Arm® Corstone™ SSE-700 Secure Enclave

Revision: r0p0

Technical Reference Manual

arm

101870_0000_03_en

Arm® Corstone™ SSE-700 Secure Enclave

Technical Reference Manual

Release Information

Document History

Issue	Date	Confidentiality	Change
0000-00	27 November 2019	Confidential	First release for r0p0 BET
0000-01	13 December 2019	Confidential	First release for r0p0 LAC
0000-02	13 February 2020	Non-Confidential	Second release for r0p0 LAC
0000-03	08 July 2020	Non-Confidential	First release for r0p0 EAC

Non-Confidential Proprietary Notice

(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

www.arm.com

Contents

Arm® Corstone™ SSE-700 Secure Enclave Technical Reference Manual

Appendix A Revisions

Preface

This preface introduces the *Arm® Corstone™ SSE-700 Secure Enclave Technical Reference Manual*.

It contains the following:

About this book

This book is for Secure Enclave component within the Arm® Corstone™ SSE-700 Subsystem.

Product revision status

The r*m*p*n* identifier indicates the revision status of the product described in this book, for example, r*1*p*2*, 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 book is written for system designers, system integrators, and programmers who are designing or programming a *System-on-Chip* (SoC) that uses the SSE-700.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction
This chapter introduces the Secure Enclave and its features.

Chapter 2 Functional description
This chapter provides a functional description of the Secure Enclave.

Chapter 3 Programmers model
This chapter provides general information about the Secure Enclave memory map and registers, and information for programming the device.

Chapter 4 Software sequences
This chapter describes software sequences in the Secure Enclave.

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

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.

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.

`mono`space
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 *Arm® Glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Timing diagrams

The following figure explains the components used in timing diagrams. Variations, when they occur, have clear labels. You must not assume any timing information that is not explicit in the diagrams.

Shaded bus and signal areas are undefined, so the bus or signal can assume any value within the shaded area at that time. The actual level is unimportant and does not affect normal operation.

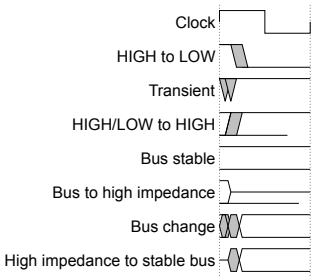

Figure 1 Key to timing diagram conventions

Signals

The signal conventions are:

Signal level
The level of an asserted signal depends on whether the signal is active-HIGH or active-LOW. Asserted means:

- HIGH for active-HIGH signals.
- LOW for active-LOW signals.

Lowercase n

At the start or end of a signal name, n denotes an active-LOW signal.

Additional reading

This section lists publications by Arm and by third parties.

See *Infocenter*, for access to Arm documentation.

Arm publications

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

- *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual* (101418)
- *Cortex®-M0+ Technical Reference Manual* (DDI 0484)
- *Arm® CoreSight™ System-on-Chip SoC-600 Technical Reference Manual* (100806)
- *Arm®v6-M Architecture Reference Manual* (DDI 0419)
- *AMBA® 5 AHB Protocol Specification* (IHI 0033)
- *AMBA® APB Protocol Specification Version 2.0* (IHI 0024)
- *AMBA® AXI and ACE Protocol Specification* (IHI 0022)
- *PrimeCell UART (PL011) Technical Reference Manual* (DDI 0183)
- *Arm® Cortex®-M System Design Kit Technical Reference Manual* (DDI 0479)

The following confidential books are only available to licensees, or require registration with Arm:

- *Arm® Corstone™ SSE-700 Subsystem Release Note* (PJDOC-1779577084-1555)
- *Cortex®-M0+ Integration and Implementation Manual* (DIT 0032)
- *Arm® CoreLink™ ADB-400 AMBA® Domain Bridge User Guide* (DUI 0615)
- *Arm® Corstone™ SSE-700 Subsystem Configuration and Integration Manual* (101419)
- *Arm® CoreLink™ PCK-600 Power Control Kit Configuration and Integration Manual* (101151)
- *Arm® Power Control System Architecture* (DEN 0050)

Feedback

Feedback on this product

If you have any comments or suggestions about this product, send an email to *support-subsystem-iot@arm.com* 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 Corstone SSE-700 Secure Enclave Technical Reference Manual*.
- The number 101870_0000_03_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 introduces the Secure Enclave and its features.

It contains the following section:

1.1 Overview

The Secure Enclave within the SSE-700, is a Cortex-M0+ based security subsystem that acts as the root-of-trust for the system. See also the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

The Secure Enclave holds and generates keys, and provides cryptographic services and security controls to the Host System. For example:

- Authenticating the firmware of the Secure Enclave itself, the Host System and the External System
- Enabling/disabling debug capabilities based on the secure state of the device

At power-on reset, it is the first system to boot and performs initial configuration of its own system and other components of the SSE-700, such as the Host System Firewall.

The Secure Enclave consists of:

- An Arm-v6 Cortex-M0+ Processor, with an in-built *Nested Vectored Interrupt Controller* (NVIC)
- Dedicated ROM and SRAM
- PL011 UART
- *Message Handling Units* (MHUs) for communication with other systems.
- 2 CMSDK Timers
- 2 CMSDK watchdogs:
  - Secure Enclave watchdog
  - SoC watchdog
- Secure Enclave Base System Control registers controlling the SSE-700 subsystem
- Secure Enclave System Control registers controlling the Secure Enclave
- Independent Clock and Power Control infrastructure
- *Security Control Bits* (SCB) controlling security access across the SSE-700 subsystem
- A Firewall that permits the Secure Enclave to access any location in the Host System address space

For more information about the Firewall used in Secure Enclave, see section *Secure Enclave Firewall* on page 2-21.

The following figure is a high-level block diagram of the components within the Secure Enclave.

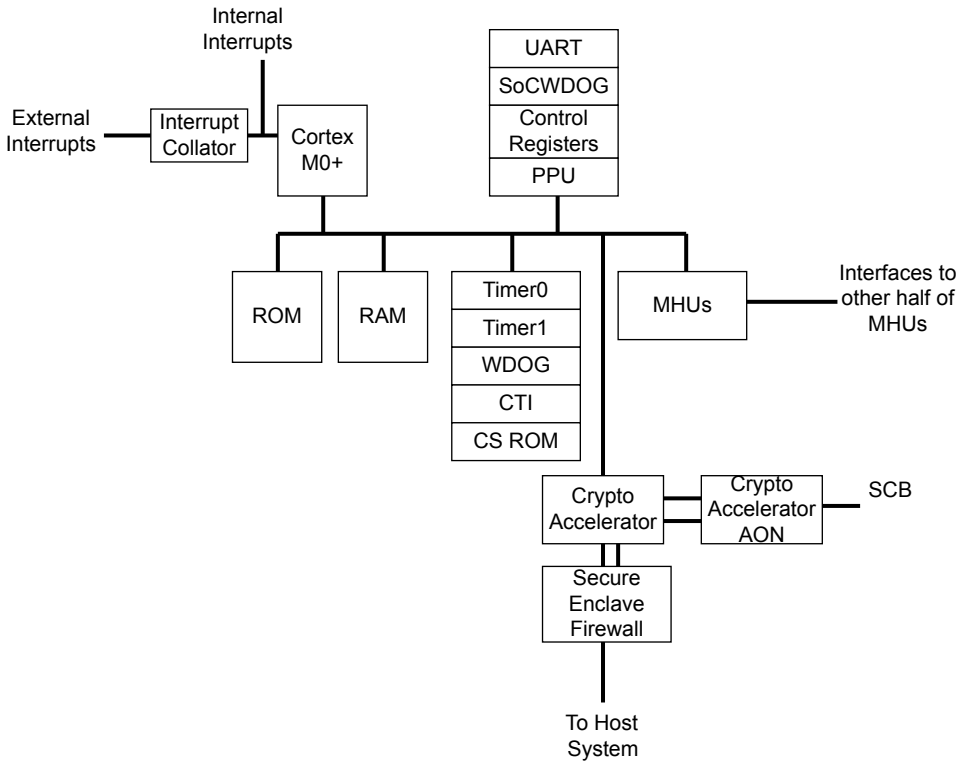

Figure 1-1 Secure Enclave block diagram

The integrator must add a Crypto Accelerator to the Secure Enclave. Therefore, the algorithms supported by the Crypto Accelerator are IMPLEMENTATION DEFINED.

To reduce complexity in security reviews, software running on the Secure Enclave is isolated by hardware. Communication between the Host System, External Systems and the Secure Enclave is done by MHUs. For more information about MHUs in the SSE-700 Secure Enclave, see *Secure Enclave MHUs* on page 2-21.

The Secure Enclave consists of two sections:

- An always-on section in the SSE-700 AONTOP power domain. This contains the following components:
  - SoC Watchdog
  - Secure Enclave UART
  - SECENCTOP PPU and PCSM
  - Crypto Accelerator Always-on
  - Secure Enclave System and Base System Control
- A switchable section in the SSE-700 SECENCTOP power domain. For more information see *2.4.1 SECENCTOP* on page 2-29.

3.2.1 Secure Enclave interrupt map on page 3-38 and *3.1.1 Secure Enclave memory map* on page 3-34 define the interrupt and address maps of the Secure Enclave.

The *3.3.1 Secure Enclave Base System Control register summary* on page 3-41 and *3.3.2 Secure Enclave System Control register summary* on page 3-52 define the registers in the Base System Control and Secure Enclave System Control blocks.

Chapter 2
Functional description

This chapter provides a functional description of the Secure Enclave.

It contains the following sections:

2.1 Secure Enclave components

This section describes Secure Enclave components.

This section contains the following subsections:

2.1.1 Cryptographic Accelerator

The Secure Enclave IMPLEMENTATION DEFINED Cryptographic Accelerator provides cryptographic services to the SSE-700. The Cryptographic Accelerator also provides lifecycle management and feature enablement based on the lifecycle state of the SoC.

2.1.2 Lifecycle States (LCS)

Using the IMPLEMENTATION DEFINED cryptographic engine, the Secure Enclave provides the *LifeCycle State* (LCS) of the SoC.

The lifecycle state is always one of the following:

Table 2-1 Lifecycle States

LCS	Description
Chip Manufacture	Initial state after manufacture
Device Manufacture	Used during device development
Secure Enable	Used when the device is deployed
Return Merchandise Authorization	Used when the device has reached end-of-life. At this point, it is not possible to boot any system other than the Secure Enclave.

The lifecycle runs in a linear direction from Chip Manufacture → Device Manufacture → Secure Enable → Return Merchandise Authorization.

To change the lifecycle state of the SoC, software executing on the Secure Enclave Cortex-M0+ must perform an IMPLEMENTATION DEFINED sequence.

