SDK 6.5.26 EA1

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 6.5.26 EA1.

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-6.5.26-EA1-Resolved-Issues-Improvements.xlsx in the RELDOCS directory in the release package.

Section 2: Devices supported in 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/BCM8882X		J2C - GA quality
BCM8848X		Q2A - GA quality
BCM8869X		J2 - GA quality
BCM8879X		Ramon - GA quality
BCM8868X		J+ - GA quality
BCM8837X/BCM8867X		JR - GA quality
BCM8866X		ARAD+ - GA quality
BCM8827X		QUX - GA quality
BCM8847X		QAX - GA quality
BCM8829X		Q2n - GA quality

Section 2.2: Preview DNX Switch Devices

Family	*Devices*	*Description*
BCM8883X		J2X - Pre quality

Section 3: Information per Device

This release is an incremental version for DPP, DNX, DNXF, DFE family devices.
The subsequent sections describe the increment in available features compared to 6.5.25, 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)**
- **BCM8829X-Family (Qumran2N)**
- **BCM8883X-Family (Jericho2X)**

Section 3.1.1: Important Notes

Before integrating the new release, review this section thoroughly.

JIRA	Module	Release-note	Affected Devices
SDK-246890	SKU	As part of the SDK release, a reference configuration for DNX devices is provided (config*.bcm). When an SKU cannot be loaded with the provided reference configuration (in most cases due to ports configuration) the SDK provides a configuration file that adjusts the reference configuration so it can be loaded with the SKU. What is changing? * SKUs configuration files were provided in an rc script format. Starting this release, the configuration will be provided in a config file format. * SKUs were separated into a file per SKU folder. Starting this release, all SKUs will be in a single file config-skus.bcm. Backward compatibility: As the dnx_sku.soc loading was removed from the default soc files, old format soc files won't be automatically loaded. In order to load SKU in an rc format (in case taken from an older release), a load command should be placed explicitly in dnx.soc	88480_B0, 88690_B0, 88790, 88800_A0

		("rcload <sku_file_name>"). Note: there is no specific action required from the user and each SKU will be able to load with the default configuration as before now.	

Section 3.1.1.1: Backward Compatible Important Notes

SW Compatibility Guidelines to 6.5.26-EA1

Please go over the list carefully.

Note: This document is written with the assumption that upgrade is done from 6.5.25. In case upgrade is done from older releases, users must first go over previous release notes.

JIRA	Module	Release-note	Affected Devices
SDK-282615	GTP	In the previous release, for UPF/GTP packets, the 8 bytes "PDU session container" extension header was always encapsulated. In this release, the 8 bytes "PDU session container" extension is no longer available. Following options are now available from this release: 4 bytes "PDU session container" extension header or without any extension header. The following two fields are available: type and RQI field. Can be set via special fields "GTP_EXT_SSN_TYPE" and "GTP_EXT_SSN_RQI".	88480_B0, 88690_B0, 88800_A0
SDK-276863	TUNNEL-IPV4, Field	Parsing packets in the following format IP-GREoMPLSxnoE (where 1<=N<=4) were wrongly set with bcmFieldQualifyIpTunnelType RAW and not GRE4 as it should be. User needs to adjust Tunnel Type values accordingly for such packets.	88480_B0, 88690_B0, 88800_A0
SDK-246424	SRv6	SRv6 USP flow is now changed for packet formats: ICMP/UDP/TCP o SRv6oIPv6oETH In previous releases, when calling bcm_tunnel_terminator_create() with tunnel types: bcmTunnelTypesSR6, bcmTunnelTypesEthSR6 or with flags: BCM_TUNNEL_TERM_UP_TO_96_LOCATOR_SEGMENT_ID, BCM_TUNNEL_TERM_UP_TO_64_LOCATOR_SEGMENT_ID, BCM_TUNNEL_TERM_GENERALIZED_SEGMENT_ID,	88480_B0, 88690_B0, 88800_A0

