HIP SDK for Windows

Release 7.2

Advanced Micro Devices, Inc.

Nov 11, 2026

INSTALL

1 System requirements for Windows 3
- 1.1 Supported OSes . . . 3
- 1.2 Windows-supported GPUs and APUs . . . 3
- 1.3 CPU support . . . 5

2 Install HIP SDK 7
- 2.1 HIP SDK quick start installation . . . 7
- 2.2 HIP SDK command line installation . . . 12
- 2.3 Setup the Windows environment . . . 12
- 2.4 Upgrade HIP SDK . . . 13
- 2.5 Uninstall HIP SDK . . . 13

3 ROCm components supported in HIP SDK 17

4 Application deployment guidelines for HIP SDK 19
- 4.1 Library distribution . . . 19

5 AMD ROCm Debugger for Windows 21
- 5.1 Installation . . . 21
- 5.2 Limitations . . . 22
- 5.3 Hello World example . . . 22

6 AMD HIP SDK for Windows 7.2 — Release Notes 25
- 6.1 What's new . . . 25
- 6.2 Fixed issues . . . 25
- 6.3 Limitations . . . 25
- 6.4 Known issues . . . 26

7 ROCm release versioning 27
- 7.1 Windows documentation implications . . . 27
- 7.2 Windows builds from source . . . 28

HIP SDK for Windows provides a subset of the ROCm platform to Microsoft Windows as described in ROCm component support in HIP SDK. It provides the runtime, APIs, and tooling to leverage the computational power of AMD GPUs to create high-performance, portable applications using the HIP programming language.

The HIP SDK consists of the AMD GPU Driver, HIP runtime, and HIP Libraries. These three parts are distributed in the HIP SDK installer. The HIP SDK is intended for developer distribution. This is in contrast to the AMD GPU driver, which is intended for all end users.

HIP SDK can run on your Windows system with Microsoft Visual Studio Code, and even includes solutions for use with the tool. However, it also has standalone tools like compilers, profilers, and debuggers for use in your own development environment.

HIP SDK code is open and hosted at https://github.com/ROCm/rocm-install-on-windows.

The documentation is structured as follows:

Installation

- System requirements for HIP SDK
- Install HIP SDK for Windows

Conceptual

- ROCm component support in HIP SDK
- Deployment guidelines

How to

- Use ROCm Debugger for Windows

About

- HIP SDK release notes
- HIP SDK release versions

CHAPTER

ONE

SYSTEM REQUIREMENTS FOR WINDOWS

1.1 Supported OSes

AMD HIP SDK supports the following Microsoft Windows variants.

Distribution	Processor architectures	Validated update
Windows 11	x86-64	22H2 (GA)

1.2 Windows-supported GPUs and APUs

The tables below show supported GPUs/APUs for AMD Radeon™ PRO, AMD Radeon™, and AMD Ryzen™. If a GPU is not listed on this table, it is not officially supported by AMD.

AMD Radeon PRO

Name	Architecture	LLVM target	Runtime	HIP SDK	AMD ROCm Debugger
AMD Radeon AI PRO R9700	RDNA4	gfx1201			
AMD Radeon PRO W7900 Dual Slot	RDNA3	gfx1100			
AMD Radeon PRO W7900	RDNA3	gfx1100			
AMD Radeon PRO W7800	RDNA3	gfx1100			
AMD Radeon PRO W7700	RDNA3	gfx1101			
AMD Radeon PRO W6800	RDNA2	gfx1030			
AMD Radeon PRO W6600	RDNA2	gfx1032			
AMD Radeon PRO W5500	RDNA1	gfx1012			
AMD Radeon PRO VII	GCN5.1	gfx906			

AMD Radeon

Name	Architecture	LLVM target	Runtime	HIP SDK	AMD ROCm Debugger
AMD Radeon RX 9070 XT	RDNA4	gfx1201			
AMD Radeon RX 9070	RDNA4	gfx1201			
AMD RX 9070 GRE	RDNA4	gfx1201			
AMD Radeon RX 9060 XT	RDNA4	gfx1200			
AMD Radeon RX 9060	RDNA4	gfx1200			
AMD Radeon RX 7900 XTX	RDNA3	gfx1100			
AMD Radeon RX 7900 XT	RDNA3	gfx1100			
AMD Radeon RX 7800 XT	RDNA3	gfx1101			
AMD Radeon RX 7700 XT	RDNA3	gfx1101			
AMD Radeon RX 7650 GRE	RDNA3	gfx1102			
AMD Radeon RX 7600 XT	RDNA3	gfx1102			
AMD Radeon RX 7600	RDNA3	gfx1102			
AMD Radeon RX 6950 XT	RDNA2	gfx1030			
AMD Radeon RX 6900 XT	RDNA2	gfx1030			
AMD Radeon RX 6800 XT	RDNA2	gfx1030			
AMD Radeon RX 6800	RDNA2	gfx1030			
AMD Radeon RX 6750 XT	RDNA2	gfx1031			
AMD Radeon RX 6700 XT	RDNA2	gfx1031			
AMD Radeon RX 6700	RDNA2	gfx1031			
AMD Radeon RX 6650 XT	RDNA2	gfx1032			
AMD Radeon RX 6600 XT	RDNA2	gfx1032			
AMD Radeon RX 6600	RDNA2	gfx1032			

AMD APU

Name	Architecture	LLVM target	Runtime	HIP SDK
AMD Ryzen AI Max+ Pro 395	RDNA3.5	gfx1151		
AMD Ryzen AI Max+ 395	RDNA3.5	gfx1151		
AMD Ryzen AI Max Pro 390	RDNA3.5	gfx1151		
AMD Ryzen AI Max 390	RDNA3.5	gfx1151		
AMD Ryzen AI Max Pro 385	RDNA3.5	gfx1151		
AMD Ryzen AI Max 385	RDNA3.5	gfx1151		
AMD Ryzen AI Max Pro 380	RDNA3.5	gfx1151		
AMD Ryzen AI 300 Series	RDNA3	gfx1150		
AMD Ryzen AI 9 400 Series	RDNA3	gfx1150		
AMD Ryzen AI MAX+ 495	RDNA3.5	gfx1151		
AMD Ryzen AI MAX+ 492	RDNA3.5	gfx1151		
AMD Ryzen AII MAX PRO 490	RDNA3.5	gfx1151		

: Supported - Official software distributions of the current HIP SDK release fully support this hardware.

