rocr_debug_agent

Release 2.0.4

Advanced Micro Devices, Inc.

Apr 11, 2025

INSTALL

The ROCR Debug Agent (ROCdebug-agent) is a library that can be loaded by the ROCm software runtime (ROCR) to provide the following functionalities:

- Print the state of all AMDGPU wavefronts that cause a queue error (such as, a memory violation, executing a `s_trap 2`, or executing an illegal instruction).
- Print the state of all AMDGPU wavefronts by sending a SIGQUIT signal to the process using `kill -s SIGQUIT <pid>` command or by pressing `Ctrl-\`, while the program is executing.

This functionality is provided for all AMDGPUs supported by the ROCm Debugger API Library (ROCdbgapi).

You can access ROCdebug-agent code on our GitHub repository.

The documentation is structured as follows:

Install

- *Installation*

Conceptual

- *ROCR Debug Agent user guide*

To contribute to the documentation, refer to Contributing to ROCm.

You can find licensing information on the Licensing page.

CHAPTER

ONE

INSTALLATION

This document provides information required to build and install ROCR Debug Agent (ROCdebug-agent) library.

1.1 Prerequisites

- A system supporting ROCm. See the supported operating systems.
- A C++ 17 compiler such as GCC 7 or Clang 5.
- The AMD ROCm software stack. See the ROCm installation instructions.
- Packages as per the OS.
  - For Ubuntu 18.04 and Ubuntu 20.04:

```
apt install gcc g++ make cmake libelf-dev libdw-dev
```

  - For CentOS 8.1 and RHEL 8.1:

```
yum install gcc gcc-c++ make cmake elfutils-libelf-devel elfutils-devel
```

  - For SLES 15 Service Pack 1:

```
zypper install gcc gcc-c++ make cmake libelf-devel libdw-devel
```

- Python 3.6 or later to run the tests.
- ROCdbgapi library. This can be installed using the ROCdbgapi package as part of the ROCm release. See the instructions to install ROCdbgapi library.

1.2 Build and install

An example command line to build and install the ROCdebug-agent library on Linux:

```
cd rocm-debug-agent
mkdir build && cd build
cmake -DCMAKE_BUILD_TYPE=Release -DCMAKE_INSTALL_PREFIX=../install ..
make
```

To specify the location for the installation, use `CMAKE_INSTALL_PREFIX`. The default location is `/usr`.

To specify a list of paths (separated by semicolons) that are used to locate the `cmake` modules, use `CMAKE_MODULE_PATH`. It is used to locate the HIP `cmake` modules required to build the tests. The default is location is `/opt/rocm/hip/cmake`.

The built ROCdebug-agent library is placed in `build/librocm-debug-agent.so.2*`.

To install the ROCdebug-agent library, use:

```
make install
```

The installed ROCdebug-agent library and tests are placed in:

- <install-prefix>/lib/librocm-debug-agent.so.2*
- <install-prefix>/share/rocm-debug-agent/LICENSE.txt
- <install-prefix>/share/rocm-debug-agent/README.md
- <install-prefix>/src/rocm-debug-agent-test/*

1.3 Test

To test the ROCdebug-agent library, use:

```
make test
```

Output:

```
Running tests...
Test project /rocm-debug-agent/build
Start 1: rocm-debug-agent-test
1/1 Test #1: rocm-debug-agent-test .............   Passed    1.59 sec

100% tests passed, 0 tests failed out of 1
Total Test time (real) =   1.59 sec
```

You can run the tests individually outside of the `CTest` harness as shown below:

```
HSA_TOOLS_LIB=librocm-debug-agent.so.2 HSA_ENABLE_DEBUG=1 test/rocm-debug-agent-test 0
HSA_TOOLS_LIB=librocm-debug-agent.so.2 HSA_ENABLE_DEBUG=1 test/rocm-debug-agent-test 1
HSA_TOOLS_LIB=librocm-debug-agent.so.2 HSA_ENABLE_DEBUG=1 test/rocm-debug-agent-test 2
```

