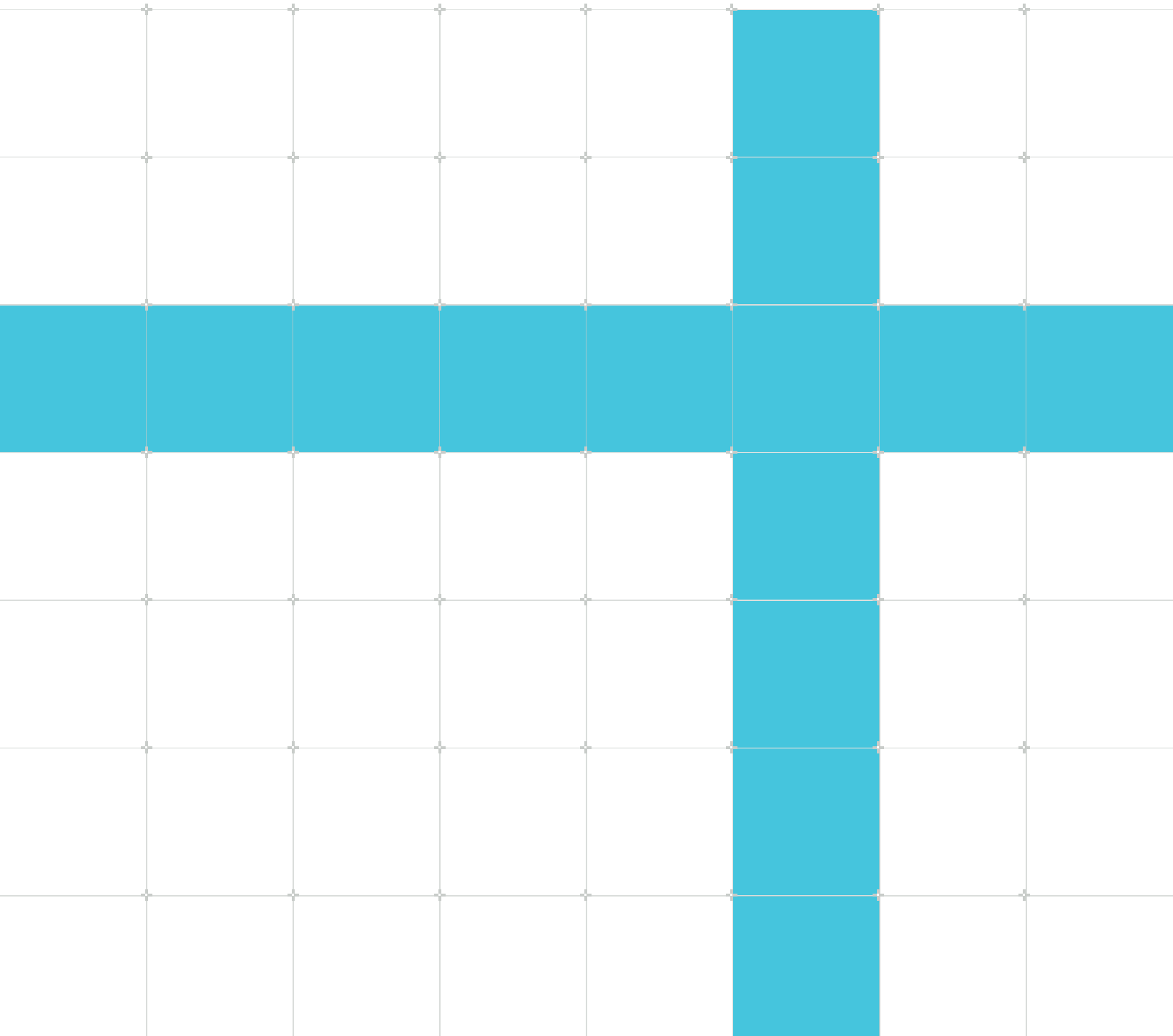
Arm Compiler for Linux Release Note

Revision 2410

Issue 00
107578_2410_00_en



Arm Compiler for Linux Release Note

This document (107578_2410_00_en) was issued on 2024-10-17. There might be a later issue at https://developer.arm.com/documentation/107578

The product revision is 2410.

See also: Proprietary notice | Product and document information | Useful resources

Start reading

If you prefer, you can skip to the start of the content.

Intended audience

Software developers

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.

This document includes language that can be offensive. We will replace this language in a future issue of this document.

To report offensive language in this document, email terms@arm.com.