: Deprecated - The current HIP SDK release has limited support for this hardware. Existing features and capabilities are maintained, but no new features or optimizations will be added. A future release will remove support.

: Unsupported - The current HIP SDK release does not support this hardware. Prebuilt HIP SDK libraries are not officially supported and might cause runtime errors.

1.3 CPU support

ROCm requires CPUs that support PCIe™ atomics. Modern CPUs after the release of 1st generation AMD Zen CPU and Intel™ Haswell support PCIe atomics.

CHAPTER

TWO

INSTALL HIP SDK

To install the HIP SDK on Windows, use the HIP SDK quick start installation or the instructions in HIP SDK command line installation. See System requirements for Windows for more information about the required environment.

2.1 HIP SDK quick start installation

For a quick summary on installing the HIP SDK on Windows, follow the steps listed on this page.

1. Download the installer.

   Download the installer from the HIP SDK download page.

   The download page lists supported OSes for each available ROCm version, and a link to download the corresponding installer. Select the download file matching the ROCm version you want to install.

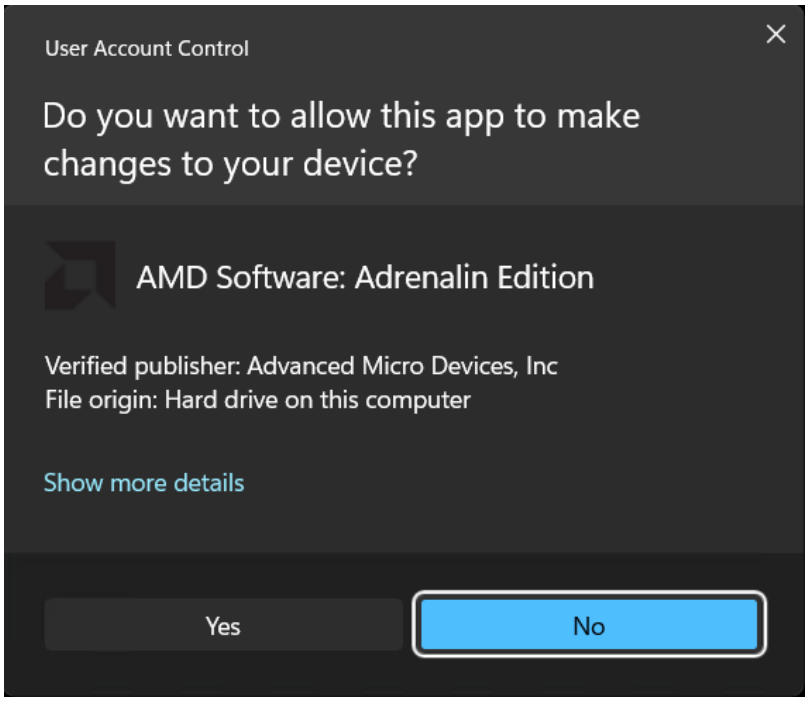
   Clicking the HIP SDK download link takes you to a license page that you must accept before the download will begin. Specify the location to save the downloaded file.

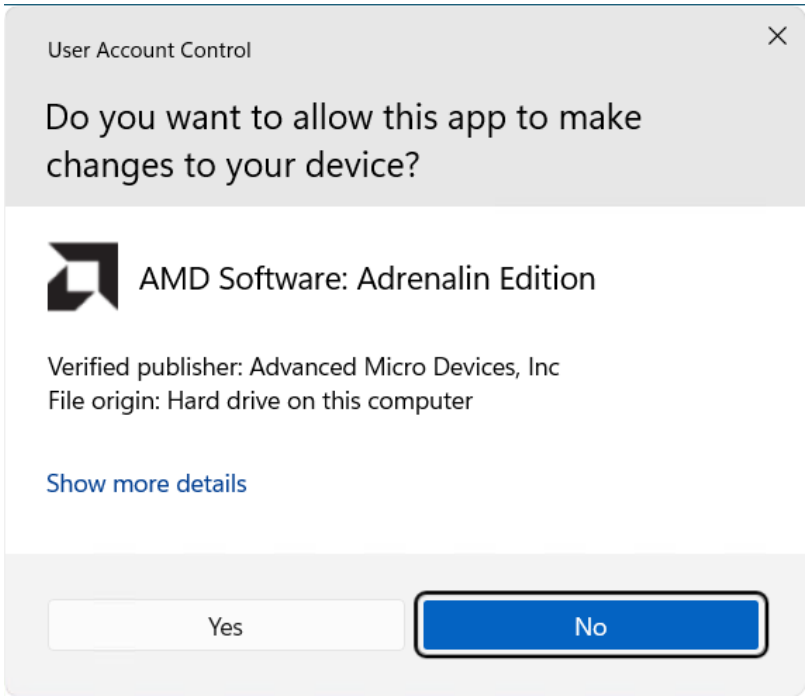
2. Launch the installer.

   To launch the AMD HIP SDK Installer, click the Setup icon shown in the following image.

   The installer requires Windows Administrator privileges, so the User Access Control (UAC) dialog box might display. Click Yes.

