



Release Note for Arm Instruction Emulator

21.0

Non-Confidential

Copyright © 2018–2021 Arm Limited (or its affiliates).
All rights reserved.

Issue 01

109554_2100_01_en



Release Note for Arm Instruction Emulator

Copyright © 2018–2021 Arm Limited (or its affiliates). All rights reserved.

Release information

Document history

Issue	Date	Confidentiality	Change
1800-01	28 March 2018	Non-Confidential	18.0 release
1801-01	30 April 2018	Non-Confidential	18.1 release
1802-01	24 August 2018	Non-Confidential	18.2 release
1803-01	5 October 2018	Non-Confidential	18.3 release
1804-01	2 November 2018	Non-Confidential	18.4 release
1900-01	5 April 2019	Non-Confidential	19.0 release
1901-01	3 July 2019	Non-Confidential	19.1 release
1902-01	30 August 2019	Non-Confidential	19.2 release
2000-01	18 February 2020	Non-Confidential	20.0 release
2001-01	21 August 2020	Non-Confidential	20.1 release
2100-01	30 March 2021	Non-Confidential	21.0 release

Proprietary Notice

This document is protected by copyright and other related rights and the practice or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of Arm. No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether implementations infringe any third party patents.

THIS DOCUMENT IS PROVIDED “AS IS”. ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, Arm makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, patents, copyrights, trade secrets, or other rights.

This document may include technical inaccuracies or typographical errors.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

The Arm corporate logo and words marked with ® or ™ are registered trademarks or trademarks of Arm Limited (or its affiliates) in the US and/or elsewhere. All rights reserved. Other brands and names mentioned in this document may be the trademarks of their respective owners. Please follow Arm’s trademark usage guidelines at <https://www.arm.com/company/policies/trademarks>.

Copyright © 2018–2021 Arm Limited (or its affiliates). All rights reserved.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349|version 21.0)

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

The information in this document is Final, that is for a developed product.

Feedback

Arm welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on <https://support.developer.arm.com>

To provide feedback on the document, fill the following survey: <https://developer.arm.com/documentation-feedback-survey>.

Inclusive language commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used language that can be offensive. Arm strives to lead the industry and create change.

We believe that this document contains no offensive language. To report offensive language in this document, email terms@arm.com.

Contents

1. Release Note for Arm Instruction Emulator 21.0.....	6
--	---

1. Release Note for Arm Instruction Emulator 21.0

Arm Instruction Emulator (ArmIE) allows users to emulate SVE instructions on AArch64 hardware. Based on the DynamoRIO dynamic binary instrumentation API, ArmIE has added features for instrumenting AArch64 and SVE instructions. See [Release History](#) below for changes in this release.

System Requirements

ArmIE is supported on the following Linux platforms:

- AArch64 Redhat 8
- AArch64 SLES 15
- AArch64 Ubuntu 18.04

Installation Instructions

To install the emulator, you should untar the release package:

```
tar -xvzf ARM-Instruction-Emulator_21.0_AArch64_<OS>_<OS_Version>_aarch64.tar.gz
```

Where:

ARM-Instruction-Emulator_21.0_AArch64_<OS>_<OS_Version>_aarch64.tar.gz

is the name of the <os> <os_version> package.

```
cd ARM-Instruction-Emulator_21.0_AArch64_<OS>_<OS_Version>_aarch64
```

Install the emulator in /opt/arm as root user:

```
sudo ./arm-instruction-emulator-21.0_Generic-AArch64_<OS>-<OS_Version>_aarch64-linux-rpm.sh
```

Or use the `-i, --install-to <directory>` option to install to a specified location.

You must read the EULA, presented to you as part of the install process, before accepting or rejecting the agreement.

All files including the EULA file, `license_agreement.txt`, are unpacked to

`/opt/arm/arm-instruction-emulator-21.0_Generic-AArch64_<OS>-<OS_Version>_aarch64-linux`

If the `-i, --install-to <directory>` option is used, files are unpacked to

<directory>/arm-instruction-emulator-21.0_Generic-AArch64_<OS>-<OS_Version>_aarch64-linux

If, after installation, you decide not to agree to the terms of the EULA you must not run the emulator and you should delete all the installed files as described in [Uninstalling](#) below.

Known Issues

There are DynamoRIO defects and issues logged at <https://github.com/DynamoRIO/dynamorio/issues> which may result in instrumentation and emulation failures. Note that thorough testing and validation of this release has not resulted in any bugs related to those issues.

There is a known DynamoRIO issue which causes hangs on some AArch64 hardware: <https://github.com/DynamoRIO/dynamorio/issues/1698> ArmlE has a workaround for this using the `--unsafe-ldstex` option. This feature is enabled by default and has now been deprecated. It will be removed in a future release.