The SSE-700 has the *Crypto Accelerator Lifecycle Control* (CALC) interface. This interface provides a physical signal to prevent advancement of the lifecycle state by accidental or malicious means. The CALC interface is `0b1` under one of the following conditions:

- *SoC Lifecycle Control* (SOCLCC) interface of SSE-700 is `0b1`. The Agent performing the sequence must assert the SOCLCC interface before releasing **PORESETn** to the SoC and keep it asserted until the transition is complete.
- GPO0 of the GPIO Control is `0b1`. The debug agent must follow these steps:
  - Use the DP to program the GPIO Control to assert the GPO0 output
  - Cause a debug reset by either:
    - Asserting the **nSRST** input
    - Setting the DP ROM CSYSRSTREQ register to `0b1` and wait for the DP ROM CSYSRSTACK to become `0b1`
  - Allow the Secure Enclave Cortex-M0+ to perform the software sequence by either:

- De-asserting the **nSRST** input
- Setting the DP ROM CSYSRSTREQ to `0b0` and waiting for the DP ROM CSYSRSTACK to become `0b0`

  Note

  The sequence either uses the **nSRST** input or the DP ROM CSYSRSTREQ/ACK handshake to cause the reset.

2.1.3 Security Control Bits (SCB)

The Secure Enclave uses the Security Control Bits (SCB) to control features within the SSE-700 subsystem.

The following table shows the bit assignment of the SCB interface as defined in the *2.2.4 Security Control Bits (SCB) interface* on page 2-26. Any Reserved bits are tied LOW.

For more information on features controlled by the SCB, except for the HOST_SYS_RST_CTRL register, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

Table 2-2 SCB Interface bit assignment

SCB offset	Name	Description
0	SECENCAUTH_DBGEN	Controls the SECENCAUTH DAZ
1	SECENCAUTH_NIDEN	
2-3	Reserved	-
4	SECENCAUTH_CHEN	Controls the Channel Gate in the SECENCAUTH DAZ
5	DPAUTH_DBGEN	Controls the DPAUTH DAZ
6	DPAUTH_NIDEN	
7	DPAUTH_SPIDEN	
8	DPAUTH_SPNIDEN	
9	COMAUTH_PEN	Controls the COMAUTH DAZ
10	COMAUTH_RRDIS	
11-12	Reserved	-
13	TPIUAUTH_DBGEN	Controls the TPIUAUTH DAZ
14	TPIUAUTH_NIDEN	
15	TPIUAUTH_SPIDEN	
16	TPIUAUTH_SPNIDEN	
17	TPIUATH_CHEN	Controls the Channel Gate in the TPIUAUTH DAZ

Table 2-2 SCB Interface bit assignment (continued)

SCB offset	Name	Description
18	COUNTERAUTH_DBGEN	Controls the COUNTERAUTH DAZ
19	COUNTERAUTH_NIDEN	
20	COUNTERAUTH_SPIDEN	
21	COUNTERAUTH_SPNIDEN	-
22	COUNTERAUTH_CHEN	Controls the Channel Gate in the COUNTERAUTH DAZ
23	HOSTEXTAUTH_NS	Controls HOSTEXTAUTH DAZ
24	HOSTEXTAUTH_S	
25	HOSTAXIAUTH_DBGEN	Controls HOSTAXIAUTH DAZ
26	HOSTAXIAUTH_NIDEN	
27	HOSTAXIAUTH_SPIDEN	
28	HOSTAXIAUTH_SPNIDEN	
29	HOSTAUTH_DBGEN	Controls the HOSTAUTH DAZ, and drives the HOSTCPUDBGAUTH and HOSTDBGAUTH interfaces.
30	HOSTAUTH_NIDEN	
31	HOSTAUTH_SPIDEN	
32	HOSTAUTH_SPNIDEN	
33	HOSTAUTH_CHEN	Controls the Channel Gate in the HOSTAUTH DAZ
34	SOC_DFTENABLE	Drives the **DFTENABLE[0]** signal of the SOCSC interface
35	PPU_DBGEN	Controls whether the PPU debug functionality is enabled
36	SECENC_FW_BYPASS	Drives the Bypass interfaces of the Secure Enclave Firewall
37	HOST_FW_BYPASS	Drives the Bypass interfaces of the Host System Firewall
38	DPEXTACG	Controls the DAACG on the DP port of the External Debug Bus
39	HOSTEXTACG	Controls the DAACG on the Host System port of the External Debug Bus
40	EXTSYS0EXTACG	Controls the DAACG on the External System 0 port of the External Debug Bus
41	EXTSYS1EXTACG	Controls the DAACG on the External System 1 port of the External Debug Bus
42-47	Reserved	-
48	HOST_CPUWAIT_WEN	Controls whether the HOST_SYS_RST_CTRL.CPUWAIT field is writeable
49	EXT_SYS0_CPUWAIT_WEN	Controls whether the EXT_SYS0_RST_CTRL.CPUWAIT field is writeable

Table 2-2 SCB Interface bit assignment (continued)

SCB offset	Name	Description
50	EXT_SYS1_CPUWAIT_WEN	Controls whether the EXT_SYS1_RST_CTRL.CPUWAIT field is writeable
51-62	Reserved	-
63	SECENC_DFTENABLE	Drives the **DFTENABLE[1]** signal of the SOCSC interface
64-127	SoC Expansion	Used for IMPLEMENTATION DEFINED use-cases within the SoC, for example, controlling the debug privileges of the External System

The SSE-700 subsystem defines the following rules for the SCB:

- The default values of an SCB are dependent on the lifecycle state of the SoC. Until the lifecycle state is known all SCB must have a value of `0b0`.
- The values of the SCBs must not change to the lifecycle dependent value until the lifecycle has been determined and the LCS interface is stable.
- In all lifecycle states the **COMAUTH – PEN** is `0b1` by default in all lifecycle states.
- In all lifecycle states the **COMAUTH – RRDIS** is IMPLEMENTATION DEFINED in all lifecycle states.
- In the Chip Manufacture lifecycle state:
  - The value of all SCBs, other than **COMAUTH** signals, is IMPLEMENTATION DEFINED.
  - It is IMPLEMENTATION DEFINED whether the SCB can be updated by either:
    - Software running on the Secure Enclave. Arm recommends that this is done as part of a certificate authentication.
    - Debug accesses using the Secure Enclave M0+
- In the Device Manufacture lifecycle state:
  - The value of all SCBs, other than **COMAUTH** signals, is IMPLEMENTATION DEFINED.
  - It is IMPLEMENTATION DEFINED whether the SCB can be updated by either:
    - Software running on the Secure Enclave. Arm recommends that this is done as part of a certificate authentication.
    - Debug accesses using the Secure Enclave M0+
- In the Secure Enable lifecycle state:
  - The values of all SCBs, other than **COMAUTH**, **HOST_CPUWAIT_WEN**, and **EXT_SYS{0-1}_CPUWAIT_WEN** signals, must be 0.
  - The **HOST_CPUWAIT_WEN** and **EXT_SYS{0-1}_CPUWAIT_WEN** signals must be 1.
  - The following SCBs must be updatable:
    - DPAUTH_{x}
    - TPIUAUTH_{x}
    - COUNTERAUTH_{x}
    - HOTEXTAUTH_{x}
    - HOSTAUTH_{x}
    - HOSTAXIAUTH_{x}
    - HOSTEXTACG
    - EXTSYS{0-1}EXTACG

    by either:
    - Software running on the Secure Enclave. Arm recommends that this is done as part of a certificate authentication.
    - Debug accesses using the Secure Enclave M0+

Note

Arm strongly recommends that both methods are implemented.

— For all other SCBs it is IMPLEMENTATION DEFINED whether the bits are updatable.

- In the Return Merchandise Authorization lifecycle state:
  — The values of all SCBs, other than **COMAUTH – PEN**, is IMPLEMENTATION DEFINED.
  — It is IMPLEMENTATION DEFINED whether the SCB can be updated by either:
    - Software running on the Secure Enclave. Arm recommends that this is done as part of a certificate authentication.
    - Debug access using the Secure Enclave M0+

Arm recommends:

- In the Chip Manufacture state, the minimum enabled features allow for a debug agent to be able to transition the SoC to the Device Manufacture state. This is achieved using the injection of certificate using the Arm CoreSight SDC-600 Secure Debug Channel, or having debug access enabled by default in the Secure Enclave. For more information see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.
- In the Device Manufacture state, the minimum enabled features allow for a debug agent to be able to debug the SoC. This can be by the injection of a certificate using the SDC-600, or having debug access enabled by default. For more information see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.
- Security requirements of the SoC in the lifecycle:
  — To prevent the Secure Enclave keys being extracted by an attacker when enabling debug, the keys must be destroyed on entry into the Return Merchandise Authorization state.
- Debuggability requirements of the SoC in the lifecycle:
  — Discovering the issues with returned devices is hampered if a debug agent is not enabled, or there is no way to re-enable debug in the Return Merchandise Authorization state.

2.1.4 Secure Enclave Cortex-M0+

The Secure Enclave contains a Cortex-M0+ processor, configured as follows:

- Little-endian data
- NVIC with support for 32 interrupts
- MPU with 8 regions
- 2 data watchpoints
- 4 breakpoints
- Halted debug support
- Privileged and Unprivileged support
- Systick Timer
- Vector Table Offset Register support
- Disables support for individual interrupts. The value is `0xEB5F_8010`.
- Architectural Clock Gating support

The SysTick timer of the Secure Enclave Cortex-M0+ uses the following clocks:

- **SECENCDIVCLK**
- **S32KCLK**

For the definitions of Clock Domains, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

2.1.5 Secure Enclave reset

The Secure Enclave can request a reset of itself, or other systems in the SoC, using any of the following mechanisms:

- A Secure Enclave or SoC watchdog reset request. When either of these events occur, the entire SoC is reset.
- A Secure Enclave software reset request triggered using the Cortex-M0+ AIRCR.SYSRESETREQ field. The entire SoC is reset, except for all logic on **AONTOPPORESETn**. For more information on Reset Domains, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*. For more information on Cortex-M0+ AIRCR.SYSRESETREQ, see the *Arm®v6-M Architecture Reference Manual*.
- Using the Secure Enclave *Base System Control registers* on page 3-41:
  — Initiate reset of the Host and External System, using the HOST_SYS_RST_CTRL.RST_REQ field.
  — Initiate reset of the SoC, using the SOC_RST_CTRL.RST_REQ field.

2.1.6 Secure Enclave peripherals

This section describes peripheral devices attached to the Secure Enclave.

ROM and RAM

The Secure Enclave has its own dedicated ROM and RAM.

For more information on the Secure Enclave ROM and RAM, see the *Secure Enclave ROM region* on page 3-35 and *Secure Enclave RAM region* on page 3-35.

Interrupt collator

The Secure Enclave has an interrupt collator enabling more than 32 interrupts to be handled by the Secure Enclave Cortex-M0+ processor.

It is controlled by the Secure Enclave Base System Control registers. For more information on these registers, see the *3.3.1 Secure Enclave Base System Control register summary* on page 3-41.

Timers

The Secure Enclave has two CMSDK timers, Timer 0 and 1. These are in the SECENCTOP power domain and use the **SECENCDIVCLK** clock.

For more information, see *2.3.3 SECENCDIVCLK* on page 2-27.

Both of the Secure Enclave CMSDK timers support being halted when the Secure Enclave Cortex-M0+ is halted by a debugger. The timer is halted when the Secure Enclave Cortex-M0+ is halted if software sets bit 1 of the CTRL register of the timer to `0b1`. For more information on the Timers see the *Arm® Cortex®-M System Design Kit Technical Reference Manual*.

Watchdogs

The Secure Enclave includes two CMSDK watchdogs:

- The Secure Enclave Watchdog is in the SECENCTOP power domain and uses **SECENCDIVCLK**
- The SoC Watchdog is in the AONTOP power domain and uses **S32KCLK**

  ——— **Note** ———

  An access to the SoC Watchdog takes 6-7 **S32KCLK** clock cycles. Accessing it too often might lead to software performance issues.

