

Release Notes for AMD NDA Users

AMD Hardware Debug Tool

Version: *HDT 6.00.823*

Release Date: *October 29, 2004*

General Notes:

This software is released to AMD NDA users only. HDT 6.0 targets an AMD Athlon™ XP, AMD Sempron™, AMD Opteron™, or AMD Athlon 64 processor-based system. AMD Possum or Macraigor Raven is required for communicating with target PC. Before using HDT, both target PC and Possum or Raven must be connected properly and powered on as well. When you use the Raven to debug a multi-processor AMD Opteron™ or AMD Athlon™ 64 target, either an HDT MP Adaptor or a Percussion card is required. Possum supports multi-processor debugging. You must disable the Sprocket if you connect a Possum to Percussion.

Important Notes:

- In order to make HDT work properly on an AMD Opteron™ or AMD Athlon™ 64 MP target, you must keep your Sprocket firmware current. HDT supports automatic update of Possum firmware. Please contact debug.tools@amd.com for the latest firmware information.
- HDT 6.0 supports both Raven/Sprocket and Possum as target – host connector. When Possum is connected, USB 2.0 will be used as host connection port. If your host PC only supports USB 1.1, a USB 2.0

plug-in card is strongly suggested to use. ***This build only supports USB 1.1 connection at very low transition speed.*** You should avoid using USB 1.1 for any setup configuration.

- Before flashing Possum firmware, power on your Possum and then wait for the operating system to enumerate the device. Once the device is ready for use, update the firmware. After updating the firmware, power off the Possum to make the new firmware effective. After this is complete, to use the HDT with the Possum, power on the Possum and again wait for the operating system to enumerate it. When the device is ready for use, launch HDT.
- **In order to use HDT 6.x with Possum on a Melody target, you must update your Melody configuration. Only Melody target with Percussion 7 board is compatible with Possum.**
- If using a Possum to connect your Melody target with Percussion 7 card, you need to short **J7** jumper on Percussion 7. Connect your Possum **HDT** header (**J1**) to **J3** of Percussion 7 (incorrectly labeled **SCAN HEADER**), and connect Possum **MP_HDT** header (**J7**) to **J10** of Percussion 7 (labeled **FUGUE HEADER**).
- If using a Raven to connect your Melody target with Percussion 7 card, you need to open **J7** jumper on Percussion 7. Connect your Raven **HDT** header to **J8** of Percussion 7 (labeled **HAMMER MP HDT HEADER**).

What's New in this Build:

1. HDT 6.0 supports AMD Opteron™, AMD Athlon™ 64 Rev E microprocessors on target, and dual core AMD Opteron processors.
2. Communication with target can be accomplished using a Possum device or Raven/Sprocket devices. Raven/Sprocket is the default communication

mechanism, but can be overridden by plugging a Possum device into a USB 2.0 port on the host computer.

3. Unlike HDT 4 and 5, HDT 6.0 mainframe GUI has been designed so that it no longer uses workspaces for processors.
4. Software breakpoints feature has been added to the Breakpoint feature.
5. HDT 6.0 is able to determine which processor goes into HDT first and what causes it go into HDT. This information is displayed in the CPU status window when requested.
6. In a multiprocessor system, when a processor goes into HDT for any reason, all remaining processors will be brought into HDT.
7. “Events Trigger” has been changed to “Events Analyzer” and its user interface has been redesigned to be similar to the Tektronix Logic Analyzer. It is a platform-based feature and supports multiprocessors.
8. “Cache Diagnostic Loader” is the diagnostic tool available in HDT6. The previous format of the ICache Download Diagnostics is incompatible with this feature.

Host Computer Requirements:

The AMD Hardware Debug Tool runs on host machine with the following minimum system requirements:

Operating Systems:

- Windows® 2000 with SP4.
- Windows XP Professional with SP1.

Important: Before upgrading an operating system, uninstall the AMD Hardware Debug Tool using *Add/Remove Program* on the Windows Control Panel.

Installation Requirements:

Do not install HDT software to network drives. Do not install HDT software from a network drive either.

HDT software supports multi-user installs in some supported operating systems. Windows 2000 and Windows XP support multi-user installs. However, local administrative rights are always required to install and uninstall the application.

When HDT is installed as a multi-user install, the administrator and every user of the machine where HDT is installed can use the application. When HDT is installed under a user's login, instead of an administrator's login, the following will occur:

1. Only the user associated with that login will be able to use HDT.
2. Administrators and other users will not be able to reinstall or use HDT.

