Verification Continuum™

VC Verification IP
USB
Release Notes

Version V-2023.09, September 2023

SYNOPSYS®

Copyright Notice and Proprietary Information

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at http://www.synopsys.com/company/legal/trademarks-brands.html.
All other product or company names may be trademarks of their respective owners.

Free and Open-Source Software Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

www.synopsys.com

Contents

1 Product Updates

The changes for version V-2023.09 of VC VIP for USB supports verification of SoC designs that incorporate USB interfaces.

Topics in this chapter as follows:

1.1 Notes for V-2023.09 Release

- USB_SVT Version: V-2023.09
- MPHY Version: V-2023.09
- SVT Version: U-2023.03
- Common Version: V-2023.09

Updates for this release is as follows:

- VIP is compatible with IEEE UVM 1800.2-2020-2.0 version for VCS U-2023.03-SP1 onwards versions.

1.2 Downloading Installation Run Files

Attention The Electronic Software Transfer (EST) system only displays products your site is entitled to download. If the product you are looking for is not available, contact est-ext@synopsys.com.

Perform the following instructions for downloading the software from Synopsys. You can download from the Download Center using either HTTPS or FTP, or with a command-line FTP session. If your Synopsys SolvNetPlus password is unknown or forgotten, go to http://solvnetplus.synopsys.com.

Passive mode FTP is required. The passive command toggles between passive and active mode. If your FTP utility does not support passive mode, use http. For additional information, see the following web page:

https://www.synopsys.com/apps/protected/support/EST-FTP_Accelerator_Help_Page.html

1.2.1 Downloading From the Electronic Software Transfer (EST) System (Download Center)

a. Navigate to http://solvnetplus.synopsys.com.
b. Enter your Synopsys SolvNetPlus Username and Password.
c. Click Sign In button.
d. Make the following selections on SolvNetPlus to download the `.run` file of the VIP.
   i. Downloads tab
   ii. VC VIP Library product releases
   iii. <release_version>
   iv. Download Here button
   v. Yes, I Agree to the Above Terms button
   vi. Download `.run` file for the VIP
e. Set the `DESIGNWARE_HOME` environment variable to a path where you want to install the VIP.

```
% setenv DESIGNWARE_HOME VIP_installation_path
```

f. Execute the `.run` file by invoking its filename. The VIP is unpacked and all files and directories are installed under the path specified by the `DESIGNWARE_HOME` environment variable. The `.run` file can be executed from any directory. The important step is to set the `DESIGNWARE_HOME` environment variable before executing the `.run` file.

1.3 Downloading Using FTP with a Web Browser

a. Follow the above instructions through the product version selection step.
b. Click the "Download via FTP" link instead of the "Download Here" button.
c. Click the "Click Here To Download" button.
d. Select the file(s) that you want to download.
e. Follow browser prompts to select a destination location.

1.4 Licensing Information

The CSI-2 product is enabled by features defined below and in the order listed. Once a required feature or a set of features are successfully checkout, the VIP stops looking for other licenses.

Licensing is required if the VIP component classes are instantiated in the design. This includes envs, agents, drivers, monitors, sequencers, and components in UVM and OVM. This includes groups, subenvs, and transactors in VMM.

Perform the VIP License check order and feature names as per the following steps:

1. VIP-USB-SVT
2. VIP-SOC-LIBRARY-SVT

2 Known Issues and Limitations

2.1 Limitation

- Euclide lint check errors are expected in UVM 1.2 with `UVM_NO_DEPRECATED` macro.
- Common Encryption and INCDIR flow is not supported with Xcelium.

3 Supported Platforms, Models, and Software

This chapter discusses the supported platforms, models, and software of USB.

3.1 Supported Methodology, OS and Simulator Versions

This version of VIP is qualified with version T-2022.09 of SystemVerilog Verification Technology (SVT). SVT is an internal portion of the VIP and provides base VIP functionality, some utilities, and support for installation and licensing.

This version supports SVT T-2022.09 and later. The version of SVT is the key for determining which versions of platform or OS and simulators have been qualified with this VIP. Whenever a new version of SVT is released, this VIP is qualified with it.

To determine the version of SVT in an existing `DESIGNWARE_HOME` installation, use:

```
$DESIGNWARE_HOME/bin/dw_vip_setup -i home
```

Note

The following syntax previously allowed by NCV is now not allowed in incisiv 14.20.* versions:
`this.<rand_arr_var>.rand_mode();`
While using NCV simulator versions < incisiv 14.20.* versions, if error is reported then use the following:
`+define+SVT_MULTI_SIM_RAND_MODE_AS_FUNCTION_ON_ARRAY`

3.1.1 Supported Methodology Versions

The simulator matrix table is available on SolvNetPlus at the following location:

https://spdocs.synopsys.com/dow_retrieve/latest/vg/snps_vip_lib/PDFs/simulator_matrix.pdf

For more information on the simulator matrix and library level updates, see VC VIP Library Release Notes.

Note

MTI is not supported.

Table 3-1 Supported Methodologies With Simulators

Methodology	VCS	Xcelium	Questasim
UVM	Supported	Supported	Supported

Methodology	VCS	Xcelium	Questasim
OVM	Supported	Supported	Not supported
VMM	Supported	N/A	N/A

Table 3-2 Supported Methodology Versions

Methodology	Supported Version	Unsupported Version
OVM	2.1.2	2.1.1_3
UVM	1.1d, 1.2, 1800.2.2017-1.0, 1800.2-2017-1.1	1.1a
VMM	1.1, 1.2	NA

3.2 Verdi Version

The USB VIP supports Verdi.

4 Documentation

VIP documentation is available in both HTML and PDF formats. Both the formats can be accessed from VIP installation area or SolvNetPlus.

Access VIP documentation from the following:

- VIP Installation Location
- SolvNetPlus

4.1 VIP Installation Location

After the VIP is downloaded and installed, product documentation is available in the following location:

$DESIGNWARE_HOME/vip/svt/<vip_title>/latest/doc/

- Accessing VIP Documentation Using SmartSearch
- PDF Documentation
- Class Reference Documentation
- HTML Documentation

4.1.1 Accessing VIP Documentation Using SmartSearch

VIP documentation can be accessed using the SmartSearch tool. This tool is an advanced search engine for retrieving information from the VIP documentation.

VIP SmartSearch provides the following capabilities:

- Advanced information search techniques that enables you to find relevant information faster
- Uses natural language for search queries
- Learns from user interaction and maintains history of searches

For more information on accessing SmartSearch for VIP documentation, see the https://solvnetplus.synopsys.com/s/article/VC-VIP-Integrating-SmartSearch-with-VIP

- For S-2021.09 onwards, install VIP SmartSearch S-2021.09 version.
- To use SmartSearch for previous documentation release (S-2021.06), install VIP SmartSearch S-2021.06 version.

4.1.2 PDF Documentation

The pdf folder lists the following documents:

- Release Notes
- Getting Started
- User Guide
- FAQ

Access PDF files from the following location:

$DESIGNWARE_HOME/vip/svt/<vip_title>/latest/doc/PDFs

You can view these files using a PDF reader in Linux or Windows.

Note To share documentation feedback with the Technical Publications team, click the Feedback link located in the footer of the PDF pages.

4.1.3 Class Reference Documentation

This on-line help contains information about the classes, functions, and member variables. This document lists class hierarchy and contents and it provides links you can use to navigate for more details.

Access class reference html documentation from the following location:

$DESIGNWARE_HOME/vip/svt/<vip_title>/latest/doc/class_ref

The HTML documentation is categorized as follows:

Table 4-1 Class Reference

Topics	Description
Main	Important information about the model being documented.
Classes	• For products other than svt, classes are categorized as configuration, agent, driver, monitor, sequencer, transaction, coverage, callbacks, sequences, exceptions, status, env and others(as applicable to each vip and methodology) • For svt class reference, classes are categorized as SVT and methodology classes.
Macros	Alphabetically keyed list of all system verilog macros.
Covergroups	Descriptions of all Covergroups present.
Protocol Checks	Protocol Checks defined in a VIP.
Sequence Page	Lists all the Sequences defined in a VIP.
Modules	Descriptions of all the system verilog modules present.
Interfaces	Descriptions of all the system verilog interfaces present.
Files	List of all source files with links to the contents of these files.
Related Docs	List of related documentation pages for the model.
?	Help page.

4.1.4 HTML Documentation

The following documentation is available in HTML format:

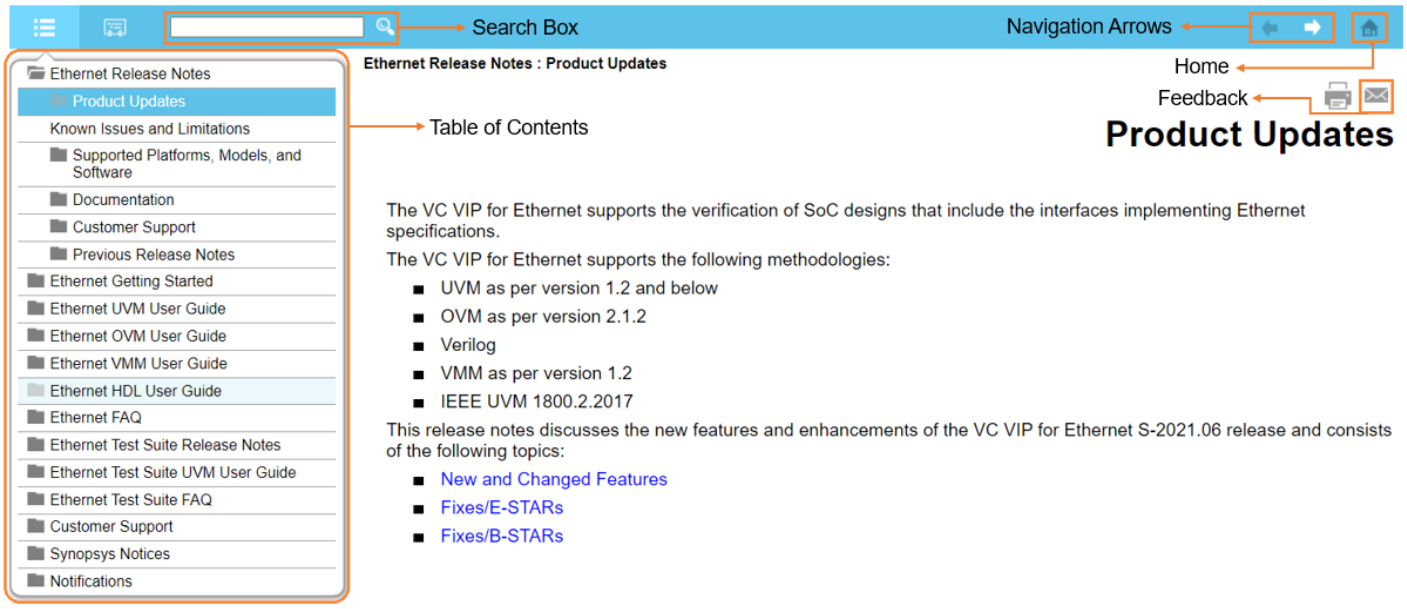
- Release Notes
- Getting Started
- User Guide
- FAQ

The html files can be viewed using a web browser in Linux or Windows. To view the HTML files, open the index.html file located in the *$DESIGNWARE_HOME/vip/svt/<vip_title>/latest/doc/user_doc_html* directory.

The key features of the HTML help are follows:

- View each guide by navigating through the bookmarks in the left pane.
- Search across the documentation by entering the search keywords in the Search box. The search is case-insensitive.
- The search returns the relevant instances from the VIP documentation collection.
- The following figure displays the HTML Help interface and the available options.

Figure 4-1 VIP HTML Help



Note To share documentation feedback with the Technical Publications team, click the Email icon.

4.2 SolvNetPlus

Access documentation by either entering keywords in the SolvNetPlus search bar or by navigating to the VIP documentation page.

For more information on accessing SolvNetPlus for VIP documentation, see the

https://solvnetplus.synopsys.com/s/article/Accessing-VC-VIP-Documentation-in-SolvNetPlus

For a seamless experience, it is recommended that you use Chrome, Firefox, and Safari browsers for accessing SolvNetPlus.

Access documentation from SolvNetPlus

- SolvNetPlus Articles
- VIP Offline Documentation

4.2.1 SolvNetPlus Articles

SolvNetPlus provides access to a collection of technical articles for all the VIPs. These articles provide useful information that can help you to troubleshoot and resolve issues, understand specific VIP use cases, and gain additional knowledge on VIPs.

To search for articles, go to the following URL and search for the topic of interest:

http://solvnetplus.synopsys.com

4.2.2 VIP Offline Documentation

To access product documentation, you are required to go to SolvNetPlus, but this requires online access. At times, you would like to download the product documentation so that you can access it on your local computer for situations where you do not have online access or where the network connection is limited or slow.

The latest collection of Synopsys VC VIP documentation is available for you to download for offline viewing. Each collection bundles the PDF files and PDF indexes of the user guides into a single, compressed tar file for each collection.

To download the compressed tar file for VC VIP, click here, and follow the instructions in the article.

For more information on offline documentation, see the VC VIP Offline Documentation article under Highlights section of VC Verification IP Documentation Landing page on SolvNetPlus.

5 Customer Support

This chapter discusses the customer support provided for USB.