  ———————————

For more information on the CMSDK Watchdog, see the *Arm® Cortex®-M System Design Kit Technical Reference Manual*.

On the first expiry of the counter, both watchdogs generate interrupts to the Secure Enclave Cortex-M0+. On the second expiry, they generate a reset request to the Reset Controller of SSE-700 and cause a reset of the entire SoC. For more information, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

When the Cortex-M0+ is halted by a debugger, both Secure Enclave watchdogs are halted.

Secure Enclave MHUs

The MHUs between the Host System and the Secure Enclave are split across the SECENCTOP and SYSTOP power domains. The MHUs between the External Systems and the Secure Enclave are split across the EXTSYS{0-1}TOP and SECENCTOP power domain.

For more information on MHUs, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

System Control registers

The System Control registers are in the AONTOP power domain and control various aspects of the Secure Enclave.

For more information see the *3.3.2 Secure Enclave System Control register summary* on page 3-52.

Base System Control registers

The Base System Control registers are in the Always ON power domain. They control the features of the Host and External Systems in the integration of the Secure Enclave System into the SSE-700 subsystem.

For more information, see the *3.3.1 Secure Enclave Base System Control register summary* on page 3-41.

UART

The Secure Enclave has a PL011 UART located in the AONTOP power domain.

The Secure Enclave supports hardware-based flow control, and uses the **SECENCDIVCLK** for both the **PCLK** and **UARTCLK** inputs of PL011. Secure Enclave only uses the combined interrupt from the UART.

For more information on the PL011 UART see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*, and the *Arm® PrimeCell® UART (PL011) Technical Reference Manual*.

Note

The SSE-700 subsystem only supports the PL011 as an RS232-compliant UART.

2.2.3 Secure Enclave UART (SECENCUART) interface on page 2-25 describes the Secure Enclave UART interface, SECENCUART.

SECENCTOP PPU

The Secure Enclave has a *Power Policy Unit* (PPU) controlling the SECENCTOP power domain.

For more information on the PPU and SECENCTOP, see the *Arm® CoreLink™ PCK-600 Power Control Kit Technical Reference Manual* and *2.4.1 SECENCTOP* on page 2-29.

Secure Enclave Firewall

The Secure Enclave has a dedicated Firewall instance.

Note

The SSE-700 subsystem has two Firewalls. This section only describes the Secure Enclave Firewall. *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual* describes the Secure Enclave dedicated Firewall instance in more detail.

This Firewall is configured as follows:

- Lockdown Extension level 0
- Save and Restore Extension level 0
- Security Extension level 1

The Secure Enclave Firewall has the same Firewall interfaces as defined in the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*, except that it does not have Lockdown and Tamper Interrupt interfaces.

The Secure Enclave Firewall is in the SECENCTOP power domain and includes two Firewall Components (FC0 – FCTLR and FC1). FC1 is on the memory path to the Host System.

The Protection Size of the Secure Enclave Firewall Component 1 is set to 4GB. *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual* defines the Firewall IMPLEMENTATION DEFINED behavior. All of this behavior applies to the Secure Enclave Firewall.

The following table describes the Firewall Component configuration:

Table 2-3 Firewall component configuration

Firewall component	PE_LVL	ME_LVL	TE_LVL	RSE_LVL	NUM_RGN	MNRS	MXRS	NUM_MPE	SINGLE_MST
FCTLR	1	0	0	1	3	7	21	1	1
FC1	2	2	2	0	8	7	32	1	1

The following configurable values apply to all Firewall Components:

- MST_ID_WIDTH is 1 bit. All transactions have a fixed StreamID of 0 for the Secure Enclave Firewall.
- SEC_SPT is `0b1`
- MA_SPT is `0b1`
- SH_SPT is `0b0`
- INST_SPT is `0b1`
- PRIV_SPT is `0b1`

For Firewall Components that use PE.1, the following regions are predefined:

- Firewall Controller:
  - As the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual* Appendix defines.
  - The Configuration Master is set to the ID of the Secure Enclave.

When the Secure Enclave Firewall terminates a read transaction, or detects a read transaction marked as either an AMBA AXI5 SLVERR or DECERR, and the Firewall's monitor logic is enabled, the read data value is set to `0xDEAD_DEAD`.

For more information on the SSE-700 Subsystem Firewall, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual* Appendix.

2.2 Interfaces

A summary of the Secure Enclave Interfaces.

In the following sections the clock, power and reset domains of each interface are defined. This section must be read while referring to the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual* as required.

This section contains the following subsections:

2.2.1 Crypto Accelerator interfaces

The interfaces listed are used to integrate a Crypto Accelerator into the SSE-700 subsystem.

The SSE-700 subsystem Crypto Accelerator is split between the SECENCTOP and AONTOP power domains. These two components are:

- Crypto Accelerator (CA)
- Crypto Accelerator Always-on (CA AON)

Crypto Clock interface

The Crypto Clock interface **CRYPTOCLKOUT** is driven by the **SECENCCLK**. It is used only for the integration of the Crypto Accelerator into the Secure Enclave.

Crypto AON Clock Interface

The Crypto AON Clock interface **CRYPTOAONCLKOUT** is driven by **SECENCCLK**. It is used only for the integration of the Crypto Accelerator Always-On into the Secure Enclave.

Crypto Reset interface

The reset signal is active-LOW asynchronous assertion, synchronous de-assertion with respect to **CRYPTOCLKOUT**. The Crypto Reset interface **CRYPTORESETn** is driven by **SECENCWARMRESETn**.

Crypto AON Reset Interface

The Crypto Reset interface **CRYPTOAONRESETn** is driven by **SEPORESETn**. The reset signal is active-LOW asynchronous assertion, synchronous de-assertion with respect to **CRYPTOAONCLKOUT**.

Crypto Accelerator Master (CAM) interface

The Crypto Accelerator Master interface is a AHB master interface, with the following properties:

- AHB3
- 32-bit address
- 32-bit data
- **CRYPTOCLKOUT**
- SECENCTOP
- **CRYPTORESETn**

For more information on the CAM interface see the SSE-700 subsystem CIM.

Crypto Accelerator Slave (CAS) interface

The Crypto Accelerator slave interface must be connected to the AXI master interface, with the following properties:

- AXI4
- 32-bit address
- 32-bit data
- **CRYPTOCLKOUT**
- SECENCTOP
- **CRYPTORESETn**

——— **Note** ———

This interface provides access only to the Host Access Region of the Secure Enclave memory map.

———————————

Crypto Accelerator Firewall Configuration (CAFWCFG) interface

The Crypto Accelerator Firewall Configuration interface is an AXI Slave, with the following properties:

- AXI4
- 32-bit address
- 32-bit data
- **CRYPTOCLKOUT**
- SECENCTOP
- **CRYPTORESETn**

——— **Note** ———

This interface provides access only to the Secure Enclave Firewall region of the Secure Enclave memory map.

———————————

Crypto Accelerator AON (CAAON) interface

The Crypto Accelerator AON interface is used for communication between the two halves of the IMPLEMENTATION DEFINED Crypto Accelerator.

The Crypto Accelerator AON interface is made up of two multi-bit signals, **CA2CAAON** and **CAAON2CA**, allowing the two halves of the Crypto Accelerator to communicate. The signals are used for:

- Information sent from the Crypto Accelerator to Crypto Accelerator Always-on. (**CA2CAAON**).
- Information sent from the Crypto Accelerator Always-on to Crypto Accelerator. (**CAAON2CA**).

The properties of the interface differ for each signal:

- **CA2CAAON**:
  - Width equal to CAAON2CA_WIDTH
  - **CRYPTOCLKOUT**
  - **CRYPTORESETn**
- **CAAON2CA**:

— Width equal to CA2CAAON_WIDTH
— **CRYPTOAONCLKOUT**
— **CRYPTOAONRESETn**

These signals are used only for the integration of the Crypto Accelerator into the SSE-700 subsystem.

———— **Note** ————

There are two instances of the CAAON interface crossing the power domain. In one instance the **CA2CAAON** and **CAAON2CA** signals are input and output respectively. In the other, the **CA2CAAON** and **CAAON2CA** signals are output and input respectively.

————————————

Crypto Accelerator lifecycle Control (CALC) interface

The Crypto Accelerator lifecycle Control interface is used to control the advance of the lifecycle of the SoC, with the following properties:

- Asynchronous
- AONTOP

Crypto Accelerator Interrupt (CAINT) interface

The Crypto Accelerator Interrupt interface connects the interrupts from the Crypto Accelerator to the Cortex-M0+ NVIC, with the following properties:

- 2 interrupts
  — any unused interrupts must be tied LOW
- **CRYPTOAONCLKOUT**
- AONTOP
- **CRYPTOAONRESETn**

Crypto Accelerator Power P-Channel (CAPWRP) interface

The Crypto Accelerator Power P-Channel interface is used to control the power mode of the Crypto Accelerato, with the following properties:

- P-Channel interface with 4 PSTATE bits and 11 PACTIVE bits. The encoding of the **PSTATE** and **PACTIVE** signals follow the recommended usage defined in the *Arm® Power Control System Architecture.*
- Asynchronous
- AONTOP
- **CRYPTOAONRESETn**

2.2.2 Debug interface

The Secure Enclave has an APB4 and Cross Trigger Channel interface. This connects the internal debug logic of the Secure Enclave's debug logic with the debug logic of the SSE-700 Subsystem.

For more information about the SSE-700 debug infrastructure, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

2.2.3 Secure Enclave UART (SECENCUART) interface

The Secure Enclave UART has a standard interface with flow control.

The following table shows the SECENCUART interface signals:

Table 2-4 SECENCUART interface

Signal	Description	I/O
SECENCUARTTX	UART Transmit data	O
SECENCUARTRX	UART Receive data	I
SECENCUARTRTSn	UART Request to Send	O
SECENCUARTCTSn	UART Clear to Send	I
SECENCUARTRIn	UART Ring Indicator	I
SECENCUARTDCDn	UART Data Carrier Detect	I
SECENCUARTDSRn	UART Data Set Ready	I
SECENCUARTDTRn	UART Data Terminal Ready	O
SECENCUARTOUT1n	UART Out1	O
SECENCUARTOUT2n	UART Out2	O

The SECENCUART interface has the following properties:

- Asynchronous
- AONTOP
- **SEPORESETn**

For the definitions of Clock, Power and Reset Domains, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

2.2.4 Security Control Bits (SCB) interface

The Security Control Bits provide an input interface to control security features within the SSE-700 subsystem.

For more details on the SCB see *2.1.3 Security Control Bits (SCB)* on page 2-16.

This interface has the following properties:

- Multi-bit signal, as defined in *2.1.3 Security Control Bits (SCB)* on page 2-16
- Asynchronous
- AONTOP
- **CRYPTOAONRESETn**

2.3 Clocks

For information on **S32KCLK** used by designs in both SSE-700 Host System and Secure Enclave, see **S32KCLK** in the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

This section contains the following subsections:

2.3.1 SECENCREFCLK

SECENCREFCLK is the reference clock for the Secure Enclave. It is only used to generate the clocks of the Secure Enclave.

2.3.2 SECENCCLK

The following table summarizes **SECENCCLK**.

