Arm® Cortex®-A73 MPCore Processor Cryptographic Extension

Revision: r1p0

Technical Reference Manual

100049_0100_05_en

Arm® Cortex®-A73 MPCore Processor Cryptographic Extension

Technical Reference Manual

Release Information

Document History

Issue	Date	Confidentiality	Change
0000-01	23 June 2015	Confidential	First release for r0p0
0001-02	27 October 2015	Confidential	First release for r0p1
0002-03	10 March 2016	Confidential	First release for r0p2
0002-04	03 June 2016	Confidential	Second release for r0p2
0100-05	22 June 2018	Non-Confidential	First release for r1p0

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.

The Arm corporate logo and words marked with ® or ™ are registered trademarks or trademarks of Arm Limited (or its subsidiaries) 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*.

Arm Limited. Company 02557590 registered in England.

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

http://www.arm.com

Contents

Arm® Cortex®-A73 MPCore Processor Cryptographic Extension Technical Reference Manual

Preface

Preface

This preface introduces the *Arm® Cortex®-A73 MPCore Processor Cryptographic Extension Technical Reference Manual*.

It contains the following:

- *About this book* on page 6.
- *Feedback* on page 8.

About this book

This document describes the optional cryptographic features of the Cortex®-A73 MPCore processor. It includes descriptions of the registers used by the Cryptographic Extension.

Product revision status

The r*m*p*n* identifier indicates the revision status of the product described in this book, for example, r1p2, where:

r*m* Identifies the major revision of the product, for example, r1.

p*n* Identifies the minor revision or modification status of the product, for example, p2.

Intended audience

This document is written for system designers, system integrators, and programmers who are designing or programming a *System-on-Chip* (SoC) that uses the Cortex®-A73 processor with the optional Cryptographic Extension.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction
This chapter describes the Cortex-A73 MPCore Cryptographic Extension.

Chapter 2 Programmers Model
This chapter describes the programmers model.

Appendix A Revisions
This appendix describes the technical changes between released issues of this document.

Glossary

The Glossary is a list of terms used in documentation, together with definitions for those terms. The 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.

Typographic conventions

italic
Introduces special terminology, denotes cross-references, and citations.

bold
Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.

`monospace`
Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.

`monospace`
Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.

`monospace italic`
Denotes arguments to monospace text where the argument is to be replaced by a specific value.

`monospace bold`
Denotes language keywords when used outside example code.

`<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

Used in body text for a few terms that have specific technical meanings, that are defined in the *Glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Additional reading

This book contains information that is specific to this product. See the following documents for other relevant information:

Arm publications

- *Arm® Cortex®-A73 MPCore Processor Technical Reference Manual* (100048).
- *Arm® Cortex®-A73 MPCore Processor Configuration and Sign-off Guide* (100050).
- *Arm® Cortex®-A73 MPCore Processor Integration Manual* (100051).
- *Arm® Architecture Reference Manual Armv8, for Armv8-A architecture profile* (DDI 0487).

Other publications

- *Advanced Encryption Standard.* (FIPS 197, November 2001).
- *Secure Hash Standard (SHS)* (FIPS 180-4, March 2012).

Feedback

Feedback on this product

If you have any comments or suggestions about this product, contact your supplier and give:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

Feedback on content

If you have comments on content then send an e-mail to *errata@arm.com*. Give:

- The title *Arm Cortex-A73 MPCore Processor Cryptographic Extension Technical Reference Manual*.
- The number 100049_0100_05_en.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

Arm also welcomes general suggestions for additions and improvements.

Note

Arm tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

Chapter 1
Introduction

This chapter describes the Cortex-A73 MPCore Cryptographic Extension.

It contains the following sections:

1.1 About the Cortex-A73 MPCore Processor Cryptographic Extension

The Cortex-A73 MPCore Processor Cryptographic Extension supports the Armv8 Cryptographic Extension.

The Cryptographic Extension adds new A64, A32, and T32 instructions to Advanced SIMD that accelerate *Advanced Encryption Standard* (AES) encryption and decryption, and the *Secure Hash Algorithm* (SHA) functions SHA1 and SHA2-256.

Note

The optional Cryptographic Extension is not included in the base product. Arm supplies the Cryptographic Extension only under an additional license to the Cortex-A73 processor and Advanced SIMD and floating-point support licenses.