This chapter consists of the following topics:

5.1 SolvNetPlus

Synopsys SolvNetPlus resides at the following location:

https://solvnetplus.synopsys.com/

It provides you with the following:

- Download Center for all VIPs
- Support
- Training
- Reference Methodology Retrieval System
- Hundreds of articles on VIP usage
- Register problem reports

5.2 Registering a Problem

Registering a Problem

To register a problem, perform any of the following tasks:

1. Go to https://solvnetplus.synopsys.com/ and open a case.
   Enter the information according to your environment and your issue.
2. Send an e-mail message to support_center@synopsys.com

   Include the Product name, Sub Product name, and Product version for which you want to register the problem.

5.3 Reporting a Problem

To report a problem, keep the following information ready before you contact technical support:

- Provide a description of the following:
  - The issue under investigation
  - Your verification environment
- Create a Value Change Dump (VCD) file.
- Generate a log file for the simulation.
- Provide other files such as translation logs.

For information on reporting a problem for each methodology, see the following sections in the respective user guide:

- The “xxx” section in Chapter A, “Appendix”, in the <*UVM userguide_name*>
- The “xxx” section in Chapter A, “Appendix”, in the <*OVM userguide_name*>
- The “xxx” section in Chapter A, “Appendix”, in the <*VMM userguide_name*>
- The “xxx” section in Chapter A “Appendix”, in the <*Verilog userguide_name*>

5.4 Telephone Support

Telephone your local support center:

- North America:

  Call 1-800-245-8005 from 7 AM to 5:30 PM Pacific time, Monday through Friday
- All other countries:

https://www.synopsys.com/support/global-support-centers.html

6 Previous Release Notes

This chapter lists the product update for previous releases.

6.1 Notes for U-2023.06-3 Release

- USB_SVT Version: U-2023.06-2
- MPHY Version: U-2023.06-1
- SVT Version: U-2023.03
- Common Version: U-2023.06-1

Updates for this release are as follows:

- Added a new configuration class attribute `enable_powerdown_change_on_block_completion` with default value 0. To enable the custom interface MAC to maintain the powerdown during an ongoing block set the configuration to 1.
- Updated the name of the `wait_for_specific_packet_transmission` task of `svt_usb_30_compliance_link_layer_system_virtual_sequence_collection` class to `wait_for_specific_packet_transmissions` as there is one existing task with different argument in parent class.
- Updated the group name of bins in `SS_ENDPOINT` to `SS_DESC`.
- Changed the direct low power u2 service execution with the task `u0_to_u2_ltssm_state_transition_with_retry` for the `ltssm_state_transaction` `LTSSM_STATE_TIMEOUT_DUE_TO_RX_LFPS_ON_NON_CONFIG_LANE`. This prevents the hanging of the test if ITP and low power execution is triggered simultaneously.
- Updated UTMI checker `chk_utmi_invalid_chirp_linestate`. Added `SVT_USB_20_USER_UTMI_LINESTATE_FILTER_DELAY` macro inside the checker. The macro value can be modified to avoid initiating the checker, whenever a glitch is introduced from the DUT.
- Update to hit UTMI checks `chk_utmi_after_resume_switch_to_hs`, `chk_utmi_linestate_invalid_resume_trans`, `chk_utmi_linestate_invalid_suspend_trans`, `chk_utmi_rx_start_delay_hs`, `chk_utmi_eop_se0_assertion`, `chk_utmi_linestate_during_chirp`, `chk_utmi_linestate_during_reset`, `chk_utmi_bitstuff_handshake` and `chk_utmi_rxactive_deassertion_to_idle`

- Updated the implementation of introducing the delay in LPMA with `delay_before_lpma_transmission` configuration. The delay is introduced before scheduling the LPMA, instead of initiating the Link to Physical transmission. This allows exceptional cases for packets to be scheduled amidst low power entry procedure. Added a new shared status attribute `ss_lp_entry_partially_confirmed` which gets high when LAU is received before scheduling the LPMA. Mapped the shared status attribute `ss_lp_entry_partially_confirmed` with the debug ports.
- The Example Home tab is updated in the Quick start guide of `tb_usb_svt_uvm_basic_sys` and `tb_usb_svt_uvm_intermediate_sys` example TB.
- Added three configuration attributes `enable_jitter_on_sof`, `enable_positive_jitter_on_sof`, and `enable_negative_jitter_on_sof`. Added Jitter in SOF micro frame timer. For High Speed the jitter can be of range /(-)96 bit times. the above attributes can be used to specify the type of jitter to be added. For Full-Speed this range can be as / 42 bit times. These range values have been defined in the user configurable macro (`SVT_USB_20_MAX_HS_SOF_JITTER_LENGTH`) and (`SVT_USB_20_MAX_FS_SOF_JITTER_LENGTH`) for HS and FS having default values as 96 and 42 respectively. Added the configuration attributes `usb_20_hs_sof_jitter_length` and `usb_20_fs_sof_jitter_length` having default value (`SVT_USB_20_MAX_HS_SOF_JITTER_LENGTH`) and (`SVT_USB_20_MAX_FS_SOF_JITTER_LENGTH`) respectively. The range has been taken from USB20 Spec section 11.2.1 and 11.2.2 for HS and FS respectively.
- Updated the VIP to reset header sequence number and buffer credit counts after U2/U3 exit as per USB4 V2 spec. Added debug signals for Gen T LCM(`svt_usb_link_ss_lcm`).
- Updated the implementation of the check `dpp_max_symbols_check` for SS mode to flag the scenario where DPPEND framing symbols are not detected after the expected payload byte count is received.
- Added a new serive `ENABLE_PCLK_IN_POWER_STATE_P3`, this service is used to enable the PCLK in P3 State
- Added the check `itp_not_received_in_tisochronoustimestampstart` in `svt_usb_protocol_ss_itp_processor`.
- Added a new service `DEASSERT_PHYSTATUS_FOR_RESET_BEFORE_PCLK_ENABLED` which is used to deassert the PhyStatus before enabling PCLK in P3 State.
- Updated UTMI checker `chk_utmi_sof_eop_invalid` implementation for the configuration when REMOTE Phy is driving SOF packet.
- Added a new service `INVALID_RXSTATUS_RSP_ON_RX_DETECTION`, this service is used to create invalid RxStatus response from the PHY during Receiver Detection.
- Added a new service `EXTEND_PHYSTATUS_RESPONSE_FOR_MULTIPLE_PCLK_DURING_RXDETECTION`, this service is used to extend the assertion of PhyStatus for multiple PCLK when RxStatus drives its value for single PCLK to appropriate code for Receiver Detection. Added a new attribute `count_of_extra_pclk_for_phystatus_asserted_in_rx_detect` to provide the number of extra PCLK for which the PhyStatus should be asserted.
- Removed the redundant check sub_type_ack_tp.
- Added a new check `unexpected_symbol_received_in_a_valid_framing_set_check` to flag the scenario where VIP detects an invalid symbol in the framing set. Since spec allows the tolerance of one symbol in framing sets, functionally VIP processes the received framing set. Added a new configuration class attribute `disable_usb_strict_framing_symbol_set_error_check` when set to 1, disables the `unexpected_symbol_received_in_a_valid_framing_set_check`. By default its value is set to 0 in the VIP.

- Added a new service `PCLK_TURNED_OFF_BFR_PHYSTATUS_RESP_FOR_POWERDOWN_P3`, this service is used to turn off the PCLK before PhyStatus response for the change in powerdown to P3. Added a new attribute `delay_between_pclk_turned_off_and_phystatus_resp_for_powerdown_change_to_p3`, this attribute is used to pass the delay value between the PCLK disabled and the PhyStatus Response.

6.2 Notes for U-2023.06-2 Release

- USB_SVT Version: U-2023.06-2
- MPHY Version: U-2023.06-1
- SVT Version: U-2023.03
- Common Version: U-2023.06-1

Updates for this release are as follows:

- Created subgroup for usb4 and `usb_ss in svt_usb_link_ss_rx_sc` for SVDOC.
- Added cover groups `gent_downstream_ltssm_pattern_coverage` and `gent_upstream_ltssm_pattern_coverage` for LTSSM pattern coverage.
- Added support for Gen T Link command aggregation. Added a new cfg attribute `"enable_gent_link_command_aggregation"` to enable link command aggregation feature. Added a new test `"usb_20_na_31_ssp_gent_bulk_out_in_xfer_with_lcac_enabled"` in Host TB with Link command aggregation & Coalesing enabled.
- Updated the VIP to evaluate the presence of Far-end Receiver termination after the `cfg.receiver_detect_time` has elapsed. This is done to avoid reporting wrong termination value in scenarios where there had been an ATTACH/DETACH of the device during the `cfg.receiver_detect_time` delay.
- Added bins to cover different sizes of Data Packet Payload transmitted and received by the VIP.
- Added new define `SVT_USB_EXCLUDE_SSIC` to exclude mphy files from being called/compiled. user has to pass `SVT_USB_EXCLUDE_SSIC` if they do not want MPHY in USB
- Added UTMI glitch filtering logic for scenario when host is driving reset. Added user configurable macro `SVT_USB_20_USER_UTMI_LINESTATE_FILTER_DELAY` which is having default value as 10ps.
- Updated the mechanism of sampling CP10 on the serial interface upon CP9 to CP10 switching. This is done to take care of the scenarios where there is a discrepancy between encoding/decoding being turned off by the VIP and by the controller.
- Updated the VIP to count TSEQs for evaluating check `polling_rxeq_gen1_min_tseq_os_count_received_check` when Polling. LFPS state is skipped using cfg.ltssm_skip_polling_lfps.
- VIP updates made to issue error if the Link Partner sends LGO_U1 in Adapter mode of operation.
- Added support for scenario when port reset is driven by host during remote wake-up.
- Updated utmi check implementation `"chk_utmi_reset_timer"`.
- Removed tsigatt timer waiting condition for passive mode on utmi interface.
- Updated the VIP to reset the USB4 LFPS handshake state machine when changing the LTSSM state.
- Updating the VIP to correctly set the counter monitoring the number of identical TS2 received by the VIP in training states

- Updated the VIP to monitor for the LFPS_STOP packet from the link partner after `U2_EXIT`, `U3_WAKEUP` or PollingLfps Timeout. The check is performed on a failed U3_EXIT attempt from the link partner. For other cases, the check is enabled based on the cfg attributes. Also, updated the VIP to fire the check if LFPS_STOP Packet is received within `treset_min` after receiving Warm Reset.
- Added new services, `"USB_SS_GENT_ENABLE_LINK_KEEP_ALIVE"` and `"USB_SS_GENT_DISABLE_LINK_KEEP_ALIVE"` to enable or disable Link Keep-Alive feature dynamically. Added a new attribute `"gent_lkai_value_n"` in the `svt_usb_link_service` class to configure the value "n" in the LKAI link command while enabling Link Keep-Alive feature using link service `"USB_SS_GENT_ENABLE_LINK_KEEP_ALIVE"`.
- Updated the VIP to not drive power-down to P2 or P3 in the middle of an ongoing data block.
- Added a new service `"USB_SS_GENT_GENERATE_TX_OOBM_PACKET"` to generate an OOBM packet from the test bench. Added a new attribute to the `svt_usb_link_service` class `"gent_tx_oobm_type"` to specify the type of OOBM to be generated via link service `USB_SS_GENT_GENERATE_TX_OOBM_PACKET`.
- Added a new callback `"pre_usb_ss_gent_tx_coalesced_link_command_transform"` to intercept an outgoing coalesced link command in the `svt_usb_link_packed_transaction` format.
- Updated the Host VIP to recognize Loopback exit handshake response from the link partner in the scenario where Host transitioned from Loopback.active to Rx.Detect due to warm reset service. In this situation link partner would initially consider warm reset pulse as Loop back exit LFPS and hence send response LFPS.
- Added support for the new signal `"usb4_router_reset"` in the custom adapter interface.
- Added new physical services `CUSTOM_INTERFACE_ROUTER_RESET_ON` and `CUSTOM_INTERFACE_ROUTER_RESET_OFF` to drive `usb4_router_reset` as 1 and 0 respectively when the VIP is operating as custom adapter PHY.

6.3 Notes for U-2023.06-1 Release

- USB_SVT Version: U-2023.06-1
- MPHY Version: U-2023.06
- SVT Version: U-2023.03
- Common Version: U-2023.06

Updates for this release are as follows:

- Added a new configuration attribute `disabled_state_delay` to control the delay between entry to disabled state and checking for path enable.
- Added support for Gen T to Gen X switching upon Disconnect-Reconnect.
- Updated the code for VIP as a device. Used the LUP reception timeout instead of the attribute for LDN reception timeout.
- Updated the VIP implementation of PIPE signal TxDataValid to maintain the signal during the non-data states.

- Added a new configuration class attribute `gent_enable_random_lcmd_coalesing_without_aggregation` to enable random integration of the link commands in the absence of link command aggregation. Added a reference test `usb_20_na_31_ssp_gent_bulk_out_in_xfer_with_random_lcmd_coalescing_enabled` with `gent_enable_random_lcmd_coalesing_without_aggregation` enabled.
- Added the implementation of transmitting twenty Polling.LFPS bursts with non-varying tRepeat, after finding no SCD2 is detected. This condition is based on the Spec Section 7.5.4.4.2, Exit from Polling.LFPSPlus, of the USB 3.2 Spec Revision 1.0.
- Added a new check `pipe_TxElecIdle_toggled_with_TxDataValid_deasserted` to flag the scenario when mac toggles TxElecidle with TxDataValid as deasserted.
- Updated the VIP to execute services `TX_CP16_ON` and `TX_CP16_OFF` based on the values of shared status attribute `physical_usb_ss_tx_cp13_to_cp16` instead of `physical_usb_ss_tx_cp78`. Mapped the shared status attribute `physical_usb_ss_tx_cp13_to_cp16` in debug ports.
- Added a new check `gent_unrecognized_lcmd_rcvd` to flag the reception of an unidentified gen T link command. Added a new check `gent_lgo_rcvd_with_pending_header_packets_check` to flag the reception of `LGO_Ux` while the link partner had pending packets in its buffer. Added a new check `gent_low_power_entry_request_accepted_with_pending_header_packets_check` to flag the acceptance of low power entry request while the link partner had pending packets in its buffer.
- Added support for Nullified Data Packet (NDP) for Gen T.