Table 2-5 SECENCCLK

Clock name	**SECENCCLK**		
Sources	Name	Default	Divider Support
	SECENCREFCLK	Yes	No
	SYSPLL	No	Yes
High-level clock gating support	No		
Components	Crypto Accelerator Secure Enclave Firewall (FCTLR and FC1)		

Inside the Secure Enclave, there are several gated versions of **SECENCCLK** to reduce dynamic power.

SECENCCLK is provided for integration of the Crypto Accelerator into SSE-700 using **CRYPTOCLKOUT** and **CRYPTOAONCLKOUT**. They cannot be used for any other logic.

2.3.3 SECENCDIVCLK

SECENCDIVCLK is an integer divided version of **SECENCCLK**. It can be configured to be either 1:1 or 1:2, depending on the programming of the Clock Divider Control register in the Secure Enclave System Control Unit.

The follwing table summarizes **SECENCDIVCLK**.

Table 2-6 SECENCDIVCLK

Clock name	**SECENCDIVCLK**		
Sources	Name	Default	Divider Support
	SECENCCLK	Yes	Yes

Table 2-6 SECENCDIVCLK (continued)

Clock name	SECENCDIVCLK
High-level clock gating support	No
Components	Secure Enclave Cortex-M0+ Secure Enclave RAM Secure Enclave ROM Secure Enclave Watchdog Secure Enclave Timer 0 and 1 Sender frames of the SEH{0-1} and SEES{0-1}{0-1} MHUs Receiver frame of the HSE{0-1} and ES{0-1}SE{0-1} MHUs Secure Enclave System Control Secure Enclave Base System Control Secure Enclave UART Secure Enclave AHB AP Secure Enclave CS ROM Secure Enclave CTI Channel Gate for SECENCAUTH SECENC PPU

2.4 Power

This section describes the Secure Enclave power control and sleep states.

This section contains the following subsections:

2.4.1 SECENCTOP

This section describes the SECENCTOP power domain.

Note

You must also read the Power section of the *Arm® Corstone SSE-700 Subsystem Technical Reference Manual*.

The SECENCTOP power domain supports the following power modes:

- ON: All logic and Secure Enclave RAM is powered
- MEM_RET: Secure Enclave RAM is in a retention state, all other logic is powered off
- OFF: All logic, including the RAM is powered off

Note

WARM_RST is supported by the SECENCTOP PPU, but Arm strongly recommends that software never directs the PPU to WARM_RST.

The SECENCTOP power domain contains the following components:

- Secure Enclave Cortex-M0+
- Secure RAM and ROM
- Secure Enclave CMSDK Timer 0 and 1
- Secure Enclave Watchdog
- Sender frame of the SEH{0-1} and SEES{0-1}{0-1} MHUs
- Receiver frame of the HSE{0-1} and ES{0-1}SE{0-1} MHUs
- Secure Enclave Firewall
- Crypto Accelerator
- Secure Enclave AHB AP
- Secure Enclave CS ROM
- Secure Enclave CTI
- Channel Gate for SECENCAUTH DAZ

The SECENCTOP domain has the following power modes:

OFF power mode

- Any debug access to the Cortex-M0+ generates an error.
- A CTI event to or from the Secure Enclave CTI is ignored and does not cause a change in the power mode.

MEM_RET power mode

- Any debug access to the Cortex-M0+ generates an error.
- A CTI event to or from the Secure Enclave CTI is ignored and does not cause a change in the power mode.

ON power mode

- Any debug access to the Cortex-M0+ is allowed.
- A CTI event to or from the Secure Enclave CTI is allowed.
- Any access to the Secure Enclave RAM is allowed.

WARM_RST power mode

- Software must never request entry into the WARM_RST power mode because it is unable to cause the SECENCTOP power domain to exit the WARM_RST power mode.
- Any debug access to the Cortex-M0+ generates an error.
- A CTI event to or from the Secure Enclave CTI is ignored and does not cause a change in the power mode.

Note

Software is responsible for guaranteeing that the Crypto Accelerator is idle before entering into any power mode other than ON.

2.4.2 Secure Enclave sleep states

The Secure Enclave has the following sleep states:

- SLEEPING:
  - The clock is gated to the following components:
    - Secure Enclave Cortex-M0+, excluding the NVIC
    - Secure Enclave RAM and ROM
  - The conditions for entering the SLEEPING state are:
    - The SCR.SLEEPDEEP bit set to `0b0`
    - Secure Enclave software executes a WFE or WFI instruction
    - **EXTDBG ROM CDBGPWRUPREQ0** is `0b0`
  - Exit from the SLEEPING state occurs when any of the following occur:
    - An interrupt to the Secure Enclave Cortex-M0+ is detected
    - **EXTDBG ROM CDBGPWRUPREQ0** is `0b1`
  - The Secure Enclave can enter the SLEEPING state in any SSE-700 subsystem power state other than BSYS.OFF.
- SLEEPDEEP:
  - The SLEEPDEEP sleep state is the same as the SLEEPING state, except that the **SCR.SLEEPDEEP** bit is set to `0b1`. Otherwise all conditions for entering and exiting from SLEEPDEEP state are the same as SLEEPING.

    Note

    For more information on:
    - SCR register, see the *Arm® Armv6-M Architecture Reference Manual*
    - EXTDBG ROM CDBGPWRUPREQ, see the *Arm® SSE-700 Subsystem Technical Reference Manual*

- SLEEPDEEP PG:
  - In the SLEEPDEEP PG state all components of the Secure Enclave in the SECENCTOP domain lose their context.
  - Entry into SLEEPDEEP PG is by the SLEEPDEEP state.
  - Secure Enclave software must carry out the following before entering the SLEEPDEEP PG state:
    - Disable the Secure Enclave Timer {0,1} and Watchdog
    - Guarantee that Secure Enclave is idle, including Crypto Accelerator
  - Conditions for entering the SLEEPDEEP PG state are:
    - All the conditions for SLEEPDEEP state are met
    - **SE_PWR_CTRL.PWR_GATE_EN** is `0b1`
    - **SECENCTOP PPU** is in the OFF or MEM_RET power mode
  - Exit from SLEEPDEEP PG state occurs when any of the following occur:
    - The Secure Enclave Base System Control **BSYS_PWR_REQ.WAKEUP_EN** is `0b1` and an interrupt is detected from one of the following sources:

- SoC Watchdog
- Host System Firewall Tamper interrupt
- Interrupt Router Tamper interrupt
- Secure Watchdog WS1
- SECENCTOP PPU
- Secure Enclave UART UARTINTR

- The Secure Enclave Base System Control **BSYS_PWR_REQ.WAKEUP_EN** is `0b1` and an interrupt is detected from one of the following sources and the interrupt is unmasked:
  - Any interrupt routed to the Secure Enclave from the Interrupt Router

    ——— **Note** ———

    By masking the interrupt in the interrupt collator, software can disable wakeup from any interrupt.

    ———————————

  - **EXTDBG ROM CDBGPWRUPREQ0** is `0b1`.
  - Any condition which causes the SECENCTOP PPU to exit the OFF or MEM_RET power mode.

— The Secure Enclave can enter the SLEEPDEEP PG state in any SSE-700 subsystem power state.

——— **Note** ———

For entry into the BSYS.OFF power state, the SECENCTOP power domain must be in OFF.

———————————

2.5 Reset

You must read this section with the reset section of the *Arm® Corstone SSE-700 Subsystem Technical Reference Manual*.

This section contains the following subsections:

2.5.1 SEPORESETn

The **SEPORESETn** reset domain contains the following logic:

- SoC Watchdog
- Secure Enclave UART
- Secure Enclave System and Base System Control registers
- Crypto Accelerator Always-on and OTP
- SECENCTOP PPU and PCSM

2.5.2 SECENCPORESETn

The **SECENCPORESETn** reset domain contains the following logic:

- Debug logic within the Cortex-M0+
- Secure Enclave AHB AP and CS ROM table
- Secure Enclave CTI
- Channel Gate for SECENCAUTH DAZ

2.5.3 SECENCWARMRESETn

The **SECENCWARMRESETn** reset domain contains the following logic:

- Non-debug logic within the Cortex-M0+
- Secure Enclave Timers
- Secure Enclave Watchdog
- Sender halves of SEH{0,1} and SEES{0-1}{0-1} MHUs
- Receiver halves of HSE{0,1} and ES{0-1}SE{0-1} MHUs
- Crypto Accelerator

Chapter 3
Programmers model

This chapter provides general information about the Secure Enclave memory map and registers, and information for programming the device.

It contains the following sections:

3.1 Memory map

This section gives an overview of the Secure Enclave Memory Map regions.

This section contains the following subsection:

3.1.1 Secure Enclave memory map

The Secure Enclave has a 32-bit address space.

All unallocated memory regions are Reserved. Access to Reserved regions results in a RAZ/WI and a bus error response, except for the Reserved locations defined by the Cortex-M0+ processor.

The following table shows the Secure Enclave memory map.

Table 3-1 Memory Map

Base address	Size	Region	Notes
0x0000_0000	32KB	Secure Enclave ROM Region	Provides access to the Secure Enclave ROM. See *Secure Enclave ROM Region* on page 3-35.
	96KB	Reserved	-
0x0002_0000	896KB	Reserved	-
0x0010_0000	765MB		
0x2F00_0000	16MB	Crypto Accelerator	Exposed on the CAM interface. For more information on the CAM interface, see *Crypto Accelerator Master (CAM) interface* on page 2-23.
0x3000_0000	128KB	Secure Enclave RAM Region	Secure Enclave RAM. See section *Secure Enclave RAM region* on page 3-35.
	896KB	Reserved	-
0x3010_0000	511MB	Reserved	-
0x5000_0000	256MB	Secure Enclave Peripheral Region	See *Secure Enclave Peripheral region* on page 3-35
0x6000_0000	2GB	Host Access Region	See *Host Access region* on page 3-36
0xE000_0000	1MB	Private Peripheral Bus	See *Private Peripheral Bus (PPB)* on page 3-36
0xE010_0000	255MB	Reserved	-
0xF000_0000	4KB	Secure Enclave CS ROM	-
0xF000_1000	4KB	Secure Enclave CTI	-
0xF000_2000	1016KB	Reserved	-
0xF010_0000	255MB	Reserved	-

Note

The Secure Enclave Memory Map applies to the Cortex-M0+. The CAS interface only has visibility of the Host Access Region.

Secure Enclave ROM region

The Secure Enclave ROM provides a 32KB area of memory starting at address `0x0000_0000`. This is where the Secure Enclave's Processor core boots from.

Secure Enclave RAM region

The Secure Enclave RAM region provides a 128KB area for accessing private on-chip RAM. This starts at address `0x3000_0000`.