CHAPTER

TWO

ROCR DEBUG AGENT USER GUIDE

To display the source text location with the machine code instructions around the wavefront's pc, compile the AMDGPU code objects with `-ggdb`. In addition, you can optionally use `-O0` to achieve a more intuitive display of the source text location, as higher optimization levels can help to reorder machine code instructions. When `-ggdb` is not used, the source line information is unavailable, and only machine code instructions starting at the wavefront's pc are printed.

```
/opt/rocm/bin/hipcc -O0 -ggdb -o my_program my_program.cpp
```

To use the ROCdebug-agent, set the `HSA_TOOLS_LIB` environment variable to the file name or path of the library and the `HSA_ENABLE_DEBUG` environment variable to `1`.

```
HSA_TOOLS_LIB=/opt/rocm/lib/librocm-debug-agent.so.2 HSA_ENABLE_DEBUG=1 ./my_program
```

If the application encounters a triggering event, ROCdebug-agent prints the state of some or all AMDGPU wavefronts.

See a sample printout:

```
Queue error (HSA_STATUS_ERROR_EXCEPTION: An HSAIL operation resulted in a hardware␣
↪exception.)

----------------------------------------------------------
wave_1: pc=0x7fd4f100d0e8 (stopped, reason: ASSERT_TRAP)

system registers:
            m0: 00000000        status: 00012461       trapsts: 20000000          mode:␣
↪000003c0
        ttmp4: 00000001        ttmp5: 00000000         ttmp6: f51a0080         ttmp7:␣
↪000000d5
        ttmp8: 00000000        ttmp9: 00000000        ttmp10: 00000000        ttmp11:␣
↪000000c0
        ttmp13: 00000000
        exec: 0000000000000001           vcc: 0000000000000000
    xnack_mask: 0000000000012460  flat_scratch: 00807fac01000000

scalar registers:
        s0: f520c000           s1: 00007fd5           s2: 00000000           s3:␣
↪00ea4fac
        s4: f51a0080           s5: 00007fd5           s6: f520c000           s7:␣
↪00007fd5
        s8: f1002000           s9: 00007fd4          s10: 00000000          s11:␣
↪00000000
       s12: f1000000          s13: 00007fd4          s14: f1001000          s15:␣
```

(continues on next page)

