

Intel(R) System Debugger NDA Release Notes



Intel® System Debugger is a modern, GUI-based architecture system software debugger. It provides a toolset to explore the state of the system, capture and analyze traces from hardware, firmware, and software sources, analyze system crash logs, and much more.

This document provides information on new features, system requirements, limitations, and legal information for all components of the Intel® System Debugger 2023 NDA for Windows* and Linux* OS hosts, which includes following tools:

- Intel® System Debugger - System Debug
- Intel® System Debugger - System Trace
- Intel® System Debugger - Crash Log
- Intel® Debug Extensions for WinDbg (Windows OS host only)
- Intel® Intelligent Debug & Validation Tool (IDV Tool) (Windows OS host only)
- Additional tools for target connection.

To obtain and install Lauterbach Trace32* software, refer to [additional instructions](#) in the User Guide.

Get Help

To submit an issue, go to the [Online Service Center](#) and select Intel® System Bring-up Toolkit NDA.

Alternatively, you can report issues at [Intel® Premier Support](#). For additional support resources, go to <https://software.intel.com/content/www/us/en/develop/tools/support.html>.

Version History

Note

Intel® System Debugger U2350 does not include the latest functional and security updates. Intel® System Debugger U2406 is targeted to be released in February 2024 and will include additional functional and security updates. Customers should update to the latest version as it becomes available.

Date	Version	Major Change Summary
December, 2023	U2350	Platform Register context for Simics® simulator connections. Multiple improvements to the MSR and PCI views. New On-Target connection support in TCA and Trace.
November, 2023	U2346	Various improvements to the I/O Port and MSR views. Support for decompsing debug tokens. Reworked the command line interface for the OnTarget Agent.
October, 2023	U2342	Support reading of multiple part IDs. Support NPU driver in windbg. Custom number of columns in the I/O Port view.
September, 2023	U2338	Improved Interrupt Descriptor Table view. DnX support in the UI. New command to dump information about pci devices like MSFT.
August, 2023	U2334	Full compatibility with RISC-V for Interrupt Descriptor Table. Re-enable telemetry. BMC support.
August, 2023	U2330	Added intrusiveness control to new ITH Windows* Driver. Intrusive flag allows to prevent power managing of Intel® Trace Hub device by OS Windows. When intrusive flag is set, trace captures will be in lossy mode, but preventing low power states. Default mode is to allow low power states, but loss messages when memory is not available for use by Intel® Trace Hub. Improved overall tool and platform stability, see release notes section for further information.
July, 2023	U2326	IDV Tool added Drag & Drop functionality for Signal Order in Setup Tabs and in the Waveform viewer. Preview of Debug Token injection and Firmware update using Intel® Download and Execute (Intel® DnX). Improved Trace Target Agent Eclipse integration, support usage of local SSH config files and auto re-connect. Added System Debug I/O Port, Raw MSR and RISC-V PageTables Views. Intel® Debug Extensions for WinDbg added support for remote hardware debugging. Updated IDE to Eclipse 2023-03 using JRE 17.
June, 2023	U2322	Implemented customization of LoadThis command by using the extension "Load Module (Custom Search)". Added raw PCI config space tab to the PCI devices view. Added Simics® simulation support for Panther Lake.
May, 2023	U2318	Improved Intel® System Debugger User Guide to include IPCCLI documentation and Intel® System Debugger Python Modules API documentation. Added new debug token configurations to support all platforms previously supported with Platform Flash Tool (PFT), added auto selection of decode catalog for Trace CLI. Implemented auto complete of remote host addresses to new Connection dialog.

Date	Version	Major Change Summary
April, 2023	U2314	Implemented auto detection for Closed Chassis Adapter + 2-Wire Adapter / I3C Adapter. Enabled Trusted Domain Execution (TDX) debugging. Add processor trace events decoding. Added Simics® simulator guest physical memory access and support for operating modes. Implemented Register Viewer based on RegDB for WinDbg*. IDV Tool: Added Platform Connected support, Rule Copy across projects and Rule Creation Date column under Setup Rules. Finally removed Intel® DFX Abstraction Layer support.
March, 2023	U2310	Improved Support for MTL-P 682 B1 Stepping. System Debugger include TDX registers on MSRs list. Added offline debug token creation flow using Flash Programming Tool (FPT).
February, 2023	U2306	System Debug support automatic hardware topology updates. Show Simics® simulator specific connection indicator in IDE. Added support for periodic Intel® Trace Hub flush to TraceCLI EmergencyBoot. Extend debug token support for: DG1, Tiger Lake platforms and Elkhart Lake-based platforms. Implemented support for offline debug token generation.
December, 2022	U2250	Added support for Hypervisor-Managed Linear Address Translation (HLAT), added Simics® simulator reverse debugging support for Intel® Debug Extensions for WinDbg, improved source level debugging support for Intel® Debug Extensions for WinDbg, implemented Eclipse* Help / Browser navigation actions (added home, backwards and forward)
November, 2022	U2246	Introduced System Trace On-target trace capability, add support to subset of Structured Binary Data specification from MIPI SyS-T Specification v1.1, Intel® Debug Extensions for WinDbg: Add debug support for Simics® simulator instances, Implemented generate status report functionality to store all available information.
September, 2022	U2238	Platform Provisioning to manage debug tokens, preview of new Trace Message Viewer, new live target information view, support for Architectural LBRs in WinDbg over Intel® DCI Extensions; Support for Flexible Return and Event Delivery (FRED).
August, 2022	U2234	Introduced new Desktop Shortcuts for Target Indicator and Intel® Debug Extensions for WinDbg, Update Eclipse base to 2022-06, Added Simics disassembly and source-level debugging support for VPU firmware (including SHAVE cores)
July, 2022	U2230	Support for breaking on BIOS POST codes (conditional IO breakpoints), support for LBR-based instruction trace on Lakemont / Minutela cores on Simics® simulation, enabled Simics® simulation support for Granite Rapids-D / Kaseyville target.
June, 2022	U2226	Added early Meteor Lake Support, new more intuitive Connection and Reset Toolbar controls, System Trace integrated a memory management library that allows users to limit the amount of memory decoding process uses, Intel® Debug Extensions for WinDbg: Added debug support for pre-NT environment, IDV Tool: Enhancements and fixes to rule editing: undo and redo, copy and paste, drag and drop.
May, 2022	U2222	Updated the debugger to use the latest Eclipse* 2022-03 base, improved Eclipse* context help (use HTML user guide), added support for Zephyr 2.7 based Intel® Internet of Things Group (IOTG) Intel® Programmable Services Engine (Intel® PSE) SDK, implemented new Intel® System Debugger - System Debug Symbol Browser View.