Secure Enclave Peripheral region

The Secure Enclave Peripheral region provides an area for accessing all internal Secure Enclave peripherals. The following table shows the memory map:

Table 3-2 Secure Enclave Peripheral region

Base address	Size	Region	Notes
`0x5000_0000`	4KB	Timer 0	CMSDK Timers. For more information, see the *Arm® Cortex®-M System Design Kit Technical Reference Manual*.
`0x5000_1000`	4KB	Timer 1	
`0x5000_2000`	4KB	Reserved	-
`0x5000_3000`	4KB	SEH 0 Sender	MHUs between Secure Enclave and Host System. For more details of these MHUs, see *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.
`0x5000_4000`	4KB	HSE 0 Receiver	
`0x5000_5000`	4KB	SEH 1 Sender	
`0x5000_6000`	4KB	HSE 1 Receiver	
`0x5000_7000`	36KB	Reserved	-
`0x5001_0000`	4KB	SEES0 0 Sender	MHUs between Secure Enclave and Host System. For more details of these MHUs, see *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.
`0x5001_1000`	4KB	ES0SE 0 Receiver	
`0x5001_2000`	4KB	SEES0 1 Sender	
`0x5001_3000`	4KB	ES0SE 1 Receiver	
`0x5001_4000`	4KB	SEES1 0 Sender	
`0x5001_5000`	4KB	ES1SE 0 Receiver	
`0x5001_6000`	4KB	SEES1 1 Sender	
`0x5001_7000`	4KB	ES1SE 1 Receiver	
`0x5001_8000`	32KB	Reserved	-
`0x5002_0000`	96KB	Reserved	-

Table 3-2 Secure Enclave Peripheral region (continued)

Base address	Size	Region	Notes
0x5008_0000	4KB	Secure Enclave System Control Register	Secure Enclave System Control Register. See section *3.3.2 Secure Enclave System Control register summary* on page 3-52
0x5008_1000	4KB	Watchdog Timer	CSMDK Watchdog. Refer to the *Arm® Cortex®-M System Design Kit Technical Reference Manual*.
0x5008_2000	44KB	Reserved	-
0x5008_D000	4KB	SECENCTOP PPU	See *SECENCTOP PPU* on page 2-21
0x5008_E000	4KB	Secure Enclave Base System Control Register	See *3.3.1 Secure Enclave Base System Control register summary* on page 3-41
0x5008_F000	4KB	SoC Watchdog	See the *Arm® Cortex®-M System Design Kit Technical Reference Manual*
0x5009_0000	4KB	UART	See the *Arm® PrimeCell® UART (PL011) Technical Reference Manual*
0x5009_1000	1084KB	Reserved	-
0x5020_0000	2MB	Secure Enclave Firewall	See *Secure Enclave Firewall* on page 2-21
0x5040_0000	252MB	Reserved	-

Host Access region

The Host Access Region is a 2GB area starting at 0x6000_0000. It gives the Secure Enclave access to the Host System address space.

All access passes through FC1 of the Secure Enclave Firewall. The mapping between the Secure Enclave and Host System address space is controlled using the regions within the Secure Enclave Firewall and the Translation Extension. For more information on Firewall translation, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

Note

The security of access to the Host System address space is controlled by the Firewall translation extension.

Cortex®-M0+ Private Peripheral Bus (PPB) region

The Private Peripheral Bus region is available only to the Cortex-M0+ core. For more details, see the *Arm®v6-M Architecture Reference Manual*.

Timers and Watchdog timers

Timers 0 and 1 and the Secure Enclave Watchdog timer are located in the SECENCTOP power domain.

The SoC Watchdog resides within the AONTOP domain of the Secure Enclave. For more information on these timers, see the *Arm® Cortex®-M System Design Kit Technical Reference Manual*.

MHUs

The Secure Enclave supports six pairs of MHUs. Each pair provides full-duplex communication between the Secure Enclave and either the Host System or an External System.

For more information on the MHUs, see the relevant section in the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.

3.2 Interrrupt map

This section gives an overview of the Secure Enclave Interrupt Map and Interrupt Expansion.

This section contains the following subsections:

3.2.1 Secure Enclave interrupt map

The following table summarises the Secure Enclave interrupt map.

Table 3-3 Interrupt Map

Interrupt number	Interrupt source	Level/edge	Notes
NMI	SoC Watchdog Timer	Level	-
0	Secure Enclave Interrupt Expansion	Level	See *3.2.2 Secure Enclave interrupt expansion* on page 3-39
1	Crypto Accelerator Interrupt 0	Level	Any unused interrupts are Reserved and must be tied LOW
2	Crypto Accelerator Interrupt 1	Level	
3	Secure Enclave Watchdog Timer	Level	-
4	Reserved	-	-
5	CMSDK Timer 0	Level	-
6	CMSDK Timer 1	Level	-
7	Host System Firewall Tamper Interrupt	Level	-
8	Interrupt Router Tamper Interrupt	Level	-
9	Secure Watchdog WS1	Level	-
10	SECENCTOP PPU	Level	-
11	UART UARTINTR	Level	-
12	Secure Enclave Firewall Interrupt	Level	-
13	Secure Enclave CTI Trigger Out 2	Level	-
14	Secure Enclave CTI Trigger Out 3	Level	-
15-20	Reserved	-	-
21	SEH0 MHU Sender Combined interrupt	Level	-
22	Reserved	-	-
23	HSE0 MHU Receiver Combined interrupt	Level	-
24	Reserved	-	-

Table 3-3 Interrupt Map (continued)

Interrupt number	Interrupt source	Level/edge	Notes
25	Reserved	-	-
26	SEH1 MHU Sender Combined interrupt	Level	-
27	Reserved	-	-
28	HSE1 MHU Receiver Combined interrupt	Level	-
29-31	Reserved	-	-

3.2.2 Secure Enclave interrupt expansion

The Secure Enclave interrupt collator provides 128 interrupts in addition to the interrupts provided by the Cortex-M0+ NVIC.

The output drives interrupt 0 in the Secure Enclave Cortex-M0+. If an interrupt is unmasked and Secure Enclave Base System Control **BSYS_PWR_REQ.WAKEUP_EN** is HIGH, these interrupts force the Secure Enclave to exit the OFF or MEM_RET power modes.

Software uses the SEC_ENC_INT_COL_ST{0-3} and SEC_ENC_INT_COL_MSK{0-3} registers to monitor the status, and mask the interrupt respectively. All interrupts routed to the Secure Enclave must be level-based as there is no logic to capture which source caused the interrupt. The Secure Enclave Expansion interrupts are referred to as SEEI{x}, where x is the interrupt number starting from 0.

The following table summarizes the Secure Enclave interrupt expansion.

Table 3-4 Secure Enclave interrupt expansion

Interrupt number (SEEI{x})	Interrupt source	Level/ edge	Notes
0	Host System Firewall Interrupt	Level	All interrupt sources are routed through the Interrupt Router. For more information about the Interrupt Router, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.
1	SDC-600	Level	
2	Host PPU Combined Interrupt	Level	
3	**REFCLK** Timer 0	Level	
4	**REFCLK** Timer 1	Level	
5	**REFCLK** Timer 2	Level	
6	**REFCLK** Timer 3	Level	
7	**S32K** Timer 0	Level	
8	**S32K** Timer 1	Level	
9-31	Reserved	-	-

Table 3-4 Secure Enclave interrupt expansion (continued)

Interrupt number (SEEI{x})	Interrupt source	Level/ edge	Notes
32-63	Shared Interrupt 32-63	Level	All interrupt sources are routed through the Interrupt Router. The source of the interrupt is IMPLEMENTATION DEFINED.
64	SEES00 MHU Sender Combined Interrupt	Level	For more information about the MHU interrupt, see the *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*.
65	ES0SE0 MHU Receiver Combined Interrupt	Level	
66	SEES01 MHU Sender Combined Interrupt	Level	
67	ES0SE1 MHU Receiver Combined Interrupt	Level	
68	SEES10 MHU Sender Combined Interrupt	Level	
69	ES1SE0 MHU Receiver Combined Interrupt	Level	
70	SEES11 MHU Sender Combined Interrupt	Level	
71	ES1SE1 MHU Receiver Combined Interrupt	Level	
72-127	Reserved	-	-

3.3 Registers

This section summarizes the Secure Enclave Base and System Control registers.

This section contains the following subsections:

- *3.3.1 Secure Enclave Base System Control register summary* on page 3-41.
- *3.3.2 Secure Enclave System Control register summary* on page 3-52.

3.3.1 Secure Enclave Base System Control register summary

This section summarizes the Secure Enclave Base System Control registers.

The Secure Enclave Base System Control registers are in the Secure Enclave Peripheral Map. They are only accessible by the Secure Enclave Cortex-M0+ processor. The Secure Enclave Base System Control registers are accessed at offset `0x5008_E000` in the Secure Enclave memory map. These registers are in the Always ON power domain.

Table 3-5 Secure Enclave Base System Control register summary

Offset	Short name	Access	Name
`0x000`	*HOST_SYS_RST_CTRL* on page 3-42	RW	Host System Reset Control
`0x004`	*HOST_SYS_RST_ST* on page 3-43	RO	Host System Reset Status
`0x008`	*SOC_RST_CTRL* on page 3-43	RW	SoC Reset Control
`0x00C`	*SOC_RST_SYN* on page 3-44	RO	SoC Reset Syndrome
`0x010`	*SEC_ENC_INT_COL_ST0* on page 3-44	RO	Secure Enclave Interrupt Collator Status 0
`0x014`	*SEC_ENC_INT_COL_ST1* on page 3-44	RO	Secure Enclave Interrupt Collator Status 1
`0x018`	*SEC_ENC_INT_COL_ST2* on page 3-44	RO	Secure Enclave Interrupt Collator Status 2
`0x01C`	*SEC_ENC_INT_COL_ST3* on page 3-45	RO	Secure Enclave Interrupt Collator Status 3
`0x020`	*SEC_ENC_INT_COL_MSK0* on page 3-45	RW	Secure Enclave Interrupt Collator Mask 0
`0x024`	*SEC_ENC_INT_COL_MSK1* on page 3-45	RW	Secure Enclave Interrupt Collator Mask 1
`0x028`	*SEC_ENC_INT_COL_MSK2* on page 3-46	RW	Secure Enclave Interrupt Collator Mask 2
`0x02C`	*SEC_ENC_INT_COL_MSK3* on page 3-46	RO	Secure Enclave Interrupt Collator Mask 3
`0x030` – `0x3FC`	-	RO	Reserved
`0x400`	*BSYS_PWR_REQ* on page 3-46	RW	Base System Power Request
`0x404`	*BSYS_PWR_ST* on page 3-47	RO	Base System Power Status
`0x408` – `0x7FC`	-	RO	Reserved
`0x800`	*SECENCCLK_CTRL* on page 3-48	RW	**SECENCCLK** Control
`0x804`	*SECENCCLK_DIV* on page 3-48	RW	**SECENCCLK** Divider
`0x808` – `0x9FC`	-	RO	Reserved

Table 3-5 Secure Enclave Base System Control register summary (continued)

Offset	Short name	Access	Name
0xA00	*CLKFORCE_ST* on page 3-49	RO	Clock Force Status
0xA04	*CLKFORCE_SET* on page 3-49	WO	Clock Force Set
0xA08	*CLKFORCE_CLR* on page 3-49	WO	Clock Force Clear
0xA10	*SEC_ENC_PLL_ST* on page 3-50	RO	PLL Status
0xA14 – 0xFCC	-	RO	Reserved
0xFD0	*PID4* on page 3-50	RO	Peripheral ID4
0xFD4	*PID5* on page 3-50	RO	Peripheral ID5
0xFD8	*PID6* on page 3-50	RO	Peripheral ID6
0xFDC	*PID7* on page 3-51	RO	Peripheral ID7
0xFE0	*PID0* on page 3-51	RO	Peripheral ID0
0xFE4	*PID1* on page 3-51	RO	Peripheral ID1
0xFE8	*PID2* on page 3-51	RO	Peripheral ID2
0xFEC	*PID3* on page 3-51	RO	Peripheral ID3
0xFF0	*CID0* on page 3-52	RO	Component ID0
0xFF4	*CID1* on page 3-52	RO	Component ID1
0xFF8	*CID2* on page 3-52	RO	Component ID2
0xFFC	*CID3* on page 3-52	RO	Component ID3

The Secure Enclave Base System Control registers support only 32-bit word aligned accesses. Access by other bit sizes or unaligned accesses are treated as 32-bit word aligned access without generating an error response.

