>ERSPAN: Fixing a bug in <code>bcm_tunnel_initiator_clear()</code>. When a ERSPAN tunnel is created through <code>bcm_tunnel_initiator_create()</code>, two EEDB entries were allocated but in <code>bcm_tunnel_initiator_clear()</code> only one was freed.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-52087	719039	56850_A0	A customer reported an issue with EPMC Egress set performance. <code>egress_set</code> on 1000 IPMC index taking 20 seconds on 3 instances. That was happening because <code>bcm_XXX_ipmc_egress_intf_set</code> used the total interface number(max to 48K in some chips) to calculate the hash value for each port and cause the performance problem. In this release IPMC Egress set performance has been improved to take approximately half the time originally reported by reducing the total interface number to calculate and using accelerated method for MY_STATION_TCAM memory field access.
SDK-52242		56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	HW works with a single granularity value for both CIR and EIR, once the granularity value is fixed(CIR), the maximum value for EIR becomes limited. Hence the issue. On high rates (EIR > CIR) the granularity value is fixed to EIR.
SDK-52246	720771	56450_A0	The subtag packet can be multicasted/ broadcasted to multiple CoE subports at a time from CPU by using switch logic (<code>pkt->tx_pbmp</code> along with flag <code>pkt->flags=BCM_TX_ETHER</code>). For this the user needs to create L2MC entry or a VLAN with the destination CoE subports as members. The packet would go through the IP-EP pipeline where the packet may be dropped by IP or EP depending on configuration. The suggestion is to send SOBMH packets from CPU, one by one to the CoE subports instead of using <code>pkt->tx_pbmp</code> .
SDK-52287	713097	88030_A0	Bit hash ID numbering now starts from 0.
SDK-52325 SDK-51797	721812	All	There is requirement from customer to perform a loopback test on a port, while making sure it looked like totally "down" from the outside, i.e. the link is down, no traffic leakage, etc. The modification of this JIRA is to add the support of MAC loopback on disabled port.
SDK-52339	722376	56850_A0 56850_A1 56850_A2	Two data error event flags were added. If a parity error is uncorrectable, the flag <code>SOC_SWITCH_EVENT_DATA_ERROR_UNCORRECTABLE</code> will be set when SDK reporting <code>SOC_SWITCH_EVENT_PARITY_ERROR</code> event to application. If a parity error is correctable, but the error correction fails, the flag <code>SOC_SWITCH_EVENT_DATA_ERROR_FAILEDTOCORRECT</code> will be set when SDK reporting <code>SOC_SWITCH_EVENT_PARITY_ERROR</code> event to application.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-52355		56850_A0 56850_A1 56850_A2	Support has been added for retry in mem insert and delete for hash tables. Inline hash memory recovery was implemented for insert and delete operations. When an insert/delete operation encounters a parity error, the inline recovery routine will be invoked. The inline recovery routine will calculate different hash buckets in different hash memory banks based on the entry that will be inserted/deleted, then restore the each bucket in these banks. For new-added hash key types in Trident2 hash tables, support for these key types in hash entry comparing routine has also been added.
SDK-52385	721101	88030_B0 88030_A0	Byte order changed in diags mem commands as requestd.
SDK-52386	722002	88030_A0	Retry 10 times if error happen, many location for post_cmd/get_response pair.
SDK-52389	721614	56850_A0	API has been added for populating egress etag qos mapping.
SDK-52412	678409	56340_A0 84756_A0	<p>Issue: ----- When a 1G fiber SFP is installed into a BCM-84756 10G port with no fiber attached, the hardware linkscan declares the port link state as up.</p> <p>Rootcause: ----- Hardware link scan can only probe a single bit in a single external phy or internal SerDes based link register on a per port basis. For PHYs such as 84756 which needs to probe more than 1 bit on more than 1 register to conclude per port based link up/down status, these PHYs must support a "squelch" function. When the squelching function is enabled/ configured, the PHY will bring system side link status up when the line side link status is up and vice versa. Then the hardware link scan can probe the internal SerDes to conclude link status. The squelching function is not supported in 84756 driver (phy84756_fcmap.c) yet.</p> <p>Fix: ---- Implemented Squelch function in py84756_fcmap.c which is the root cause for the issue mentioned in this JIRA. Tested the squelch function implementation with 1G, 10G ports for the system side and the line side link as follows. i) Enabled system side squelch, could observe system side link goes down when Tx is disabled on the line side. ii) Enabled line side squelch, could observe line side link goes down when Tx is disabled on the system side. Squelch function is not invoked from the init function of PHY84756 fcmap driver. So user/customer needs to call squelch function explicitly using SOC_PHY_CONTROL_TX_LANE_SQUELCH whenever they wish to enable squelch on either line side Or system side.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-52442	696223	56850_A2	In previous releases, an issue was reported in the parallel vertical scan operation. When attempting a 1-D slice vertical scan, the results would not come out if <code>horizontal_min=0</code> and <code>horizontal_max=0</code> . In this release the TSC diagnostics interface has been modified to return the proper H right max and left max values.
SDK-52454		88650_A0 88660_A0	Basic bridging Egress multicast: We assume over VLAN APIs that egress multicast exist. Now no error is returned when egress multicast does not exist
SDK-52458		88650_B0 88660_A0	In L3 forwarding, when using external TCAM for forwarding tables, they were actually defined as ACL tables, resulting in a large software state and reduced efficiency in configuration. The forwarding tables in the external TCAM are now defined as LPM tables, reducing memory consumption and enhancing configuration periods. The API calling sequence remains identical.
SDK-52459		88660_A0	DEFAULT BEHAVIOR CHANGE (ARAD+ only). When using external TCAM for forwarding, RPF and forwarding searches were performed on duplicated databases. RPF and forwarding searches are now performed on a single database, using SIP and DIP respectively in search keys and resulting in increased (doubled) routing table capacity. Note that in case of IPv4/6 + RPF forwarding query, external ACL databases IDs are changed to 1 and 3. The actions sizes for ACL databases have also changed accordingly: The action size for ACL database 0 is 64 bits. The action size for ACL database 1 is 32 bits. The action size for ACL database 2 is 16 bits. The action size for ACL database 3 is 24 bits. All of the changes above apply only to ARAD+ devices. ARAD devices behavior remains unchanged.
SDK-52564		56850_A1	Fixed traffic drops observed with ingress-traffic after creation of L2GRE access port with match criteria as <code>MATCH_PORT_VLAN</code> .
SDK-52591	725728	56450_A0	Added support to enhance the number of child nodes per scheduler node in BCM56450. The restriction of maximum of 64 child nodes has been removed in cosq APIs.
SDK-52636		88030_A0	Added support for 4x10G 20x1G 1xHG TDM with specific assignment of Warp Core to CLPORT & XTPORT
SDK-52650		56960_A0	Added new Port Prbs Polynomial type <code>BCM_PORT_PRBS_POLYNOMIAL_X58_X31_1</code> .
SDK-52734		88650_A0 88650_B0 88660_A0	Indication if the Warmboot is supported by the device added to avoid Warmboot errors in regression for a version where Warmboot mode is not compiled.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-52751	726121	56545_A0 56545_A1 56545_B0	In the previous release <code>bcm_l2_cache_set()</code> did not allow setting priority > 7 on Firebolt4. In this release the API to add an entry <code>L2_USER_ENTRY</code> table now supports 4bits of PRIORITY on those devices where the priority field is 4 bits.
SDK-52789	728470	56450_A0	In the previous release an assert was raised when running TR53 on KT2. TR 53(DDR Memory Fill/Verify)is not valid for KT2. This issue has now been addressed by changing the TR rule for TR53 to exclude KATANA2. Instead of TR53 customer shall use TR140 for KT2.
SDK-52805	728606	88750_A0	FE1600: added a extra sleep after soft init and before un-isoalte
SDK-52837	729120	56840_A0 56640_A0 56440_A0 56450_A0 56850_A2	Added new field <code>rx_decap_tunnel</code> to <code>bcm_pkt_t</code> structure. This field determines the type of outer tunnel decapsulation, if any, on the received packet.
SDK-52842		56640_B0	Added external field entry move support on TR3 to manage the field entry priorities appropriately.
SDK-52871 SDK-54669	729527	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	An issue was reported with <code>MY_STATION_TCAM</code> not being correctly programmed for the trunk-based TRILL ports . To address this, trunk relevant fields in <code>MY_STATION_TCAM</code> are now correctly programmed for the termination of TRILL packets.
SDK-52892	622534	56846_A0	In the previous release, <code>bcm_port_fault_get()</code> failed on 1G SFP. This has been resolved.
SDK-52896	716978	56840_A0	Support calculating non-unicast trunk hash destination for TD/TD+/TR3/TD2.
SDK-52921	730103	88650_A0 88650_B0 88650_B1	Add entries using <code>bcm_trill_multicast_entry_add</code> with <code>c_vlan=0</code> is now supported in the following Trill mode: Trill VL (<code>trill_mode=1</code>) Multicast prune mode does not include VSI (<code>trill_mc_prune_mode=0</code>)
SDK-52942	727724	56334_B0 56334_A0	Bcm56334 10G ports has 2 different macs and both of them show counters increasing when receiving packets. In this release, a fix was introduced to address the issue where incorrect values were being retrieved. Fixed the issue to get 10G statistics only from 10G mac while speed set to 10G. And while speed set to 1G, only get statistics from 1G mac counter.
SDK-52965	730480	56634_A0	In the previous release there was no support for <code>bcmCosqStatOutBytes</code> and <code>bcmCosqStatOutPackets</code> stats in <code>bcm_cosq_stat_set</code> and <code>bcm_cosq_stat_get</code> for Triumph.This issue has now been addressed by adding the support for Triumph.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53025	729729	88030_B0	fix the issue of interlaken port disabled by linkscan thread on bcm88030 device
SDK-53028		56340_A0	Added code to find valid port block in the given list of port blocks. Previously used macro was checking only the first block in the list. The fix iterates through the entire set of valid port blocks to find the corresponding port block.
SDK-53044		56850_A0 56850_A1 56850_A2	When processing ser fifo, if the block is IPIPE block, we will only set mask upon the pipe, and take the lock of accessing sbs_control. And unlock it before returning in this function.
SDK-53046		88650_A0	In Rx thread, when parsing the packet header, the parsing was supported only for little endian. Now big endianness is also supported.
SDK-53059	730593	88650_A0	VPLS: Add support in PWE2PWE. Example in cint_vswitch_cross_connect_p2p_multi_device.c: function "run" should be called with two PWE ports and type1=type2=2.
SDK-53067	730463	All	"rtag" field removed from bcm_l2_addr.
SDK-53104	720590	56840_A0 56640_A0 56843_B0	Added support for the API's bcm_cosq_stat_sync_get, bcm_cosq_stat_sync_get32 on Trident, Triumph Family, Valkyrie. Similar to bcm_cosq_stat_get(), value returned is software accumulated counter synced with the hardware counter.
SDK-53157	732567	All	Updated API documentation that FP action DoNotCheckVlan cannot be set along with action IncomingMplsPortSet as DoNotCheckVlan is set by default whenever IncomingMplsPortSet action is set
SDK-53203	722629	88650_B0 88660_A0 88670_A0	In 6.4.1 we introduce an optimized way to decapsulate overlay headers (L2GRE and VXLAN) for the case of multicast. So far 2-pass solution was introduced where on the second pass IPMC addresses added to the Tunnel termination database in order to terminate the IP header in the second pass. Using SOC property: DEFAULT_LOGICAL_INTERFACE_IP_TUNNEL_OVERLAY_MC user can set one global LIF ID for all IPMC termination on the second pass. see cint_l2gre.c for more details.
SDK-53248		88650_A0	We exhibit the various ipmc flows via two main functions in cint_ipmc_flows.c: ipmc_flows_rif_ipmc_enabled() and ipmc_flows_rif_ipmc_disabled().
SDK-53253	731741	56334_B0	In the previous release, SDK read back whole mpls label action table to reuse existing entries when invoking the function bcm_tr_mpls_get_vc_and_swap_table_index. In this release, SDK adds an option not to reuse entries to address performance concern.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53264 SDK-63568	733415	88650_A0	<p>Allow egress snooping for MIPs with out-LIF on system headers. By default, Arad does not provide any out-LIF information when snooping OAM packets at the egress. To allow this behavior set the soc property <code>custom_feature_egress_snooping_advanced</code> to 1. When MIP packets are snooped at the egress, the snooped copy will be prepended with an FTMH and a DSP extension. <code>FTMH.DSP_EXT_PRESENT</code> will be set to 1 and the DSP extension will include the out-LIF. The snoop command for egress snooping (up-MIP) will always be 2 and for ingress (down-MIP) always 1. Thus, when changing the snooping behavior by calling <code>bcm_rx_snoop_set()</code> with 2 or 1 in the <code>snoop_cmnd</code> field, the snooping will be updated for all MIPs in the system. Likewise when OAM frames will be snooped by a MIP at the egress, the snooped copy will always have <code>FTMH.MCID_OR_OUTLIF==2</code>. By default only multicast LTM packets are snooped to the CPU. The default behavior may be changed with <code>bcm_oam_action_set()</code>. Calling this function allows setting a new snoop destination or snooping other types of frames. The calling sequence is as following: 1. Configure a <code>bcm_rx_snoop_config_t</code> with the desired behavior (i.e. probability, size, dest_port, etc.) 2. Call <code>bcm_rx_snoop_set()</code> with the <code>bcm_rx_snoop_config_t</code> configured above and the <code>snoop_cmnd</code> field set to 1 (ingress). 3. Set a new trap with <code>bcm_rx_trap_create()</code> and <code>bcm_rx_trap_set()</code>. For the latter call, the <code>snoop_cmnd</code> field in the <code>bcm_rx_trap_config_t</code> struct should be set to 1 (ingress snoop command). 4. Call <code>bcm_oam_action_set()</code> with the desired configurations. The destination field in the <code>bcm_oam_endpoint_action</code> struct should be set to the trap code from step 3 using the macro <code>BCM_GPORT_TRAP_SET()</code>. The function <code>bcm_oam_action_set()</code> will update the egress snooping configurations to match those configured above for the ingress snooping. An example of this can be seen in the function <code>mip_egress_snooping_advanced()</code> in <code>examples/dpp/cint_oam.c</code>.</p> <p>Notes: 1. In this configuration only MIP snoop is allowed (snooping MEP packets is not supported). 2. Since there are 2 snoop commands used by all MIPs in the system (one each for the ingress and egress), changing the snoop behavior for one MIP will affect all other MIPs in the system. 2. When the packet gets snoop, the forwarded copy uses forwarding strength 3. If soc property <code>block_trap_strength_pmf_0/1</code> is set to lower strength then the packet will not get forwarded. 3. To get the described behavior JIRAs SDK-54865, SDK-54726 should be used as well.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53292		88650_A0 88650_B0	new soc property - <code>scheduler_fabric_links_adaptation_enable</code> when enabled, the scheduler will take current links' states into consideration when generating credits. (mostly useful in multi stage systems)
SDK-53293		88650_A0	When trying to use the System RED Cint to configure DP discard for multiple VOQs. An error occurs after configuring the first 64 VOQs. This error was due to alloc manager miss configuration. The fix is to Change alloc manager system, red max entities (<code>_DPP_AM_TEMPLATE_SYS_RED_DP_PR_MAX_ENTITIES</code>) from 64 to Multiple NOF Queues by 2. We are multiplying since we have 2 pointer per Queue.
SDK-53319	733446	All 56850_A0 56850_A1 56850_A2	Fixed <code>bcm_vxlan_vpn_create</code> , <code>bcm_l2gre_vpn_create</code> API to replace UUC/MC/BC IPMC index using <code>BCM_VXLAN_VPN_REPLACE</code> , <code>BCM_L2GRE_VPN_REPLACE</code> .
SDK-53323	734007	88030_A0	Instead calculate total good packet using the register RUCA..
SDK-53376		56850_A0 56850_A1 56850_A2	An issue was reported where I3 ip6route show was displaying NEGATIVE free entry values. In this release, the way to calculate the free number and the total number of IPv6 entries has been adjusted to address this issue and also to cover the number of IPv6/64 entries. It depends on <code>bcm_switch_object_count_get</code> to get the following objects: <code>bcmSwitchObjectL3RouteV6Routes64bMax</code> <code>bcmSwitchObjectL3RouteV6Routes128bMax</code> <code>defip_64_free</code> <code>defip_128_free</code>
SDK-53380		88750_A0 88650_A0	The previous issue is that deinit can only be exercised after successfully init, it is not acceptable in case a single CPU controls multiple devices, if one device fails in init, we have to reset the CPU which affect other devices. The current fix is that we support the partial deinit which will dealloc the resource that was allocated in previous init failure, then it can do the normal init which means one device init failure don't need to reset the CPU.
SDK-53405	721824	88650_A0	Scheduling elements prints were added to the <code>gport</code> command. Additionally, "gport count" or "gport c" will print a summary of all gport types count. NOTE: <code>bcm_cosq_gport_traverse</code> was extended to include SEs of all types!!!
SDK-53433	731111	All	In TDM bypass mode, in the FDT - the IRE TDM mask mode is configured by default for not reading from RTP link-integrity, but rather using the RTP reachable bitmap. When calling <code>bcm_fabric_tdm_direct_routing_set()</code> the usage of RTP link-integrity is enabled, without the ability to go back. So static link configuration will not be ignored, and the user can configure active links for TDM bypass mode.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53444	725754	56840_A0	While rebooting the system an additional flap was seen on port during the BCM INIT. This has been resolved In phy init code, by disabling it first in case the attached phy is not an external phy and is not in warmboot procedure.
SDK-53451	735769	56640_A0 56640_A1 56640_B0	On link up, the pause and MTU max values were not retrieved from config, resulting in all the config values becoming obsolete and default values being programmed. Fix is retrieval of the data from the config and override the default values so that the port values reflect the configured parameters.
SDK-53452 SDK-52881 SDK-48849	722247	56548_A0 56546_A0 56545_A0 56544_A0 56542_A0 56541_A0 56540_A0 56545_A1 56540_B0 56541_B0 56546_B0 56544_B0 56547_A0 56545_B0 56542_B0	For Apollo2 device, the L3_DEFIP memory was partitioned into fixed size giving 2K indexes for v4/64V6 and 2K for 128V6. The change is to partition the L3_DEFIP table with the user defined values. User can give any number to change the max number of 128V6 entries and V4/64V6 entries. This user defined partition scheme is already supported for TR3/TD2. user needs to set these soc properties . ipv6_lpm_128b_enable=1 =====> This sets the new scheme active. num_ipv6_lpm_128b_entries = XXXX =====> Number of 128V6 routes Without this configuration, the SDK will set the table with default route tables. 128v6=2048 and v4/64V6=2048
SDK-53482	88650	A0	Valgrind is a tool that reports cases where code uses uninitialized data. Currently when Valgrind is run and warmboot is done, some warnings are emitted for usage of uninitialized data. The source of these warnings are uninitialized data that are written to a file. In one case, there was a problem with 64 bit systems that would read bad data -- this was fixed. In the other cases, unused and uninitialized data was always written to the warmboot file. This data written to the warmboot file is now initialized without any implication on code logic.
SDK-53487	736250	56850_A0 56850_A1 56830_A1 56850_A2 56830_A0 56830_A2	In the previous release, the restriction that the queues in strict priority mode must be in consecutive order on Trident2 family was not documented. In this release, this issue has been addressed by documenting the restriction.
SDK-53506	716783	56850_A0 56850_A1 56850_A2	In the previous release, when the first strict priority member was a unicast queue, the function bcm_td2_sched_check_constraints() returned *ucmap=1 which was not correct. In this release, this issue has been addressed by setting *ucmap=0.
SDK-53507	736772	56850_A0 56850_A1 56850_A2	This JIRA is a duplicate of SDK-53600. In which the API bcmPortControlMmuDrain is improved to check the empty state of each nodes and queues after the cells are drained.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53508	736774	56850_A0 56850_A1 56850_A2	In the previous version, in TD2, when the weights of queues were changed dynamically leaving the schedule mode unchanged, the traffic would be disrupted. In this release, the API <code>bcm_cosq_port_sched_set()</code> and <code>bcm_cosq_gport_sched_set()</code> have been improved to allow weights to be changed dynamically when the schedule mode is not changed.
SDK-53517		56850_A0 56850_A1 56850_A2	Added ability to support three ALPM profiles to provision different Pivot reservations.
SDK-53556	735811	56640_A0 56641_A0 56850_A0 56640_B0 56644_B0 56850_A1 56850_A2	Index for COS_MAP_SEL table was being incorrectly set for CPU as ingress port (0). The higig packets ended up in a wrong queue due to wrong index for CMIC ports the index is retrieved from the soc layer and programmed. Corrected the index appropriately in TD2 and TR3 devices.
SDK-53561		56846_A0 56850_A0	TD+ L2_ENTRY table is shared between the two pipelines. Only X-pipe has SBUS access to the shared L2_ENTRY table. TD2 has independent L2_ENTRY_x and L2_ENTRY_y tables, but only access type 4 is supported in memory write operation for these two tables. The combination of L2_ENTRY table and access type 2 for Y-pipe has been added into skipped memory list.
SDK-53574	737396	All	Fixed buffer overrun in fall-back implementation of <code>sal_strncpy</code> . Note that this implementation is not used in any of the primary system environments such as Linux and VxWorks.
SDK-53584		56850_A0	L2X table is read via DMA manner by default. If DMA fails, the table entries will be read via PIO manner again. If there is a parity error in L2X table, both DMA read and PIO read will trigger parity error reporting. The SER logging feature has been implemented to detect and filter the duplicate parity errors for the customer application.
SDK-53600	737427	56850_A2	The API <code>bcmPortControlMmuDrain</code> has been improved to check the empty state of each node and queue after the cells are drained.
SDK-53602		88650_A0 88650_B1 88660_A0	TDM bypass traffic whose destination is the same FAP usually does not go through the fabric. Forcing of TDM bypass traffic to the fabric can be enabled/disabled using: <code>bcm_fabric_control_set(unit, bcmFabricForceTdmBypassTrafficToFabric, 1/0);</code> The current state can be retrieved using: <code>bcm_fabric_control_get(unit, bcmFabricForceTdmBypassTrafficToFabric, &enabled);</code>
SDK-53611	737404	56634_B0	CPU can send ethernet packet and higig packet. For local switch disable feature, the register <code>ILOCAL_SW_DISABLE_DEFAULT_PBM_64</code> should be configured for CPU port when CPU is sending higig packet, but that is missed in SDK. This issue has been fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53628		88660_A0	Trill: Internal implementation was changed to remove my nickname duplication in SOC_PPC_MYMAC_TRILL_INFO. Fix does not change Trill multi-homing application
SDK-53630		88650_A0 88650_B0 88650_B1 88660_A0	Tunnel APIs replace: 1.Support bcm_tunnel_initiator_create to replace dip, vlan, dscp, ttl, sip and type of IPv4 tunnel with tunnel_id of tunnel or l3a_intf_id or l3a_tunnel_idx of intf. In case of IPv6 tunnel, dip6 sip6, ttl and type can be replaced. In case of ERSPAN tunnel on ARAD+, span_id and l3_intf_id can be replaced. 2.Support bcm_tunnel_terminator_create to replace tunnel_if, if_class and flags (BCM_TUNNEL_TERM_USE_OUTER_DSCP or BCM_TUNNEL_TERM_USE_OUTER_TTL) with tunnel_id.
SDK-53636	737820	All	Added Level 2 warmboot recovery support for the following switch controls: bcmSwitchUseGport bcmSwitchL2PortBlocking bcmSwitchCallbackAbortOnError
SDK-53656	738788	88650_A0	In Ingress Field Processor, validation of ISQ range in bcm_field_action_add() is incorrect, and as a result does not allow usage of the full range. This is fixed.
SDK-53657	737782	56846_A0	When upgrading from sdk-6.2.5 to sdk-6.3.5 scache space for the differential state was not allocated for few modules. Fixed scache reallocation for RX, NIV, VXLAN modules during warm upgrades.
SDK-53661	737925	56850_A0 56850_A1 56850_A2	TD2 TDM Oversubscription Issue, tx failure with mixed 10G/40G configuration was failing. This has been addressed by updates to the oversub group sorting algorithm.
SDK-53672	739010	All	The validation logic which is valid for SQ/MC was preventing the creation of dynamic queues when the indexes were more than 8 (0 -15). Fixed the code to support dynamic queue indexing as well. The SQ/MC index validation is done for non dynamic ports only.
SDK-53673	738994	56850_A2	Corrected the value of macro definition (BCM_IPMC_RANGE_IP6 and BCM_IPMC_RANGE_PIM_BIDIR).
SDK-53674	739094	56450_B0 56450_A0	bcm_cosq_gport_attach API was not able to allocate non-contiguous queues to support more WRR queues/nodes. A new flag BCM_COSQ_GPORT_SCHEDULER_WFQ has been introduced to support this option.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53684		88660_A0	DEFAULT BEHAVIOR CHANGE. In Field Processor, when using external TCAM, valid entry priorities values are now limited to the range of 0 to 4194303 (2^{22}). Additionally, when creating an ACL group in the external TCAM, the user may specify the max entries priorities they intend to use in this group by setting the <code>max_entry_priorities</code> parameter in <code>bcm_field_group_config_t</code> . Indicating the max entry priorities will result in improved control performance of the external TCAM driver. Note that if the <code>max_entry_priorities</code> parameter is set, the valid entry priorities values for the configured group are limited to the range of 0 to <code>max_entry_priorities</code> . Also note that <code>max_entry_priorities</code> parameter is only supported for external TCAM.
SDK-53732	732324	88650_A0	In Field Processor, the user can qualify packets according to the trap-code (<code>bcmFieldQualifyRxTrapCode</code>). The <code>bcm_field_qualify_RxTrapCode</code> expects only a <code>bcm_rx_trap_t</code> parameter, indicating which trap. Thus, it does not support User-Defined traps (since no ID can be specified). Besides, for <code>bcmRxTrapL2Cache</code> Trap-code, two possible set of traps can be qualified: 1. By default, the programmable traps are qualified. Due to their HW value (not divisible by 4), only the 2 first programmable traps are qualified. 2. If the SOC property <code>custom_feature_trap_l2_cache_field_reserve_mc_hit</code> is set, then the 8 Reserve-Multicast traps are qualified instead
SDK-53741	738835	88650_A0 88650_B0 88660_A0	In BCM886XX, the L2 traverse HW allows the definition of flexible rules to traverse and modify the MAC Table entries, including a flexible mask (both on entry key and payload). The <code>bcm_l2_match_masked_traverse</code> is implemented, and examples of L2 traverse can be found in <code>\$SDK/src/examples/dpp/cint_l2_traverse.c</code> .
SDK-53757	733995	88650_A0 88650_B0 88650_B1	Clear ipv6 tunnel using <code>bcm_tunnel_initiator_clear()</code> is now supported.
SDK-53763		88660_A0	1. Add support of enable/disable learn functionality in <code>bcm_port_learn_set</code> function for Trill port. 2. Add <code>cint</code> with learning disable for virtual RBridge that receive packet with ingress nickname equal to own virtual nickname.
SDK-53770		88650_A0 88660_A0	Advanced VLAN Edit: Added an example functions for QoS mapping configuration in <code>cint_advanced_vlan_translation_mode.c: qos_default_settings(), add_qos_mapping() and set_qos_mapping()</code> .
SDK-53776	739518	88030_B0	Support 4Gbits DDR part, allow row sharing and provide a "TmuAllocDump" to show detailed DRAM usage for bcm88030

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53794		88650_A0 88650_B0 88650_B1	Replace functionality in L3: Added support to replace intf, mac_addr, vlan, port, qos_map_id and encap_id by bcm_l3_egress_create with BCM_L3_REPLACE flags. Added support to replace intf_array by bcm_l3_egress_ecmp_create with BCM_L3_REPLACE flags.
SDK-53800	739936	All 56450_A0	When bcm_mpls_port_add() was called with new label and BCM_MPLS_PORT_REPLACE flag, a new mpls entry was being added without deleting the old mpls entry with old label. This has been fixed.
SDK-53802	740202	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	Updated TSC transcription algorithm to resolve incorrect TDM programming for partial TSC configuration
SDK-53822		56850_A1 56850_A2 56850_A0	provided new objects to get the information about a) Max 128B V6, 64B V6, V4 entries for a given configuration b)used 128B V6, 64B V6, V4 entries c) Free 128B V6, 64B V6, V4 entries
SDK-53830		88650_A0 88650_B0	When calling bcm_port_match_add with an egress match, with a valid input but a remote port, the return value should be BCM_E_NONE without any configuration. Instead return value is BCM_E_NOT_FOUND.
SDK-53875	737326	56850_A0 56850_A1 56850_A2	There was no BCM API to control the L2 learning per VXLAN logical port before. Now bcm_port_learn_get/set can be re-used to get/set the L2 learning per VXLAN logical port.
SDK-53876	740022	56850_A0	bcmFieldQualifyMhOpcode and bcmFieldQualifySourceVirtualPortValid qualifiers offsets are updated for Ingress Field Processor to match with regfile (56850). Problem : SDK was unable to use bcmFieldQualifyMhOpcode and bcmFieldQualifySourceVirtualPortValid qualifiers in the Key format - FPF3 in Ingress Field Processor. Solution : bcmFieldQualifyMhOpcode and bcmFieldQualifySourceVirtualPortValid qualifiers offsets are updated for Ingress Field Processor to match with regfile (56850). This is done in Initialization routine of Ingress Field Processor for the Key Format - FPF3 .

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53885	740483	56450_A0	Fixed the issue where entry in EGR_MPLS_VC_SWAP_LABEL_TABLE is replaced when a different MPLS port uses the same VC label but having different properties. With this fix a new entry will be created in the above mentioned table instead of replacing the existing entry.
SDK-53890		88650_A0	Fix of building errors occurred when the Makefile includes KBP flags : + FEATURE_LIST := KBP + KBP_DEVICE := KBP_ALG and missing WB flags: - CFGFLAGS += - DBCM_WARM_BOOT_SUPPORT - CFGFLAGS += - DBCM_WARM_BOOT_SUPPORT_SW_DUMP - CFGFLAGS += - DBCM_EASY_RELOAD_WB_COMPAT_SUPPORT
SDK-53894		56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	Support added for software assisted virtual port L2 flush. In addition, there was an issue where the software copy of the L2 entries learnt on external ESM was out of sync with the hardware,. This is now handled by correctly extracting and updating external L2 entries into internal software copy.
SDK-53907	740307	88650_A0	Fixed alloc manager failure in case PWE is created over LAG. The scenario that caused failure: 1. Tunnel application setup using l3_egress_object on a LAG port. 2. Create a PWE on the tunnel interface
SDK-53912	739785	56850_A0 56340_A0	Added ability to support ETAG(Port Extender VLAN Tag) tunneled mirror.
SDK-53919	740350	All	STG Id - 0 is reserved and used for internal purposes only and VLANs should not be added to this group. Updated the users guide with the above information.
SDK-53926	740455	88650_A0	In Metering HW, rate configuration is performed via a mantissa and exponent representation. An internal SW function is used to convert from a rate to the mantissa and exponent representation, and takes as input the maximal allowed exponent and mantissa. Due to a bug, in some cases this function could return an out-of-bounds value (bigger than the maximal mantissa), which causes an assertion failure when the value is written to the HW. This is now fixed.
SDK-53934		56850_A0 56850_A1 56850_A2	In previous releases, if we configured L3Depth larger than 0, it was not possible that single-wide or double-wide entries could be moved to other banks to free its original space for wider entries like double-wide or quad-wide entry. In this release, single-wide or double-wide entries can be moved to other banks to free its original space for wider entry, and the total utilization of L3 benefits from this.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53935		56850_A0	In earlier releases, <code>bcm_l2_matched_traverse</code> API call did not have a way to retrieve Static only entries. Code has been added to deal with the action for the STATIC only entry. When the action is for the STATIC only entry, we now set <code>STATIC_BIT</code> in both data and mask fields.
SDK-53940		56850_A2	There is one issue for eye scan extrapolation that yields 1e-0.0 BER. The reason is that the sample points used in the extrapolation are less than 2 and the eye scan function rejects its calculation. Also the sample points are not screened, but to include all nodes, to fit the extrapolation equations. This JIRA is to fix these issues by picking or creating the proper sample points for extrapolation. For the PRBS error count, the read back from the PRBS status register is 2X for some counting modes, but the calculation equation is expected to be 1X in eye scan, while 2X in the eye margin. This JIRA irons out the difference by changing the eye margin to expect 1X as well.
SDK-53946		88650_B1 88660_A0	Important note: in Fiber channel APIs, due to an API change, the user must replace <code>bcm_fcoe_zone_entry_t</code> to <code>vsan.vsan</code> by <code>bcm_fcoe_zone_entry_t</code> to <code>vsan_id</code> , e.g. in <code>bcm_fcoe_zone_add</code> API.
SDK-53952	741900	56450_A0	Resource leakage issue in <code>EGR_MPLS_VC_AND_SWAP_LABEL_TABLE</code> caused by <code>bcm_mpls_port_add()</code> API is fixed.
SDK-53955	740686	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	In some configurations, ISM memory buckets greater than what is desired was being allocated. This has been addressed by correctly allocating just what is needed, by checking for boundary conditions.
SDK-53956		88650_A0 88650_B0 88660_A0	Egress compensation can be configured for egress ports using the API <code>bcm_cosq_control_set</code> (<code>bcmCosqControlPacketLengthAdjust</code>). When the compensation is configured for port with header type <code>XGS_DiffServ</code> , <code>XGS_HQoSan</code> error will occur. Fixed.
SDK-53960		88650_A0	when running on little Endian CPU (gto is big Endian) some field BCM APIs may fail, for example following : create pre-selector egress PMF entry with qualifier <code>bcmFieldForwardingTypep4Ucast</code> . The fix was in the internal function <code>"shr_bitop_range_copy"</code>
SDK-53961		88650_B0	5.75G support is added for ILKN mode
SDK-53963	741711	56850_A2	Fixed VXLAN/L2GRE tunnel initiator's udp port update functionality.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-53968	740158	56850_A0	Fixed to validate VXLAN and L2GRE VPN during <code>port_delete</code> and <code>port_get</code> API.
SDK-53972		88650_A0	Petra-B-ARAD system: initialize values correctly for system-headers under Petra-B ARAD system
SDK-53992		56640_A0 56850_A0 56640_A1 56640_B0 56850_A1 56850_A2	When a new V6 prefix group is being created and if the start index falls in paired TCAM the following is being done. 1. Check if the previous prefix group has free entries in unpaired TCAM. If yes, set the start of the new prefix group to be that index. 2. if the previous group doesn't have any free entries in unpaired tcam, try to move entries up. if entries can be moved up, then set start as start - 1 of the next prefix group.
SDK-53993	742520	56450_A0	The <code>bcm_port_match_add()</code> API was writing the data into wrong entry in <code>vlan_xlate</code> table because the search key did not include the field <code>source_type=1 (sglp)</code> . As a result it was not matching the existing entry. Modified <code>bcm_mpls_port_match_add()</code> API to include the <code>SOURCE_TYPE</code> field as part of key for adding entry in <code>VLAN_XLATE</code> table.
SDK-53994	741664	88650_B0 88650_B1	L3: TTL scope entries were not freed when the RIF is deleted.
SDK-53996	741161	56640_A0	Fixed the problem of wrong tunnel index generation in SDK. The tunnel index was wrongly multiplied by 4 before being written into hardware. it made the tunnel index space to use only quarter entries in hardware. Rest of the tunnel indexes values were out of the limit to be written into hardware and resulted in error. Due to this reason, the capwap tunnels were able to scale only to quarter of full hardware space.
SDK-53998	737239	56800_A0 56334_A0	In previous SDK, the statistics <code>snmplfOutDiscards</code> returned wrong value on some old devices. This snmp counter was mapped to <code>MMU_CTR_MC_DROP_MEMm</code> , but this memory was not existed in some old devices, so the counter should be mapped to some other memories. This issue has now been resolved.
SDK-54001		All	Show KNET protocol override option in CLI help. Added proper support for protocol override in <code>bcm_knet_filter_get</code> API.
SDK-54004		56640_B0	Added the support code for the new SKU BCM56044.
SDK-54009	739826	88650_A0 88650_B0 88650_B1 88660_A0	In L2, during access of static entries (get or delete operations), the parsing of the MAC entry age field was incorrect, causing an internal function failure. The parsing error is fixed, including the removal of an unneeded HW access.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54014		88660_A0	In BCM88660, the user can select packets which tries to transplant a static MAC Table entry during learning. Specifically, when a statically inserted MACT entry is matched in the learning lookup, but the Source-Port is mismatched, the entry is not modified. If the user wants to match in Ingress Field Processor such packets, The fix includes: 1. Setting Out-LIF valid bit when inserting a static L2 entry with no OutLIF and no valid EEI. 2. Modifying the is-dynamic-entry indication to fix the transplant indication that arrives to the FP. 3. Running <code>cint_field_drop_static_sa_transplant.c</code> (new CINT example) to drop such packets
SDK-54015		88650_A0 88650_B0 88650_B1 88660_A0	In the policer module, when calling the <code>bcm_policer_create</code> and <code>bcm_policer_set</code> functions, the <code>BCM_POLICER_REPLACE</code> can be used to replace the configuration of a meter, or the template that the meter points to. A cint example has been added that shows how to use the <code>BCM_POLICER_REPLACE</code> flag to change the configuration of a meter. For details see the function <code>metering_replace_example</code> in <code>cint_policer_metering_example.c</code> .
SDK-54019		88650_B1 88660_A0	Provide a CINT example for followed improvement. In case of L2GRE and VXLAN bud node, MC packet received from overlay recycle port couldn't be decapsulated correctly at previous release. Now the issue was fixed based on new soc property <code>default_logical_interface_ip_tunnel_overlay_mc</code> .
SDK-54021		88650_A0 88650_B0	Bug was fixed in event handling of events with high RMEP indices for Arad A0/B0.
SDK-54031	743203	88650_A0 88650_B0 88660_A0	Configuration of the Credit Discount of ISQs is now possible using: <code>bcm_cosq_control_set(unit, isq_gport, cosq, bcmCosqControlPacketLengthAdjust, header_size);</code>
SDK-54034	743244	56850_A0 56850_A1 56850_A2	Added <code>bcmFieldQualifySrcNivGport, bcmFieldQualifyDstNivGport, bcmFieldQualifyDstGport</code> Qualifiers. In this JIRA, these new qualifiers are initialized only for TD2. Updated <code>bcmFieldQualifySrcGport</code> Qualifier in TD2 to support Niv source GPORT.
SDK-54035		56850_A2	This JIRA is to fix the port status (ps) command about the speed reporting problem for 11G forced speed modes for the TSC driver.
SDK-54037	739743	All	On certain devices which do not support the blocking of <code>KNOWN_MCAST</code> type of traffic a fix has been added to no longer return error. This issue was originally reported on Raven