Documentation

Refer to the Arm Developer website for supporting information about Arm Instruction Emulator: <https://developer.arm.com/tools-and-software/server-and-hpc/arm-architecture-tools/arm-instruction-emulator>

This includes tutorials and other useful information for custom analysis of your code: <https://developer.arm.com/tools-and-software/server-and-hpc/compile/arm-instruction-emulator/resources>

Further help and tutorials for all of Arm's HPC tools can be found online at: <https://developer.arm.com/tools-and-software/server-and-hpc/help/help-and-tutorials>

There is also a `README.txt` file in `/opt/arm/arm-instruction-emulator-21.0_Generic-AArch64_<OS>-<OS_Version>_aarch64-linux`.

If you require support please contact support-hpc-sw@arm.com

Uninstalling

Run the `<directory>/arm-instruction-emulator-21.0_Generic-AArch64_<OS>-<OS_Version>_aarch64-linux/uninstall.sh` script.

For a default installation, `<directory>` is `/opt/arm`.

Release History

This section provides release history information.

Changes in Release 21.0

Removed Features

- None

New Features and Enhancements

- Added support for the following v8.2 instructions, based on detection of hardware capability at startup:
 - Scalar half-precision variants of `ucvtf` and `scvtf`.
 - Half-precision `fabd`.
- Updated GCC runtime support from GCC 9.2 to GCC 10.2.

Bug Fixes

- Fixed some hardware feature checks for v8.2 and v8.3 emulation.
- Fixed a v8.3 `fcmla` emulation error.

Other Changes

- The license that is included in ArmIE 21.0 is set to expire after 31st March 2022.

Changes in Release 20.1

Removed Features

- None

New Features and Enhancements

- Added support for the following v8.2 instructions based on detection of hardware capability at startup: `fmov`, `scvtf`, `frecpe`, `fabs`, `fcmtz`, `fcvtzs`, `frintn`, `frsqrt`, `frsqrts`, `fmax`, `fmaxp` and `fdiv`, `scvtf`, `ucvtf`

Bug Fixes

- Some SVE2 instructions that were not being emulated correctly have been fixed.
- CAS-164399-V9L7T0

Output from the `-s` option now shows double-quoted client parameter strings correctly.

Other Changes

- The license that is included in ArmIE 20.1 is set to expire after 31st August 2021.

Changes in Release 20.0

Removed Features

- The `--unsafe-ldstex` has been deprecated and is enabled by default. A new option, `--safe-ldstex`, has been added in the unlikely event that `--unsafe-ldstex` needs to be disabled.

New Features and Enhancements

- Added command option `--safe-ldstex`, (see 'Removed Features' above).
- Added support for the emulated v8.2 instructions: `frsqrt`, `frsqrts`, `fmax`, `fmaxp` and `fdiv`. Optimization of emulated v8.2 instructions: `fmax` and `fma`.
- Adds support for the full diagnostic reporting of illegal instructions (SIGILL), in other words, displaying the illegal encoding and its subject binary PC address.

- Added `armie` command line option, `--arg-iclient`, to enable the passing of arguments to instrumentation clients.
- Consistency updates to Region-of-Interest (ROI) feature, originally extended from memory tracing instrumentation client to instruction counting, opcode counting, and instruction tracing clients.
- Added an example of using Region-of-Interest (ROI) macros in Fortran source code in the `samples/fortran_roi` directory.
- Updated GCC linker/loader support from GCC 8.2 to GCC 9.2.

Bug Fixes

- Critical bug fix for OpenMPI applications.
- Improved installation compatibility with Arm Compiler for Linux and Arm Performance Libraries. This fixes a bug which required users to load the ArmIE environment module before loading the Arm Compiler for Linux and Arm Performance Libraries environment module.

Other Changes

- License is set to expire after 31st January 2021.

Changes in Release 19.2

Removed Features

- None.

New Features and Enhancements

- Emulation of the following v8.2 and v8.3 instructions based on detection of hardware capability at startup:
 - v8.2 half-precision instructions: `fadd`, `fsub`, `fmul`, `fmul (by element)`, `fmla`, `fmla (by element)`, `fmadd`, `fmsub`, `fmls`, `fmls (by element)`, `fneg`, `fcmp`.
 - v8.3: `fcmla`, `fcadd`
- Upgraded to be compatible with GCC 8.2 linker/loader, (from GCC 7.1).

Bug Fixes

- None.

Other Changes

- License set to expire after 31st August 2020.

Changes in Release 19.1

Removed Features

- None.

New Features and Enhancements

- Addition of an `uninstall.sh` script

Bug Fixes

- None.