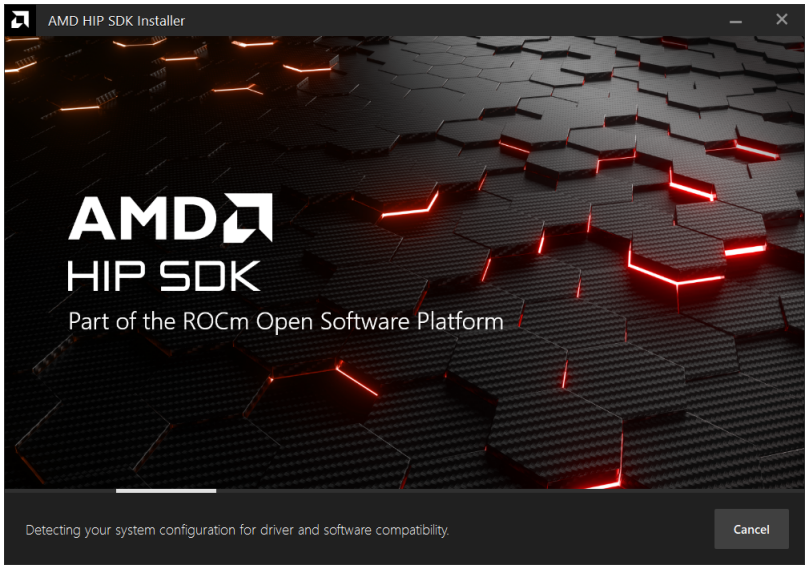




The installer executable will temporarily extract installer packages to C:\AMD, which it removes after completing the installation. You'll see the "Initializing install" window during extraction.

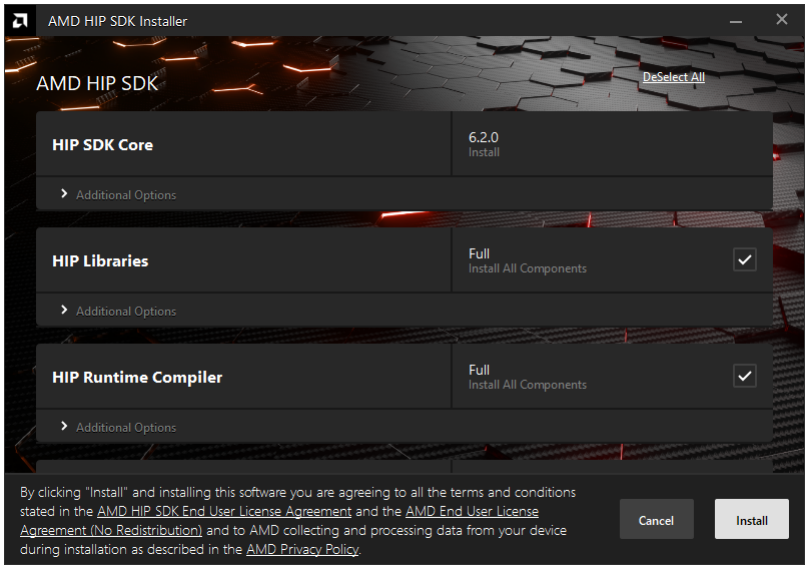


The installer will then detect your system configuration to determine which installable components are applicable.



3. Customize the install.

   When the installer launches, it displays a window that lets you customize the installation. By default, all components are selected for installation.



   a. HIP SDK installer

      The HIP SDK installation options are listed in the following table.

HIP components	Install type	Additional options
HIP SDK Core	7.2	Install location
HIP Libraries	Full, Partial, None	Runtime, Development (Libs and headers)
HIP Runtime Compiler	Full, Partial, None	Runtime, Development (headers)
HIP Ray Tracing	Full, Partial, None	Runtime, Development (headers)
Visual Studio Plugin	Full, Partial, None	Visual Studio 2017, 2019, 2022 Plugin
AMD ROCm Debugger	Full, Partial, None	AMD ROCm Debugger (ROCgdb)

Note: The select/deselect all options only apply to the installation of HIP SDK components. To install the bundled AMD Display Driver, manually select the install type.

Tip: Should you only wish to install a few select components, deselecting all, then selecting individual components may be more convenient.

b. AMD display driver

The HIP SDK installer bundles an AMD Radeon Software PRO 25.30 installer. The supported install options and types are summarized in the following tables:

Install option	Description
Install Location	Location on disk to store driver files.
Install Type	The breadth of components to be installed.
Factory Reset (optional)	A Factory Reset will remove all prior versions of AMD HIP SDK and drivers. You will not be able to roll back to previously installed drivers.

Install type	Description
Full Install	Provides all AMD Software features and controls for gaming, recording, streaming, and tweaking the performance on your graphics hardware.
Minimal Install	Provides only the basic controls for AMD Software features and does not include advanced features such as performance tweaking or recording and capturing content.
Driver Only	Provides no user interface for AMD Software features.

Note: You must perform a system restart for a complete installation of the Display driver.

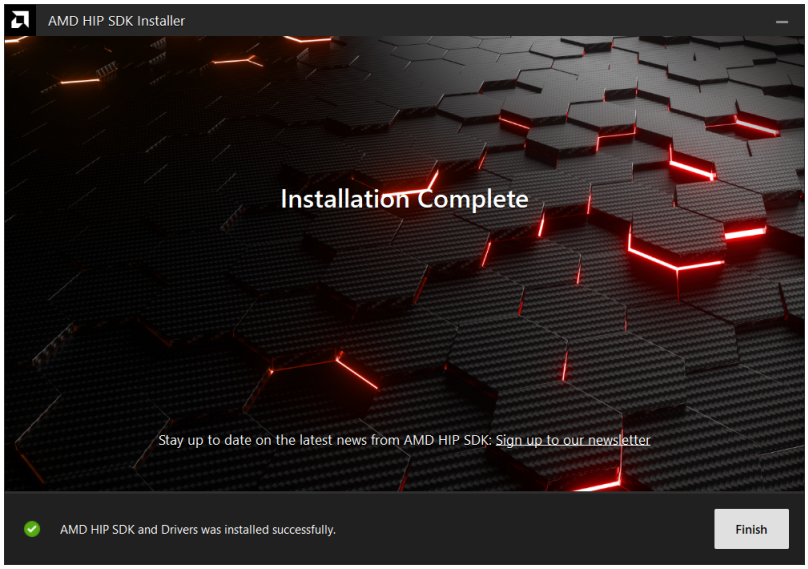
4. Install components.