6.4 Notes for U-2023.06 Release

- USB_SVT Version: U-2023.06
- MPHY Version: U-2023.06
- SVT Version: U-2023.03
- Common Version: U-2023.06

6.5 Notes for U-2023.03-3 Release

- USB_SVT Version: U-2023.03-3
- MPHY Version: U-2023.03
- SVT Version: U-2023.03
- Common Version: U-2023.03

Updates for this release are as follows:

- Converted the internal `svt_error` to validate `rx_buffer_credit_count_exceeded_check` to flag the scenarios where Rx buffer count exceeds the maximum value.
- Updated the VIP Link FC related to the U3 state specific bins in adapter mode of operation.
- The VIP has been updated to align with the compliance pattern of the compliance mode master after transmitting the PING LFPS signaling, without referencing the link partner with similar speed capability.

6.6 Notes for U-2023.03-2 Release

- USB_SVT Version: U-2023.03-2

- MPHY Version: U-2023.03
- SVT Version: U-2023.03
- Common Version: U-2023.03

Updates for this release are as follows:

- A new algorithm is added for host and device mode, both hub and non-hub mode for improved performance.
  This update includes the following:
  - Periodic TP
  - Async TP if number of Async TPs is more than the threshold (configurable attribute)
  - Periodic DP if available remote header credits are equal to a threshold (configurable attribute)
  - Async TP
  - DP (both Async and Periodic)
- Separated SSPx1 and SSPx2 Link FC bins and the HVP in adapter mode of operation
- Updating UTMI check `rxactive_deassertion_to_idle_check` for UTMI16 bit
- In adapter mode of operation, updated the VIP LINK FC code to ignore the bins related to the corrupted `LGO_U1` transmission from the VIP since U1 state is not supported in adapter mode
- Added check to trigger failure when rate changes during ongoing service like powerstate change or rate change etc., in powerstate P0 or P2.
- Added attributes to maintain the sent count of TS ordered sets in polling and recovery configuration to support the callback.
- V2RESET is added to reset the variables to validate the receive of every LFPS pattern thrice.
- Added `shared_status` to notify USB4 of the disconnect. Shared status name is `usb4_disconnect`.
- Added a new endpoint configuration `nump0_for_end_of_bulk_out_stream_xfer`. When set to 1 it sends the last DP with NumP=0 for Bulk Out Stream transfer. Default value is 0.
- Implemented the round robin selection for ISOC DPs.
- Updates to enable VIP to support the injection of `LENGTH_ERRORS` on Link Commands and Packets at the link layer while operating in Adapter Mode.
- USB3X Constraining the `FRAME_SKP_INSERT_ERROR`. During SSP mode the link command exception is not injected.
- Added else part in `ss_dev_non_isoc_in_post_packet_xmit`. Else is required when there is an exception to the packet and VIP generates an illegal response, that has to be handled by the VIP. In `ss_dev_non_isoc_out_post_packet_xmit` check the `get_tp_subpacket_type_val` and `tp_subpacket_type` are same which means there is no exception injected on the packet before deciding the correct `TP_ACK` and updating the variables.
- Added support for scenario where vbus off and hostdisconnect are updated at same time during data transfer.
- A new attribute `gent_enable_max_burst` is added in `svt_usb_configuration`. If this bit is set, then VIP updates the `ep_cfg.max_burst_size` at runtime according to the value of speed. The update is determined with respect to the value of speed post the Phase 3 of lane initialization in USB4. For more information, see the Table 9-20 of section 9.4.3.4 SuperSpeed Endpoint Companion Descriptor from the Universal Serial Bus 4 (USB4) Version 2.0 Specification. The

ep_cfg.max_burst_size is set to max value and VIP is reconfigured. For example, if the value of negotiated speed is 80GBPS and max_burst_size is set to less than 31, then ep_cfg.max_burst_size is set to max value (63).

Note This configuration is valid only for Bulk EP when #vip_generation = GEN_T. For GENX it is set to 0.

- Renamed the check response_timeout_check to thost_transaction_timeout_check.
- USB GENT_T support added:
  - Width of seq_num and nump in packets updated according to GEN_T
  - Support added to make sublink device notification as deprecated. In a SET_ADDRESS transfer over GENT, there is no sublink device notification sent from VIP. If the SET_ADDRESS transfer receives a packet, then VIP reports an error.
  - Width of packet sequence number increased to 7 bits and support for seq_num roll-over from 127 to 0 added.
  - Maximum 96DP support added in ISOC running in GENT. Set the endpoint_cfg.ssp_isoc_companion = 1 and endpoint_cfg.ssp_bytes_per_interval = 96*1024. Beyond this value, error is reported in VIP.
  - Max burst support of 63 in case of Bulk EP added. For remaining EPs burst values remain same as USB3.x. Set the burst using endpoint_cfg.max_burst_support.
  - Device descriptors are updated according to GEN_T Spec. You can change values inside the GEN_T descriptor through sequence svt_usb_device_framework_response_virtual_sequence_collection where format_descriptor_payload API is present. Old descriptors specified in USB3.x are present in same API.
  - ITP fields correction and bus_interval_adjustment_control are set as reserved and deprecated in GEN_T Spec. A checker is added if ITP fields set to non_zero values.
  - A new attribute gent_enable_max_burst is added in svt_usb_configuration. If this bit is set, then VIP updates the ep_cfg.max_burst_size at runtime according to the value of speed. The update is determined with respect to the value of speed post the Phase 3 of lane initialization in USB4. For more information, see the Table 9-20 of section 9.4.3.4 SuperSpeed Endpoint Companion Descriptor from the Universal Serial Bus 4 (USB4) Version 2.0 Specification. The ep_cfg.max_burst_size is set to max value and VIP is reconfigured. For example, if the value of negotiated speed is 80GBPS and max_burst_size is set to less than 31, then ep_cfg.max_burst_size is set to max value (63).

Note This configuration is valid only for Bulk EP when #vip_generation = GEN_T. For GENX it is set to 0.

  - Gen T specific checks added:
    - gent_ss_isoc_transfer_payload_size
    - gent_ssp_bytes_per_interval_greater_than_allowed
    - gent_sublink_dev_notification_check
    - gent_itp_rcvd_non_zero_correction

  - gent_itp_rcvd_non_zero_bus_interval_adjustment_control
  - gent_seq_num_of_received_ack
  - gent_nump_of_received_ack
  - gent_seq_num_of_received_extra_dp
  - gent_seq_num_of_received_dp
  - gent_received_unexpected_nump_val
  - gent_received_unexpected_ss_isoc_ack_seq_num
  - gent_isoc_unexpected_nump_val_range
- Configuration class attribute vip_generation updated to be set as svt_usb_configuration::GEN_T to enable Gen T mode.
- The link_packed_transaction_type_enum in class svt_usb_types updated with Out of Band Message (OOBM) types.
- Support to transmit and receive OOBM packets added.
- Updates to bypass advertisement and Link Management Packet (LMP) exchange added.
- Support for new link command word formats and link command coalescing added
- Framing symbols dropped from the link commands
- Length field replica and framing symbols dropped from the packets as per Gen T specifications
- Width of header sequence number increased to 7 bits and support for header sequence number roll-over from 127 to 0 added
- Added a new class svt_usb_link_ss_gent_ltssm with support for simplified LTSSM
- Added support to directly wake up from U0 state by configuring cfg.usb_ss_initial_ltssm_state as U0
- Support for U2/U3 entry-exit scenarios added
- Support for U2 PM entry timeout scenario added
- Support for Random reset in-between transfers added
- Support for Disconnect-Reconnect flow (without Gen T to Gen X switching)
- Support to drop Gen T link commands with SS_LINK_COMMAND_DROP service added
- Support to corrupt Gen T link commands with SS_LINK_COMMAND_TRANSFORM service added
- Support for transition from U2 to Error state due to link ready handshake failure added
- New attributes added to svt_usb_link_packed_transaction class: is_coalesced_link_command to identify coalesced link command. It should be set to 1 in the absence of USB4 adapter for the coalesced link commands being sent to the VIP
- Added a new configuration attribute gent_end_packed_trans_in_usb3_physical_layer to end an outgoing svt_usb_link_packed transaction object in USB 3.x physical layer in the absence of USB4 adapter
- Added the svt_usb_configuration class attributes:
  - t_oobm_polling_timeout : Timeout value for scheduling OOBM packet in GENT LTSSM
  - t_port_ready_timeout : Timeout value for Link Ready handshake in GENT LTSSM
  - portreset_delay : Value portreset_delay in GENT to indicate internal reset and advertise the same with PORT RESET and PORT RESET DONE OOBMs.

- `error_state_delay` : Value `error_state_delay` in GENT specifies the time duration between entry to Error state and transmission of OOBM by the device VIP
- `disconnect_state_delay` : Value `error_state_delay` in GENT specifies the time duration between entry to Disconnect state and checking for path enable.
- `gent_end_packed_trans_in_usb3_physical_layer` : Marks the outgoing `svt_usb_link_packed_transaction` object ended in `svt_usb_physical_adapter` class. This should be used only in the absence of USB4 adapter VIP in Gen T mode.
- `gent_enable_tx_rx_packed_trans_prints_in_usb3_physical_layer` : Enables the prints for incoming and outgoing `svt_usb_link_packed_transaction` objects in `svt_usb_physical_adapter` class.

- New checks added:
  - `gent_link_ready_handshake_failed_check`
  - `expected_gent_error_check`
  - `gent_unexpected_link_error_oobm_received_from_host_check`
  - `gent_unexpected_portreset_oobm_received_from_device_check`
  - `gent_unexpected_portresetdone_oobm_received_from_host_check`
  - `gent_unexpected_u3_resume_oobm_received_in_non_u3_state_check`
  - `gent_unexpected_u3_resume_oobm_received_from_host_check`

- New covergroups added for Gen T:
  - `gent_downstream_packet_type_coverage_tx`
  - `gent_downstream_packet_type_coverage_rx`
  - `gent_downstream_oobm_type_coverage_tx`
  - `gent_downstream_oobm_type_coverage_rx`
  - `gent_downstream_lgood_coverage_tx`
  - `gent_downstream_lgood_coverage_rx`
  - `gent_downstream_type1_lcrd_coverage_tx`
  - `gent_downstream_type1_lcrd_coverage_rx`
  - `gent_downstream_type2_lcrd_coverage_tx`
  - `gent_downstream_type2_lcrd_coverage_rx`
  - `gent_downstream_low_power_and_nop_lcmds_coverage_tx`
  - `gent_downstream_low_power_and_nop_lcmds_coverage_rx`
  - `gent_downstream_ltssm_fsm`
  - `gent_upstream_packet_type_coverage_tx`
  - `gent_upstream_packet_type_coverage_rx`
  - `gent_upstream_oobm_type_coverage_tx`
  - `gent_upstream_oobm_type_coverage_rx`
  - `gent_upstream_lgood_coverage_tx`
  - `gent_upstream_lgood_coverage_rx`
  - `gent_upstream_type1_lcrd_coverage_tx`
  - `gent_upstream_type1_lcrd_coverage_rx`
  - `gent_upstream_type2_lcrd_coverage_tx`
  - `gent_upstream_type2_lcrd_coverage_rx`
  - `gent_upstream_low_power_and_nop_lcmds_coverage_tx`
  - `gent_upstream_low_power_and_nop_lcmds_coverage_rx`

- gent_upstream_ltssm_fsm
- Converted internal svt_error to check rx_buffer_credit_count_exceeded_check to flag the scenarios where Rx buffer count exceeds the maximum value.
- USB GEN_T Limitations:
  - More Tests to be added in TestSuite with a separate Tab for tests applicable for Gen T
  - Link command aggregation not supported
  - Link-active feature not supported
  - Disconnect-connect with Gen T to Gen X switching not supported

More scenarios to be added for functional and checks coverage

6.7 Notes for U-2023.03-1 Release

- USB_SVT Version: U-2023.03-1
- MPHY Version: U-2023.03
- SVT Version: U-2023.03
- Common Version: U-2023.03

Updates for this release are as follows:

- Updated PCLKRate value correctly corresponding to rate and width when cfg.drive_initial_rate_0 is set to 1.
- Removed invalid link FC bins in adapter mode of operation, tx_sync_embedded_within_tseq_os, rx_sync_embedded_within_tseq_os, polling_portconfig_rx_rtconfig_set_sspx2, polling_portconfig_tx_rtconfig_set_sspx2.
- After completing the rate change data transmission is enabled.
- Updated VIP to monitor the TS1/TS2 OS received by the VIP and perform the following checks. The VIP check rx_os_link_functionality_bits_check triggers in case of violation.
  - Both the reset and loopback bits should be avoided for the link functionality.
  - In adapter mode of operation, the link functionality symbol should not have loopback bit set since loopback mode of operation is not supported.