1.2 Revisions

This section describes the differences in functionality between product revisions:

r0p0

First release.

r0p1

No technical changes.

r0p2

No technical changes.

r1p0

No technical changes.

Chapter 2
Programmers Model

This chapter describes the programmers model.

It contains the following sections:

2.1 About the programmers model

This section describes the registers of the Cortex-A73 MPCore Cryptographic Extension and provides programming information.

See the *Arm® Architecture Reference Manual Armv8, for Armv8-A architecture profile* for more information.

This section describes:

- *2.1.1 Identifying the implemented cryptographic instructions* on page 2-13.
- *2.1.2 Disabling the Cryptographic Extension* on page 2-13.

2.1.1 Identifying the implemented cryptographic instructions

Software can identify the implemented cryptographic instructions by reading:

- ID_AA64ISAR0_EL1 in the AArch64 execution state.
- ID_ISAR5_EL1 in the AArch64 execution state.
- ID_ISAR5 in the AArch32 execution state.

2.1.2 Disabling the Cryptographic Extension

To disable the Cryptographic Extension for each individual core, assert the corresponding bit of the **CRYPTODISABLE** input signal. This signal is only sampled during reset of the processor.

When **CRYPTODISABLE** is asserted:

- Executing a cryptographic instruction results in an UNDEFINED exception.
- The ID registers described in *Table 2-1 Cryptographic Extension register summary* on page 2-14 indicate that the Cryptographic Extension is not implemented.

2.2 Register summary

The following table lists the instruction identification registers for the Cortex-A73 MPCore Cryptographic Extension:

Table 2-1 Cryptographic Extension register summary

Name	Execution state	Description
ID_AA64ISAR0_EL1	AArch64	See *2.3.1 AArch64 Instruction Set Attribute Register 0, EL1* on page 2-15.
ID_ISAR5_EL1	AArch64	See *2.3.2 AArch32 Instruction Set Attribute Register 5* on page 2-16.
ID_ISAR5	AArch32	See *2.3.3 Instruction Set Attribute Register 5* on page 2-18.

2.3 Register descriptions

This section describes the Cortex-A73 Cryptographic Extension registers.

Table 2-1 Cryptographic Extension register summary on page 2-14 provides cross-references to individual registers.

2.3.1 AArch64 Instruction Set Attribute Register 0, EL1

The ID_AA64ISAR0_EL1 characteristics are:

Purpose

Provides information about the optional cryptographic instructions that the processor can support.

Note

The optional Cryptographic Extension is not included in the base product. Arm supplies the Cryptographic Extension only under an additional license to the Cortex-A73 MPCore processor and Advanced SIMD and floating-point support licenses.

Usage constraints

This register is accessible as follows:

EL0	EL1 (NS)	EL1 (S)	EL2	EL3 (SCR.NS = 1)	EL3 (SCR.NS = 0)
-	RO	RO	RO	RO	RO

Configurations

ID_AA64ISAR0_EL1 is architecturally mapped to external register ID_AA64ISAR0_EL1.

Attributes

ID_AA64ISAR0_EL1 is a 64-bit register.

The following figure shows the ID_AA64ISAR0_EL1 bit assignments.

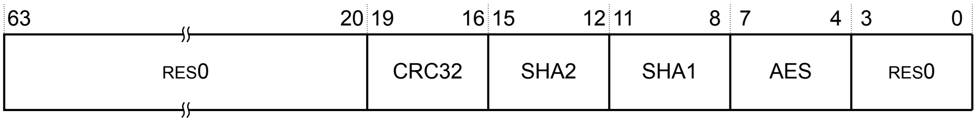

Figure 2-1 ID_AA64ISAR0_EL1 bit assignments

The following table shows the ID_AA64ISAR0_EL1 bit assignments.

Table 2-2 ID_AA64ISAR0_EL1 bit assignments

Bits	Name	Function
[63:20]	-	Reserved, RES0.
[19:16]	CRC32	Indicates whether CRC32 instructions are implemented. The value is: `0x1` CRC32 instructions are implemented.