HDT does not support the switching user function in Windows XP. In all operating systems that support multi-user functionality, the current user must log out, and the next user must log in to switch users.

HDT does not support multiple installations on one host computer for different users. Only one HDT installation is allowed on one host machine.

Host Machine Requirements

Microprocessor:

AMD Athlon XP processor or later running at 900 MHz or faster.

RAM Memory:

At least 256 MB RAM mandatory-required.

Hard Disk Space:

200 MB free space for installation, additional space is used during operations for features that collect large amounts of data.

Ports:

- If using a Macreigor Raven, Enhanced Parallel Port (EPP) protocol (278/378) is supported. No other device can be used on the parallel port when using the AMD Hardware Debug Tool. Ensure that EPP is set in the BIOS setup menu of the host machine.
- If using a USB HDT Multi-processor Adapter, at least a USB 1.1 port is required. USB 2.0 is recommended.

Graphics/Monitor:

With a minimum resolution of 1024 x 768 true color. A 17-inch monitor is recommended.

Target Machine Requirements:**Target Processor:**

- AMD Athlon XP
- AMD Sempron
- AMD Opteron
- AMD Athlon 64

Motherboard:

TAP connector provided on PGA boards or Percussion board.

An HDT Multiprocessor Adapter or Percussion board is required to debug an AMD Opteron™ or AMD Athlon™ 64 processor-based MP system.

User's Knowledge Requirements

Mastery of the x86 architecture is needed before the AMD Hardware Debug Tool can be used effectively.

Limitations:

1. You have to re-launch HDT 6.0 software if you switch the target, change target/host connection, or add more processors on the board.
2. When you install HDT 6.0 on Windows XP, you may get the Microsoft warning message. Click “Continue Anyway” to ignore the message.



3. Disassembly from the Reset Vector does not work unless PHYS_TYPE field is set to *000h* instead of the default *240h*.
4. Using the disassembly window to set HW and SW breakpoints does not apply to Real Mode or 16-bit Legacy Mode.
5. HDT only works when the built-in LPT1: port is set at EPP mode. HDT is unable to detect whether the host parallel port is in EPP mode. Users must set it correctly using the BIOS setup menu.
6. HDT does not support any plug-in parallel port.
7. When the target stops at a SW breakpoint, then you resume target program, HDT 6.0 will Single-Step first then resume the target program. If you do not want the Single-Step before resume, you must disable or remove this SW breakpoint.

8. User will be unable to save any customized MSR, which is created using the Single MSR feature, to the feature data file. If HDT is closed, all customized MSR's will be lost. Ask for HDT support from debug.tools@amd.com if user needs to add MSR's permanently.
9. Customizable user settings will not persist when performing an upgrade from one version of HDT to a newer version.

Known Bugs and Workaround:

1. **Problem:** When using "Auto PHYS_TYPE" function in Memory Dump, Disassembly and Target Memory Search features, the MTRRFIXDRAMMODEN bit of the SYSCFG MSR has been set prior to reading MTRR registers from target. If the target is in a protected or long mode, this operating may clean up the Cache Tag of target.
Workaround: Manually define PHYS_TYPE value to protect Cache Tag arrays from zeroing.
2. **Problem:** The layout file does not contain all feature windows format and size information. Some migration may be needed after retrieving the layout file.
Workaround: None
3. **Problem:** If you install HDT 6 on an operating system with MDAC 2.5 or earlier, you may get the unhandled exception error when launching HDT 6 – "System.InvalidOperationException: The .Net Data OLE DB Provider (System.Data.OleDb) requires Microsoft Data Access Components (MDAC) version 2.6 or later".
Workaround: Rebooting the host system may resolve the problem. If not, you need to install MDAC 2.8 from this site:

<http://www.microsoft.com/downloads/details.aspx?FamilyID=6c050fe3-c795-4b7d-b037-185d0506396c&DisplayLang=en>
4. **Problem:** When using step over on a branch with three hardware breakpoints already used, the following message will pop up "No more HW breakpoints available. Remove a HW breakpoint and try again".
Workaround: Before stepping over a branch instruction, make sure at least two hardware breakpoints are available for use. If above message occurs, close current Disassembly window and open another one; doing this will update the breakpoints correctly.
5. **Problem:** When using BTHB Code Flow to analyze trace entries, if the Start Address and Stop Address range is not equal or larger than 3 one-line entries that are 36 bytes, exception will occur. For example, if you select BTHB base, i.e. 0xA0000, as Start Address, you have to have at least a number equal or greater than 0xA0024 as Stop Address.