6.8 Notes for U-2023.03 Release

- USB_SVT Version: U-2023.03
- MPHY Version: U-2023.03
- SVT Version: U-2023.03
- Common Version: U-2023.03

6.9 Notes for U-2022.12-3 Release

- USB_SVT Version: U-2022.12-2
- MPHY Version: U-2022.12
- SVT Version: T-2022.09
- Common Version: U-2022.12

Updates for this release are as follows:

- For USB4 Adapter mode of operations, link level coverage is updated.
- Added backward compatibility for using sequence and implementing your tests. If none of the attributes are passed from the test, then by default VIP initiates the Low Power entry. If behavior type is used as `HOST_DUT_NON_ERROR_INSERTION` or `DEV_DUT_NON_ERROR_INSERTION`, then it is set as `dut_driver_host` or `dut_driver_dev` , the entry in initiated from DUT.
- The value of `PM_ENTRY_TIMER` for USB4 has changed based on USB4.0 ECN. The value has changed from 16us to 36us.
- Checker check `rx_lcmd_unexpected_advertisement_lcmd_check` triggers only when VIP receives LRTY before sending and receiving the Header Sequence number advertisement. Previously, the check fired before the completion of Rx Type1 Type2 buffer advertisement. However, this can be ignored because the necessary condition for the link partner to send a packet is the completion of header sequence number advertisement and the availability of remote credits.
- In USB4 mode, added link level service commands to enable VIP to transmit `link_packed_transactions` when required by the Testbench.
- In USB4 Adapter mode of operation, a gap pattern is provided for the reception of LGOOD advertisement after VIP state transition to U0 since the idle symbols (used in native mode) are not exchanged.
- A negative gap pattern is provided between the reception of idles in U0 after transitioning from HotReset and the transmission of corrupted LGOOD.
- Adding the support in the VIP to send `RX_TERM_ON` and `RX_TERM_ON` link packed transactions on the link service request `SVT_USB_ADAPTER_PACKED_TRANS_COMMAND`.
- Updating physical layer for eUSB bus post the repeater configuration, irrespective of the time of issuing attach service request.
- In USB4 Adapter mode of operation, the `disable_scrambling` bit of the TS1/TS2 OS transmitted by the VIP requires to be set.
- Added a check `rx_os_link_functionality_bits_check` in the VIP Link layer to verify if the Link Partner asserts the disable scrambling bit in TS1/TS2 OS in Adapter mode of operation.
- Updated the VIP code to restrict issuing the error when it tries to initiate a low power exit requested by the higher layer immediately after responding to the remote wakeup request and moving to recovery. Since the low power exit is already in progress, the service request to initiate another low power exit will be dropped by the VIP.
- Updating UTMI check `sof_eop_invalid_check` to have condition for UTMI 16bit.
- Added checks to identify any invalid change in Signals Rate, PCLKRate, Width, RxStandby. Checkers added : `pipe_unexpected_width_change_signaling`, `pipe_unexpected_pclkrate_change_signaling`, `pipe_unsupported_pclk_change_request`, `pipe_unexpected_rxstandby_change_signaling`. Added configuration attribute `disable_pipe_pclkrate_and_width_checks` to disable checkers for invalid change in PLCKRate and Width. Default value is set 0, to disable the checkers need set configuration attribute to 1.

6.10 Notes for U-2022.12-2 Release

- USB_SVT Version: U-2022.12-2
- MPHY Version: U-2022.12

- SVT Version: T-2022.09
- Common Version: U-2022.12

Updates for this release are as follows:

- Added support of mapping eusb2 signaling strength change on agent's interface via debug events. Configuration variable to enable the feature is `enable_eusb2_signal_strength_debug` with default value 0.
- Added new CRC Error types to support value error injection in addition to mask based exception. The new exception is `CRC32_VALUE_ERROR`(Appilcable for USB 3.x), `CRC16_VALUE_ERROR` (Applcaible for USB 3.x and USB 2.0) and `CRC5_VALUE_ERROR` (Applicable for USB 2.0). If the injected value is same as the calculated CRC then the exception will not be injected and a message is displayed that the values are same. Updated the `get_description` method to display the corrupted and original CRC. The `get_crc*_val` methods take a new input which return the CRC without the exception injected also.
- In Hub emulation mode, support added for 18 type1 and 18 type2 credit release, support added for reordering of packets for Hub DFP. TP will be selected before Type1 and before Type2. Additionally, changes are made for the delay calculation for the transmitted packet from the device behind hub.
- Updated `utmi` check `rxactive_deassertion_to_idle_check` for FS and LS to wait for 1 FS/LS bit J after SE0 during EOP and Added compile time user configurable macro `SVT_USB_USER_HS_RXACTIVE_DEASSERTION_TO_IDLE_DELAY` and `SVT_USB_USER_FS_LS_RXACTIVE_DEASSERTION_TO_IDLE_DELAY` having default value as 9 and 3 respectively.
- Updated the feature of `suspendm` assertion during disconnect with valid configured delays on UTMI bus.
- Updated `transfer_ended` and `transaction_ended` function of callback `svt_usb_protocol_monitor_20_device_def_cov_data_callback` for abruptly aborted transfers.
- Added the directed service to move to U3 in the task `"auto_retry_u3_wakeup_after_tu3wakeupretry_delay_when_lfps_response_timeout"` as Handshake at the VIP Side can be successful, and VIP can transition to Recovery and then `SS_Inactive` due to which, when DUT tries again after `tU3WakeupRetryDelay` to initiate the U3 Wakeup, it will not be able to successfully do the handshake as no response `lfps` signal will be provided by the VIP. For this reason direct service is used to move the VIP to U3 again from `SS_INACTIVE/RECOVERY`. Expected error handled in test cases.
- Updated the VIP to flag the scenario where both reset and loop back bits are found to be set to 1 in the received ordered-sets.
- Updated the link level coverage code in Adapter mode of operation.

6.11 Notes for U-2022.12-1 Release

- USB_SVT Version: U-2022.12-1
- MPHY Version: U-2022.12-1
- SVT Version: U-2022.12-1
- Common Version: U-2022.12-1

Updates for this release are as follows:

- For including support for standard service descriptors in `svt_usb_types` class. The following are added:
  - Structure definitions corresponding to USB 3.1 Protocol Specification, section 9.6 Standard USB Descriptor Definitions.
  - Hierarchy encapsulation structures to allow descriptor structures to be grouped.
  - Static convenience functions for packing/unpacking descriptor data from bytes to corresponding structures, and for creating printable tabular representation of current structure contents.
  - Padding with 0's to IN Data Stage payload returned for Control Transfer if `setup_data_w_length` is greater than payload length created by `format_descriptor_payload()` function.
  - Updated `format_lang_id_payload()` function (used for creating payload for String Descriptors) to avoid setting bString data bytes to non-printable ASCII character codes, and instead to set descriptor bString data bytes to printable ASCII / UTF-8 character code sequence A, B, C, and so on.
- Added a Link Layer check `expected_lrty_check` to verify the link partner to send LRTY without VIP sending an LBAD.
- For Bulk OUT Stream `START_STREAM_END` is the state when the device responds to the reject stream by the host. Post the VIP OUT send 1 ERDY VIP exit IDLE state and go to `START_STREAM` so that device can respond to the reject stream of the Host. In the START STREAM state VIP sends 1 less ERDY as it has already sent 1 ERDY in IDLE state.
- Resetting the VIP status attribute `enable_usb_adapter_mode` if the VIP is operating in the native USB3X mode and not in the adapter mode. The issue is observed in the mode switching scenario where the VIP transitions from the adapter mode to the native mode of operation and the status attribute is not updated.
- Replaced the service from the case `LTSSM_U3_LFPS_TIMEOUT` to transition from U3 to `SS_DISABLED` with the U3 to `SS_INACTIVE`, as `SS_DISABLED` can cause the far-end high impedance. If tu3RxdetDelay is smaller than tNoLFPSResponseTimeout, then DUT can transition to Rx.Detect instead of U3.
- Triggering the notification when the VIP link receives packets which are used by the VIP Protocol layer in the implementation of the checks mentioned in table 8-13 of the USB 3.2 Specification. The notification is triggered from the link layer because the Protocol layer does not receive packets that are retried or received between the transmission of LBAD and reception of LRTY.
- Updates made to display the type of the training set tunneled packet received by the VIP when the check `rx_os_ltssm_state_check` is fired.
- Added attribute to indicate byte offset of symbol/byte in gen2 `DATA_BLOCK` in `svt_usb_data`. Added attribute to indicate start offset inside the `DATA_BLOCK` and attribute to align packet to specific offset inside the `DATA_BLOCK` in `svt_usb_packet`. The `svt_usb_link_ss_tx.sv` is updated to align the packet to specific offset.
- Updating Reset detection timing in Low Power state. At least 1 FS bit time duration of SE0 detection required to exit Low Power state.
- Added task `re_enumeration_post_normal_transfer()` in `post_normal_transfer` logic, for scenarios where `warm_reset` is before `post_normal_transfer`. Removed `enable_report_expected_dut_checks=0, enable_report_expected_vip_checks=0` from tests

as it can nullify task `set_*_checks_to_be_expected`. Removed `include_post_configure_phase_enumeration_sequence = 0` from the tests except where transitions directly happening from the polling state.

- Updated repeater to use standard USB2.0 serial bus GLS control variables while inserting GLS delays on DP/DM.
- Updated FS/LS EOP sent from eUSB controller to have controlled `gls_delay` when enabled.
- Added `test ts.usb_20_hs_fs_ls_30_na_eusb2_repeater_1p2_control_message_before_pull_up.sv` in device tb covering eusb 1p2 spec section 3.3.8.1 and a functional coverage bin `control_message_before_pull_up` for the same behavior.
- Updated checker `rxactive_deassertion_to_idle_check` to sample both linestate and rxactive before verifying the specification rule.

6.12 Notes for U-2022.12 Release

- USB_SVT Version: U-2022.12
- MPHY Version: U-2022.12
- SVT Version: U-2022.12
- Common Version: U-2022.12

Updates for this release is as follows:

To use INCDIR flow, the following SVT+NVS titles should be installed in common Designware home.

- svt_ethernet
- svt_i2c
- svt_mipi_i3c
- svt_uart
- svt_usb

6.13 Notes for T-2022.09-3 Release

- USB_SVT Version: T-2022.09-3
- MPHY Version: T-2022.09-3
- SVT Version: T-2022.09-3
- Common Version: T-2022.09-3

Updates for this release are as follows:

- Updated eUSB2 Host Repeater VIP for `config_idle` idle time interval between repeater configuration and next available signaling.
- Added a switch `SVT_USB_SS_DISABLE_SDS_PRIORITIZED_OVER_SKP_CHECK` for disabling check `sds_prioritized_over_skp_check`.
- Added a new function in `svt_usb_configuration` class `scale_ss_timer_values` to scale the ESS timers based on the `scaling_factor` argument.
- Updated the VIP timer used for delaying link commands for `fuse_length` of zero is interpreted as an immediate (0) instead of infinite (1) timeout request.

- Removed the dependency of define `SVT_USB_ESS_IPD_SCALEDOWN_FACTOR_IN_NS` for adding `inter_pkt_delay` while transmitting packets.
- Updated VIP to not send the EOB=1 on end of every transfer. Added configuration attribute to determine the behavior.
- Added new endpoint attribute which controls the NumP behavior on the OUT EP. NumP is randomized between 1 and MBS. The First ACK has NumP as MBS and subsequent ACKs have below half of MBS.
- Updated `suspendM` assertion during the disconnect feature to assert `suspendM` after `Termselect` moves to FS.

6.14 Notes for T-2022.09-2 Release

- USB_SVT Version: T-2022.09-2
- MPHY Version: T-2022.09-2
- SVT Version: T-2022.09-2
- Common Version: T-2022.09-2

Updates for this release are as follows:

- Added Individual Disconnect signal for Host and Device eUSB Repeater Mode. Individual disconnect signals would help seamless transition from one repeater mode to another at runtime.
- Added analysis port `transaction_ended_port` in protocol layer. These are added to avoid race which can occur with `NOTIFY_USB_TRANSACTION_ENDED`, if two transactions end at same time.
- Added a check to verify if the VIP receives an LFPS Tunnelled Packet with the Warm Reset bit set to 0b within the reset time after receiving an LFPS Tunnelled Packet with the Warm Reset bit set to 1b.
- Enabled link layer check (`twtdch_timing_check`) for all interfaces. Checker is expected to shout message where `ChirpK` from device is not responded with `ChirpKJ` from host within specification defined range.
- Added new variable `erdy_after_eob_eq_1` so the ERDY from VIP device after DP with EOB=1 can be controlled. The testbench may decide not to send the ERDY by setting this to 0.
- Updated `tse0_dr_hs/fs/ls` checker for Resume to message when an unexpected strong 0 followed by weak 0 is observed on the bus.
- Added/reused VIP checks, to verify if the Link Partner sends Tunnelled USB Packets in valid LTSSM state/substates. Added/reused the below checks:
  - `rx_scd_lbpm_ltssm_state_check` to detect if the link partner sends SCD or LBPM tunnelled packets in unexpected LTSSM state/substate. This is newly added.
  - `rx_os_ltssm_state_check` to detect if the link partner sends TS1/TS2/SDS OS tunnelled packets in unexpected LTSSM state/substate. This is newly added.
  - `rx_lcmd_hp_ltssm_state_check` to check if the link partner sends LCMD or Packets in unexpected LTSSM state/substate. This is reused from USB3X.
- Removing interface and RTL configuration dependencies from `svt_usb_20_na_30_ss_ltssm_state_transition_virtual_sequence` when attribute `ltssm_state_transaction` is set as `LTSSM_U0_UX_RECOVERY_LOOPBACK_RX_DETECT_POLLING_U0`

- Guarding the "dual_lane_unexpected_skew_check" and "dual_lane_ssp_start_block_check" during Loopback state since the specification says that the transmitter lane-to-lane skew does not need to be maintained in Loopback.