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54038		88650_A0	In Field diagnostics (mode 3) entries validation is performed for all banks. This causes a segmentation fault because it should only be for banks that their owner is PMF, since the entry management for other owners is not performed by PMF. Fixed.
SDK-54042		88650_B1	In BCM L3 file, the macro <code>DPP_VRF_VALID(_vrf)</code> definition included a limitation that was not correct for Arad devices. This caused an error when trying to create a L3 interface with <code>VRF>255</code> . The macro definition is changed to support the Arad's limitation.
SDK-54053	743221	56640_A0 56640_B0	On parity error in MMU counters the hardware was not clearing the entries. As a fix, when the parity error happens we now clear parity status and then clear the entry.
SDK-54055		88650_A0	Trill:bcm_trill_multicast_entry_get is now supported.
SDK-54058 SDK-53881 SDK-53879 SDK-53878 SDK-53880		88650_A0	<p>The Drop-Precedence (DP) is a value that represents QoS internally. The DP is a 2 bit value that represents the colors green (0), yellow (1-2), and red (3).</p> <p>Today, when a packet was received in the device, and its DP was resolved to 2 (or yellow), the device would change it to 1 (also yellow) when sending the packet. This is a result of an attempt to always represent yellow as 1. However in some cases -- for instance when a packet is sent to the device and receives a DP of 2, and this DP should remain 2 -- this will cause problems for the user.</p> <p>This is now fixed by keeping the DP at the same value instead of changing it to 1.</p>
SDK-54063	743248	56643_A0	<code>XMAC_OSTS_TIMESTAMP_ADJUST</code> accounts for delays during the mac stage. This register was always being programmed to zero. Now, This register will be configured with proper value for non GE ports, and for GE ports this will be configured to zero. The value of this register doesnot matter for GMII/MII speeds
SDK-54064	743921	88650_A0 88650_B0 88650_B1	Resolved schan time out when setting pfc refresh timer. The error is caused by reading non-existent register.
SDK-54067		All	Converted MAID value to network byte order before writing to HW table in order to avoid CCM convergence issues arising due to host processor endianness.
SDK-54072	744057	56850_A0	Updated <code>bcmFieldQualifyDstNivGport</code> , <code>bcmFieldQualifyDs</code> <code>tGport</code> Qualifiers to support Niv GPORT at EFP in TD2
SDK-54075		88650_A0 88650_B0 88650_B1 88660_A0	VLAN-compression: Delete correctly global VLAN range, in case of no ports refer to it.
SDK-54083	735871	88650_A0 88650_B0 88650_B1	<code>bcm_l2_addr_add()</code> returns error when trunk tgid is used and more than 256. This issue was due to wrong define max value. We fixed the define value.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54087	743745	56850_A0	In the previous release <code>bcm_mirror_port_dest_add</code> failed with -18 on NIV ports. This issue has been fixed.
SDK-54088		56850_A0	In earlier releases <code>bcm_l3_init()</code> should clear <code>rh_ecmp_flowset</code> but this function did not work. This has been resolved.
SDK-54092		88650_A0 88660_A0	During Driver initialization, all the meter were initialized to use Meter-profile 0. This was unnecessary and removed, since the HW table was already initialized to zero.
SDK-54093	743673	88650_A0	Ingress packet size limit is set to (16KB-128bytes) if the DRAM buffer size is 512 bytes or higher, and set to 8KB if the DRAM buffer size is 256 bytes.
SDK-54100		88650_A0	Different threads (Counter Processor, Rx LOS) were failing due to forbidden access to the device during Warm-boot procedure. In counter processor case exceptional access is allowed. For RX LOS initialization postponed till after the Warm Boot finished
SDK-54114		88650_A0	In Field Processor, the Diagnostics command "BCM> diag field res" displays information on databases and banks. The number of entries per bank and database displayed by the diagnostic was incorrect when the entry (bank 0, line 0) was occupied by this Database. Fixed.
SDK-54115	743990	88650_B1	Template management: Resolved issue in a template allocation mechanism that caused on some cases a crash in <code>bcm.user</code> when resource fails to be allocated. This could happen when asking for more profiles than device capable for example: asking more LLVP profile in <code>bcm_port_tpid_class_set</code> .
SDK-54117		88650_A0	<code>port_enable_set</code> API changed: The API is no longer stop EGQ or disable NBI FIFOs. Instead it drops the traffic in the NIF.
SDK-54131	744562	88650_A0 88650_B0 88650_B1 88660_A0	CINT: Adjust <code>vswitch_p2p_init</code> , <code>qos_map_gport</code> to suit the correct number of parameters when called from <code>cint_qos_l2.c</code> , <code>qos_map_l2_run_with_untagged</code> respectively.
SDK-54148		88650_A0 88640_A0	In BCM L3 file, some errors were returned with a generic "TODO err message" text. All error messages in <code>l3.c</code> file are now meaningful.
SDK-54154		88650_A0	In Field Processor, internal and external TCAM tables shared a limitation for the number of uninstalled entries. This limitation is now separated for internal and external TCAMs, in order to allow better control of limitations and memory allocations.
SDK-54162	744768	All	Fixed VXLAN/L2GRE issue with <code>bcm_vxlan_tunnel_initiator_destroy</code> API associated with same-SIP, Multi-DIP scenario.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54168	738971	56850_A0 56850_A1 56850_A2	In previous release, parity error occurred at second half memory of <code>ING_L3_NEXT_HOP</code> could not be corrected. This has been resolved by correcting the memory depth of <code>ING_L3_NEXT_HOP</code> in SER.
SDK-54171		88650_A0 88660_A0	Move trill deprecated tests from 88640 devices to a deprecated folder.
SDK-54174	744799	88650_A0 88640_A0 88650_B0 88650_B1 88660_A0	VLAN: Remove an incorrect "entry not found" error when calling <code>bcm_vlan_port_create()</code> .
SDK-54183	745284	88650_A0 88660_A0	VLAN: Calling <code>bcm_vlan_port_create</code> with the <code>BCM_VLAN_PORT_CREATE_WITH_ID</code> flag, prompts an API check whether the requested ID is available. There was a bug where this check for valid ID would always check if a FEC ID is free, even though the VLAN port might be a multicast ID or a LIF ID. This bug was fixed, and now every VLAN port created <code>WITH_ID</code> would check the appropriate resource is free.
SDK-54185		All	knetctrl filter show did not display the <code>dest_proto</code> field. Show overridden protocol type when showing knet filter information in bcm shell. This has been addressed.
SDK-54186	743815	56850_A0 56850_A1 56850_A2	Added SDK Support of ETHERTYPE key in FPF1 Mode 6 in Trident2 Chipset.
SDK-54191		88650_A0 88650_B0 88660_A0	Changed replace logic in <code>bcm_l3_intf_create</code> . After change, when creating I3 intf for the first time, REPLACE flag should not be added. when creating existing I3 intf, REPLACE flag should be added.
SDK-54192	738575	88660_A0	DSCP/EXP marking when bridging allows the user to change the DSCP value of the IP header or the EXP value of the MPLS header of a packet, even when the packet is only bridged. To perform DSCP/EXP marking during bridging, the device is configured to set DSCP and EXP according to the assigned TC, DP, QoS profile and InLIF profile of a packet (map). Due to a SW bug, when configuring a map with TC 4 .. 7, nothing would be configured. As a result packets that have TC 4..7 assigned to them and that DSCP/EXP should be performed on them will get invalid DSCP and EXP values. This fix resolves this issue.
SDK-54194	745534	56850_A0	An SDK crash issue was reported when trying to call <code>bcm_l2_addr_replace()</code> with > 8K MAC address configured on various of vxlan tunnels. This was resolved by correcting the memory allocation. The system now allocates memory for l2 freezing according to the actual size of <code>SOURCE_VPM</code> instead of 8192.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54195		88660_A0	MPLS tunnel works in two modes: Uniform and Pipe. For Pipe mode, struct <code>bcm_mpls_egress_label_t</code> has two flags: <code>BCM_MPLS_EGRESS_LABEL_EXP_SET</code> , <code>BCM_MPLS_EGRESS_LABEL_EXP_COPY</code> , to distinguish between different MPLS pipe modes. In ARAD Pipe mode supports only <code>BCM_MPLS_EGRESS_LABEL_EXP_SET</code> flag. ARAD PLUS supports global configuration of these settings, which is set using switch control <code>bcm_switch_control_set(unit, bcmSwitchMplsPipeTunnelLabelExpSet, 1)</code> ; Default of the behavior is <code>EXP_COPY</code> . The flags should be set in consistency with the global configuration. If <code>BCM_MPLS_EGRESS_LABEL_EXP_SET</code> flag is set but <code>bcmSwitchMplsPipeTunnelLabelExpSet</code> switch control is not called, an error will be generated. The same with copy - If <code>BCM_MPLS_EGRESS_LABEL_EXP_COPY</code> flag is set but <code>bcmSwitchMplsPipeTunnelLabelExpSet</code> switch control is called, an error will be generated. See an example of use in: <code>cint_mpls_lsr.c</code> <code>mpls_pipe_mode_exp_set</code> function
SDK-54202	741184	56240_B0	MMU_INTR_MASK bits for CI0, CI1 and CI2 remain reset (set to 1) based on available memory banks. If number of external banks available is 0 then all for all CI0,1,2 the mask will be set. If it is 1 then the mask will be set for CI1 and CI2. if it is 2 then mask will be set only for CI2 and if it is 3 then mask will not be set for any of CI0-CI2.
SDK-54203		56440_A0 56440_B0	Parity checks will now be turned off for non-existent external DDR memory banks.
SDK-54205	738767	56850_A0 56850_A1 56850_A2	It was reported that small packets will be dropped if <code>ENQ_ASF_HS_OVERSUB_EN</code> hasn't been set for the 40G ports which are in oversubscription cut through mode. The issue has been resolved as below: Add 40G/30G ports with oversub to <code>ENQ_ASF_HS_OVERSUB_EN</code> during init. As no matter the ports are enabled CT or not, these ports can always be in <code>ENQ_ASF_HS_OVERSUB_EN</code> .
SDK-54209 SDK-61547	744936	56340_A0 56342_A0	Issue :- Segmentation Fault was observed when more than 254 Flex Counters were created in VFP region in Helix4. Fix :- Maximum number of Flex counters per pool were wrongly assigned during init. Updated the code with correct values.
SDK-54211	682994	88650_A0 88650_B0 88660_A0	A bug in <code>bcm_oam_action_set()</code> causing certain OAM frames to be erroneously prepended with an additional set of system headers in certain situations was fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54212	690179	88650_B0	The following bug was fixed: OAM endpoints deletion is not releasing internal allocations in case of insertion of a MEP, calling <code>bcm_oam_endpoint_action_set</code> and then deleting the endpoint. After performing this sequence several times a failure will be returned.
SDK-54213	736267	88650_A0 88650_B0 88650_B1 88660_A0	Add received oversized frame counter (ROVR) value to <code>snmplfInErrors</code> counter (<code>bcm_stat_get</code> API).
SDK-54215		88650_A0 88660_A0	Added documentation for a traffic example and additional documentation per function in <code>cint_qos.c</code>
SDK-54220	745537	88750_B0	When the CL72 mode is enabled, snake test with external loopback failed on fe1600, fixed.
SDK-54230	741970	88650_B0 88660_A0	1588 Termination: Added support to following 1588 termination classification (in addition to already supported 1588oE, 1588oUDPoIPoE): 1. 1588oUDPoIPoIPoE 2. 1588oUDPoIPoMPLSoE 3. 1588oEoMPLSoE Packets will be identified as 1588 packets regardless the forwarding header: Switching (Ethernet forwarding) Routing (IPv4/ MPLS forwarding) or Tunnel (IP/MPLS) termination.
SDK-54233		88650_A0 88650ACP_A0 88650_B0 88650_B1	Change diag pp DB_LIF_lkup_info diagnostic to print lif information instead of the rif information it was displaying previously.
SDK-54235		88650_A0 88660_A0	Egress VLAN Edit: EVE Operations are processed per packet after an ESEM lookup that yields an Out-LIF with a value up to 64K. ESEM entries for Out-LIFs with value above 32K, produced an incorrect Out-LIF value that in turn processed an incorrect EVE action. The fix enables correct EVE behavior for Out-LIFs above 32K as well.
SDK-54236		88650_B0 88660_A0	In external Tcam, in the application file kbp.c, compilation warnings may appear due to a wrong return value variable type. The variable type is fixed.
SDK-54246	733382	88650_A0	When calling <code>bcm_oam_init(0)</code> , <code>counter_engine_source_0</code> was used for <code>INGRESS_OAM</code> and <code>counter_engine_source_1</code> was used for <code>EGRESS_OAM</code> , regardless of the soc property configurations. After the fix, any one of the 4 <code>counter_engine_source_Ns</code> may be used for egress/ingress oam, however if OAM is used, at least on counter engine must be set to <code>EGRESS_OAM</code> and at least one must be set to <code>INGRESS_OAM</code> .
SDK-54253	746153	All	Implemented <code>bcm_field_qualify_data_get</code> API for all devices supporting User Defined Function in Field module. This helps to display qualifier data fed into User Defined Function during Field entry creation.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54262		56850_A0 56850_A1 56850_A2	Using the API <code>bcm_cosq_stat_sync_get()</code> to retrieve statistics resulted in incorrect values for counters that are wider than 32bits. The reason being only the initial 32bits were being retrieved. Now the width of the counter is fetched before retrieving the counter value.
SDK-54263	744368	88650_A0 88650ACP_A0 88650_B0 88650_B1	When replacing PWE using <code>bcm_mpls_port_add_api</code> with a new push profile, the old push profile was not freed which might cause resources leak. This issue is fixed. Note that replacing push profile is supported only in case PWE is protected.
SDK-54264		88650_B0 88660_A0	Required changes in SDK in order to support KBP-SDK 1.1.1 for external TCAM.
SDK-54266		88650_A0 88660_A0	Fixed a bug in the bcm shell diagnostic function "diag pp cc". The vlan port information displayed in the diagnostic was missing some fields. These fields will now be displayed correctly.
SDK-54269	746371	88660_A0	OAM: Incorrect CCM interval was previously used.
SDK-54271		88650_A0 88660_A0 88670_A0	VPLS: Up to now, calling <code>bcm_mpls_port_create</code> would always allocate both InLif and OutLif. Current enhancement allocates OutLif resource only in case it's required by HW. In other words, in case of PWE unprotected P2P, OutLif is not allocated and can be used for other applications.
SDK-54279	738771	56850_A0 56850_A1 56850_A2	Some PHYs always set the bit <code>XLMAC_RX_LSS_STATUS.LOCAL_FAULT_STATUS</code> no matter the actual speed the port is running at. This leads to always displaying local faults with the CLI command "port xe", which would confuse the customers when the port is running at speeds less than 10G. Now the local faults will be displayed only if the bit <code>XLMAC_RX_LSS_CTRL.LOCAL_FAULT_DISABLE</code> is clear.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54307		56850_A0	<p>This query contained two questions related to the setting for <code>num_ipv6_lpm_128b_entries</code> in ALPM mode.</p> <p>The first question was about setting <code>num_ipv6_lpm_128b_entries</code> to 3072 with URPF enabled. This question was answered by Zheng Wang, and it looks like we do not support this configuration.</p> <p>The second question was about confirming the table sizes for I3 routes when varying the settings <code>ipv6_lpm_128b_enable</code>, <code>l3_alpm_enable</code>, and URPF. It looks like the table was mostly right with a small modification in IPv6 64-bit mode (<code>ipv6_lpm_128b_enable=0</code>).</p> <p>2. Disable IPv6-128(config add <code>ipv6_lpm_128b_enable=0</code>)</p> <p>2-1 Combined mode(config add <code>l3_alpm_enable=2</code>) IPv4-32(non-URPF)/(URPF) IPv6-64(non-URPF)/(URPF) 128K/64K 85K/21K</p> <p>2-2 Parallel mode(config add <code>l3_alpm_enable=1</code>) IPv4-32(non-URPF)/(URPF) IPv6-64(non-URPF)/(URPF) 64K/16K 21K/5K</p>
SDK-54309		88650_A0	KBP compilation fix for not GTO processors
SDK-54314		88660_A0	Add diag counter graphical representation for - <code>EQO_RQP_DISCARD_SOP_COUNTER</code>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54322		88650_A0	<p>1. In Ingress parser, the support of a single IPv6 extension header parsing is added, where only Hop-by-Hop extension is supported. 2. In Ingress Parser, the custom macro allocation is now dynamic. There are four configurable macros (aka custom macros), that are programmed to identify a header. These custom macros are dynamically allocated according to the enabled features (enabled by soc properties). The following soc properties determine the custom macros in the parser: -</p> <ul style="list-style-type: none"> -bcm886xx_ipv6_ext_hdr_enable - new soc property that enables IPv6 header extension parsing, requires two custom macros. -bcm886xx_fcoe_switch_mode - enables FCoE, requires two custom macros. -custom_feature_udp_parse_disable - UDP custom macro is configured by default, however, if needed it can be disabled by this soc property. Note that if disabling UDP parsing, then VxLAN and 1588oUDP are affected. -trill_mode - enables Trill, requires one custom macro. -bcm886xx_vxlan_enable - enables VxLAN, requires one custom macro. <p>In the specific case of UDPoIPv4oEth, enabling or disabling VxLAN changes the value of parser object end-leaf, which is used in Trap in case there is an error in the Header size or in case of invalid packet format code. 3. In Ingress Field Group, a new qualifier bcmFieldQualifyExtensionHeaderType is introduced, which refers to the Next Header field in first IPv6 extension header after IPv6 header.</p>
SDK-54323		88650_A0	<p>Due to inefficient internal implementation, the allocation manager was taking a lot of time during the warmboot recovery. The implementation was changed to reduce the number of function called during restoration and to accelerate the warm reboot.</p>
SDK-54328	743038	All	<p>When configuring an OAM endpoint, L3_LOCK was not being released when an endpoint was in multiple maintenance domain levels. This was corrected.</p>
SDK-54329	735713	88750_A0 88650_A0	<p>Due to miss-configuration some corrupted cells not dropped as expected. Fixed.</p>
SDK-54343		88650_A0	<p>11.25G ILKN speed support is added</p>
SDK-54344		88650_A0	<p>Device bring up fail when more than 191 ports are defined. fixed.</p>
SDK-54346	746652	56850_A0 56850_A1	<p>bcm_l3_cleanup was causing ASSERT error with L3 Egress Mode enabled, nh_index -1 could be used for bcm_xgs3_nh_del as array index and eventually could cause array bounds write and break the defensive area of allocated memory. Added nh_index parameter check to avoid invalid access.</p>
SDK-54347	738808	All	<p>bcmPortControlFabricSourceKnockout was not documented in BCM SDK manual. Added documentation for bcmPortControlFabricSourceKnockout.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54352	743979	56850_A0 56850_A1 56850_A2	In previous SDK, the COS_MAP_SEL table on TD2 sometimes was constructed incorrectly by bcm_cosq_gport_mapping_set API. The root cause of this problem was that the SDK would use ing_port to generate a index of the COS_MAP_SEL table, but this index was overwritten incorrectly and caused the problem. This issue has been resolved.
SDK-54357		56850_A2	The TX squelch function will be persistent through phy enable (on) function, so mac_loopback with port disabled could be operable with the helps of the above functions.
SDK-54363		88660_A0	PON: Trap packets that are send to PON port 12Bytes were added by egress-editor. 12Bytes padding is now removed.
SDK-54369	747308	56850_A2	Fixed VXLAN/L2GRE Tunnel Terminator State modification during multicast_port_create
SDK-54378		88650_A0	To debug more easily warmboot issues, a SW state dump is available via BCM>diag ssdump The SW state dump output to screen can now be disabled.
SDK-54385	747110	88650_B1	In the HW implementation of the Exact Match (EM) tables, a defrag machine can be enabled for all the EM tables. This machine was enabled only for Large-EM, and it is now enabled by default for all the EM tables.
SDK-54395		88650_A0 88660_A0	Support binding 32 LIF cos profiles to InLIF in case of local switching enabled.
SDK-54398	746146	56854_B0 56850_A2	In previous releases, when one interrupt was raised rather than CHIP Function's, only it would beprocessed and the CHIP Function's was lost. In this release they will be processed one by one through comparing all the irqState with irqMask.
SDK-54400	746935	56850_A0	Fixed EGR_PORT_TO_NHI_MAPPING during multicast egress object destroy
SDK-54414	739326	56640_B0	following phy diag command is created to be able to poke into core0,1,3 and MLD register for100G plus port. And the format is phy diag pbm reg core0(core1, core2, mld) aer reg_addr (for read) phy diag pbm reg core0(core1, core2, mld) aer reg_addr write_value (for write)
SDK-54420	746955	All 56850_A0	Only physical gport type supported in function bcm_l2_addr_delete_by_vlan_gport_multi for specific usage, added support for trunk gport type accordingly.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54423		88650_A0	<p>Vlan Translation: a new feature is added to support configurable VLAN translation for IP packets according to 5-tuples (DIP, SIP, IP-next protocol, TCP/UDP src port, TCP/UDP dst port). In SW, the sequence to enable the feature is as follows: 1. Set SOC property: <code>vlan_translation_match_ipv4</code>. 2. Set VT port profile via <code>bcm_vlan_control_port_set</code> API using <code>bcmVlanPortPreferIP4</code> attribute. 3. Create VSI and add ports to VSI (create InLif). 4. Create Field Group using <code>bcm_field_group_create()</code> set QSET with <code>bcmFieldQualifyStageIngressVlanTranslation</code> and all 5-tuples qualifiers. 5. Configure ASET with <code>bcmFieldActionIngressGportSet</code> action, and call <code>bcm_field_group_action_set()</code>. 6. Add entries to created field group.</p> <p>This feature cannot coexist with EVB support</p> <p>A new CINT is added for example: <code>cint_field_flexible_qinq_example.c</code></p>
SDK-54424		56850_A0	Added new shell module - cosq. currently supported sub modules are compensation get/set for ingress/egress
SDK-54426		88650_A0 88660_A0	BFD doesn't work properly on management system (one CPU that controls more than one device).
SDK-54429		All	Added new API <code>bcm_stat_clear_single()</code> to clear a single port stat
SDK-54435		88660_A0	Important note: SOC property <code>ipmc_vpn_lookup</code> was misused in code. Default value of soc property was set to 1 but the actual SW implementation is default value 0. Default value of <code>ipmc_vpn_lookup</code> changed from 1 to 0 to match SW implementation.
SDK-54436		88660_A0	Support to enable global IPMC function when <code>ipmc_vpn_lookup_enable=0</code> , and IPV4 compatible MC packets forwarding is according to <RIF,G,SIP> regardless the VRF value.
SDK-54438		88650_A0	<p>Added diag for header size difference on ingress and egress.</p> <p>usage: <code>cosq comp ing voq=<id></code> - show ingress compensation <code>cosq comp egr port=<id></code> - show egress compensation <code>cosq comp ing [voq=<id>] Compensation=<value></code> - set ingress compensation <code>cosq comp egr [port=<id>] Compensation=<value></code> - set egress compensation</p> <p>if only compensation value is give (without port or voq), then all ports/voqs are set with the given compensation value.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54441		88650_B0 88660_A0	OAM ARAD+ RDI can not be set by user. It is updated automatically according to: 1. Scanner LOC discovery on RMEP with same index as the MEP. 2. Received packet information. The mode can be set in <code>bcm_oam_endpoint_create</code> using the following flags: <code>BCM_OAM_ENDPOINT2_RDI_FROM_RX_DISABLE</code> , <code>BCM_OAM_ENDPOINT2_RDI_FROM_LOC_DISABLE</code>
SDK-54442 SDK-54129 SDK-54128		88650_A0 88660_A0	The meter feature has two possible modes of operation - 32K or 64K. In 32K mode, each packet has up to 2 meters with an ID spanning from 0 to 32K-1. In 64K mode, each packet has just 1 meter, with an ID spanning from 0 to 64K-1. The default Meter-ID is 0: - In 64K mode, Meter-ID 0 is set as an invalid pointer. Thus, the meter processor does not perform metering on a packet if its Meter-ID has not been modified. - In 32K mode, Meter-ID 0 was not set as invalid. Since meter 0 is defined to allow the maximal rate, there was no issue with traffic loss. However, it was affecting the color (drop precedence) given to the packet at egress. E.g., if a packet was yellow or red, its color could change to green, ignoring the incoming color, even if a valid Meter-ID was not set to this packet. Meter-ID 0 is now invalid also in 32K mode. Thus, a packet with default Meter-ID will not have its color changed by metering. Additionally, for backward compatibility sake, a SOC property is available to configure the device to set meter pointer 0 as valid: set the SOC property <code>custom_feature_meter_pointer_0_enable</code> to 1.
SDK-54460		56850_A0	In earlier releases, Embedded NH's MAC and Port information was absent in I3 table traverse. This has been resolved.
SDK-54484	745674	56850_A0 56850_A1 56850_A2	<code>BCM_L2_REPLACE_MATCH_UC</code> and <code>BCM_L2_REPLACE_MATCH_MC</code> are provided for specifying which type of MAC entries will be performed the delete operation. Using the <code>BCM_L2_REPLACE_DELETE</code> flag and <code>BCM_L2_REPLACE_MATCH_MC</code> or <code>BCM_L2_REPLACE_MATCH_UC</code> or both to delete all Unicast entries, Multicast entries or both respectively. Using the <code>BCM_L2_REPLACE_DELETE</code> without either <code>BCM_L2_REPLACE_MATCH_MC</code> nor <code>BCM_L2_REPLACE_MATCH_UC</code> is the same as both are set.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54500		88660_A0	In FCoE module, the NPV switch support is added. To configure it: 1. Set the relevant ports to be N_Port by using the <code>bcm_port_control_set</code> API with type <code>bcmPortControlFcoeNetworkPort</code> . 2. Add new routes for source routing by setting in <code>bcm_fcoe_route_add</code> API the flags to <code>(BCM_FCOE_SOURCE_ROUTE BCM_FCOE_HOST_ROUTE)</code> . Refer to <code>cint_fcoe_route.c</code> (<code>fcoe_fcf_npv_example</code> function) for configuration example. When setting the NPV functionality, 2 new FLP programs are required.
SDK-54501 SDK-51080		88650_A0 88660_A0	The template management is an internal module managing the profiles according to their attributes. A diagnostic has been added to display: 1. The profile-ID range per template IDs 2. How many objects are pointing to each profile 3. The raw content of each profile
SDK-54505		88650_B1	OAM packets of all opcodes trapped to the CPU at the egress (up-MEPs) will include the OAM-ID on the FHEI. In 6.3, to attain this behavior the soc property <code>custom_feature_oam_upmep_oam_id_on_fhei</code> should be set to 1.
SDK-54509		88650_A0	In general, SW state must be handled per unit, since multiple device SDK can run on the same CPU. Multiple global SW states have been found not to be defined per unit. Fixed.
SDK-54511		56850_A0	Changed the error type to Parameter error (<code>BCM_E_PARAM</code>) from <code>BCM_E_UNAVAIL</code> for the invalid relative offset input parameter in <code>bcm_field_data_qualifier_ethertype_add()</code> API.
SDK-54515		88660_A0	DEFAULT BEHAVIOR CHANGE (ARAD+ only). Out AC: Out ACs can be created in pairs by calling <code>bcm_vlan_port_create</code> with a <code>BCM_VLAN_PORT_WITH_ID</code> flag and pairs of <code>vlan_port_id</code> . A problem occurs when creating a pair of Out ACs (15 MSBs) with the odd entry created first. When the even entry is created second, the odd entry gets corrupted. The issue detailed above affects the Out AC creation. Pairs of Out AC can be created correctly after the fix in any order. The fix include changing by default all empty EEDB entries to be with bit 34 set to 0.
SDK-54519		56850_A0 56850_A1 56850_A2	In the previous release, hash bits were not being calculated in <code>soc_td2_l2x_hash()</code> function. This has been fixed.
SDK-54529		88650_A0 88650_B0 88660_A0	OAM RDI clear event does not generate a callback
SDK-54533 SDK-57729		56340_A0 56340M_A0	Added separate Ingress Qualifier Init routine for Helix4 device with required offset changes as per Regfile
SDK-54536		88030_A0	It is not necessary to guard against oversubscribing the fabric

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54544	748071	56840_A0 56440_A0 56850_A0 56450_A0	We received a customer request to make an I2 entry not routable by resetting the flag BCM_L2_L3LOOKUP. In the previous release, some chips in the Trident family did not remove the entry from my_station_tcam table when the flag BCM_L2_L3LOOKUP was cleared. This has now been resolved.
SDK-54545	741393	All 56850_A0 56850_A1 56850_A2	There was a bug in the SDK when configuring the port using the following function in trident2 platform bcm_port_control_set(unit,port,bcmPortControlExtenderType,BCM_PORT_EXTENDER_TYPE_SWITCH) The SDK was setting the PORT.VT_KEY_TYPE_2 field correctly, but was not setting the PORT.VT_PORT_TYPE_SELECT field. However PORT.VT_PORT_TYPE_SELECT_2 field should be set. Fixed the port configuration in function 'bcm_port_control_set'
SDK-54551 SDK-50401	747647	56850_A0	Support has been added for TD2 for bcm_port_subsidary_ports_get API.
SDK-54557	742238	88650_B1	A priority list is a data structure that keeps a list sorted according to some priority. This data structure is used for TCAM management of the TCAM entries according to priority. Due to a SW bug, invalid memory is read and returned in a local function when trying to get the previous element of the first element, which can cause an invalid memory access. This fix resolves this issue, by returning the head in the aforementioned case.
SDK-54567	748978	88650_A0 88650_B0 88650_B1	The bcm shell diagnostic command "diag pp dblif" support: - for vxlan: key: vni, data: vsi - for l2gre: key: vsid, data: vsi.
SDK-54571	749766	56643_A0 56644_A0 56643_A1 56644_A1	Issue observed was P_START_SPRI was not programmed correctly and this was evident in CLI output of LLS command where the FC "first child" calculation does not match the index of first SP child. Fix provided - in port sched dynamic mode, the P_START_SPRI is correctly configured to the index of first SP child.
SDK-54573	745949	88650_A0 88650_B0 88660_A0	bcm_port_tpid_class_get() should call the SOC_PPD_LL_PPARSE_INFO_clear before using the SOC_PPD_LL_PPARSE_INFO structure.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54585	742690	All	During VPLS Virtual routing using Ingress Field Processor, REDIRECT TO DVP Action [bcmFieldActionRedirect] in Field module takes Virtual Port information from ING_DVP_2_TABLE for devices like Trident, Trident2, Triumph3. Currently we were configuring ING_DVP_TABLE only with next hop entries and support to configure ING_DVP_2_TABLE was missing. Hence added code to configure next hop entries in ING_DVP_2_TABLE during VP port add , through this JIRA. Also added code to delete next hop entries in ING_DVP_2_TABLE during VP port delete.
SDK-54589	749529	56850_A0 56850_A1 56850_A2	Offset state was not being properly cleaned up when programming flex hash. In this release we have corrected the UDF_CONDITIONAL_CHECK_TABLE_RAMM configuration flow when destroy a flexible hash entry.
SDK-54604		56450_A0 56440_B0 56450_B0	In previous releases bcmCosqControlBandwidthBurstMax and bcmCosqControlBandwidthBurstMin could not update refresh rate based on burst and shaping rate. The implementation has been modified to calculate refresh rate and update shaper configuration.
SDK-54605	735909	56640_B0	In single lane or dual lane mode, if the autoneg is enabled, firmware mode 0 should be used. However all the 4 lanes of that core firmware was to set to 0 instead of relevant lanes only. This has been fixed.
SDK-54606	651774	56850_A2	The supports for per-lane PHY controls have been added in TSCMOD.
SDK-54610	750318	All	The counter thread could end up in a continuous loop when sbusdma was busy/not initialized, In this release we have added timeout to break from this loop.
SDK-54615	748837	56224_B0 56224_A0	Background: ===== bcm_vlan_translate_add(), delete, egress_add and egress_delete functions were not implemented to handle wildcard port parameter. Problem: ===== bcm_vlan_translate_add(), delete, egress_add and egress_delete functions were throwing error when wildcard port parameter was passed. Solution: ===== Added port specific check in bcm_vlan_translate_add(), delete, egress_add and egress_delete functions which allows user to perform vlan operations by passing wildcard port parameter.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54617	777127	56640_A0 56634_A0 56640_A1 56640_B0 56634_B0	In previous release, the schan response type for devices with ISM, e.g. Triumph3 and Helix4 is not properly checked. The following response types <code>SCHAN_GEN_RESP_L2_MOD_FIFO_FULL</code> , <code>SCHAN_GEN_RESP_MAC_LIMIT_THRESHOLD</code> and <code>SCHAN_GEN_RESP_MAC_LIMIT_DELETE</code> have been added in schan response type checking in the routine <code>soc_mem_generic_insert()</code> .
SDK-54619		88650_B0	Added diag "cosq flush" to flush all egress queues per port
SDK-54620		88660_A0	In L2 bridging, the number of MACT entries can be limited globally, per FID or LIF. During packet SA learning, events are sent to the OLP when the MACT entry number limit is exceeded. In BCM88660, a new functionality allows to disable sending these limit-reached messages, reducing the number of created events, by setting the <code>switch_control_bcmSwitchL2LearnLimitToCpu</code> to 0. In this case, in the HW, an interrupt is enabled to indicate that the limit was reached.
SDK-54621		88650_B1 88660_A0	For a packet performing a Traffic Management (TM) processing, the regular egress processing removes the system headers (i.e. FTMH and its extensions if exist, PPH and its extensions if exist, User-Header). An improvement allows to remove only the FTMH header and its extensions if exist. If the <code>custom_feature_otmh_keep_pph<port> SOC</code> property is set, then on this port all the headers starting from the PPH header are preserved.
SDK-54625		88650_A0	increased the table size for warmboot in <code>arad.soc</code> in order to have enough storage space when running with OAM application.
SDK-54635 SDK-37263		56846_A1	In the previous release, SDK only configured the mac driver of current mode when invoking the <code>mac_control_set()</code> function. In this release, we will do <code>mac_control_set()</code> in both XMAC and UniMAC MAC driver except for some special cases. This has been fixed.
SDK-54638	727800	56640_A0 56643_A0 56640_A1 56643_A1 56640_B0 56643_B0	An issue was reported where external FP failed to qualify IPv6/TCP-IP packets with given L4SrcPort and L4DstPort. Corrected the offset of the qualifiers L4SrcPort, L4DstPort for external FP (<code>_FP_EXT_ACL_L2_IPV6_ACL</code>) during qualifiers init.
SDK-54640		88650_A0 88650_B0 88650_B1	For a TRILL Multicast entry, the get/delete APIs did not check both port and MC group match. It could cause deletion of an incorrect entry. This is fixed