Date	Version	Major Change Summary
April, 2022	U2218	Skipped.
March, 2022	U2214	Added remote debug support for hardware platforms, introduced advanced rule creation functionality for Intelligent Debug & Validation Tool (IDV Tool), fixed issues.
March, 2022	U2210	Enabled tracing of <code>ptwrite</code> instruction and power events by System Debug, integrated Target Connection Assistant into <code>ExdiServer</code> console of WinDbg, fixed issues. For details, see Change History.
January, 2022	U2206	Added new environment and launcher scripts, updated the debugger to use the latest Eclipse* 2021-12 base, upgraded the BSOD bugcheck database, enabled Intelligent Debug & Validation Tool (IDV Tool) to toggle rule-checking for historical captures, extended target support, fixed issues. For details, see Change History.
December, 2021	U2152	Introduced Log4j version update.
December, 2021	U2150	Created a single Eclipse* perspective for all Intel® System Debugger components, enabled support for Python* 3.8 by TraceCLI and Target Agent, enabled TraceCLI to install and remove trace extensions, added support for hardware breakpoints by WinDbg over Intel® Direct Connect Interface (Intel® DCI), added support for Windows* 11 and Rocky Linux* host OS, fixed issues. For details, see Change History.

To check Release Notes for previous product versions:

- Online: open the [online book](#) and expand the Version drop-down menu on the top-left.
- Offline: go to the [Intel® System Debugger download page](#) at Intel® Products and select a required version from the drop-down menu.

Supported Systems and Tools

See the child pages of this section for detailed information about host operating systems, target platform configurations, connection methods, and additional tools supported in this release.

Host Operating Systems

Intel® System Debugger 2023 NDA supports the following host operating systems:

- Microsoft Windows* 11
- Microsoft Windows* 10 21H1/21H2
- Microsoft Windows* Server 2019
- Microsoft Windows* Server 2022
- Ubuntu* 20.04 LTS
- Ubuntu* 22.04 LTS
- Rocky Linux* 8, 9
- Fedora Linux* 37, 38, 39

- RHEL 9
- SUSE Linux Enterprise 15 SP
- Debian* 11

Due to compatibility issues, Intel® System Debugger 2023 NDA does not support older Linux versions than those mentioned above.

For questions and assistance, contact [Customer Support](#).

Supported Platforms and Probes

Each Intel® System Debugger tool supports its own set of platforms and probes. The following probes are commonly used for connection:

- Intel® In-Target Probe (Intel® ITP) - XDP3

ITP-XDP3 in the table.

- Intel® Direct Connect Interface (Intel® DCI) Out of Band (OOB) via Intel® Silicon View Technology (Intel® SVT) Closed Chassis Adapter (CCA).

CCA (DCI OOB) in the table

- Intel® DCI 2-wire via Intel® SVT Closed Chassis Adapter (CCA).

CCA (DCI 2-wire) in the table.

- Intel® SVT Debug Class (DbC) USB Debug Cable.

DbC in the table.

- Intel® DCI OOB via Intel® SVT Closed Chassis Adapter (CCA) + Intel® DCI USB Debug Class (DbC). This method suggests simultaneous use of a DbC cable and a CCA adapter.

CCA + DbC in the table.

- Simics® simulator – virtual target simulation.

See also

[Learn More about Debug Probes and Technologies](#)

Debug and Trace

The table below provides an overview of target/probe combinations supported by System Debug and System Trace components of the Intel® System Debugger NDA in the present release.

Note

The core count of the target platform component does not affect connection and debugging experience. If you face any issues, refer to the [Known Issues section](#) of Release Notes or contact Customer Support.