6.15 Notes for T-2022.09-1 Release

- USB_SVT Version: T-2022.09-1
- MPHY Version: T-2022.09-1
- SVT Version: T-2022.09-1
- Common Version: T-2022.09-1

Updates for this release are as follows:

- Added a new attribute choose_ack_over_dp. ACK is prioritized over a DP in case there are simultaneous requests. This feature is added since USB controller features are similar. Currently in the VIP it is random selection as the specification is not clear. This random feature leads to credit issue and performance drop.
- Updated the expected_recovery_check message to specify Type 1 or Type 2 credit_hp_timer timed out as well as the timeout during the link credit advertisement.
- Categorized USB2.0 link checkers in Host, Device, and Common checks category. Added agent configuration variable enable_link_chk_pass_cov and enable_link_chk_fail_cov for enabling passing and failing coverage respectively. Fail coverage model is operational and pass coverage model is under testing.
- Added support in HS Device Repeater to delay Strobe signal, after the Low Power entry is detected as success. Feature is enabled after configuration variable enable_delay_before_low_power_strobe is set to 1. Default value is 0. Delay value is configured with configuration variable delay_before_low_power_strobe with default value of 50ns.
- Updated to restart the HP_RESPONSE_TIMER with maximum fuse value after receiving a packet before receiving an acknowledgment for the transmitted Header. Previously, the HP_RESPONSE_TIMER was restarted with the maximum fuse value on receiving the Data Packet only.
- Updated the sequence for test usb_20_na_30_ss_ltssm_polling_lfps_to_compliance_mode to wait for shared_status.physical_usb_ess_working_speed to change to svt_usb_types::ESSG2 upon CP8 to CP9 transition.
- Updated the VIP adapter code to avoid the transmission of tunneled TS1 OS in the middle of tunneled LFPS_STOP packets in recovery substate after exiting Low Power States of U2 or U3. The tunneled TS1 OS was sent before all the three LFPS_STOP tunneled packets could be sent because of different threads are used to schedule the tunneled LFPS and the training ordered sets and tries to put into a single output channel.
- Updated the VIP(as MAC) to transmit 64 0s and 1s while transmitting CP13-CP16 patterns instead 32 0s and 1s.

6.16 Notes for T-2022.09 Release

- USB_SVT Version: T-2022.09
- MPHY Version: T-2022.09
- SVT Version: T-2022.09
- Common Version: T-2022.09

6.17 Notes for T-2022.06-3 Release

- USB_SVT Version: T-2022.06-3
- MPHY Version: T-2022.06-3
- SVT Version: T-2022.06
- Common Version: T-2022.06-3

6.17.1 Update

Update for this release are as follows:

- Added delay variable `suspendm_deassertion_delay_during_connect` to control suspendm de-assertion once connected after disconnect when `suspend_assertion_during_disconnect` bit set to one.
- Added protocol service to inject error in otg3 bit of Port Capability LMP.
- Added a new check `sds_prioritized_over_skp_check` to flag the scenario where DUT prioritizes SDS over SKP.
- Removed the define `SVT_USB_ESS_IPD_SCALEDOWN_FACTOR_IN_NS`. The delay for schedule link command is calculated on `link_command_transmit_delay` attribute passed using the callback.
- Updated the VIP to identify the received data and evaluate checks over it when BlockAlignControl is detected in the non-data state with powerdown as P0.
- Guarded the triggering of `sds_prioritized_over_skp_check` under the cfg `enable_skp_checks`. Updated the SVDoc for this check definition.
- Updated the VIP to set start block after every 16 symbols while sending pseudo random sequence during CP9 compliance pattern.
- Added support for 128b/132b encoding of CP9 when `cfg.enable_128b_132b_encoding_for_cp9` is enabled.
- Updated the implementation of `compliance_pattern_ordering_check` to disable it around CP0-CP1 transition boundary and re-enable five consecutive CP1s are received.
- Added a new cfg `move_non_config_lane_to_p0_in_polling_lfps_state` in order to move non-config lane to P0 in Polling.LFPS state.
- Updated the VIP to start incrementing the non-skp start blocks only after receiving the first SKP OS when the VIP moves from non-data state to a data state.
- Backward Comptability Change: Compile time macro (`+define+SVT_MPHY_NO_SERIAL_INTERNAL_CLK`) needed to switch off Mphy internal clock generation with latest releases.
- Required updated the check implementation of `pipe_dut_mac_invalid_tx_detect_rx_loopback_deassert_before_phy_response` to sample the Reset signal from the pins while evaluating the check.
- Added VIP support to send 128b/132b encoded CP9s in local loopback state when `cfg.enable_128b_132b_encoding_for_cp9` is enabled. Excluded the check `non_identical_os_received_across_lanes_check` for loopback state because as per spec, the loopback operation is performed on a per lane basis in dual-lane mode.
- Added a new check `sds_followed_by_data_block_check` to flag error if the VIP does not receive a data block after SDS, SKP being an exception.

- Updated CP9 detection algorithm for the scenario where `cfg.cp9_zeros` is set to maximum value.
- Adding macro `USB_NVS_PHY` for scenarios where nVS Phy is operational.
- Updating disconnect flag to get reset post repeater mode selection.

6.18 Notes for T-2022.06-2 Release

- USB_SVT Version: T-2022.06-2
- MPHY Version: T-2022.06-2
- SVT Version: T-2022.06
- Common Version: T-2022.06-2

6.18.1 Update

Update for this release are as follows:

- Updated USB2.0 Protocol layer exception `USB_20_W_LENGTH_PAYLOAD_MISMATCH` to effectively corrupt `setup_data_w_length` value in Setup Stage.
- Added `tx_pkt_preprocessing_delay` support for sublink speed device notification packets.
- Following tests removed from the verilog and vmm testbenches:
  `tb_usb_svt_vmm_basic_sys/tests/ts.basic_additional_utmi.sv`
  `tb_usb_svt_verilog_sys/tests/ts.pipe3_bulk_in_stream.v`
  `tb_usb_svt_verilog_sys/tests/ts.pipe3_tx_dp_error.v`
  `tb_usb_svt_verilog_sys/tests/ts.utmi.v`
- Added option to cut off VBUS immediately using cfg `immediate_vbus_cut_off` instead of service.
- Added option to move to CONNECT state in device repeater once vbus cut off from `CONNECT_ACKNOWLEDGE`.
- Adding timescale tolerance configuration variable for eUSB checkers to operate in sync where timescale rounded off value do not match with value configured.
- Added customized feature to cut off suspendm signal during disconnect assertion on host using cfg `suspend_assertion_during_disconnect`.
- Updated Inter Packet delay scenarios where a packet is ready to be transmitted on physical bus but is waiting for an ongoing reception to complete. In such scenarios a minimum inter packet delay of opposite direction is needed to meet specification allowed range
- Added a new `service_type svt_usb_link_service::SKP_COMMAND` and a new `svt_usb_link_service` class attribute `skp_length_ssp` to support variable length SKP insertion in SSP.
- Updating `tfiltse0` checker to operate only in Low Power Exit scenario.
- Added support of `tx_pre_processing_delay` in LMPs.
- Updated the messaging of `sync_ts_ratio_check` to report correct number of received and expected TS1/TS2 ordered sets in between SYNCs.
- Improved messaging to take care invalid powerdown value.
- Updated the VIP to identify and inject 10-bit disparity errors in received data only for SS speed. `TEN_BIT_ERROR` is not applicable at SSP speed. `BYTE_ERROR` is now supported for SSP and SS serial

- Added feature to initialize link and physical state based upon UTMI Reset signal assertion in run time. Added delay cfg variable `suspendm_assertion_delay_during_disconnect` to control the assertion of suspend signal during disconnect mode when `suspend_assertion_during_disconnect` cfg bit is set to 1. For host mode, suspendm signal assertion based upon HostDisconnect signal and for device mode based upon OpMode signal.
- Updated the minimum and maximum values of eUSB Control Message and other low power signaling timers for eUSB checker to work within threshold. Function `update_eusb_20_tolerance_values` added in configuration class to update timers.
- Mapped RX DC balancing signals to the debug ports.

6.19 Notes for T-2022.06-1 Release

- USB_SVT Version: T-2022.06-1
- MPHY Version: T-2022.06-1
- SVT Version: T-2022.06
- Common Version: T-2022.06-1

6.19.1 Update

Update for this release are as follows:

- Added support of eUSB version 1.2. Configuration variable `eusb_20_errata_version` for configuring the VIP to operate in eUSB version 1.2. Default value of `eusb_20_errata_version` is 1.1. After the value of `eusb_20_errata_version` is re-configured to 1.2 the VIP operates on the updated version. Updates specific to version 1.2 are as follows:
  - Minimum to maximum range of existing timers are updated as per version 1.2.
  - Renamed the timer `tpr_hs_reset_to_fs` to `tpr_reset_from_hs` for differentiating suspend and reset condition driven from host.
  - Added new timer tcmretry to decide delay between retry attempt of control message.
  - Added support of Glitch filtering in FSL/LS repeater mode. Cross-Over interval observed on Dp/Dm bus is reported on eDp/eDm bus.
  - Added support for optional `auto_resume` feature in Native and Repeater mode. Configuration variable named `enable_eusb_20_auto_resume` is 0 by default.
  - Added configuration variable `enable_eusb_20_fs_ls_continuous_sync_and_data_j` for FS and LS operation. In FS/LS while transmitting the SYNC pattern and differential data, the logic '0' for Data J may be driven continuously or may be maintained by rpd after being driven for `tse0_dr_lsfs`. When configuration `enable_eusb_20_fs_ls_continuous_sync_and_data_j` is disabled to 0, eUSBr drives Sync/Data J for only `tse0_dr_lsfs`. Default value of configuration variable is 1.
  - Added support in eUSPr to issue Port Reset if Control Message ACK is not received.
  - Added check in repeater to flag error if Control Message LX is detected second time in LX state.
  - Control Message RAP and its processing is updated.
    - Failed Control Message RAP will be retried for the count specified in macro `SVT_USB_PHYSICAL_RAP_CTRL_MESSAGE_RETRY_COUNT`, before aborting the control message. Default value of macro is 10.

- Added `eusb2_block_port_configuration` configuration variable to block Repeater/Port configuration for RAP command.
- Added check `eusb2_unexpected_cm_rap` to expect control message RAP only in Default state.

- Updated Passive monitor `tend_to_end` delay timer to include `tse0_dr_hs` time for HS packets.
- Updated Suspend sequence to wait for an additional delay corresponding to a Control Message with maximum width.
- Updated the size of static array `non_byte_data2short_str` in the `svt_usb_data` class.
- Added option in UTMI to get Txvalid de-assertion once Host disconnect asserted during chirping.

6.20 Notes for T-2022.03-3 Release

- USB_SVT Version: T-2022.03-3
- MPHY Version: T-2022.03-3
- SVT Version: T-2022.03
- Common Version: T-2022.03-3

6.20.1 Update

Update for this release are as follows:

- Added option to recover state machine from transmit state after the transaction is completed.
- Updated eUSB2 checker `tse0_dr_hs_fs_ls_missing_edp_checker` and `tse0_dr_hs_fs_ls_missing_edm_checker` to consider pull strength.
- Added capability in eusb2 repeater to switch over device mode to host mode in run time and vice versa with updating pulldown variables.
- Updated accelerated sequence collection for `ep_num` and `dev_addr` variable out of randomize for 20 transfer sequence.
- The VIP has been updated to drive the Ux exit response LFPS in U0 state due to link partner in low power state and sends exit LFPS without sending LPMA.
- Set Feature Test mode request is updated for Windex lower byte as 0 and higher byte as Test Mode selector.
- Feature added for Phy with delay between the PowerDown change and TxElecidle Assertion. Configuration `assert_txelecidle_before_powerdown_change` is added to enable this feature.
- Added the service to transform any symbol set to Gen1 TS1.
- Updated device link state machine to handle runtime disconnect in concurrent mode.
- Added service `USB_SS_DROP_RECEIVED_IDLES` to drop the received idle symbols by VIP.
- Added service `USB_SS_STOP_DROPPING_RECEIVED_IDLES` to continue receiving idle symbols by VIP.

6.21 Notes for T-2022.03-2 Release

- USB_SVT Version: T-2022.03-2
- MPHY Version: T-2022.03-2

- SVT Version: T-2022.03
- Common Version: T-2022.03-2

6.21.1 Update

Update for this release are as follows:

- Updated eUSB2 checker `rx_eop_missing_check` to speed specific EOP check support.
- USB 20 transfer sequence base class variable `ep_num` and `dev_addr` are udpated to be non-random. You can provide the intended `ep_num` and `dev_addr` from test case.
- Added configurable eUSB2 repeater state machine transition delay using the configuration variable `eusb2_rptr_state_machine_transition_delay`. Default value is 1ps.
- Added compile time macro `SVT_USB_XML_GEN_DISABLE` to disable XML generation.
- Updated runtime disconnect logic to drop an ongoing transfer if state enters into disconnect or power off.
- Updated device UTMI Opmode transition to non-driving on Power Down or Vbus Off.
- Updated UTMI host monitor agent to process HS Suspend Signaling on basis of termination settings.