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54641	746928	88650_A0 88650_B0 88650_B1	STG: A "STG" diag cli command is added to operate or display STG info of device. The usages of "STG" command are listed as below. BCM.0> stg Usage (STG): Usages: stg create [<id>] - Create a STG; optionally specify ID stg destroy <id> - Destroy a STG stg show [<id>] - List STG(s) stg add <id> <vlan_id> [...] - Add VLAN(s) to a STG stg remove <id> <vlan_id> [...] - Remove VLAN(s) from a STG stg stp - Get span tree state, all ports/STGs stg stp <id> - Get span tree state of ports in STG stg stp <id> <pbmp> <state> - Set span tree state of ports in STG (disable/block/listen/learn/forward) stg default [<id>] - Show or set the default STG
SDK-54642	750484	88230_C0	1) Changed #if/#else/#endif comment at #endif to match #if, which was changed from BCM_FE2000_SUPPORT to BCM_SBX_SUPPORT. 2) Changed several internal functions beginning with string_to to static functions to make the more unique to the specific source file.
SDK-54646		56340_A0	SOC EGRESS METERING LOCK is not unlocked on exceptions which led to crash on event processing. Fixed in the exceptions to unlock the semaphore.
SDK-54661	750105	88230_C0 88230_B0 88230_A0	Fixed Make procedure for 88230 devices
SDK-54669 SDK-52871	787225	56850_A0	Previously, trunk based MY STATION TCAM was not programmed for VXLAN and TRILL. Now it is programmed as I3 egress object is created.
SDK-54672	749143	All	Issue :- While doing warm boot(level 2) two times with intra slice double wide group, virtual map information in fp was not recovered properly after the first warm boot and this downgrades the recovery level from level 2 to level 1 during second warm boot. Fix :- While doing level 2 warm boot, after the warm boot succeeds, recreate the virtual map information based on the group information that was recovered.
SDK-54680		88650_A0 88660_A0	MPLS:bcm_mpls_port_add() supports BCM_MPLS_PORT_REPLACE flag to replace egress label only if the tunnel port id is protected. A problem occurs when calling bcm_mpls_port_add() with valid egress label but with BCM_MPLS_PORT_REPLACE. It doesn't return fail. bcm_mpls_port_add() returns BCM_E_UNAVAIL if the parameters include a valid egress label but with BCM_MPLS_PORT_REPLACE.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54688		56846_A0 56845_B0 56845_A2 56844_A0 56850_A0 56855_A0 56843_B0 56854_B0 56854_A0 56850_A1	Issue :- A) Mirror resources(Entries configured in im_mtp_index/ em_mtp_index using MirrorIngress/MirrorEgress fp actions) are not cleaned while deleting FP entry in case FP installation FAIL case. B) Only 3 FP mirror actions are allowed even though there are 4 mirror indexes available in im_mtp_index/ ex_mtp_index. Fix :- ----- A) Added Support to clean up the Mirror Resources as well along with FP entry delete in case FP installation failure. B) It is a hardware limitation where only 3 FP mirror actions are allowed.
SDK-54689		88660_A0	Fix for 88660 egress multicast traffic getting stuck in high egress multicast bandwidth.
SDK-54692	747803	88650_A0 88650_B0 88660_A0	OAM: Deleting a MEP with Long MEG ID fails with assertion.
SDK-54711		88650_B0 88660_A0	User-Header is a fabric header located between system-headers (FTMH, PPH) and start of packet (e.g., Ethernet). The user-header size is set via field_class_id_size SOC property. User-Headers-0/1 can have a total sizes of 0, 8b, 16b, 24b or 32b. The value of 24 bits was not enabled. This is fixed.
SDK-54715	722160	88660_A0	In metering, color blind meters are used to do metering without referring to the color of the incoming packet. Currently when a meter is configured to be color blind, it will always drop incoming red packets by error, instead of ignoring the color. This fix corrects this behavior.
SDK-54722		88650_A0 88660_A0	In Field Processor, when creating or destroying TCAM entries, a time consuming debug code section was running. This code section has been removed, resulting in significant decrease of TCAM entries creation and destroying running time.
SDK-54725		All 56850_A0	Support added in 'bcm_l2_addr_delete_by_vlan_gport_multi' API to flush L2 entries based on virtual ports, deletes based on virtual port trunks are also supported.
SDK-54726		88650_A0 88660_A0	The CCM and Loopback programs in the egress PRGE loaded LFEMs that were not used.
SDK-54731		88650_A0 88660_A0 88670_A0	The error message macros in the soc layer were renamed as following: _SOCDNX_SAND_IF_ERR_EXIT --> _SOCDNX_SAND_IF_ERR_EXIT _SOCDNX_EXIT_WITH_ERR --> _SOCDNX_EXIT_WITH_ERR In addition, a new macro was added: _SOCDNX_IF_ERR_EXIT_MSG

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54748	749898	56450_B0 56440_A0 56450_A0 56440_B0	bcm_port_rate_egress_set API allowed user to configure only the recommended minimum burst value irrespective of the passed burst argument. The API implementation has been modified to compute the shaping parameters based on shaping rate and burst value and configure the hardware tables accordingly.
SDK-54755		88650_A0 88660_A0 88650_B0 88650_B1	PON: SDK now supports also L3 subnet source-bind. For more information see <code>src/examples/dpp/pon/cint_pon_general_anti_spoofing.c</code>
SDK-54761	752348	All	Fixed potential endless loop during PCIe Deemphasis settings, by limiting the range to search for PCIe Capabilities registers to valid range.
SDK-54763	751831	56450_A0	Support has been added for following features for Katana2: 1. Advanced URPF lookup where 2 lookups, both DIP and SIP, are performed using single L3_DEFIP entry at line rate. 2. Capability to add IPv6 LPM entries with subnet mask greater than 64 bits. By default 1K entries are reserved for Ipv6 LPM entries with subnet mask > 64 bits. The default behaviour can be overridden by setting config variable <code>num_ipv6_lpm_128b_entries=0</code>
SDK-54775	750966	88650_A0 88650ACP_A0 88650_B0 88650_B1 88660_A0	L2CP (Layer2 Control Protocol) traps were not updated correctly when calling multiple times <code>bcm_l2_cache_set</code> .
SDK-54776	751147	88660_A0	The OAM and BFD applications are using TCAM HW to identify some OAM packets on transit tunnels, what causes them to be trapped. A SW bug was allowing using a prefix for this key, and multiple Databases were created (for specific forwarding-types). The fixes are: 1. when OAM is enabled, all the packets performs a look-up into this Database at the forwarding stage (i.e. for any forwarding type). 2. since the HW key length for this TCAM Database, there is no place for prefix and this Database is using exclusively now the TCAM banks 12 and 13. 3. the different Databases were unified to a single Database, since forwarding-type is part of the key.
SDK-54779	748470	56850_A0 56850_A1 56850_A2	In the previous release, the feature of cosq warmboot in TRIDENT2 was not supported. In this release, this issue has been addressed by syncing the left members of <code>_bcm_td2_mmu_info[unit]</code> .
SDK-54792		56640_B0	On TR3, if <code>EGR_ING_PORTm</code> register is not configured, L3 traffic received on EHG port seen as source mac and destination mac zero on cpu port. Added configuration for <code>EGR_ING_PORT</code> .
SDK-54802	738723	88750_A0	In polling mode, the hardware IRQ mask is always zero. This is TRUE for all devices. Implementation fixed to achieve this.
SDK-54810	752795	56450_A0	Support added for BCM56450 (Katana2) to match 3 MPLS labels in UDF.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54819 SDK-53920	752509	56450_A0 56440_B0	During port shutdown traffic buffered in the queues for the port was not flushed. Implemented queue flush during port shutdown . Implemented the thresholds reset and replay for the flush activity to be completed during congestion scenarios.
SDK-54840	753240	88660_A0	Broad Sync API: implemented ToD get function (<code>bcm_time_capture_get()</code>).
SDK-54845	753234	88650_A0	fixed C++ compilation error: added missing <code>"#include <soc/dpp/SAND/Utils/sand_footer.h>"</code> at the end of <code>arad_debug.h</code> .
SDK-54846	752653	88650_A0 88660_A0	Enabled setting the he Packet-TC to Queue-TC mapping for ISQs using <code>bcm_cosq_port_mapping_set()</code> .
SDK-54848		88650_A0 88660_A0	CINT: New example for IPv4 routing over 802.1q where the VSI/RIF is explicitly supplied by the user and not determined by the incoming VLAN value. This allows VSI routing interface that straddles several LANs. The example uses In-LIF (per In-Port x VLAN) and Out-LIF (per VSI x Out-Port) to achieve that. Reference: <code>cint_ip_route_explicit_rif.c</code> . In the CINT example a routing scheme with two different Routing Interfaces(RIF) that are based on <Port, VLAN> added.
SDK-54849		88650_A0	IMPORTANT: Injection of TM packet with user define header is not supported on systems which have OAM yet (SDK-57826). Background: user headers are optional internal system headers located after the FTMH and PPH headers (extensions included). The User header can be used for different purposes: - Cascaded ingress egress ACLs, to transmit data from Ingress FP to egress FP - Various work-arounds The user header size is configured via <code>field_class_id_size_X</code> SOC property. Issue: when injecting TM packets with additional headers after ITMH (e.g. PPH or OAM-TS) and if the user headers are used, the user must include the user headers in the packet after the additional headers and before the payload (e.g. before the Ethernet header). Set this mode via the SOC property <code>custom_feature_injection_with_user_header_enable</code> . In this mode, the user header is not added: injected TM packets must be injected with a User-Header with the same size as the configured user-header size (<code>field_class_id_size_X</code>). If the destination port of the TM packets are Ethernet port, the user also must set the <code>custom_feature_user_header_always_remove</code> SOC property.
SDK-54865		88650_A0 88660_A0	issues in snoop APIs: <code>bcm_rx_snoop_get()</code> now returns the same size and probability as the values entered by <code>bcm_rx_snoop_set()</code> .

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54869		88660_A0	BFD non-accelerated mep is restored incorrectly after WB.
SDK-54870		88660_A0	In Field Processor range APIs, internal commands were added to skip Warm-boot on these APIs during Warm-boot validation.
SDK-54877		88650_A0 88660_A0	Diag improvement: IPv4 multicast routing table can be displayed from diagnostic shell Diag pp IPv4_MC.
SDK-54880		88650_A0 88660_A0	Diag improvement: The allocation manager section now displays general information regarding all pools. In addition, support was added to the detailed information options of the IVE/EVE pools. The "hw" option was renamed to "direct".
SDK-54903	751870	88660_A0	OAM: in arad plus, packet below the lowest MEP level was not trapped with trap code <code>error_level</code> as it should be but was forwarded.
SDK-54906	749980	54240_C0 54280_A0 54282_A0 54285_C0 54290_A0 54292_A0 54295_A0	WarmBoot support for BCM54240/5428x/5429x has been added.
SDK-54907		88650_A0 88660_A0	Warmboot: The TPID profile stores up to two expected TPIDs for each port. Each profile can be used by multiple ports and should be discarded when no port is using it. Performing Warmboot has doubled the correct number of ports that are attached to each Port Profile. Thus, preventing proper discard of the TPID Profile when no ports are attached to it and eventually causing an init error that may happen after 256 Warmboots. The WB for Port TPID profile now functions correctly.
SDK-54921		88650_B1	In Egress Field Processor, when configuring <code>bcmFieldActionStat</code> action, a validation is performed on the data field value. The validation is incorrect, and in case the value is out of range, it will not be identified. This is fixed.
SDK-54923	752947	88650_B0 88650_B1	For stacking systems, the KeepAlive application allows the CPU to retrieve the stacking link topology by sending unicast packets from CPU to CPU. The implementation is performing a specific process in the second stacking device when the <code>FTMH.Stacking-Route-History.MSB</code> is set. However, this process should be done only for Unicast packets. This is fixed
SDK-54927	752923	56450_B0 56450_A0	In previous releases <code>bcm_cosq_gport_delete</code> API could return <code>BCM_E_TIMEOUT</code> during congestion scenarios. This issue has been fixed in API implementation by adjusting bandwidth and flush the packets completely.
SDK-54931		56854_B0	If there was an error in the internal functions of the <code>ecmp</code> create routines. the software state was not cleared. Made changes to clean the s/w state in case there is some error in internal routines or h/w writes of <code>ecmp</code> creation.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54937	752666	88650_A0 88660_A0	gport shell command shows incorrect voq id (-1) for ingress shaping queues. Fixed.
SDK-54939		88650_A0	In L2 Control Protocol traps, the attributes of the programmable traps and of the Reserved-Multicast traps are saved in the <code>rx_virtual_traps</code> variable between the RX-trap API definition and the L2-cache setting. This variable was not restored correctly after warmboot because the struct was not saved to external storage. Fixed. ISSU: if upgrading from an earlier version, this data is not restored.
SDK-54945		56340_A0 56344_A0	Big Endian mode has been added to the SDK when using Helix4 with iProc and latest LDK release To build SDK in Big Endian mode, type "make ENDIAN_MODE=BE".
SDK-54947	752756	56440_A0	SDK support for 1588 Transparent Clock is added as part of this JIRA.
SDK-54971		88650_A0	In Field Processor, the cascaded value width is set via <code>bcmFieldControlCascadedKeyWidth</code> . The <code>bcmFieldQualifyCascadedKeyValue</code> qualifier has a length equal to this value. However, the <code>bcmFieldActionCascadedKeyValueSet</code> action had always a constant length of 20 bits. This length is reduced to {4 + cascaded width}, where 4 bits are needed for HW encoding. This improvement can be disabled by setting <code>custom_feature_increased_cascaded_action</code> to 1.
SDK-54980	753002	88650_A0 88660_A0	6.3.4 introduced a new feature called IGMP and Compatible-MC after existing tunnel (VXLAN, L2GRE, VPLS) in ARAD+. See <code>cint_igmp_example.c</code> for application explanation and valid packet flows. In HW it required to enable Second-stage-parsing in order to make the feature work. Second-stage-parsing should be enabled only for MPLS TT programs. By mistake we enabled Second-stage-parsing to MPLS, IPV6 and Trill while the correct configuration should enable it to MPLS only. The issue cause Packet-format-code to be Ethernet instead of IPV6 (or Trill).
SDK-54982		88650_B0 88650_B1 88660_A0	At egress, a new feature allows to maintain the User Defined Headers (UDH) before the packet exits the device, by defining <code>UDH_ETH</code> property. As a result, UDH is stamped pre-pending the packet headers. Enable this port by configuring the following SOC properties: 1. Update the port header type SOC property definition to <code>UDH_ETH</code> for this port: <code>-tm_port_header_type_out_[port#]</code> . <code>BCM88650=UDH_ETH</code> 2. Update the User Header sizes according to the <code>field_class_id_size_X</code> SOC property - see its documentation for the acceptable values.
SDK-54984		88650_A0	Fix an error when setting egress port bandwidth (<code>bcm_cosq_gport_bandwidth_set</code> , using <code>GPORT_LOCAL</code>) to low rate relative to other ports.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-54992	753214	56840_A0	Updated Tx packet padding logic in Linux KNET module to properly handle RCPU encapsulation. The previous code could cause Tx data corruption if the padding required reallocation of socket buffer.
SDK-55003		88650_A0	In Rx thread, more internal fields (from FTMH, PPH and their extension headers) are parsed into <code>bcm_pkt_t</code> . A complete description of the parsed fields will be added to the TM User Manual. The parsing is done for ports of type CPU and STACKING.
SDK-55007	752699	88650_A0 88650_B0 88650_B1	Port: <code>ilkn_interface_status_oob_ignore</code> can be used to force ILKN interface status indication. If ILKN interface status indication is forced up after the ILKN OOB interface is enabled, a low number of error message will be sent from ILKN interface. After the fix, ILKN lane and interface status indication will be forced up before the ILKN OOB interface is enabled.
SDK-55023	734742	88650_A0	A new corrective action added at this interrupt handler which checking if the interrupt is cleared every 10ms. The mechanism stops only if the interrupt clear or period of 500ms passed. moreover we are suggesting to use <code>force_unmask</code> option for this interrupt in order to force unmasking the interrupt at the end of interrupt handler. The following is driver reference for this action: <code>uint32* flags; int inter = /*interrupt number*/;</code> <pre>rc=soc_interrupt_flags_get(unit, inter,&flags);BCMDNX_IF_ERR_EXIT(rc); if (value == 0) { SHR_BITCLR(&flags, SOC_INTERRUPT_DB_FLAGS_FORCE_UNM ASK); } else { SHR_BITSET(&flags, SOC_INTERRUPT_DB_FLAGS_FORCE_UNM ASK); } rc=soc_interrupt_flags_set(unit, inter, flags); BCMDNX_IF_ERR_EXIT(rc);</pre>
SDK-55026		88650_A0	XGS MAC extender port support 1G extension capabilities when ARAD/ARAD+ is connected to XGS devices to extend 1G capabilities in chassis. Several Ethernet Inport properties weren't configured right for XGS MAC extender port for example: custom macros for Trill header parsing were not set. Fixed.
SDK-55036		56850_A2	<code>ENQ_ASF_HS_OVERSUB_EN</code> is enabled during init for all the ports in TD2 [SDK-54205] hence the <code>ASF_ENABLE_HS_PORT_EP_CREDIT_CHK</code> also should be set to 0 on init.
SDK-55038	754635	All	Add support for customer-supplied call-backs in the Linux KNET kernel module. The call-backs allow another kernel module to modify packet data and meta data before an Rx packet is handed off from the KNET driver to the Linux network stack, or when a Tx packet is handed off from the Linux network stack to the KNET driver.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55067		88750_A0 88650_A0 88750_B0 88650_B0 88650_B1 88660_A0	Added a mechanism to control logging and console messages formats. Please look at file <code>src/appl/diag/bsldnx.c</code> , function <code>bsldnx_cons_init()</code> .
SDK-55071		56850_A0	Implemented new data formats (macros) to match on the incoming packets with or without VNTAG/CNTAG/ETAG/ICNM packets. For example: setting <code>BCM_FIELD_DATA_FORMAT_F_VNTAG</code> flag, an entry is created in UDF_TCAM to validate on the incoming packets tagged with VNTAG. Likewise, setting <code>BCM_FIELD_DATA_FORMAT_F_NO_VNTAG</code> flag, an entry is created in UDF_TCAM to validate on the incoming packets without VNTAG.
SDK-55081		56640_A0 56540_A0 56640_A1 56640_B0 56540_B0	Before the code change <code>bcm_tr2_cosq_gport_get</code> function is returning only <code>BCM_COSQ_GPORT_UCAST_QUEUE_GROUP</code> , now code is added so that it returns flags as per the type <code>BCM_COSQ_GPORT_VLAN_UCAST_QUEUE_GROUP</code> (for vlan gport) <code>BCM_COSQ_GPORT_DESTMOD_UCAST_QUEUE_GROUP</code> (for dmvoq gport) <code>BCM_COSQ_GPORT_MCAST_QUEUE_GROUP</code> (for multicast gport port)
SDK-55083	753905	56340_A0 56340M_A0	ISM total calculation was simplified. Previously total was incremented initially and from then, every time when the number of entries were bumped up. Now the increment will be done only when we allocate the memory from a bank to a table.
SDK-55084	753827	88650_A0 88660_A0	Trap PWE TTL=0/1 is now supported: 1) <code>bcmRxTrapMplsTtl0</code> , <code>bcmRxTrapMplsTtl1</code> traps are now supported. 2) To set trapping PWE packets with TTL<=1 use <code>bcm_mpls_port_t.vccv_type=bcmMplsPortControlChannelTtl</code> . Example can be found in <code>cint_vswitch_vpls.c</code>
SDK-55095		88660_A0	Trill Warmboot: Upon warmboot, Trill init called to HW access as it shouldn't be.
SDK-55101		88650_B0 88660_A0	Required changes in SDK in order to support KBP-SDK 1.2.1 and higher. The changes include configuration of newly used instructions and their transport layer implementation.
SDK-55102		88650_A0 88650_B0 88660_A0	During initialization, the SOC property configuring the OTMH Destination extension has an uninitialized value, instead of being disabled by default. Fixed.
SDK-55107		88650_A0	Trill warmboot: Upon warmboot, trill sw states were not restored.
SDK-55109		88660_A0	ROP transctions failed when using LE CPU. Fix ROP access endianness.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55132	752736	All 56850_A0 56850_A1 56850_A2	Software state and Ref-counts were not maintained across warmboot. Therefore After warmboot, <code>soc_profile_mem_get</code> api would not be able to retrieve the <code>l3_iif_profile</code> entry as the software states/ref-count are reset and not recovered. Added support to recover the <code>l3_iif_profile</code> state during level-2 warmboot. The bitmap for valid L3_IIF entries are stored in scache. After warmboot, The <code>l3_iif</code> entries are read from scache and ref-counts are set for <code>L3_iif_profiles</code> indexes.
SDK-55143		88650_B0 88660_A0	Required changes in SDK in order to support KBP-SDK 1.2.1 for external TCAM are introduced.
SDK-55161		88650_A0 88650_B0 88660_A0	IMPORTANT - API SIGNATURE CHANGE: For better coherency, the Multicast-ID parameter was changed in the <code>bcm_l2_addr_t</code> structure: the <code>l2mc_index</code> variable was changed to <code>l2mc_group</code> . If used, the user must adapt its calling sequence accordingly.
SDK-55162		88660_A0 88670_A0	IP Routing-Over-Overlay (ROO) refers to a set of protocols/applications where the L2 forwarding to the Host/Next-Hop router is not accomplished by simple 802.1q bridging, but by L2-Overlay protocols (VXLAN, etc). BCM8866X supports ROO Host Unicast over VXLAN. See <code>cint_vxlan_roo.c</code> for cint example and Programmer's Reference Guide for more details.
SDK-55167	755011	56455_A0 56640_A0 56850_A0 56640_A1 56640_B0 56850_A1 56850_A2	Problem: PacketRes enumerations getting remapped internally causing data ,mask mismatch during qualifier installation. Solution: Updated code to qualify packet Resolution in below 2 ways: 1) print <code>bcm_field_qualify_PacketRes(0,0, BCM_FIELD_PKT_RES_L3UNKNOWN, BCM_FIELD_PKT_RES_L3UNKNOWN);</code> 2) print <code>bcm_field_qualify_PacketRes(0,0, BCM_FIELD_PKT_RES_L3UNKNOWN, BCM_FIELD_EXACT_MATCH_MASK);</code> This is also documented as valid set of mask values
SDK-55175	749262	88660_A0	Extracting a BCM88660 that is configured to VSC128 cell format mode, caused performance degradation in the system. Fixed.
SDK-55184		56850_A0	Earlier SDK releases did not allow configuring MTU value for vxlan access ports. This release now supports setting/resetting MTU for vxlan access ports through <code>bcm_vxlan_port_add()</code> API.
SDK-55205	751154	56850_A0	ENABLE_1588MPLSf flag is used to enable/disable encapsulation and decapsulation for PTP packets over MPLS. Memory validation check is added to avoid crash while accessing memory for chips that donot have this flag. TD2 does not have this feature.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55222		88660_A0 88670_A0	MPLS: When adding MPLS termination label using api <code>bcm_mpls_tunnel_switch_add</code> , action <code>BCM_MPLS_SWITCH_ACTION_POP</code> , next protocol after MPLS label will be calculated from next header first nibble and not from the Lif table. Flags <code>BCM_MPLS_SWITCH_NEXT_HEADER_L2</code> , <code>BCM_MPLS_SWITCH_NEXT_HEADER_IPV4</code> , <code>BCM_MPLS_SWITCH_NEXT_HEADER_IPV6</code> are not supported.
SDK-55225		88650_A0 88650_B0 88650_B1 88660_A0	BFD creation of accelerated endpoints with remote destination is fixed and now working without configuring any OAMP instances. Irrelevant validation checks were removed as well. Restrictions on <code>endpoint_id</code> and <code>local_discr</code> fields: 1. In case endpoint is accelerated to the OAMP, endpoint id should be equal to lowest 16 bits of <code>local_discr</code> . 2. In case endpoint is accelerated to the OAMP or endpoint type is <code>bcmBFD TunnelTypeUdp</code> , BFD <code>local_discr</code> msbs (bit number 16 and above) should be constant for all endpoints. 3. Non-accelerated endpoint cannot be created <code>WITH_ID</code> . Also fixed error in creating oam/bfd endpoint with id 4096.
SDK-55229		88650_B0 88660_A0	When using external TCAM, usage of the diagnostics command "kbp print" may have caused a segmentation fault. This happened due to inappropriate use of unallocated memory and is now fixed.
SDK-55243		All	Improved execution time of <code>bcm_l3_intf_create()</code> in XGS devices. In addition, removed deadlock with VLAN APIs such as <code>bcm_vlan_control_vlan_set()</code> .
SDK-55263	739431	56540_A0 56440_A0 56450_A0	Fix for PTP operation using little-endian host.
SDK-55274	756256	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	Problem: Src/Dst IP6 qualifier was sent to 32 bit EFP qualifier routine <code>[_field_efp_qualify32]</code> for TD2 devices which was internally causing the mask to be reset to 0. [since offset width calculation is assuming width to be 32 but actual width is 128]. Since its a 32 bit routine, the last 32 bit part of mask was getting reset here. Solution: Added appropriate checks to make sure that only 32 bit IP address falls into the check and hence mask will not get reset.
SDK-55280	750005	56440_A0	Support has been added for proper reload of <code>MAC_BLOCK</code> table during warmboot for BCM5644x devices.
SDK-55283	756559	All	Removed StrataXGS restriction from <code>bcm_tx_array</code> documentation that all packets should have same values for Source module, Source port, PFM and Internal Priority as it does not exist now.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55286	755758	56850_A0 56850_A1 56850_A2	When L2X table parity error was detected and processed in Y-pipe context, the <code>acc_type</code> list for Y-pipe would be iterated to decode memory id via routine <code>soc_addr_to_mem_extended()</code> . The <code>acc_type</code> of L2X table is 4, not in the list for Y-pipe, and this would cause memory decode fail. So the <code>acc_type</code> 4 has been added into the list for Y-pipe to fix this issue.
SDK-55288		88660_A0	Trill Multi-homing connectivity. Define up to 3 virtual rbridges in system was not correctly supported and leads to memory leak.
SDK-55293	739558	88650_B1	In L2 forwarding, when MAC learning mode was centralized, the aging time accuracy has been improved: the aging time is maximal whether the entry has been inserted by this device or not. A SOC property (<code>custom_feature_centralized_ownership</code>) allows the user to work in previous mode.
SDK-55296		88660_A0	OAM: When replacing entries in the in the O-EM 1/2 tables, instead of deleting the entries and then inserting, it is possible to replace the entries in one fell swoop. Previous configuration might have caused packet loss in the time between the deletion and creation of new entry. Likewise <code>oem1/2_entry_delete()</code> did not wait for the task to complete before returning. This bug was fixed as well.
SDK-55298		88650_A0 88660_A0	When using lag over a stacking system with <code>number_of_trunks=[512/256/128/64]</code> packets might be dropped. When a FAP resolves a LAG destination, it passes the packet to the next stacking FAP with the LAG id and part (8bit) of the lb-key. Since only a part of the lb-key passes, the next FAP may conclude a different destination for the packet. As a result, the packet can be sent back to a FAP that already passed this packet, resulting in dropping the packet. This fix makes the FAP pass the packet to the next stacking FAP with the Destination System Port (DSP) (instead of the lag id), so that next FAP(s) will forward the packet according to the DSP and will not need to recalculate the destination. No change in default behavior, the feature is disabled by default. In order to enable this fix on 6.3.7, the following SOC property configuration is needed: <code>custom_feature_stamp_uc_destination.BCM88650=1</code>
SDK-55299		88660_A0	OAM diagnostics: Lookups are displayed in parsed format (key and result, if found). The relevant command is <code>diag oam lu</code> and the output is for example: IHB OEMA last lookup: Key=0X2002, result=0X60000080 OEMA key: ingress: 0, OAM LIF: 0x1001 OEMA payload: MP profile: 0x3, MEP bitmap: 0x0, MIP bitmap: 0x80, counter index: 0x0 IHB OEMB last lookup: Key=0X1001e, result=0X0 OEMA key: ingress: 0, MDL: 7, OAM LIF: 0x1001, your disc: 0 Not found.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55305		88660_A0 88670_A0	The masks for Vlan gport id and MPLS gport id has been extended from 24 bits to 26 bits.
SDK-55317 SDK-55378		56340_A0	On Helix 4, Bank 0 in every stage is disabled in ISM. This is taken care of while allocating banks for ISM tables, but while configuring the hash_offset for each bank, the disabled bank was not taken into account. Now the number of disabled banks are calculated and is added to the bank number in each stage.
SDK-55323	755643	56850_A0	For Trident_2, when station tcam entries were being recovered during warmboot, they were not being checked for validity. Even blank entries were being counted as valid entries and so, after recovery, the table showed up as full. So, when a new entry was added after warmboot, it returned no resources. This validity check is now added for Trident2
SDK-55326	715274	88650_A0 88650_B0 88650_B1 88660_A0	PON application: ARAD/ARAD+ Ingress parser used to take two global TPIDs to get the tag formats of packets for each port. With this improvement, ARAD/ARAD+ ingress parser can take an additional global TPID as inner TPID, besides the original two global TPIDs, to get the tag formats for each port. Currently only packets with single outer TPID, which is equal to additional TPID, can be parsed as single s-tag or single c-tag. Please see more information in cint_pon_additional_tpids.c
SDK-55329		88650_A0 88650_B0 88650_B1 88660_A0	APIs receiving a bcm_gport_t input argument as a destination, will now work properly when the gport type is MODPORT, and the given module port is not defined in the local device.
SDK-55332	753717	88650_A0 88650ACP_A0 88650_B0 88650_B1 88660_A0	In Ingress Field Processor, the qualifiers bcmFieldQualifyInnerSrcMac and bcmFieldQualifyInnerDstMac can be taken from second or third header in stack (first or second after Ethernet header). In order to indicate which header to consider, one of the qualifiers bcmFieldQualifyIp4, bcmFieldQualifyIp6, bcmFieldQualifyMpls can be used in QSET. If one of these qualifiers exists, then the inner mac will be taken from the third header, otherwise from the second.
SDK-55335	708385	88650_B0	fixed the prbs issue going out the analog part for 8b/10b encoding speed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55336		88660_A0 88670_A0	Implemented VRRP and multiple mymac termination for Jericho, and added several upgrades for ARADPLUS: 1. Removed the constraint of disabling VRRP when multiple mymac termination mode was used. In the past, when using multiple mymac termination, l3_vrrp_max_vid soc property had to be set to 0, and calling VRRP APIs would return error. Now setting l3_vrrp_max_vid to 0 would be ignored, and calling VRRP apis in this case will have no effect. 2. VSI 0 is now legal to use for both features. Calling one of the APIs with VSI 0 would configure all VSIs with the selected VRID / mac address in one call, instead of looping over all VSIs. It can be used for features like MPLS-TP multicast packets where 01-00-5E-90-00-00 should terminate Ethernet header.
SDK-55339		88650_A0	Slow start mechanism for FMQs (using bcmCosqGportTypeGlobalFmqGuaranteed control) is not functional. Fixed.
SDK-55344		88650_A0 88650_B0 88650_B1	BFD: fields that are only used by endpoints accelerated to the OAMP are configured only for relevant endpoints. Likewise in endpoint_destroy().
SDK-55345		88660_A0	OAM: RDI indication on outgoing packets from the OAMP might be inconsistent.
SDK-55346		88650_A0 88660_A0 88650_B0	OAM: In Arad, all MEG levels 0-7 may be used. In Arad+, level 0 is unavailable by default, however this may be used if the classifier is used in Arad mode - if the soc property "oam_classifier_advanced_mode" is set to 0.
SDK-55347		88650_B1	OAM: For trapped DM packets (both up and down, NTP or 1588), the packet will be prepended with the 4 MSBs of the time (the 4 LSBs appear in the OAM-TS). In other words, the packet format will be FTMH+OAM-TS+PPH+4 time MSBs+packet. To use the old format where there is only the 34 bits in the TS unset soc property "custom_feature_oam_dm_tod_msb_add_enable=0" (1 by default).
SDK-55350		88660_A0	Adjusted cint_system_vswitch_vpls.c to fit PWE/LSP pipe mode.
SDK-55352	756202	All	In the previous release there was a coding issue with the usage of sizeof operator. The object used to calculate the sizeof operation in a function was passed as a value instead of passing it by reference. Hence sizeof operator was returning a wrong value. This issue has now been addressed in this release.
SDK-55353	757018	88640_A0	Bug in counter processor calculation of counter ids from counter set ids was fixed.
SDK-55359	756745	88650_A0 88650_B0 88650_B1 88660_A0 88670_A0	Changed Init sequence prints.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55360	757697	All	In the previous release, on TD2, the double bit ECC error notifications from the EP following init cycles were seen on rare occasion. In this release, this issue has been addressed by initializing all of packet buffers to the value of zero.
SDK-55361	749578	All 56850_A0 56850_A1 56850_A2	There are three commands have been added. l3 nat_ingress show l3 nat_egress add l3 nat_egress show These commands enhancements to the BCM diag shell to both program and show NAT status.
SDK-55362	757471	88030_A0	The code to set and check individual bit fields of the PPE variable is now automatically generated by the tools.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55375	755943	56850_A0	<p>Problem : Below qualifier was not getting recovered and actions were getting recovered as colour specific. Action Parameters were not getting recovered for CopyToCpu and EcnNew</p> <p>Qualifier: DstClassL3 Actions: CopyToCpu,,EcnNew,DropCancel,PrioIntNew,EgressMask,Drop,EgressPortsAdd,SwitchToCpu Cancel</p> <p>Solution: Actions mentioned in the list are expected to work this way since action internally sets colour specific actions. Hence after recovery we read from hardware or cache and display individual actions because we cannot confirm if actual action led to these or they were individually configured. Quoting the part in warmboot section in api document where information regarding the above mentioned actions and its behavior is mentioned. "There are some color-dependent actions that may get aliased during Warm boot recovery. For example, the SDK cannot distinguish whether the application added bcmFieldActionDrop or specifically added bcmFieldActionGpDrop, bcmFieldActionYpDrop and bcmFieldActionRpDrop. This is true for all recovery levels."</p> <p>For CopyToCpu, we are passing param0=1 and param1=0, param0=1 -> means that we are matching the rule_id param1=x -> x is the rule_id value that we are planning to match. This code will internally check if param0=1 . If so sets a field MATCHING_RULE as param1 in FP_POLICY_TABLE. Now while recovering we check if MATCHING_RULE !=0 and then recover param1 as rule_id and param0 as 1. Due to this logic, if param0=1 and param1=0, we set MATCHING_RULE as 0. When we recover we dont know if this is due to rule_id=0 or no rule_id configured, because default value for MATCHING_RULE = 0. [we dont have any hardware fields to save param0 to check if rule_id is to be matched or not] This rules out possibility of configuring param1 as 0 with param0=1 if warmboot recovery required. U can configure param1 as 1-127 with param0=1.</p> <p>Code for recovery of parameter of EcnNew Action and for recovery of DstClassL3 qualifier, has been done through this JIRA. EcnNew has a new Field for Triumph3 and Trident2 to keep the value [G_NEW_ECNf] which was missed to be recovered. DestClassL3 was not getting recovered due to double wide mode slice number being passed wrongly. These two problems are handled.</p>
SDK-55387	752326	88650_A0	<p>Configuring a discrete WFQ weight for a CL (using bcmCosqControlDiscreteWeightLevel0..3 controls) with the same weight already assigned by another element failed. Fixed.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55388	756617	88750_A0 88650_A0 88750_B0	The port enable indication might be wrong after warm boot sequence. As a result RX LOS application will not get reliable state of the port and might try to reset the port. Fixed.
SDK-55392		88650_A0	In internal SOC functions related egress port header type setting, beautify the code by introducing #defines instead of hard numbers.
SDK-55396	757120	56850_A0 56850_A1 56850_A2	For TD2 L3_ENTRY table, the case that parity error located in dedicated L3 banks was missed when UFT shared banks are used. Entry index checking for TD2 L3_ENTRY table when retrieving SRAM info via routine <code>_soc_trident2_mem_sram_info_get()</code> has been added for entry indexes in dedicated L3 banks.
SDK-55415 SDK-59514		88650_A0 88650_B0 88650_B1	"g *" command will display MAC regs only once for channelized ports.
SDK-55426		88650_A0 88660_A0	Setting OCB threshold for ingress queues is done with a voq handle. Use this macro to create the relevant gport handle: <code>BCM_GPORT_UNICAST_QUEUE_GROUP_SET</code> . If used to be the case where setting OCB threshold for ingress queues used a voq-connector handle; this is no longer a valid calling sequence.
SDK-55434		88660_A0	In Field processor, at ingress, the Compare operation performs a comparison between the two halves of key-D in second cycle. The comparison first performs a XOR between the two halves and then AND with a predefined mask. The XOR operation is not enabled and therefore the compare result is incorrect. This was fixed.
SDK-55443		88650_A0 88650_B0 88650_B1 88660_A0	PWE: <code>bcm_mpls_tunnel_initiator_create</code> api can be used to update PWE next tunnel used in encapsulation of multicast PWE packets (unicast packets won't be effected). This functionality is available only when PWE is not protected and MPLS tunnel is used by a FEC entry. Example can be found in <code>cint_vswitch_vpls.c</code> <code>switch_pwe_tunnel</code> function.
SDK-55456		56850_A0 56850_A1 56830_A1 56850_A2 56830_A0 56830_A2	In the previous release, flexible counter thread could occasionally report a huge counter statistic when the two hardware counters belonging to two ports which locate at different pipelines rolled over at the same time. In this release, this issue has been addressed by handling rollover for individual pipelines.
SDK-55460		56850_A2	The access type of <code>ING_NEXT_HOP</code> table is defined as 1 per regfile <code>bcm56850_a0</code> . This access type was missed in TD2 Y-pipe list in SER correction routine. The access type 1 has been added into TD2 Y-pipe list in SER correction routine to resolve this problem.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55464		88650_A0 88650_B1 88660_A0	diag nif shell command speed improvement, PHY rate is measured once for each interface, instead of measuring PHY rate for all interface channels. no modifications required in customer applications.
SDK-55470	758460	88650_A0	An updating logic happens when creating VLAN port with BCM_VLAN_PORT_REPLACE and BCM_VLAN_PORT_WITH_ID. If a new key to be added is different from the existed old key, the updating logic removes the old key and adds the new key. An error occurred when the updating logic removed the old key of egress AC for CEP ports. The updating logic compared the new key with an uninitialized old key to check whether the new key is different with the old key. The issue detailed above affected Out AC replacing of CEP ports. The correct egress AC key can be removed after the fix.
SDK-55471		All	<p>=== FOR THE CUSTOMER USING SDK-6.3.X Customer needs to follow below instructions to create new build target.</p> <ol style="list-style-type: none"> 1. copy \$SDK/systems/user/gto-2_6 \$SDK/systems/user/custom-3_10 2. modify 2 lines in \$SDK/systems/user/custom-3_10/Makefile override kernel_version=3_10 platform=myboard-\$ (kernel_version) 3. copy \$SDK/make/Makefile.linux-gto-2_6 \$SDK/make/Makefile.linux-custom-3_10 and modify CROSS_COMPILE, TOOLCHAIN_BIN_DIR, KERNDIR appropriately. 4. copy \$SDK/make/Makefile.linux-kmodule-2_6 \$SDK/make/Makefile.linux-kmodule-3_10 5. cd \$SDK/systems/linux/user/custom-3_10 && make <p>=== FOR THE CUSTOMER USING SDK-6.4.X Customer needs to follow below instructions to create new build target.</p> <ol style="list-style-type: none"> 1. copy \$SDK/systems/user/gto-2_6 \$SDK/systems/user/custom-3_10 2. modify 2 lines in \$SDK/systems/user/custom-3_10/Makefile override kernel_version=3_10 platform=myboard-\$ (kernel_version) 3. copy \$SDK/make/Makefile.linux-gto-2_6 \$SDK/make/Makefile.linux-custom-3_10 and modify CROSS_COMPILE, TOOLCHAIN_BIN_DIR, KERNDIR appropriately. 4. cd \$SDK/systems/linux/user/custom-3_10 && make
SDK-55479	739565	88030_B0	Note.
SDK-55487		88950_a0 88750_A0	Added logging information during initialization.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55495		88650_A0	BFD: bugs that hindered calling <code>bcm_bfd_endpoint_create()</code> with the flag <code>BCM_BFD_ENDPOINT_UPDATE</code> set for <code>type= bcmBFD TunnelTypeMplsTpCc</code> and <code>bcmBFD TunnelTypeMpls</code> were fixed.
SDK-55500	758887	88650_A0 88650ACP_A0 88650_B0 88650_B1 88660_A0	In the <code>cint</code> <code>cint_policer_metering_example.c</code> , the function <code>header_compensation_example</code> used the wrong function to set header compensation. This is now fixed.
SDK-55501		88650_A0 88660_A0	In Field Processor, when creating a new Field Group, it is verified that the key can be allocated with the existing occupation of the program's instructions. However, the verification does not consider the used key bitmap which may indicate that all LSB/MSB keys are used. In which case, the algorithm should disregard the relevant (LSB/MSB) instructions. This is fixed.
SDK-55502	759144	56450_B0 56450_A0	<code>soc_mem_config_set()</code> (is set to <code>sal_config_set()</code> in our local SDK environment with SAL implementation) may or may not be available with customer code. so <code>assert</code> is not considered good idea. If <code>soc_mem_config_set</code> not available and <code>auto_portgroup</code> and <code>auto_polarity_flip</code> is set true, SDK will suggest settings on screen so that end user can re-update <code>config.bcm</code> accordingly. Also made auto generated config variables unit specific (i.e. <code>portgroup_<num>.unit=<lanes></code>) happens with <code>auto_polarity_flip</code> and <code>auto_portgroup</code> config variables. This is relevant in multi unit setup.
SDK-55515	752139	56640_A0 56440_A0 56850_A0 56440_A1 56640_A1 56640_B0 56440_B0 56850_A1 56850_A2	<code>bcm_port_learn_set</code> is used to control the learning behavior on a port. The learning behavior can be set/modified using this API. This API was not supporting vlan virtual ports previously. Now, support is added to modify learning behavior for vlan virtual ports.
SDK-55518	757054	56634_A0 56634_B0	<code>START_BY_START</code> error interrupt was not being handled resulting in high CPU utilization. Added handler for this error, to clear the interrupt status register when set.
SDK-55524	759557	88660_A0	<code>bcm_port_loopback_get</code> bug fix for ILKN port in 2 Caui+ ILKN mode (BCM 88660)
SDK-55528		88650_A0 88660_A0	OAM: <code>bcm_oam_endpoint_action_set</code> supports new actions: <code>bcmOAMActionUcFwdAsData</code> , <code>bcmOAMActionMcFwdAsData</code> to configure forwarding the packet instead of trapping/snooping. The destination when calling this api with the actions above should be <code>BCM_GPORT_INVALID</code> . This scenario is useful in case of MIP where we should forward the data as is without any special OAM action.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55531		56340_A0	The variables to calculate the tokens are integers (4 bytes) but on multiplying two integer variables results in a much bigger number which cannot be accommodated in 4 bytes of allocated memory. This was impacting the vstorm control feature. Declared a temp variable of long integer (8 bytes) to store the resultant value to fix the issue.
SDK-55537		88660_A0	When doing metering on packets, it is possible to compensate for Ethernet inter-packet gap (IPG) and/or Ethernet preamble by setting the switch control bcmSwitchMeterAdjustInterframeGap to 20. This will add 20 bytes to the packet size for meter compensation calculation. Currently due to a software bug, this switch control is not set, and no compensation is performed. This is now fixed.
SDK-55540		88650_A0 88650_B0 88650_B1 88660_A0	An access to an HW table (EGQ-VSI-Profile memory) was performed with a uint32 variable, although the table width is 33 bits. It resulted in a memory corruption. This is fixed.
SDK-55542	755020	88650_A0 88650_B0 88650_B1	Ring Port: G.8032 Ring-Port can be associated with multiple VLAN-Ports using <code>bcm_port_class_set()</code> . De-associating a VLAN-Port from a Ring-Port where the physical port is on remote device have sometimes left the de-associated VLAN-Port in a state where it can't be reused and failed when referred by VLAN-Port APIs. The issue was fixed, so that remote VLAN-Ports that are de-associated can always be reused.
SDK-55543	759990	88030_B0	EML_144 supported added to tools: <code>INDEX_TYPE_144 LKUP_EML_144</code> Note that EML_144 can not be mixed with EML_176.
SDK-55559	760422	56643_A1	The new support for the below port configuration has been implemented. Device =56643 Frequency (MHz)= 450 Option = 4 GbE Port Group (XC[12:0]) = 36 x GbE+1 x GbE High Speed Port Gr 1 (WC[2:0])= 4 x XFI High Speed Port Gr 2 (WC[6:3])= 2 x HG[42] + x F.H [42]" AXP Port Guaranteed Bandwidth = 5G
SDK-55560		56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56340_A0 56344_A0 56342_A0 56342M_A0 56340M_A0	add an workaround for TR3 and HX4 on both cases <code>AT_L2_Limit_019</code> and <code>AT_L2_Limit_042</code>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55567		All 56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	Problem: bcm_field_qualify_data_get was not working for little endian hosts because of ordering of bytes during copy. Solution: If the host is little endian, updated code to swap the bytes in the expected ordering required for further processing.
SDK-55583		56640_A0 56640_A1 56640_B0	Currently, Policers in cascade mode are mapped based on dot1P priorities of the incoming packet. Two new policer group modes are added : bcmPolicerGroupModelIntPriCascade and bcmPolicerGroupModelIntPriCascadeWithCoupling which map the incoming packets to policers based on internal priority. Policers work in cascade mode where bandwidth flows from higher priority to lower priority.
SDK-55600	719068	56449_B0 56445_B0 56440_B0 56447_B0 56443_B0 56441_B0 56446_B0 56448_B0 56442_B0	Fixed crash observed during Level 2 warmboot on BCM56440.
SDK-55604	760276	56224_B0 56224_A0	Issue :- After warmboot, Recovered Entries were being shown as Disabled. Fix :- The Entries were actually recovered properly both in H/W and S/W But code changes to mark the entries are enabled was missing. Added the code change to mark the recovered entries as enabled.
SDK-55615	758680	88650_A0 88650ACP_A0 88650_B0 88650_B1 88660_A0	Counter processor example cint, cint_voq_count.c, was updated. The bcmCosqGporYellowAcceptedPkts counter type replaced by bcmCosqGportNotGreenAcceptedPkts. This change reflects change in the counter processor counters in FULL_COLOR counting mode from version 6.3.2