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.

Contents

1. Arm Compiler for Linux 24.10 Release Note

Arm conventions and proprietary notices, including confidentiality status, terminology statement, and product release status, can be found at the end of this document.

2. Contents

This release note contains the following sections:

3. Release overview

This is the 24.10 release of the Arm Compiler for Linux.

Product description

The Arm Compiler for Linux 24.10 suite provides a complete compiling environment for natively developing and tuning your server and HPC applications on Arm-based platforms.

The suite contains the following packages:

- Arm C/C++/Fortran Compiler 24.10

  Arm Compiler is a Linux user-space C/C++ and Fortran compiler tailored for scientific computing, HPC, and enterprise workloads.
- Arm Performance Libraries 24.10.0

  Arm Performance Libraries contains optimized math functions, such as linear algebra and Fast Fourier Transforms, for Arm AArch64 implementations, including those with SVE. It is compatible with Arm C/C++/Fortran Compiler 24.10 and GCC 14.2.0.

  Arm Performance Libraries is optimized for a number of microarchitectures. The latest information is available on the Arm Developer website:

  https://developer.arm.com/Tools%20and%20Software/Arm%20Compiler%20for%20Linux#Supported-Devices
- GCC 14.2.0

  For convenience, and to provide the optimal experience of using Arm Performance Libraries and GCC on the latest Arm server and HPC systems, a build of GCC 14.2.0 is provided. The GCC 14.2.0 build is also provided for OpenMP/libgfortran compatibility with Arm Performance Libraries.

  As a GNU tool suite, the GPL-licensed source code can be downloaded separately.

  https://gcc.gnu.org/gcc-14/changes.html

Release status

This is the 24.10 release of the Arm Compiler for Linux software.

These deliverables are being released under the terms of the agreement between Arm and each licensee (the “Agreement”). All planned verification and validation is complete. The release is suitable for volume production under the terms of the Agreement.

Release contents

The following subsections describe:

- The product parts that are delivered as part of this release.
- Any changes since the previous release.

- Any known issues and limitations that exist at the time of this release.

Deliverables

The following product parts are delivered as part of this release:

- Arm C/C++/Fortran Compiler 24.10
- Arm Performance Libraries 24.10.0
- GCC 14.2.0
- Release Notes (this document)
- Documentation

  Arm Compiler for Linux reference guides are available in install_location/share-24.10*/documentation. The guides that are in the '/documentation' location are also available on the Arm Developer website:

  https://developer.arm.com/Tools%20and%20Software/Arm%20Compiler%20for%20Linux#Resources

  The same Arm Developer web page also contains links to tutorials, installation guides, and application porting guides.

  Documentation and release notes might change between product releases. For the latest documentation bundle, check the product download page.

  Arm tests PDFs only in Adobe Acrobat and Acrobat Reader. Arm cannot guarantee the quality of its PDFs when used with any other PDF reader. Adobe reader products are available at https://www.adobe.com.

Differences from previous release

Arm Compiler for Linux 24.10 includes various internal changes that resolve defects and improve performance.

The following subsections describe the significant differences from the previous release of Arm Compiler for Linux.

3.1 Additions and changes:

This section describes the new features or components added, or any significant technical changes to features or components, in the 24.10 release.

- Arm Compiler for Linux suite 24.10:
  - Upgraded the version of GCC, from GCC 13.2.0 to GCC 14.2.0.
  - Red Hat Enterprise Linux 7 (RHEL-7) support was deprecated in the 24.04 release and has now been removed.
- Arm Performance Libraries 24.10:

- The Arm Neoverse N3 core is added as a new microarchitecture target.
- Tunings for Neoverse V2 systems now include Google Axion (in addition to NVIDIA Grace and AWS Graviton4).
- The random number generator (RNG) functions now support all of the SOBOL functionality in the OpenRNG interface.
  - Note that the RNG functions are now available as open source: https://gitlab.arm.com/libraries/openrng
- Increased performance for:
  - ?GEMM, in particular large parallel problems.
  - FFTs:
    - Large 1D FFT cases may now be multi-threaded.
    - In-place multi-dimensional FFT cases now use significantly less memory.
    - Cases involving large prime factors on Windows.
  - MT19937 RNG skipahead.
- All sparse matrix formats now multi-threaded for sparse triangular solve.
- Support for LAPACK version 3.12.0.
- Performance improvements in libamath, for:
  - Vectorized versions of sincospi, sincospif, modf and modff.