6.22 Notes for T-2022.03-1 Release

- USB_SVT Version: T-2022.03-1
- MPHY Version: T-2022.03-1
- SVT Version: T-2022.03
- Common Version: T-2022.03-1

6.22.1 Update

Update for this release are as follows:

- Enhanced `H_L0_IDLE` and `D_L0_IDLE` state to process Resume and Remote Wake-up scenario respectively for scenarios where CM.LX is not issued by Controller while initiating L1/L2 state.
- Backward incompatible change: Verbosity level of the following USB2.0 VIP checkers have been elevated from `UVM_WARNING` to `UVM_ERROR` `resume_detection_timing_check, rx_packet_routed_check, expected_recovery_check, unmatched_symbols_check, twtrsm_timing_check, tdrsmup_timing_check, tdrsmdn_timing_check, tfiltse0_timing_check, twtrev_timing_check, t2susp_timing_check, twtrsths_timing_check, tuch_timing_check, tuchend_timing_check, twtfs_timing_check, twtdch_timing_check, inter_pkt_delay_same_source_timing_check, inter_pkt_delay_diff_source_timing_check, tfilt_timing_check, end_of_resume_reset_timing_check, max_inter_pkt_delay_same_source_timing_check, tdcnn_timing_check, token_data_size_limit_check, usb_20_bit_stuff_check, usb_20_eop_byte_boundary_check, resistor_register_settings_check, usb_20_hsic_bus_keeper_check, ulpi_txcmd_after_link_aborts_phy, ulpi_link_abort_phy_in_same_cycle, chk_utmi_txready_assertion_after_txvalid, rx_sync_min_length_check, rx_eop_max_length_check, phy_rx_buffer_overflow, unexpected_physical_service_request, data_packet_check.`

You are required to update the USB2.0 tests by changing the calling of function `demote_expected_uvm_warning` to `demote_expected_uvm_error` for any of the above mentioned checker.

The following messages are for `demote_expected_uvm_warning` as these are VIP internal state machine warning messages and not any protocol check:

- SOF processor was not granted access
- Moving to state `XCVR_STATE_IDLE` following the detection of nothing to transmit
- Could not insert exception

Note All USB2.0 Test Suite tests have been updated with the change of calling function, wherever applicable.

Compile time macro `+define+SVT_USB_BACKWARD_COMPATIBLE_CHECK_SEVERITY` can be defined, if the verbosity change update is not required.

- UTMI check to monitor Clock Unit intervals difference between assertion of TXVALID and TXREADY is controlled with a compile time macro `SVT_USB_20_TXVALID_TO_TXREADY_ASSERTION_UTMI_CLOCK_UI` with default value of 2.
- Link to overcome the enabled transmit state when the PHY is not updated with the linestate value during the transfer while disconnection is in process on eusb2 line in native mode.
- The `tddis_timing_check` check is updated to be a note message by default from VIP.
- Updated the Control Message decoding logic to assert respective control message flag after half duty cycle of `tcm_clk_min` duration has expired after the last STROBE pulse detected on eDp pin.
- Added the configuration variable `block_suspendm_deassertion`, `block_l1suspendm_deassertion` and `block_sleepm_deassertion` to block SuspendM, L1SuspendM, SleepM signal toggling in Low Power scenarios. Default value of configuration variable is 0.
- The eUSB2 timers values have been updated in multiples of FS or LS UI, wherever applicable. Earlier the calculated values where directly used.
  - Passive monitor entry to `reset_or_restart_s_state` from ENABLED state is after the expiration of twtrev timer. After the entry of `reset_or_restart_s_state`, passive monitor decode resetting by continuous SE0 on bus and if chirpK does not reach the HS within tdrst time, then full speed is resumed.
    So twtrev timer in host is configured for host cfg to match the DUT full speed reversion time.
  - The eUSB2 timers minimum and maximum range are updated as per the upcoming version by default. To keep the timers running without this change, the following listed timer configuration variable requires the following tolerance:

```
real lsfs_tolerance = 400ps;
real hs_tolerance = 20ps;
real fs_nominal_bit_time = 83328ps;
<agent_configuration>.tse0_dr_lsfs_min = <agent_configuration>.tse0_dr_lsfs_min -
lsfs_tolerance;
<agent_configuration>.tse0_dr_lsfs_max = <agent_configuration>.tse0_dr_lsfs_max +
lsfs_tolerance;
```

```
<agent_configuration>.tse0_dr_hs_min = <agent_configuration>.tse0_dr_hs_min -
hs_tolerance;
<agent_configuration>.tse0_dr_hs_max = <agent_configuration>.tse0_dr_hs_max +
hs_tolerance;
<agent_configuration>.tcm_se1_1x_min = <agent_configuration>.tcm_se1_1x_min -
lsfs_tolerance;
<agent_configuration>.tcm_se1_1x_max = <agent_configuration>.tcm_se1_1x_max +
lsfs_tolerance;
<agent_configuration>.tcm_se1_8x_min = <agent_configuration>.tcm_se1_8x_min -
lsfs_tolerance;
<agent_configuration>.tcm_se1_8x_max = <agent_configuration>.tcm_se1_8x_max +
lsfs_tolerance;
<agent_configuration>.tdr_k_se1_1x_min = <agent_configuration>.tdr_k_se1_1x_min -
lsfs_tolerance;
<agent_configuration>.tdr_k_se1_1x_max = <agent_configuration>.tdr_k_se1_1x_max +
lsfs_tolerance;
<agent_configuration>.tdr_k_se1_8x_min = <agent_configuration>.tdr_k_se1_8x_min -
lsfs_tolerance;
<agent_configuration>.tdr_k_se1_8x_max = <agent_configuration>.tdr_k_se1_8x_max +
lsfs_tolerance;
<agent_configuration>.tcm_clk_max = 2*fs_nominal_bit_time;
```

For example,

```
host_cfg.tse0_dr_lsfs_min = host_cfg.tse0_dr_lsfs_min - lsfs_tolerance;
dev_cfg.tse0_dr_lsfs_min = dev_cfg.tse0_dr_lsfs_min - lsfs_tolerance;
```

6.23 Notes for T-2022.03 Release

- USB_SVT Version: T-2022.03
- MPHY Version: T-2022.03
- SVT Version: T-2022.03
- Common Version: T-2022.03

6.23.1 Update

Update for this release is as follows:

VIP is compatible with IEEE UVM 1800.2-2020-1.0/1.1 versions.

6.24 Notes for S-2021.12-3 Release

- USB_SVT Version: S-2021.12-3
- MPHY Version: S-2021.12-3
- SVT Version: S-2021.09
- Common Version: S-2021.12-3

6.24.1 Updates

Updates for this release are as follows:

- Added check to receive LFPS STOP after three WARM RESET LFPS packets are received. Check name `lfps_usb4_no_lfps_stop_after_warm_reset` is initiated, if less than three LFPS STOP packets are received for cases where three are not required (U3 Wakeup). The cfg attributes are added to determine the LFPS STOP after U3 Wakeup or determine whether one or three LFPS STOP Packets are to be detected on U3 Wakeup.
- Updated SOF service interval processing to priortize L1 entry as compared to issuing SOF token.
- Updated custom interface such as powerdown should not change in p3 if wakeup signal asserted after u3 wakeup is initiated.
- USB2 coverage groups are updated to remove invalid cross coverage groups for specific USB2 speeds.

6.25 Notes for S-2021.12-2 Release

- USB_SVT Version: S-2021.12-2
- MPHY Version: S-2021.12-2
- SVT Version: S-2021.09
- Common Version: S-2021.12-2

6.25.1 Updates

Updates for this release are as follows:

- Updated Host Repeater Disconnect in Low power mode scenarios where device disconnected after Host accepted the valid Remote wakeup duration.
- During LS mode disconnection in runtime, updated Opmode value 01 to 00 sync with Xcvrselect 01 to 10 value.
- Adding host configuration variable `enable_eusb2_repeater_end_of_resume_duration_check` (default value 0) to keep `eusb2_repeater_end_of_resume_duration_check` checker enabled or disabled.
- Added `prioritize_ss_isoc_in_transfer` in `svt_usb_host_configuration` to prioritize ISOCHRONOUS IN transfer over all other transfers in SS. For example, if a BULK IN and ISOC IN are enabled, then ISOC bursts are prioritized and sent on bus. After ISOC IN is successfully completed (EOB/LPF 1 received), BULK transfer packets are sent. This is only valid for SS as concurrent IN bursts are possible in SSP.
- Updating HS `H_LX` state of Host eUSB2 Repeater to accept Suspend state J only if eSE1 is not received from Host Controller.
- Updating End of Reset duration for a HS Device Connected to FS Host to be TSTROBE instead of FS UI.
- Updated adjust_remote_buffer_count checker for header packet note message with `ALLOW_EXCEED_MAX_HEADER_BUFFER_ERROR` as this error is expected.
- Updated physical reception logic to process partially received packet like partial SOF after reset completion.
- Added option to recover even Linestate signal. SYNC and EOP by UTMI PHY is not updated for valid data with Txvalid during EXTSE1 in eusb2 bus.

- Added checks to identify unique OS packet (`os_usb4_invalid_os_type_check`) and Link functionality byte if SDS packet to be 0 (`os_usb4_link_functionality_non_zero_for_sds_check`).

6.26 Notes for S-2021.12-1 Release

- USB_SVT Version: S-2021.12-1
- MPHY Version: S-2021.12-1
- SVT Version: S-2021.09
- Common Version: S-2021.12-1

6.26.1 Updates

Updates for this release are as follows:

- Added warning message for decoding Z in any of dp or dm line in serial interface.
- Updated Host Repeater HS End of resume signaling duration to drive 2 LS UI of J state. The `eusb2_repeater_end_of_resume_duration_check` checker is added in host passive agent to monitor the duration of HS end and resume signaling in host repeater mode.
- To avoid the reset BI Count in an ITP even if LTSSM state moves to the Low Power or Non Data State (SS Inactive, Rx Detect). If the `cfg.reset_bi_count_with_ltssm_state` is set, then BI Count resets in Non Data State.
- To reset BI Count of an ITP to 0 in between of a simulation. Bus Interval Boundary does not reset. Service `USB_SS_ITP_RESET_BI_COUNT` is provided from TB. Valid only for Host VIP.
- Updated Repeater's Control Message ACK sending logic for accepting Valid Control Message and generate ACK if Serial bus strength goes to Weak SE0.

6.27 Notes for S-2021.12 Release

- USB_SVT Version: S-2021.12
- MPHY Version: S-2021.12
- SVT Version: S-2021.09
- Common Version: S-2021.12

6.28 Notes for S-2021.09-3 Release

- USB_SVT Version: S-2021.09
- MPHY Version: S-2021.09-3
- SVT Version: S-2021.09
- Common Version: S-2021.09-3

6.28.1 Updates

Updates for this release are as follows:

- Added support of randomly disconnecting device in Low Power mode exit scenarios.

- Added a `svt_usb_data` class object `fs_ls_utmi_linestate_eop_end_time` to mark timestamp of FS/LS EOP SE0 to Non-SE0 transition on linestate signal for UTMI/ULPI interface. Added object value is used to calculate precise inter packet delay for UTMI/ULPI interface.

6.29 Notes for S-2021.09-2 Release

- USB_SVT Version: S-2021.09
- MPHY Version: S-2021.09-2
- SVT Version: S-2021.09
- Common Version: S-2021.09-2

6.29.1 Updates

Updates for this release are as follows:

- Adding a configuration variable `enable_synchronous_host_utmi_resume` which is by default kept to "0". Set to "1" if Host is required to wait for UTMI Clock before updating OpMode to 2'b10 and start resume.
- Updated the VIP to set start block as 0 and block header type as CONTROL_BLOCK for the extra symbols in case of `LONG_SYMBOL_SET_ERROR` error insertion scenarios.
- The `use_of_deprecated_feature` checker has been removed and is replaced by `svt_error`. Added macro(`SVT_USB_REDUNDANT_CHECKS`) to retain removed checks declaration for backward compatibility.
- Elevated the severity of the following checks from warning to error :
  - `start_framing_check`
  - `dph_pending_check`
  - `dph_dpp_symbols_check`
  - `dpp_max_symbols_check`
  - `unexpected_symbol_received_in_a_valid_framing_set_check`
  - `unmatched_symbols_check`
  - `unexpected_skp_rcvd_check`
  - `vbus_starvation`
  - `ignored_physical_service_request`
  - `unexpected_mac_signaling`
  - `pipe3_mac_reset_check`
  - `data_converter_8b10b_decode_in_low_power`
  - `data_converter_8b10b_disparity`
  - `pipe_pclk_check`
  - `pipe_mac_reset_check`
  - `pipe_unexpected_receiver_detection_signaling`
  - `phy_rx_block_header_with_error`
  - `expected_recovery_check`
  - `ltssm_state_change_request_check`
  - `bcnt_error_count_failed_check`

- detected_unexpected_symbol_in_recovery_check
- polling_active_identical_ts1_received_check
- polling_active_identical_ts2_received_check
- polling_configuration_identical_ts2_received_check
- recovery_active_identical_ts1_received_check
- recovery_active_identical_ts2_received_check
- recovery_configuration_identical_ts2_received_check
- sds_os_not_received_check

A Backward compatible define SVT_USB_BACKWARD_COMPATIBLE_CHECK_SEVERITY Is added to revert the severity of the above checks back to WARNING. This define will be removed in future releases.

- Added option in repeater to transmit test mode pattern even Control Message (CM). In device mode the test fails to arrive from UTMI PHY on eusb2 lines.

6.30 Notes for S-2021.09-1 Release

- USB_SVT Version: S-2021.09
- MPHY Version: S-2021.09-1
- SVT Version: S-2021.09
- Common Version: S-2021.09-1

6.30.1 Updates

Updates for this release are as follows:

- Any unknown x value comes in edp/edm ignored by repeater during run time.
- Disconnect during l2 entry and disconnect before l2 entry test and sequence updated for both single and dual mode matching the scenario sections.
- During suspend, any resume K change without the presence of clock, is updated after one UTMI clock to avoid any glitch.
- Updated invalid address field exception to randomize exception value between 0 to 127.
- Updated the scenario where VIP fails to initiate low power entry if the previous low power entry request is aborted due to reset.
- Updated eUSB Repeater FS/LS EOP logic for a valid 0.66UI SE0 interval received on USB2.0 serial bus before accepting the signaling as a valid EOP. The FS/LS EOP receiving logic of eDSPr/eUSPr is updated to ignore any value for the first 0.66UI of EOP interval. Feature can be disabled by resetting the configuration variable named enable_eusb_20_repeater_fs_ls_eop_asymmetry to 0. By default feature is enabled.
- Added support in eusb2 repeater to ignore X value driven in edp/edm on initial announcement session
- Added SVT_USB_20_UTMI_LINESTATE_FILTER_DELAY macro to control any SE1/SE0 coming at time of resume during suspend when clock absent. Default value is 10ps.

- Removed `use_of_deprecated_feature` checker. Converted `use_of_deprecated_feature` checker to `svt_error`.
- Added a new attribute `transmit_cp9_in_local_loopback_ssp` to `svt_usb_link_service` class to enable the VIP to transmit CP9 ordered sets in Local loopback active state in SSP mode.
- Added Replaceable Macro named `` `SVT_USB_BCD_SS_LOW_BYTE(8'h00), `SVT_USB_BCD_SS_HIGH_BYTE(8'h03), `SVT_USB_BCD_HS_LOW_BYTE(8'h10), `SVT_USB_BCD_HS_HIGH_BYTE(8'h02), `SVT_USB_BCD_FS_LS_LOW_BYTE (8'h10), `SVT_USB_BCD_FS_LS_HIGH_BYTE(8'h02) `` defining USB Specification Release Number in Binary-Coded Decimal value. Accordingly updated framework sequence checker named `device_descriptor_ss_bcdUSBlowbyte_check`, `device_descriptor_ss_bcdUSBhibyte_check`, `device_descriptor_hs_bcdUSBlowbyte_check`, `device_descriptor_hs_bcdUSBhibyte_check`, `device_descriptor_fs_bcdUSBlowbyte_check`, `device_descriptor_fs_bcdUSBhibyte_check`, `device_descriptor_ls_bcdUSBlowbyte_check`, `device_descriptor_ls_bcdUSBhibyte_check` to check the speed specific macros as the maximum acceptable valid value of BCD.
- Added support to avoid regular set/get interface with hub while Hub enumeration is on.
- Disabled speed-lane compatible check in negotiation test.
- Added support for BLR compliance mode state.
- Added a new link service `USB_SS_SET_TRANSMIT_BLR_COMPLIANCE` to initiate BLR compliance mode transition from the VIP, applicable only for SS.
- Added a new cfg `blr_compliance_mode_four_skp_os_delay` used to configure the time delay after which BLR compliance mode master schedules four consecutive SKPs.

6.31 Notes for S-2021.06-3 Release

- USB_SVT Version: S-2021.06-3
- MPHY Version: S-2021.06-3
- SVT Version: S-2021.06
- Common Version: S-2021.06-3

6.31.1 Updates

Updates for this release are as follows:

- Update related to scenario, where the ISOC transfer is aborted and then the service interval is lapsed. The device might send the data packets requested by the previous ACK, so save the `num_outstanding_requested_dps` to `preexisting_num_outstanding_requested_dps` and use that to tolerate the DPs. Its is mandatory to set the `ess_isoc_warn_for_orphan_dps` to 1 for the check to be a warning.
- In LPM mode, modified utmi suspend and sleep signal to be asserted after moving to Full speed and linestate J appeared.
- Updated Host SOF generation logic to consider `RESTART_S`,`TRANSMIT_R` and `SEND_EOR` internal states and not generate SOF in these states.

- Added new configuration variable in host named `prioritize_isoc_transfer`. If enabled, then ISOC transfer is prioritized over other transfers.
- Added a fatal message in the VIP which informs about the VIP not being attached when the LFPS is received or sent.
- FS/LS Native eUSB Mode updated to process missing digital Ping scenario as a start of Disconnect from Host. For example, extended SE1 for Host.
- Optimized VIP html class reference document by disabling generation of check coverage class html files.
- `UVM_INFO`/ERROR/WARNING messages of 20 transfer and physical layer sequence file updated to display correct string in response to string format specifier.
- Strengthened `tse0_dr_hs_fs_ls` checker on L1/L2 suspend/resume conditions to consider gate level delays in between edp/edm signals.

6.32 Notes for S-2021.06-2 Release

- USB_SVT Version: S-2021.06-2
- MPHY Version: S-2021.06-2
- SVT Version: S-2021.06
- Common Version: S-2021.06-2

6.32.1 Updates

Updates for this release are as follows:

- Updated the scenario where Host rejects U3 entry service when U1/U2 entry initiated by Host is in progress and Host receives LXU in response of `LGO_U1/LGO_U2`.
- Low Power Exit service is blocked and not send from VIP when ITP is in progress.
- Resetting Bus interval count for ITP when LTSSM State is U3.
- Added an `is_valid` check for `PIPE3_IF` configured for SSP speed or dual-lane as `PIPE3_IF` is supported only for Gen 1X1
- Added the following two attributes in the `shared_status` : `rx_lfps_type` which reports the type of LFPS received by the VIP and `lfps_received` which is an event that triggers whenever an LFPS is detected by the VIP.
- Added a new configuration `enable_ltssm_coverage_for_active_agent` to enable LTSSM coverage for the active VIP agent. Its default value is set to zero.
- Updated the scenario where device VIP moves from U1 to U2 due to inactivity timeout despite `cfg.u2_inactivity_upstream_enabled` set to 0. Device should move to U2 state due to timeout only when this cfg is set to 1.
- The following scenario has been updated, where the Host retries U3 entry post re-entering U0 post PM LC timer timeout despite traversing through Rx.detect state in between Recovery and U0.
- Added a new attribute to the `svt_usb_symbol_set` class to identify whether the symbol set is generated using the link service or the regular VIP flow.
- Updating Host VIP working with TypeC while swapping modes. For example, `usb_20_dr_swap_supported` mode to use Vbus toggling as a trigger mechanism to move out of Disconnected state instead of sampling the value.

- Added update in the VIP to report detection of CP9 pattern in CTS compliant loopback state.
- Reverted sequence `svt_usb_20_na_31_pipe4_serial_ssp_link_errors_block_header_error_system_virtual_sequence` to issue control transfer instead of Bulk for VIP-IIP runs.

For endpoints that support stream it can happen that the when the first DP which is either a Prime or a reject stream its rejected by a STALL so the `xfer.next_seq_num_to_be_acked` is 0 and the `next_dp_seq_num gets reset. xfer.next_seq_num_to_be_acked` never gets updated since the first DP of the actual transfer never gets sent. So when the `ustream_ep_mgr_state` is `PRIME_PIPE` or `START_STREAM_END` or DISABLED dont reset the `next_dp_seq_num`.

6.33 Notes for S-2021.06-1 Release

- USB_SVT Version: S-2021.06-1
- MPHY Version: S-2021.06-1
- SVT Version: S-2021.06
- Common Version: S-2021.06-1

6.33.1 Updates

Updates for this release are as follows:

- Updated `cfg.reset_bi_count_u3` to 1 by default. The ITP BI count is reset to 0 whenever link moves to U3.
- Added twtfs timing check on link.
- Added support of vbus off, eop error, and alignment error test to run with EUSB2 UTMI PHY configuration.
- Removing dependency of SVT compile time macro from USB SVT package.
  Backward compatibility change: Setup compiles nVS and SVT USB together for SVT macro definition in their test bench as per the VIP requirement which is being used in run time.
- Updated the scenario, Host rejects U3 entry service when U1/U2 entry initiated by Host is in progress.
- Updated USB 20 protocol block to work in scenario where configuration swapping causes component type to change from Host to Device and vice versa.
- Updated the scenario where VIP moves from Hot.reset.exit to U0 upon receiving the SDS which was transmitted by link partener while moving from polling/recovery idle to Hot.reset.active state in cio mode. The VIP should have considered the SDS that is received after the TS2 reset bit was asserted and then de-asserted.
- Added a new configuration `enable_fast_credit_release` to enable VIP to release credits faster upon receiving a packet.
- Updated PHY not to enter preamble mode, during PID check error where Preamble PID is received instead of ACK.

6.34 Notes for S-2021.06 Release

- USB_SVT Version: S-2021.06
- MPHY Version: S-2021.06

- SVT Version: S-2021.06
- Common Version: S-2021.06

6.34.1 Updates

Updates for this release are as follows:

- Updated the scenario where Unmatched DPP END framing is received.
- Added link timing checks tuch, tdrsmdn, tdrsmup and tfiltse0 for all interfaces.
- Added support for Euclide (Eclipse based IDE) for lint rule checking. The VIP works seamlessly with Euclide IDE when configured with testbench rule setting and would not result in any fatal errors.
- UVM 1.2 is supported without `UVM_NO_DEPRECATED` macro.

6.35 Notes for R-2021.03-3 Release

- USB_SVT Version: R-2021.03-3
- MPHY Version: R-2021.03-3
- SVT Version: R-2020.12
- Common Version: R-2021.03-3

6.35.1 Updates

Updates for this release are as follows:

- Updated the scenario for the Host upon successful entry to U3 state to work as expected after resetting the counter `start_u3_entry_count`.
- Added option to drive UTMI pin Reset signal high in runtime when reset service issued with cfg based control naming `drive_utmi_reset_on_runtime`. By default, the value of `drive_utmi_reset_on_runtime` is 0 to assert UTMI Reset pin during the initialization to initialize all UTMI signals.
- Restructured `svt_usb_20_hs_fs_ls_30_na_handshake_transfer_system_virtual_sequence` to work with third party simulator with varied versions.
- Updated the scenario where unmatched DPP END framing received and not to run into NOA.
- Updated `svt_usb_protocol_ss_itp_processor.sv` file to work correctly with third party simulator on protected package.
- Passive monitor has been supported to control the disconnect during transmit state and recover.
- Updated the internal delta calculation in case of calculating ITP time difference.

6.36 Notes for R-2021.03-2 Release

- USB_SVT Version: R-2021.03-2
- MPHY Version: R-2021.03-2
- SVT Version: R-2020.12
- Common Version: R-2021.03-2

6.36.1 Updates

Updates for this release are as follows:

- Added a new link service `USB_SS_SET_TRANSMIT_LOCAL_LOOPBACK` to set local loopback bit to 1 in the outgoing TS2s.
- Added support for CTS Compliant Loopback LTSSM state. A new cfg is added to enable this feature : `loopback_cts_compliant`.
- Added Support in VIP to check shorter linestate SE0 duration instead of 1 FS/LS clock cycle when mapping done earlier by eUSB2 repeater after EOR completion in the case of disconnect during reset on host repeater mode.
- Error scenario has been implemented to enter recovery when Block header on lane0 and Block header on lane1 are both valid but not identical.
- Added support to send SKP symbols in between BDAT. A new cfg `enable_skp_between_bdat_symbols` is added to enable this feature.
- Grouped SVDOC USB2.0 checks coverage based on interfaces as categorized after `usb_20_signal_interface` setting.
- Update to support scenario where upon SCD timeout in Polling. LFPSPLUS state link issues speed change request before issuing `TX_LFPS_OFF` service in case of active LFPS being on the line.
- Enhanced ITP processor algorithm for ITP timing checks. To maintain backward compatibility define `SVT_USB_PL_OLD` added. By default new algorithm is working. With new algorithm in place `cfg.ignore_dl_df_itp` will also be deprecated. If an ITP is received with delayed bit set,then Note is shown rather than warning. An ITP is received with `dl_bit` set if link moves to recovery in between, due to which packet is delayed. For every deferred ITP, there is an error and for all delayed ITPs there is a warning.
- Checker `deferred_itp_rcvd` added reports error if ITP is received with `df_bit` set.
- Checker `delayed_itp_rcvd` added reports warning if ITP is received with `dl_bit` set.
- Checker `received_unexpected_delta` added which reports error if ITP with delta greater than 7500 is received.
- Turning off the default value of `shared_status.random_df_on` so that ITP `df_bit` is not randomized when sent from host.
- Update VIP to support scenario where VIP is unable to identify inverted SYNC when the VIP enters recovery from a low power state.
- Added support in VIP to check shorter linestate (< 1 clock cycle SE0) in between announcement and connect session drive by repeater.

6.37 Notes for R-2021.03-1 Release

- USB_SVT Version: R-2021.03-1
- MPHY Version: R-2021.03-1
- SVT Version: R-2020.12
- Common Version: R-2021.03-1

6.37.1 Updates

Updates for this release are as follows:

- VIP has been updated to unblock a deadlock caused in system reset sequence due to LTSSM state not getting updated to Rx.detect
- VIP updated to allow low power services after completion of U1/U2 entry-exit upon U1/U2 inactivity.
- ALL ITP checks are added with sub group as `SS_ITP`.
- Added a new check `unexpected_data_on_non_config_lane` to report unexpected data reception on the non-config lane when the VIP and the link partner have settled in the single-lane.

6.38 Notes for R-2021.03 Release

Updates for this release are as follows:

- USB_SVT Version: R-2021.03
- MPHY Version: R-2021.03
- Common Version: R-2021.03
- SVT Version: R-2020.12

6.38.1 Updates

Added the following:

- Support for bypassing initial speed settlement with config variable named `bypass_initial_settlement` for usb2.0 serial and eusb2.0 interface.
- Added `DRIVE_EXTSE1` physical service control to drive port reset from host with physical service in runtime.
- Added a new check `hot_reset_active_min_ts2_os_count_received_check` to make sure that the DUT meets the handshake conditions for the Hot.reset state.
- Added a new check `non_identical_os_received_across_lanes_check` applicable for USB 3.2 mode. The check makes sure that identical ordered-sets are received on both the lanes during training states.
- Added option for bypassing port reset and port config/announcement in both repeater and host/device VIP with variable `bypass_eusb2_port_reset_port_config`.
- Adding support for putting on hold a frequent retried Non-ISOC transfer on a specific endpoint. Number of attempts before holding an endpoint is configured from the endpoint configuration variable named `max_retry_due_to_nak_before_moving_to_next_ep`, `max_retry_due_to_timeout_before_moving_to_next_ep`, `max_retry_due_to_error_before_moving_to_next_ep`. All having default value of -1 (disabled). Example TB test `ts.basic_additional_20_utmi_host_phy_device_mac_frequent_retry_non_isoc_transfers.sv` shows usage of this feature.
- Added support for Euclide (Eclipse based IDE) for lint rule checking. The VIP works seamlessly with Euclide IDE when configured with testbench rule setting and would not result in any fatal errors.
  - UVM 1.2 is supported without `UVM_NO_DEPRECATED` macro.

- For resolving SVT related errors, contact Synopsys support for SVT T release.

6.39 Notes for R-2020.12-3 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.12-3
- MPHY Version: R-2020.12-3
- Common Version: R-2020.12-3
- SVT Version: R-2020.12

6.39.1 Updates

Added the following:

- Support added to handle BW between exit lfps. Where Bandwidth is initiated after exit LFPS.
- Added a new link service `USB_SS_GENERATE_TX_PACKET` that generates a packet from the Testbench to be transmitted by the link.
- Added a new callback `pre_usb_ss_tb_packet_out_chan_put` to bypass a received packet from being sent to the Protocol Layer
- Adding support of Setup and Hold time check in USB2.0 UTMI/ULPI Interface signal. Configuration variable named `enable_utmi_setup_time_check` `enable_utmi_hold_time_check` or `enable_ulpi_setup_time_check` `enable_ulpi_hold_time_check` if set to 1, then input UTMI or ULPI signals need to be constraint with setup and hold time is set by the following macro `SVT_USB_SETUP_TIME_UTMI_CHECK_TIME_FS` or ULPI and `SVT_USB_HOLD_TIME_UTMI_CHECK_TIME_FS` or ULPI.
- Adding support of X and Z values checker on UTMI/ULPI Interface signal. Configuration variable named `enable_utmi_check_For_X_Z` or `enable_ulpi_check_For_X_Z` used to enable the checker.
- To disable checker for any specific UTMI/ULPI pin configuration variable named `disable_utmi_<signal>_x_check` or `disable_ulpi_<signal>_x_check` need to be set to 1.
- Updated `shared_status` attributes `ltssm_state` and `ltssm_substate` to update after the required services for VIP has been performed in an LTSSM state.
- Added new attributes in `svt_usb_status` class. For example, `ltssm_reset_state` and `ltssm_present_substate` to update as soon as link enters a particular state.
- Added delay variable `eusb2_delay_before_port_repeater_config` to wait before port or repeater configuration. Default value is 0us.
- Added delay variable `eusb2_delay_before_initial_extse1` to wait before driving EXTSE1 initially. Default value is 0us.

6.40 Notes for R-2020.12-2 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.12-2
- MPHY Version: R-2020.12-2
- Common Version: R-2020.12-2
- SVT Version: R-2020.12

6.40.1 Updates

Added the following:

- Updated test `usb_20_na_31_ssp_ltssm_polling_portconfig_to_ss_inactive` test to initiate SS.Inactive to SS.disabled transition after P0 to P2 transition is completed.
- Updated `u2_inctivity_timer` to avoid race condition when start/stop timer is called before any active thread is completed.

6.41 Notes for R-2020.12-1 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.12-1
- MPHY Version: R-2020.12-1
- Common Version: R-2020.12-1
- SVT Version: R-2020.12

6.41.1 Updates

Added the following:

- Updated sequence `svt_usb_20_na_30_ss_random_ss_disabled_transition_system_virtual_sequence` to initiate SS.disabled to Rx.detect transition after P0 to P3 transition completes.
- Adding Analog Mixed Signal modeling support of eUSB serial bus.
- Added a new check `unexpected_lfps_signaling_on_non_config_lane` to report error if LFPS is detected on the non-config lane.
- Failed link send out to SYNC ordered-set in Polling.Active state due to common counter(for outgoing TSEQs and TS1/TS2s) being used for scheduling SYNC has been updated.
- Coverage support added for Hub Descriptor, Device, Configuration, and BOS descriptor.

6.42 Notes for R-2020.12 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.12
- MPHY Version: R-2020.12
- Common Version: R-2020.12
- SVT Version: R-2020.12

6.42.1 Updates

Added the following:

- Added support for IEEE UVM 1800.2.2017-1.0 and IEEE UVM 1800.2-2017-1.1.
- VIP is compatible with VCS +lint= `LRM_1800_2009` option.
- Ported all passive monitor checker on UTMI interface to active agent.
- `Ta_bdis_acon` timer of OTG A device updated to operate once SE0 post Suspend J detected.

- Coverage for eusb2 repeater enhanced for RAP command to initiate in power management state.
- Added UTMI checker for usb packet traffic on UTMI bus, when state is in L1/L2 suspend state.

6.43 Notes for R-2020.09-3 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.09-3
- MPHY Version: R-2020.09-3
- Common Version: R-2020.09-3
- SVT Version: Q-2020.03

6.43.1 Updates

Added the following:

- VIP clock recovery logic updated to work with VCS compile time switch -xprop.
- Updating sequence `svt_usb_20_na_30_ss_ltssm_state_transition_virtual_sequence` for test `usb_20_na_30_ss_ltssm_compliance_mode_to_ss_disabled_due_to_directed_transition` to initiate SS.disabled to Rx.detect transition after P0 to P3 transition completes.
- Updated debug interface to map serial interface signals for both the lanes.

6.44 Notes for R-2020.09-2 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.09-2
- MPHY Version: R-2020.09-2
- Common Version: R-2020.09-2
- SVT Version: Q-2020.03

6.44.1 Updates

Added the following:

- Fix to block low power initiation by the VIP while it has already received a low power entry request
- New services link added to enable/disable this feature : `USB_SS_ENABLE_HANDLE_CONCURRENT_LGO` and `USB_SS_DISABLE_HANDLE_CONCURRENT_LGO`
- SOF interval rule checking is updated to work with scenario where EXTSE1 is on the bus and frame timer expires.
- eUSB mode LS clock recovery logic updated to have eDp bus as input. Improving accuracy of clock recovery.
- Added support for bit add/remove error scenarios for Gen2 serial interface
- Added wait of clocking block before driving extra bit for bit add error scenario
- Supported back to back RAP command operation in eusb2 mode.
- Supported RAP command operation in Lx state in eusb2 mode.

- Added a new cfg `enable_flush_implementation_queue_for_ended_packet` to flush packet implementation queues once the packet ends for enhanced performance for high payload transfers.
- Soft disconnect handled in repeater when device repeater already in `L0_TX` state

6.45 Notes for R-2020.09-1 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.09-1
- MPHY Version: R-2020.09-1
- Common Version: R-2020.09-1
- SVT Version: Q-2020.03

6.45.1 Updates

Added the following:

- Disconnect state entry is supported in host state machine for FS/LS during the resetting state.
- The Device VIP is updated to stop U1/U2 inactivity timers while tPingTimeout timer(configured using `device_cfg.t_ping_timeout`) is on.
- New services are added to start/stop U1/U2 inactivity timers `USB_SS_START_U1_U1_INACTIVITY_TIMERS` and `USB_SS_STOP_U1_U1_INACTIVITY_TIMERS`.
- The VIP can now send two back to back SYNCs due to transition from Polling to Hotreset on the boundary of 32 TS1/TS2s.
- The compliance mode state where the the slave does not move from CP9 to CP10 after receiving PING.LFPS when `cfg.cp9_zeros` is configured to the maximum value has been addressed.
- The VIP can now accept erroneous ordered sets(TS2) when the error symbol matches the TS1A ordered set.

6.46 Notes for R-2020.09 Release

Updates for this release are as follows:

- USB_SVT Version: R-2020.09
- MPHY Version: R-2020.09
- Common Version: R-2020.09
- SVT Version: Q-2020.03

6.46.1 Updates

Added the following:

- Added support for IEEE UVM 1800.2.2017.
- Hub and USB3 device framework sequence update: Host and Device xfer ended waited in Sequence. Warning is thrown in place or error in sequence if transfer is aborted.
- Timer is calibrated with 1ns. `` `SVT_USB_WAIT_XFER_START `` is added to delay start of a transfer before initiating.
- Packet services are allowed in states other than U0.

- Dependency of define `SVT_MULTI_SIM_JUMP_STATEMENTS_IN_AUTOMATIC_METHOD` is excluded as latest versions of all simulators support return in task.
- Sequence configuration `svt_usb_device_framework_dev_xfer_ended` added to enable wait for `set_address` and `get_descriptor` device xfer ended.
- Macro `SVT_USB_CUSTOMIZED_RECONFIGURE` added to customize Reconfigure Support. If enabled it will make reconfigure independent of the testbench.

6.47 Notes for Q-2020.06-3 Release

Updates for this release are as follows:

- USB_SVT Version: Q-2020.06-3
- MPHY Version: Q-2020.06-2
- Common Version: Q-2020.06-2
- SVT Version: Q-2020.03

6.47.1 Updates

Added the following:

- Added `hs_micro_frame_tolerance` and `fsls_frame_tolerance` configuration variable with default values of 62.5ns and 500ns respectively as per section 7.1.12 of USB specification.
- Adding configuration variable named `min_tinactivity_hs_bit_time` to consider a HS Non-Idle state to be valid critera for discontinuing tinactivity timer only if its minimum duration is 1 HS Bit time.
- Added new timer `tattdb_min` to check less than tattdb timer (tattdb - `` `SVT_USB_TATTDB_DEBOUNCE_INTERVAL ``) for passive monitor to decode connect as hostdisconect signal takes time minimum 2.5us time to clear. Debouce Macro default duration is 3us.
- Updated in eusb2 Native mode to not drive `tsedr_hsfslse0` timer of St0 during remote wakeup.
- Updated SOF checking rules to operate after First SOF and not after entering into L0 state.
- Before device mode VIP drive chirpK, SE0 will be decoded for tfiltse0 duration for valid reset detection without clock presence and now vip modified to check that after clock arrived in utmi interface.

6.48 Notes for Q-2020.06-2 Release

Updates for this release are as follows:

- USB_SVT Version: Q-2020.06-2
- MPHY Version: Q-2020.06-1
- Common Version: Q-2020.06
- SVT Version: Q-2020.03

6.48.1 Updates

Added the following:

- Updated Host passive monitor to identify and isolate L1 resume exit scenario either to be either Host initiated or Device initiated.

- Added Td timer support in eUSB Control Message. Default value of `cfg.td` is 0ns.
- Updated RAP support to work with eUSB Native mode.
- Updating eUSB logic to handle randomly cutting off Vbus anytime during Protocol Reset.
- Added a feature in link TX to prioritize a packet over link commands using packet class attribute `is_express_packet`.
- Added an update for compliance mode state to check for `cfg.speed` instead of working speed while scheduling compliance patterns.
- Changed configuration class attribute `u2_timeout_factor` from static to dynamic in order to reconfigure it during run time.
- Updated the check implementation of `pipe_dut_mac_invalid_back_to_back_mac_requests` check to trigger an error if RESET occurs while processing a service request from MAC.

6.49 Notes for Q-2020.06-1 Release

Updates for this release are as follows:

- USB_SVT Version: Q-2020.06-1
- MPHY Version: Q-2020.06
- Common Version: Q-2020.06
- SVT Version: Q-2020.03

6.49.1 Updates

Added the following:

- Updated UTMI checker for tdrst timing and trstrcy check.
- Sequence support added for eusb2 utmi mode on short reset test case.
- Added Resuming state wait condition for all suspend resume/remote-wakeup sequence. This does not transition to enable state without entering resume state without disconnecting or resetting.
- Added support for payload corruption in a tunneled packet using link packed transaction exception.
- Fixed the compliance mode issue where the transmitter gets blocked after scheduling the last CP6.
- Added update for disabling data when physical reset is bypassed during power on reset or system reset.

6.50 Notes for Q-2020.06 Release

Updates for this release are as follows:

- USB_SVT Version: Q-2020.06
- MPHY Version: Q-2020.06
- Common Version: Q-2020.06
- SVT Version: Q-2020.03

6.50.1 Updates

Added the following:

- Support to issue warm reset when there is no active packet or link command.
- New classes to insert exceptions in `svt_usb_link_packed_transaction`.
- Compliance mode support for USB 3.2
- A new check `pipe_unexpected_rate_change_signaling` to verify the validity of the rate change request.