SDK 6.5.22 EA4

Release Notes

Table of Contents

No table of figures entries found.

Section 1: About This Document

This document contains the release notes for DNX devices affected by the Broadcom network switching Software Development Kit (SDK) release 6.5.22 EA4.

The document provides a general description of the SDK and its new features. It also describes the DNX chips supported by the release, BCM API additions or changes, resolved issues, and any relevant open issues.

Only new features are described in this document. For a comprehensive review of the DNX SDK features and issues, refer to earlier release notes for SDK 6.5.x.

For the full resolved list (Both Bugs and Improvement), please reference the file SDK-6.5.22-EA4-Resolved-Issues-Improvements.xlsx in the RELDOCS directory in the release package.

Section 2: New Devices added to this release

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

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

Section 2.1: Supported DNX Switch Devices

Family Devices	***Description***
BCM8828X	Q2U - GA quality
BCM8880X	J2C - GA quality
BCM8848X	Q2A - GA quality
BCM8869X	J2 - GA quality
BCM8879X	Ramon - GA quality
BCM8868X	J+ - GA quality
BCM8837X/BCM8867X	JR - GA quality
BCM8827X	QUX - GA quality
BCM8847X	QAX - GA quality

Section 3: Information per Device

This release is an increment version for DPP, DNX, DNXF, DFE family devices.

The subsequent sections describe the increment in available features compared to 6.5.21, backward-compatible notes, major bug-fixes and known issues.

It is very important to carefully go over the release-notes prior to adapting a new release.

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

Section 3.1: DNX-Family

This section includes the following family devices:

- **BCM8869X-Family (Jericho2)**
- **BCM8880X/BCM8882X-Family (Jericho2C)**
- **BCM8848X-Family (Qumran2A)**
- **BCM8828X-Family (Qumran2U)**

Section 3.1.1: Important Notes

Before integrating the new release, review this section thoroughly.

JIRA	Module	Description	Affected Devices
-	SRv6	Important changes were done in SRv6 applications that are not backward compatible. Please see more information in RN of SDK-234338 and SDK-242856	88480_A0, 88480_B0, 88690_B1, 88800_A0

Section 3.1.3.1: Backward Compatible Important Notes

SW Compatibility Guidelines 6.5.21 to 6.5.22-EA4

Please go over the list carefully.

Note: This document is written with the assumption that upgrade is done from 6.5.21 to 6.5.22 EA4. In case upgrade is done from older releases, users must first go over previous release notes.

JIRA	Module	Description	Affected Devices
SDK-242856	SRv6	SRv6 Layer is now collapsed to IPv6 layer in case of SRH.SL=0. Following changes done due to that: 1. Packet structure such as IPv4(or any payload)oSRv6(SL=0)oIPv6oETH used to create in previous releases layer protocol stack of: 0:ETH, 1:IPv6, 2:SRv6_Beyond, 3: IPv4 now it will create: 0:ETH, 1:IPv6, 2:IPv4 2. The indication for SRv6 with SL=0 is set on the IPv6 qualifier. The IPv6 first additional header enumerator now has two new values: IPv6_SRv6, and IPv6_SRv6_SL_0 Note, in case the first additional header is IPv6_SRv6_SL_0, the rest of the IPv6 qualifier bits 7-13 are changed and now include: <field name="Nof_Sids_Single_Pass_Term" bits="7:7"/> <field name="Nof_Sids_extended_Pass_Psp_Usp" bits="8:8"/> The above has no affect on the bcm_tunnel_terminator_create API, as it handles correctly both additional headers, in case the TunnelType is bcmTunnelTypeSRv6 or bcmTunnelTypeEthSR6. But it does require changes on the SRv6 Field-Processor applications. 3. The application reference code has updated accordingly, it now Identifies the SRv6 USP/USD flows with the updated IPv6 qualifier as a preselector-qualifier instead of the SRv6_Beyond layer protocol and its qualifier. 4. Note, in order to support the USD termination in 2-pass process, for a packet of ETHoSRv6oIPv6oETH (L2VPN tunnel termination), it is required to call a new API: bcm_switch_control_set(unit, bcmSwitchIngressVlanEditClassNull, 1); This is in order to allocate the last IVEC value to skip the FHEI IVEC append to system headers, and thus	88480_A0, 88480_B0, 88690_B1, 88800_A0

