Trace Hub Debugger

# Trace Hub Debugger

#### **Document Revision 1.00**

#### **Sep 15, 2017**



Confidential, NDA Required

Copyright ©2016

American Megatrends, Inc.

5555 Oakbrook Parkway

Suite 200

Norcross, GA 30093 (USA)

All Rights Reserved

Property of American Megatrends, Inc

# Legal

Disclaimer

This publication contains proprietary information which is protected by copyright. No part of this publication may be reproduced, transcribed, stored in a retrieval system, translated into any language or computer language, or transmitted in any form whatsoever without the prior written consent of the publisher, American Megatrends, Inc. American Megatrends, Inc. retains the right to update, change, modify this publication at any time, without notice.

For Additional Information

Call American Megatrends, Inc. at 1-800-828-9264 for additional information.

Limitations of Liability

In no event shall American Megatrends be held liable for any loss, expenses, or damages of any kind whatsoever, whether direct, indirect, incidental, or consequential, arising from the design or use of this product or the support materials provided with the product.

Limited Warranty

No warranties are made, either expressed or implied, with regard to the contents of this work, its merchantability, or fitness for a particular use. American Megatrends assumes no responsibility for errors and omissions or for the uses made of the material contained herein or reader decisions based on such use.

Trademark and Copyright Acknowledgments

Copyright ©2017 American Megatrends, Inc. All Rights Reserved.

American Megatrends, Inc.

5555 Oakbrook Parkway

Suite 200

Norcross, GA 30093 (USA)

All product names used in this publication are for identification purposes only and are trademarks of their respective companies.

# Table of Contents

[Document Information 4](#_Toc493271125)

[Purpose 4](#_Toc493271126)

[Audience 4](#_Toc493271127)

[Overview 5](#_Toc493271128)

[Hardware 5](#_Toc493271129)

[Supported Platforms 6](#_Toc493271130)

[Installation 7](#_Toc493271131)

[The Intel® CCA Hardware LED Indicators 8](#_Toc493271132)

[Using Trace Hub In Visual eBios 10](#_Toc493271133)

[Setup Instructions: 11](#_Toc493271134)

[Selecting Platform Configuration 11](#_Toc493271135)

[Perform Basic DCI Functional Test 13](#_Toc493271136)

[Troubleshooting 15](#_Toc493271137)

# Document Information

## Purpose

This document provides information about Trace hub feature integrated with Visual eBios (VeB).

## Audience

The intended audiences are BIOS developers, Generic Chipset Porting Engineers, OEM Porting Engineers, and AMI OEM Customers.

# Overview

Intel platforms From Skylake and later generation intel cpu and platform supports hardware debugging and Trace hub infrastructure via USB 3 port.

Debug functionality controlled by CPU and Trace functionality controlled by PCH.

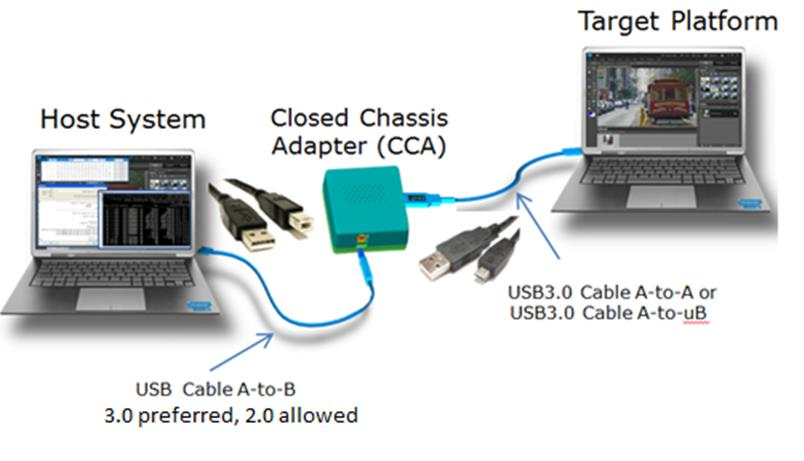
Need Hardware device called INTEL SVT Closed chassis adopter to be connected USB 3 Ports and target. Link: https://designintools.intel.com/product\_p/itpxdpsvt.htm

Future platform from cannon lake will include this CCA device in the platform itself and just USB 3 debug cable is enough to perform this operation.