(continued from previous page)

```
↪00007fd4
      s16: f5186070          s17: 00007fd5          s18: f100e070          s19:␣
↪00007fd4
      s20: f5186070          s21: 00007fd5          s22: f100e070          s23:␣
↪00007fd4
      s24: 00004000          s25: 00010000

vector registers:
       v0: [0] 00000000 [1] f1002004 [2] f1002008 [3] f100200c [4] f1002010 [5]␣
↪f1002014 [6] f1002018 [7] f100201c [8] f1002020 [9] f1002024 [10] f1002028 [11]␣
↪f100202c [12] f1002030 [13] f1002034 [14] f1002038 [15] f100203c [16] f1002040 [17]␣
↪f1002044 [18] f1002048 [19] f100204c [20] f1002050 [21] f1002054 [22] f1002058 [23]␣
↪f100205c [24] f1002060 [25] f1002064 [26] f1002068 [27] f100206c [28] f1002070 [29]␣
↪f1002074 [30] f1002078 [31] f100207c [32] f1002080 [33] f1002084 [34] f1002088 [35]␣
↪f100208c [36] f1002090 [37] f1002094 [38] f1002098 [39] f100209c [40] f10020a0 [41]␣
↪f10020a4 [42] f10020a8 [43] f10020ac [44] f10020b0 [45] f10020b4 [46] f10020b8 [47]␣
↪f10020bc [48] f10020c0 [49] f10020c4 [50] f10020c8 [51] f10020cc [52] f10020d0 [53]␣
↪f10020d4 [54] f10020d8 [55] f10020dc [56] f10020e0 [57] f10020e4 [58] f10020e8 [59]␣
↪f10020ec [60] f10020f0 [61] f10020f4 [62] f10020f8 [63] f10020fc
       v1: [0] 00000000 [1] 00007fd4 [2] 00007fd4 [3] 00007fd4 [4] 00007fd4 [5]␣
↪00007fd4 [6] 00007fd4 [7] 00007fd4 [8] 00007fd4 [9] 00007fd4 [10] 00007fd4 [11]␣
↪00007fd4 [12] 00007fd4 [13] 00007fd4 [14] 00007fd4 [15] 00007fd4 [16] 00007fd4 [17]␣
↪00007fd4 [18] 00007fd4 [19] 00007fd4 [20] 00007fd4 [21] 00007fd4 [22] 00007fd4 [23]␣
↪00007fd4 [24] 00007fd4 [25] 00007fd4 [26] 00007fd4 [27] 00007fd4 [28] 00007fd4 [29]␣
↪00007fd4 [30] 00007fd4 [31] 00007fd4 [32] 00007fd4 [33] 00007fd4 [34] 00007fd4 [35]␣
↪00007fd4 [36] 00007fd4 [37] 00007fd4 [38] 00007fd4 [39] 00007fd4 [40] 00007fd4 [41]␣
↪00007fd4 [42] 00007fd4 [43] 00007fd4 [44] 00007fd4 [45] 00007fd4 [46] 00007fd4 [47]␣
↪00007fd4 [48] 00007fd4 [49] 00007fd4 [50] 00007fd4 [51] 00007fd4 [52] 00007fd4 [53]␣
↪00007fd4 [54] 00007fd4 [55] 00007fd4 [56] 00007fd4 [57] 00007fd4 [58] 00007fd4 [59]␣
↪00007fd4 [60] 00007fd4 [61] 00007fd4 [62] 00007fd4 [63] 00007fd4
       v2: [0] 22222222 [1] 11111125 [2] 1111111b [3] 11111123 [4] 1111111d [5]␣
↪1111111c [6] 1111111a [7] 1111111d [8] 1111111a [9] 1111111b [10] 1111111c [11]␣
↪11111118 [12] 11111123 [13] 1111111c [14] 11111119 [15] 11111117 [16] 1111111d [17]␣
↪11111114 [18] 1111111b [19] 11111117 [20] 1111111a [21] 1111111d [22] 11111118 [23]␣
↪11111120 [24] 11111118 [25] 1111111c [26] 1111111d [27] 1111111e [28] 1111111a [29]␣
↪11111122 [30] 1111111e [31] 11111120 [32] 11111123 [33] 11111119 [34] 1111111c [35]␣
↪1111111d [36] 11111116 [37] 1111111a [38] 1111111d [39] 1111111c [40] 11111113 [41]␣
↪11111115 [42] 1111111d [43] 1111111f [44] 1111111e [45] 1111111c [46] 1111111f [47]␣
↪1111111e [48] 11111117 [49] 11111115 [50] 1111111a [51] 11111121 [52] 1111111f [53]␣
↪1111111b [54] 1111111b [55] 11111124 [56] 11111116 [57] 11111125 [58] 11111123 [59]␣
↪1111111b [60] 1111111a [61] 11111119 [62] 11111118 [63] 11111123
       v3: [0] 11111111 [1] 11111111 [2] 11111111 [3] 11111111 [4] 11111111 [5]␣
↪11111111 [6] 11111111 [7] 11111111 [8] 11111111 [9] 11111111 [10] 11111111 [11]␣
↪11111111 [12] 11111111 [13] 11111111 [14] 11111111 [15] 11111111 [16] 11111111 [17]␣
↪11111111 [18] 11111111 [19] 11111111 [20] 11111111 [21] 11111111 [22] 11111111 [23]␣
↪11111111 [24] 11111111 [25] 11111111 [26] 11111111 [27] 11111111 [28] 11111111 [29]␣
↪11111111 [30] 11111111 [31] 11111111 [32] 11111111 [33] 11111111 [34] 11111111 [35]␣
↪11111111 [36] 11111111 [37] 11111111 [38] 11111111 [39] 11111111 [40] 11111111 [41]␣
↪11111111 [42] 11111111 [43] 11111111 [44] 11111111 [45] 11111111 [46] 11111111 [47]␣
↪11111111 [48] 11111111 [49] 11111111 [50] 11111111 [51] 11111111 [52] 11111111 [53]␣
↪11111111 [54] 11111111 [55] 11111111 [56] 11111111 [57] 11111111 [58] 11111111 [59]␣
↪11111111 [60] 11111111 [61] 11111111 [62] 11111111 [63] 11111111
```