Wait for the installation to complete, as shown in the following image.



5. Complete installation.

   After the installation is complete, the installer window might prompt you for a system restart. Click Finish or Restart in the lower-right corner, as shown in the following image.



Note: If the installer terminates mid-installation, you can safely remove the temporary directory created under C:\AMD. Installed components don't depend on this folder unless you explicitly chose this as the install folder.

2.2 HIP SDK command line installation

The following information provides instructions for installing from the Windows command line. To start the installation, follow these steps:

1. Download the installer from the HIP-SDK download page.

2. Launch the installer. Note that the installer is a graphical application with a WinMain entry point, even when called on the command line. This means that the application lifetime is tied to a window, even on headless systems where that window may not be visible.

```
Start-Process $InstallerExecutable -ArgumentList $InstallerArgs -NoNewWindow -Wait
```

Important: Running the installer requires Administrator privileges.

Command line options are listed in the following table:

Install option	Description
-install	Command used to install packages, both driver and applications. No output to the screen.
-install -boot	Silent install with auto reboot.
-install -log <absolute path>	Write install result code to the specified log file. The specified log file must be on a local machine. Double quotes are needed if there are spaces in the log file path.
-uninstall	Command to uninstall all packages installed by this installer on the system. There is no option to specify which packages to uninstall.
-uninstall -boot	Silent uninstall with auto reboot.
/? or /help	Shows a brief description of all switch commands.

Note: Unlike the GUI, the CLI doesn't support selectively installing parts of the SDK bundle.

To install all components:

```
Start-Process ~\Downloads\Setup.exe -ArgumentList '-install','-log',"${env:USERPROFILE}\installer_log.txt" -
↪NoNewWindow -Wait
```

2.3 Setup the Windows environment

To run HIP SDK on the Windows environment, you must use the following steps:

1. Add HIP installation to the $PATH of the to the System variables using the System Properties -> Environment Variables -> Path -> Add command:

```
C:\Program Files\AMD\ROCm\7.1\bin
```

You can also set it for a new terminal using the following command:

```
$env:PATH += ;C:\\Program Files\\AMD\\ROCm\\7.1\\bin
echo $PATH
```

2. Use hipconfig or hipInfo to test the installation and make sure the commands are loaded:

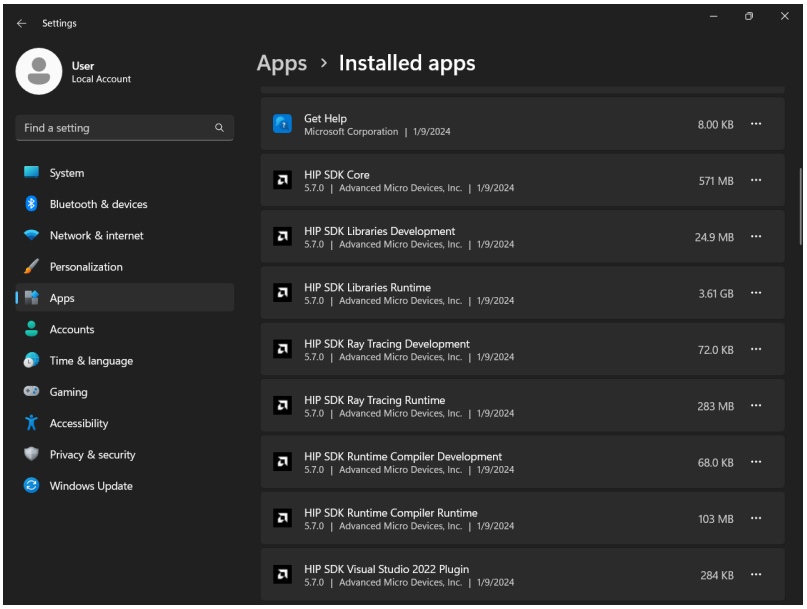
```
hipInfo
hipconfig
```

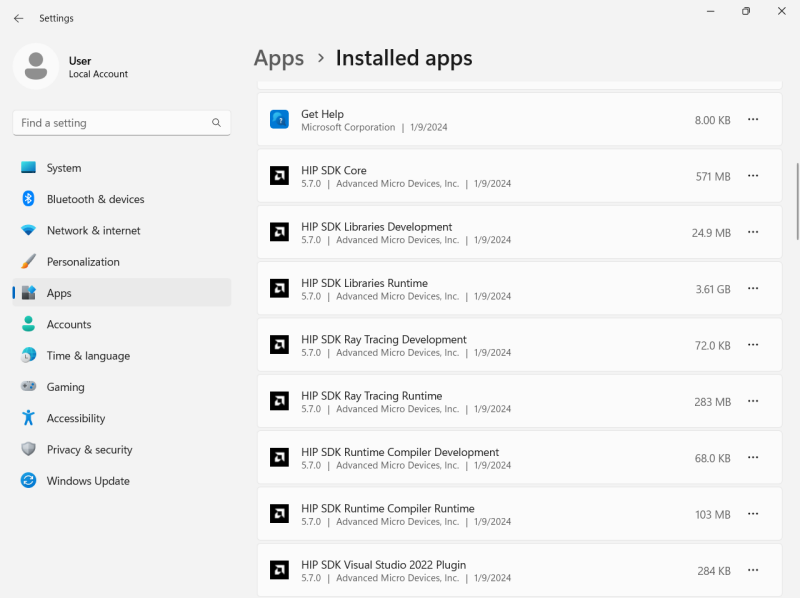
2.4 Upgrade HIP SDK

To upgrade the HIP SDK, you can run the installer for the newer version without uninstalling the existing version. You can also uninstall the HIP SDK before installing the newest version.