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55620	758957	56640_A0 56540_A0	<p>Two issues fixed as a part of this JIRA:</p> <p>Issue 1: Linkscan SW mode becomes NONE after executing the ibod WAR. Fix: IBOD sync recovery function <code>_bcm_tr3_ibod_sync_recovery_port()</code> is invoked from different threads and as the function is not properly protected, there is difference in the linkscan states. Provided the synchronization using <code>IBOD_LOCK</code>.</p> <p>Issue 2: When <code>bcm_port_enable_set</code> is getting called even before the LS thread is updated its bitmap ,so when <code>bcm_port_enable_set</code> calls <code>_bcm_tr3_ibod_sync_recovery_port</code> it takes the snap shot of port mode ,which comes to "BCM_LINKSCAN_MODE_NONE" so later on at the end of the function when it update the port mode it removes it from LS ,that where we see some times port is not part of linkscan.</p> <p>Fix: During the ibod WAR execution, the links of the port are set to link UP forcefully by invoking the API <code>_bcm_esw_link_force()</code> API with flags <code>_BCM_LINK_STATUS_NO_CALLBACK</code>.</p> <p>The flags are introduced newly and if the flag is set, the link state change notification is ignored to the registered linkscan users in function <code>_bcm_esw_linkscan_update_port</code>.</p>
SDK-55621		88650_B1	<p>When replacing existing MTU value using <code>bcm_l3_intf_create</code> api, the MTU value might in some cases change to 0 instead of the requested value. This happens in case MTU value is unique for certain L3 Intf</p>
SDK-55630		88660_A0	<p>OAM: when calling <code>bcm_oam_loss_add()</code> with the flag <code>BCM_OAM_LOSS_SINGLE_ADDED</code> set, loss management will be based on LMM PDUs, otherwise on CCM PDUs.</p>
SDK-55631	758623	88650_B1	<p>It is now possible to assign ports with a vlan translation port property, and create IP tunnel terminators that use <code>{SIP,DIP,Next_protocol,Port_property}</code> as key for tunnel termination. To activate this mode, use soc property: <code>bcm886xx_ip4_tunnel_termination_mode= 4 or 5</code> For an example, see <code>cint_ip_tunnel_term.c</code>, call <code>ipv4_tunnel_term_next_protocol_example</code> with <code>use_port_property=1</code>.</p>
SDK-55632		88650_B1	<p>In FLP program selection initialization, some program IDs may have been overridden due to static program ID allocation that followed dynamic program ID allocation. For example, there were conflicts between FCoE and MAC-in-MAC FLP programs. All dynamic allocation of program IDs is now after static allocation, so that no program ID override can be caused. Note that if ISSU is performed, the fix will not apply.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55639		56850_A2	In earlier releases, nexthop and ecmp reference count were not decreased when replacing vxlan port. This has been resolved.
SDK-55654	754909	56850_A0 56850_A1 56850_A2	Fixed DMA abort sequence in KNET Linux kernel module.
SDK-55661	761066	56548_A0 56547_A0	Support for F.HG[42] [SDK-46947] has been ported to the 6_3_branch.
SDK-55681		56850_A2	In the previous release, assertion happened when <code>bcm_cosq_port_mapping_set</code> was called in ETS mode. In this release, this issue has been addressed by configuring a correct field of <code>COS_MAPm</code> and modifying the queue mode of HG ports to the value of zero.
SDK-55683		53394_A0	Added SPI slave mode support of BCM56150 family. Only pure register access path is available in this mode without interrupt and DMA and the access speed is pretty slow in comparison to PCIe.
SDK-55691		88650_B1	In L3, when calling the API function <code>bcm_l3_host_add()</code> , a lock may have been taken but not released in some cases. The lock is always released now before exiting the function.
SDK-55710		88650_A0 88650_B0 88660_A0	OAM: Deleting a MEP with RX configurations only (gport field in <code>endpoint_create</code> api is <code>BCM_GPORT_INVALID</code>) was failing.
SDK-55712 SDK-55535		88650_A0 88660_A0	Add the option to Use Dram saved config Parameters, and in case there are no Parameters to Perform Shmoo on init. Set this option as Default. # 2 = Use Dram saved config Parameters, if no Parameters Perform Shmoo on init. Default option. # 1 = Perform Shmoo on init. # 0 = Use Dram saved config Parameters, if no Parameters do nothing. <code>ddr3_auto_tune.BCM88650=2</code> Also, as default Load DRAM tuning properties from local File (<code>/home/negev/bcm88650_dram_tune.soc</code>). <code>RcLoad</code> will not fail if file not found.
SDK-55713		88650_B0 88650_B1 88660_A0	Broad Sync API: implemented all missing <code>bcm_time_*</code> APIs.
SDK-55715		88650_A0	PWE: verification case of updating TPIDs per PWE using <code>bcm port tpid</code> APIs <code>bcm_port_tpid_add/delete</code> does not work correctly (API always update TPIDs regardless of gport type)
SDK-55719		88650_A0 88650_B0 88660_A0	OAM: <code>api_bcm_oam_endpoint_get</code> returns incorrect flags in field <code>flags2</code> .
SDK-55720		88650_A0 88660_A0	In Ingress Field Processor, when using TM programs per port profile (soc property <code>post_headers_size</code> is set), the program selection shuffle algorithm resets lines of Ethernet programs due to incorrect range calculation. This is fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55722	761214	56450_A0 56450_B0	<p>External DDR has 1024 columns but SDK was assuming it as fixed 2048. Due to this, calculated max bist address was crossing boundary and SDK was throwing assertion(crash) message. Issue is fixed by below two steps 1) Added safety check for max BIST Addr. With this, even if user passes wrong parameter for running DDR TR 140 test case, assertion (i.e. crash) will be avoided.</p> <p>2) Used config properties (ext_ram_columns and ext_ram_banks) to set NUM_COLUMNS and BANKS of external DDR memory. i.s.o. fixed 2048 and 8. Default will be 1024 and 8. Settings will be displayed on screen. With this, if required, user can change values based on connected DDR capability.</p>
SDK-55727		88650_A0 88650_B0 88660_A0	OAM: Mac-In-Mac OAM packet identification causes non-oam packets to be trapped to OAM engine.
SDK-55730		56850_A0 56850_A1 56850_A2	In the previous release, bcm_td_cosq_gport_detach intermittently returned BCM_E_RESOURCE incorrectly when the schedule nodes were not used up. In this release, this issue has been addressed by releasing the schedulers which are used by legacy setup once ETS mode is enabled.
SDK-55736		88650_B1	<p>In FCoE application, a new improvement allows the support for VSAN assignment from VFT or VSI (according to a device configuration) and supports a default VFT value per incoming port. The calling sequence is: 1. Set the default VSAN assignment between VFT (by default) or VSI via bcm_port_control_set(unit, port = -1, type=bcmPortControlFcoeFabricSel, value = bcmPortFcoeVsanSelectVft / bcmPortFcoeVsanSelectOuterVlan)</p> <p>2. If the mode is VFT, set the default VFT per port via bcm_port_control_set(unit, port, type = bcmPortControlFcoeFabricId, value);</p> <p>Note: FCoE application cannot co-exist with the usage of the Field Processor bcmFieldQualifyInterfaceClassProcessingPort qualifier at external stage (bcmFieldQualifyStageExternal) due to the usage of the same HW resource (the port key profile in forwarding stage).</p>
SDK-55740 SDK-56736	757357	All	sand_err code mechanism shouldn't be used without initialization. If init sequence failed before initializing the error mechanism & deinit try to use it Segmentation error will occur. In order to solve this problem we are not using sand_error mechanism at deinit sequence.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55766		56640_A0	Currently, Policers in cascade mode are mapped based on dot1P priorities of the incoming packet. Two new policer group modes are added : bcmPolicerGroupModelIntPriCascade and bcmPolicerGroupModelIntPriCascadeWithCoupling which map the incoming packets to policers based on internal priority. Policers work in cascade mode where bandwidth flows from higher priority to lower priority.
SDK-55770	762574	56850_A0	In previous releases, if multi-thread user accessed L3 memory with different view (i.e., L3_ENTRY_IPV4_UNICAST, L3_ENTRY_IPV4_MULTICAST), the physical memory could be corrupted because different view of same physical memory was using different LOCK, and the entry movement between banks could happen with invalid LOCK protection. In this release, different views of same physical memory are pointed to same LOCK, so the protection is effective.
SDK-55793	757103	88650_A0 88650_B0 88660_A0	VPLS: Enabled modification of working Incoming-PWE configuration under traffic by first creating new instance (the traffic will be moved to the new instance), then deleting the old entries using bcm_mpls_port_delete(). Example can be found in cint_vswitch_vpls.c when make_before_break field is set to 1.
SDK-55803 SDK-55946		88660_A0	When using bcm_port_control_set with the control bcmPortControlEgressModifyDscp, an inlif profile is expected in the port argument. An issue was found when one of the ports 0-16 is disabled. In this case when using an inlif profile with the same number as a disabled port, the API will produce an error, even though the argument is valid. This is now fixed.
SDK-55818	761770	56334_B0 56334_A0	In the previous release, SDK delete old next hop entry before new entry was installed when invoking the bcm_mpls_port_add API with BCM_MPLS_PORT_REPLACE flag asserted. In this release, this has been changed to delete old entry after new entry is installed.
SDK-55822		88650_A0	LUT ROP transactions was failing when using LE CPU. Fix LUT ROP access endianness Also improve KBP code by: Add NULL checks at XPT layer. Add ARAD_KBP_ROP_DEBUG_PRINTS define around prints to Improve access time.
SDK-55823		88650_A0 88660_A0	Function that related to Petra-B in Trill moved to trill.c/h files. Remove initializing of sw-states(mc_trill_route_info_db, mc_trill_root_src_db) from ARAD.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55825		88650_A0 88650_B0	<p>IMPORTANT: In Rx parsing the <code>src_gport</code> and <code>dst_gport</code> interpretation and values were switched.</p> <p>Before, due to a bug, the <code>dst_gport</code> had the same interpretation as <code>src_gport</code>. From now on, the <code>src_gport</code> is the Source-Port where the packet enters the device and <code>dst_gport</code> is where the packet exits the device.</p>
SDK-55830	763499	88650_B0 88650_B1 88660_A0	<p>Trill Ingress learning: For TRILL egress MC RBridge, it learns according to the native SA and VSI. The original instruction which is used to learn native SA is incorrect for ingress learning. It caused ARAD to learn a random SA. The correct SA can be learned after fixing the instruction of lookup native SA.</p>
SDK-55831	762481	56340_A0 56344_A0 56342_A0 56342M_A0 56340M_A0	<p>Helix4/Triumph3 supports SW based aging and when age interval was modified, the new value did not take effect immediately. Code changes have been added to notify the SW Aging thread when age interval is updated, to take immediate effect.</p>
SDK-55840	761378	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	<p>In earlier releases entries[] can be potentially used without initialization in <code>_bcm_td_cosq_wred_set</code>. This has been resolved.</p>
SDK-55850		56846_A0	<p>Support has been added for HG[11] and force cl72 on TD+.</p>
SDK-55857		88650_A0 88650_B0 88650_B1	<p>IMPORTANT: the interpretation (and value) of <code>pkt->pkt_len</code> has been changed.</p> <p>In Packet parsing, 2 fields in <code>bcm_pkt_t</code> are referring to the packet length: 1. The <code>tot_len</code> (total length) field is unchanged, and corresponds to the packet length as received 2. The <code>pkt_len</code> field is changed to correspond to the packet length without the internal headers (i.e. system header size as FTMH, PPH, etc.). The previous value of <code>pkt_len</code> was equal to <code>tot_len</code>.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55859	758730	56640_A0	<p>Two issues are resolved as a part of this JIRA. Here is the description:</p> <p>Issue 1: When a port gets the link up notification and also a remote fault is detected on the port, the port remains down and the <code>linkup_bitmap</code> and <code>fault_bitmap</code> for the port are set. After this state if the port is removed from the SW linkscan mode (the <code>fault_bitmap</code> is cleared) and if a link up notification occurred after adding the port back to SW linkscan. The function returns without setting the link status (as <code>linkup_bitmap</code> is already set to up) and it causes the SW link status for the port in down status.</p> <p>Solution- While clearing the <code>fault_bit</code> map, also clear the <code>link_bitmap</code>. So that when the linkscan for the port is enabled, it updates the state in the next iteration as per the new link status.</p> <p>Issue 2: When <code>bcm_port_enable_set</code> is getting called even before the LS thread is updated its bitmap ,so when <code>bcm_port_enable_set</code> calls <code>_bcm_tr3_ibod_sync_recovery_port</code> it takes the snap shot of port mode ,which comes to "BCM_LINKSCAN_MODE_NONE" so later on at the end of the function when it update the port mode it removes it from LS ,that where we see some times port is not part of linkscan .</p> <p>Solution- For this particular scenario, while restore the linkscan mode after the ibod WAR, retrieve the current linkscan mode and comparing it with the mode it got set before ibod WAR (<code>BCM_LINKSCAN_MODE_NONE</code>) , if it is not same, the linkscan mode is not restored.</p>
SDK-55882		88650_A0	<p>In Warmboot module, some fixes are inserted to prevent some uncaught <code>wb_engine</code> set/get timing issue (in <code>ipmc</code> module on 6.3 branch). Besides, the error mechanism in <code>wb_engine</code> is changed to raise assertions when uninitialized SW database is accessed.</p>
SDK-55885		88650_B0 88650_B1 88660_A0	<p>In case of User Defined Header, Egress Programmable Editor default program incorrectly removes some data bytes from the packet. the fix updates the <code>additional_bytes_to_remove</code> to 0. This way no additional bytes are removed beside the system and network headers.</p>
SDK-55889	762107	88650_B1 88660_A0	<p>In Field Processor, when creating Direct Extraction field group, only one (1) qualifier is allowed to be used as filter qualifier per entry. When calling <code>bcm_field_qualify_data()</code> for an entry, and then calling <code>bcm_field_qualify_xxx()</code> the operation succeeds when an error should be produced. This is fixed.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55902		56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56340_A0 56640_A1 56643_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0 56344_A0 56342_A0 56342M_A0 56340M_A0	Problem: When REGEX feature is enabled, compilation of SDK for linux kernel mode fails. This is because of inclusion of ctype.h file, which is not available directly. Solution: Removed the inclusion of ctype.h. Also re-define the logic in the function isprint() as a new function local to the file, since isprint() is dependent on ctype.h. Affected platforms: All platforms where REGEX is supported.
SDK-55903		56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56340_A0 56640_A1 56643_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0 56344_A0 56342_A0 56342M_A0 56340M_A0	Problem: When REGEX feature is enabled, the compilation of SDK in Linux Kernel mode fails because of variable declarations mixed up with code. Solution: Moved the variable declarations to the beginning of the function and removed some dead code to get the compilation working. Affected Platforms: All platforms where REGEX is supported.
SDK-55913	763695	88650_B1 88660_A0	OAM may be initialized without setting any of the counter_engine_source_{0,1,2,3} soc properties to EGRESS/INGRESS_OAM. Notice that in this case LM functionality is not supported.
SDK-55915	764134	56850_A2	In earlier releases, in ALPM mode, even if we disabled URPF, the bits URPF_LOOKUP_CAMx in register L3_DEFIP_KEY_SEL was still 1. Switching back and forth between urpf and non-urpf could result in URPF_CAM_LOOKUPx bits always set to 1. This has been addressed by making sure register settings are set correctly every time urpf switch control changes, and not just the first time.
SDK-55919	764630	56850_A0 56850_A1 56850_A2	Previously, bcm_vxlan_port_add with BCM_VXLAN_PORT_REPLACE overwrote CML_FLAGS set by bcm_port_learn_set. It is fixed now.
SDK-55920	742940	88030_A0	Fix EML304 and EML424 lookup for bcm88030
SDK-55921	764681	56850_A0	In earlier releases, nexthop and ecmp reference count were not decreased when replacing vxlan port. This has been resolved.
SDK-55935	763171	56850_A0	In earlier releases, the disabled pbmp of flexible ports was not recovered during the warmboot. This has been resolved.
SDK-55942	764885	56850_A0	Implemented following IFP missing actions on TD2. bcmFieldActionPortPriIntCosQNew bcmFieldActionRpPortPriIntCosQNew bcmFieldActionYpPortPriIntCosQNew bcmFieldActionGpPortPriIntCosQNew
SDK-55945		88650_A0 88650_B0 88660_A0	Allocation manager malfunction was fixed in OAM and L3 applications. The bug was in allocating new profile resources (oam endpoint new actions, ttl scope) instead of existing profile.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55956	764773	88660_A0	<p>In trap module, the <code>bcm_l2_cache_set</code> API is used to configure Reserve-Multicast and Programmable traps. This API returns an index, which can be used to delete the trap with <code>bcm_l2_cache_delete</code>.</p> <p>Due to a SW bug, <code>bcm_l2_cache_delete</code> was allocating another trap instead of deleting the allocated one. This is fixed. Besides, <code>bcm_l2_cache_get</code> was returning incorrectly the EtherType (and its mask) parameters. This is fixed.</p>
SDK-55964	742713	88650_B0 88650_B1 88660_A0	VLAN-Port Protection: Replace functionality of 1:1 protected VLAN Port to update <code>failover_id</code> is now available
SDK-55967	755351	88650_B0 88650_B1 88660_A0	OAM/BFD: When calling <code>bcm_bfd_init()</code> after <code>bcm_oam_init()</code> , not all BFD functionalities were properly initialized. Analogously when calling <code>bcm_oam_init()</code> after <code>bcm_bfd_init()</code> .
SDK-55968	756702	88660_A0	OAM: configuring correct counter pointer for accelerated loss management, as well as correctly stamping counters on CCM based LM.
SDK-55970		56440_A0	<p>The parity protection on TCAM tables is implemented via SER engine and a SRAM table that is utilized to store parity bits of TCAM entries. Only enabling SER engine for the new-added <code>L3_DEFIP</code> table but not clearing its corresponding SRAM portion will leave the parity bits of <code>L3_DEFIP</code> table in an uninitialized state with random values. if the table is dumped, SER engine will check entry parity bits of table entries, this will trigger many parity errors reported. Besides adding <code>L3_DEFIP</code> table into SER engine protection list, memory clear operation for <code>L3_DEFIP</code> has also been added to initialize the parity bits of <code>L3_DEFIP</code> table into correct values.</p>
SDK-55972	764939	56850_A0 56850_A1 56850_A2	Code for Warmboot support of <code>MPLS_EXP_MAP</code> has been added.
SDK-55974		88650_A0 88650_B0 88650_B1	<p>When using external TCAM, the access ROP mechanism was substantially improved. The following new compilation flags are available:</p> <p><code>ARAD_KBP_ROP_OPTIMIZATION</code> - enable ROP performance optimization.</p> <p><code>ARAD_KBP_DISABLE_IHB_LOOKUP_REPLY_FOR_ROP_TRANSMIT</code> - enable ROP optimization without reading the IHB reply registers.</p> <p><code>ARAD_KBP_ROP_TIME_MEASUREMENTS</code>, <code>ARAD_PP_KBP_TIME_MEASUREMENTS</code> - enable time measurements.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-55997		56640_A0 56544_A0 56542_A0 56541_A0 56540_A0 56524_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0 56524_B0 56540_B0 56541_B0 56544_B0 56542_B0	Enhancement:- There are 16 FP physical tcam slices with 512 entries per slice. There are 8 physical FP meter pools with 1024 entries per meter pool. Currently only 8 physical tcam slices are allowed to access the 8 FP meter pools. Requirement was to ensure the 16 physical tcam slices are allowed to access the 8 FP meter pools Support :- The 8 FP Physical meter pools are split into 16 logical meter pools so that the 16 FP tcam slices can attach to the 16 Logical FP meter pools.
SDK-55998		56240_B0	Support has been added for the new Saber SKUs BCM56245 and BCM56246 with support for 256k buffer entries/192MB buffering.
SDK-56009	765570	88650_A0 88650_B0 88650_B1 88660_A0	In Rx Trap module, an error is fixed when calling <code>bcm_rx_trap_type_create(unit, 0, type, &trap_id)</code> with 'type' as one of the following : - <code>bcmRxTrapIpv4SipEqualDip</code> - <code>bcmRxTrapIpv4DipZero</code> - <code>bcmRxTrapIpv4SipIsMc</code>
SDK-56013	765696	56850_A2	Fixed <code>tunnel_initiator_delete</code> followed by <code>tunnel_initiator_create</code> . In previous releases, this case could result in an abort of the SDK.
SDK-56015		88650_A0 88650_B0 88660_A0	OAM: MIPs default behavior was changed to the following: MIPs are transparent to all OAM packet types except for LTM unicast, LTM multicast and LBM unicast. If a MIP receives any other OAM packet with destination address == MIPs MAC address (configured in the <code>dest_mac_address</code> field in <code>bcm_oam_endpoint_create()</code>), the packet will be trapped to the CPU with trap code <code>oam-error-level</code> . If the destination address != MIPs MAC address then the packet will be forwarded (it was trapped to the CPU until now).
SDK-56017	765489	56840_A0	Enhanced warmboot shutdown to detach and close KNET device when present. This allows the application to remove the KNET kernel module without exiting.
SDK-56022		56850_A0 56850_A1 56850_A2	In the previous release, <code>bcm_vxlan_port_delete</code> returned <code>BCM_E_NOT_FOUND</code> for default VPN associated NW port. The problem was caused by the flex-counter detachment on VFI table. Originally the detachment was implemented in the deletion of VxLAN logical port and thus the operation on VFI table was executed repeatedly when deleting many logical ports in the same VFI. In this release this issue has been fixed by moving the operation to the VPN destroy.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56024		56850_A0	There's a bit in the VLAN_XLATE table called VLAN_ACTION_VALID, It must be enabled to process XLATE_DISABLE_VLAN_CHECKS for VXLAN virtual ports, but disable it for VXLAN access ports to drop packets at ingress. They have conflict. To solve the problem, a new flag has been added that allows the customer to control the bit, The new flag is BCM_VXLAN_PORT_ENABLE_VLAN_CHECKS.
SDK-56033	765288	56850_A0	Problem: Multiple Mirror Ingress actions were not removed during bcm_field_action_remove_all because during the action remove routine, we removed first MirrorIngress action and then returned without further processing the remaining actions Solution: Updated code to loop through all the actions [in case of similar group of actions] to remove each one of them in action remove routine.
SDK-56038	766065	56850_A0 56850_A1 56850_A2 56455_A0	During warmboot, the reference count for DSCP_TABLE has been updated to reflect the coldboot state.
SDK-56040 SDK-56095	766058	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In earlier releases bcm_esw_port_dscp_map_get() was taken care only for BCM_PRIO_DROP_FIRST, not taken care for other CNG values. This has been resolved.
SDK-56043		88660_A0	During warm-boot validation, multiple issues have been found: 1. The bcm_l2_init was considered as a separate API: when called, the L2 module was detached and re-attached. This is fixed since L2 is initialized during BCM init and cannot be considered as separate API 2. When the device is initialized in TM (Traffic Management) mode, some init code was accessing by mistake uninitialized SW DB. This is fixed. 3. In L2 module, the freeze state (e.g. set by the bcm_l2_addr_freeze API) was not restored correctly after Warm-boot. This is fixed
SDK-56045	766017	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	Committed Information Rate (CIR) and Committed Burst Size (CBS) configured in ICAP policer were not recovered correctly during warm boot on TR3 device. The ICAP policer recovery logic is updated to fix this issue.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56047	761668	56850_A2	The customer requested configuration of RTAG7_HASH_CONTROL_4.VXLAN_PAYLOAD_HASH_SELECT_A/B to meet their hash requirement. For Trident2 and subsequent XGS devices, 2 switch controls bcmSwitchHashVxlanPayloadSelect0 and bcmSwitchHashVxlanPayloadSelect1 have been provided to support the requirement.
SDK-56053		88650_B1 88660_A0	IP Tunnel CINT: In a GRE termination example in cint_ip_tunnel_term.c, a tunnel configuration was changed to use the correct GRE enum type.
SDK-56058	766252	56850_A2	Fixed specific sequence of (SIP, multi-DIP)-add followed by delete and then add of vxlan_tunnel_initiators.
SDK-56068	765431	56640_A0 56640_A1 56640_B0	In the previous release there was an issue reported where TR3 semlock was out of order when creating 2 OAM sessions with same vlan/port, different level. This issue of memory locks not being released in failure case in OAM code for OAM_OPCODE_CONTROL_PROFILEm and ING_SERVICE_PRI_MAPm has been fixed.
SDK-56069		56340_A0	while merging the ranges of TCP and UDP, on range not equal pointer index should increment. In this case no increment is done which leads to infinite loop on the same pointer. Fixed the indexing increment on no range match.
SDK-56071		88650_B0 88650_B1 88660_A0	OAM: For UP-MEPs, all OAM frames trapped to the FPGA/CPU will be prepended with one set of internal headers, specifically an FTMH, PPH and a FHEI, with the OAM-ID on the FHEI. Formerly some frames included two sets of internal headers.
SDK-56074	750523	56440_A0 56440_A1 56440_B0	Issue: After setting spn_BCM5644X_CONFIG to 1 to split HG2 and HG3 into GE24 - GE31, the number of priority groups for these ports were not updated. Fix: After setting spn_BCM5644X_CONFIG to 1 to split HG2 and HG3 into GE24 - GE31, the number of priority groups for these ports are changed from 7 to 0, before configuring the priority group related registers/tables in BCM5644x devices.
SDK-56100	751146	56450_B0 56450_A0	Support has been added for APIs bcm_port_timesync_config_set() and bcm_port_timesync_config_get() for BCM5645x devices.
SDK-56108	762032	88660_A0	OAM: Enable creating accelerated MPLS OAM endpoint after Ethernet endpoint
SDK-56122	763713	All	Added PORT_INIT check to all bcm_port_XXX functions to avoid their invocation before port subsystem is initialized.
SDK-56123	753886	56243_B0 56240_B0 56243_A0 56242_A0 56242_B0	Enabled OAM endpoint addition and deletion multiple times without any error