		keeping termination size optimal. Once this is enabled, number of Ingress VLAN edit class reduced by 1. Note: It is not required from the user to configure that IVEC value in addition to the above API call. 5. USP flow is changed and now must have two In-LIFs involved. First LIF must be created (bcm_tunnel_terminator_create) with tunnelType bcmTunnelTypeSR6, and the second must be in lower priority, and with tunnelType bcmTunnelTypeIpAnyIn6, to match the packet after the SRv6 extension is terminated. This is considered as a bug, and has an Errata under SDK-246424. Flow will be changed in next release. 6. For SRv6 Endpoint flow (SL!=0), the Layer above SRv6_Beyond is not set anymore on the layer protocols array (in previous release it was set). Note: SRv6.Next-header (field in packet) is available in PMF qualifier instead.	
SDK-242856	RCH	The API bcm_l2_egress_create had an input parameter dest_encap_id, and an input flag BCM_L2_EGRESS_DEST_ENCAP_ID, both were not used in the API configuration but the customer could provide them. Now, SDK returns error when they are used. The RCH.Outlif field is always getting the value of Outlif_0 of the current ETPP pass (which builds the RCH) and not according to API configuration. For SRv6 Extended T.Encap flow: A new configuration is now added to indicate if to update the OutLIF-Stack in the second pass of the ingress-pipe, by setting flag BCM_L2_EGRESS_EXTENDED_COPY_DEST_ENCAP.	88480_A0, 88480_B0, 88690_B1, 88800_A0
SDK-242675	OAMP	LMM/SLM packet can not be transmitted from OAMP to NIF (Network InterFace) because its length is too short. LMM/SLM packets will now come with zero padding when sending from OAMP.	88480_B0, 88690_B1, 88800_A0
SDK-242668	RCH	In the previous release, RCH processing of extended encapsulation uncollapse (bcmPortControlRecycleAppExtendedEncapUncollapse) was handled incorrectly at the 2nd pass. Instead of doing regular processing of PP in the 2nd pass, the application used a dedicated 2ND pass DBs (as used in the drop and continue application). The behavior is incorrect and now fixed. The processing is now doing the normal termination processing.	88480_B0, 88690_B1, 88800_A0
SDK-241829	Multicast	APIs unsupported flag are now enforced. The following flags are unused and deprecated. With this fix, multicast APIs will return an error if someone trying to use them: BCM_MULTICAST_TYPE_L2 BCM_MULTICAST_TYPE_L3	88480_A0, 88480_B0, 88690_B1, 88800_A0

		BCM_MULTICAST_TYPE_VPLS BCM_MULTICAST_TYPE_SUBPORT BCM_MULTICAST_TYPE_MIM BCM_MULTICAST_TYPE_WLAN BCM_MULTICAST_TYPE_VLAN BCM_MULTICAST_TYPE_TRILL BCM_MULTICAST_TYPE_NIV BCM_MULTICAST_TYPE_EGRESS_OBJECT BCM_MULTICAST_TYPE_L2GRE BCM_MULTICAST_TYPE_VXLAN BCM_MULTICAST_TYPE_EXTENDER BCM_MULTICAST_TYPE_MAC BCM_MULTICAST_TYPE_FLOW BCM_MULTICAST_TYPE_MASK BCM_MULTICAST_DISABLE_SRC_KNOCKOUT BCM_MULTICAST_INGRESS_MMC_BUFFERS	
SDK-240924	L2	In shell diagnostic commands "l2 add" and "l2 clear", an input argument "tgid" may be used. "tgid" should be encoded according to BCM_TRUNK_ID_SET() and not only the group-ID. This is now enforced and the command will return an error in case tgid is not properly encoded. Error information like “Input parameter tgid is not properly encoded. Please use BCM_TRUNK_ID_SET() to encode” will be invoked.	88480_A0, 88480_B0, 88690_B1, 88800_A0
SDK-240116	PMF	Remove support from SDK for 'field key info' diagnostic. Same diagnostic can be seen using "field context info".	88480_A0, 88480_B0, 88690_B1, 88800_A0
SDK-239082	PMF-Diag	Removed support from SDK for PMF-diagnostics 'field port'. Instead, Data can be found in the several DBAL tables: for TM ports: DBAL tables FIELD_PMF_PTC_INFO_IPMF1, FIELD_PMF_PTC_INFO_IPMF3, FIELD_PMF_PTC_INFO_IFWD, FIELD_PMF_OUT_TM_PORT_INFO_EPMF for PP ports: DBAL tables: FIELD_PMF_PP_PORT_INFO_IPMF1, FIELD_PMF_PP_PORT_INFO_IPMF3 and EGRESS_PP_PORT field FIELD_PMF_PP_PORT_INFO_IFWD for LIF ports; InLIF DB (e.g. IN_AC_INFO_DB, IN_LIF_FORMAT_PWE, IN_LIF_FORMAT_EVPN, IN_LIF_FORMAT_LSP, IN_LIF_IPvX_TUNNELS, INNER_ETH_VLAN_EDIT_CLASSIFICATION, INNER_ETH_VLAN_EDIT_CLASSIFICATION_VLAN_EDIT)	88480_A0, 88480_B0, 88690_B1, 88800_A0
SDK-239080	PMF-Diag	Remove support from SDK for 'field system' and 'field key last' diagnostics. "Field system" shows the context selection rules, same can be achieved by using "field context list" diagnostic "field key last" shows the key that was constructed for last packet, same can be	88480_A0, 88480_B0, 88690_B1, 88800_A0