3.2 Resolved issues:

Describes any technical issues that are resolved in the 24.10 release.

- In September 2023, a security vulnerability was disclosed on the GCC toolchain: CVE-2023-4039. The GCC 14.2.0 package distributed with ACfL 24.10 now contains the mitigations for this vulnerability. For more details, please read the full update at:

  https://developer.arm.com/Arm%20Security%20Center/GCC%20Stack%20Protector%20Vulnerability%20AArch64
- A segmentation fault has been occurring in specific cases while querying `TargetTransformInfo` object when targeting SVE. This was a regression introduced in ACfL 24.04 which has been fixed for the ACfL 24.10 release.
- A segmentation fault has been occuring in rare cases during type legalisation within `SelectionDAG`. This was a regression introduced in ACfL 24.04 which has been fixed for the ACfL 24.10 release.
- Bug fixed in Arm Performance Libraries for user-reported FFT wisdom export crash.
- ?[io]matcopy functions transpose correctly for row-major matrices in Arm PL.

Known limitations

The following subsection describes any issues that are known at the time of this release.

3.3 Open technical issues:

There are no open technical issues in the 24.10 release.

4. Get started

This section describes information to help you get started with accessing, setting up, and using Arm Compiler for Linux.

For more information, see the Get Started information on the Arm Developer website:

https://developer.arm.com/documentation/101458/2404/Get-started

Licensing information

Use of Arm Compiler for Linux is subject to the terms and conditions of the applicable End User License Agreement ("EULA"). A copy of the EULA can be found in the 'license_terms' folder of your product installation.

You do not require a license to use this Arm Compiler for Linux package.

Prerequisites

If any of the following tools are not already installed by your Linux distribution, you must install them before installing Arm Compiler for Linux. These packages can be installed with the appropriate package manager for your OS:

- SLES: awk environment-modules glibc-devel gzip python3 tar
- RHEL: environment-modules glibc-devel procps python3
- Amazon Linux: environment-modules glibc-devel gzip procps python3 tar
- Ubuntu: environment-modules libc6-dev python3

Note: The minimum supported version for Python is version 3.6.

You must have at least 2 GB of free hard disk space to both download and unpack the Arm Compiler for Linux package. You must also have an additional 6 GB of free space to install the package.

Download the product

Arm delivers the files through the Arm Developer website:

https://developer.arm.com/Tools%20and%20Software/Arm%20Compiler%20for%20Linux#Technical-Specifications

Unpack the product

To unpack the package, extract the tar file contents using a tar utility:

```
tar -xvf package_name.tar
```

Directory structure:

Shows the top-level directory structure of this installer package, which is available after you unpack the bundle:

```
license_terms/
arm-compiler-for-linux_24.10*.sh
RELEASE_NOTES.txt
```

Install the product

To install Arm Compiler for Linux, navigate into the extracted package directory (package_name) and run the installation script as a privileged user. Pass any options to configure the installation:

```
cd path/to/package_name/
./arm-compiler-for-linux_24.10*.sh [option]...
```

Some common installation options are:

- For a headless installation and to automatically accept the EULA, use the `--accept` option.
- To install to an alternate location to the default, use the `--install-to install_location` option.

For a full list of supported installation options pass the `-h` or `--help` options to the installer script.

To learn more about installing Arm Compiler for Linux, see:

https://developer.arm.com/documentation/102621/0100/Install-Arm-Compiler-for-Linux

The installer displays the EULA and prompts you to agree to the terms. Type 'yes' at the prompt to continue.

All the packages are unpacked to install_location/package_name with environment modulefiles available under install_location/modulefiles. The default installation location is /opt/arm/. Local installs have the same directory structure starting from your chosen installation root.

RPM and DEB files

The install packages contain RPM (.rpm) files, for Linux distributions that use the Red Hat Package Manager (including SLES and Amazon Linux), or DEB (.deb) files, for Debian-based Linux distributions.

To extract the .rpm or .deb files from the installer, run the installer script with the `-s` or `--save-packages-to directory_location` option. If directory_location is not an empty directory, you also need to include the `-f` or `--force` option. The installer script requires you to accept the EULA. If you accept the EULA, the .rpm or .deb files extract to directory_location.