Target Components	Connection Methods: System Debug	Connection Methods: System Trace
Ash Creek Falls CPU / Lewisburg PCH	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Alder Lake CPU / Alder Lake PCH-N	ITP-XDP3, CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI 2-wire), DbC
Alder Lake CPU / Alder Lake PCH-P	ITP-XDP3, CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC
Alder Lake CPU / Alder Lake PCH-S	ITP-XDP3, CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI 2-wire), DbC, Simics® simulator
Arctic Sound GPU	Simics® simulator	Lauterbach* MIPI60 probe
Arrow Lake CPU / Meteor Lake PCH-S	ITP-XDP3, CCA (DCI 2-wire), DbC, Simics® simulator	ITP-XDP3, CCA (DCI 2-wire), DbC, Simics® simulator
Bay Trail SOC (MinnowBoard MAX*, Valleyview2)	ITP-XDP3, Simics® simulator	
Broxton I SOC (Apollo Lake)	DbC, Simics® simulator	DbC
Broxton P SOC (Apollo Lake)	DbC, Simics® simulator	DbC
Broxton P SOC: Gordon Peak Modular Reference Board (GP MRB)	DbC, Simics® simulator	DbC
Cannon Lake CPU / Cannon Point PCH-LP	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Cascade Lake CPU / Kaby Lake PCH-H (Glacier Falls)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Cascade Lake CPU / Lewisburg PCH (Purley Refresh)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Coffee Lake CPU / Cannon Point PCH-H	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC

Target Components	Connection Methods: System Debug	Connection Methods: System Trace
Coffee Lake CPU / Kaby Lake PCH-H	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Coffee Lake CPU / Cannon Point PCH-LP	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Comet Lake CPU / Comet Lake PCH-H	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Comet Lake CPU / Cannon Point PCH-H	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Comet Lake CPU / Comet Lake PCH-V	CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Comet Lake CPU / Tiger Lake PCH-H (Rocket Lake S)	ITP-XDP3, CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC
Comet Lake CPU / Comet Lake PCH-LP	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Cooper Lake CPU / Lewisburg PCH (Cedar Island)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Denverton SOC (Car Branch 1, Car Branch 2, Denverton)	DbC, Simics® simulator	DbC
DG1 PCI Express Card GPU	Simics® simulator	Lauterbach* MIPI60 probe
DG2 PCI Express Card GPU	Simics® simulator	Lauterbach* MIPI60 probe
Diamond Rapids CPU	Simics® simulator	Simics® simulator
Elkhart Lake CPU / Mule Creek Canyon PCH	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC
Emerald Rapids CPU / Emmitsburg PCH (Eagle Stream Refresh)	ITP-XDP3, CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator
Gemini Lake SOC	DbC, Simics® simulator	DbC
Gemini Lake Refresh SOC (Gemini Lake +)	DbC, Simics® simulator	

Target Components	Connection Methods: System Debug	Connection Methods: System Trace
Grand Ridge CPU / Emmitsburg PCH	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator
Granite Rapids CPU	ITP-XDP3, Simics® simulator	ITP-XDP3, Simics® simulator
Granite Rapids D (Kaseyville) CPU	ITP-XDP3, Simics® simulator	ITP-XDP3, Simics® simulator
Ice Lake CPU / Cedar Fork PCH	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Ice Lake CPU / Lewisburg PCH (Whitley)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Ice Lake CPU / Ice Lake PCH-LP	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC
Ice Lake CPU / Ice Lake PCH-N	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC
Jasper Lake CPU / Jasper Lake PCH-N (Jasper Lake +)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Kaby Lake CPU / Kaby Lake PCH-H	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Kaby Lake CPU / Sunrise Point PCH-H	ITP-XDP3, CCA (DCI OOB), Simics® simulator	CCA (DCI OOB)
Kaby Lake CPU / Sunrise Point PCH-LP	CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Lakefield SOC	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC
Lunar Lake CPU / Lunar Lake SoC M	ITP-XDP3, CCA (DCI 2-wire), DbC, Simics® simulator	ITP-XDP3, CCA (DCI 2-wire), DbC, Simics® simulator
Meteor Lake CPU / Meteor Lake PCH-S	ITP-XDP3, CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI 2-wire), DbC, Simics® simulator
Raptor Lake CPU / Alder Lake PCH-S	ITP-XDP3, CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator
Rocket Lake CPU / Comet Lake PCH-H	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Rocket Lake CPU / Tiger Lake PCH-H (Tatlow)	ITP-XDP3, CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC

Target Components	Connection Methods: System Debug	Connection Methods: System Trace
Sapphire Rapids CPU / Alder Lake PCH-S (Fishhawk Falls)	ITP-XDP3, CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator
Sapphire Rapids CPU / Emmitsburg PCH (Eagle Stream)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC, Simics® simulator
Sierra Forest CPU	ITP-XDP3, Simics® simulator	ITP-XDP3, Simics® simulator
Sierra Forest CPU / Emmitsburg PCH	ITP-XDP3, Simics® simulator	ITP-XDP3, Simics® simulator
Skylake CPU / Sunrise Point PCH-H	ITP-XDP3, CCA (DCI OOB), Simics® simulator	CCA (DCI OOB)
Skylake Server CPU / Kaby Lake PCH-H (Basin Falls)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Skylake Server CPU / Lewisburg PCH (Anderson Creek, Purley)	ITP-XDP3, CCA (DCI OOB), CCA+DbC, DbC, Simics® simulator	CCA (DCI OOB), CCA+DbC, DbC
Skylake CPU / Comet Lake PCH-LP	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	
Skylake CPU / Sunrise Point PCH-LP	ITP-XDP3, CCA (DCI OOB), Simics® simulator	CCA (DCI OOB)
Snow Ridge CPU / Cedar Fork PCH (Jacobsville)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Spring Hill CPU / Ice Lake PCH-LP	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC
Tiger Lake CPU / Tiger Lake PCH-H	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator
Tiger Lake CPU / Tiger Lake PCH-LP	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator	CCA (DCI OOB), CCA (DCI 2-wire), DbC, Simics® simulator
Tunnel Creek SOC (Lincroft)	ITP-XDP3, Simics® simulator	
Whiskey Lake CPU / Comet Lake PCH-LP	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Whiskey Lake CPU / Cannon Point PCH-LP	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC

Target Components	Connection Methods: System Debug	Connection Methods: System Trace
Whiskey Lake CPU / Sunrise Point PCH-LP (Amber Lake Y 4+2)	ITP-XDP3, CCA (DCI OOB), DbC, Simics® simulator	CCA (DCI OOB), DbC
Ponte Vecchio (PVC) PCI Express Card GPU		Lauterbach* MIPI60 probe
Battlemage		Lauterbach* MIPI60 probe, Simics® simulator
Panther Lake	Simics® simulator	Simics® simulator
Clearwater Forest	Simics® simulator	Simics® simulator
Granite Rapids Workstation (Meteor Lake PCH)	Simics® simulator	Simics® simulator

Crash Log

Intel® System Debugger - Crash Log supports the following target components in the present release:

- Alder Lake CPU (variants: M, N, P, S)
- Alder Lake PCH (variants: N, P, S)
- Broxton I, Broxton P
- Comet Lake PCH (variants: H, LP)
- Cannon Point PCH (variants: H, LP)
- Emmitsburg PCH
- Ice Lake CPU (variants: H, I, S3, S4, UN/YN)
- Ice Lake PCH (variants: H, LP, N)
- Jasper Lake CPU
- Jasper Lake PCH-N
- Lake Field CPU (base die and compute die)
- Meteor Lake CPU / Meteor Lake PCH-S
- Mule Creek Canyon PCH
- Raptor Lake CPU
- Rocket Lake (variants: S, U)
- Sapphire Rapids SP CPU
- Tiger Lake CPU (variants: H, R, S, UN/YN)
- Tiger Lake PCH (variants, H, K, LP)

Additional Tools Versions

Tool	Used on OS host	Version
OpenIPC	Windows* OS and Linux* OS	23.46.13351.200
IPCCLI	Windows* OS and Linux* OS	4.23.24604.100
Lauterbach Trace32* (T32 scripts supported by Intel)	Windows* OS	23.42.695.200

Installation

Once you have downloaded an executable installation file from Intel® Products, the wizard will guide you through the installation procedure. Check the text below for additional information that may be useful during installation.

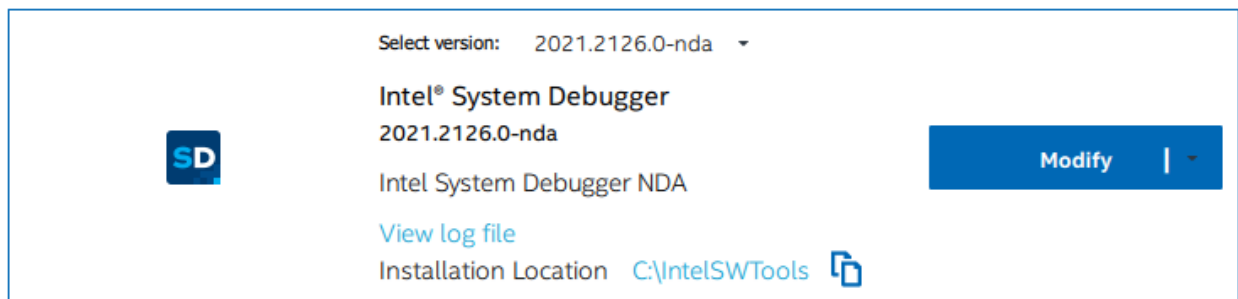
Installing on Windows* OS Host

The default installation directory (referred to as `<install-dir>` in primary documentation) is `C:\IntelSWTools\system_debugger\<version>-nda`, where `<version>` is a four-digit number indicating the year and a workweek when the release is out (for example, 2138).

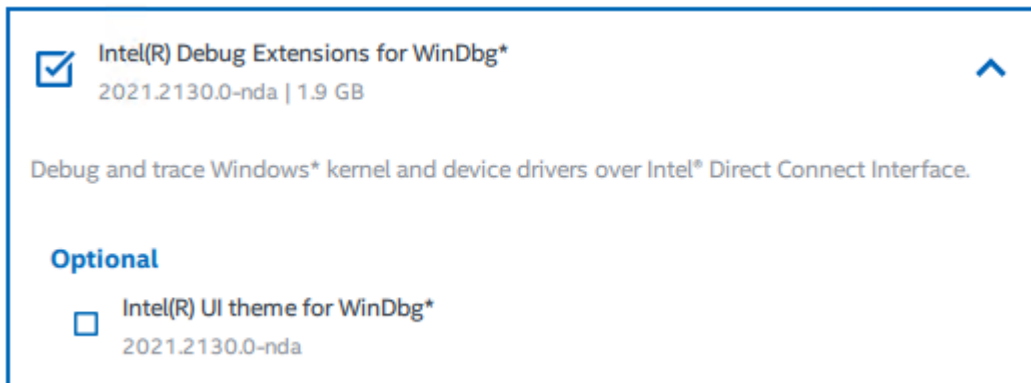
Selecting UI Theme for Intel® Debug Extensions for WinDbg

By default, a special theme designed by Intel is applied to your WinDbg installation. You can modify your theme selection (uninstall or reinstall it) after the product installation as follows:

1. Launch your installer again.
2. Select the latest version of Intel® System Debugger NDA, click **Modify**, and select **Add/Remove components**.