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56128		56840_A0 56850_A2	In earlier releases, <code>mac_xl_egress_queue_drain()</code> blindly adds the PORT back to <code>EPC_LINK_BMAP</code> regardless previous <code>EPC_LINK_BMAP</code> state. This has been resolved.
SDK-56140 SDK-70338	766375	56640_A1	Problem: When external TCAM table size is configured for IPv4 routes, IPv6 routes were not getting properly programmed/getting hit in the internal TCAM. Solution: Added support for this specific case of having all IPv4 routes on external TCAM and all IPv6 routes on internal TCAM.
SDK-56142	765705	56850_A0 56850_A2	In the previous release, customer reported that the rate is not accurate after changing rate from VERY HIGH PPS to low PPS. This issue had been solved by adding condicision while in <code>_bcm_trx_rate_meter_portmode_set()</code> , while adding dlf value, not need to refer to previous setting in register/memory.
SDK-56154		56640_A0 56544_A0 56542_A0 56541_A0 56540_A0 56524_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0 56524_B0 56540_B0 56541_B0 56544_B0 56542_B0	Enhancement:- There are 16 FP physical tcam slices with 512 entries per slice. There are 8 physical FP meter pools with 1024 entries per meter pool. Currently only 8 physical tcam slices are allowed to access the 8 FP meter pools. Requirement was to ensure the 16 physical tcam slices are allowed to access the 8 FP meter pools Support :- The 8 FP Physical meter pools are split into 16 logical meter pools so that the 16 FP tcam slices can attach to the 16 Logical FP meter pools.
SDK-56160	766445	56850_A0	In previous releases, L2 polling thread can process a MAC address insert/delete/move within a bucket, but it cannot process the scenario that a MAC address move from a bucket in a bank to another bucket in another bank. In this release, processing the scenario that a MAC address move from a bucket in a bank to another bucket in another bank has been added in L2 polling thread.
SDK-56189		88650_B0 88660_A0	Required changes in SDK in order to support KBP-SDK 1.2.3 and higher. The changes include configuration of a newly used instruction and its transport layer implementation.
SDK-56190	767623	56850_A0 56850_A1 56850_A2	In previous release, <code>bcm_l3_route_add</code> API may returned <code>Not_Found</code> if with an IPv6 <code>VRF_GLOBAL</code> route entry in ALPM mode even if ALPM memory table had enough space. In this release, it can be added successfully.
SDK-56193		88650_A0 88650_B0 88650_B1 88660_A0	COSQ: CNM profile can be allocated by calling <code>bcm_cosq_qcn_config_set</code> . Fixed an error of CNM profile initialization. Corrected the number of entries be added to template init ID.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56194		88650_A0	In Warmboot module, asserts were recently inserted to avoid modifying an uninitialized SW database. These asserts are replaced by regular error mechanism. Thus, most set/get SW DB functions are modified to return also an error value.
SDK-56195		56850_A0 56850_A1 56850_A2	In the previous release, PORT table LOCK would not be released if gport validation failed when operating PORT table. In this release, PORT table LOCK will be successfully released if gport validation fails when operating PORT table.
SDK-56199		88660_A0	OAM: when calling <code>bcm_oam_loss_get()</code> the near/far fields returned were mixed up.
SDK-56203		88650_A0 88650_B0 88650_B1 88660_A0	<p>XLPORT Overrun/Underrun Workaround ----- -----: The Arad driver implements a sequence to recognize and recover the port from XLPORT Overrun/Underrun issue (see BCM88650 errata sheet). To activate the sequence during device init use the following soc property: <code>custom_feature_nif_recovery_enable=1</code> (default is disabled on 6.3.x, and enabled on 6.4.x).</p> <p>The sequence might perform several iterations when trying to recover the port. To limit number of iteration use the following SoC property: <code>custom_feature_nif_recovery_iter</code> (default is 3). Note that from lab experience the port is recover within single iteration.</p> <p>Limitations: 1. The SW WA works for XLP0 only. 2. The SW WA is called during init and isn't available for dynamic port.</p>
SDK-56215	754083	All 56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56640_A0 56850_A0 56843_B0 56841_A3 56846_A1 56841_B0 56640_A1 56640_B0 56850_A1 56850_A2	In the previous release, <code>bcm_cosq_port_mapping_set</code> and <code>bcm_cosq_mapping_set</code> returned <code>BCM_E_RESOURCE</code> incorrectly when there was one unused profile of the <code>COS_MAP</code> table on Trident/Trident2/Triumph3. In this release, this issue has been addressed by setting the <code>MC_COS1f</code> and <code>UC_COS1f</code> of the <code>COS_MAP</code> table at the same time.
SDK-56222	767209	56846_A0	<p>During warm boot upgrade from SDK 6.2.9 to SDK 6.3.3, data qualifiers (UDFs) are not recovered in field module as there is a mismatch between field qualifier count (<code>bcmFieldQualifyCount</code>) in 6.2.9 and 6.3.3. The more field qualifiers are added in SDK 6.3.3.</p> <p>The issue is fixed in SDK 6.3.8 by storing <code>bcmFieldQualifyCount</code> in scache and by mapping the recovered field qualifier Id to the appropriate data qualifier.</p>
SDK-56225	767847	88650_A0	E2E scheduler port shaper is limited from below. Added fix such that in case requested rate is lower than allowed, the minimal rate will be set.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56244	765693	56840_A0 56850_A2	The guideline for <code>bcm_cosq_gport_mapping_set</code> is improved in this release by specifying that it can be used on chips which support ETS(Enhanced Transmission Selection) feature regardless of that the ETS mode is enabled.
SDK-56249		88650_B0	Egress shaper for ILKN interfaces don't work properly, causing unexpected behavior(wrong rates). The shaping for ILKN interface is set using: <code>bcm_cosq_gport_handle_get(0,bcmCosqGportTypeLocalPort,gport_info);</code> Fixed!
SDK-56253 SDK-1417	768344	84328_B0	Issue Reported: G40 Port Disable not working as expected Fix: Register sequence is modified to fix this issue.
SDK-56254	765972	88650_A0 88650_B0 88650_B1	OAM: when creating a MIP and calling <code>bcm_oam_action_set()</code> for that MIP, the profile was handled incorrectly and might have caused packet drop. This issue was fixed and resources are properly managed and freed.
SDK-56272	767442	56340_A0	In earlier releases, Helix 4 GS GE48 was using incorrect lane information in the PHY driver. The port affected using quad smgii was using lane 2 and not lane 0, This has been fixed.
SDK-56280		All	BMW CPU platform was removed from SW and SQA nightly builds and tests 2 years ago. BMW is not built or validated on any recent SDK releases in either SDK-6.3.x or SDK-6.4.x train. The last SW release that supported BMW was SDK-6.2.0 on Aug 16, 2012. BMW binaries were removed starting with SDK-6.2.1 (Oct 22, 2012) onward. By the time of SDK-6.4.2 release, it will have been exactly 2 years since the platform was discontinued in official releases. The problem is that SDK Platform guide has not been updated in the longest time. Now that we are about to publish an updated version together with SDK-6.4.1 release, this JIRA will update the document to match what has been published on DocSAFE. Releases posted on DocSAFE did not include BMW binaries for 2 years now.
SDK-56291	768458	All	The definitions of <code>COUNTER_ATOMIC_BEGIN/END</code> in <code>COUNTER</code> thread adopted <code>sal_splhi/sal_spl</code> as mutex lock to protect some small critical sections, which can cause a considerable performance loss due to its overhead and coverage scope. Replacing the old one with a new lock mechanism, the <code>sal_spinlock</code> primitives can be more efficient especially for protecting small critical sections somewhere like in <code>COUNTER</code> thread. <code>sal_spinlock</code> can be used in Linux user space, Linux kernel and vxworks, even in interrupt context. To be noted, it can't be used recursively.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56295		88650_A0 88650_B0 88660_A0	BFD accelerated endpoint that is handled in remote gport - SW DB is not restored correctly after WB.
SDK-56306	769032	88030_A0	note.
SDK-56317		56846_A0 56846_A1	In previous releases, created multipaths more than max capacity could corrupt existing ECMP groups and return wrong value -1 if ECMP group size of TD+ configured to 256 as TD device. In this release, it returns BCM_E_FULL (-6) if creating ECMP multipaths more than max capacity.
SDK-56340	755455	88650_A0	Port enable sequence was fixed to support 1588 on 1G ports.
SDK-56350		88650_A0 88650_B0 88650_B1 88660_A0	<p>The "multiple packet dequeue" feature which is meant for usage in low latency credit request profiles can now be configured using the bcm_cosq_delay_tolerance_level_set/get APIs. The feature is activated for a credit request profile if the following new flag is used in the flags field of the structure: BCM_COSQ_DELAY_TOLERANCE_IS_LOW_LATENCY.</p> <p>In release 6.4.1 all the credit request profiles named</p> <p>BCM_COSQ_DELAY_TOLERANCE*_LOW_DELAY will have this feature set. In 6.3.* releases the default profiles are not changed, though this can be done manually. Example of changing one predefined profile manually:</p> <pre>bcm_cosq_delay_tolerance_level_get(unit, BCM_COSQ_DELAY_TOLERANCE_200G_LO W_DELAY, &delay_tolerance); delay_tolerance.flags = BCM_COSQ_DELAY_TOLERANCE_IS_LOW_ LATENCY; bcm_cosq_delay_tolerance_level_set(unit, BCM_COSQ_DELAY_TOLERANCE_200G_LO W_DELAY</pre>
SDK-56352		88660_A0	Fixed ECN (Explicit Congestion Notification) to work correctly in 88660
SDK-56353 SDK-56332	768573	88650_A0 88650_B0 88660_A0	<p>In Policer rate computation function, the exponent and mantissa configuration was fixed in case the required value is too small.</p> <p>When allocating a meter with a very low rate (for instance when using</p> <pre>bcm_policer_config_t.max_pkbits_ sec = 128), the driver produces an error, even though this is a valid rate. This is now fixed.</pre>
SDK-56355	767767	88660_A0	In L2 module, when working in centralized mode, the LIF-valid bit entry was not received correctly on learn events (i.e., when the CPU was inserting learnt entries via BCM SDK). The LIF-valid bit is now set correctly on the learn events and matches the payload of the device learned entry.
SDK-56379		All	Support has been added for resolving the modern GPORT types (TRILL, VXLAN, NIV, L2GRE, etc.) in the Diag Shell.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56387	769040	56450_B0 56450_A0	Support has been added for IFP qualifier <code>bcmFieldQualifyRouterAlertLabelValid</code> for BCM5645x devices.
SDK-56409		All	PTP clocks can now be re-created in order to change the "immutable" clock parameters. The only restriction is that the number of clock ports on re-creation may not exceed the value used on the initial creation.
SDK-56410	769158	88650_A0	TCDP mapping using <code>bcm_cosq_gport_egress_map_set</code> uses profiles of mapping, and associate each port to relevant profile. The API supported up to 4 different profiles, although HW support up to 8 profiles (when new profile is required but not available, the API return an error). The API was fixed to support 8 profiles as the HW.
SDK-56425	767797	88650_A0 88660_A0	SER interrupts were not signaled to CPU and not counted, due to being masked by a set of override bits called monitor bits. This was fixed to allow proper logging and handling of SER interrupt events by the SDK.
SDK-56439		88650_A0 88650_B0 88660_A0	Ethernet OAM does not recover from Warm-boot. This is fixed.
SDK-56440		88650_A0 88660_A0	MPLS Tunnel initiator clear all API does not clean up MPLS WB information as well.
SDK-56441		88650_A0 88660_A0	During Warmboot in vswitch module, the VSI MSTP was always restored, even if it was cleared before the warmboot. This restoration is skipped upon Warmboot.
SDK-56446	759287	88650_A0 88660_A0	Fix <code>low_vid_verify</code> value in <code>bcm_vswitch_port_delete</code> function (<code>arad_pp_frwrdr_trill.c</code>).
SDK-56447	763576	88650_A0 88660_A0	When creating an ECMP group using <code>bcm_l3_egress_ecmp_create</code> , if the 'ecmp' parameter is NULL, a segmentation fault was occurring. This is now fixed - the software checks that the 'ecmp' parameter is not NULL.
SDK-56451		88650_B0 88660_A0	Required changes in SDK in order to support KBP-SDK 1.2.3 for external TCAM are introduced.
SDK-56452	760578	56450_B0 56450_A0	When 1 + 1 protection switching is enabled/disabled (with label swapping on IPMC group), the <code>MPLS::LABEL_ACTION_SWAP</code> field of <code>EGR_L3_NEXT_HOP</code> table need to be set/cleared respectively to achieve the functionality. This support has now been added.
SDK-56455 SDK-56327	769233	56224_B0 56224_A0	Issue :- IpType Qualifier was not recovered properly after warmboot. Fix :- Recovery of IpType Qualifier was not handled properly in BCM56624. Added Code to recover IpType qualifier after warmboot.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56464	765386	56640_A0 56640_A1 56640_B0	<p>Problem: When there is no signature configured, if there is traffic being sent to signature matching engine(SME) for deep packet inspection, the SME goes into a hung state. Even after configuring some signatures later on, it doesn't indicate any match though it receives matching traffic.</p> <p>Solution: Do not let any traffic to be forwarded to SME until at least one signature is configured successfully. Also, stop the traffic from being forwarded to SME while detaching the last active engine. This is achieved by modifying flow tracker configuration register field.</p>
SDK-56476		88650_A0 88660_A0	In Field Processor's diagnostics, in case field groups were created however no entry was inserted, an error is produced when calculating the number of entries from an empty bitmap. This is fixed.
SDK-56482	768774	56450_B0 56450_A0	<p>Added support for associating a MPLS label to a given protection switching group for BCM5645x devices. API</p> <p><code>bcm_mpls_tunnel_switch_add()</code> can be used to achieve this by passing the protection switching group id in "failover_id" member of structure "bcm_mpls_tunnel_switch_t".</p>
SDK-56492	769633	56850_A0 56850_A1 56850_A2	The related EGR_PORT_TO_NHI_MAPPING was not cleaned when the last port was removed from the trunk where a VXLAN logical port is created. Now it is fixed by adding the specific implementation for VXLAN.
SDK-56495	768732	88650_A0 88650_B0 88650_B1 88660_A0	In Field Processor, at Egress, the support of two new qualifiers is introduced: <code>bcmFieldQualifySid</code> (MAC-in-MAC I-SID) and <code>bcmFieldQualifyMplsForwardingLabelAction</code> . Both qualifiers are mapped internally to the EEI value.
SDK-56507		56640_A0 56640_A1 56640_B0	<p>Previously</p> <p><code>bcm_cosq_port_bandwidth_set()</code> failed on 56640 HSP ports. HSP ports handling for <code>bandwidth_set</code> API has now been added.</p>
SDK-56514		56850_A0 56854_B0	In previous releases, SER correction for MMU CTR block was not implemented. In this release, MMU CTR block SER correction logic has been implemented. Once parity error is detected in tables in MMU CTR block, the corrupted table entry will be cleared.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56522		56640_A0 56850_A0	Issue here is SDK support is missing to recover EFP Secondary selectors during Warm Boot. So EFP qualifiers which need secondary selectors will fail to be recovered in Warm Boot. Now support has been added for recovering all EFP secondary selectors come from following different registers 1. EFP_CLASSID_SELECTORr - (HX4, KT2, TR3, TD2) 2. EFP_KEY4_DVP_SELECTORr - (HX4, KT2, TR3, TD2) 3. EFP_KEY4_MDL_SELECTORr - (HX4, KT2, TR3, TD2, GreyHound, Enduro) So this fix is applicable to HX4, KT2, TR3, TD2, GreyHound, Enduro
SDK-56533	769718	56850_A2	Fixed multicast module to return error when deleting member from a MC group that was already destroyed.
SDK-56554	770975	56850_A0 56850_A1 56850_A2	Support has been added for the ability to transmit even if port is down.
SDK-56572	771276	88660_A0	When using external TCAM for forwarding, serial IP and RPF, then high rate lookups return sometimes wrong results. This is fixed: serial lookups in external TCAM are always returning reliable results at any supported rate.
SDK-56577		88650_A0 88660_A0	Removing sw database MC-ID -> nickname. Nickname can be extracted from trill_port_id database (encap_id field).
SDK-56578		88650_A0 88660_A0	New sequence for ECMP creation using forward-group port instead of trill-port-ecmp.
SDK-56580	772058	88650_B1 88660_A0	QOS: Fixed the ability to set Inner-PCP to TC/DP table in bcm_qos_map_add.
SDK-56581		88650_A0	In Field Processor diagnostics, the actions offsets are incorrect when cascaded action is used. This is fixed.
SDK-56591	768899	56850_A0 56850_A1 56850_A2	New API bcm_l3_egress_stat_counter_sync_get() added to retrieve l3 egress stats after updating the software copy of the counter value with the hardware counter value.
SDK-56594 SDK-57957	769099	56440_A0 56445_A0 56440_A1 56445_A1 56444_A1 56450_A0 56340_A0 56449_B0 56445_B0 56440_B0 56447_B0 56443_B0 56441_B0 56446_B0 56448_B0 56344_A0 56342_A0 56442_B0 56342M_A0 56340M_A0 56455_A0 56456_A0 56450_B0	In earlier releases scheduler configuration with weight value 0 was incorrectly configured in WRR mode. This has been fixed in this release to configure scheduler configuration with weight value 0 to be in STRICT PRIORITY mode.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56597	772109	56850_A2	soc_alpm_insert: Route Insertion Failed due to DEFIP AUX Operation timeout. On expiry of poll for ALPM hardware operations, soc_timeout_check requires that the status register needs to be read one more time to confirm operation has completed. This support has been added.
SDK-56607		88650_B0	Fix initial shaper to interface mapping. This fix has no functional impact.
SDK-56608	765207	56450_A0 56450_B0	When the physical port associated with MPLS port is replaced by using bcm_mpls_port_add() API with flag BCM_MPLS_PORT_REPLACE, the properties associated with old physical port is not cleared. Appropriate check has been added to clear the properties associated with the old physical port for BCM5645X devices.
SDK-56610	772885	56450_A0 56450_B0	gport_attach function can be called passing cosq value as 0,1,2.. so on. When coaq value was passed instead of -1, code was not handling it properly, So same hw_index was allocated again and again, Now checks are provided so that unique hw_cosq value is assigned for different values of cos
SDK-56611	772970	88650_A0 88650_B0 88660_A0	After Hard_Reset was called , CPU port was stuck. Resolved in the hard reset code by resetting CMIC TXi credits.
SDK-56615	772971	56450_A0 56450_B0	WRR scheduling under sub ports could not work due to missing weight configuration in L0 nodes. This configuration issue has been corrected to get the expected scheduling behavior.
SDK-56628		88660_A0	BFD: for BFD endpoints of type bcmBFD TunnelTypeMpls (BFD PDUs are encapsulated by UDP, IP, MPLS, Eth), IP TOS, TTL may be configurable through the fields ip_tos, ip_ttl. Note that the protocol dictates that the IP TTL be set to 1.
SDK-56629		88650_A0 88650_B1 88660_A0	When compiling with INCLUDE_KBP compilation flag, a large memory allocation for Field Processor software state was performed, related to external TCAM. This large memory allocation is now performed only if ELK usage is indicated via SOC properties.
SDK-56635		88650_A0 88650_B0 88660_A0	In some scenarios, trunk ports lb_key_min and lb_key_max values do not cover all lb_key_range [0:255] which results in packet drop. This issue is fixed.
SDK-56636		88650_A0	At SOC layer, a new mechanism to improve the performance of entry insertion for Large-Exact-Match, Small-Exact-Match and TCAM databases has been implemented. By default, this mechanism is enabled. To disable this mechanism, unset the compilation flag ARAD_FAST_REGISTERS_AND_FIELDS_ACCESS.
SDK-56641		56850_A2	In earlier releases, VxLAN multicast was treated as non-Layer3 multicast. It caused VxLAN multicast group to still have members after being re-created. This has been resolved.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56644		56440_B0	<p>EGR_L3_NEXT_HOP table has overlapping views, for ex: L3, MPLS, SD_TAG etc., and for a given entry in EGR_L3_NEXT_HOP table the fields corresponding to a particular view, decided by ENTRY_TYPE field, should only be modified. But some fields of L3 view (overlapping with MAC_DA_PROFILE_INDEX field of MPLS view) were always getting modified resulting in wrong EGR_MAC_DA_PROFILE entry getting overwritten when a given L3 egress object is updated using the flags BCM_L3_REPLACE BCM_L3_WITH_ID.</p> <p>Protection (i.e, check for appropriate ENTRY_TYPE value before modifying fields in L3 view) has been added to overcome the issue.</p>
SDK-56646		88650_A0 88660_A0 88670_A0	<p>Fixed a problem in bcm_mpls_port_add. The issue caused the driver to crash with a segmentation fault when the API is called with the REPLACE flag.</p>
SDK-56647		88650_A0 88650_B0 88660_A0	<p>In FCoE, when adding a route via bcm_fcoe_route_add API with flags BCM_FCOE_LOCAL_ADDRESS BCM_FCOE_HOST_ROUTE, the entry was not be added correctly to the forwarding database. This is fixed.</p>
SDK-56649	772044	88660_A0	<p>In metering when the SOC property policer_color_resolution_mode is set to 1, the meter processor outputs the following DP values: green - 0 yellow - 1 meter processor red - 2 ethernet policer red - 3.</p> <p>Due to a software bug, when the meter processor gave a packet a color of yellow, the actual DP would be 2, instead of 1. This is now fixed.</p>
SDK-56657 SDK-54730		88660_A0	<p>Currently, unless specified by SOC property, ethernet policers drop all packets that arrive red to the device. Color blind ethernet policers allow to do rate policing even for packets that arrive red to the device.</p> <p>This fix introduces the ability to change ethernet policers to be color blind or color aware dynamically. To set color blind ethernet policing, both the ethernet policer and aggregate policer associated with a port and traffic class must be set to be color blind. To set an ethernet policer to be color blind, the BCM_RATE_COLOR_BLIND flag can be used when calling bcm_rate_bandwidth_set. To set an aggregate policer to be color blind, the BCM_POLICER_COLOR_BLIND flag can be used when calling bcm_policer_set with an aggregate policer.</p>
SDK-56688		56340_A0	<p>In the previous release the packet/byte fields were not working correctly in regex reports. The packet and Byte counter registry values are now retrieved and updated in the match reports.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56691		88650_A0 88650_B0	In Rx parsing, the <code>src_gport</code> and <code>dst_gport</code> meaning were unclear. It is fixed to: - <code>src_gport</code> : incoming port where the packet was introduced into Arad - <code>dst_gport</code> : outgoing port where the packet exits from Arad
SDK-56693		56340_A0	When only engine 0 is enable, the CSF table will not be updated no toggling on CSF valid signal. When other engine are enabled, the CSF table will be loaded when the valid signal of other engines toggle. Hence enabling starts from engine 1 and engine 0 will be enabled at last.
SDK-56700	774184	88650_A0 88650_B0 88650_B1 88660_A0 88670_A0	When calling <code>bcm_mpls_port_add</code> with <code>pwe id > 32K</code> , error printouts are provided but the API returns <code>BCM_E_NONE</code> . This is fixed and error is returned.
SDK-56701	773800	All	In earlier releases diag shell would intermittently crash in "l3 egress show" command. This has been resolved.
SDK-56709	773764	56334_B0 56334_A0	Issue: ==== Remote trunk identifier bit has to be ignored while setting the <code>srcTrunk</code> mask. The MSB of the <code>modId</code> represents Remote trunk bit and hence it was calculated based on the width of the qualifier. However, the width of the qualifier varies for different devices. Due to this, for devices like Enduro, the bit was positioned wrongly and was ignoring trunk bit instead of the remote trunk bit. Fix: === Instead of using the width of qualifier, the bit position of the <code>trunkBit</code> minus 1 (<code>trunk_bit_pos - 1</code>) is used to calculate the remote trunk bit position and ignoring the bit by masking the bit to 0.
SDK-56714	758491	56450_A0 56440_B0	Issue : ----- In katana and katana2 the <code>rqe_port_config</code> register was programmed with <code>cos_mode=1</code> when extended queueing was enabled but <code>cos_mode=1</code> is not valid for this register and this causes traffic to go through cos 0 always Fix ---- If extended queueing is enabled then we program <code>cos_mode</code> with value 0 in <code>RQE_PORT_CONFIG</code> .
SDK-56720	769698	56224_B0 56224_A0	Output of "trunk show" command displays the port names correctly by resolving the gports for all XGS devices.
SDK-56725		56850_A0 56855_A0 56850_A2	In previous release, the functions <code>bcm_vxlan_stat_attach</code> and <code>bcm_vxlan_stat_counter_get</code> took high execution time, about 13000 usec per call, which couldn't meet customer expectations. In this release, the functions have been optimized, and they takes about 100 usec per call. The performance has been improved.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56741	763657	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56340_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0 56344_A0 56342_A0 56342M_A0 56340M_A0	In earlier releases, the validation on the PORT for ETS was incorrect on the return value which might lead to wrong COSQ mapping. Fixed the validation.
SDK-56753 SDK-46271		56640_A0 56643_A0 56640_A1 56643_A1 56640_B0 56643_B0	Problem: Due to a hardware bug, the hardware team had recommended to disable bus parity protection for a bunch of memories which includes IESMIF. However, the SDK still has the bus parity enabled on IESMIF and this is causing spurious parity errors in the cases where ESM accesses are involved. Solution: Disable the bus parity protection for IESMIF by default, to workaround the hardware issue.
SDK-56756	773877	56540_A0 56540_B0	Previously, "l3 ip6route show" command was broken on Firebolt-4. This is due to that <code>soc_feature_l3_shared_defip_table</code> is not supported on Firebolt-4 and thus <code>bcm_switch_object_count_get</code> called in this command returns an error. It is fixed by adding the additional check on <code>soc_feature_l3_shared_defip_table</code> to avoid calling <code>bcm_switch_object_count_get</code> for Firebolt-4.
SDK-56761		56540_A0 56340_A0 56540_B0	In Apollo2 and Helix4 devices, during an OAM CCM timeout event, remote endpoint index passed from SDK to OAM event callback function was not correct, this issue has been addressed.
SDK-56763	772471	56850_A0 56850_A1 56850_A2	In the previous release, the API <code>bcm_cosq_gport_bandwidth_set</code> would set the shaper on a wrong scheduler node. In this release, this issue has been addressed by setting the software resources which have been assigned to the HSP ports.
SDK-56765		88660_A0	Add driver support to new Arad SKU - 88363
SDK-56770	774767	88650_B0	Trill learning: In TRILL multicast, ingress learning, MACT learning is disabled at Egress Router-Bridge for TRILL multicast packets otherwise unrelated MACs (Link Layer SA) are learned.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56779	774862	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In earlier release, TD2 had 48 HIGIG trunks and SDK was not able to record/maintain the bitmap of higig trunk override id which was larger than 31. This issue has been fixed in this release.
SDK-56781	774909	56850_A0 56850_A1 56850_A2	The root cause of this issue is that when L2 addresses are learnt on Y pipeline, the hit bits of the corresponding L2X table entries are not set. But <code>bcm_l2_matched_traverse</code> function will read all L2X table entries and check the hit bits, so the traverse function can't find the L2 address learnt on Y pipeline. Code has been added to update the hit bits of L2X entries when the L2 addresses are learnt on Y pipe line.
SDK-56786	773228	All	Support has been added for displaying counter register's alias name which register's name larger than 13 characters
SDK-56789		56340_A0 56547_A0	For 802.1AS packet, RX and TX timestamps are now enabled for BCM56340 family of devices
SDK-56801	774468	56440_A0 56445_A0 56440_A1 56450_A0 56440_B0 56450_B0	In earlier releases, Enabling of <code>tcam_protect_write</code> resulted in incorrect computation of the number of entries per slice on Katana. This issue has been fixed by correcting size of the FP TCAM value used for computing the size of each slice.
SDK-56805		88660_A0	ARP downstream checking didn't use separately, now the fixes resolve this issue.
SDK-56821		56820_B0	In Scorpion, IP Packets with 0x9100 (other than default TPID 0x8100) outer tag are treated as untagged and non IP packets when these packets ingress on YPIPE and egress on XPIPE. This behavior is detected by the EFP when it is configured to match anything beyond the L2 header. The software work around is added in SDK to fix this issue by changing the access type of the per-port register <code>EGR_SRC_PORT</code> .
SDK-56840	772939	88650_A0 88650_B0 88650_B1 88660_A0	In MAC-in-MAC, when using API <code>bcm_l2_addr_add()</code> , multicast group destination was not supported in BMACT Forwarding table. Multicast group destination is now supported and can be added to BMACT forwarding table.
SDK-56848	776418	82328_A0	Added PHY BCM82322 support. This PHY supports 10G,20G and 40G modes
SDK-56850	776440	56450_A0 56450_B0	Issue : Support for ECAP CopytoCpu is missing on KT2. Fix : Added Support for ECAP CopytoCPU in KT2 in SDK
SDK-56854		88650_A0 88650_B0 88660_A0	In FCoE zoning, when adding an entry, all entry actions were applicable (allow, deny, redirect), but the same action (allow) was always executed. A validation is introduced so that only the action allow is applicable.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56869		56450_A0 56450_B0 56640_B0 56440_B0	<p>PTP master sends Announce messages to PTP slaves. PTP slaves compares announce messages received from several PTP master to decide which PTP master to choose to synchronize time. Following three new fields are added to PTP master information structure.</p> <p>ClockAccuracy: The clockAccuracy indicates the expected accuracy of a clock when it becomes grandmaster or in event it becomes grandmaster. Various granularities are possible. This specifies time is accurate to within 25 ns/100 ns/250 ns/1 ?s/2.5 ?s/10 ?s etc.</p> <p>OffsetScaledLogVariance: The offsetScaledLogVariance indicates inherent precision of a clock. This is the precision of the timestamps included in message issued by clock when it is not synchronzied to another clock using the protocol. The reference clock when not synchronized to another clock may be an atomic clock, a GPS receiver, a stable local oscillator, a suite of clocks synchronized via NTP, etc. These sources may contribute to the variance estimate. The value of offsetScaledLogVariance can also be a staic constant determined by manufacturer.</p> <p>StepsRemoved: The distance measured by the number of boundary clocks between the local clock and the foreign master is used when two Announce messages reflect the same foreign master. The distance is indicated in the stepsRemoved field of Announce messages.</p>
SDK-56876	776002	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	<p>The scheduler configuration with weights value 0 will be considered as STRICT_PRIORITY. Fixed the same behavior in SDK.</p>
SDK-56878	776733	88650_B1	<p>The bcm_cosq_control_set/get(unit, 0, 0, bcmCosqControlAdmissionTestProfileA, bitmap) APIs did not work correctly if the bitmap bits for PFC or LLFC VSQ types were set. This was fixed.</p>
SDK-56884		88650_A0 88650_B0	<p>MIM: DEFAULT BEHAVIOR CHANGE . Encoding of returned handler station_id for MIM is now changed in l2 station APIs. bcm_l2_station_get() API failed in some cases when LSB for MyMac was considered to be global instead of per ingress port. This happened when the MIM global LSB bit in the created station_id was wrongfully set due to an overlap in the station_id encoding. This is fixed by changing the encoding of the station_id so that there is no overlap with the MIM LSB global indication bit. The MIM global LSB indication bit in station_id changed from bit 7 to bit 16.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56887		88660_A0	Default value of Chicken bit EGQ_CFG_BUG_FIX_CHICKEN_BITS_REG_1_CFG_BUG_FIX_87_DISABLE was changed to disable (instead of enable) as it doesn't provide any new functionality.
SDK-56888 SDK-56945	742236	88650_A0 88650_B0 88650_B1 88660_A0	Support reflector functionality in accordance with RFC-2544 (benchmarking methodology). This JIRA contains IP+MAC swap functionality (swap the SIP with the DIP, SA with DA) as well as a light MAC-only-swap functionality (swap the SA with the DA). For the former, the soc property RFC2544_reflector_mac_and_ip_swap_port should be set to the reflector port. All packets arriving at the ETPP with the Out-TM-port set to the reflector port will have their MAC addresses and IP addresses swapped, and the packet will be prepended with a PTCH with the SSP set to the original Out-PP-Port. The reflector port should be defined as a recycle port and the IP routing should be done at the second pass. The light MAC-only swap functionality can be used analogously with the soc property RFC2544_reflector_mac_swap_port For a more detailed account (For example setting an egress-PMF rule modifying the Out-TM-port), refer to cint_benchmarking_methodology.c
SDK-56903		56850_A0	Adding a flag BCM_NIV_VNTAG_L_BIT_FORCE_1 to choose if frames can be headed back towards the Interface Virtualizer that it originated from.
SDK-56913	759274	All	In earlier releases on overflow the DMA timeout/overflow stat was cleared only when the entries were available. On entry empty this was not getting cleared. Fixed in the changes when the entry is empty.
SDK-56917	777278	56340_A0	SME match not reported for twitter and webex signatures due to hex representation of ASCII. Provided support in SDK API <code>bcm_regex_match_set()</code> to parse hex representation of ASCII Alphabets.
SDK-56925		88650_A0 88650_B0 88650_B1 88660_A0	PON: In previous release, DHCP IPv6 anti-spoofing wasn't working when soc property <code>l3_source_bind_mode</code> is IPV6, now fixed this issue.
SDK-56929		56850_A0 56850_A1 56850_A2	In earlier releases, next hop information was not initialized before using it. This has been resolved.
SDK-56931		56850_A0 56850_A1 56850_A2	In previous releases, the API <code>bcm_l3_egress_get</code> returned <code>BCM_E_INTERNAL</code> in vxlan case. A new case <code>_bcmVpTypeVxlan</code> has been added to fix this issue. Now if the case is vxlan, the <code>egr->port</code> will be set to vxlan and the API will return <code>BCM_E_NONE</code> .
SDK-56954		56850_A0	In earlier releases, source trunk table was not being cleared up if a customer used an incorrect sequence. This has been resolved.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56956	774358	88650_A0 88650_B0 88660_A0	In Field Processor, in Direct Extraction field groups, it is possible to set a bias value as part of the extraction field configuration. The procedure failed when the bias value is negative. This is fixed.
SDK-56960		88650_A0 88650ACP_A0 88650_B0 88650_B1	QOS: Qos map id can be destroyed by calling <code>bcm_qos_map_destroy</code> . Improvement in entry deletion for <code>bcm_qos_map_destroy</code> by adding new SW DB to record each entry is occupy or not.
SDK-56961		88660_A0	BFD: When calling <code>bcm_bfd_endpoint_create()</code> with the flag <code>BCM_BFD_ENDPOINT_REPLACE</code> set and <code>type==bcmBFDtunnelTypeMplsTpCc</code> , static registers were mismanaged, causing such calls to fail.
SDK-56962	776131	88650_A0 88650_B0 88660_A0	The OAM DM DOWN program at the egress editor has been fixed so that only appropriate packets will select this program. Previously this program was catching other packets as well which caused outgoing packet corruption.
SDK-56964		56850_A1 56850_A2 56850_A0	In earlier releases the related <code>EGR_PORT_TO_NHI_MAPPING</code> was not cleaned when the last port was removed from the trunk where a VXLAN logical port was created. This is fixed by adding the specific implementation for VXLAN.
SDK-56975	774350	56850_A0 56850_A1 56850_A2	Customers requested more granularity in <code>bcm_vxlan_vpn_create</code> . To enable this modifications were made to <code>BCM_VXLAN_VPN_WITH_VPNID</code> to meet this goal. Before this change, when customer created a vpn, <code>BCM_VXLAN_VPN_WITH_VPNID</code> was required, and both VFI and VNID were created. After this change, the behavior is as follows: When create a VXLAN VPN: If use <code>BCM_VXLAN_VPN_WITH_VPNID</code> , both VFI and VNID will be created. If not, use flag <code>BCM_VXLAN_VPN_REPLACE</code> , only VFI will be created. When updating an existing VXLAN VPN (<code>BCM_VXLAN_VPN_REPLACE</code> should be used. If use both <code>BCM_VXLAN_VPN_REPLACE</code> and <code>BCM_VXLAN_VPN_WITH_VPNID</code> , both VFI and VNID will be created. If only use <code>BCM_VXLAN_VPN_REPLACE</code> , the VNID will be removed.
SDK-56980	777710	56240_B0	In previous releases, If the given port was configured with WRR scheduling and then warmboot was done the SW did not recover the correct scheduling algorithm back after the warmboot. The hardware continued to have correct value. This has been resolved.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-56988		56850_A0 56850_A1 56850_A2	Customer wanted to use VLAN and VFI flex counter simultaneously. But in the previous release, SDK assigned VLAN and VFI counter in the same pool, and this would cause the VFI counter to not be updated when the packet hit two memories. Now the customer can use the SOC property <code>ing_share_flex_counter_pool=split(vlan,vfi)</code> to prevent VLAN and VFI from sharing the same pool.
SDK-56991	778526	56850_A2	In earlier releases, when using <code>bcm_vxlan_port_add()</code> API with <code>BCM_VXLAN_PORT_REPLACE</code> flag, it will clear the flex counter configuration if this vxlan port has attached with flex counter. This has been resolved.
SDK-56994		56850_A0 56850_A1 56850_A2	It was found that network facing flex counters were not working for both <code>bcmStatGroupModeSvpType</code> and <code>bcmStatGroupModeSvpType</code> group modes. After investigation we located the RCA was the counter offsets were not set correctly in previous implementation. The issue was fixed by adjusting the counter offset for both <code>bcmStatGroupModeSvpType</code> and <code>bcmStatGroupModeDvpType</code> group modes.
SDK-56995	777713	56845_A2 56850_A0 56850_A1	In the previous release, when using RPCs and calling <code>bcm_vxlan_stat_counter_get()</code> , the values in the <code>counter_indexes[]</code> parameter are not being properly propagated from the client to the server. This has been resolved.
SDK-56999	773690	56850_A0 56850_A1 56850_A2	Added support for port extender failover.
SDK-57002	778714	56850_A2	In earlier releases, SDK code was not able to resolve the ports for which id was larger than 64 in BITMAP of <code>IFP_REDIRECTION_PROFILE</code> table on TD2 after warmboot. This has been resolved.
SDK-57004		56640_A0 56340_A0	In previous release, the schan response type for devices with ISM, e.g. Triumph3 and Helix4 is not properly checked. The following response types <code>SCHAN_GEN_RESP_L2_MOD_FIFO_FULL</code> , <code>SCHAN_GEN_RESP_MAC_LIMIT_THRESHOLD</code> and <code>SCHAN_GEN_RESP_MAC_LIMIT_DELETE</code> have been added in schan response type checking in the routine <code>soc_mem_generic_insert()</code> .
SDK-57009		56850_A0 56850_A1 56850_A2	In previous releases, <code>bcm_vxlan_stat_detach</code> took high execution time because redundant memory operation was executed. In this release, we remove memory read operation and use <code>soc_mem_write</code> instead of <code>soc_mem_write_range</code> conditionally to save time, then the execution time can be reduced a lot.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57027		56850_A0 56850_A1 56850_A2	In earlier releases, Trunk useful information was cleared by VXLAN API. This has been resolved.
SDK-57032		56850_A0 56850_A1 56850_A2	In earlier releases, <code>bcm_vxlan_port_get()</code> could not get the <code>BCM_VXLAN_PORT_DROP</code> and <code>BCM_VXLAN_PORT_MULTICAST</code> flags correctly. This has been resolved.
SDK-57034	775986	56450_A0 56242_A0	Issue : Packet based WRED profiles was not restored properly in katana/katana2 after warmboot in earlier releases .As part of fix during warmboot we scan packet based WRED table and update the software profile.
SDK-57038		88650_B1	Documentation only: <code>stat_if_pkt_size</code> description in <code>config-sand.bcm</code> example was misleading. The correct description can be found in user manual or in <code>property.h</code> . Description in <code>config-sand.bcm</code> fixed as well.
SDK-57054	778731	88650_B1	add more detail prints and update the UM. Changing jira to improvment
SDK-57075		88650_B0	Arad initialization time significantly improved for channelized interface configuration.
SDK-57077		88650_A0 88650_B0 88650_B1 88660_A0	IMPORTANT CHANGE FOR PWE P2P: OAM PWE P2P was not identified as OAM in the classifier because of wrong lif id (0 value instead of the real LIF-ID value). This is fixed by setting valid LIF-ID for PWE P2P. The change may cause same-interface to be invoked for PWE P2P case when In-LIF PWE P2P ID is equals Out-LIF ID.
SDK-57078		88650_A0 88650_B0 88650_B1 88660_A0	OAM: Supporting down MEPs in the format CFMEthoMplsoEth in OAM classifier. In order to enable initialization of the OAM TCAM to identify CFMEthoMplsoEth, set soc property <code>custom_feature_oam_downmep_pwe_classification</code> to 1. This feature supports CFM identification per-md level only. This feature does not support identification per opcode. All CFM packets will be associated with opcode=1 (CCM). Inner Ethernet frames with 0 or 1 VLAN tags preceding the CFM EtherType are supported. For a more detailed explanation (including examples), consult <code>cint_oam_cfm_o_eth_o_pwe_o_eth.c</code>
SDK-57080	766661	88650_B1	TRILL and FCoE could not be supported simultaneously on the same device, due to an overlap in FLP (i.e. forwarding HW block) programs allocation. TRILL and FCoE can now be supported and coexist on the same device.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57082		88650_A0 88660_A0	<p>Important Note: the default Drop Precedence (DP) mapping of a yellow packet with DP=2 has been changed.</p> <p>Usually the final DP (Drop Precedence) given by the meter (or the In-DP) is unchanged, and can be from 0-3. In the past the final DP was always changed from 2 to 1 when passed to ingress, meaning that the only available DP results were 0, 1 and 3 (at ingress). To support this old behavior the SOC property <code>policer_color_resolution_mode</code> is introduced. When <code>policer_color_resolution_mode=1</code>, if the final DP is 2, this DP is mapped to 1 instead (at ingress).</p>
SDK-57083	776583	88650_B0 88650_B1 88660_A0	<p>IMPORTANT: for improved performance after <code>bcm_field_group_install</code> call, it is recommended to set <code>USING_TCAM_PRIO_LIST_INVERSE_SCAN</code> compilation flag.</p> <p>In Field processor entry insertion procedure, the user can: - after initialization, define all the entries and then insert them in one call (<code>bcm_field_group_install</code>) - on-the-fly, insert the entries dynamically one by one (<code>bcm_field_entry_install</code>)</p> <p>The advantage of the first case is the absence of TCAM shuffling, since the entries are sorted according to their priority before their insertion.</p> <p>In this case, the limiting factor in the entry performance was the entry insertion in the priority sorted list, an internal data structure detailing for each priority the acceptable TCAM location range. The scanning of this list was always performed from the first node to the last one, even if in the sorted case the inserted entry was the last one. This scanning has been changed to scan from the end, if the compilation flag <code>USING_TCAM_PRIO_LIST_INVERSE_SCAN</code> is set. We highly recommend to users to set this compilation flag for performance improvement.</p>
SDK-57085		88650_A0 88660_A0	<p>If <code>bcm_mpls_tunnel_initiator_create</code> is called with <code>WITH_ID</code> flag and an existing egress tunnel id, this is illegal configuration. We added a check to verify this won't happen.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57100	778739	56850_A0 56850_A1 56850_A2	In Trident2, IP_FRAG_INFO(2bit) is defined in 5 field selectors (F1_6, F1_15, F2_1, F3_3 and IFP_PAIRING_FIXED). But In SDK, IP_FRAG_INFO in these 5 different selectors are initialized with 2 different qualifiers as below which is wrong. Modify the SDK to make it consistent i.e use bcmFieldQualifyIpFrag qualifier at all places. F1_6 - initialized for bcmFieldQualifyIpInfo F1_15 - initialized for bcmFieldQualifyIpFrag F2_1 - initialized for bcmFieldQualifyIpFrag F3_3 - initialized for bcmFieldQualifyIpInfo IFP_PAIRING_FIXED - initialized for bcmFieldQualifyIpFrag Now IP_CHECKSUM_OK is 1 bit field and part of FIXED part of IngressFieldProcessor key. Currently SDK does not have support for this 1 bit field and bcmFieldQualifyIpInfo qualifier is used to initialize IP_CHECKSUM_OK bit.
SDK-57102	779185	56850_A0 56850_A1 56850_A2	In earlier releases, If adding I3 host entry to HW failed, SDK should decrease the related reference count but this function did not work when it has a multipath flag. This has been resolved.
SDK-57104	779184	56526_A0 56524_A0 56521_A0 56526_B0 56524_B0	For BCM_5652x devices, whenever a tpid other than the default tpid was created, reference count of default tpid was decremented once but was not incremented during deletion. During repeated creation and deletion, this reference count became negative resulting in error. This has been fixed by incrementing default tpid reference count upon deletion of tpid thus providing support for repeated creation and deletion of tpid on a port
SDK-57105		56850_A2	The customer requested configuration of RTAG7_HASH_CONTROL_4.VXLAN_PAYLOAD_HASH_SELECT A/B to meet their hash requirement. For Trident2 and subsequent XGS devices, 2 switch controls bcmSwitchHashVxlanPayloadSelect0 and bcmSwitchHashVxlanPayloadSelect1 have been provided to support the requirement.
SDK-57107		56850_A2	The customer requested configuration of RTAG7_HASH_CONTROL_4.VXLAN_PAYLOAD_HASH_SELECT A/B to meet their hash requirement. For Trident2 and subsequent XGS device, 2 switch controls bcmSwitchHashVxlanPayloadSelect0 and bcmSwitchHashVxlanPayloadSelect1 have been provided to support the requirement.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57123		56850_A0 56850_A1 56850_A2	Issue:- bcmSwitchL3Max128BV6Entries switch control setting caused assertion failed message due to array index overflow in array defip_tcam_log_index and defip_tcam_urpf_log_index of SOC_CONTROL. Fix:- Modified the soc_trident2_mem_config function to make sure l3_defip_index_remap won't exceed the physical size, and the arrays can be initialized after that.
SDK-57132	757170	88650_B1 88660_A0 88670_A0	OAM: Packets trapped by the OAM classifier with an incorrect level by an up-MEP will include two sets of system headers. The inner set will include the DSP, SSP on the FTMH, as well as a PPH and FHEI, the outer set will include a FHEI with the CPU-Trap-Code field set to 0xa2 (bcmRxTrapOamLevel). Similarly for packets that arrive at an up-MEP from the passive side. In this case the CPU-Trap-Code on the outer FHEI will be 0xac (bcmRxTrapOamPassive) and the inner set of system headers will be as above. This behavior may be enabled by setting the soc property "custom_feature_oam_additional_FTMH_on_error_packets" to 1.
SDK-57133	748626	88650_A0 88660_A0	When ilkn_tdm_dedicated_queuing feature is enabled, non-TDM ports can't reach wire speed. (blocked in ~60G). Fixed.
SDK-57141	779921	56840_A0 56850_A2	Problem: lpbm mask setting was missing during field entry movement, which gets called when a higher priority field entry is installed. Solution: lpbm mask was set properly during field entry movement for Trident Series of devices.
SDK-57164	763730	56440_A0 56450_B0	In the earlier release the tag information derivation was incorrect for PPD_TYPE=2, for PPD_TYPE=2 the tag information is present in packet itself and get derived through it, HG header contain tag information in case of PPD_TYPE=0 and 1 only where the outer tag get stripped out and added in the hg header, while the inner tag is still derived from the packet itself. This issue has been addressed and fixed in this release.
SDK-57184 SDK-57276		88660_A0	Bug found and fixed in BCM command diag prge_last causing "default null" program to be incorrectly printed.
SDK-57187	776877	56440_A0	For Katana2, bcm_cosq_gport_bandwidth_set was not setting the l2 shaper properly. As part of the fix l2 shaper will be configured properly.
SDK-57188	780510	56450_A0 56450_B0	bcm_mpls_port_add allocates two VPs in VPWS case, one for access and one for network port. But when bcm_mpls_port_add was invoked for second time to add network port to VPWS a new VP was being allocated instead of reusing the already allocated VP. Added fix to not allocate new VP if already allocated.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57199		88650_B1	IMPORTANT: DEFAULT BEHAVIOR CHANGE FCoE packets were dropped when FCoE switch was not enabled (bcm886xx_fcoe_switch_mode = 0). From now on, FCoE packets are treated as Ethernet packets when FCoE is disabled.
SDK-57201	779706	88650_A0 88650_B0 88650_B1 88660_A0	STG: STG APIs create/destroy STGs and set/get spanning tree status of ports in STGs. Certain STG APIs (bcm_stg_create_id/bcm_stg_destroy/bcm_stg_stp_set/bcm_stg_stp_get/bcm_stg_detach) didn't release the mutex when existing with a non-zero value. The issue detailed above can cause deadlock when using certain STG APIs. Mutex can be correctly released after the fix.
SDK-57204		88650_A0 88650_B0 88660_A0	a fix to allow future ISSU capability for alloc manager
SDK-57207	777630	56640_A0 56640_A1 56640_B0	Issue: Packets of size 64 to 75 bytes getting dropped for XE ports. Root Cause: The runt threshold value for XE ports was getting set as 76 instead of the correct value 64. Hence packets of size 64-75 bytes were getting dropped. Fix: For Triumph3 and Katana2, put explicit checks to ensure that runt threshold value is set to correct value, i.e. RUNT_THRESHOLD_XE = 64, RUNT_THRESHOLD_GE = 64 and RUNT_THRESHOLD_HG = 76. Also optimized the function mac_x_init for multiple READ and WRITE for XMAC_RX_CTRL and XMAC_TX_CTRL. Added a single write common for all devices instead of multiple instances as was present previously.
SDK-57215		88650_A0 88660_A0 88670_A0	Trill multicast adjacency BCM API implemented with new APIs: bcm_trill_multicast_adjacency_add/delete bcm_trill_multicast_adjacency_delete_all bcm_trill_multicast_adjacency_traverse Example can be found in cint_trill.c file in function mult_adjacency.
SDK-57220	780270	56850_A0 56850_A1 56850_A2	When programming MPLS_ACTION_IF_BOS=0x5 (0x5 = L3_ECMP) for a given MPLS label, the next hop entry type was set to be 1 for sending out the regular L3 packet in the previous release. In this release, the next hop entry type is set to be 0.
SDK-57224	780313	56850_A0 56850_A1 56850_A2	In earlier releases, BCM_L2_REPLACE_DES_HIT_CLEAR flag was not supported in XGS devices. This has been supported. This flag only can reset the HITDA field in L2_ENTRY table.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57230	758870	88660_A0	VLAN: L2 FECs can be used either for protection or to group LIFs like in the case of the PON application, in which the flag <code>BCM_VLAN_PORT_FORWARD_GROUP</code> is applied at <code>bcm_vlan_port_create()</code> . Removal of a L2 FEC using <code>bcm_vlan_port_destroy()</code> for a protection FEC, is performed at once for both the working and the protecting FECs upon removal of the Working path. This logic was applied also in cases where the FEC wasn't used for protection as in the case of <code>FORWARD_GROUP</code> , but only the specified FEC was removed as only one FEC is used for this type of applications. This logic caused <code>FORWARD_GROUP</code> FECs with odd id number not to be deleted as if they represent a protecting path. This was fixed, so that the protection working/protecting state condition upon FEC removal is applied only for protection FECs.
SDK-57235		56340_A0	On every DMA interval the counter value are read from <code>FT_EXPORT_FIFO</code> table and the value is populated in the report. On every read the value gets reset leading to the cumulative values not getting retained in the END report when the flow expires. With this release read of the registries clear of counter values is avoided to retain the cumulative value for the END report.
SDK-57239	778949	88650_B0	OAM: The following bugs have been fixed: When updating endpoints with <code>bcm_oam_endpoint_create()</code> with the <code>BCM_OAM_ENDPOINT_REPLACE</code> flag set, the SW DBs were incorrectly updated causing subsequent calls to <code>bcm_oam_endpoint_destory()</code> to fail. Similarly for BFD endpoints of type <code>bcmBFD TunnelTypeUdp</code> , multi-hop.
SDK-57245	781014	56450_A0 56450_B0	<code>FLEX_CTR_BASE_COUNTER_IDX</code> and <code>FLEX_CTR_POOL_NUMBER</code> were not being restored during mpls entry replace operation. Added fix to restore the FLEX counter fields and update during replace operation.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57258	779428	56450_A0 56450_B0	<p>Release notes from all sub-task JIRAs</p> <p>=====</p> <p>===== A) SDK-57533 KT2 (Unicore and Warpcore) Several issues with configuration 12 (issue 1)</p> <p>=====</p> <p>===== Problem 1) Null get was returning value 0 which is valid phy master value (i.e. slave). Fix 1) Added dummy <code>get_master()</code> in <code>xgs16g1l</code> driver. Code is changed to return <code>MS_NONE</code> and that makes phy -master as <code>NONE</code></p> <p>Problem 2: Proper medium was being detected in WARM-BOOT scenario only Fix 2: Corrected Copper/Fiber Medium detection concern in WC driver(which was applicable in Warmboot case only) Removed surrounding warm-boot condition in init part and now correct medium is returned</p> <p>Problem 3) Speed 1G was not advertised when port comes up as HG port and later converted to XE port due to max-speed set to >10000 Fix 3) Corrected WC 1g speed issue by checking additional <code>XE_PORT</code> type along with current speed before advertising speed</p> <p>Problem 4: TR 19 issue was happening with medium fiber i.e. test case was forcefully setting speed to 1G Fix 4) Added WC driver name check before forcing speed to 1G and by-passed concern</p> <p>=====</p> <p>===== B) SDK-57534 KT2 (Unicore and Warpcore) Several issues with configuration 12 (issue 2)</p> <p>=====</p> <p>===== Problem 1) Problem was happening due to wrong use of portgroup config variable in init phase and auto portgroup creation in flex-io operation i.e. assumed RXAUI related port group setting while converting hg port to 2 lane XE ports.</p> <p>Fix 1) Removed auto portgroup creation decision in flex-io operation and now user needs to set portgroup prior to flex-io operation and SDK initialization accordingly. If User doesn't use <code>auto_portgroup</code> config variable and doesn't set portgroup config variable prior to flex-io operation, SDK will throw "Behavior not guaranteed" message</p> <p>=====</p> <p>===== C) SDK-57535 KT2 (Unicore and Warpcore) Several issues with configuration 12 (issue 3)</p> <p>=====</p> <p>=====</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57263	774859	88650_A0 88650_B0 88650_B1 88660_A0	In some cases when using the diagnostic 'diag pp pktm', the meter pointer assigned to the packet would be displayed as invalid, even when the meter pointer assigned to the packet was valid. This is now fixed.
SDK-57270		88650_A0 88650_B0 88660_A0	Field Processor: Redirecting at egress according to a GPort of type System-Port was not supported. This is fixed. Reflector: The function <code>setup_port_for_reflector_program()</code> in <code>cint_benchmarking_methodology.c</code> has been changed so that the Egress FP rule modifies only the out-TM-port (by calling only the <code>bcmFieldActionRedirect</code> without <code>bcmFieldActionStat</code> actions). For a more detailed account, see <code>cint_benchmarking_methodology.c</code>
SDK-57272		88650_A0 88650_B0 88660_A0	Diag pp dblif used to return 0 for the <code>has_cw</code> (in case lif is pwe) with no relation to the real value of. Now, it is returned depending on the real value.
SDK-57277	780887	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A2 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	Issue:- In parallel mode, if VRF=0, then hardware looks only in global bucket space for bucket match, so route with VRF=0 is not allowed to be inserted to ALPM table. But the examination code was not working for the first VRF=0 route insertion. Fix:- Adding VRF=0 is disallowed explicitly in parallel mode. Update the document for this restriction.
SDK-57283		88650_A0 88650_B0 88660_A0	There was a value mismatch between set and get by calling <code>bcm_switch_control_port_set/get</code> APIs, where <code>type=bcmSwitchHashIP4Field0</code> . This mismatch is fixed.
SDK-57289	779367	88650_B1 88660_A0	When using external TCAM, control-plane writes to the external TCAM could sometimes fail when performed during line speed traffic. This issue is fixed by setting "CpuRecordPrio" field in register "TransmitCfgs" to '1' in the external TCAM application initialization.
SDK-57290	781195	88650_A0 88660_A0	Fix <code>bcm_petra_trill_port_delete</code> functionality. Add calling of <code>_bcm_dpp_mc_to_trill_remove</code> function, that removes <code>sw db mc_id</code> to nickname.
SDK-57333	739837	56850_A0 56850_A1 56850_A2	Issue:- In previous implementation for BST index resolution, if cosq value -1 was used as input, cosq 0~7 were used to retrieve the index. but by default the max cosq number is 3. So the insertion was triggered. Fix:- replace cosq 0~7 by 0 ~ <code>COS_MAX(unit) - 1</code> .