Workaround: Click “Continue” on exception window, close the current BTHB Code Flow window, and open another BTHB Code Flow to continue.

6. **Problem:** HT Trace Data Viewer filtering for "All Commands" in "Filter by Command" option doesn't work. No data appear when this filter is selected.

Workaround: This filter is not needed, since it would show every packet available. If this filter is accidentally removed, press Remove Filter button to restore data.

7. **Problem:** IO APIC Registers do not allow paste between fields on the data grids in the columns DEST MODE, INT PIN POL, REM IRR, TRIG MODE, and INT MAST

Workaround: Manually enter data values.

8. **Problem:** If a data grid cell is currently focused on and has been changed, it might not properly refresh.

Workaround: To make sure desired cell is refreshed, first focus on another cell by clicking it and then refresh values in the feature.

9. **Problem:** ICache Line Translator printing format may be too hard for user to read. The data is spread out between the sheets based on the longest line and causes several of the printouts to print blank empty boxes with no data. It's hard to tell which matches which.

Workaround: None

10. **Problem:** Copy and pasting values larger than the destination size will not work correctly and may cause an exception in some features.

Workaround: Manually enter data values for values with sizes larger than destination size.

11. **Problem:** Performance Monitor setup window background color does not change to the current “Background Invalid Color” when All Nodes/All Cores is selected. By default, this color is yellow.

Workaround: None

12. **Problem:** Entire contents of tables and grids cannot be copied. Only a single cell can be copied at a time.

Workaround: None

13. **Problem:** ICache Disassembly does not properly handle reader mode (when a file from a different target configuration is loaded into the feature). The feature throws an exception.

Workaround: None

14. **Problem:** In HTTrace Control's Clause Definition form, changing the command type of a trigger will reset the address. This can be bothersome

because the address needs to be input each time.

Workaround: None

15. Problem: Saving a large HTTrace Data Viewer dump file in the binary or memory dump buffer only formats can sometimes throw an out of memory exception if system memory is low.

Workaround: This is dependent on how much system memory is available at the time of saving the file. Monitor the memory usage and only attempt to save large buffers to these formats if current memory usage is relatively low.

16. Problem: In HTTrace Control's Clause Definition form, changing the match item for one trigger type will change the match item for the other. For instance, if Trigger 0's match values are changed, Trigger 1's will be as well. These are both accessing different bits in the Trace Control register, so this shouldn't happen.

Workaround: Use HTTrace Register feature to initially set up these match values. These correspond to the cLDTE_n0/ncLDTE_n0 bits for trigger 0 and cLDTE_n1/ncLDTE_n1 bits for Trigger 1, where cLDT is for coherent and ncLDT is for NonCoherent. After setting these up, refresh the Trace Control feature to utilize these new values.

17. Problem: Saving a large text file in the HTTrace Data Viewer feature will take a very long time.

Workaround: None.

18. Problem: In HTTrace Data Viewer, if the Address filter is applied to the data with addresses smaller than 40 bits, the feature will throw an exception.

Workaround: None

19. Problem: HDT attempt to update Possum Firmware can fail. This appears to be related to how quickly Windows can "recognize" that Possum is ready to communicate. Even though the Possum enumerates, if the AMD FX2 device has not shown up in the Device Manager, the firmware download will fail.

Workaround: Wait until the AMD FX2 device is displayed in the Device Manager. It is not sufficient to wait until the "New hardware is ready to use" message, since some systems don't display it.

20. Problem: HDT6 reboots host system on exit. When the File->Exit option is chosen from the main menu, the host system can reboot.

Workaround: None.

21. Problem: Disassembly does not properly load binary files.

Workaround: None

22. Problem: Resizing of column widths in Disassembly window do not stay

after adjustment. After single stepping or refreshing data, the columns return to the default width.

Workaround: None.

23. Problem: HDT does not disassemble SKINIT instruction.

Workaround: None

24. Problem: Cannot write values from loaded saved memory dump bin file back to the original target the file was created from.

Workaround: None

25. Problem: The register features allow users to click Write Changes button after applying data with the Write All with all Nodes / Cores is selected. Since there aren't any changes the Write Changes button should not be active.