3. Expand the Intel® Debug Extensions for WinDbg component, (un)check the **Intel® UI theme for WinDbg*** box, and let the installer modify the product.



Without this component installed, WinDbg will use the default theme by Microsoft*, which will not affect product performance.

Installing on Windows* OS Host for On-target usage

On-target trace collection requires the installation of only a few components. It is recommended to install the System Trace package and its hard dependencies like DFLaunch and OpenIPC Debug Library.

In addition, On-target trace requires the installation of Intel® Trace Hub (Intel® TH) driver version 10.0.22000.549 or later. For information on how to get to the driver, please follow the System Trace documentation.

Installing on Linux* OS Host

Important

You need to install the product as root.

The default installation directory (referred to as `<install-dir>` in primary documentation) is `/opt/intel/oneapi/system_debugger/<version>-nda`, where `<version>` is a four-digit number indicating the year and a workweek when the release is out (for example, 2138).

New Features

Target Connection Assistant

- Added a new dialog in the UI to show the latest version of Intel® System Debugger.
- Added support in the CLI for connecting to the local system, called On-Target connection.

Intel® System Debugger - System Debug

- Enabled Platform Register context for Simics® simulator connections. From Simics® simulator Base 173 a new context, called Platform Registers has been accessible. This context is now accessible through the

Platform Registers Dictionary view. R/W actions can be performed to the register that are show on this context. Also they can be added to Platform Register Watch view and perform R/W operation on the bitfields through the Register Editor View. Due to the high among of register that this context contains the search is also enabled.

- Added limited support for the extended PCI configuration space. Current implementation supports the extended PCI configuration space during early EFI boot stage only. Added a new context called PCIExtCfg. This context is now accessible through the PCI Devices view, when the extended PCI configuration space is enabled with the new button.
- Added hexdump export format for I/O Port, MSR and PCI raw views.
- Added write access for PCI Devices view.
- Extended Load Current Module, LoadPEIM, BIOSROM and LoadDXE support for BIOS (EDK II) GCC builds.

Intel® System Debugger - System Trace

- Added TraceCLI command line options for Catalog Service configuration (check 'intel_systrace catalog_remote -help')
- intel_ontarget_trace now supports platform detection using TCA, as well as detection of BIOS, CSME and Audio FW version. This information will be used to select right catalog.
- Added support for ARL-H.
- Added support for GNR-D.

Intel® Debug Extensions for WinDbg

- Support for extraction of crashlog data from live target and memory dump using Intel® Debug Extensions.

Fixed Issues

Target Connection Assistant

- Fixed various issues with the UI terminal on recent versions of Windows 11.

Intel® System Debugger - System Debug

- Fixed kernel module loading with "Load Linux Kernel Modules from Kernel Image Folder" button.
- Fixed source loading in module view for elf symbols.
- Fixed PCI Devices raw view for devices with address 0.

Intel® System Debugger - System Trace

- Fixed ACE catalog matching in TraceCLI.
- Fixed memory extraction by using trace extraction service.
- Fixed pre-BIOS NPK tracing from Bootstall Throws Errors.

- Fixed early boot script fails during capture. Note: due to bad power supply might get Power On events during power Off, that would confuse early boot script.
- Run memory sanitizer and thread sanitizer on engine and tms services. Fix various number of issues that improves tool stability.

Intelligent Debug & Validation Tool (IDV Tool)

- Enabling analog capture has been blocked for signals in BMC configurations, BMC supports only digital signals.
- Fixed capture time value auto-multiplication if time unit was not entered.

Known Issues

Cannot launch Target Indicator, getting error about missing VCRUNTIME140_1.dll / Cannot do any tracing with “Failed to load platform” error (Windows* host only)

Issue: This should not be an issue, but in case it appears - it means Microsoft* C++ Redistributables package is broken.

Workaround:

1. Go to “Add/Remove Programs”.
2. Uninstall all “Microsoft C++ Redistributables”.
3. Uninstall Intel® System Debugger.
4. Install Intel® System Debugger again.

Trace and debug operations on Alder Lake S require changes in platforms OS workaround settings

Issue: Power state transition (S5, S4, WR) with probe mode detection method TapPolling causes debug tool hang. Meanwhile, TapPolling is required for run control on targets with OS running. Starting with U2122 release, Intel® System Debugger NDA enables you to manage probe mode detection logic based on use cases.

Workaround: Follow instructions at [<install_dir>/documentation/tca/usage/connecting_gui.html](install_dir/documentation/tca/usage/connecting_gui.html) .

Depending on your goals, complete step 7 as follows:

- For BIOS debugging or FW/SW tracing, select “Target runs no OS” (Default). Debugging and tracing on platform that does power transitions requires this option selected.
- For working with Architectural Event Traces (AET) or WinDbg over Intel® Direct Connect Interface (Intel® DCI), select “Target runs an OS”.

Disconnecting and reconnecting Intel System Debugger to Alder Lake S target with survivability flag enabled causes NorthPeak (NPK) hardware hang

Issue: During NPK tool disconnect and reconnect, NPK IP is power gated and then power un-gated and the default register value causes timeout and hang.