2.5 Uninstall HIP SDK

All components, except the Visual Studio plug-in, should be uninstalled through Control Panel > Add/Remove Program. You can uninstall HIP SDK components through the Windows Settings app. Navigate to "Apps > Installed apps", click the ellipsis (…) on the far right next to the component you want to uninstall, then click "Uninstall".





CLI

Launch the installer. Note that the installer is a graphical application with a WinMain entry point, even when called on the command line. This means that the application lifetime is tied to a window, even on headless systems where that window may not be visible.

```
Start-Process $InstallerExecutable -ArgumentList $InstallerArgs -NoNewWindow -Wait
```

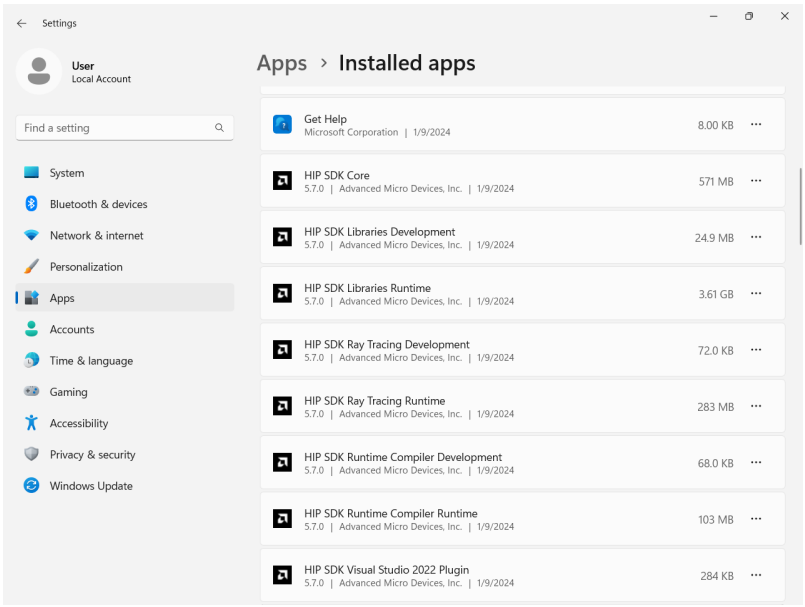
Important: Running the installer requires Administrator privileges.

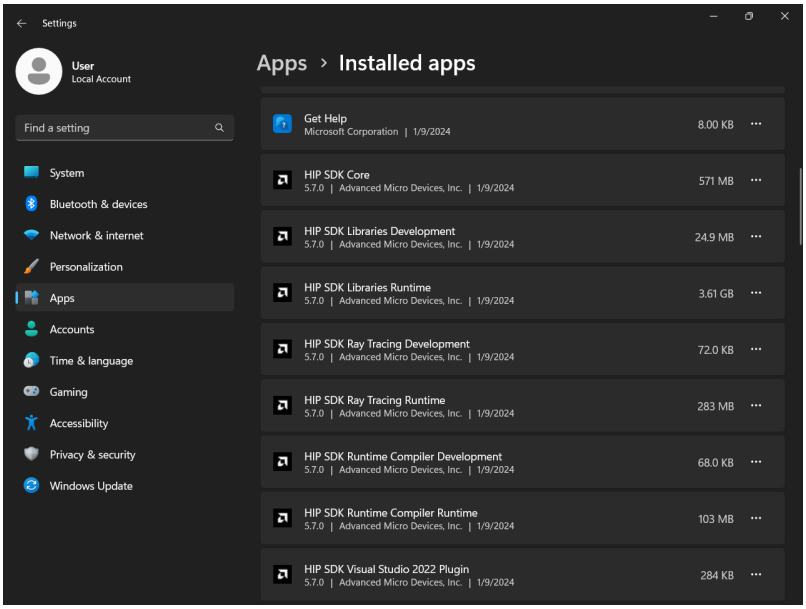
To uninstall all components, use the following code:

```
Start-Process ~\Downloads\Setup.exe -ArgumentList '-uninstall' -NoNewWindow -Wait
```

GUI

Uninstallation of HIP SDK components can be done through the Windows Settings app. Navigate to “Apps > Installed apps” and click the ellipsis (…) on the far right next to the component you want to uninstall. Click “Uninstall”.


Settings
User
Local Account
Find a setting
System
Bluetooth & devices
Network & internet
Personalization
Apps
Accounts
Time & language
Gaming
Accessibility
Privacy & security
Windows Update
Apps › Installed apps
Get Help
Microsoft Corporation | 1/9/2024
8.00 KB
HIP SDK Core
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
571 MB
HIP SDK Libraries Development
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
24.9 MB
HIP SDK Libraries Runtime
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
3.61 GB
HIP SDK Ray Tracing Development
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
72.0 KB
HIP SDK Ray Tracing Runtime
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
283 MB
HIP SDK Runtime Compiler Development
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
68.0 KB
HIP SDK Runtime Compiler Runtime
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
103 MB
HIP SDK Visual Studio 2022 Plugin
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
284 KB


Settings
User
Local Account
Find a setting
System
Bluetooth & devices
Network & internet
Personalization
Apps
Accounts
Time & language
Gaming
Accessibility
Privacy & security
Windows Update
Apps › Installed apps
Get Help
Microsoft Corporation | 1/9/2024
8.00 KB
HIP SDK Core
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
571 MB
HIP SDK Libraries Development
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
24.9 MB
HIP SDK Libraries Runtime
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
3.61 GB
HIP SDK Ray Tracing Development
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
72.0 KB
HIP SDK Ray Tracing Runtime
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
283 MB
HIP SDK Runtime Compiler Development
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
68.0 KB
HIP SDK Runtime Compiler Runtime
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
103 MB
HIP SDK Visual Studio 2022 Plugin
5.7.0 | Advanced Micro Devices, Inc. | 1/9/2024
284 KB

CHAPTER

THREE

ROCM COMPONENTS SUPPORTED IN HIP SDK

The HIP SDK brings a subset of ROCm to developers on Windows. The collection of features enabled on Windows is referred to as the HIP SDK. These features allow developers to use the HIP runtime, HIP math libraries and HIP Primitive libraries. The following table shows the differences between Windows and Linux releases.