(continues on next page)

(continued from previous page)

```
       v4: [0] f10115b0 [1] 0000000a [2] 00000005 [3] 00000009 [4] 00000004 [5]␣
↪00000001 [6] 00000001 [7] 0000000a [8] 00000004 [9] 00000005 [10] 00000008 [11]␣
↪00000002 [12] 00000008 [13] 00000001 [14] 00000006 [15] 00000005 [16] 00000005 [17]␣
↪00000001 [18] 00000001 [19] 00000002 [20] 00000006 [21] 00000006 [22] 00000002 [23]␣
↪0000000a [24] 00000001 [25] 00000001 [26] 0000000a [27] 00000006 [28] 00000001 [29]␣
↪00000008 [30] 0000000a [31] 00000009 [32] 00000009 [33] 00000007 [34] 0000000a [35]␣
↪00000007 [36] 00000003 [37] 00000003 [38] 00000008 [39] 00000001 [40] 00000001 [41]␣
↪00000002 [42] 00000005 [43] 00000009 [44] 00000005 [45] 00000005 [46] 0000000a [47]␣
↪00000003 [48] 00000004 [49] 00000001 [50] 00000002 [51] 0000000a [52] 0000000a [53]␣
↪00000001 [54] 00000007 [55] 0000000a [56] 00000004 [57] 0000000a [58] 00000008 [59]␣
↪00000006 [60] 00000008 [61] 00000001 [62] 00000004 [63] 00000009
       v5: [0] 00007fd4 [1] 00007fd4 [2] 00007fd4 [3] 00007fd4 [4] 00007fd4 [5]␣
↪00007fd4 [6] 00007fd4 [7] 00007fd4 [8] 00007fd4 [9] 00007fd4 [10] 00007fd4 [11]␣
↪00007fd4 [12] 00007fd4 [13] 00007fd4 [14] 00007fd4 [15] 00007fd4 [16] 00007fd4 [17]␣
↪00007fd4 [18] 00007fd4 [19] 00007fd4 [20] 00007fd4 [21] 00007fd4 [22] 00007fd4 [23]␣
↪00007fd4 [24] 00007fd4 [25] 00007fd4 [26] 00007fd4 [27] 00007fd4 [28] 00007fd4 [29]␣
↪00007fd4 [30] 00007fd4 [31] 00007fd4 [32] 00007fd4 [33] 00007fd4 [34] 00007fd4 [35]␣
↪00007fd4 [36] 00007fd4 [37] 00007fd4 [38] 00007fd4 [39] 00007fd4 [40] 00007fd4 [41]␣
↪00007fd4 [42] 00007fd4 [43] 00007fd4 [44] 00007fd4 [45] 00007fd4 [46] 00007fd4 [47]␣
↪00007fd4 [48] 00007fd4 [49] 00007fd4 [50] 00007fd4 [51] 00007fd4 [52] 00007fd4 [53]␣
↪00007fd4 [54] 00007fd4 [55] 00007fd4 [56] 00007fd4 [57] 00007fd4 [58] 00007fd4 [59]␣
↪00007fd4 [60] 00007fd4 [61] 00007fd4 [62] 00007fd4 [63] 00007fd4
       v6: [0] 00007ffe [1] 00007ffe [2] 00007ffe [3] 00007ffe [4] 00007ffe [5]␣
↪00007ffe [6] 00007ffe [7] 00007ffe [8] 00007ffe [9] 00007ffe [10] 00007ffe [11]␣