RPM files are signed by Arm's HPC GPG key. DEB files are not signed. To verify RPM files, you can download and import the Arm's HPC GPG key, and check the signatures:

1. Download the Arm HPC GPG public key from:

https://developer.arm.com/-/media/files/keys/GPG-PUB-KEY-ARM-HPC-SW-TOOLS.PUB

2. Import the GPG key, run:

```
rpm --import GPG-PUB-KEY-ARM-HPC-SW-TOOLS.PUB
```

3. Check the signature of an .rpm file, run:

```
rpm -K rpm_file
```

To install Arm Compiler for Linux using rpm/deb files, navigate into the extracted rpm/deb files directory, run:

```
rpm -i list_of_rpm_files
dpkg -i list_of_deb_files
```

Note: Arm does not recommend that you install directly from the .rpm or .deb files. Only experienced users who are comfortable with this type of installation route should attempt to install the Arm Compiler for Linux package using this method.

Run the product

The following information can be used to run the product:

1. Load the environment module:
   - Ensure you have access to modules, replace /opt/arm with install_location if necessary, and use:

     module use /opt/arm/modulefiles module avail
   - For Arm C/C++/Fortran Compiler, use:

     module load acfl/24.10

   To also use Arm Performance Libraries, include the `-armpl` compiler option when linking your executable. You do not need to load the Arm Performance Libraries modulefile.
   - For GCC 14.2.0 only, use:

     module load gnu/14.2.0
   - For GCC 14.2.0 with Arm Performance Libraries, use:

     module load gnu/14.2.0 module load armpl/24.10.0
2. Generate your executable binary.

To generate an executable binary with Arm Compiler for Linux, compile your program with Arm C/C++/Fortran Compiler and specify any options ([options]), the output binary name (-o binary), and the input file (input):

```
{armclang|armclang++|armflang} [options] -o binary input
```

Refer to the GCC documentation to see the equivalent command syntax for the GCC compiler.

Examples

Example code is included in this suite as part of Arm Performance Libraries. This code can be found at:

```
install_location/ARMPL_Name*ARMPL_Version*/examples*
```

Examples that use, and do not use, SVE are included for each of Arm C/C++/Fortran Compiler and GCC.

Multiple examples directories are provided in the installation. The suffix of the directory name indicates whether the examples inside link to the 32-bit ('_lp64') or 64-bit ('_ilp64') integer variants, and sequential (no suffix indicator) or OpenMP ('_mp') multi-threaded variants, of Arm Performance Libraries.

The default set of examples in the 'examples' directory link to the sequential, 32-bit integers variant of Arm Performance Libraries.

To build the default set of examples:

- For Arm Compiler for Linux:
  1. Copy the 'examples' directory somewhere writeable:

     cp -r install_location/armpl-24.10.0*arm-linux-compiler/examples ./ cd examples
  2. Load the Arm Compiler for Linux environment module:

     module load acfl/24.10
  3. Build the examples:

     make
- For GCC:
  1. Copy the 'examples' directory somewhere writeable:

     cp -r install_location/armpl-24.10.0*gcc/examples ./ cd examples
  2. Load the GCC environment modules:

     module load gnu/14.2.0 module load armpl/24.10.0
  3. Build the examples:

make

- Note: The examples above only work with the modules listed in their respective step 2. For instance, the GCC examples will not work if the Arm Compiler for Linux variant of the performance libraries is already loaded. The modules environment can be cleaned with “module purge”
- Note: For Linux distributions that use the Red Hat Package Manager,if you want to change the install location of ACfL, you need to first uninstall the package and then re-install it to the desired location. This is only required when installing the product as a privileged user.

For more information about the Arm Performance Libraries examples, see:

https://developer.arm.com/documentation/102574/0100/Compile-and-test-the-examples

Uninstall

For convenience, this package includes an “uninstall.sh” script at:

```
install_location/share-24.10*/uninstall.sh
```

This script attempts to uninstall all the components supplied as part of Arm Compiler for Linux. However, if other packages outside of this product depend on the GCC component, GCC will not be uninstalled. Packages extracted using the –save-packages-to option and installed using rpm/dpkg commands should be uninstalled in the same way and deleted manually.

5. Support

The documentation that is available for Arm Compiler for Linux can be found on the product resources page on the Arm Developer website:

https://developer.arm.com/Tools%20and%20Software/Arm%20Compiler%20for%20Linux#Resources

You can also find a subset of that documentation, available in

```
install_location/share-24.10*/documentation.
```

These deliverables are being released under the terms of the agreement between Arm and each licensee (the “Agreement”). All planned verification and validation is complete. The release is suitable for volume production under the terms of the Agreement.

