Radeon™ GPU Analyzer 2.4.1 – Release Notes

# Highlights

* Introducing support for DirectX® Ray Tracing (DXR) in a new mode of the command line tool (-s dxr):
  + Compile DXR shaders to generate RDNA2™ ISA disassembly, hardware resource usage statistics, live register analysis and control flow graphs.
  + The new mode does not require a ray-tracing-enabled GPU, any AMD GPU with the latest drivers should work.
* Improved performance in the GUI app to better handle shaders with large disassembly.
* Updated the static analysis engine with bug fixes and support for the latest RDNA2™ instructions.
* Removed Legacy OpenCL™ mode (-s cl). OpenCL support for RDNA™ would be added in the future in a new, offline, mode of the tool.

# Notable Bug Fixes

**GUI application**:

* gfx1030 architecture is now presented as RDNA2™.
* [[GitHub-78](https://github.com/GPUOpen-Tools/radeon_gpu_analyzer/issues/78)]: crash/freeze with large disassembly.

# Known Issues

## Vulkan Live Driver mode (-s vulkan)

* Source to disassembly correlation is not supported by AMD’s shader compiler at the moment and is therefore not supported in the UI.
* Keyboard navigation is not supported from some views in the GUI application.
* The RGA layer is a beta feature. It fails to extract the shaders and pipeline state from certain Vulkan apps.
* Notifications about the fact that modified SPIR-V binary does not match the disassembly will not appear for loaded projects (in case that you changed the SPIR-V code, did not build and re-loaded the project).

## ROCM OpenCL (rocm-cl mode)

* RDNA™ targets are not supported as targets for ROCM-CL modes.
* The Lightning Compiler does not support disassembling of binaries for Graphics IP v7 targets.
* OpenCL C++ kernels are not yet supported by the Lightning Compiler.
* No support for live register analysis and control-flow graph generation in this mode.

## GUI Application

* “Correlation Disabled” notification in the source code editor is not being saved for projects after they were closed.
* Certain SALU instructions are being misclassified as VALU instructions.
* Cycle estimate for certain VALU instructions appears as “Varies” instead of 4.

## OpenGL (opengl mode)

Resource usage statistics for OpenGL mode only displays usage of SGPRs and VGPRs.

## DirectX12 (dx12 mode)

Live register analysis & CFG generation requires using the --isa option to generate ISA disassembly.

## DXR mode (-s dxr)

In Pipeline mode, pipeline binary extraction is not supported when more than a single pipeline is generated.

## Vulkan offline (vk-offline, vk-spv-offline, vk-spv-txt-offline modes)

SPIR-V support limitations:

1. The following capabilities, which are defined in the SPIR-V spec, are currently not supported:

CapabilityGroupNonUniformClustered,

CapabilityStoragePushConstant16,

CapabilityStoragePushConstant8,

CapabilityInputAttachmentArrayNonUniformIndexingEXT

CapabilityAddresses

CapabilityLinkage

CapabilityKernel

CapabilityVector16

CapabilityFloat16Buffer

CapabilityInt64Atomics

CapabilityImageBasic

CapabilityImageReadWrite

CapabilityImageMipmap

CapabilityPipes

CapabilityDeviceEnqueue

CapabilityLiteralSampler

CapabilityGenericPointer

CapabilityFragmentDensityEXT

CapabilityPhysicalStorageBufferAddressesEXT

CapabilitySampleMaskOverrideCoverageNV

CapabilityGeometryShaderPassthroughNV

CapabilityShaderViewportIndexLayerNV

CapabilityShaderViewportMaskNV

CapabilityShaderStereoViewNV

CapabilityPerViewAttributesNV

CapabilityMeshShadingNV

CapabilityImageFootprintNV

CapabilityFragmentBarycentricNV

CapabilityComputeDerivativeGroupQuadsNV

CapabilityShadingRateNV

CapabilityGroupNonUniformPartitionedNV

CapabilityRayTracingNV

CapabilityComputeDerivativeGroupLinearNV

CapabilitySubgroupShuffleINTEL

CapabilitySubgroupBufferBlockIOINTEL

CapabilitySubgroupImageBlockIOINTEL

1. SPIR-V modes currently only support the following extensions:

SPV\_KHR\_post\_depth\_coverage

SPV\_KHR\_shader\_ballot

SPV\_KHR\_subgroup\_vote

SPV\_KHR\_device\_group

SPV\_KHR\_multiview

SPV\_KHR\_shader\_draw\_parameters

SPV\_KHR\_16bit\_storage

SPV\_KHR\_storage\_buffer\_storage\_class

SPV\_KHR\_8bit\_storage

SPV\_EXT\_nonuniform\_qualifier

SPV\_EXT\_shader\_stencil\_export

SPV\_EXT\_shader\_viewport\_index\_layer

SPV\_AMD\_shader\_ballot

SPV\_AMD\_shader\_trinary\_minmax

SPV\_AMD\_shader\_explicit\_vertex\_parameter

SPV\_AMD\_gcn\_shader

SPV\_AMD\_gpu\_shader\_half\_float

SPV\_AMD\_gpu\_shader\_half\_float\_fetch

SPV\_AMD\_texture\_gather\_bias\_lod

SPV\_AMD\_gpu\_shader\_int16

SPV\_AMD\_shader\_fragment\_mask

SPV\_AMD\_shader\_image\_load\_store\_lod

SPV\_KHR\_variable\_pointers

SPV\_EXT\_nonuniform\_qualifier

SPV\_GOOGLE\_decorate\_string

SPV\_GOOGLE\_hlsl\_functionality1

SPV\_KHR\_vulkan\_memory\_model

# Notes for OpenCL Mode Users

The “-s rocm-cl” mode uses the Lightning Compiler package that ships with RGA, which is based on clang.

As of version 2.0, RGA allows developers to replace the Lightning Compiler package that ships with the product with a user-provided LLVM-based package. For more information, see the Radeon GPU Analyzer GUI app’s help manual, or run the command line tool with –s rocm-cl –h as arguments (look for the “Alternative ROCm OpenCL compiler” section).

# Notes for DirectX 11 (dx11) Mode Users

RGA’s DirectX 11 (-s dx11) mode will use the driver that is associated with the primary display adapter, by default. If your primary display adapter is not an AMD GPU, or if you would like RGA to use a driver that is associated with a different display adapter that is installed on your system, use the --adapters and --set-adapter <id> command line switches in order to instruct RGA to use the relevant driver.

# System Requirements

It is generally recommended to use RGA with the latest Radeon Software version. Specifically, to target the RDNA™ architecture, the latest Radeon Software version is required (except for all Vulkan® modes and the rocm-cl mode, which are independent of the driver).

## Vulkan (vulkan)

To use the installed driver in Vulkan mode:

1. Vulkan SDK 1.1.97.0 or later is required.
2. Latest Adrenalin or amdgpu-pro driver is required.

## Vulkan offline (vk-offline, vk-spv-offline, vk-spv-txt-offline)

All Vulkan offline modes (vk-offline, vk-spv-offline and vk-spv-txt-offline) are independent of the installed driver and graphics hardware and should work on any x86-based system.

## DirectX 12 (dx12) and DirectX 11 (dx11)

It is recommended to use the latest Adrenalin drivers for the best experience in DirectX 12 and DirectX 11 modes.

## ROCM OpenCL (rocm-cl mode)

ROCM OpenCL mode (rocm-cl) is independent of the installed driver and graphics hardware and should work on any x86-based system.

## OpenGL (-s opengl)

OpenGL mode on Linux requires the latest amdgpu-pro driver.