Workaround: Disable survivability mode if debugging across low power states is not required.

Low power debug is not supported for Alder Lake P due to missing Low Voltage Margining (LVM) overrides

Issue: Alder Lake P Best Known Configuration (BKC) does not support low power.

Workaround: N/A

Unicode error returned during target selection via console

Issue: When selecting a target with a command like `trace.interactive_setup()`, an error like *UnicodeEncodeError: 'latin-1' codec can't encode character 'u2122' in position 24: ordinal not in range(256)* is displayed.

Workaround: Before starting the console session, execute the following command:

- Windows: `set PYTHONIOENCODING=UTF8`
- Linux: `export PYTHONIOENCODING=UTF8`

Cannot connect Intel® System Debugger to Alder Lake P target with USB3 Debug Class (DbC)

Issue: Debug connection is not working with USB3 DbC connected to Alder Lake P. Reset break is also not functional.

Workaround: There are two ways to resolve connectivity issues with Intel System Debugger if USB3 DbC is detected:

- In the final step of establishing target connection (“Connection Summary” dialog box), expand the “Platform Debug Configuration” tab and uncheck the “Configure High Bandwidth Transport” box.
- Modify IFWI via Intel® Flash Image Tool (Intel® FIT) to enable USB2 port and avoid USB3 DbC detection.

Cannot establish USB DbC connection to debug the 10th generation Intel® NUC with Alder Lake M or Alder Lake P

Issue: When attempting to connect to the target using USB DbC, both Target Indicator and Windows* Device Manager report errors on the USB device (USB DbC probe).

Workaround: N/A

Target Indicator is not visible when being started on Linux* hosts with GNOME 3.26 and above (Linux* host only)

Issue: On Linux* hosts with GNOME 3.26 and above, the Target Indicator is not visible when being started because GNOME 3.26 removes the system tray that the Target Indicator heavily relies upon.

Workaround: You can start the Target Indicator executable with the `--window-only` (or short `-w`) flag, which makes the Target Indicator avoid using the system tray.

Cannot start the Target Indicator from a file explorer with older package versions (Linux* host only)

Issue: Starting the Target Indicator executable from a file explorer (for example, Nautilus) fails, since it is not recognized as an executable file.

Workaround: Upgrade the “file” package to a version higher than 5.36. Start the Target Indicator from the command line or via the provided application link (the package upgrade does not affect it). See the original [bug report](#).

Reset causes `libusb` error and loss of run control on Linux* host

Issue : Warm or cold reset of the target connected to Comet Lake CPU/Comet Lake PCH-H (CML-S) in user interface or command line (ipccli) causes `libusb` error and loss of run control. Cold reset is necessary to connect to the target again.

Workaround: Use the Intel® DCI USB DbC2 instead of USB DbC3 on the port used for the debug connection. You can configure this in the Intel® Flash Image Tool (Intel® FIT)

1. Launch the Intel® FIT (obtained from Intel® Validation Internet Portal) and change the following settings within your current configuration:

- Debug > Direct Connect Interface Configuration > Direct Connect Interface (DCI) Enabled > Yes
- Debug > Early USB DBC over Type-A Configuration > USB2 DbC port enable > USB2 Port <X>

where X identifies the post number. Depending on the target, set the following value:

- 1 - to enable USB2 DbC on the vertical USB type-A port on CML-U DDR4 boards
- 3 or 4 - to enable USB2 DbC on either of the two stacked USB type-A ports respectively on CML-U LP3 boards
- 8 - to enable USB2 DbC on the vertical USB type-A port on CML-S boards with CMP-H or CNP-H chipset.

2. Build the image and flash the target with the new image.

Disconnecting a probe crashes the target (Windows* host only)

Issue: Intel® Silicon View Technology (Intel® SVT) Closed Chassis Adapter (CCA) and Intel® In-Target Probe (Intel® ITP) - XDP3 probes are supporting hot plug/unplug from a target, whereas for Intel® DCI DbC the connection is bidirectional. If a probe connection is lost, the debugger reports a Power Loss event.

Workaround: If the target was running, reconnect the probe (the disconnection has no effect on the running target). If the target was halted, reboot the target and restart the debugging session (the debug context is lost).

Platform security policy might inhibit debugger operation

Issue: In some platforms, the security policy might disable JTAG access to the CPU. This is intended to prevent reverse-engineering. In this case, the Intel® System Debugger will successfully connect to the target, but will not be able to discover any CPUs on the JTAG bus and will report that no CPU is available.

Workaround: Ensure that that platform firmware has enabled access to the CPUs via JTAG. You can do it by flashing a special “debug” firmware into the target. In some cases the CPU or the CPU module might physically disable JTAG access, especially in production or near-production versions. In this case, please work with the platform business unit to obtain a JTAG-enabled hardware.

Issue: When you attempting to archive Intel® DAL logs using the commands `itp.daldebugloggerlevel("default")` followed by `itp.loggerarchive()`, a “Permission Denied” error is printed.

Workaround: Before every execution of the commands above, navigate to `<install-dir>\tools\DAL_<version>` and modify the `PublicLogging.log4net` file to replace “DALBase.log” with a location where you have write permissions (for example, `C:\Temp\DALBase.log`).

Cannot connect Intel® System Debugger to Raptor Lake CPU / Alder Lake PCH-S target with Intel® In-Target Probe (Intel® ITP) - XDP3

Issue: Debug connection is not working with Intel® ITP XPD3 connected to Raptor Lake CPU / Alder Lake PCH-S.