↪00007ffe [12] 00007ffe [13] 00007ffe [14] 00007ffe [15] 00007ffe [16] 00007ffe [17]␣
↪00007ffe [18] 00007ffe [19] 00007ffe [20] 00007ffe [21] 00007ffe [22] 00007ffe [23]␣
↪00007ffe [24] 00007ffe [25] 00007ffe [26] 00007ffe [27] 00007ffe [28] 00007ffe [29]␣
↪00007ffe [30] 00007ffe [31] 00007ffe [32] 00007ffe [33] 00007ffe [34] 00007ffe [35]␣
↪00007ffe [36] 00007ffe [37] 00007ffe [38] 00007ffe [39] 00007ffe [40] 00007ffe [41]␣
↪00007ffe [42] 00007ffe [43] 00007ffe [44] 00007ffe [45] 00007ffe [46] 00007ffe [47]␣
↪00007ffe [48] 00007ffe [49] 00007ffe [50] 00007ffe [51] 00007ffe [52] 00007ffe [53]␣
↪00007ffe [54] 00007ffe [55] 00007ffe [56] 00007ffe [57] 00007ffe [58] 00007ffe [59]␣
↪00007ffe [60] 00007ffe [61] 00007ffe [62] 00007ffe [63] 00007ffe
       v7: [0] 3d3495ac [1] bd0dfb7a [2] bcc1143a [3] bca64d59 [4] bc112d79 [5]␣
↪3cbcc8c8 [6] 3ce69f7c [7] 3de967fe [8] bdee8d4d [9] 3c9e426b [10] bc6d380f [11]␣
↪3c18495c [12] be38843f [13] bd5a1da8 [14] 3d80c7e4 [15] bc978798 [16] 3cd52d8d [17]␣
↪bd58d230 [18] 3e2e91ac [19] bca54a71 [20] 3c3cea13 [21] 3c888a4b [22] 3de0a868 [23]␣
↪3d220de3 [24] 3ce4d6f8 [25] bc033ce0 [26] bb38519f [27] b9a4b621 [28] bd800802 [29]␣
↪bdb04d27 [30] bc826d02 [31] bd4aa05d [32] 3dae9483 [33] b921dac8 [34] 3d194f79 [35]␣
↪bd1ccbd9 [36] bd45f9c5 [37] bc1b4cb0 [38] 3db1ab4b [39] 3e0487ab [40] 3d37f334 [41]␣
↪3b983eb8 [42] 3caba2a4 [43] bd8944ea [44] be01bee7 [45] bbbf22d8 [46] 3d076472 [47]␣
↪bd2eb34c [48] 3c3da426 [49] 3d754b6d [50] 3c08a069 [51] bcdeca32 [52] be12e2e4 [53]␣
↪3c92d0e2 [54] 3d1480e4 [55] 3d817751 [56] 3db0072c [57] 3d6fc70b [58] bd6a67a1 [59]␣
↪3da0f9ed [60] 3b67b5e6 [61] bdb8002e [62] 3cd0a9b9 [63] 386eee2b

Local memory content:
    0x0000: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0020: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0040: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0060: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0080: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x00a0: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
```