Table 2-2 ID_AA64ISAR0_EL1 bit assignments (continued)

Bits	Name	Function
[15:12]	SHA2	Indicates whether SHA2 instructions are implemented. The possible values are: `0x0` No SHA2 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x1` `SHA256H`, `SHA256H2`, `SHA256SU0`, and `SHA256SU1` are implemented. This is the value if the implementation includes the Cryptographic Extension.
[11:8]	SHA1	Indicates whether SHA1 instructions are implemented. The possible values are: `0x0` No SHA1 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x1` `SHA1C`, `SHA1P`, `SHA1M`, `SHA1H`, `SHA1SU0`, and `SHA1SU1` are implemented. This is the value if the implementation includes the Cryptographic Extension.
[7:4]	AES	Indicates whether AES instructions are implemented. The possible values are: `0x0` No AES instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x2` `AESE`, `AESD`, `AESMC`, and `AESIMC` are implemented, plus `PMULL` and `PMULL2` instructions operating on 64-bit elements. This is the value if the implementation includes the Cryptographic Extension.
[3:0]	-	Reserved, RES0.

To access the ID_AA64ISAR0_EL1:

```
MRS <Xt>, ID_AA64ISAR0_EL1 ; Read ID_AA64ISAR0_EL1 into Xt
```

ID_AA64ISAR0_EL1[31:0] can be accessed through the internal memory-mapped interface and the external debug interface, offset `0xD30`.

Register access is encoded as follows:

Table 2-3 ID_AA64ISAR0_EL1 access encoding

op0	op1	CRn	CRm	op2
11	000	0000	0110	000

2.3.2 AArch32 Instruction Set Attribute Register 5

The ID_ISAR5_EL1 characteristics are:

Purpose

Provides information about the instruction sets that the processor implements in AArch32.

Note

The optional Cryptographic Extension is not included in the base product. Arm supplies the Cryptographic Extension only under an additional license to the Cortex-A73 MPCore processor and Advanced SIMD and floating-point support licenses.

Usage constraints

This register is accessible as follows:

EL0	EL1 (NS)	EL1 (S)	EL2	EL3 (SCR.NS = 1)	EL3 (SCR.NS = 0)
-	RO	RO	RO	RO	RO

Configurations

ID_ISAR5_EL1 is architecturally mapped to AArch32 register ID_ISAR5. See *2.3.3 Instruction Set Attribute Register 5* on page 2-18.

Attributes

ID_ISAR5_EL1 is a 32-bit register.

The following figure shows the ID_ISAR5_EL1 bit assignments.

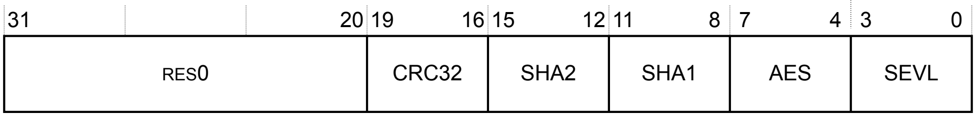

Figure 2-2 ID_ISAR5_EL1 bit assignments

The following table shows the ID_ISAR5_EL1 bit assignments.

Table 2-4 ID_ISAR5_EL1 bit assignments

Bits	Name	Function
[31:20]	-	Reserved, RES0.
[19:16]	CRC32	Indicates whether CRC32 instructions are implemented in AArch32 state. The value is: `0x1` CRC32 instructions are implemented.
[15:12]	SHA2	Indicates whether SHA2 instructions are implemented in AArch32 state. The possible values are: `0x0` No SHA2 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x1` `SHA256H`, `SHA256H2`, `SHA256SU0`, and `SHA256SU1` are implemented. This is the value if the implementation includes the Cryptographic Extension.
[11:8]	SHA1	Indicates whether SHA1 instructions are implemented in AArch32 state. The possible values are: `0x0` No SHA1 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x1` `SHA1C`, `SHA1P`, `SHA1M`, `SHA1H`, `SHA1SU0`, and `SHA1SU1` are implemented. This is the value if the implementation includes the Cryptographic Extension.

Table 2-4 ID_ISAR5_EL1 bit assignments (continued)

Bits	Name	Function
[7:4]	AES	Indicates whether AES instructions are implemented in AArch32 state. The possible values are: `0x0` No AES instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x2` `AESE`, `AESD`, `AESMC`, and `AESIMC` are implemented, plus `PMULL` and `PMULL2` instructions operating on 64-bit elements. This is the value if the implementation includes the Cryptographic Extension.
[3:0]	SEVL	Indicates whether the `SEVL` instruction is implemented. The value is: `0x1` `SEVL` is implemented to send event local.