The Secure Enclave Base System Control registers have the following behavior:

- Any read to a write-only register is treated as RAZ.
- Any write to a read-only register is treated as WI.

In both cases no error is generated.

Host System Reset Control (HOST_SYS_RST_CTRL) register

The following table gives a bit-level description of the HOST_SYS_RST_CTRL register.

Table 3-6 HOST_SYS_RST_CTRL register

Bits	Name	Description	Type	Reset
[31:2]	-	Reserved	RO	`0x0000_0000`
[1]	RST_REQ	Reset request for Host System `0b0`: No reset requested `0b1`: Reset requested	RW	`0b0`
[0]	CPUWAIT	CPU Wait control `0b0`: Host System's CPUWAIT signal is de-asserted. `0b1`: Host System's CPUWAIT signal is asserted. When CPUWAIT becomes `0b0` attempts to set it back to `0b1` are ignored. This field only returns to `0b1` when any of the following occur: • External Power on Reset • Internal Power on Reset • Debug reset • Host System reset A request to set this field `0b0` at the same time as the field is to be set to `0b1` results in the field being set to `0b1`. This field becomes RO when HOST_CPUWAIT_WEN is LOW. Any attempt to set this field to `0b0` by software is ignored.	RW	`0b1`

Host System Reset Status (HOST_SYS_RST_ST) register

The following table gives a bit-level description of the HOST_SYS_RST_ST register.

Table 3-7 HOST_SYS_RST_ST register

Bits	Name	Description	Type	Reset
[31:3]	-	Reserved	RO	`0x00_0000`
[2:1]	RST_ACK	Status of reset request `0b00` – No reset requested `0b01` – Reset request unable to complete `0b10` – Reset request complete `0b11` – Reserved	RO	`0b00`
[0]	-	Reserved	RO	`0b0`

SoC Reset Control (SOC_RST_CTRL) register

The following table gives a bit-level description of the SOC_RST_CTRL register.

Table 3-8 SOC_RST_CTRL register

Bits	Name	Description	Type	Reset
[31:2]	-	Reserved	RO	`0x0000_0000`
[1]	RST_REQ	Reset request for SoC. `0b0`: No reset requested `0b1`: Reset requested	RW	`0b0`
[0]	-	Reserved	RO	`0b0`

SoC Reset Syndrome (SOC_RST_SYN) register

The following table gives a bit-level description of the SOC_RST_SYN register.

Table 3-9 SOC_RST_SYN register

Bits	Name	Description	Type	Reset
[31]	SEC_ENC	Indicates the source of the last reset is captured in the SE_RST_SYN register in the Secure Enclave System Control registers	RO	UNKNOWN
[30:4]	-	Reserved	RO	`0x000_0000`
[3]	SOC_WDOG	Indicates the last reset of the SoC was caused by the SoC Watchdog	RO	UNKNOWN
[2]	-	Reserved	RO	`0x000_0000`
[1]	nSRST	Indicates that the last reset of the SoC was caused by either: • **nSRST** pin being asserted • DP ROM **CSYSRSTREQ** being asserted	RO	UNKNOWN
[0]	POR	Indicates that the last reset of the SoC was caused by either: • **PORESETn** pin being asserted • DP **CDBGRSTREQ** being asserted • SOC_RST_CTRL.RST_REQ bit set to `0b1`	RO	UNKNOWN

Secure Enclave Interrupt Collator Status 0 (SEC_ENC_INT_COL_ST0) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_ST0 register.

Table 3-10 SEC_ENC_INT_COL_ST0 register

Bits	Name	Description	Type	Reset
[31:0]	SSI_ST	Status of the Secure Enclave Expansion Interrupt (SEEI) {0-31} after the respective mask field from SEC_ENC_INT_COL_MSK0 register has been applied. SEEI0 assigned to bit[0]. `0b0`: Interrupt is de-asserted `0b1`: Interrupt is asserted	RO	UNKNOWN

Secure Enclave Interrupt Collator Status 1 (SEC_ENC_INT_COL_ST1) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_ST1 register.

Table 3-11 SEC_ENC_INT_COL_ST1 register

Bits	Name	Description	Type	Reset
[31:0]	SSI_ST	Status of the Secure Enclave Expansion Interrupt (SEEI) {32-63} after the respective mask field from SEC_ENC_INT_COL_MSK1 register has been applied. SEEI32 assigned to bit[0]: `0b0`: Interrupt is de-asserted `0b1`: Interrupt is asserted	RO	UNKNOWN

Secure Enclave Interrupt Collator Status 2 (SEC_ENC_INT_COL_ST2) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_ST2 register.

Table 3-12 SEC_ENC_INT_COL_ST2 register

Bits	Name	Description	Type	Reset
[31:0]	SSI_ST	Status of the Secure Enclave Expansion Interrupt (SEEI) {64-95} after the respective mask field from SEC_ENC_INT_COL_MSK2 register has been applied. SEEI64 assigned to bit[0]: `0b0`: Interrupt is de-asserted `0b1`: Interrupt is asserted	RO	UNKNOWN

Secure Enclave Interrupt Collator Status 3 (SEC_ENC_INT_COL_ST3) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_ST3 register.

Table 3-13 SEC_ENC_INT_COL_ST3 Register

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	`0x0000_0000`

Secure Enclave Interrupt Collator Mask 0 (SEC_ENC_INT_COL_MSK0) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_MSK0 register.

Table 3-14 SEC_ENC_INT_COL_MSK0 register

Bits	Name	Description	Type	Reset
[31:0]	SSI_MSK	Configures whether Secure Enclave Expansion Interrupt (SEEI) {0-31} generates an interrupt to the Secure Enclave Cortex-M0+ core, with SEEI0 assigned to bit[0]: `0b0`: Interrupt is unmasked `0b1`: Interrupt is masked	RW	`0x0000_0000`

Secure Enclave Interrupt Collator Mask 1 (SEC_ENC_INT_COL_MSK1) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_MSK1 register.

Table 3-15 SEC_ENC_INT_COL_MSK1 register

Bits	Name	Description	Type	Reset
[31:0]	SSI_MSK	Configures whether Secure Enclave Expansion Interrupt (SEEI) {32-63} generates an interrupt to the Secure Enclave Cortex-M0+ core, with SEEI32 assigned to bit[0]: `0b0`: Interrupt is unmasked `0b1`: Interrupt is masked	RW	`0x0000_0000`

Secure Enclave Interrupt Collator Mask 2 (SEC_ENC_INT_COL_MSK2) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_MSK2 register.

Table 3-16 SEC_ENC_INT_COL_MSK2 register

Bits	Name	Description	Type	Reset
[31:0]	SSI_MSK	Configures whether Secure Enclave Expansion Interrupt (SEEI) {64-95} generates an interrupt to the Secure Enclave Cortex-M0+ core, with SEEI64 assigned to bit[0]: `0b0`: Interrupt is unmasked `0b1`: Interrupt is masked	RW	`0x0000_0000`

Secure Enclave Interrupt Collator Mask 3 (SEC_ENC_INT_COL_MSK3) register

The following table gives a bit-level description of the SEC_ENC_INT_COL_MSK3 register

Table 3-17 SEC_ENC_INT_COL_MSK3 register

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	`0x0000_0000`

Base System Power Request (BSYS_PWR_REQ) register

The following table gives a bit-level description of the BSYS_PWR_REQ register.

Table 3-18 BSYS_PWR_REQ register

Bits	Name	Description	Type	Reset
[31:6]	-	Reserved	RO	`0x0000_0000`
[5:3]	SYSTOP_PWR_REQ	Select SYSTOP power domain behaviour when no activity in the domain: `0b000` – No request for logic or volatile memory to be powered. `0b001` – No request for logic to be powered, but volatile memory must be retained. `0b01x` – Request for logic to be powered, but volatile memory can be either powered or retained. `0b1xx` – Request for logic and volatile memory to be powered.	RW	`0b000`

Table 3-18 BSYS_PWR_REQ register (continued)

Bits	Name	Description	Type	Reset
[2]	DBGTOP_PWR_REQ	Select DBGTOP power domain behaviour when no activity in the domain: `0b0` – No request for DBGTOP to be powered `0b1` – Request for DBGTOP to be powered	RW	`0b0`
[1]	REFCLK_REQ	Request **REFCLK**: `0b0` – No request for **REFCLK** to be supplied `0b1` – Request for **REFCLK** to be supplied	RW	`0b0`
[0]	WAKEUP_EN	Secure Enclave wakeup enable: `0b0` – Wakeup for Secure Enclave is disabled `0b1` – Wakeup for Secure Enclave is enabled	RW	`0b0`

Base System Power Status (BSYS_PWR_ST) register

The following table gives a bit-level description of the BSYS_PWR_ST register.

Table 3-19 BSYS_PWR_ST register

Bits	Name	Description	Type	Reset
[31:6]	-	Reserved	RO	`0x0000_0000`
[5:3]	SYSTOP_PWR_ST	SYSTOP power domain status: `0b000`: SYSTOP is in the OFF or WARM_RST power mode `0b001`: SYSTOP is in the MEM_RET power mode `0b010`: SYSTOP is in the FUNC_RET power mode `0b100`: SYSTOP is in the ON power mode All other values are Reserved	RO	UNKNOWN
[2]	DBGTOP_PWR_ST	DBGTOP power domain status: `0b0`: DBGTOP is in the OFF or WARM_RST power mode `0b1`: DBGTOP is in the ON-power mode	RO	UNKNOWN
[1]	-	Reserved	RO	`0b0`
[0]	-	Reserved	RO	`0b0`

Note

The values in these registers are driven from the **PPUHWSTAT** outputs of the respective PPU. The values are only valid if the respective PPU is not making a transition. For more information on the PPU, see the *Arm® Power Policy Unit Architecture Specification, version 1.1*.

SECENCCLK Clock Control (SECENCCLK_CTRL) register

The following table gives a bit-level description of the SECENCCLK_CTRL register.

Table 3-20 SECENCCLK_CTRL register

Bits	Name	Description	Type	Reset
[31:24]	ENTRY_DELAY	Configure number of idle clock cycles before clock is gated	RW	`0x00`
[23:16]	-	Reserved	RO	`0x00`
[15:8]	CLKSELECT_CUR	Currently selected clock source for **SECENCCLK**: `0x00`: Clock gate `0x01`: **SECENCREFCLK** `0x02`: **SYSPLL** All other values are Reserved.	RO	UNKNOWN
[7:0]	**CLKSELECT**	Select the clock source for **SECENCCLK**: `0x00`: Clock gate `0x01`: **SECENCREFCLK** `0x02`: **SYSPLL** All other values are Reserved. Selecting a value which is Reserved can cause a deadlock.	RW	`0x01`

SECENCCLK Clock Divider (SECENCCLK_DIV) register

The following table gives a bit-level description of the SECENCCLK_DIV register.