Workaround: Use a different debug probe (see Supported Platforms and Probes for available connection options).

Reset Break issues on Granite Rapids and Meteor Lake

Issue: Reset break does not work on properly Granite Rapids and Meteor Lake platforms. Planting and hitting reset break does not work reliable. Once it may work, other times it may fail sporadically to halt all cores.

Workaround:

- Meteor Lake users can try to use Intel® System Debugger 2246.
- For Granite Rapids there is currently no workaround.

Resume or step from Software Breakpoint issue on Granite Rapids D

Issue: Resume or step from Software Breakpoint issue on Granite Rapids D does not work properly. An attempt to resume execution or perform a step on a target halted in a Software Breaks does not work properly and can lead to a MCA exception.

Workaround: Use should use Hardware Breakpoints instead of Software Breakpoint.

Target Connection Assistant

Sapphire Rapids 4-socket modular system is not detected automatically

Issue: Sapphire Rapids 4-socket system is not automatically detected by the Target Connection Assistant.

Workaround:

1. Launch Intel® System Debugger, start establishing target connection, and choose “Manually select target” (for instructions, see the Intel System Debugger documentation).
2. In the final step of the connection wizard, click the gear icon and update the new OpenIPC /Provider Configuration / Configuration Name to “SPR_EBG_XDPA_Modular_4Mhz”

Intel® System Debugger - System Debug

Intel® System Debugger: Debugging in SEAM root mode is known to currently be unstable

Issue: Debugging in SEAM root mode is known to currently be unstable

Workaround: N/A

Intel® System Debugger: TD guest debugging might require the target to be Red unlocked

Issue: TD guest debugging might require the target to be Red unlocked

Workaround: N/A

Intel® System Debugger - System Trace

Rocky Linux 8 Support

Issue: Trace does not support Rocky Linux 8 because of GLIBC version mismatch (/lib/x86_64-linux-gnu/libm.so.6: version `GLIBC_2.29' not found).

Workaround: Update to Rocky Linux 9

CentOS* 7 and OpenSuse* 15 host OS are not supported (Linux* host only)

Issue: CentOS* 7 and OpenSuse* 15 host OS are not supported for System Trace use cases.

Workaround: N/A

Various error messages referring to Interview_Decoder.dll or visa64.dll are displayed (Windows* host only)

Issue: If the host system has IVI tools installed, the tool erroneously tries to load visa64.dll from C:\Windows\System32 or Interview_Decoder.dll from <install-dir>system_debugger_2020_ndasystem_tracebin.

Workaround: Do any of the following:

- Temporarily remove `C:\Windows\System32\visa64.dll` or uninstall the IVI tools.
- Check the information on the [Dynamic-Link Library Search Order](#) but be aware of the potential security implications.

On Discrete Graphic platforms (DG1/DG2), errors occur during connection attempt of trace capture start

Issue: Due to security policies in DG1 and DG2, access to North Peak Test Access Port over Lauterbach Processor Trace Interface connection is blocked, An error message “Failed Intel® Trace Hub hardware detection check!” appears.

Workaround: Currently, you can configure DG1/DG2 tracing only through firmware running on the target. To enable this tracing capability, launch Intel® Flash Image Tool (Intel® FIT), open Debug Settings, and configure the “Emergency mode”. To start tracing, configure the firmware as described above, connect the debug probe, deselect “Configure target during capture start and stop”, and press the “Start Capture” button in the Trace Tool. The error message “Error in connecting to the target. Target does not have power.” will appear but you can safely ignore it.

Issue: Using TraceCLI to capture traces fails with “Unknown mode specified, falling back to port list” error message (<https://jira.devtools.intel.com/browse/DEBGGR-11176>).

Workaround: Use Eclipse-based System Trace instead of TraceCLI.

Issues when using CCA 2wire for trace streaming

Issue: Connection get lost or invalid/no trace packets are received when using CCA 2wire while trace streaming is active.

Workaround: Use a different probe for trace streaming or use MTB as trace destination

Intel® Debug Extensions for WinDbg

Compatibility issues of Intel® Debug Extensions for WinDbg for Windows* 10 SDK

Issue: WinDbg over Intel® DCI plugin is not compatible for Windows* SDK versions 1809 (10.0.17763) and 1903 (10.0.18362.1).

Workaround: Install the latest Windows* SDK version 22000.

Latest Windows* 11 SDK is not able to download symbols.

Issue: Windbg over Intel® DCI plugin is not able to download symbols with Windows* 11 SDK versions 22H2 (10.0.22621.0).

Workaround: Install previous Windows* SDK version 22000 (10.0.22000.194), or latest version (build 25197 or newer) from Windows Insider Preview Builds.

Error collecting full memory dump in SPR targets

Issue: Windbg over Intel® DCI plugin is not able to collect full memory dump in SPR.

Intelligent Debug & Validation Tool (IDV Tool)

Captures are reported as starting later than the actual trigger

Issue: Sporadically captures are reported as starting later than the actual trigger.

Workaround: Trigger the same capture again.

Time difference between Analog and Digital samples

Issue: When capturing any channel as digital and analog a time difference between the two can be observed.

Workaround: None

Explanation: The analog values get sampled at a much lower speed and have to be transferred from the ADC to the IDV tool serially. Therefore an analog sample may have to wait n cycles, with n being the number of active analog channels up to a maximum of 16.