Note: HIPBlasLT is now available for Windows users in HIP SDK version 6.4.2 and later, supported on gfx1101.

Component	Linux	Windows
Driver	AMD GPU driver	AMD GPU driver
Compiler	hipcc/amdclang++	hipcc/clang++
Debugger	rocgdb	ROCm Debugger for Windows
Profiler	ROCProfiler	Radeon GPU Profiler
Porting Tools	HIPIFY	HIPIFY
Runtime	HIP (open sourced)	HIP (closed source)
Math Libraries	Supported	Supported
Primitives Libraries	Supported	Supported
Communication Libraries	Supported	Not available
AI Libraries	MIOpen, MIGraphX	Not available
System Management	ROCm SMI, RDC, rocminfo	hipInfo
AI Frameworks	PyTorch, TensorFlow, etc.	Not available
CMake HIP Language	Enabled	Unsupported
Visual Studio	Not applicable	Plugin available
HIP Ray Tracing	Supported	Supported

CHAPTER

FOUR

APPLICATION DEPLOYMENT GUIDELINES FOR HIP SDK

Independent Software Vendors (ISVs) deploying applications using the HIP SDK depend on the AMD GPU drivers, HIP runtime library and HIP SDK libraries. AMD's support policies for developer tools allow Independent Software Vendors (ISVs) the stability to plan the usage of a tool chain–AMD provides backward compatibility for applications dynamically linked to the HIP runtime.

AMD GPU drivers are distributed with the HIP runtime included. Each HIP runtime is associated with a HIP compiler version. Applications built with a particular HIP compiler should document its associated HIP runtime version and AMD GPU driver as minimum version requirements.

Note: Applications do not distribute the HIP runtime. Instead, end users use the HIP runtime provided by an AMD GPU Driver.

We recommend that ISV applications using the HIP SDK libraries distribute the HIP SDK Library as part of its installer package. This avoids requiring your end users to install the HIP SDK.

4.1 Library distribution

The HIP SDK is distributed via a Windows installer. This distribution system is only intended for software developers and testers. There are two types of ISV applications that use the HIP SDK.

- ISV applications with a dependency on the HIP runtime and select HIP header-only libraries (rocPRIM, hipCUB, and rocThrust).

  These ISV applications require end users to install an AMD GPU driver, which has a HIP runtime library bundled with it. Developers should ensure that these ISV applications have a minimum HIP runtime library version associated with them. Because the HIP runtime library doesn't have semantic versioning, the ISV application can't check for compatibility.

  Tip: AMD works to avoid breaking API/ABI compatibility unless the major version number of the HIP runtime is incremented.

  ISV applications can run without a warning if the driver's HIP major version is the same as that associated with its build compiler. You can set your ISV applications to throw a warning if the HIP major version is higher than that associated with its build compiler.

- ISV applications with a dependency on the HIP runtime and one or more dynamically linked HIP libraries (including the HIP runtime library).

  These ISV applications should distribute the dynamically linked HIP library in their installer package. This avoids having your end users manually install the HIP SDK. One benefit of this model is that less

disk space is need, as only the required binaries are distributed in the ISV application. This method also avoids having your end user required to agree to the licensing agreements for the entire HIP SDK.

With this approach, the version checks recommended for the ISV application (including dynamically linked HIP libraries) follows the same requirements as the previous category.

Each dynamically linked HIP library also has a minimum HIP runtime requirement. You can add minimum HIP version checks for each dynamically linked HIP library. The minimum HIP version check for the HIP runtime is usually sufficient if dynamically linked HIP libraries are from the same SDK package as the HIP compiler.

Note: AMD does not support static linking to any components distributed in the HIP SDK.