		achieved using "field last info" diagnostic	
SDK-234248	VLAN-Port	BCM_VLAN_PORT_VLAN_TRANSLATION_TWO_VLAN_TAGS is obsolete and removed because now all AC formats support two VLAN TAGs. In case application used BCM_VLAN_PORT_VLAN_TRANSLATION_TWO_VLAN_TAGS , do same sequence just without BCM_VLAN_PORT_VLAN_TRANSLATION_TWO_VLAN_TAGS flag.	88480_A0, 88480_B0, 88690_B1, 88800_A0
SDK-227881	WIDE-DATA	API bcm_field_presel_set, qual_type=bcmFieldQualifyAcInLifWideData - Extend the in-lif wide data size that is used for IPMF1 context selection from 1 (msb) bit to 8 (msb) bits. Please note, that the mask value meaning is changed. mask=0x1, used to indicate the msb bit of the in-lif wide data. After the change, mask=0x80 indicate on the msb bit.	88480_A0, 88480_B0, 88690_B1, 88800_A0
SDK-210913	TDM	The default context ID for all ILKN channels is 0. From this release, context_id 0 now is invalid (traffic is drop) instead of being a valid one. When new ports are set, by default they will have invalid context_id and will not forward traffic until a valid context will be assigned to them. Set a valid TDM context ID by calling API bcm_tdm_ingress_context_create with context_id > 0.	88480_A0, 88480_B0, 88800_A0
SDK-193159	VXLAN	VXLAN L2VPN QOS now supports egress native PCP&DEI remark profile of IP tunnel. In previous releases, QOS was determined by remark profile 0 only.	88690_B1
SDK-234338	SRv6	SRv6 End point PSP is now supported per packet, and not only as a global mode. The classification of a packet as a PSP packet can be done per next DIP. Due to that, significant change in Application Reference was done in iPMF. See SRv6 UM and appl_ref changes for more information	88480_A0, 88480_B0, 88690_B1, 88800_A0
-	DBAL	DBAL Field Enums types are changed from this release: 1. There is no more defines for max value of enum (DBAL_NOF_ENUM_*). Instead use dbal_fields_max_value_get API. 2. dbal_enum_value_result_type_* do not exist anymore. Instead use dbal_result_type_t. 3. DBAL_NOF_RESULT_TYPE_* do not exist anymore. Instead use API dbal_tables_table_nof_res_type_get with the DBAL table name.	88480_A0, 88480_B0, 88690_B1, 88800_A0
SDK-218577	MPLS	The explicit null Label and other reserved MPLS labels can be terminated per port (VLAN domain) as the general labels by the following configurations: 1. call bcm_switch_control_indexed_set() with index "bcmSwitchMplsSpecialLabelAutoTerminateDisable" to disable auto termination for a particulate label; 2. call bcm_mpls_tunnel_switch() to add the label termination entry. The input parameter "port" indicates the VLAN domain.	88480_A0, 88480_B0, 88690_B1, 88800_A0

		Note, in this case, the special label should be BoS. In addition, all the reserved MPLS labels (0~15) cannot be configured as general labels. In previous release, the functionality was global, now it is per Incoming-port (VLAN-Domain).	
SDK-238679	L2	From this release, MACT/FID limit is not enforced for static entries insertion.	88690_B1

