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


Preface

About this book .................................................................................................................. 6
Feedback .......................................................................................................................... 8

Chapter 1

Introduction

1.1 About the Cortex-A73 MPCore Processor Cryptographic Extension ...................... 1-10
1.2 Revisions .................................................................................................................. 1-11

Chapter 2

Programmers Model

2.1 About the programmers model ................................................................................ 2-13
2.2 Register summary .................................................................................................... 2-14
2.3 Register descriptions ............................................................................................... 2-15

Appendix A

Revisions

A.1 Revisions .................................................................................................................. Appx-A-21
Preface

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

It contains the following:

- *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 rm pn identifier indicates the revision status of the product described in this book, for example, r1p2, where:

rm  Identifies the major revision of the product, for example, r1.
pn  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.
Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example:

\[ \text{MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>} \]

**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 Integration Manual* (100051).

**Other publications**

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 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 on page 1-10.
- 1.2 Revisions on page 1-11.
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 on page 2-13.
• 2.2 Register summary on page 2-14.
• 2.3 Register descriptions on page 2-15.
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>
<thead>
<tr>
<th>Name</th>
<th>Execution state</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID_AA64ISAR0_EL1</td>
<td>AArch64</td>
<td>See 2.3.1 AArch64 Instruction Set Attribute Register 0, EL1 on page 2-15.</td>
</tr>
<tr>
<td>ID_ISAR5_EL1</td>
<td>AArch64</td>
<td>See 2.3.2 AArch32 Instruction Set Attribute Register 5 on page 2-16.</td>
</tr>
<tr>
<td>ID_ISAR5</td>
<td>AArch32</td>
<td>See 2.3.3 Instruction Set Attribute Register 5 on page 2-18.</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th></th>
<th>EL0</th>
<th>EL1 (NS)</th>
<th>EL1 (S)</th>
<th>EL2 (SCR.NS = 1)</th>
<th>EL3 (SCR.NS = 0)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
</tr>
</tbody>
</table>

**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](image_url)

The following table shows the ID_AA64ISAR0_EL1 bit assignments.

<table>
<thead>
<tr>
<th>Bits</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>[63:20]</td>
<td>-</td>
<td>Reserved, RES0.</td>
</tr>
<tr>
<td>[19:16]</td>
<td>CRC32</td>
<td>Indicates whether CRC32 instructions are implemented. The value is: 0x1 CRC32 instructions are implemented.</td>
</tr>
</tbody>
</table>
Table 2-2  ID_AA64ISAR0_EL1 bit assignments (continued)

<table>
<thead>
<tr>
<th>Bits</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>[15:12]</td>
<td>SHA2</td>
<td>Indicates whether SHA2 instructions are implemented. The possible values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0x0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0x1</td>
</tr>
<tr>
<td>[11:8]</td>
<td>SHA1</td>
<td>Indicates whether SHA1 instructions are implemented. The possible values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0x0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0x1</td>
</tr>
<tr>
<td>[7:4]</td>
<td>AES</td>
<td>Indicates whether AES instructions are implemented. The possible values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0x0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0x2</td>
</tr>
<tr>
<td>[3:0]</td>
<td>-</td>
<td>Reserved, RES0.</td>
</tr>
</tbody>
</table>

To access the ID_AA64ISAR0_EL1:

```assembly
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

<table>
<thead>
<tr>
<th>op0</th>
<th>op1</th>
<th>CRn</th>
<th>CRm</th>
<th>op2</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>000</td>
<td>0000</td>
<td>0110</td>
<td>000</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>EL0</th>
<th>EL1 (NS)</th>
<th>EL1 (S)</th>
<th>EL2</th>
<th>EL3 (SCR.NS = 1)</th>
<th>EL3 (SCR.NS = 0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
</tr>
</tbody>
</table>

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](image)

The following table shows the ID_ISAR5_EL1 bit assignments.

<table>
<thead>
<tr>
<th>Bits</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>[31:20]</td>
<td>-</td>
<td>Reserved, RES0.</td>
</tr>
</tbody>
</table>
| [19:16] | CRC32      | Indicates whether CRC32 instructions are implemented in AArch32 state. The value is:  
|        |            | $0 \times 1$ CRC32 instructions are implemented.                           |
| [15:12] | SHA2       | Indicates whether SHA2 instructions are implemented in AArch32 state. The possible values are:  
|        |            | $0 \times 0$ No SHA2 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension.  
|        |            | $0 \times 1$ 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:  
|        |            | $0 \times 0$ No SHA1 instructions are implemented. This is the value if the implementation does not include the Cryptographic Extension.  
|        |            | $0 \times 1$ 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)

<table>
<thead>
<tr>
<th>Bits</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
</table>
| [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:

\[
\text{MRS <Xt>, ID_ISAR5_EL1} \quad \text{Read ID_ISAR5_EL1 into Xt}
\]

Register access is encoded as follows:

<table>
<thead>
<tr>
<th>op0</th>
<th>op1</th>
<th>CRn</th>
<th>CRm</th>
<th>op2</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>000</td>
<td>0000</td>
<td>0010</td>
<td>101</td>
</tr>
</tbody>
</table>

## 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:

<table>
<thead>
<tr>
<th>EL0 (NS)</th>
<th>EL0 (S)</th>
<th>EL1 (NS)</th>
<th>EL1 (S)</th>
<th>EL2</th>
<th>EL3 (SCR.NS = 1)</th>
<th>EL3 (SCR.NS = 0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>-</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
<td>RO</td>
</tr>
</tbody>
</table>

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](image)

The following table shows the ID_ISAR5 bit assignments.

<table>
<thead>
<tr>
<th>Bits</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>[31:20]</td>
<td>-</td>
<td>Reserved, RES0.</td>
</tr>
</tbody>
</table>
| [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

<table>
<thead>
<tr>
<th>Change</th>
<th>Location</th>
<th>Affects</th>
</tr>
</thead>
<tbody>
<tr>
<td>First release</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Change</th>
<th>Location</th>
<th>Affects</th>
</tr>
</thead>
<tbody>
<tr>
<td>No technical changes</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Change</th>
<th>Location</th>
<th>Affects</th>
</tr>
</thead>
<tbody>
<tr>
<td>No technical changes</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Change</th>
<th>Location</th>
<th>Affects</th>
</tr>
</thead>
<tbody>
<tr>
<td>No technical changes</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Change</th>
<th>Location</th>
<th>Affects</th>
</tr>
</thead>
<tbody>
<tr>
<td>No technical changes</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>