## Hardware

DCI supports two hardware configurations (depending on the Intel® platforms):

* Intel® DCI-OOB (DCI over an Intel® SVT Closed Chassis Adapter [Intel® CCA])
* Intel® DCI-USB (DCI over a USB3.0 Debug Cable)

Figure 1-1. Hardware Configuration for Intel ® DCI-OOB with Intel® CCA

The Intel® CCA requires USB3 A-to-B cable to connect the host to the Intel® CCA, and a 6” A-to-A, A-to-uAB or A-to-C USB3 cable to connect the Intel® CCA to the target.

Figure 1-2. Hardware Configuration for USB3 Hosting DCI with USB3.0 Cable

USB-hosted DCI requires a USB3 Debug Cable (where the VBUS wires are isolated) to connect the debug host to the target. Using standard A to A USB3 cables can keep the target from transitioning through all power states correctly

## Supported Platforms

The following are the supported platforms where Trace hub can be used

|  |  |  |
| --- | --- | --- |
| **Platform** | **NDA** | **Library** |
| SkyLake CPU + SkyLake PCH |  | Yes |
| Kaby lake CPU + Kabylake PCH |  | Yes |
| Kaby lake CPU + SkyLake PCH |  | Yes |
| Apollo Lake |  | Yes |
| Skylake Server + Lewisburg PCH |  | Yes |
| Skylake Server + Kabylake PCH |  | Yes |
| Basin Falls |  | Yes |
| Purley | Yes |  |

# Installation

For the initial version of trace hub, install Intel® System Studio 2017 NDA version from [here](https://registrationcenter.intel.com/en/forms/?productid=2336&SupportCode=ENA&pass=yes).

**Note**: Downloading of NDA version requires user details to be provided to Intel and approval from Intel for the same. Once approved Intel will send out the instruction details in the registered email.

Download VEB label Buildtools\_31 or later and use this VEB to download Debugger module, and open project.

Download Debugger module (AptioV/trunk/AptioV/Binary/Modules/Debugger Labeled Debugger\_39), reopen the project, AptioV debugger and trace hub debugger will be installed automatically.

Restart the VEB, you will able to see the “DCI Debug” and “Debug” menu. The “DCI Debug” is the trace hub debugger.

# The Intel® CCA Hardware LED Indicators

The Intel® CCA provides two LED indicators labelled as FIRMWARE and DCI CONNECT. The following table shows the Intel® CCA status depending on the LED color that displays.

|  |  |  |
| --- | --- | --- |
| **Firmware LED** | **DCI Connect LED** | **Intel CCA Status** |
| OFF | Off | Intel® CCA driver not found or failed to load correctly, or cable not connected to port. |
| Flashing Green | Off | The FPGA firmware of Intel® CCA is updating. This happens when a new Intel® DAL version is installed and run for the first time, if the new Intel® DAL also contains a new Intel® CCA FPGA update. The FW update may take 3–5 minutes to complete. |
| Off | Red | Driver loaded and USB host f/w uploaded. Ports’ SSXTx is in e-Idle (USB Phy not responding—may be powered off or disabled). |
| Green | Red | Firmware moved from above state to the BSSB idle state (PCH exited eIdle on SSTx and Connection Status in BSSB = “Connected”). |
| Green | Flashing Red | BSSB idle state (above state) and Intel® DAL s/w is attempting to connect with s/w driven connection patterns. |
| Green | Alternating Red / Orange | Same as above, but firmware is attempting to connect with the connection pattern (without s/w intervention). |
| Green | Orange | From the above two states, Intel® CCA then detects ExI status packet. |
| Green | Green | Target connection established—s/w sets the “Do Enable” (read IO enable) bit in the Intel® CCA. The Intel® CCA is ready for debugging! |
| Green | Flashing green | Target connection established. Tx traffic in progress. |