Table 3-21 SECENCCLK_DIV register

Bits	Name	Description	Type	Reset
[31:21]	-	Reserved	RO	`0x000`
[20:16]	CLKDIV_CUR	Current value of integer divider applied to **SYSPLL**: `0x00`: Divide by 1 `0x01`: Divide by 2 … `0x1F`: Divide by 32	RO	UNKNOWN

Table 3-21 SECENCCLK_DIV register (continued)

Bits	Name	Description	Type	Reset
[15:5]	-	Reserved	RO	`0x000`
[4:0]	CLKDIV	Select the value of the integer divider applied to **SYSPLL**: `0x00`: Divide by 1 `0x01`: Divide by 2 … `0x1F`: Divide by 32	RW	`0x00`

Clock Force Status (CLKFORCE_ST) register

The following table gives a bit-level description of the CLKFORCE_ST register.

Table 3-22 CLKFORCE_ST register

Bits	Name	Description	Type	Default
[31:1]	-	Reserved	RO	`0x0000_0000`
[0]	SECENCCLK_FORCE_ST	Status of **SECENCCLK** clock force: `0b0`: High-level clock gating is enabled `0b1`: High-level clock gating is disabled	RO	`0b0`

Note

The SECENCCLK_FORCE_ST field applies to both high-level clock gating on **SECENCCLK** and **SECENCDIVCLK**.

Clock Force Set (CLKFORCE_SET) register

The following table gives a bit-level description of the CLKFORCE_SET register.

Table 3-23 CLKFORCE_SET register

Bits	Name	Description	Type	Reset
[31:1]	-	Reserved	RO	`0x0000_0000`
[0]	SECENCCLK_FORCE_SET	Set SECENC_FORCE_ST in CLKFORCE_ST register. Writing a value of `0b0` has no effect. This field always reads as `0b0`.	WO	`0b0`

Clock Force Clear (CLKFORCE_CLR) register

The following table gives a bit-level description of the CLKFORCE_CLR register.

Table 3-24 CLKFORCE_CLR register

Bits	Name	Description	Type	Reset
[31:1]	-	Reserved	RO	`0x0000_0000`
[0]	SECENCCLK_FORCE_CLR	Clear SECENC_FORCE_ST in CLKFORCE_ST register. Writing a value of `0b0` has no effect. This field always reads as `0b0`.	WO	`0b0`

Secure Enclave Phase Locked Loop (PLL) Status (SEC_ENC_PLL_ST) register

The following table gives a bit-level description of the SEC_ENC_PLL_ST register.

Table 3-25 SEC_ENC_PLL_ST register

Bits	Name	Description	Type	Reset
[31:1]	-	Reserved	RO	`0x0000_0000`
[0]	**SYSPLLLOCK_ST**	Status of the **SYSPLLLOCK** input. `0b0`: PLL is not locked `0b1`: PLL is locked	RO	UNKNOWN

Peripheral ID 4 (PID4) register

The following table gives a bit-level description of the Peripheral ID 4 register.

Table 3-26 PID4 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	`0x00_0000`
[7:4]	Size	Number of 4KB occupied by the System ID block. This field is deprecated.	RO	`0x0`
[3:0]	DES_2	JEP Continuation	RO	`0x4`

Peripheral ID 5 (PID5) register

The following table gives a bit-level description of the Peripheral ID 5 register.

Table 3-27 PID5 fegister

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	`0x0000_0000`

Peripheral ID 6 (PID6) register

The following table gives a bit-level description of the Peripheral ID 6 register.

Table 3-28 PID6 register

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	0x0000_0000

Peripheral ID 7 (PID7) register

The following table gives a bit-level description of the Peripheral ID 7 register.

Table 3-29 PID7 register

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	0x0000_0000

Peripheral ID 0 (PID0) register

The following table gives a bit-level description of the Peripheral ID 0 register.

Table 3-30 PID0 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PART_0	Bits [7:0] of part ID	RO	0x77

Peripheral ID 1 (PID1) register

The following table gives a bit-level description of the Peripheral ID 1 register.

Table 3-31 PID1 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	DES_0	Bits [3:0] of JEP 106 Identity	RO	0xB
[3:0]	PART_1	Bits [11:8] of part ID	RO	0x0

Peripheral ID 2 (PID2) register

The following table gives a bit-level description of the Peripheral ID 2 register.

Table 3-32 PID2 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	REVISION	Major revision of the System ID block.	RO	0x0
[3]	JEDEC	Indicates the use of JEDEC JEP106 identification scheme.	RO	0b1
[2:0]	DES_1	Bits [6:4] of JEP 106 Identity	RO	0b011

Peripheral ID 3 (PID3) register

The following table gives a bit-level description of the Peripheral ID 3 register.

Table 3-33 PID3 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	REVAND	Minor revision of the System ID block	RO	0x0
[3:0]	CMOD	Customer modification field	RO	0x0

Component ID 0 (CID0) register

The following table gives a bit-level description of the Component ID 0 register.

Table 3-34 CID0 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PRMBL_0	Preamble 0	RO	0x0D

Component ID 1 (CID1) register

The following table gives a bit-level description of the Component ID 1 register.

Table 3-35 CID1 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	CLASS	Class of the component	RO	0xF
[3:0]	PRMBL_1	Preamble 1	RO	0x0

Component ID 2 (CID2) register

The following table gives a bit-level description of the Component ID 2 register.

Table 3-36 CID2 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PRMBL_2	Preamble 2	RO	0x05

Component ID 3 (CID3) register

The following table gives a bit-level description of the Component ID 3 register.

Table 3-37 CID3 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PRMBL_3	Preamble 3	RO	0xB1

3.3.2 Secure Enclave System Control register summary

This section summarizes the Secure Enclave System Control registers.

The Secure Enclave System Control Registers are in the Secure Enclave Memory Map. These registers are only accessible to the Secure Enclave Cortex-M0+ processor. The Secure Enclave System Control registers are accessed at offset `0x5008_0000` in the Secure Enclave memory map. These registers are in the Always ON power domain.

Table 3-38 Secure Enclave System Control register summary

Offset	Short name	Access	Name
`0x000`	*SE_RST_SYN* on page 3-54	RO	Secure Reset Syndrome
`0x004`	-	RO	Reserved
`0x008`	*nSE_PWR_CTRL* on page 3-54	RW	Secure Enclave Power Control
`0x00C` – `0x010`	-	RO	Reserved
`0x014`	*SE_GP0* on page 3-54	RW	Secure Enclave General Purpose 0
`0x018`	*SE_GP1* on page 3-54	RW	Secure Enclave General Purpose 1
`0x01C`	*SE_GP2* on page 3-54	RW	Secure Enclave General Purpose 2
`0x020`	*SE_GP3* on page 3-54	RW	Secure Enclave General Purpose 3
`0x024 -0x02C`	-	RO	Reserved
`0x030`	*SE_CLK_DIV* on page 3-55	RW	Secure Enclave Clock Divider Control
`0x034 -0x0F4`	-	RO	Reserved
`0x0F8`	*SE_BLD_CFG* on page 3-55	RO	Secure Enclave Build Configuration
`0x0FC - 0xFCC`	-	RO	Reserved
`0xFD0`	*Peripheral ID 4 (PID4) register* on page 3-50	RO	Peripheral ID4
`0xFD4`	*Peripheral ID 5 (PID5) register* on page 3-50	RO	Peripheral ID5
`0xFD8`	*Peripheral ID 6 (PID6) register* on page 3-50	RO	Peripheral ID6
`0xFDC`	*Peripheral ID 7 (PID7) register* on page 3-51	RO	Peripheral ID7
`0xFE0`	*Peripheral ID 0 (PID0) register* on page 3-51	RO	Peripheral ID0
`0xFE4`	*Peripheral ID 1 (PID1) register* on page 3-51	RO	Peripheral ID1
`0xFE8`	*Peripheral ID 2 (PID2) register* on page 3-51	RO	Peripheral ID2
`0xFEC`	*Peripheral ID 3 (PID3) register* on page 3-51	RO	Peripheral ID3
`0xFF0`	*Component ID 0 (CID0) register* on page 3-52	RO	Component ID0
`0xFF4`	*Component ID 1 (CID1) register* on page 3-52	RO	Component ID1
`0xFF8`	*Component ID 2 (CID2) register* on page 3-52	RO	Component ID2
`0xFFC`	*Component ID 3 (CID3) register* on page 3-52	RO	Component ID3

The Secure Enclave System Control registers only support 32-bit word aligned accesses. Access by other bit sizes or unaligned accesses, are treated as 32-bit word aligned accesses without generating an error reponse.

The Secure Enclave System Control registers have the following behavior:

- Any read to a write-only register is treated as RAZ.
- Any write to a read-only register is treated as WI.

In both cases no error is generated.

Secure Enclave Reset Syndrome (SE_RST_SYN) register

The following table gives a bit-level description of the Secure Enclave Reset Syndrome register for Secure Enclave internal sources.

Table 3-39 SE_RST_SYN register

Bits	Name	Description	Type	Reset
[31:2]	-	Reserved	RO	`0x0000_0000`
[1]	WD_RESET	Indicates that the last reset cause of the Secure Enclave was caused by the Secure Enclave Watchdog or not: `0b0`: Last reset of the Secure Enclave was not caused by the Secure Enclave Watchdog `0b1`: Last reset of the Secure Enclave was caused by the Secure Enclave Watchdog	RO	`0b0`
[0]	SW_RESET	Indicates that the last reset cause of the Secure Enclave was caused by the Secure Enclave software reset request: `0b0`: Last reset of the Secure Enclave was not caused by the Secure Enclave software reset request `0b1`: Last reset of the Secure Enclave was caused by the Secure Enclave software reset request	RO	`0b0`

Note

This register must be used in conjunction with the SOC_RST_SYN register in the Secure Enclave Base System Control registers.

Secure Enclave Power Control (SE_PWR_CTRL) register

The following table gives a bit-level description of the SECENTOP Power Domain Control register.

Table 3-40 SE_PWR_CTRL register

Bits	Name	Description	Type	Reset
[31:1]	-	Reserved	RO	`0x0000_0000`
[0]	PWR_GATE_EN	SECENCTOP can enter OFF or MEM_RET: `0b0`: SECENCTOP must remain in ON. `0b1`: SECENCTOP can enter OFF or MEM_RET next time the Secure Enclave is idle.	RW	`0b0`

Secure Enclave General Purpose {0-3} (SE_GP{0-3}) register

The following table gives a bit-level description of the SE_GP{0-3} register.

Table 3-41 SE_GP{0-3} register

Bits	Name	Description	Type	Reset
[31:0]	GP	General-purpose data register	RW	`0x0000_0000`

Secure Enclave Clock Divider Control (SE_CLK_DIV) register

The following table gives a bit-level description of the SE_CLK_DIV register.