OS

This suite is supported on the following Linux platforms:

- AArch64 RHEL 8 and 9
- AArch64 SLES 15 Service Packs 5 and 6
- AArch64 Ubuntu 20.04 and 22.04
- AArch64 Amazon Linux 2 and 2023

Full information about the platforms supported by Arm Compiler for Linux is available on the Arm Developer website:

https://developer.arm.com/Tools%20and%20Software/Arm%20Compiler%20for%20Linux#Supported-Devices

6. Release history

A full release history (with release notes) for Arm Compiler for Linux is available on the Arm Developer website:

https://developer.arm.com/documentation/101458/latest

7. Glossary

The Arm Glossary is a list of terms that are used in Arm documentation, together with definitions for those terms. The Arm Glossary does not contain terms that are industry standard unless the Arm meaning differs from the generally accepted meaning.

See the Arm Glossary for more information: https://developer.arm.com/glossary.

8. Non-Confidential Proprietary Notice

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349)

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.

Web Address

https://developer.arm.com

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 terms. To report offensive language in this document, email terms@arm.com.

Proprietary Notice

This document is protected by copyright and other related rights and the use 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 Limited ("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 the subject matter of this document infringes any third party patents.

The content of this document is informational only. Any solutions presented herein are subject to changing conditions, information, scope, and data. This document was produced using reasonable efforts based on information available as of the date of issue of this document. The scope of information in this document may exceed that which Arm is required to provide, and such additional information is merely intended to further assist the recipient and does not represent Arm's view of the scope of its obligations. You acknowledge and agree that you possess the necessary expertise in system security and functional safety and that you shall be solely responsible for compliance with all legal, regulatory, safety and security related requirements concerning your products, notwithstanding any information or support that may be provided by Arm herein. In addition, you are responsible for any applications which are used in conjunction with any Arm technology described in this document, and to minimize risks, adequate design and operating safeguards should be provided for by you.

This document may include technical inaccuracies or typographical errors. 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, any patents, copyrights, trade secrets, trademarks, or other rights.

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.

Reference by Arm to any third party's products or services within this document is not an express or implied approval or endorsement of the use thereof.

This document consists solely of commercial items. You shall be responsible for ensuring that any permitted 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 this document shall prevail.

The validity, construction and performance of this notice shall be governed by English Law.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

PRE-1121-V1.0

Product and document information

Read the information in these sections to understand the release status of the product and documentation, and the conventions used in Arm documents.

Product status

All products and services provided by Arm require deliverables to be prepared and made available at different levels of completeness. The information in this document indicates the appropriate level of completeness for the associated deliverables.

Product completeness status

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

Revision history

These sections can help you understand how the document has changed over time.

Document release information

The Document history table gives the issue number and the released date for each released issue of this document.

Document history

Issue	Date	Confidentiality	Change
0100-00	17 October 2024	Non-Confidential	Initial release

Change history

The Change history tables describe the technical changes between released issues of this document in reverse order. Issue numbers match the revision history in Document release information on page 22.
This is the current revision of this document.

Conventions

The following subsections describe conventions used in Arm documents.

Glossary

The Arm Glossary is a list of terms used in Arm documentation, together with definitions for those terms. The Arm Glossary does not contain terms that are industry standard unless the Arm meaning differs from the generally accepted meaning.

See the Arm Glossary for more information: developer.arm.com/glossary.

Typographic conventions

Arm documentation uses typographical conventions to convey specific meaning.

Convention	Use
italic	Citations.
bold	Interface elements, such as menu names. Terms in descriptive lists, where appropriate.
`monospace`	Text that you can enter at the keyboard, such as commands, file and program names, and source code.
`monospace underline`	A permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.
`<and>`	Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example: `MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>`
SMALL CAPITALS	Terms that have specific technical meanings as defined in the *Arm® Glossary*. For example, **IMPLEMENTATION DEFINED**, **IMPLEMENTATION SPECIFIC**, **UNKNOWN**, and **UNPREDICTABLE**.

Caution

We recommend the following. If you do not follow these recommendations your system might not work.

Warning

Your system requires the following. If you do not follow these requirements your system will not work.

You are at risk of causing permanent damage to your system or your equipment, or harming yourself.

This information is important and needs your attention.

A useful tip that might make it easier, better or faster to perform a task.

A reminder of something important that relates to the information you are reading.

Useful resources

This document contains information that is specific to this product. See the following resources for other useful information.

Access to Arm documents depends on their confidentiality:

- Non-Confidential documents are available at developer.arm.com/documentation. Each document link in the following tables goes to the online version of the document.
- Confidential documents are available to licensees only through the product package.