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57341	780620	56649_A0	When using "bcm_l2_learn_port_set" API to enable Class Based Learning for a trunk port, the function was returning error even though hardware programming was successful. This was because the API was trying to access another table which is not relevant for trunk ports and was using mod id value "-1" for this. The issue was resolved by adding an early return after programming the relevant Trunk table.
SDK-57343	782070	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	Communication between aging thread and other l2_addr_delete APIs thread is synchronized by binary semaphore. Occasionally when the aging thread was stopped and restarted, there was a mismatch between semaphore give and take between aging thread and other API threads. This has been fixed.
SDK-57349	781836	88650_B1 88660_A0	L3 VRRP: In some cases, if there was an error in the l3_vrrp APIs, the L3 mutex was not released. The error has been fixed, and the mutex will always be released.
SDK-57354		56840_A0	After clear operation through bcm_esw_l2_clear(), the data in structure bcm_l2_match_ctrl was sometimes released while the background thread L2MOD still needed to refer to the invalid data. This sometimes led to a crash of L2MOD. Currently the data in bcm_l2_match_ctrl won't be released in bcm_esw_l2_clear() in order to avoid this race condition.
SDK-57434		56850_A0 56850_A1 56850_A2	In previous releases, memory write operation to Ingress Pipeline tables during a ING_RESET_CONTROL Operation causes inadvertent writes to L3_TUNNEL, UDF_CAM and ING_FC_HEADER_TYPE Tables. In this release, a new flag SOC_F_MEM_CLEAR_HW_ACC indicating whether ING_HW_RESET_CONTROL is used to clear a table was added. ING_HW_RESET_CONTROL action will only happen during system initialization. In any other cases, table clear is done via table SLAM operations.
SDK-57437		56850_A0	api bcm_vxlan_port_delete is working.
SDK-57459	782198	88650_B0 88650_B1 88660_A0	Fixing memory leak issue in TRILL. Destroy TRILL port didn't free allocated memory (Add BCM_FREE to bcm_dpp_mc_to_trill_remove function).
SDK-57462	757100	88650_A0 88650_B0 88650_B1 88660_A0	Fixed l2 show diagnostic output for VPLS interface.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57469	780971	88650_A0 88650_B0 88650_B1 88660_A0	Add support for split-horizon for MPLS-Tunnel-initiator. This is useful when PWE label is built using EEI (label+push profile) and outlif that points to the EEDB is MPLS-Tunnel-initiator. In this case the PWE inherits it's orientation (HUB/ SPOKE) from the next tunnel. To set the orientation of MPLS tunnel use <code>bcm_port_class_set</code> with <code>class=bcmPortClassForwardEgress</code> and <code>port=mpls tunnel gport</code> .
SDK-57470		88650_A0 88650_B0 88650_B1 88660_A0	Reflector (RFC-2544): Ethernet Reflector program (Swaping MAC addresses) has been updated to support double tagged packets. IP program will only support single tagged packets.
SDK-57476		56850_A0 56850_A1 56850_A2	In earlier releases <code>bcm_stat_group_create</code> could get stuck in loop for egress SVP counters under scaled set-up. The issue was due to macro <code>FLEX_COUNTER_DEFAULT_EGR_DVP_ATTRIBUTE_1_TABLE_POOL_NUMBER</code> not being defined correctly for TD2, which led to endless loop when the egress flex counter pool were exhausted. It was defined to 5 for all the chips include TD2 but actually it should be less than 4 for TD2 as TD2 only has 4 egress flex counter pools. The fix was to define separate macro for TD2.
SDK-57487		56850_A0 56850_A1 56850_A2	Previously, 1-bit error reporting enabling logic and SER correction logic for all MMU tables that are CPU accessible on TD2 was not fully implemented. They have been implemented in this release.
SDK-57498	783084	56450_A0 56450_B0	In <code>bcm_qos_map_create</code> API an untagged PHB variable was being used uninitialized and that resulted in an unexpected <code>ING_UNTAGGED_PHB</code> entry being created. This was fixed to prevent untagged PHB variable from creating an unexpected <code>ING_UNTAGGED_PHB</code> entry.
SDK-57500	783310	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	In the previous release, the CPU port was not removed when the API <code>bcm_multicast_egress_delete_all</code> was called on Trident2. In this release, this issue has been addressed by removing the CPU port when the API <code>bcm_multicast_egress_delete_all</code> is called.
SDK-57503		56340_A0	Problem: <code>bcm_regex_policy_policer_attach</code> results in a crash because of internal compatibility check being done between level0 and level1 meters. Solution: Hierarchical meters are not supported on regex policies. Hence the compatibility check is disabled until we support hierarchical meters.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57505	783296	88650_A0	Fixed packet loss related to Reset CMIC interface in soft reset sequence.
SDK-57507	776846	88030_A0 88030_B0	Modification to CORE_PORT_MODE & PHY_PORT_MODE must be made with MAC in reset. Not following this rule may leave MAC in a state that no packets can be received or all received packets are runts. Other than this symptom, the MAC doesn't report anything wrong.
SDK-57515	780895	88650_A0 88660_A0	In L2 learning, traversing over the MACT to get all the inserted entries while learning, may result in an infinite loop in some rare cases. This is due to mis-handling of a rare state in an internal buffer. This is fixed.
SDK-57525	782992	56850_A0 56850_A1 56850_A2	<p>Customer requested a mechanism to find out the entropy label used for a given vxlan flow.</p> <p>In order to provide the requested mechanism, bcm_switch_pkt_info_hash_get has been modified to return the entropy label used if the packet is for the vxlan.</p> <p>For packets encapsulated into VxLAN tunnels, Entropy label is generated using RTAG7 hash. By using bcm_switch_pkt_info_hash_get(), entropy label used for VxLan can be retrieved.</p> <p>Note that entropy label is piggy backed in dst_intf.</p> <p>Example) hash_info.flags = BCM_SWITCH_PKT_INFO_HASH_UDP_SOURCE_PORT print bcm_switch_pkt_info_hash_get(unit, &hash_info, &dst_gport, &dst_intf);</p>
SDK-57533		56450_A0 5645_A0 56450_B0 56456_A0 56455_A0	<p>Problem 1) Null get was returning value 0 which is valid phy master value (i.e. slave). Fix 1) Added dummy get_master() in xgxs16g1l driver. Code is changed to return MS_NONE and that makes phy -master as NONE</p> <p>Problem 2: Proper medium was being detected in WARM-BOOT scenario only Fix 2: Corrected Copper/Fiber Medium detection concern in WC driver(which was applicable in Warmboot case only) Removed surrounding warm-boot condition in init part and now correct medium is returned</p> <p>Problem 3) Speed 1G was not advertised when port comes up as HG port and later converted to XE port due to max-speed set to >10000 Fix 3) Corrected WC 1g speed issue by checking additional XE_PORT type along with current speed before advertising speed</p> <p>Problem 4: TR 19 issue was happening with medium fiber i.e. test case was forcefully setting speed to 1G Fix 4) Added WC driver name check before forcing speed to 1G and by-passed concern</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57534		56450_A0 5645_A0 56450_B0 56456_A0 56455_A0	<p>Problem 1) Problem was happening due to wrong use of portgroup config variable in init phase and auto portgroup creation in flex-io operation i.e. assumed RXAUI related port group setting while converting hg port to 2 lane XE ports.</p> <p>Fix 1) Removed auto portgroup creation decision in flex-io operation and now user needs to set portgroup prior to flex-io operation and SDK initialization accordingly. If User doesn't use auto_portgroup config variable and doesn't set portgroup config variable prior to flex-io operation, SDK will throw "Behavior not guaranteed" message</p>
SDK-57539	781348	56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	<p>Logical table Register settings of ACL_L2IP4_ONLY partition of external TCAM are not accommodating source mac and destination mac addresses in final key. In this JIRA, LTR settings of ACL_L2IP4_ONLY are modified to have source mac and destination mac at right offsets in final key.</p>
SDK-57543	781991	56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56850_A0 56855_A0 56843_B0 56841_A3 56846_A1 56841_B0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	<p>In the previous release, when L2_MOD_FIFO mode was used and station movements happened, only one "ADD" notification would be issued on TD+, which was not incorrect. In this release, this issue has been improved by notifying one ""DEL" notification and one "ADD" notification in this kind of situation.</p>
SDK-57548	783511	56850_A0 56850_A1 56850_A2	<p>It was reported that all packets appear to be store and forward on the port when the INIT_VALUE was set to 0x3 during chip initialization for 1G mode.</p> <p>The issue was fixed by modifying the egress credit to 12 for all the speeds lower than 10Gbps.</p>
SDK-57550	777385	56450_A0 56450_B0	<p>THDO_QCONFIG_CELL could not be configured for packet processing ports greater than 128. This issue has been fixed to support complete range of packet processing ports(sub ports).</p>
SDK-57556		56850_A0 56850_A1 56850_A2	<p>corrected the test script by add the flag BCM_IPMC_RPF_FAIL_TOCPU to set value of IPMC_EXPECTED_L3_IIF_MISMATCH_TO CPU to 1</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57558		56850_A0 56850_A1 56850_A2	In previous releases, <code>bcm_vxlan_stat_detach</code> still took high execution time because of some unnecessary memory operation and extra overhead. In this release, we cut some unnecessary memory operations and redundant codes to save time, therefore execution time are reduced.
SDK-57571		56540_A0 56540_B0	On TR3 device, if the number of COSQs is changed from default value (4) to 8, after the warm boot recovery, the number of COSQs still shows as 4, since this information is not stored in the persistent storage (scache). The issue is fixed by storing this information in scache and retrieving it during warm boot level 2 recovery.
SDK-57582	782398	56624_A0 56224_A0	When using autosync for warmboot, in case of remote link down event, the scache state was not being synced to the current state. So the link state might not be recovered properly during warmboot. This is fixed by marking the scache state as dirty when a remote linkdown event occurs, so that the state is synced later. Also, <code>link_mask2</code> is not being recovered properly after warmboot. Now, this will be stored in scache and will be recovered during warmboot.
SDK-57584		88650_A0 88650_B0 88660_A0	BFD: When calling <code>bfd_endpoint_create()</code> with <code>type==bcmBFD TunnelTypeMpls</code> an additional TCAM entry is needed. Due to limited resources only 128 TCAM entries may be used for OAM/BFD. This JIRA verifies that this amount has not been exceeded and that TCAM indexes used are in the range 0-127.
SDK-57600	780870	88650_A0 88660_A0	Add push profile free when deleting PWE. Fixes resource push profile exhaustion when adding several MPLS tunnels and PWEs.
SDK-57625	786811	56340_A0	<code>bcm_cosq_gport_attach</code> is returning failure for some ports (<code>RESOURCE_UNAVAIL</code>) due to unavailability of L0 nodes.scheduler list is not reset completely on mmu soc reinitialize. Fixed the scheduler list reset.
SDK-57627	787010	56450_A0	Provided the API sequence on how to configure the burst rate
SDK-57630		88660_A0	OAM: fixed the <code>loss_farend/nearend</code> fields to return correct values (expressed in 100th of percent) in <code>bcm_oam_loss_get()</code>
SDK-57648		88650_A0	A new ISSU error detection mechanism is inserted to test if the variable layout in external storage has been changed between two versions without a proper ISSU handling. The code is under a new compilation flag <code>BCM_ISSU_SANITY</code> , in order to be used mainly internally for regression and debug and to be transparent to the user. If the compilation flag is used, both the old and the new versions must be compiled with that flag. NOTE that ISSU is not available if versions were compiled with different compilation flags.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57650		88650_A0 88660_A0	Failover ID values '-1' & '-2' are reserved for FEC Protection as 'No Protection' and 'Facility Protection' respectively. Those reserved values failed with an internal error. Now those values, when used with the BCM_FAILOVER_WITH_ID flag, produce an error that states that the value is out of range.
SDK-57652	781357	88650_A0 88660_A0	Ring Port: There was an error in bcm_vlan_port_find() where the returned failover_port_id was incorrect for a G.8032 Ring Port. The error has been fixed, and the failover_port_id is now returned correctly.
SDK-57663	783147	All 56846_A0 56844_A0 56846_A1	In earlier releases, field groups auto expansion was not recovered in Level 2 Warm Boot. In this release we now allocate an unused bit in scache layout of field module to store the groups auto expansion capability.
SDK-57669	770442	88650_A0 88650_B0 88650_B1 88660_A0	Added validity check that returns an error when user configure cos profile that is > 16 for PWE P2P.
SDK-57689		88650_A0 88660_A0	1. Reference only - added port tables to the reference list of SER-memories, which enables SER handling for these tables 2. Added dynamic memory indication for some memories. Refer to JIRA item SDK-56498 for details.
SDK-57691		88650_A0 88650_B0 88650_B1	bcm shell command "diag ssdump" was disabled. It's now enabled.
SDK-57707	787634	56640_A0	For some MACs in L2 cache, BPDU flag was not being set. This was causing ports to drop BPDUs when in STP blocked state. This issue was due to overwriting of flags for these MACs. Corrected the flags to CPU BPDU.
SDK-57725	788276	88030_A0 88030_B0	src/appl/diag/ledproc.c: Change the previous common code into C3 dedicated code, in order to not affect other modules.
SDK-57739		56850_A0 56850_A1 56850_A2	add a new case AT_L3MC_Rep_009 to verify the flag of BCM_IPMC_RPF_FAIL_DROP
SDK-57743		56850_A0 56850_A1 56850_A2	In previous release, five variables were calculated based on stat_counter_id, and three of them were used as subscript to access arrays without checking their legal ranges. Therefore, memory access violation happened. This problem has been resolved through adding proper check to those parameters to ensure the validity of their values.
SDK-57744	787141	88750_A0 88650_A0	bcm_fabric_link_status_get retrieves several link status indications. some of these link indications are sticky and should be cleared. This indications changed to be cleared on read. Meaning that this API retrieves the status since the last call.
SDK-57745		88650_A0 88660_A0	In Field Processor, the validation of the action size was changed incorrectly, such that it verifies that the size is smaller the MAX size instead of smaller or equal to it. This is fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57749		56340_A0 56640_A0	BCM56640, BCM56340 support Software Aging. The L2 entries are aged out if HITSA and HITDA are both 0. New capability is added to age out entries based on ONLY HITSA and not consider the HITDA. This is done by setting the config property 'l2x_age_only_on_hitsa' to 1.
SDK-57751		56340_A0 56640_A0	BCM56640, BCM56340 support Software Aging. The L2 entries are aged out if HITSA and HITDA are both 0. New capability is added to age out entries based on ONLY HITSA and not consider the HITDA. This is done by setting the config property 'l2x_age_only_on_hitsa' to 1.
SDK-57769	784039	56334_B0 56334_A0	In previous SDK releases, there are no SER correction support for several MMU blocks on Enduro, and thus once a parity error occurs in these blocks, it cannot be corrected and the error will be detected continuously. Fixed overview: The feature of SER correction for these MMU blocks on Enduro have been implemented. In addition, SER injection function has been added as well.
SDK-57774	781863	56850_A2	In the previous release, when the deleted I3 interface which had been added into the multicast group was re-created with the same id, the I3 interface was not attached to the multicast group, which was incorrect. In this release, this issue has been addressed by ensuring that the next hop index will be allocated and de-allocated by multicast module when the I3 interface's encapsulation id is added into and deleted from the multicast group.
SDK-57775	788841	56150_A0	Fixed possible race condition in SOC initialization routines.
SDK-57791	774941	88650_A0	In Policer module, in some cases <code>bcm_policer_create</code> fails incorrectly when the mode is <code>bcmPolicerModeCoupledCascade</code> , due to an internal software usage of an uninitialized structure. This is now fixed.
SDK-57802	788015	88650_A0 88650_B0 88660_A0	Fixed failure when deleting MPLS label in ILM table when using <code>bcm_mpls_tunnel_switch_delete</code> and SOC property 'mpls_termination_label_index_enable=1'
SDK-57812		88650_B0 88660_A0	When using external TCAM for ACL and/or forwarding databases, its configuration was not restored after warmboot. A preliminary support is added to restore external TCAM configuration during warmboot.
SDK-57813		88650_A0 88650_B0 88650_B1 88660_A0 88670_A0	In previous version, <code>BCM_VLAN_PORT_WITH_ID</code> wasn't working in forward group.now support this function.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57821 SDK-58760		88650_A0 88660_A0 88670_A0	BFD: In <code>bcm_bfd_endpoint_create()</code> the field <code>remote_flags</code> was changed to monitor the <code>Flags</code> field on incoming BFD frames (as opposed to monitoring <code>Flags</code> in <code>bcm_bfd_endpoint_get()</code> as well as setting the <code>Flags</code> on outgoing packets). The field <code>local_flags</code> was added and is used to control the <code>Flags</code> on outgoing BFD frames and (this is consistent with fields such as <code>remote_state/local_state</code> , <code>remote_daig/local_daig</code> , etc.).
SDK-57828 SDK-56669		88650_A0 88660_A0	BFD: addition of the filed <code>loc_clear_threshold</code> for <code>bcm_bfd_endpoint_create()</code> . This determines the amount of BFD frames received by the OAMP before a loss of continuity is cleared and a <code>bcmBFDEventEndpointTimein</code> event is triggered. This may be set at 0,1,2,3. Default behavior remains unaffected.
SDK-57844		56850_A0	In earlier release, adding one more <code>IPV6_64B</code> entry to table already with full <code>IPV6_64B</code> entries and some free <code>IPV6_128B</code> entries would result in inconsistency in software tables. Then trying to insert another route with same prefix would cause the process to fall into an infinite loop. This issue has been resolved.
SDK-57853		88660_A0	Trill warmboot. Sw state trill alloc link list size was not correctly calculated at warmboot trill restore, causing incorrect size after warmboot
SDK-57863		88650_A0 88660_A0	BFD: Adding accelerated endpoint with <code>bcm_bfd_endpoint_create()</code> while in <code>local_discr</code> field any of the bits 13-15 is set, caused error.
SDK-57866		88650_A0	In Rx trap module, in the allocation of a programmable trap, the error validation was incorrect. Fixed.
SDK-57889	789898	56340_A0	Aging thread runs independently , wait on semaphore for age time and wakes up & runs the bulk operation. L2 bulk operation stops and starts the aging thread. Here aging thread does not account for elapsed time it has already spent on semaphore. Lets say, aging thread already spent 25 sec and port delete operation is called at that time, aging thread stopped and restarted. Here MAC will be deleted at (25 + 30) sec = 45 sec. This is adjusted by keeping a log of elapsed time on semaphore during aging thread exit and readjusting it, during next time aging thread started.
SDK-57911	789281	88650_A0	The previous is that pass the <code>NULL</code> pointer of <code>uc/mc/bc</code> for <code>_bcm_petra_vlan_flooding_per_lif_get()</code> caused the segment fault. The fix is that pass the <code>uc/mc/bc</code> for <code>_bcm_petra_vlan_flooding_per_lif_get()</code> and get the value by the <code>uc/mc/bc</code> .