		BCM_TUNNEL_TERM_MICRO_SEGMENT_ID - SDK allowed Termination even in case the next layer is: UDP/TCP/ICMP. Application reference code was written to identify the packets that points to layer type UDP/TCP/ICMP, with a terminated SRv6 header - and perform a USP flow over those packets. From this release, there is no special tunnel-termination behavior for the mentioned tunnel types and input flags. A new trap is hit in that case, bcmRxTrapSrv6Usp Application reference code was updated to perform USP flow over packets that hit the above trap code. Users are expected to configure the trap in the forwarding DBs - based on the DIP (last DIP of an SRH list) For more details: An update of the reference application is done, and CINTs too. For more details see SRv6 UM, USP Flow section	
SDK-281788	NIF	FEC NONE cannot be used for NIF ETH port with the following speed: 50G - 1 lane, 100G - 2 lanes, 200G - 4 lanes	88480_B0, 88690_B0, 88800_A0
SDK-277702	ERSPAN	In the previous release, GRE "Protocol Type" for ERSPANv3 was set to 0x88BE by mistake, In this release, it has been fixed as 0x22EB.	88480_B0, 88690_B0, 88800_A0
SDK-275448	TDM	TDM traffic rarely gets stack due to TXQ overflow. This fix eliminates the phenomena. With this fix TDM and non-TDM traffic are no longer allowed to pass through the same egress queue.	88480_B0, 88800_A0
SDK-283675	QOS-ECN	In the previous release, classic ECN packets with ECN bits b'01 were classified and remarked as L4S ECN incorrectly. In this release, this issue has been addressed by correcting ecn_eligible and CNI bits mapping.	88480_B0, 88800_A0

SDK-285979	GTP	The Flow gport type changed from Tunnel to FLOW_LIF, for example the use of BCM_GPORT_TUNNEL_TO_L3_ITF_LIF is not valid anymore.	88480_B0, 88690_B0, 88800_A0
SDK-280728	OAM	In the previous release, Loopback and TST allocated a wrong trap id and added it to SAT trap within SDK. Cint configuration was incorrect and others applications may have caused SAT trapping to fail. In this release, this issue has been addressed. In order to make those applications functional, the trap code of Loopback and TST should be allocated and configured by user. Note: For Up MEP, the trap code bcmRxTrapEgTxOamUpMEPDest2 must be used in order to trap LB or TST packet to SAT port. For other types MEP, trap code bcmRxTrapUserDefine must be used. User has to do the following SAT configuration after creating trap id for MEP: 1, Add SAT trap by bcm_sat_ctf_trap_add(). 2, Map trap qualifier to session id by bcm_sat_ctf_trap_data_to_session_map(). 3, Configure CTF identification with trap, session id, tc and dp by bcm_sat_ctf_identifier_map(). The above steps need to be added to configurations of bcm_oam_loopback_add() and bcm_oam_tst_rx_add(). For reference, see how they are used in the cint dnx_oam_set_tst_trap()	88690_B0
SDK-238018	OAMP	In the previous release, DM features is supported through soc property "up_mep_tod_recycle_port_core_x=<recycle-port-x>". In this Soc property up_mep_tod_recycle_port_core_x cannot longer be used, instead user has to configure the recycle port by bcm_oam_control_indexed_set() rather than the soc, the type is bcmOamControlUpMepDmRecyclePort, index is core_id and arg is recycle port.	88480_B0, 88690_B0, 88800_A0
SDK-281853	L3-ECMP	The EUDP (ECMP User-defined profile) was updated as follows: 1. An ECMP can now be updated (using the BCM_L3_REPLACE flag) with a new EUDP profile. 2. The bcm_l3_egress_ecmp_get returns the	88480_B0, 88690_B0, 88800_A0,