Stop trigger does not work when manual start trigger is used.

Issue: When capturing signals the stop trigger condition will not work if a manual start trigger was used.

Workaround: Configure a start trigger condition.

Change History

This page contains update summary of three previous releases. To check full Release Notes for previous product versions:

- Online: open the [online book](#) and expand the Version drop-down menu on the top-left.
- Offline: go to the [Intel® System Debugger download page](#) at Intel® Products and select a required version from the drop-down menu.

Intel® System Debugger 2023 NDA Update 2346

Check the summary below or see the [online version](#).

New Features

Target Connection Assistant

- Added an option in the UI to decompose debug tokens getting back knobs which the token was created with.

Intel® System Debugger - System Debug

- Write access for I/O Port and MSR views.
- Automatic address alignment for I/O Port view.
- Binary export format for I/O Port, MSR and PCI raw views.

Intel® System Debugger - System Trace

- Reworked the command line interface for the OnTarget Agent.
 - Added dedicated options to start/stop trace collection from the command line.
 - Added new command line flag 'fw-only' to only capture FW logs (without ETW messages).
 - The 'intrusive' flag is now enabled by default, use 'non-intrusive' to disable it.

Intel® Debug Extensions for WinDbg

- Support for creating full memory dump in non-existing folder.

Fixed Issues

Target Connection Assistant

- Fixed an issue that would cause connecting to a target to freeze due to the Windows Terminal on Windows 11.

Intel® System Debugger - System Debug

- Fixed ITrace view action management on a running target.

Intel® System Debugger - System Trace

- Fixed an issue with the loading of the default Profile in the trace configuration editor when using the ITP-XDP3 probe.

Intel® Debug Extensions for WinDbg

- Fixed exception when breaking preserving state of core.

Intel® System Debugger 2023 NDA Update 2342

Check the summary below or see the [online version](#).

New Features

Target Connection Assistant

- Added the ability to select multiple pid.bin files when clicking the “Load from File” button.

Intel® System Debugger - System Debug

- Added customizable number of columns to the I/O Port view
- Added option for copying the address from the Interrupt Descriptor Table view

Intel® System Debugger - System Trace

- Added support for Linux kernel tracing, see “Linux kernel tracing” chapter in the System Trace user guide for details
- Added OOB MSM trace support to Granite Rapids platform
- Rename VPU trace name to NPU
- Added support for Lunar Lake platform for TraceCLI Earlyboot

Intel® Debug Extensions for WinDbg

- Added support for Intel Neural Processing Unit (NPU) driver in Intel® Debug Extensions for WinDbg over Simics® simulator.
- Added OpenIPC version detection.

Fixed Issues

Target Connection Assistant

- A styling issue was fixed which would make it difficult to see which entry in the connection configuration wizard was selected.

Intel® System Debugger - System Debug

- Fixed I/O Port view endianness display
- Fixed environment variable enablement for hardware breakpoints over reset

Intel® System Debugger - System Trace

- Fixed trace streaming using CCA 4-wire probe
- Fixed timestamp issues (showing only ??) for TraceCLI Earlyboot

Intel® Debug Extensions for WinDbg

- Fixed !lpi and !cpuinfo functions for LNL platforms.

Intelligent Debug & Validation Tool (IDV Tool)

- Removed misleading tooltip message over vertical white bar at the end of waveform which said that waveforms to the right existed.

Intel® System Debugger 2023 NDA Update 2338

Check the summary below or see the [online version](#).

New Features

Target Connection Assistant

- Add UI support for debug token injection and generation via Intel(R) Download and Execute (DnX). Add DnX probe detection to correctly show status in the IDE.
- Add DnX command line interface (CLI). DnX python module provides “dnx” extension point to run DnX commands directly from the command line.

Intel® System Debugger - System Debug

- Improved the Interrupt Descriptor Table view to better support other CPU architectures
- Refresh event added for Address, Size and Count fields in I/O Port and MSR view on “Enter” key
- Added support for finding the bootstrap processor (BSP) on the Intel architecture

Intel® System Debugger - System Trace

- Added support for tracing Intel® Trace Hub memory and extraction on Battlemage platform

Intel® Debug Extensions for WinDbg

- Introducing support for pci devices discovery. Adds WinDbg* command to dump information about pci devices in a manner similar to MSFT.

Fixed Issues

Target Connection Assistant

- Fixed mapping of reset type in the Intel(R) CSME reset logic.
- Fixed debug token creation UI on Linux. Properly display debug token creation status.

Intel® System Debugger - System Debug

- Fixed the overlapping lines and the bit selection display in the Register Editor view

- Fixed the bit group “Size” in Platform Register view
- “Copy Stack Details” enabled for all threads and stack frames

Related Documentation

Primary documentation (Get Started and User Guide) for Intel® System Debugger is available in the following formats:

- [Online at Resource & Design Center](#) (click the link or search for number 648476)
- Offline HTML located at `<install_dir>/documentation/index.html`
- Embedded Eclipse* Help. Go to Help (or press F1) and select User Guide.

Documentation for OpenIPC is available at `<install_dir>/tools/python3/Lib/site-packages/ipcccli/html/index.html`.

Notices and Disclaimers

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Code names are used by Intel to identify products, technologies, or services that are in development and not publicly available. These are not “commercial” names and not intended to function as trademarks.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.