Table 3-42 SE_CLK_DIV register

Bits	Name	Description	Type	Reset
[31:21]	-	Reserved	RO	`0x000`
[20:16]	CLKDIV_CUR	Current value of integer divider applied to **SECENCCLK**: `0x00`: Divide by 1 `0x01`: Divide by 2 … `0x1F`: Divide by 32	RO	UNKNOWN
[15:5]	-	Reserved	RO	`0x000`
[4:0]	CLK_DIVIDER	Controls the clock divider ratio: 0: 1:1 ratio between **SECENCDIVCLK** clock and **SECENCCLK** clock 1: 1:2 ratio between **SECENCDIVCLK** and **SECENCCLK** 2 - 31: Reserved	RW	`0x01`

Secure Enclave Build Configuration (SE_BLD_CFG) register

The following table gives a bit-level description of the SE_BLD_CFG register.

Table 3-43 SE_BLD_CFG register

Bits	Name	Description	Type	Reset
[31:16]	**RAM_SIZE**	Secure Enclave RAM size. The value of this field is the size of the Secure Enclave RAM in KB. A value of `0x00` is Reserved. For example, a value of `0x40` indicates a RAM of 64KB whilst a value of `0x80` indicates a RAM of 128KB.	RO	`0x80`
[15:0]	**ROM_SIZE**	Secure Enclave ROM size. The value of this field is the size of the Secure Enclave ROM in KB. A value of `0x00` is Reserved. For example, a value of `0x20` indicates a ROM of 32KB whilst a value of `0x40` indicates a ROM of 64KB.	RO	`0x20`

Peripheral ID 4 (PID4) register

The following table gives a bit-level description of the Peripheral ID 4 register.

Table 3-44 PID4 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	Size	Number of 4KB occupied by the System ID block. This field is deprecated.	RO	0x0
[3:0]	DES_2	JEP Continuation	RO	0x4

Peripheral ID 5 (PID5) register

The following table gives a bit-level description of the Peripheral ID 5 register.

Table 3-45 PID5 fegister

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	0x0000_0000

Peripheral ID 6 (PID6) register

The following table gives a bit-level description of the Peripheral ID 6 register.

Table 3-46 PID6 register

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	0x0000_0000

Peripheral ID 7 (PID7) register

The following table gives a bit-level description of the Peripheral ID 7 register.

Table 3-47 PID7 register

Bits	Name	Description	Type	Reset
[31:0]	-	Reserved	RO	0x0000_0000

Peripheral ID 0 (PID0) register

The following table gives a bit-level description of the Peripheral ID 0 register.

Table 3-48 PID0 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PART_0	Bits [7:0] of part ID	RO	0x73

Peripheral ID 1 (PID1) register

The following table gives a bit-level description of the Peripheral ID 1 register.

Table 3-49 PID1 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	DES_0	Bits [3:0] of JEP 106 Identity	RO	0xB
[3:0]	PART_1	Bits [11:8] of part ID	RO	0x0

Peripheral ID 2 (PID2) register

The following table gives a bit-level description of the Peripheral ID 2 register.

Table 3-50 PID2 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	REVISION	Major revision of the System ID block.	RO	0x0
[3]	JEDEC	Indicates the use of JEDEC JEP106 identification scheme.	RO	0b1
[2:0]	DES_1	Bits [6:4] of JEP 106 Identity	RO	0b011

Peripheral ID 3 (PID3) register

The following table gives a bit-level description of the Peripheral ID 3 register.

Table 3-51 PID3 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	REVAND	Minor revision of the System ID block	RO	0x0
[3:0]	CMOD	Customer modification field	RO	0x0

Component ID 0 (CID0) register

The following table gives a bit-level description of the Component ID 0 register.

Table 3-52 CID0 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PRMBL_0	Preamble 0	RO	0x0D

Component ID 1 (CID1) register

The following table gives a bit-level description of the Component ID 1 register.

Table 3-53 CID1 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:4]	CLASS	Class of the component	RO	0xF
[3:0]	PRMBL_1	Preamble 1	RO	0x0

Component ID 2 (CID2) register

The following table gives a bit-level description of the Component ID 2 register.

Table 3-54 CID2 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PRMBL_2	Preamble 2	RO	0x05

Component ID 3 (CID3) register

The following table gives a bit-level description of the Component ID 3 register.

Table 3-55 CID3 register

Bits	Name	Description	Type	Reset
[31:8]	-	Reserved	RO	0x00_0000
[7:0]	PRMBL_3	Preamble 3	RO	0xB1

Chapter 4
Software sequences

This chapter describes software sequences in the Secure Enclave.

It contains the following sections:

4.1 SECENCTOP power domain

The SECENCTOP power domain is controlled using:

- PWR_GATE_EN field in the Power Control register of the Secure Enclave System Control registers. For more information see *3.3.2 Secure Enclave System Control register summary* on page 3-52.
- Power policy of the SECENC PPU
- Cortex-M0+ in a DeepSleep state

Arm strongly recommends that the SECENCTOP PPU policy is set to either dynamic OFF or dynamic MEM_RET, depending on whether software requires the retention of the contents of the Secure Enclave's SRAM. It is assumed that the PPU policy is either set to dynamic OFF or dynamic MEM_RET.

Note

Setting the SECENCTOP PPU policy to another policy may lead to different behavior.

To enter the SECENCTOP power domain into the MEM_RET or OFF power mode, software must do the following:

- Save any state which needs to be retained that otherwise is lost by entering either the MEM_RET or the OFF power state.
- Set the PWR_GATE_EN bit `0b1`.
- Set the SCR.SLEEPDEEP bit `0b1` and execute a WFI.

To select between entering MEM_RET or OFF, software can change the power policy in the PPU:

- To enter MEM_RET software must set the policy to dynamic MEM_RET.
- To enter OFF, software must set the policy to OFF.

Arm strongly recommends that the SECENCTOP PPU is always programmed for dynamic transitions.

When the SECENCTOP power domain exits the MEM_RET or OFF power mode, software must set the PWR_GATE_EN bit `0b0` before performing any other actions.

4.2 Advancing lifecycle states

This section gives example sequences for advancing lifecycle states: one uses the SSE-700 **SOCLCC** signal, and the other uses the GPIO Control component.

This section contains the following subsections:

4.2.1 Using the SOCLCC signal

The following example sequence uses the **SOCLCC** signal:

Procedure

1. Assert the **SOCLCC** signal of the SSE-700 subsystem HIGH.
2. De-assert **PORESETn** to the SSE-700 subsystem.
3. The Secure Enclave Cortex-M0+ CPU performs an IMPLEMENTATION DEFINED software sequence to advance the lifecycle state. The software sequence depends on the Cryptographic Accelerator implemented in the Secure Enclave.
4. When the transition is complete, de-assert the **SOCLCC** signal LOW.

4.2.2 Using the Debug Port

The following example sequence uses the *Debug Port* (DP):

Procedure

1. Set the GPIO Control GPO0 output HIGH.
2. Cause a debug reset by either:
   - Asserting **nSRST** LOW.
   - Asserting **CSYSRSTREQ** of the DP ROM HIGH, and waiting for DP ROM **CSYSRSTACK** to become HIGH.
3. Release the SSE-700 subsystem from debug reset by:
   - De-asserting the **nSRST** HIGH.
   - De-asserting DP ROM **CSYSRSTREQ** LOW and waiting for the DP ROM **CSYSRSTACK** to become LOW.
4. The Secure Enclave Cortex-M0+ CPU performs an IMPLEMENTATION DEFINED software sequence to advance the lifecycle state. This sequence depends upon the Cryptographic Accelerator implemented in the Secure Enclave.

Appendix A
Revisions

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

It contains the following section:

- *A.1 Revisions* on page Appx-A-63.

A.1 Revisions

This appendix describes changes between released issues of this book.

Table A-1 Issue 0100-00

Change	Location	Affects
First release for BET	-	BET release

Table A-2 Differences between issue 0000-00 and issue 0000-01

Change	Location	Affects
Numerous minor technical updates applied throughout document	All sections	r0p0 LAC release
Included PL011 UART in Overview	For more information see *1.1 Overview* on page 1-12	
Extended description of Secure Enclave components list	*1.1 Overview* on page 1-12	
Re-cast of Lifecycle States advancement description	*2.1.2 Lifecycle States (LCS)* on page 2-15	
Identified DFTENABLE[1] as signal	*2.1.2 Lifecycle States (LCS)* on page 2-15	
Re-cast SSE-700 subsystem SCB rules	*2.1.3 Security Control Bits (SCB)* on page 2-16	
Re-aligned style of binary values from 0/1 to `0b0/0b1`	All sections	
Extended cross-refs to Timers	*Timers* on page 2-20	
Changed intro to Base System Control registers	*Base System Control registers* on page 2-21	
Additional information added to the Secure Enclave Firewall	*Secure Enclave Firewall* on page 2-21	
Removed reference to protection size of firewall component	*Secure Enclave Firewall* on page 2-21	

Table A-2 Differences between issue 0000-00 and issue 0000-01 (continued)

Change	Location	Affects
Removed reference to integration of the Crypto Accelerator into the Secure Enclave	*Crypto Reset interface* on page 2-23	r0p0 LAC release
Added text explaining the Crypto Reset interface CRYPTORESETN is driven by SECENCWARMRESETN	*Crypto Reset interface* on page 2-23	
Removed reference to integration of the Crypto Accelerator AON into the Secure Enclave	*Crypto AON Reset Interface* on page 2-23	
Reverted to bullet list description for Crypto Accelerator Master interface	*Crypto Accelerator Master (CAM) interface* on page 2-23	
Changed all references to signal names ending in 'N' to 'n'	All sections	
Removed information relating to signals as conmponents of AONTOP and SECENTOP power domains	*Crypto Accelerator AON (CAAON) interface* on page 2-24	
Removed reference to *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*	*2.2.4 Security Control Bits (SCB) interface* on page 2-26	
Re-organized the enumeration of Memory map, interrupt map, and registers	*Chapter 3 Programmers model* on page 3-33	
Merged notes column at lines 64-71, added reference to *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*	*3.2.2 Secure Enclave interrupt expansion* on page 3-39	
Added reference to *Arm® Corstone™ SSE-700 Subsystem Technical Reference Manual*	*MHUs* on page 3-36	
Reversed clock names in descriptions	*Secure Enclave Clock Divider Control (SE_CLK_DIV) register* on page 3-55	

Table A-3 Differences between issue 0000-01 and issue 0000-02

Change	Location	Affects
Changed from confidential to non-confidential	-	r0p0 LAC non-confidential release
Numerous minor editorial updates applied throughout document. Post LAC Release Review.	All sections	r0p0

Table A-4 Differences between issue 0000-02 and issue 0000-03

Change	Location	Affects
Change confidential to non-confidential	-	r0p0 EAC
Removed extraneous section descriptions, made editorial changes and corrections, added references	Throughout the manual	
Updated publications list	*Additional reading* on page 8	
Updated functional description information	*2.1.3 Security Control Bits (SCB)* on page 2-16, *Timers* on page 2-20, *2.3.2 SECENCCLK* on page 2-27, *2.4.1 SECENCTOP* on page 2-29	
Updated programmers model information	*3.2.2 Secure Enclave interrupt expansion* on page 3-39, *Peripheral ID 1 (PID1) register* on page 3-51, *Secure Enclave Reset Syndrome (SE_RST_SYN) register* on page 3-54, *Secure Enclave Power Control (SE_PWR_CTRL) register* on page 3-54, *Peripheral ID 1 (PID1) register* on page 3-51	
Updated software sequences information	*4.2.2 Using the Debug Port* on page 4-61	