		BCM_L3_ECMP_USER_PROFILE in case the ECMP is using a EUDP. 3. The bcm_l3_egress_ecmp_get returns only the base FEC and not an array of FECs in case it uses a EUDP same as in the bcm_l3_egress_ecmp_create API with an EUDP. 4. When creating an ECMP the max_paths field must be set to 0 instead of being forced to be a positive number as this value isn't affecting the ECMP. 5. When creating an ECMP, the RPF mode and the tunnel priority mode must be set to default as the EUDP already holds this information, when getting this ECMP these fields will also be set to the default state and the actual state could be retrieved from the EUDP get API.	
SDK-284063	NIF	"nif tx" and "nif rx" format was changed: - "nif tx" diagnostic is now "nif tx show" - "nif rx" diagnostic is now "nif rx show"	88480_B0, 88690_B0, 88800_A0
SDK-263579	L2-Learning	To accommodate the change of the entry learn strength, the learn data was updated Please see more details in RN excel SDK-263579. Any Field-Processor applications that use to quality/modify the Learn-data directly (Qualifiers/Actions LearnPayload) need to be adjusted accordingly.	88480_B0, 88690_B0, 88800_A0

Section 3.1.2: 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

Example of CMODEL compilation:
Copy pre compiled mdb and kaps libraries into the relevant build folder.
For CMODEL 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-linux-64-cmodel/
cp $SDK/libs/bin/dnx/CModel_libkaps.a $SDK/build/unix-linux-64-cmodel/libkaps.a
cp $SDK/libs/bin/dnx/CModel_libmdb.a $SDK/build/unix-linux-64-cmodel/libmdb.a
Additional mdb and kaps libraries flavors can be found under $SDK/libs/bin/.

Compile SDK
cd $SDK/systems/sim/dpp
make -j 5 MAKE_LOCAL=$SDK/make/local/dnx/Make.pkg.dnx_only_sim_cmodel
target_suffix=-cmodel

Common config files:
ln -fs $SDK/rc/rc.soc
ln -fs $SDK/rc/dnx.soc
ln -fs $SDK/rc/jer2pemla-ucode.bcm
ln -fs $SDK/tools/sand/db
ln -fs $SDK/rc/config-skus.bcm
ln -fs $SDK/rc/dnx_dram
ln -fs $SDK/rc/cmicfw/linkscan_led_fw.bin
ln -fs $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 -fs $SDK/rc/bcm88690_board.bcm
ln -fs $SDK/rc/bcm88690_legacy_interop_board.bcm

BCM8880X/BCM8882X specific links:
ln -fs $SDK/rc/config-j2c.bcm config.bcm
ln -fs $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

BCM8883X specific links:
ln -fs $SDK/rc/config-j2x.bcm config.bcm
ln -fs $SDK/rc/bcm88830_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.
See Section 3.1.1: Important Notes

Section 3.2.1.1: Backward Compatible Important Notes

SW Compatibility Guidelines 6.5.25 to 6.5.26-EA1

Please go over the list carefully.

Note: This document is written with the assumption that upgrade is done from 6.5.25 to 6.5.26-EA1. In case upgrade is done from older releases, users must first go over previous release notes.

None

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 6.5.25, 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 6.5.25 to 6.5.26-EA1

Note: This document is written with the assumption that upgrade is done from 6.5.25. In case upgrade is done from earlier releases to 6.5.25, it must first go over previous SDK release notes.

JIRA	Module	Release-note	Affected Devices
SDK-279349	CRPS	Counters could not be incremented for large queue IDs when using VOQ counting. This is now fixed by selecting the range.start of the first counter engine configured for VOQ counting as the VOQ count base. The VOQ count base is shared by all the counter engines which are configured for VOQ count, thus it should be changed in the middle. The VOQ counter engines are selected based on (range.start - voq_count_base)/counter_sets_per_counter_processor. Before this change, the VOQ count base by default is 0, and might be changed later on which fails the counting on those engines previously configured silently. After this change, VOQ count base can only be changed after all the counter engines configured for VOQ counting are removed.	88270_A0, 88470_B0, 88670_B0, 88680_A0

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.