
Morello Supplement to the Arm C Language Extensions

Release ACLE Q3 2020

Arm Limited or its affiliates.

Sep 24, 2020

Contents

1	Preamble	2
1.1	Abstract	2
1.2	Keywords	2
1.3	Non-Confidential Proprietary Notice	2
1.4	Confidentiality status	3
1.5	Product Status	3
2	About This Document	4
2.1	Change Control	4
2.2	References	5
2.3	Terms & Abbreviations	5
3	Scope	6
4	Predefined macros	7
4.1	__ARM_FEATURE_C64	7
4.2	Capability Permissions	7
4.3	Deviation from CHERI	7
5	Builtin functions	8
5.1	Check subset and conditionally unseal or return null	8
5.2	Check subset and conditionally unseal	8
5.3	Convert pointer to capability offset (zeroing form)	8
	Index	9

Document number: 102273

Date of Issue: 30/09/2020

Preamble

1.1 Abstract

This document specifies the Arm C Language Extensions to enable C/C++ programmers to exploit the Morello architecture with minimal restrictions on source code portability.

1.2 Keywords

Predefined macros, builtin functions

1.3 Non-Confidential 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, third party 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.

If any of the provisions contained in these terms conflict with any of the provisions of any click through or signed written agreement covering this document with Arm, then the click through or signed written agreement prevails over and supersedes the conflicting provisions of these terms. 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.

Words and logos 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 <http://www.arm.com/company/policies/trademarks>.

Copyright © 2011, 2020. 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

1.4 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.

1.5 Product Status

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

Web Address <http://www.arm.com>

About This Document

2.1 Change Control

2.1.1 Current Status and Anticipated Changes

The following support level definitions are used by the ACLE specifications:

Release Arm considers this specification to have enough implementations, which have received sufficient testing, to verify that it is correct. The details of these criteria are dependent on the scale and complexity of the change over previous versions: small, simple changes might only require one implementation, but more complex changes require multiple independent implementations, which have been rigorously tested for cross-compatibility. Arm anticipates that future changes to this specification will be limited to typographical corrections, clarifications and compatible extensions.

Beta Arm considers this specification to be complete, but existing implementations do not meet the requirements for confidence in its release quality. Arm may need to make incompatible changes if issues emerge from its implementation.

Alpha The content of this specification is a draft, and Arm considers the likelihood of future incompatible changes to be significant.

All content in this document is at the **Alpha** quality level.

2.1.2 Change History

Issue	Date	Change
00alpha	30th September 2020	Alpha release

2.2 References

This document refers to, or is referred to by, the following documents.

Ref	URL or other reference	Title
<i>ACLE-morello</i>	This document	Morello Supplement to the Arm C Language Extensions
ACLE	Document number: 101028	Arm C Language Extensions
<i>CHERI</i>	https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-947.pdf	CHERI C/C++ Programming Guide

2.3 Terms & Abbreviations

Capability The capability data type is an unforgeable token of authority which provides a foundation for fine grained memory protection and strong compartmentalisation.

Permissions The permissions mask controls how the capability can be used - for example, by authorizing the loading and storing of data and/or capabilities.

Deriving a capability A capability value CV2 is said to be derived from a capability value CV1 when CV2 is a copy of CV1 with optionally removed permissions and/or optionally narrowed bounds (base increased or limit reduced).

Sealing a capability When a capability is sealed it cannot be modified or dereferenced, but it can be used to implement opaque pointer types.

Chapter 3

Scope

The Morello Supplement to the Arm C Language Extensions highlights the language features added on top of the CHERI programming language to further exploit the Morello architecture. We strongly recommend reading the CHERI Pure-Capability C/C++ Programming Guide as preliminary material: <https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-947.pdf>

Chapter 4

Predefined macros

ACLE introduces several predefined macros that define how the C/C++ implementation uses the Morello architecture.

4.1 `__ARM_FEATURE_C64`

This macro indicates that the code is being compiled for the C64 ISA.

4.2 Capability Permissions

The following macros indicate capability permissions:

Name	Value
<code>__ARM_CAP_PERMISSION_EXECUTIVE__</code>	2
<code>__ARM_CAP_PERMISSION_MUTABLE_LOAD__</code>	64
<code>__ARM_CAP_PERMISSION_COMPARTMENT_ID__</code>	128
<code>__ARM_CAP_PERMISSION_BRANCH_SEALED_PAIR__</code>	256

Those can be used to form a bitmask that is acceptable for `cheri_perms_and()` and `cheri_perms_clear()`. The value of each macro corresponds to the permission bit as it appears in the architecture documentation.

4.3 Deviation from CHERI

The macro `__CHERI_CAP_PERMISSION_PERMIT_CCALL__` is not available on the Morello architecture.

Chapter 5

Builtin functions

ACLE standardizes builtin functions to access the Morello architecture. These are the following:

5.1 Check subset and conditionally unseal or return null

```
void* __capability  
__builtin_morello_subset_test_unseal_or_null(const void* __capability a,  
                                             const void* __capability b)
```

Assuming two valid capabilities a and b, with the former being sealed and the latter being unsealed, if a can be derived from b, then it unseals a and returns it, otherwise it returns a null capability.

5.2 Check subset and conditionally unseal

```
void* __capability  
__builtin_morello_chkssu(const void* __capability a,  
                        const void* __capability b)
```

Assuming two valid capabilities a and b, with the former being sealed and the latter being unsealed, if a can be derived from b, then it unseals a and returns it, otherwise it just returns a.

5.3 Convert pointer to capability offset (zeroing form)

```
void* __capability  
__builtin_morello_cvtz(const void* __capability a, size_t b)
```

If the specified offset b is zero, then it returns a null capability, otherwise it sets the offset of capability a to b and returns a. If capability a is sealed then the returned capability is marked invalid.

Index

C

Capability, [5](#)

D

Deriving a capability, [5](#)

P

Permissions, [5](#)

S

Sealing a capability, [5](#)