Other Changes

- License extended to 31st March 2021.

Changes in Release 19.0

Removed Features

- None.

New Features and Enhancements

- The regions-of-interest feature (ROI), has been added to the following instrumentation clients:

```
api/samples/inscount_emulated.cpp  
api/samples/instrace_emulated.c  
api/samples/memtrace_emulated.c  
api/samples/opcodes_emulated.cpp
```

Bug Fixes

- An ELF loader bug which resulted in failures when running Fortran binaries has been fixed.

Other Changes

- Documentation has been updated to include more information about analysing specific aspects of runtime performance using existing instrumentation clients, or developing your own. See:
 - <https://developer.arm.com/tools-and-software/server-and-hpc/arm-architecture-tools/arm-instruction-emulator>
 - <https://developer.arm.com/tools-and-software/server-and-hpc/arm-architecture-tools/arm-instruction-emulator/tutorials>
 - <https://developer.arm.com/tools-and-software/server-and-hpc/arm-architecture-tools/arm-instruction-emulator/structure-of-an-instrumentation-client>
 - <https://developer.arm.com/tools-and-software/server-and-hpc/arm-architecture-tools/arm-instruction-emulator/building-custom-analysis-instrumentation>
 - <https://developer.arm.com/tools-and-software/server-and-hpc/arm-architecture-tools/arm-instruction-emulator/building-an-emulation-aware-instrumentation-client>

Changes in Release 18.4

Removed Features

- None.

New Features and Enhancements

- None.

Bug Fixes

- This version can successfully run multi-threaded subject binaries, including those linked with OpenMP libraries.

Other Changes

- None.

Changes in Release 18.3

Removed Features

- None.

New Features and Enhancements

- Emulation has been separated from instrumentation based on the addition of an emulation interface to the DynamoRIO API allowing users to develop their own instrumentation clients:

```
drmgr_is_emulation_start()  
drmgr_is_emulation_end()  
drmgr_get_emulation_instr_data()  
drmgr_create_emulation_start()  
drmgr_create_emulation_end()  
emulated_instr_t
```

Two instruction count clients, an opcodes client and a memory trace client have been provided as examples. Another emulation feature to simplify iteration over blocks with emulated instructions is planned for a future release. If accepted upstream, this emulation-aware iterator will simplify instrumentation when running with emulation clients.

Bug Fixes

- None

Other Changes

- A script, `enc2instr.py`, which can be used to disassemble instruction encodings to mnemonics. Useful for disassembling the output from `drmgr_get_emulation_instr_data()`. The script is based on `llvm-mc` and defaults to disassembling SVE, but can be any legal `-mattr` value supported by `llvm-mc`.
- All license files are now in the `license_terms` directory.

Changes in Release 18.2

Removed Features

- None.

New Features and Enhancements

This release begins the process of separating emulation clients from instrumentation clients. The implementation of an instruction count client and memory tracing clients are the first examples of such a separation.

Bug Fixes

- A bug in counting non-SVE instructions has been fixed.

Other Changes

- A helper option to output the full drrun command used by armie: `-s, --show-drrun-cmd`.

Changes in Release 18.1

Removed Features

- None.

New Features and Enhancements

- There are significant improvements in runtime performance principally due to optimisations in SVE emulation functions (mostly floating-point emulation including faster FCMLA) and memory copying.

Bug Fixes

- Floating point corrections including:
 - Handling of inactive vector elements.
 - Floating point <-> integer conversions.
 - FPSR updates by and for modifying instructions.
 - Operand ordering for commutative operations.
 - Handling of NaNs.

Other Changes

- None.

Changes in Release 18.0

This is the first release which integrates ArmIE with the Open Source DynamoRIO dynamic binary instrumentation platform (<http://www.dynamorio.org/home.html>).

The previous release of Arm Instruction Emulator was 1.2.1. The version numbering scheme has changed with this release in order to remain consistent with other Arm AArch64 HPC software tools.

Removed Features:

The functionality enabled by the following command line options is not available in this release. They may be added in future releases.

- `-d, --debug`

Enabled assertion checks in the emulator.

- `-o, --output`

Redirected all messages generated by armie to a file.

- `-p, --profile-period`

Enabled the performance profiler.

- `-msve-memtrace`

Enabled memory tracing.

New Features:

- Arm Instruction Emulator 18.0 runs using DynamoRIO clients to provide SVE emulation and limited instrumentation for both emulated SVE instructions and native AArch64 instructions. Previous releases emulated SVE instructions with no coverage of native AArch64 instructions.

Bug Fixes:

- None. This is the first release based on DynamoRIO.

Other Changes:

- None.