Section 3.1.4: SDK build & load

Compile and set config files :

setenv SDK `pwd`

Example of Intel GTS CPU compilation :

```
# Copy pre compiled mdb and kaps libraries into the relevant build folder.
# For Intel GTS CPU 64b build flavor, Following are the relevant 2 libraries and the
#  relevant build folder (names in build folder must be libkaps.a & libmdb.a):
mkdir -p $SDK/build/unix-user/x86-64-fc28/
cp  $SDK/libs/bin/dnx/GTS_64B_libkaps.a  $SDK/build/unix-user/x86-64-fc28/libkaps.a
cp  $SDK/libs/bin/dnx/GTS_64B_libmdb.a   $SDK/build/unix-user/x86-64-fc28/libmdb.a
# Additional mdb and kaps libraries flavors can be found under $SDK/libs/bin/.

# Compile SDK
cd $SDK/systems/linux/user/x86-64-fc28/
make -j 5 MAKE_LOCAL=$SDK/make/local/dnx/Make.custom.gts
```

Common config files:

```
ln -fs $SDK/rc/rc.soc
ln -fs $SDK/rc/dnx.soc
ln -fs $SDK/rc/config-jer2pemla.bcm
ln -fs $SDK/tools/sand/db
ln -fs $SDK/rc/dnx_sku
ln -fs $SDK/rc/dnx_dram
ln -sf $SDK/rc/cmicfw/linkscan_led_fw.bin
ln -sf $SDK/rc/cmicfw/custom_led.bin
```

BCM8869X specific links:

```
ln -fs $SDK/rc/config-jr2.bcm config.bcm
```

```
ln -fs $SDK/rc/bcm88690_revB_board.bcm
ln -sf $SDK/rc/bcm88690_board.bcm
ln -sf $SDK/rc/bcm88690_legacy_interop_board.bcm
```

BCM8880X/BCM8882X specific links:

```
ln -fs $SDK/rc/config-j2c.bcm config.bcm
ln -sf $SDK/rc/bcm88800_board.bcm
```

BCM8848X/BCM8828X specific links:

```
ln -fs $SDK/rc/config-q2a.bcm  config.bcm
ln -fs $SDK/rc/bcm88480_board.bcm
```

Run :

```
./bcm.user
```

Section 3.2: DNXF-Family (BCM88790-Family)

Section 3.2.1: Important Notes

Before integrating the new release, review this section thoroughly.
None

Section 3.2.1.1: Backward Compatible Important Notes

SW Compatibility Guidelines 6.5.21 to 6.5.22-EA4

Please go over the list carefully.

Note: This document is written with the assumption that upgrade is done from 6.5.21 to 6.5.22-EA4. In case upgrade is done from older releases, users must first go over previous release notes.

JIRA	Module	Description	Affected Devices
SDK-238031	Access	Remove "access" shell command	88790

Section 3.3: DPP-Family - BCM88670/680/470/270 Family GA Release

This release is for:

- BCM88670 (Jericho) family product lines.
- BCM88270 (QUX) family product line
- BCM88470 (QAX) family product line
- BCM88680 (Jericho+) family product line

The subsequent sections describe the increment in available features compared to 6.5.21, major bug-fixes and known issues. Before integrating the new release, review the “Backward compatible important notes” section.

Section 3.3.1: Important Notes

Before integrating the new release, review this section thoroughly.

None

Section 3.3.2: Backward Compatible Important Notes

SW Compatibility Guidelines 6.5.21 to 6.5.22-EA4

Note: This document is written with the assumption that upgrade is done between 6.5.21 to 6.5.22-EA4. In case upgrade is done from earlier releases to 6.5.21, it must first go over previous SDK release notes.

None

Section 3.4: DFE-Family - BCM88770 (FE3600) Release

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

This is a sustaining release.

Section 4: Compatibility

Section 4.1: SDK and PCIe FW Compatibility

Below table shows the firmware version compatible with the SDK release.

Switch SDK Release	*PCIe FW Release*
6.5.22-EA4	2.5.4

Section 4.2: SDK and KBPSDK lib Compatibility

Below table shows the KBPSDK lib compatible with the SDK release.

Switch SDK Release	*KBPSDK lib*
6.5.22-EA4	1.15.4.1