Workaround: None

26. Problem: The Write Only Register Fields in the MSR list shows data after user clicks window Grid. They can open the bitmap which will show ALL F's even if nothing has ever been written to the window.

Workaround: None

27. Problem: Files from other targets loaded into BTHB Data Viewer load incorrectly and refresh causes empty error box to appear. The file never loads in reader view which would indicate that the data doesn't match the target system anyway.

Workaround: None

28. Problem: The BKDG shows the register MSR C001_100D was removed on Rev E or later. It is still present in HDT6.

Workaround: None.

29. Problem: The Ctrl-F command does not appear to work if a feature is floating; it will only work if it is docked. Other global commands may also be experiencing this problem if the feature is floating, but some appear to be working, like Print.

Workaround: Dock the feature to take advantage of the global commands.

30. Problem: On an MP system, Program Registers may display the fields from the Subfields grid multiple times.

Workaround: Only one set of these fields will be able to access the target, so use these fields only. The other fields should not be able to read or write any data to the target system.

31. Problem: Extremely large memory usage after dumping several Memory Dump and HTTrace buffers, which makes system performance sluggish.

Workaround: Large memory dumps will improve if the user has more than the minimum recommended amount of memory and speed on the Host system. Also, try to avoid keeping multiple features with large dumps open at a single time; this will free memory on the Host system.

Known Issues in Help Document

1. **Extra Information for ICache Disassembly: The ICache Disassembly disassembles data from the ICache and/or L2 Cache arrays.** The feature also has the ability to track the CS:rIP if the physical address representing the data being disassembled matches the CS:rIP value. Users can perform a single step action via the context menu, and the CS:rIP will be tracked provided it matches the physical address of the data displayed.
2. **Description for Context Menu Item "List Header On Load" in ICache Diagnostic Loader.** This menu option lets the user see all the header settings that are in each diagnostic file as they are loaded into the system. This is used primarily to verify that the diagnostic header is set up properly.
3. **Description for Adding an Else-If/Then clause to a state:** To add an Else-If/Then clause into a state, select the clause right click and select "Add If-Then Clause" from the pop-up menu. The new clause will appear below the selected clause.
4. **Description for "Step Into", "Step Over", "Jump rIP to Next Row", and "Run to Cursor" menu items on the pop-up menu of Disassembly feature:**

Step Into: This command causes the program to advance one instruction in memory. Functions as Single Step.

Step Over: This command sets a hardware breakpoint on the next instruction, runs the program to that breakpoint as if Exit rIP was performed, then stops and removes the breakpoint. If the current CS:rIP instruction is a conditional branch instruction, an extra hardware breakpoint is set and removed on the instruction of the branch .

Jump rIP to Next Row: This command moves the rIP to the address of the next instruction displayed in the row following the current CS:rIP instruction. This command provides a mean to exit an infinite loop.

Run to Cursor: This command sets a hardware breakpoint on the instruction of the cursor, runs the program to that breakpoint as if Exit rIP was performed, then stops and removes the breakpoint. The instruction referenced by the cursor may not be reachable from the current

instruction. The polling window is displayed when the program runs and the breakpoint has not been hit.

Those menu items are unavailable when "Auto Track CS:rIP" is **not** selected on the Disassembly Dialog.

Problem Report

If you find a problem, we ask that you please email the following information to debug.tools@amd.com

- Describe the problem or issue.
- Describe why you believe this is a problem.
- Describe your proposed solution, change, or improvement.
- Briefly describe the steps or sequence of events that led to your observation.
- How frequently did the problem occur?
- What messages did HDT display?
- What version HDT are you using? This information is in the window titlebar of the About AMD HDT 6.0 Dialog, which can be opened by selecting Help → About ...
- What operating system and service pack are you using?
- How much free disk space do you have?
- Describe your system in regards to the following:
 - o Processor type and speed
 - o Motherboard make and model
 - o System memory size
 - o Video graphics card make and model
 - o Video screen resolution (for example, 1024x768x16 bpp)
 - o Other installed Input/Output cards (for example sound cards, other PCI cards)

Enhancement Requests

Please email the following information about a desired enhancement or change to debug.tools@amd.com

- Which version of HDT 6.0 are you using? This information is in the window title bar of the About AMD HDT 6.0 Dialog, which can be opened by selecting Help → About AMD HDT 6.0.
- Describe the desired enhancement or change.
- Indicate to us how important this is to you using a scale of 1 to 5 where 1 is most important and 5 least important.