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57942		88750_A0 88750_B0	When using BCM88750 repeater, due to miss-configuration some corrupted cells might be dropped at the repeater ingress while it should be dropped at the destination device. Fixed.
SDK-57952		56640_A0	In TR3 device we were not able to delete last set of vlan service queues in the given port. We are now able to delete all the vlan service queues.
SDK-57958	782029	88650_A0 88650_B0 88650_B1 88660_A0	MPLS bug fixed: when creating and deleting a tunnel label and PWE WITH_ID, if they are re-created and the lif ids swapped, the tunnel termination fails and the tunnel label is used for forwarding. Issue is due to uninitialized values (Destination and Destination valid) when setting MPLS tunnel termination in LIF table.
SDK-57961	777725	56640_B0 56850_A2	V4 routes take half entry in shared defip tables. There can be some defip indexes containing V4 routes that does not have other half filled due to prefix restrictions. When SDK tries to recover number of indexes used for V4 prefixes in defip tables, it divides total number of routes by half but does not account about half entries. Due to this, SDK gives wrong number for total available 64/128 V6 route entries. This lead to overwriting of already existing routes in the defip tables. The fix is to count the number of half entries in defip tables and then use them during derivation of total free entries left in defip table for 64V6 and 128V6. This way SDK can return BCM_E_FULL error at table full and will not overwrite the existing entries
SDK-57962	747614	88650_A0 88650_B0 88650_B1 88660_A0	In L2 MAC table, when using API <code>bcm_l2_addr_delete_by_vlan_port ()</code> to flush all the entries on a specific PWE, all the entries that are on the same MPLS-FEC as the specified PWE were flushed. This is fixed and now only entries that are on the specific PWE (PWE-label+MPLS-FEC) are flushed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57969	790396	56850_A0 56850_A1 56850_A2	<p>In earlier releases, <code>bcm_ipmc_traverse</code> did not get IPMC v6 entries after level 2 warmboot. "ipmc ip6table show" shows nothing, but the HW table <code>L3_ENTRY_IPV6_MULTICAST</code> showed all installed entries.</p> <p>There were two root causes as follows: 1. The first <code>L3_IPMC</code> entry (index == 0) was reserved for default behavior for all chips except Katana and Katana2. In cold boot, we can find it was initialized in routine <code>bcm_esw_ipmc_init</code>. But routine <code>bcm_tr_ipmc_reinit</code> considered it as normal <code>L3_IPMC</code> entry. 2. Flag of <code>BCM_L3_IP6</code> was not set when recovering <code>L3_ENTRY_IPV6_MULTICAST</code> to l3 entry in warmboot. This caused the 'ipmc ip6table show' not work since low level driver considered it as IPV4 entries.</p> <p>The solutions are as follows: Routine <code>_bcm_tr_ipmc_reinit</code> has been changed to reserve first <code>L3_IPMC</code> entry for the default behavior for both Kanata and Katana2.</p> <p>Flag <code>BCM_L3_IP6</code> has been set in recovering <code>L3_ENTRY_IPV6_MULTICAST</code> to l3 entry in warmboot in this release.</p>
SDK-57975	790586	56450_B0	<p>VLAN parameter check was being verified for both VPLS and MIM VPNs, though the VPN is of VPLS type. Fixed to validate VLAN only for the matching VPN type VPLS or MIM accordingly.</p>
SDK-57977	769739	88650_A0 88650_B0 88650_B1 88660_A0	<p>In Ingress Field Processor, cascaded Field groups are using the <code>bcmFieldActionCascadedKeyValueSet</code> action to transmit an action value as part of the key value. This action is always performed in the HW. If unset, a zero value is expected to be transmitted.</p> <p>The entry TCAM action encoding was incorrect if:</p> <ol style="list-style-type: none"> 1. The Field group ASET was including <code>bcmFieldActionCascadedKeyValueSet</code> 2. No action value was explicitly set for <code>bcmFieldActionCascadedKeyValueSet</code> (even a zero value) <p>This is fixed.</p>
SDK-57984		56640_A0 56640_A1 56640_B0	<p>In earlier releases, after the successful warm boot recovery of VFP configuration with flex stat on TR3, if one of the field qualifiers is deleted, reinstalling of field entry failed with invalid parameter. The issue is now fixed by passing the right parameter (<code>offset_mode</code>) in the internal function while updating flex stat for the field entry.</p>
SDK-57994		88650_A0	<p>In L3 IP applications, the max VRF value is limited to 4095 in HW, but a segmentation fault was occurring when using a VRF larger than 255. This was due to a warmboot engine variable saving a VRF bitmap with an inappropriate size. Fixed. ISSU is handled transparently.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-57998		56640_A0 56340_A0	For BCM56640 and BCM56340, if config property <code>l2x_age_only_on_hitsa</code> is set, aging occurs only on HITSA, ignoring HITDA. In accordance with this the behavior of <code>BCM_L2_HIT</code> flag (<code>bcm_l2_addr_t</code> structure) is also changed, when this config variable is set. If the config variable is not set, <code>BCM_L2_HIT</code> flag is set for the entries passed to callbacks which have either HITSA or HITDA set. Also, in cases when an entry is being added, if this flag is set in the input <code>bcm_l2_addr_t</code> structure, both HITSA and HITDA will be set for the added entry. If the config variable is set, <code>BCM_L2_HIT</code> flag is set for those entries which have HITSA set. In case adding an entry, if this flag is set in the input <code>bcm_l2_addr_t</code> structure, only HITSA is set. To set HITDA, <code>BCM_L2_DES_HIT</code> flag has to set explicitly. Similar changes are reflected in <code>l2 show diag</code> shell command.
SDK-57999		88650_A0	In L2CP traps set via <code>bcm_l2_cache_set</code> , the handling of L2CP trap index of type Multicast and ports was incorrect. This is fixed.
SDK-58006		88650_A0 88650_B0 88650_B1 88660_A0	Cint: <code>cint_ip_tunnel.c</code> . ip tunnel was created in cint with incorrect ttl and dscp. Caused by SDK-55162. Consequently, checking ttl or dscp values in ip tunnels were failing when using <code>cint_ip_tunnel.c</code> .
SDK-58016		56850_A2	In the previous release, when configuring same Virtual Port (NIV Port) under multiple Mirroring sessions to get multiple copies, the function " <code>_bcm_xgs3_mtp_slot_port_indexes_get</code> " was called with the input port" parameter set to be physical gport. In this release, this function is called with the input port" parameter set to be physical unit port in this situation.
SDK-58023	736724	88650_A0 88650_B0 88650_B1 88660_A0	Added support for Split-Horizon filter in AC P2P VSI service. Up until now AC P2P service set always as Spoke, now user can configure and decide to set it either as Spoke or Hub using <code>BCM_VLAN_PORT_NETWORK</code> flag.
SDK-58043	790661	56450_A0 56450_B0	<code>mpls_port</code> add for VPLS configuration was using uninitialized pp port for given modid and portid. Fixed to convert given modid and portid to appropriate pp port before actual use. Another issue was that during subport trunk add for mpls VPLS port <code>source_trunk_map</code> table for the required entry is not configured appropriately due to incorrect modid and portid use internally. Fixed to convert pp port to appropriate modid and portid to update the required entry in <code>source_trunk_map</code> table.
SDK-58046		56850_A2	Customer found that a flex counter was automatically attached after calling <code>bcm_vxlan_port_add()</code> API. The root cause was the information of the L3 view was not cleared after changing the entry type. Now the issue has been fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58047		56850_A0	Problem Statement: DMVOQ functionality is not working Incorporated all the required change in SDK to make DMVOQ to work, for ex: Programming E2ECC_TX_PORTS_NUM, Programming of INTFI_CFG, Programming of CONGESTION_STATE_BYTES , implementation to support hlgig trunk ,Warmboot changes, Hw_index allocation is not proper, handled all the error cases ,Corrected the programming of FC_MAP_TBL, FC_ST_TBL, MMU_INTFI_OFFSET_MAP_TBL
SDK-58081	791991	56850_A0 56850_A1 56850_A2	Customer found rate limit on broadcast traffic was affected by changing DLF rate limit on the same port on TD2. It was the expected behavior in the system but some special improvement could be implemented for TD2. Now an improvement has been added to reduce this impact.
SDK-58087	792172	56450_A0 56450_B0	Subscriber delete was failing as the corresponding entry in REPL_LIST list was not removed when there are no EGR_L3_INTF/ EGR_L3_NEXT_HOP interfaces valid for replication of subscriber traffic and instead a new entry with NULL interfaces was added. The user is now able to delete the subscriber replication entries after correcting the behavior to remove the entry from the replication list when there are no replication interfaces.
SDK-58102		88650_A0 88660_A0	OAM: The following bug was fixed: Protection packets might not be sent until an event is registered.
SDK-58119	792999	56850_A2	Issue :- Qualifying SrcVxlanGport in Lookup Stage returns Internal Error Fix :- During Stage Lookup Qualifiers init routine, assigning secondary Qualifier for SrcVxlanGport was missed out. Hence the issue. Added Secondary qualifiers for SrcVxlanGport
SDK-58132		88650_A0 88660_A0	In some cases in bcm88650_A0 , bcm88650_B0 and bcm88660_A0 (when the SOC property RATE_COLOR_BLIND is set to 0) the driver would crash when calling bcm_policer_destroy_all. This is due to a software bug where uninitialized memory is used. This issue is now fixed.
SDK-58135		88650_A0 88650_B0 88660_A0	PWE bug fixed: When deleting mpls port, push profiles of the PWE is deleted as well as all the push profiles of the MPLS tunnels that the PWE is pointing on.
SDK-58136	760903	88650_B1	The original issue was that ARAD PCIe controller dropped the completion when accessing the null space CMIC descriptor address that caused the CMIC logic to wait for completion forever. The current fix is that we enable the ENABLE_PURGE IF USERIF_TIMESOUT in which case CMIC HW will inject a fake completion after timer expires and injects a ECRC error so that DMA engines will come out gracefully.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58138	791882	88650_A0 88650_B0 88650_B1 88660_A0	PWE P2P: Model was always Uniform, no matter what user configuration is. This was fixed - both Uniform and Pipe models can be configured.
SDK-58140	792399	88650_B1 88660_A0	Prevented false alarm memory ECC errors from happening when using the direct base queue to modport mapping mode, and not configuring queue mappings for queues that receive traffic or credits for a scheduler.
SDK-58144	792827	56224_B0	Multicast init fails during warmboot in <code>bcm_esw_mcast_init()</code> on some flavors of RAVEN chipset which do not have native L3 support. Provided a runtime check in <code>bcm_esw_mcast_init()</code> to check whether L3 features are supported on chipset.
SDK-58173		56440_B0	Problem - IPMC replication of L3 Multicast traffic over L3 interfaces is not working. Two issues are observed 1. While adding the IPMC configuration <code>L3_IPMC MMU_MC_REDIRECTION_PTR</code> is being modified which is not intended action in " <code>bcm_esw_ipmc_add</code> " Removed the code to reconfigure the <code>MMU_MC_REDIRECTION_PTR</code> " <code>tr_ipmc_write</code> " 2. when subscriber replication on <code>EGR_L3_INTF</code> , <code>EXT_MC_QUEUE_LIST0/</code> <code>EXT_MC_QUEUE_LIST1</code> field <code>L3_REPL_PTR0/1</code> is not programmed properly taking into account that <code>EGR_L3_INTF</code> index shall offset into (8k+ index) while configuring <code>L3_REPL_PTR</code> . It should have been programmed to be 8K+ <code>EGR_L3_INTF(index)</code> . IPMC replication of L3 Multicast traffic over L3 interfaces is working after programming the index correctly in <code>EXT_MC_QUEUE_LIST0/</code> <code>EXT_MC_QUEUE_LIST1</code> - field <code>L3_REPL_PTRx</code> for L3 interfaces.
SDK-58188	791948	88650_A0 88650_B0 88650_B1 88660_A0	In L2 MAC table, when using API <code>bcm_l2_replace_match()</code> with flag <code>BCM_L2_REPLACE_PROTECTION_RING</code> , the user supplied port mask was not used. This is fixed and now port mask is taken into consideration correctly.
SDK-58194		88650_A0	Fix bug in print encapsulation name in case of SOP only encapsulation. No effect on customer application.
SDK-58198		88650_A0	Fixed - <code>bcm_port_loopback_set()</code> with <code>BCM_PORT_LOOPBACK_PHY_REMOTE</code> for Arad fabric links. Fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58208		88650_A0	<p>In external TCAM ACL, the actions are different between BCM88650 and BCM88660 due to the common IP database in BCM88660: - The first action (bcmFieldActionExternalValue0Set) can be used for forwarding (or ACL) in both devices, with size of 48 bits - The second action (bcmFieldActionExternalValue1Set) can be used for RPF (or ACL) in BCM88650, with size of 16 bits. In BCM88660, it can be used as ACL with size 32 bits. - The second action (bcmFieldActionExternalValue2Set) can be used for RPF (or ACL) in BCM88660, with size of 16 bits. In BCM88650, it can be used as ACL with size 32 bits. - The third action (bcmFieldActionExternalValue3Set) can be used as ACL with size 24 bits.</p> <p>The distinction between BCM88650 and BCM88660 devices was performed only according to the compilation flag (BCM_88660_A0) and not according to the unit type.</p>
SDK-58237		56850_A2	Added 10G XFI FEC support.
SDK-58243	790578	88650_A0 88640_A0 88670_A0	FP: Fixed cases in which upper bits of the result were not initialized to Zeros while getting a Field.
SDK-58275		88650_A0 88660_A0	OAM: Following bug was fixed: When soc property "custom_feature_egress_snooping_advanced" is on and calling bcm_oam_endpoint_action_set() with the destination set as a trap, function may fail to properly update the classifier for up MEPs.
SDK-58315		88650_A0 88660_A0	Ring Protection: Added support for an optional sequence to perform Fast Flush. The sequence is comprised of encoding the Ring Port (FEC) as a gport of type FORWARD Port and calling bcm_l2_replace with BCM_L2_REPLACE_MATCH_DEST instead of BCM_L2_REPLACE_PROTECTION_RING.
SDK-58324		All	Fixed bug in KNET SKB allocation when using BCM API Rx buffers (use_rx_skb=0) on virtual KNET network interfaces which do not strip the VLAN tag.
SDK-58345		56850_A0 56855_A0 56850_A2	In the previous release, customer found the flex counters were not cleared after detaching. It was because on TD2, SDK only cleared X-pipe counters after detaching, but not Y-pipe counters. Now this issue has been fixed on TD2.
SDK-58358		88650_A0 88660_A0 88670_A0	<p>Jericho Protection: BCM API added for all Jericho Protection enhancements. Separate Protection tables are implemented. For Arad, the usage of bcm_failover_create() is modified and it's required to specify a failover type by setting one of the failover type flags: BCM_FAILOVER_FEC, BCM_FAILOVER_INGRESS & BCM_FAILOVER_L2_LOOKUP. The failover ID itself is now encoded with a failover type.</p>

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58360		88750_A0 88650_A0	bcm_fabric_link_status_get retrieve the fabric link status. This specific link status should be cleared on read. However, the entire quad status was cleared . Fixed.
SDK-58364	794704	56850_A0 56850_A1 56850_A2	In previous release, when customers changed one port encapsulation on TD2, the Rx Max Size was reset but Egress MTU is not. This was an omission and has been resolved.
SDK-58375	795815	88750_A0 88750_B0	Fixed a bug in "diag queues" command, the command previously printed a wrong occupied link number.
SDK-58382		56243_B0 56243_A0 56242_B0	When one port is moved to PHY loopback on 56243_A0, it turns all 4 ports in loopback. Provided a protective check in 56243_A0 internal phy driver to use broadcast mode of PHY config only when lane 0 is disabled, else use single lane config mode.
SDK-58392		88660_A0	bcm_port_enable set bug fix: In case that 2 CAUI ports and ELK are configured, The CGE1 traffic was dropped when the ELK port was disabled.
SDK-58393	794812	56541_A0 56540_A0 56540_B0 56541_B0	Wrong meter table size configuration in BCM5654X devices resulting in failure of bcm_policer_destroy, is fixed.
SDK-58395		All	1588/PTP servo configuration takes bridge_time parameter in struct bcm_ptp_servo_config_s which normally configured in seconds. Comments are modified to reflect the same.
SDK-58398	793706	88750_A0	SER interrupts were not signaled to CPU and not counted, due to being masked by a set of override bits called monitor bits. This was fixed to allow proper logging and handling of SER interrupt events by the SDK.
SDK-58407	769153	56640_A0	On TR3 device ,for HSP ports the sdk was not programming HES_PORT_CONFIG register , hence the scheduling was not proper .As part of the fix we are programming it properly .
SDK-58409		56640_A0 56641_A0 56642_A0 56643_A0 56644_A0 56645_A0 56648_A0 56640_A1 56643_A1 56644_A1 56640_B0 56644_B0 56643_B0 56648_B0 56649_B0 56649_A0	Previously, the software control structure for IBOD was not stored in scache, and hence during warmboot, the state was lost, which caused the workaround (IBOD WAR) to run again on all the effected ports. Now, the related fields in the control structure are stored in scache and are recovered during warmboot.
SDK-58410		88650_A0 88660_A0	OAM: when calling bcm_oam_action_set(), a newly allocated trap code will be returned via the field rx_trap, when applicable.
SDK-58423	793902	All 56850_A0 56850_A1 56850_A2	In the previous release, the function rx_higig2_vpn_resolve didn't parse VNI field in HIGIG2 header when it was VxLan. In this release, support has been added to parse VNI field in HIGIG2 header when it is VxLan.
SDK-58425	796179	56440_B0 56450_B0	For BCM5645x devices, support has been added for L3 packets, ingressing from trunk member ports, to be able to be trapped to CPU.



Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58428	796509	56450_B0	Fixed the issue of CCM RX not working after deletion and re-creation of RMEP due to wrongful deletion of OAM lookup key as part of RMEP delete.
SDK-58445		56640_A0 56640_A1 56640_B0	Problem: When the cascaded TCAMs are present, and the lookups are configured to include L2, L3, and L2+ACL, the system will forward a few thousand packets and then hang when the ESM bandwidth is oversubscribed. This will manifest itself as an ETU response FIFO underrun. Solution: Number of lanes to the TCAM interface is different when there are multiple TCAMs present. Fixed the issue by setting the number of lanes based on the number of TCAMs present in the system.
SDK-58460		88660_A0	Routing Over VXLAN feature in BCM 88660 . At ingress node, UDP length calculation was incorrect when sending a packet from TOR to Overlay network. API require to configure per native router interface, the expected amount of native Ethernet VLAN tags to be built at the native Ethernet header. BCM calling sequence: when creating the native router interface: (bcm_l3_intf_create), fill the member: native_routing_vlan_tags from bcm_l3_intf_t with the expected amount of native Ethernet vlan tags to be built at the native Ethernet header. see cint_vxlan_roo.c for more information. See cint_vxlan_roo.
SDK-58461	794274	88650_A0 88650ACP_A0 88650_B0 88650_B1 88660_A0	Problematic part is removed. there should be no limitation to 16 indices.
SDK-58462		88650_A0 88650_B0 88660_A0	Uninitialized variables were found in both PMF compare operation function and in LIF & RIF profile management function. This is fixed.
SDK-58463		88650_A0 88650_B0 88660_A0	In TCAM management, a sorted linked-list is used to handle the entry priority ranges: given a priority, the sorted list indicates the acceptable line range to insert a new TCAM entry. When looking to insert / get a node of this linked-list, the lookup function scans the list until it finds a node with lower priority and a node with higher priority. This lookup was not always correct. This is fixed.
SDK-58466		88650_B0 88650_B1 88660_A0	Required changes in SDK in order to have full support of external TCAM configuration restoration after warmboot. Note that this support requires the use KBP-SDK 1.2.5 and higher.
SDK-58472	793850	88650_B1	Remove ECC 1bit and 2 bit monitoring from ILKN memories of lanes that are not in use.
SDK-58487		88650_B0 88650_B1 88660_A0 88670_A0	In Ingress Field Processor, for field groups of bcmFieldGroupModeDirectExtraction type, if an entry has a 1x1 mapping (action=extracted-field) then use action macro FES instead of FEM.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58493		88650_A0	- Memory error protection: Enabling the ECC & Parity error mechanism sequence moved to the end of <code>soc_init</code> sequence. The reason is that this mechanism should be activated only when all memories are initiated, to avoid false alarms. - Shadowing (caching) mechanism: Memory shadows were not updated according to HW memory values after initialization sequence. Consequently, the cache was invalid until the appropriate entry was set explicitly following a driver call that translates to the corresponding table entry. Trying to use cache prior to writing the entry, e.g. as part of parity error-correction interrupt handler, would fail. The fix ensures that all the cache memories are updated with HW by default, as part of driver initialization.
SDK-58528	688507	88650_A0 88650_B0 88650_B1 88660_A0	OAM: Bug in OAM classification exposed only when adding the build option <code>DEBUG_OPTIMIZE=TRUE</code> . This bug causes incorrect oam module initialization when running in optimized mode.
SDK-58553	796964	56744_A0	In previous release, HG port with 25G speed config couldn't be added to config file. HG port with 25G speed config was enabled by modifying SDK to accept such configuration.
SDK-58565	797854	56850_A2	In the previous release, in <code>bcm_tr2_vlan_gport_add</code> , the <code>ing_port_bitmap</code> was overwritten by IPMC groups member port bitmap when we updated the <code>ing_port_bitmap</code> in <code>VLAN_TAB</code> . In this release, the <code>ing_port_bitmap</code> is read out first and then ORed with IPMC groups member port bitmap.
SDK-58568		88650_A0 88650_B0 88660_A0	In IPv6 Multicast FLP program, a new SOC property (<code>custom_feature_ipv6_mc_forwarding_disable</code>) is implemented. When this SOC property is enabled, the IPv6 Multicast FLP program is initialized with default values, where no key (and no lookup) is defined.
SDK-58596		56850_A2	Redirect to NIV virtual port from Ingress Stage Field Processor support was missing in earlier releases. The support is now added.
SDK-58630	792060	56850_A0 56850_A1 56850_A2	In previous releases, command " <code>I3 ip6route show</code> " would result in a cause when ALPM was enabled. This has been resolved by adding proper check to return feature unavailable instead of crashing when ALPM is enabled.
SDK-58649	792420	88660_A0	On Arad+, when calling <code>bcm_trunk_set</code> where all members have <code>BCM_TRUNK_MEMBER_INGRESS_DISABLE</code> set, an assert occurs. This assertion was fixed.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58651		56850_A0 56850_A1	Issue :- Hashing is not happening when FP RedirectEgressNextHop action is used to forward packets to ECMP group. Solution:- When the IFP wants to route an IP packet to an ECMP group, it must use the L3_SWITCH action. This will ensure that the correct ECMP hash selectors are applied to the packet. (bcmFieldActionL3Switch) Only when the IFP wants to "route" a non-IP packet to an ECMP group, it must use the "Redirect to ECMP Group" action.
SDK-58667		56640_B0	a new enum value is added to enable the rx_lox external pin for 100G application
SDK-58695	799873	56450_A0	For BCM5645x devices, support has been added for L3 packets, ingressing from trunk member ports, to be able to be trapped to CPU.
SDK-58733		All	New build toolchain XLP SDK 3.0.2 effected on GTR and WRX based SVKs.
SDK-58752	783027	All	OAM: Support handling of CCM packets in the OAMP for packet types with 2 Ethernet headers (i.e. Mac in Mac). This requires correctly setting PPH.FWD_HEADER_OFFSET to the start of the OAM header.
SDK-58769		56450_A0 56450_B0	Problem statement: DMVOQ is not working when fabric is in between and trunk ports are used in fabric Fix description: Incorporated the changes so that DMVOQ works properly, when fabric is connected inbetween the modules and trunk ports are used to connect the egress module
SDK-58784		88650_A0 88650_B0 88650_B1	Extend Warmboot engine capabilities to support easily the removal of variables in future ISSU scenarios. This is not affecting ISSU in older versions.
SDK-58787		88750_B0	BCM88750 support running memory bist test on device initialization (enabled by soc property "bist_enable"). A block reset was added after memory bist test in order to make sure that memory is re-initialized.
SDK-58791		88650_A0 88660_A0	BFD: For Arad B1 and bellow allow all types of BFD Flags on frames transmitted from the OAMP. When calling bcm_bfd_endpoint_create(), the local_flags field may be any value between 0 and 0x3f.
SDK-58821	672112	88750_A0 88750_B0	"phy measure sfi" diagnostic command is used to measure fabric link rate. Using this command caused to the fabric link counters to stop. Fixed.
SDK-58863	797630	88650_A0 88660_A0 88670_A0	arp extension feature: Backward compatibility for arp extension feature. BCM_88660_A0 can now work in BCM_88650 mode using custom soc property custom_feature_next_hop_mac_ext_arad_compatible. Using BCM_88650 mode implies learning extension header always appended when PP packet is sent to the fabric.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-58870	800874	56850_A0 56850_A1 56850_A2	Provided the debug (FP+Verbose) prints to display the UDF chunks allocated during data qualifier create <code>bcm_field_data_qualifier_create()</code> and <code>bcm_field_data_qualifier_get()</code> API. It helps to understand the chunks used in the UDF offset during creation.
SDK-58893	798171	88030_B0	add per channel counters for "show counter" CLI command on bcm88030
SDK-58900		88650_B1 88660_A0 88670_A0	In Ingress Field Processor, in Direct Extraction field groups diagnostics, the source key selection for the action was read from wrong location. This is fixed.
SDK-58943		88650_A0 88660_A0	Allocation of MPLS push profile is now supported through <code>bcm_mpls_port_add</code> api. To allocate push profile, api should be called with <code>BCM_MPLS_PORT_WITH_ID</code> and <code>BCM_MPLS_PORT2_TUNNEL_PUSH_INFO</code> flags set. <code>mpls_port_id</code> is used to indicate the push profile. Function <code>system_aux_push_profile_to_push_profile_id</code> in <code>cint_system_vswitch_encoding.c</code> should be used to set the encoding of the id to be of type push profile. <code>egress_label</code> field and <code>BCM_MPLS_PORT_CONTROL_WORD</code> flag indicate the push profile properties. Example of usage can be found in <code>vswitch_vpls_allocate_push_profile</code> function in <code>cint_vswitch_vpls.c</code> .
SDK-58944		88650_B0 88650_B1 88660_A0 88670_A0	Documentation for <code>cint_field_presel_advanced_mode</code> added to readme file
SDK-58984	802028	56248L_A0 56248L_B0 56450_A0 56450_B0	The TTL value of VC label was not retained when replacing l3 egress object using API <code>bcm_l3_egress_create()</code> with "flags = <code>BCM_L3_WITH_ID BCM_L3_REPLACE</code> ". The issue has been fixed for BCM5645x and BCM56248L devices.
SDK-59142	804236	56063 56064 56062 56060 53400_A0 53406_A0 53404_A0 53403_A0 53402_A0 53401_A0	The RCPU support is aligned with the declared feature set in Make.local for Greyhound and Ranger2 devices in this release. In previous release, a warning message of "feature not available" was shown if RCPU was removed from feature set.
SDK-59202	763574	56854_A2	In the 6.3.7 release, SDK initialization would occasionally fail on BCM56854 devices. This has been resolved.
SDK-59532		88660_A0	OAM: In Arad+, when defining a MEP on a LIF on which another passive MEP or MIP resides, the second MEP may change the first MEP's non accelerated profile, thus changing the behavior of the first MEP.
SDK-59540		88660_A0	OAM: When using OAM classification in Arad mode on Arad+, an incorrect number of non-accelerated MEP-profiles may be allocated which may cause inconsistent behavior.

Table 141:

Number	CSP #	Chips	Release Notes For 6.4.1
SDK-59785		88650_B0 88660_A0	Required changes in SDK in order to support KBP-SDK 1.3.0 for external TCAM are introduced.
SDK-59954		88650_A0 88660_A0 88650_B0	De-allocation of scheduling elements fixes: 1. Non-composite HR SE couldn't be deleted. 2. FQ SE might deallocate from the wrong range, when range odd_even=1.

Section 15: Resolved Issues for 6.4.0

The following issues are resolved in version 6.4.0 of the SDK.

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-46565	633504	56334_B0	In PTP/1588 application for Keystone processor there was an issue where setting VLAN priority <>"0" resulted in loss of communication between ToP and Host. In this release the VLAN priority mask has been corrected for Keystone.
SDK-47155	620527	56440_A0	In previous release, is, the <code>egress_tunnel_if</code> was only returned if the flag <code>BCM_MPLS_PORT_NETWORK</code> was set - which was incorrect because it should be set as well if <code>BCM_MPLS_PORT_EGRESS_TUNNEL</code> is used. In this release the following has been updated: retrieving <code>mpls_port->egress_tunnel_if</code> no longer depends on <code>network_port_flag</code> . We now check the <code>egr_l3_next_hop</code> entry type, if it is MPLS type, then we now set the <code>BCM_MPLS_PORT_EGRESS_TUNNEL</code> flag and retrieve the <code>egress_tunnel_if</code> .
SDK-47170	641741	56440_A0	During warmboot the SDK does not distinguish if the replication is on nexthop or L3 interface. This causes a warmboot failure when the replication is on nexthop. Fixed the warmboot logic to identify if the replication is on nexthop or L3 interface as per the configuration in HW replication table.
SDK-47774		88650_A0	In IP routing, the L3VPN-Default-Routing feature was not implemented: <code>BCM_L3_INGRESS_GLOBAL_ROUTE</code> had no effect upon calling. This is fixed: if the L3VPN-Default-Routing attribute is set, the IP routing lookups of the packet are <VRF, DIP> key and <0, DIP> if not found. No RPF check is performed.
SDK-47997	660499	88030_A0	The individual tests can now be configured to retain their configuration parameters upon termination of that test. That parameters will then apply to all subsequent tests. The default behavior is to roll back all the configurations. However, if it is desired to make a particular parameter persistent, it should have the line <cleanup> 0 </cleanup> on it.
SDK-49699	677743	88030_A0	New feature to support multiple cos levels and strict priority queue selection
SDK-49700	685812	88030_A0	The API <code>soc_sbx_caladan3_cop_policer_token_number_get()</code> is used to read token number of a policer.
SDK-49806		88650_A0 88650_B0	In PON application, in IPv6 Source bind implementation, the code has been changed to be more generic.

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-49819		88650_A0	Calling soc_dpp_wb_engine_deinit on one unit zeroed internal structs that contain information for all units in the system. problem is now FIXED, deinit will zero only structs belong to the specified unit.
SDK-49861		88650_A0 88650_B0 88650_B1 88660_A0	When working in MESH mode, VoQ must be mapped to a legal VoQ connector. Therefore adding/deleting a VOQ when it is under traffic is forbidden (these operations map the VOQ to an invalid VOQ). A verification was added in MESH mode, such that when voq is unmapped (a.e. mapped to an invalid connector), an error will be thrown if traffic still arrives to the VoQ. Note that this fix doesn't provide full protection, and is intended to catch an invalid state where possible. It is the application responsibility to make sure that the VoQ currently being unmapped doesn't receive any traffic.
SDK-49932	689754	88650_A0	In L3, in BCM886XX, the IPv6 host table is shared in TCAM with regular IPv6 forwarding table. However, bcm_l3_host_add API was supported for IPv6 but not bcm_l3_host_remove and bcm_l3_host_find. This is fixed.
SDK-50064	687256	56643_B0	MCSPRI was programmed with offset of 1024. The bit length of register MCSPRI is sufficient for absolute index and no offset is required. Fixed in by writing the actual index in the registry with no offset.
SDK-50087	690469	88030_A0	When immediate values are used for hstore they are checked for: <ul style="list-style-type: none"> • range (38 -256) • That the index plus the length does not exceed 256
SDK-50144	692372	88750_A0 88650_A0 88750_B0 88650_B0 88650_B1	In eyescan.h SOC function soc_port_phy_eyescan_res_print is no longer available for use. The print function has been moved to diagnostics shell, and is called from "phy diag eyescan" command.
SDK-50231	691831	88030_A0	A bug in the prior releases of the MDE manifested itself in the following way: If a 64 bit register is accessed that would result in a latency violation (e.g. it was the target of a 'hread' instruction but was subsequently accessed before the header load latency) , it crashed the assembler instead of reporting the violation gracefully. This is now resolved.
SDK-50337	692830	88030_A0	Packets arriving on the 1G ports were being redirected to incorrect queues due to incorrect PR ICC config mismatching. This has been fixed
SDK-50365	694983	88650_A0 88650_B0	Making sure bcm is attached before trying to detach it.
SDK-50368		88750_A0 88650_A0 88640_A0	Unused SOC properties (e.g. policer_fairness_enable) defined in config-sand.bcm were removed from this file

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-50437	695853	88030_A0	There was a bug in the previous releases of the MDE that prevented correct parsing of variable length headers only for the first header (any variable length header that came afterwards has been parsing correctly all along; this is how IPv4 headers have been parsed for a long time). This bug has been fixed in the current release.
SDK-50440	695544	88030_A0	A summary CSV sheet is generated showing switch, key and port usage. To use either option: -suo "file-name" --output_summary_csv "file-name"
SDK-50441	695303	88030_A0	The ability to configure the ingress and egress queue parameters on a per queue basis is now supported by the MDE. In other words, it is possible to assign different parameters to each of the 64 ingress and 64 egress queues.
SDK-50442	695307	88030_A0	In the earlier releases of the MDE, the PPE property table did not get cleaned of the previous values consistently after a test has been run. This has been fixed.
SDK-50477	696358	88030_A0	Previous releases of the MDE had a bug with the following characteristics: If a masked 64 bit register (e.g of the form rr0[40:20]) was the destination of an 'hload' or 'hread' operation, the values read were put in starting at the lowest bits, i.e. the masking bits were ignored (SDK-50477). This has been fixed with this release.
SDK-50490	695720	88030_A0	Sync attribute has been added to CMU counter config. Valid values are "true" or "false".
SDK-50519	696880	88030_A0	Previous releases of the MDE had a bug where the TSR did not get updated for instructions in the egress task. This has been fixed.
SDK-50530		88650_A0 88660_A0	When setting FabricMC using Egress+Ingress MC, the OUTLIF in IRR_MCDB must be - '0'
SDK-50569	697394	88030_A0	Previous releases of the MDE implicitly limited the length of header fields to 32 bits (MAC fields were broken to 6 byte-length fields). This is now resolved.
SDK-50570	697442	88030_A0	Earlier releases of the MDE had a bug that prevented new direct-mapped tables to be added. This has been fixed in this release.
SDK-50571	697639	88030_A0	If the ingress/egress queues in the packet header get mangled (e.g. due to a microcode bug) the model now reports this.
SDK-50675		88650_B1	88550 and 88560 are Arad-SKU chips without Interlaken. During the Arad initialization, the 88550 & 88560 SKU were incorrectly considered TDM-only devices - fixed.
SDK-50718	699557	88030_A0	Ports are initialized in the following order: 6,13,7,14,12,11,10,2,3,4,5,8,9,0,1

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-50724	699541	88030_A0	<p>To have two tables share the same memory the user must create two tables of the same width and size using the same ports and with the second table using the base address of the first table. For example:</p> <pre>sample8 { table_capacity =(32 * 100) /*# "g3p1"."ocm"."Sample LRP OCM Port 8 Table." "Sample LRP OCM Port 8 Table. Test table." */ index { test8i: 7 } entry { ocm_port (LRP_PORT_8, width=32) { pad:1 test8:31:0 } } } sample9 { table_capacity =(32 * 100) /*# "g3p1"."ocm"."Sample LRP OCM Port 9 Table." "Sample LRP OCM Port 9 Table. Test table." */ index { test9i: 7 } entry { ocm_port (LRP_PORT_8, width=32, mem_base=sample8::base) { pad:1 test9:31:0 } } }</pre>
SDK-50748	699893	88030_A0	DM table results will be ordered correctly in the results registers.
SDK-50756		88650_A0 88650_B0 88650_B1 88660_A0	Added new diagnostics to display voq/vsq programmable counters: diag counter voq/vsq Queue=x (Interval=y) diag counter voq Basequeue=x (Interval=y)
SDK-50759		88650_A0 88650_B0 88650_B1 88660_A0	Added new diag "diag cosq voq id=<id> detail=1" to print given VOQ's attributes.
SDK-50779	696166	88650_A0	<p>New APIs were added to dynamically enable/disable counter collection by counter processor engines:</p> <pre>bcm_switch_service_get bcm_switch_service_set</pre> <p>for more details about these APIs see Arad PP user manual (886X0-PG3XX)</p>
SDK-50812	700562	88030_A0	<p>The MDE now supports configuring the PPE variable in a similar manner as the property table entry. The configuration can be global or per-test. For example, to configure some fields in the ingress variable, enter the following within the ppe configuration (inside the scope of <ppe-m>... </ppe-m>: <ing_ppevar> <field-m> <name>mim transit</name> <value>1</value> </field-m> <field-m> <name>lsp_gal</name> <value>1</value> </field-m> <field-m> <name>vrrp</name> <value>1</value> </field-m> </ing_ppevar></p>
SDK-50849		88750_A0 88650_A0 88650_B0 88650_B1 88660_A0	<p>1. MBIST (internal memories BIST) is fixed to work on 88660. MBIST can be enabled at startup using the <code>bist_enable soc</code> property.</p> <p>2. Starting with the 6.3.2 release, enabling of the mbist output is done using: <code>dbm socdnx +mem +VERBOSE +err -cnt</code> Instead of the previous: <code>dbm soc +mem +VERBOSE +nor +err</code></p> <p>Using the reporting line above, all the memory debugging information previously available (for 88650 and for 88750) is displayed as before.</p>
SDK-50894	701166	88030_A0	<p>Release 144 of the MDE had an issue that the PPE header checker and LAG template must be specified, even if the application does not need it. Omitting these two optional parameters crashed the MDE. This has been fixed.</p>

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-50963		88650_A0 88650_B0 88660_A0	When using User-Header (e.g. in cascaded Ingress-Egress ACL or in VMAC), the user-header should be removed before the packet exits the system. This was not the case for OTMH program and Mac-in-Mac. This has been corrected.
SDK-50972	701844	88650_A0 88650ACP_A0 88650_B0 88650_B1	Table EGQ_FQP_NIF_PORT_MUX need to be tuned to avoid packet drops. Optimized internal table arrangement to prevent underrun and insure the desired ports rate. The fix is applied only on driver initialization.
SDK-51035		88650_A0 88650_B0 88650_B1	VLAN assignment according to port,protocol: VLAN assignment procedure is according to profile. Increased the number of port protocol entries per profile from 10 to 16
SDK-51048	700857	56850_A0 56850_A1 56850_A2	PFM_RULE_APPLY field in IGMP_MLD_PKT_CONTROL register cannot be controlled in previous release. Added support for controlling this bit by pkt protocol control approach.
SDK-51093	705776	88030_A0	The condition (header access latency) was considered cleared after one cycle (i.e. next instruction) rather than two cycles.
SDK-51170		88660_A0	OAM: Support RDI generation method. Generation method is configured through the bcm_oam_endpoint_create api with the following flags2: BCM_OAM_ENDPOINT2_RDI_FROM_RX_DISABLE /* RDI bit on outgoing packets may be taken from RDI indication on received packets. */ BCM_OAM_ENDPOINT2_RDI_FROM_LOC_DISABLE /* RDI bit on outgoing packets may be taken from LOC indication of peer endpoint. */
SDK-51184	705114	88030_A0	The LUG is out of date with respect to the COP load latency, the correct value is 40.
SDK-51351	709776	88030_A0	I can see the code the load latency is changed from 37 to 40
SDK-51368	707551	56830_A1 56830_A0 56830_A2	BCM56830 is considered as a switch instead of a fabric and attached with proper drivers. SDK implementation has been corrected based on this determination
SDK-51498	696152	88130_B0	QE2000 to Sirius traffic issue was resolved with a fix to bcm_fabric_crossbar_connection_set () to set up both A and B plane connections to support plane crossover.
SDK-51528	711580	All	Fixed the issue with packet drop counter when the packet is dropped by policer.
SDK-51541 SDK-50704		88650_A0	In order to detect and fix ECC2 and parity errors, one can use the BCM switch control bcmSwitchCacheTableUpdateAll. The procedure will go over all cached memories, read them from HW, and in case it detects an error, a matching interrupt will be initiated to be corrected by the appropriate corrective action. When caching memories, it is recommended to update all cached memories before initiating a WB/ISSU cycle. The cached memories are read from the HW during WB/ISSU. Updating all cached memories ensures that all potential errors are handled using the available shadow data.