CHAPTER

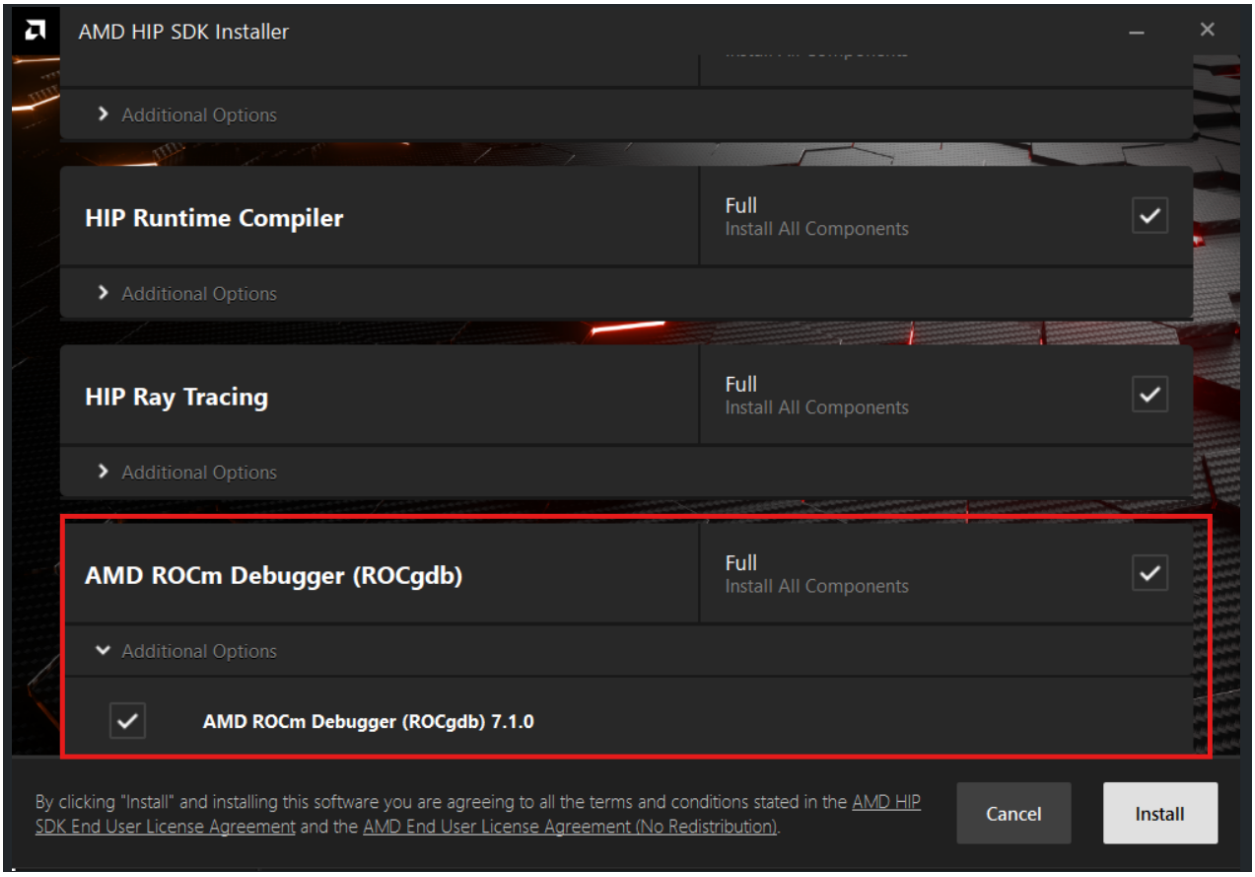
FIVE

AMD ROCM DEBUGGER FOR WINDOWS

The AMD ROCm Debugger for Windows is an AMD source-level debugger based on the GNU Debugger (GDB). Windows ROCgdb is included in the HIP SDK installer from version 7.2, and enables heterogeneous debugging on the ROCm software that consists of an x86-based host architecture along with supported AMD GPUs. For Windows ROCgdb to functional properly the driver included with the HIP SDK package must be used.

5.1 Installation

Windows ROCgdb can be installed by checking the "AMD ROCm Debugger (ROCgdb)" option (see below) during installation of the HIP SDK package.



5.2 Limitations

ROCgdb on Windows has the following limitations currently:

- Windows is supported only on the gfx120x architectures (gfx1200 and gfx1201).
- Multi-GPU configurations involving more than one AMD GPU are not supported.
- Generating or loading AMD GPU core dumps is not supported on Windows.
- Python scripting is not supported.
- Due to an AMD HIP runtime limitation, it is not possible to pass intercepted signals (SIGFPE, SIGSEGV, etc.) to the inferior, even if the user requests otherwise. Signals are always suppressed by the runtime.
- The HIPCC compiler on Windows defaults to producing PDB files containing CodeView debug information for the host code. ROCgdb does not support this debug format. To enable host code debugging, pass the -gdwarf -Wl,-debug:dwarf options to HIPCC to generate DWARF debug information instead, which ROCgdb supports.
- The HIPCC compiler on Windows emits host code that targets the Microsoft x64 ABI conventions and MSVC C++ ABI. ROCgdb does not fully support these, which may lead to incorrect symbol names, misprinted C++ objects, and similar issues when debugging host code. These limitations do not affect debugging AMD GPU device-side code.

5.3 Hello World example

Debugging GPU code with ROCgdb on Windows is just like on Linux, as you'll see with this example.

1. Copy the following code to a file named example.cpp in a directory of your choice (for example, C:\rocgdb-example). The code contains a simple GPU kernel that adds two integers and returns the result. The result is printed on the console.

```
#include <hip/hip_runtime.h>
#include <assert.h>

__global__ void do_an_addition(int a, int b, int *out)
{
  *out = a + b;
}

int main ()
{
  int *result_ptr, result;
  /* Allocate memory for the device to write the result to. */
  hipError_t error = hipMalloc(&result_ptr, sizeof(int));
  assert(error == hipSuccess);

  /* Run `do_an_addition` on one block containing one HIP thread. */
  do_an_addition<<<dim3(1), dim3(1), 0, 0>>>(1, 2, result_ptr);

  /* Copy result from device to host. This acts as a
     synchronization point, waiting for the kernel dispatch to
     complete. */
  error = hipMemcpyDtoH(&result, result_ptr, sizeof(int));
  assert(error == hipSuccess);
```

(continues on next page)

(continued from previous page)

```
  printf("result is %d\n", result);
  assert(result == 3);
  return 0;
}
```

2. Open a Windows Command Prompt. We will use this prompt to run the compiler and the debugger.
3. Put the HIP SDK bin directory in the PATH:

```
set "HIP_SDK_DIR=C:\Program Files\AMD\ROCm\7.1"
set "PATH=%HIP_SDK_DIR%\bin;%PATH%"
```

4. This makes the hipcc and rocgdb programs available in the PATH.
5. Switch the current directory to where you put the example.cpp source file earlier:

```
cd C:\rocgdb-example
```

6. Compile the example.cpp HIP program using hipcc:

```
C:\rocgdb-example>hipcc example.cpp -o example.exe --offload-arch=gfx1201 -gdwarf -Wl,-debug:dwarf -O0
```

   Notes:

   - The --offload-arch=gfx1201 option targets AMD Radeon RX 9070 series products. For AMD Radeon RX 9060 series products, use --offload-arch=gfx1200 instead.
   - The -gdwarf -Wl,-debug:dwarf options instruct hipcc to emit DWARF debug information for the host code. Otherwise, hipcc emits PDB (Microsoft) debug information, which ROCgdb does not yet understand.
   - The -O0 option disables compiler optimizations.