# Using Trace Hub In Visual eBios

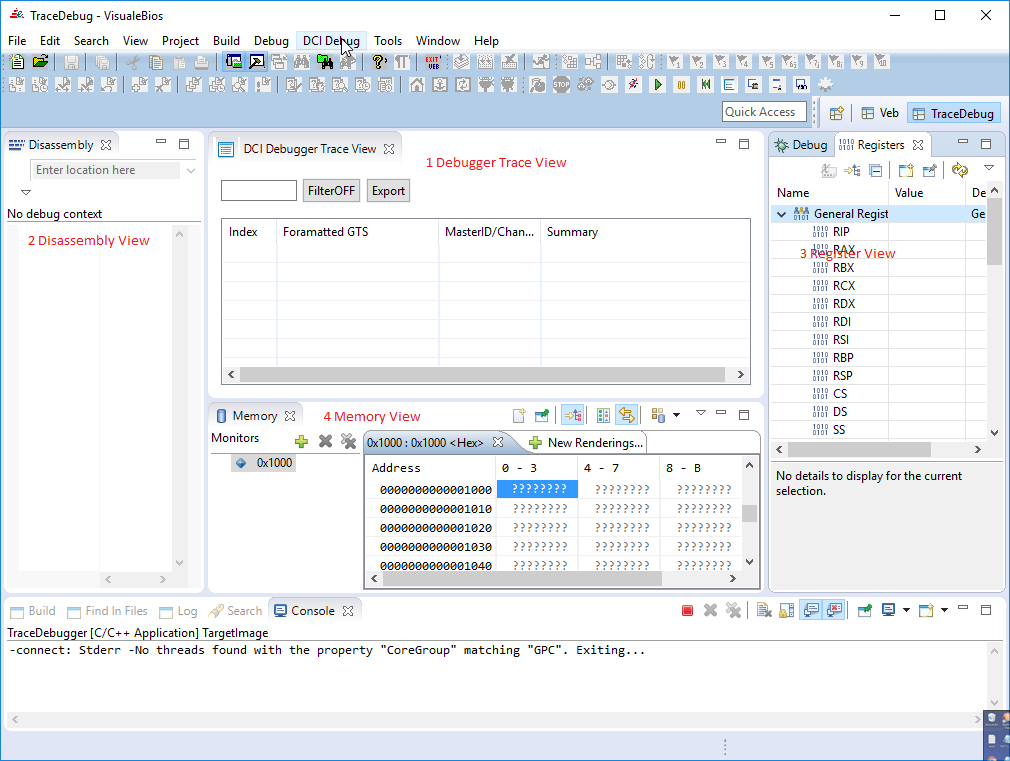
Users can debug a target using the Intel Trace hub device from within Visual eBios(VeB). The user s presented with a new perspective to view debug information. The following image depicts the Trace hub perspective a user will see.

Figure 1: : Trace Hub Debugger Views

Currently Trace hub debugger support Connect to the target, disconnect from the target, halt the target, continue, reset the target, read register, and read memory. They can be accessed from the menu “DCI Debug”.

## Setup Instructions:

For any AptioV project that supports Trace hub, to enable trace hub support, build the project with token TRACE\_HUB\_STATUS\_CODE\_SUPPORT enabled. This SDL token can be found in the generic module “IntelClientCommonPkg”.

**Note**: the token may be cloned in another module and turned off.

Flash the generated firmware image and boot the target, and enter into the BIOS setup and enable the following setup questions.

|  |  |
| --- | --- |
| **Setup Question** | **Value** |
| **PCH-IO Configuration**->**Trace Hub Configuration Menu**->**Trace Hub Enable Mode** | ***Host Debugger*** |
| **PCH-IO Configuration**->**DCI Enable (HDCIEN)** | **Enabled** |
| **CPU Configuration**->**Debug Interface** | **Enabled** |
| **CPU Configuration**->**Direct Connect Interface** | **Enabled** |

Then connect the target and host by using the INTEL SVT Closed chassis adopter, for the target side, need to use the USB 3 debug port. For the host side, USB2 or USB3 connection are fine.

## Selecting Platform Configuration

Before connecting, need to select the correct configuration file, to connect to the target. Use menu “**DCI Debug**-> **Configuration**”

You will see the following dialog.