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-51643		56340_A0	Fixed and tested on GTO/BCM56340A. BCM init and rc failures aren't happening anymore.
SDK-51689	713650	88650_B1	In BCM8865X, a bug at egress HW was mishandling packets that being terminated to size of 192-255 Bytes. In BCM88660, this HW bug was fixed. Enabling this bugfix during the Driver init is inserted.
SDK-51821	716070	88030_A0	Added check for invalid combination (Simple64 & Automatic mode): Error! [87509] null->0:0->0.1 = Counters group ertctr: Simple64 counters don't support automatic mode.
SDK-51827	716807	88030_A0	Resolves ucode reload issue seen in 2_146 and TOT.
SDK-51881	702602	56640_B0 56850_A2	Vlan Service queuing bugs addressed. 1. gport_attach/detach to take care of internally attaching the given number of Queues during add. 2. Queue alignment of Vlan queuing changed to 1(no alignment required).
SDK-52216	711504	56846_A0 56845_B0 56845_A2 56844_A0 56842_A0 56840_A0 56820_A0 56820_B0 56800_A0 56746_A0 56745_A0 56744_A0 56743_A0 56740_A0 56725_A0 56720_A0 56700_A0 56689_B0 56685_B0 56685_A0	A request was made to add the ability to override "protocol" field in SKB before pushing packet into network stack. the following fields were added in the packet filter structure to support this request. : int dest_proto; /* If non-zero this value overrides the default protocol type when matching packet is passed to network stack. */ int mirror_proto; /* If non-zero this value overrides the default protocol type when matching packet is passed to network stack. */ add corresponding fields in packet filter structure to configure the desired protocol type.
SDK-52355		56850_A0 56850_A1 56850_A2	Support has been added for retry in mem insert and delete for hash tables. Inline hash memory recovery was implemented for insert and delete operations. When an insert/delete operation encounters a parity error, the inline recovery routine will be invoked. The inline recovery routine will calculate different hash buckets in different hash memory banks based on the entry that will be inserted/deleted, then restore the each bucket in these banks. For new-added hash key types in Trident2 hash tables, support for these key types in hash entry comparing routine has also been added.
SDK-52381	717920	56850_A0	In earlier releases, L3 Conflict Get, bcm_td2_l3_conflict_get() was broken. This has been resolved.
SDK-52443	705504	88650_A0	Case Summary: Traffic drops at ingress on a newly added LAG member, if it is the first member on the ARAD device after cross connection created. To avoid the problem, API support was added. Using the following function flags bcm_trunk_member_addbcm_trunk_member_delete and bcm_trunk_set the user can update only egress or ingress tables. The user can update only egress tables, configure relevant port parameter and then update ingress port, with this sequence there will be no traffic drop.
SDK-52448	723913	56450_A0	HQOS support is added for UNI ports on Katana2
SDK-52471	723924	88030_A0	Order issue addressed in template generated code.

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-52514	725210	88030_A0	Release 149 had a bug where the MDE crashed if a hash template was not configured for the ingress queue. This has been addressed in this release.
SDK-52521	724174	56850_A0	In the previous release, in function <code>_soc_td2_alloc_sched()</code> , HQOS hierarchy was being assumed. If users did not use the same hierarchy as defined in <code>_td2_port_lls_config()</code> , issues would be seen. In this release, a LLS port doesn't clear other ports' hardware resource when <code>bcm_cosq_gport_add()</code> is called on Trident2 chips.
SDK-52722		88650_A0	<code>bcm_l3_ingress_create</code> now returns an error if the flag <code>BCM_L3_INGRESS_WITH_ID</code> is not enabled (instead of just ignore)
SDK-52758	727046	88030_A0	BCM88030: fixed bug where MPLS label was over-writing the IP address RCE key field.
SDK-52767		88750_A0 88650_A0 88750_B0 88650_B0 88650_B1 88660_A0	"show features" diagnostics was added.
SDK-52895	729741	All	RPC has been enabled for the HASH bank APIs.
SDK-52970	730058	All	<code>L2_matched_traverse</code> used a loop to test availability of MOD FIFO, and if MOD FIFO became hung, the loop became endless and eventually caused MOD FIFO thread to become dead. Added a timeout to break out of the loop if MOD FIFO hangs.
SDK-53012	715940	88030_A0	Fixed the wrong action type for mirror & drop in egress RCE action table.
SDK-53021	720668	56850_A0	Updated the documentation related to <code>BCM_PORT_CONGESTION_CONFIG_DESTMOD_FLOW_CONTROL</code>
SDK-53070	688151	56850_A2	Two command options are added for the eye margin functional calls. The syntax example is <code>phy diag xe0 veye lane=0xc time_upper_bound=16</code> The "lane" option specifies which lane in a given port is enabled for the eye measurement. If the lane is not specified, the default is 0 which means all the lanes are enabled for the eye measurement in a given port. 0xC means lane 2 and lane 3 of the port is enabled. Each binary bit of the value represents a lane. The "time_upper_bound" is to specify the total max time limit for a given eye measurement node. Its unit is second. The default is 256 seconds. Note that this option only accepts the value equal to or larger than 4 seconds.
SDK-53077	731557	88030_A0	For IPv4, we now could use <code>max_capacity_limit/240</code> to estimate the number of tcam entries need to be used. For IPv6, use <code>max_capacity_limit/168</code> to estimate it.

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-53112	677748	88030_A0	<p>As of the current release of the MDE, the TMU MAC table subkeys can be split into three (as opposed to two: VSI and MAC) fields ONLY for testing on the model as follows:</p> <p>An optional 1-bit field, called 'bmac' can be specified as part of the MAC subkey. This means that the MAC subkey can be optionally split into three fields (1 bit BMAC, 15 bit VSI and 48 bit MAC address) This is meant to used in simulating a PBB (Mac-in-Mac) header where the bmac field can be 1 to specify that the MAC address in the subkey is a bridging MAC or 0 to specify that it is a customer (inner) MAC. Since this field is optional, the default value is 0 which covers both the cases of a non-PBB header as well as the inner MAC of a PBB header.</p> <p>The UcTst.xml file that is provided with this release has placeholder examples of this new usage.</p>
SDK-53115	731716	56850_A0 56850_A1 56850_A2	<p>For TCAM memories protected by SER engine, corrupt bitmaps have been added to track SER errors detected on them. SER correction logic will filter duplicated SER errors via this corrupt bitmap.</p>
SDK-53127	730044	56334_B0 56334_A0	<p>In an earlier release switching double tagged frames between layer 2 logical ports on Enduro was inconsistent with TR3/TR2 behavior. This has been correction by synchronizing the behavior of double tagged frames switching on Enduro with TR2's behavior.</p>
SDK-53198	733029	56640_A0 56440_A0 56641_A0 56450_A0	<p>The problem in existing code was - <code>bcm_l3_egress_get()</code> was not able get the <code>mpls_qos_map_id</code> i.e logical qos id (if object was created by <code>bcm_qos_map_create()</code>). In this release to solve this issue, a new routine <code>_egr_qos_hw_idx2id()</code> has been created. This converts the hardware index to logical qos id. This function is used in the <code>bcm_l3_egress_get</code> to retrieve the <code>mpls_qos_map_id</code>. This function can handle both the <code>qos_id</code> created by <code>bcm_qos_map_create()</code> and <code>bcm_mpls_exp_map_create()</code> .</p>

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-53204	732754	88030_A0	<p>Certain configuration parameters are required to set and configure the ILKN OOB flow control. The following illustrates how the required parameters are used and when to use them Some parameters are specific to Caladan3 chip.</p> <p>1) Enable Interlaken Flow control when there is an interlaken port</p> <ul style="list-style-type: none"> • <code>fc_oob_type_<ilx>=2</code> <p>2) Enable Interlaken flow control when there is no interlaken port</p> <ul style="list-style-type: none"> • <code>fc_type_il_line=1</code> • <code>fc_type_il_fabric=1</code> <p>3) Default Calendar length is 64, if not the following has to be set appropriately</p> <ul style="list-style-type: none"> • <code>fc_calendar_length_il_line=<len></code> • <code>fc_calendar_length_il_fabric=<len></code> <p>4) Debugging 1.Ignore the FC OOB status</p> <ul style="list-style-type: none"> • <code>ilkn_interface_status_oob_ignore=1</code> <p>2.Enable Loopback of FC data</p> <ul style="list-style-type: none"> • <code>fc_oob_loopback_<ilx> = 1</code>
SDK-53218	727679	88650_A0 88650_B0 88650_B1	<p>Port TPIDs: When deleting TPID to default behavior with API <code>bcm_port_tpid_delete</code> or <code>bcm_port_tpid_delete_all</code> , TPID profile wasn't changed correctly.</p>
SDK-53311	733395	56850_A2	<p>Operations in <code>soc_l2x_freeze/thaw()</code> for TD2 have been optimized by using <code>ING_MISC_CFG2_CML_NEW_OVERRIDE/CML_MOVE_OVERRIDE</code> to disable/enable the learning instead of modifying individual port/svp table entries.</p>
SDK-53346	731211	88650_A0	<p>The member ID of Trunk port as it written to the <code>IHP_PTC_SYS_PORT_CONFIG & IHP_VIRTUAL_PORT_TABLE</code> was not the index in the IRR lag mapping table. We fixed lag member delete/add to keep this tables synced.</p>
SDK-53348	733779	88030_A0	<p>In the previous release an issue was discovered with clearing Interlaken counters. This has been fixed. When using "clear counters", we now clear those counts for both hardware counters and the software variable, then we will reset those counts.</p>

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-53367	735381	88030_A0	<p>SDK-53367 PTN6500 - line rate test cause "ped egress drop" issue. This is actually a COP access constraint that was not covered. The constraint is the 2nd rule of the following:</p> <p>1st rule: Independent of the targeted instance, COP accesses always have AT LEAST a 64 instruction resource shadow. (covered by 51017 R-000-10 COP Resource Shadow (load) and 51018 R-001-10 COP Load Latency)</p> <p>2nd rule: Per COP instance, the assembler maintains a counter (initialized to 0). The counter value must be <= 128 in order to access the associated COP instance without violating the constraint. The counter is maintained as follows: 1.) Starts at 0. 2.) Add 128 to the counter when the associated COP instance is accessed. 3.) For each instruction slot, if the counter is non-zero, and it has been at least 64 instructions since the associated COP instance has been accessed, subtract 2 from the counter. Note that this is not the same as the COP port resource shadow (see 1st rule) which is common to both COP instances.</p> <p>New resource constraint: R-002-10 New error number: 51064</p> <p>Example error message:</p> <p>Error! [51064] copTest.lrp3->90:5->1.76 => 223:5->1.199 = constraint R-002-10 COP per instance access constraint violation. shadow:1 Next safe instruction for COP instance 0 access is: 200</p>
SDK-53370	726683	88650_B1	<p>When using FCoE example CINT, the FCoE header in the forwarded packet was omitted. The trap ID that is used for FCF workaround, which fixes the forwarding header offset was wrong and is now fixed.</p>
SDK-53385	721111	88650_A0	<p>In RX snoop, the number of HW snoop commands is 16, where 0 is reserved for packets which are not snooped. Due to a SW bug, the number of available snoop commands was 15 and not 16. This is fixed.</p> <p>This fix was reverted in 6.3.5 because it breaks ISSU and can be taken from TOT as a patch.</p>
SDK-53414	734150	56850_A0	<p>In the previous release we did not support HG13 on TD2. In this release support has been added for speed 13000M. Additionally in</p> <pre>soc_td2_port_asf_speed_set(), if speed = 0xe, speed 13000M duplex full will be selected.</pre>
SDK-53449	733944	56850_A0 56855_A0 56854_B0 56854_A0 56850_A1 56851P_A1 56851_A1 56850_A2 56851_A2 56851P_A2 56854_A2 56853_A2 56852_A2 56855_A2 56851_A0 56852_A0 56852_A1 56853_A0 56853_A1	<p>In the previous release.</p> <p>bcmportControlDoNotCheckVlan was being overwritten by unrelated port API calls. This has been fixed.</p>

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-53453	675993	56846_A0 56845_B0 56840_A0 56640_A0 56440_A0 56850_A0 56855_A0 56843_B0 56340_A0 56640_B0 56440_B0 56850_A1 56850_A2 56344_A0 56342_A0	Added support for MIM payload tpid select and MIM hash by using payload or tunnel header.
SDK-53488	736297	88650_A0	ARAD does not support Type-4 VCCV (GAL over PWE). We propose a solution to trap GALoPWEoLSPoETH packets to CPU by using bcmRxTrapMplsUnexpectedNoBos trap. Field processor is used to change the MPLS InLif to PWE InLif, so the trapped packet contains PWE InLif in the PPH. NOTE: In ARAD soc property custom_feature_mpls_termination_check_bos_disable should be set. In ARAD+ no soc property is required. For usage example see cint_gal_o_pwe_o_mpls.c
SDK-53515	734789	5615_A0	HR2: QSGMII running as SGMII mode was showing the wrong duplex attribute. In this release we have fixed the duplex get function for qsgmii serdes in sgmi mode
SDK-53558	716344	88030_A0	The exception byte counter was not incremented when a packet is dropped due to drop tag (or) drop untag configuration. This is fixed.
SDK-53560	719326	88030_A0	The pvv2e.hit bit handling is fixed. The soc_sbx_g3p1_utils_pvv2e_update() & soc_sbx_g3p1_utils_pvv2e_add() functions sets the hit bit by default.
SDK-53563	736727	56334_B0 56334_A0	Fixed error return value of bcm_mpls_label_stat_get/get32 on Enduro
SDK-53612	728198	88650_B1	When working in 2P or 1P mode, ISQ root shaper doesn't work correctly (traffic is not shaped regardless shaper configuration).
SDK-53613	735136	88650_A0 88650ACP_A0 88650_B0 88650_B1	When setting WRED using bcm_cosq_gport_discard_set, and using min/max threshold values close to the limit of 256MB-1, an error was returned. This was fixed, and the range of the min/max WRED thresholds was extended up to 2GB. The actual value that can be specified is up to 2GB-1 sue to the range of the int structure field that specifies it.

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-53639	737816	All	<p>bcmFieldQualifyL3Ingress qualifier offsets are updated for Ingress Field Processor to match with regfile (56850).</p> <p>Problem : Previously the qualifier set was showing "Feature Unavailable" error during group create. This was due to missing initialization of L3Ingress qualifier.</p> <p>Solution: With this fix the group create will cause "No resources for operation" error for the qualifier set mentioned above. This is because after adding bcmFieldQualifyL3Ingress to the Groups QSET the KEY width is exceeding what TD2 IFP H/W can support.</p> <p>Customer has to remove either bcmFieldQualifyInterfaceClassL3 or bcmFieldQualifyIntPriority qualifier from the Groups QSET set to add bcmFieldQualifyL3Ingress to existing Group. OR Customer has to create a new Field Group with bcmFieldQualifyL3Ingress qualifier in it.</p>
SDK-53640		56334_B0 56334_A0	<p>In earlier releases a crash was introduced when initializing BCM56634 via changes added in soc_do_init. In this release we have added device checking for the new block of code introduced to change the PCIe SerDes deemphasis on certain devices (fix for SDK-50513).</p>
SDK-53650		88650_A0 88650_B0 88650_B1	<p>Fixed the crash in bcm_dpp_rx_packet_parse when called with BCM_ARAD_PARSE_PACKET_IN_INTERRUPT_CONTEXT.</p> <p>Registered are not accessed when working in interrupt context.</p>
SDK-53654		88650_A0 88650_B0 88650_B1 88660_A0	<p>Fixed "diag rates sch" shell command crash which is caused by reading non-existent register.</p>
SDK-53675		88650_A0 88650_B0 88660_A0	<p>BFD packets may now be trapped using pre-defined traps. When calling bcm_bfd_endpoint_create(), the remote_gport field may be set to a valid gport for trapping BFD frames to that gport, GPORT_INVALID for the default behavior or remote_gport may be set to a pre-configured trap code. For the latter, call bcm_rx_trap_type_create() to get a trap code, bcm_rx_trap_set() to set the trap code with a valid dest_port configured in bcm_rx_trap_config_t, BCM_RX_TRAP_UPDATE_DEST and BCM_RX_TRAP_TRAP flags set. Then set remote_gport to the said trap code before calling bcm_bfd_endpoint_create().</p>
SDK-53731	739297	88750_A0 88750_B0	<p>"diag queues" command shell wasn't functional over dual pipe.</p>
SDK-53767		88650_A0	<p>cleaned HW access that were causing error prints during warm reboot (due to statistic threads that would perform HW access)</p>

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-53824		56450_A0	In previous releases a crash could occur with subport configuration. This has been addressed by correcting the wrong assumption of COE subport configuration for calculating <code>op_nodes</code> for physical ports. Now it is purely based on number of <code>op_nodes</code> consumed by each physical port in sequence (CPU,LPBK,1..40)
SDK-53826		88660_A0	PON: <code>bcm_vlan_port_create</code> set incorrect configuration when having 3 tags manipulation under <code>bcm886xx_vlan_translate_mode=1</code> .
SDK-53837		88650_B0 88650_B1	Fix documentation of <code>cint_vswitch_cross_connect_p2p.c</code> to load all the cints in correct order.
SDK-53839		88650_B0 88650_B1	VPLS: Added <code>cint_cint_vswitch_vpls.c</code> support in index mpls mode that enables termination of up to 3 labels. Index mode is set using soc property <code>mpls_termination_label_index_enable</code> .
SDK-53867	740320	56850_A0 56850_A1 56850_A2	One of the following solutions can be used to address the persistent link flap problem with CR4 + AutonegOn on ports: (a) Do NOT enable <code>RX_SERDES_LOS</code> and Fast linkscan property in the configuration. This means to disable the SOC property <code>rx_serdes_los</code> , or, EXCLUDE the port(s) from the SOC property <code>rx_fast_los_link_{port}</code> . (b) If the user wants to be able to remove/add ports into fast linkscan dynamically, user can now disable the fast linkscan port control "bcmPortControlRxFastLOS": <code>bcm_port_control_set/get(unit, port, bcmPortControlRxFastLOS, ...)</code> .
SDK-53891		88650_A0	Relevant only for TDM bypass mode: Warm boot would reset some of the TDM fabric direct routing configuration, and cause later configuration of it to be incorrect.
SDK-53946		88650_B1 88660_A0	Important note: in Fiber channel APIs, due to an API change, the user must replace <code>bcm_fcoe_zone_entry_t->vsan.vsan</code> by <code>bcm_fcoe_zone_entry_t->vsan_id</code> , e.g. in <code>bcm_fcoe_zone_add</code> API.
SDK-54096		88650_A0	The private header that includes the packet size was supported in previous version, but collide with other PP features that are supported by the egress editor. This fix resolves the issues when trying to enable multiple PP features while still maintaining the usage of the "size header" addition on top of the packet header.
SDK-54378		88650_A0	To debug more easily warmboot issues, a SW state dump is available via <code>BCM>diag ssdump</code> The SW state dump output to screen can now be disabled.

Table 142:

Number	CSP #	Chips	Release Notes For 6.4.0
SDK-56158	756172	56440_A0	Problem description: When L2 MAC Table is full, customer was unable to create the BFD session due to MAC table full issue. Fix description: This issue is fixed by calling <code>_bcm_l2_hash_dynamic_replace</code> function, if <code>soc_mem_insert</code> function returns <code>BCM_E_FULL</code> . and also code changes are done in <code>_bcm_l2_hash_dynamic_replace</code> function to support <code>BFD_KEY</code> for Katana device.
SDK-66055		56344_A0	Add the port config property <code>BCM56340_4X10=1</code> will have the HG[21] port configured as it is not the default port config.

Section 16: Appendix

STATUS OF BCM56960 IFP QUALIFIER TESTING

Table 143: Field Qualifiers

Qualifier Name	SDK-6.4.4 EA8 Test Status	SDK-6.4.4 Test Status
bcmFieldQualifyCnTag	Not tested	Not Tested
bcmFieldQualifyColor	Pass	Pass
bcmFieldQualifyDosAttack	Fail	Pass
bcmFieldQualifyDrop	Pass	Pass
bcmFieldQualifyDSCP	pass	Pass
bcmFieldQualifyDstClassField	pass	Pass
bcmFieldQualifyDstClassL2	pass	Pass
bcmFieldQualifyDstClassL3	pass	Pass
bcmFieldQualifyDstGport	Pass	Pass
bcmFieldQualifyDstHiGig	Not tested	Pass
bcmFieldQualifyDstIp	pass	Pass
bcmFieldQualifyDstIp6	Pass	Pass
bcmFieldQualifyDstIp6High	Pass	Pass
bcmFieldQualifyDstIpLocal	Fail	Pass
bcmFieldQualifyDstL3Egress	Pass	Pass
bcmFieldQualifyDstL3EgressNextHops	Pass	Pass
bcmFieldQualifyDstMac	Pass	Pass
bcmFieldQualifyDstMimGport	Not tested	Pass
bcmFieldQualifyDstMimGports	Not tested	Not Tested
bcmFieldQualifyDstMplsGport	Fail	Fail
bcmFieldQualifyDstMplsGports	Not tested	Not Tested
bcmFieldQualifyDstMulticastGroup	Fail	Fail
bcmFieldQualifyDstMultipath	Fail	Fail
bcmFieldQualifyDstNivGport	Pass	Pass
bcmFieldQualifyDstNivGports	No Support	No Support
bcmFieldQualifyDstPort	Pass	Pass
bcmFieldQualifyDstTrunk	Fail	Pass
bcmFieldQualifyDstVlanGport	Fail	Fail
bcmFieldQualifyDstVlanGports	No Support	No Support
bcmFieldQualifyDstVxlanGport	Not tested	Not Tested
bcmFieldQualifyDstVxlanGports	Not tested	Not Tested
bcmFieldQualifyEtherType	Pass	Pass
bcmFieldQualifyExtensionHeader2Type	Fail	Fail
bcmFieldQualifyExtensionHeaderSubCode	Pass	Pass
bcmFieldQualifyExtensionHeaderType	Pass	Pass
bcmFieldQualifyFabricQueueTag	Not tested	Not Tested
bcmFieldQualifyFibreChanDFCtl	Not tested	Not Tested
bcmFieldQualifyFibreChanSrcBindCheck	Not tested	Not Tested
bcmFieldQualifyFibreChanSrcFpmaCheck	Not tested	Not Tested
bcmFieldQualifyFibreChanSrcId	Not tested	Not Tested

Table 143: Field Qualifiers

Qualifier Name	SDK-6.4.4 EA8 Test Status	SDK-6.4.4 Test Status
bcmFieldQualifyFibreChanType	Not tested	Not Tested
bcmFieldQualifyFibreChanVFTHopCount	Not tested	Not Tested
bcmFieldQualifyFibreChanVFTVersion	Not tested	Not Tested
bcmFieldQualifyFibreChanVFTVsanId	Not tested	Not Tested
bcmFieldQualifyFibreChanVFTVsanPri	Not tested	Not Tested
bcmFieldQualifyFibreChanZoneCheck	Not tested	Not Tested
bcmFieldQualifyFibreChanVFTHopCount	Not tested	Not Tested
bcmFieldQualifyForwardingType	Pass	Pass
bcmFieldQualifyForwardingVlanId	Pass	Pass
bcmFieldQualifyForwardingVlanValid	Not tested	Not Tested
bcmFieldQualifyGenericAssociatedChannelLabelValid	Not tested	Not Tested
bcmFieldQualifyHiGig	Pass	Pass
bcmFieldQualifyIcmpTypeCode	Pass	Pass
bcmFieldQualifyIngressStpState	Pass	Pass
bcmFieldQualifyInnerIrpProtocol	Fail	Fail
bcmFieldQualifyInnerTpId	Pass	Pass
bcmFieldQualifyInnerVlanCfi	Fail	Pass
bcmFieldQualifyInnerVlanId	Pass	Pass
bcmFieldQualifyInnerVlanPri	Fail	Pass
bcmFieldQualifyInPort	Pass	Pass in Per Pipe Mode
bcmFieldQualifyInPorts	Pass	Pass in Per Pipe Mode
bcmFieldQualifyIntCongestionNotification	No Support	No Support
bcmFieldQualifyInterfaceClassL2	Fail	Pass
bcmFieldQualifyInterfaceClassL3	Pass	Pass
bcmFieldQualifyInterfaceClassPort	Pass	Pass
bcmFieldQualifyInterfaceClassVPort	Pass	Pass
bcmFieldQualifyIntPriority	Pass	Pass
bcmFieldQualifyIp4	Pass	Pass
bcmFieldQualifyIp6	Pass	Pass
bcmFieldQualifyIp6FlowLabel	Pass	Pass
bcmFieldQualifyIpFlags	Pass	Pass
bcmFieldQualifyIpFrag	Pass	Pass
bcmFieldQualifyIpmcStarGroupHit	Fail	Pass
bcmFieldQualifyIpInfo	Pass	Pass
bcmFieldQualifyIpProtocol	Pass	Pass
bcmFieldQualifyIpTunnelHit	Pass	Pass
bcmFieldQualifyIpType	Pass	Pass
bcmFieldQualifyISid	Not tested	Not Tested
bcmFieldQualifyL2CacheHit	Fail	Pass
bcmFieldQualifyL2DestHit	Pass	Pass
bcmFieldQualifyL2Format	Pass	Pass
bcmFieldQualifyL2GreSrcIpHit	Not tested	Pass
bcmFieldQualifyL2GreVfiHit	Not tested	Not Tested
bcmFieldQualifyL2PayloadFirstEightBytes	Fail	Fail

Table 143: Field Qualifiers

Qualifier Name	SDK-6.4.4 EA8 Test Status	SDK-6.4.4 Test Status
bcmFieldQualifyL2SrcHit	Pass	Pass
bcmFieldQualifyL2SrcStatic	Pass	Pass
bcmFieldQualifyL2StationMove	Fail	Pass
bcmFieldQualifyL3DestHostHit	Pass	Pass
bcmFieldQualifyL3DestRouteHit	Pass	Pass
bcmFieldQualifyL3Ingress	Pass	Pass
bcmFieldQualifyL3Routable	Pass	Pass
bcmFieldQualifyL3SrcHostHit	Fail	Pass
bcmFieldQualifyL4DstPort	Pass	Pass
bcmFieldQualifyL4Ports	Not tested	Not Tested
bcmFieldQualifyL4SrcPort	Pass	Pass
bcmFieldQualifyLoopback	Not tested	No Support
bcmFieldQualifyLoopbackType	Not tested	No Support
bcmFieldQualifyMacAdrsNormalized	Pass	Pass
bcmFieldQualifyMHOpcode	Pass	Pass
bcmFieldQualifyMirrorCopy	Not tested	Not Tested
bcmFieldQualifyMimSrcGportHit	Not tested	Not Tested
bcmFieldQualifyMimVfiHit	Not tested	Not Tested
bcmFieldQualifyMirrorCopy	Not tested	Pass
bcmFieldQualifyMplsControlWordValid	Not tested	Not Tested
bcmFieldQualifyMplsForwardingLabel	Not tested	Not Tested
bcmFieldQualifyMplsForwardingLabel Action	Not tested	Not Tested
bcmFieldQualifyMplsForwardingLabel Bos	Not tested	Not Tested
bcmFieldQualifyMplsForwardingLabel Exp	Not tested	Not Tested
bcmFieldQualifyMplsForwardingLabel Id	Not tested	Not Tested
bcmFieldQualifyMplsForwardingLabel Ttl	Not tested	Not Tested
bcmFieldQualifyMplsLabel1Hit	Not tested	Not Tested
bcmFieldQualifyMplsLabel2Hit	Not tested	Not Tested
bcmFieldQualifyMplsOuterLabelPop	Not tested	Not Tested
bcmFieldQualifyMplsStationHitTunnel Terminated	Not tested	Not Tested
bcmFieldQualifyMplsTerminated	Not tested	Not Tested
bcmFieldQualifyMyStationHit	Not tested	Pass
bcmFieldQualifyNatDstRealmId	Pass	"Functionality Not covered. API Testing is covered"
bcmFieldQualifyNatNeeded	Pass	"Functionality Not covered. API Testing is covered"
bcmFieldQualifyNatSrcRealmId	Not tested	"Functionality Not covered. API Testing is covered"
bcmFieldQualifyNormalizeIpAdrs	Pass	Pass
bcmFieldQualifyNormalizeMacAdrs	Pass	Pass
bcmFieldQualifyOuterTpid	Pass	Pass
bcmFieldQualifyOuterVlan	Pass	Pass
bcmFieldQualifyOuterVlan	Pass	Pass
bcmFieldQualifyOuterVlanCfi	Fail	Pass
bcmFieldQualifyOuterVlanId	Pass	Pass
bcmFieldQualifyOuterVlanId	Pass	Pass

Table 143: Field Qualifiers

Qualifier Name	SDK-6.4.4 EA8 Test Status	SDK-6.4.4 Test Status
bcmFieldQualifyOuterVlanPri	Fail	Pass
bcmFieldQualifyPacketRes	Pass	"Functionality Not covered. API Testing is covered"
bcmFieldQualifyRangeCheck	Pass	Pass
bcmFieldQualifyRouterAlertLabelValid	Not tested	Not Tested
bcmFieldQualifyRtag7AHashLower	Pass	Pass
bcmFieldQualifyRtag7AHashUpper	Pass	Pass
bcmFieldQualifyRtag7BHashLower	Pass	Pass
bcmFieldQualifyRtag7BHashUpper	Pass	Pass
bcmFieldQualifySrcClassField	Fail	Pass
bcmFieldQualifySrcClassL2	Pass	Pass
bcmFieldQualifySrcClassL3	Not tested	Pass
bcmFieldQualifySrcGport	No Support	No Support
bcmFieldQualifySrcGports	Fail	No Support
bcmFieldQualifySrcIp	Pass	Pass
bcmFieldQualifySrcIp6	Pass	Pass
bcmFieldQualifySrcIp6High	Pass	Pass
bcmFieldQualifySrcMac	Pass	Pass
bcmFieldQualifySrcMimGport	Not tested	Pass
bcmFieldQualifySrcMimGports	Not tested	Not Tested
bcmFieldQualifySrcModPortGport	Pass	Pass
bcmFieldQualifySrcModPortGports	Not tested	Not Tested
bcmFieldQualifySrcModuleGport	Not tested	Pass
bcmFieldQualifySrcMplsGport	Pass	Pass
bcmFieldQualifySrcMplsGports	Not tested	Not Tested
bcmFieldQualifySrcNivGport	Not tested	Fail
bcmFieldQualifySrcNivGports	No Support	No Support
bcmFieldQualifySrcPort	Pass	Pass
bcmFieldQualifySrcTrunk	Fail	Fail
bcmFieldQualifySrcTrunkMemberGport	Not tested	Not Tested
bcmFieldQualifySrcVirtualPortValid	Not tested	Pass
bcmFieldQualifySrcVlanGport	Fail	Fail
bcmFieldQualifySrcVlanGports	No Support	No Support
bcmFieldQualifySrcVxlanGport	Not tested	Fail
bcmFieldQualifySrcVxlanGports	Not tested	Not Tested
bcmFieldQualifyStageIngress	Not tested	Pass
bcmFieldQualifyTcpControl	Pass	Pass
bcmFieldQualifyTos	Pass	Pass
bcmFieldQualifyTranslatedVlanFormat	Pass	Pass
bcmFieldQualifyTrillEgressRbridgeHit	Not tested	Not Tested
bcmFieldQualifyTrillIngressRbridgeHit	Not tested	Not Tested
bcmFieldQualifyTtl	Pass	Pass
bcmFieldQualifyTunnelType	Not tested	Not Tested
bcmFieldQualifyVlanFormat	Pass	Pass
bcmFieldQualifyVlanTranslationHit	Pass	Pass
bcmFieldQualifyVnTag	Fail	Pass
bcmFieldQualifyVpn	Not tested	Not Tested
bcmFieldQualifyVrf	Pass	Pass
bcmFieldQualifyVxlanFlags	Not tested	Pass

Table 143: Field Qualifiers

Qualifier Name	SDK-6.4.4 EA8 Test Status	SDK-6.4.4 Test Status
bcmFieldQualifyVxlanHeaderBits56_63	Not tested	Not Tested
bcmFieldQualifyVxlanHeaderBits8_31	Not tested	Not Tested