7. Run the just-compiled program to confirm it is working:

```
C:\rocgdb-example>.\example.exe
result is 3
```

8. You can now run the just-compiled program under ROCgdb, stopping execution in the do_an_addition GPU kernel function, like so:

```
C:\rocgdb-example>rocgdb example.exe

GNU gdb (ROCm) 18.0.50.20251029-git
Copyright (C) 2025 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-w64-mingw32".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
    <https://github.com/ROCm-Developer-Tools/ROCgdb/issues>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
```

(continues on next page)

(continued from previous page)

```
Reading symbols from example.exe...
(gdb) break do_an_addition
Function "do_an_addition" not defined.
Make breakpoint pending on future shared library load? (y or [n]) y
Breakpoint 1 (do_an_addition) pending.
(gdb) run
Starting program: C:\rocgdb-example\example.exe
[New Thread 12180.0x286c]
[New Thread 12180.0x1aa4]
[New Thread 12180.0x4e8c]
[New Thread 12180.0x3578]
[New Thread 12180.0x3af4]
[Switching to thread 7, lane 0 (AMDGPU Lane 1:2:1:1/0 (0,0,0)[0,0,0])]
Thread 7 "do_an_addition" hit Breakpoint 1, with lane 0, do_an_addition (a=1, b=2,␣
↪out=0x300004000) at example.cpp:6
6         *out = a + b;
(gdb)
```

You can find more information about ROCgdb in the copy of the ROCgdb manual that is installed with the HIP SDK. The manual is located under the share\html subdirectory, for example, for an SDK installed at C:\Program Files\AMD\ROCm\7.1, the manual is located under C:\Program Files\AMD\ROCm\7.1\share\html.

CHAPTER

SIX

AMD HIP SDK FOR WINDOWS 7.2 — RELEASE NOTES

This page documents the release history, new features, improvements, limitations, and known issues for the AMD HIP SDK on Windows. It is organized for developers, QA, and partner teams who need a reliable, reference-style view of what changed across releases.

6.1 What's new

1. Support for ROCm 7.2
2. Support for AMD Ryzen™ AI 300 Series, AMD Ryzen™ AI 400 Series, and AMD Ryzen™ AI Max 400 Series and AMD Ryzen™ AI Max+ 400 Series.

6.2 Fixed issues

- Various performance improvements to ROCm Debugger (ROCgdb) on Windows
- Improved stability with various math libraries.

6.3 Limitations

The following items reflect the current state of HIP SDK for Windows and the ROCm Debugger (ROCgdb) as documented in internal notes and public guidance. Where applicable, workarounds are provided.

ROCm Debugger (ROCgdb) on Windows Limitations

- Architecture support: Windows is currently supported on the AMD Radeon™ RX 9000 series graphics products. Other architectures are not supported with AMD ROCm Debugger for Windows.
- Multi-GPU: Configurations with more than one AMD GPU are not supported with AMD ROCm Debugger for Windows.
- Core dumps: Generating or loading AMD GPU core dumps is not supported on Windows.
- Python scripting: Not supported.
- Signals: Due to a HIP runtime limitation, intercepted signals (SIGFPE, SIGSEGV, etc.) cannot currently be passed through to the inferior on Windows; signals are always suppressed by the runtime.
- Host debug info: HIPCC on Windows defaults to PDB/CodeView for host debug info, which ROCgdb does not support. Workaround: compile with -gdwarf -Wl,-debug:dwarf to emit DWARF host debug info supported by ROCgdb.

- ABI compatibility: HIPCC emits host code targeting Microsoft x64 ABI and MSVC C++ ABI. ROCgdb does not fully support these, which may cause incorrect symbol names or misprinted C++ objects during host debugging. Device-side GPU debugging is not affected..

6.4 Known issues

- Intermittent soft hang may be observed in rocPRIM rocprim.device_adjacent_find.
- rocSPARSE may experience hipMemcpy errors & test abort in some cases.
- Applications using the HIP 6 runtime (amdhip64_6.dll) show ~20% lower performance on Strix Halo with Adrenalin 26.10-branch drivers (26.6.2, 26.6.4). Applications on the HIP 7 runtime (amdhip64_7.dll) are unaffected.

Quick links

CHAPTER

SEVEN

ROCM RELEASE VERSIONING

Windows will follow Linux version numbers, as Windows ROCm releases are based on Linux ROCm releases. However, not all Linux ROCm releases will have a corresponding Windows ROCm release. The following table shows all ROCm releases. Releases with both Windows and Linux are referred to as a joint release. Releases with only Linux support are referred to as a skipped release (from a Windows perspective).

ROCm version	Linux support	Windows support
5.5		
5.6		
5.7		
6.0		
6.1		
6.2		
6.3		
6.4		
7.1		
7.2		

In general, Windows releases trail Linux releases. If you want to support both Linux and Windows using a single ROCm version, refrain from upgrading ROCm until there is a joint release.

7.1 Windows documentation implications

The ROCm documentation website contains both Windows and Linux documentation. Just below each article title, a convenient article information section states whether the page applies to Linux only, Windows only, or both. To find the exact Windows documentation for a HIP SDK release, refer to the ROCm documentation with the same Major.Minor version number while ignoring the patch version (which is only relevant for Linux releases). For convenience, we will continue to include Windows documentation in release documentation for skipped Windows releases.

Windows release notes contain only information pertinent to Windows.

7.2 Windows builds from source

Not all source code required to build Windows from source is available under a permissive open source license. We only provide Windows build instructions for projects that can be built from source on Windows using a toolchain that has closed source build prerequisites. The ROCm manifest file is not valid for Windows. AMD does not release a manifest or tag our components in Windows. You can use corresponding Linux tags to build on Windows.