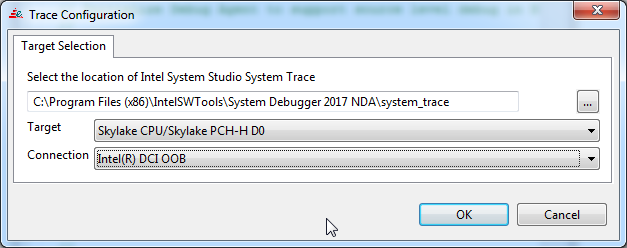


Figure : Trace hub Configuration

* Select the location of Intel System Studio System Trace application, usually the location is “C:\Program Files (x86)\IntelSWTools\System Debugger 2017\system\_trace”
* Select the target information from the list. For example, 6th Generation Intel Platform I/O (Skylake PCH-LP)
* Select the connection “**Intel(R) DCI OOB**”. When you use the INTEL SVT Closed chassis adopter, please always select this.
* If target is running, use menu “**DCI Debug**-> **Connect**”. The INTEL SVT Closed chassis adopter should be starting to connect to the target, after it connect the INTEL SVT Closed chassis adopter target light should be green. If there are trace message comes, you should be able to see it in the “Trace Message view”.
* Use menu DCI Debug-> halt. The target should be halt. You can read memory, read register, and see the disassemble code.

Currently Trace hub debugger has limited support for the debug function and full functionality will be added in future.

# Perform Basic DCI Functional Test

Once the DCI hardware setup is ready, software is installed, and the Platform configuration is selected (through Visual eBios), perform the following steps to establish basic access and verify basic functionality through DCI.

Make sure the AMITraceHubClient.exe and AMITraceHubServer.exe are exist in the VisualeBios folder.

* Using the VEB make sure that to select the correct platform and connection method
* Open command prompt window and change directory to Visual eBios location
* Run command “**AMITraceHubClient -connect**” to connect to the platform
* Wait for the platform to complete the DCI connectivity sequence
* If needed refer to the section “The Intel® CCA Hardware LED Indicators” for details
* After the software finishes the initialization sequence, use the following basic commands (in the order listed) to test out the basic function of DCI:

|  |  |
| --- | --- |
| **Command** | **Function** |
| AMITraceHubClient -halt | The CPU enters probe mode and saves states. |
| AMITraceHubClient -printreg <Register> | Print the value of register mentioned |
| AMITraceHubClient -readmem <address> <length> | Read <length> byte memory from <address> |
| AMITraceHubClient -run | The CPU exits probe mode and restores states. |
| AMITraceHubClient -disconnect | Ends the current session and terminate the server |

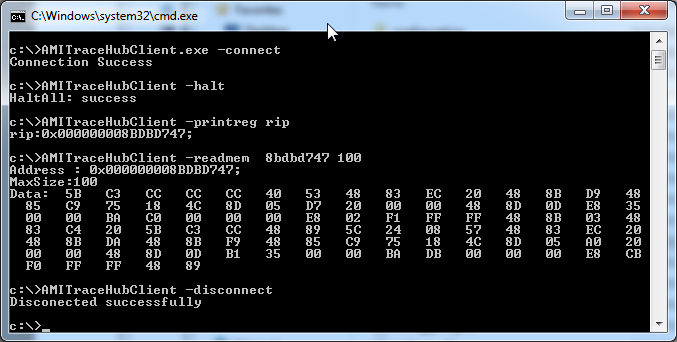
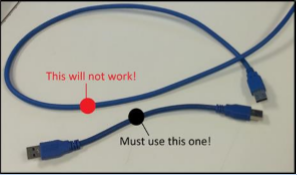


Figure Example of a console after the DCI is connected

# Troubleshooting

1. **Issue**: The Intel® CCA must be connected to the target through the appropriate 6” Intel® CCA cable.

**Solution**: Refer to section 2.4.1. for the to select the appropriate 6” cable to connect between the Type-A target USB connector of the Intel® CCA (labeled as Target) and the target’s USB3 capable Type-A, Type-uAB, or Type-C connector.

1. **Issue:** Intel® DAL connects to Intel® In-Target Probe (Intel®ITP) instead of DCI when both Intel® In-Target Probe (Intel® ITP) and DCI are present.

**Solution**: Don’t have Intel® In-Target Probe (Intel® ITP) and DCI connected to the debug host at the same time.

1. Issue: VeB Trace view stops receiving Trace messages

**Solution**: Stop Tracehub debugger by disconnecting and connect again