To access the ID_ISAR5_EL1:

```
MRS <Xt>, ID_ISAR5_EL1 ; Read ID_ISAR5_EL1 into Xt
```

Register access is encoded as follows:

Table 2-5 ID_ISAR5_EL1 access encoding

op0	op1	CRn	CRm	op2
11	000	0000	0010	101

2.3.3 Instruction Set Attribute Register 5

The ID_ISAR5 characteristics are:

Purpose

Provides information about the instruction sets implemented by the processor in AArch32.

Note

The optional Cryptographic Extension is not included in the base product. Arm supplies the Cryptographic Extension only under an additional license to the Cortex-A73 MPCore processor and Advanced SIMD and floating-point support licenses.

Usage constraints

This register is accessible as follows:

EL0 (NS)	EL0 (S)	EL1 (NS)	EL1 (S)	EL2	EL3 (SCR.NS = 1)	EL3 (SCR.NS = 0)
-	-	RO	RO	RO	RO	RO

The ID_ISAR5 must be interpreted with ID_ISAR0, ID_ISAR1, ID_ISAR2, ID_ISAR3, and ID_ISAR4.

Configurations

ID_ISAR5 is architecturally mapped to AArch64 register ID_ISAR5_EL1. See *2.3.2 AArch32 Instruction Set Attribute Register 5* on page 2-16.

There is one copy of this register that is used in both Secure and Non-secure states.

Attributes

ID_ISAR5 is a 32-bit register.

The following figure shows the ID_ISAR5 bit assignments.

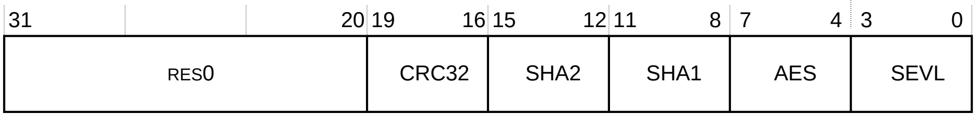

Figure 2-3 ID_ISAR5 bit assignments

The following table shows the ID_ISAR5 bit assignments.

Table 2-6 ID_ISAR5 bit assignments

Bits	Name	Function
[31:20]	-	Reserved, RES0.
[19:16]	CRC32	Indicates whether CRC32 instructions are implemented in AArch32 state. The value is: `0x1` CRC32 instructions are implemented.
[15:12]	SHA2	Indicates whether SHA2 instructions are implemented in AArch32 state. The possible values are: `0x0` No SHA2 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x1` `SHA256H`, `SHA256H2`, `SHA256SU0`, and `SHA256SU1` are implemented. This is the value if the implementation includes the Cryptographic Extension.
[11:8]	SHA1	Indicates whether SHA1 instructions are implemented in AArch32 state. The possible values are: `0x0` No SHA1 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x1` `SHA1C`, `SHA1P`, `SHA1M`, `SHA1H`, `SHA1SU0`, and `SHA1SU1` are implemented. This is the value if the implementation includes the Cryptographic Extension.
[7:4]	AES	Indicates whether AES instructions are implemented in AArch32 state. The possible values are: `0x0` No AES instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension. `0x2` `AESE`, `AESD`, `AESMC`, and `AESIMC` are implemented, plus `PMULL` and `PMULL2` instructions operating on 64-bit elements. This is the value if the implementation includes the Cryptographic Extension.
[3:0]	SEVL	Indicates whether the `SEVL` instruction is implemented. The value is: `0x1` `SEVL` is implemented to send event local.

To access ID_ISAR5:

```
MRC p15, 0, <Rt>, c0, c2, 5; Read ID_ISAR5 into Rt
```

Appendix A
Revisions

This appendix describes the technical changes between released issues of this document.

It contains the following section:
- *A.1 Revisions* on page Appx-A-21.

A.1 Revisions

This section describes the technical changes between released issues of this document.

Table A-1 Issue 0000-01

Change	Location	Affects
First release	-	-

Table A-2 Differences between issue 0000-01 and issue 0001-02

Change	Location	Affects
No technical changes	-	-

Table A-3 Differences between issue 0001-02 and issue 0002-03

Change	Location	Affects
No technical changes	-	-

Table A-4 Differences between issue 0002-03 and issue 0002-04

Change	Location	Affects
No technical changes	-	-

Table A-5 Differences between issue 0002-04 and issue 0100-05

Change	Location	Affects
No technical changes	-	-