(continues on next page)

(continued from previous page)

```
    0x00c0: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x00e0: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0100: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0120: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0140: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0160: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x0180: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x01a0: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x01c0: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111
    0x01e0: 22222222 11111111 22222222 11111111 22222222 11111111 22222222 11111111

Disassembly for function vector_add_assert_trap(int*, int*, int*):
    code object: file:////rocm-debug-agent/build/test/rocm-debug-agent-test#offset=14309&
↪size=31336
    loaded at: [0x7fd4f100c000-0x7fd4f100e070]

/rocm-debug-agent/test/vector_add_assert_trap.cpp:
55        c[gid] = a[gid] + b[gid] + (lds_check[0] >> 32);
    0x7fd4f100d0c4 <+196>:    s_waitcnt vmcnt(0) lgkmcnt(0)
    0x7fd4f100d0c8 <+200>:    v_add3_u32 v2, v2, v4, v3
    0x7fd4f100d0d0 <+208>:    global_store_dword v[0:1], v2, off
    0x7fd4f100d0d8 <+216>:    s_or_saveexec_b64 s[0:1], s[0:1]
    0x7fd4f100d0dc <+220>:    s_xor_b64 exec, exec, s[0:1]
    0x7fd4f100d0e0 <+224>:    s_cbranch_execz 65503  # 0x7fd4f100d060 <vector_add_assert_
↪trap(int*, int*, int*)+96>

53          __builtin_trap ();
    0x7fd4f100d0e4 <+228>:    s_mov_b64 s[0:1], s[6:7]
=>  0x7fd4f100d0e8 <+232>:    s_trap 2
    0x7fd4f100d0ec <+236>:    s_endpgm

End of disassembly.
Aborted (core dumped)
```

The supported triggering events are:

- **Memory fault**

A memory fault occurs when an AMDGPU accesses an inaccessible page. This leads to printing information about the memory fault, as shown below:

```
System event (HSA_AMD_GPU_MEMORY_FAULT_EVENT: page not present or supervisor privilege,␣
↪write access to a read-only page)
Faulting page: 0x7fbe4cc01000
```

There could be multiple memory faults, but the information about only one is printed.

A memory fault does not specify the wavefront that caused it. However, the stop reason for each wavefront is available. For example:

```
wave_0: pc=0x7fbe4cc0d0b4 (stopped, reason: MEMORY_VIOLATION)
```

- **Assert trap**

This occurs when an `s_trap 2` instruction is executed. The `__builtin_trap()` language builtin or `llvm.trap`

LLVM IR instruction can be used to generate this AMDGPU instruction.

- **Illegal instruction**

This occurs when the hardware detects an illegal instruction.

- **SIGQUIT**

A SIGQUIT signal can be sent to a process with the `kill -s SIGQUIT <pid>` command or by pressing `Ctrl-\`. See the `--disable-linux-signals` option for more information.

2.1 Options

Options are passed using the ROCM_DEBUG_AGENT_OPTIONS environment variable as shown:

```
ROCM_DEBUG_AGENT_OPTIONS="--all --save-code-objects" \
HSA_TOOLS_LIB=librocm-debug-agent.so.2 HSA_ENABLE_DEBUG=1 ./my_program
```

The following table lists the supported options:

Table 2.1: ROCdebug-agent options

Option	Description
`-a, --all`	Prints all wavefronts. If not specified, only wavefronts with a triggering event are printed.
`-s [DIR], --save-co`	Saves all loaded code objects. If the directory is not specified, the code objects are saved in the current directory. The file name in which the code object is saved is the same as the code object URI with special characters replaced by '_'. For example, the code object URI `file:///rocm-debug-agent/rocm-debug-agent-test#offset=14309&size=31336` is saved in a file with the name `file____rocm-debug-agent_rocm-debug-agent-test_offset_14309_size_31336`.
`-o <file-pat --output=`	Saves the output produced by the ROCdebug-agent in the specified file. By default, the output is redirected to `stderr`.
`-d, --disable`	Disables installation of `SIGQUIT` signal handler, so that the default Linux handler can dump a core file. By default, the ROCdebug-agent installs a `SIGQUIT` handler to print the state of all wavefronts when a `SIGQUIT` signal is sent to the process.
`-l <log-leve --log-lev`	Changes the ROCdebug-agent and ROCdbgapi log level. The log level can be none, info, warning, or error. The default log level is none.
`-h, --help`	Displays the usage and aborts the process.

CHAPTER

THREE

LICENSE

The University of Illinois/NCSA Open Source License (NCSA)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal with the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimers in the documentation and/or other materials provided with the distribution.
- Neither the names of Advanced Micro Devices, Inc, nor the names of its contributors may be used to endorse or promote products derived from this Software without specific prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE.