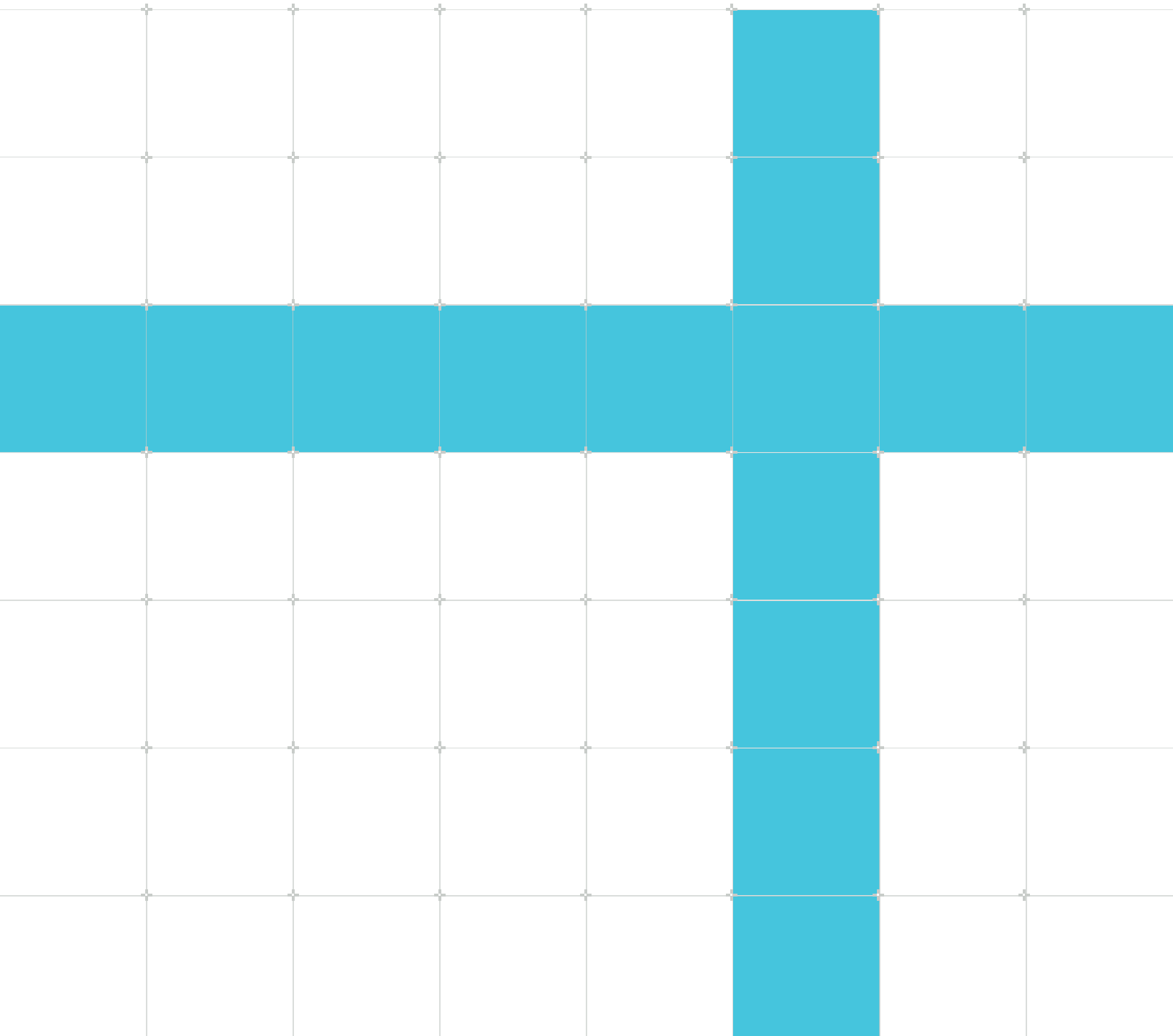
Arm® Corstone™ SSE-123 Example Subsystem

Revision: r0p0

Technical Overview

Non-Confidential

Issue 02
101371_0000_02_en



Arm® Corstone™ SSE-123 Example Subsystem

Technical Overview

Release Information

Document history

Issue	Date	Confidentiality	Change
0000-00	22 March 2019	Non-Confidential	First release for r0p0
0000-01	10 April 2020	Non-Confidential	Second release for r0p0
0000-02	30 June 2022	Non-Confidential	Third release for r0p0

Proprietary Notice

REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word "partner" in reference to Arm's customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349)

Additional Notices

Some material in this document is based on IEEE 754-2008 IEEE Standard for Binary Floating-Point Arithmetic. The IEEE disclaims any responsibility or liability resulting from the placement and use in the described manner.

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.

Feedback

Arm welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on https://support.developer.arm.com.

To provide feedback on the document, fill the following survey: https://developer.arm.com/documentation-feedback-survey.

Inclusive language commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used language that can be offensive. Arm strives to lead the industry and create change.

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

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

Contents

1. Introduction

1.1 Product revision status

The r*x*p*y* identifier indicates the revision status of the product described in this manual, for example, r*1*p*2*, where:

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

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

1.2 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 Arm® Corstone™ SSE-123 Example Subsystem.

1.3 Conventions

The following subsections describe conventions used in Arm documents.

Glossary

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

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

Typographic conventions

Convention	Use
italic	Citations.
bold	Interface elements, such as menu names. Signal names. Terms in descriptive lists, where appropriate.
`monospace`	Text that you can enter at the keyboard, such as commands, file and program names, and source code.
`monospace bold`	Language keywords when used outside example code.

Convention	Use
`monospace underline`	A permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.
`<and>`	Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example: `MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>`
SMALL CAPITALS	Terms that have specific technical meanings as defined in the *Arm® Glossary*. For example, **IMPLEMENTATION DEFINED**, **IMPLEMENTATION SPECIFIC**, **UNKNOWN**, and **UNPREDICTABLE**.

Recommendations. Not following these recommendations might lead to system failure or damage.

Requirements for the system. Not following these requirements might result in system failure or damage.

Requirements for the system. Not following these requirements will result in system failure or damage.

An important piece of information that needs your attention.

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

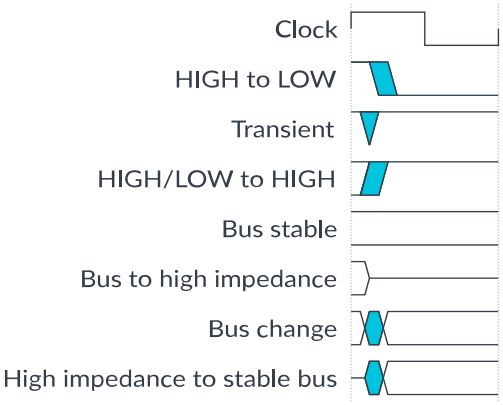
A reminder of something important that relates to the information you are reading.

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 1-1: Key to timing diagram conventions



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.

1.4 Additional reading

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

Table 1-2: Arm publications

Document name	Document ID	Licensee only
Arm® SSE-123 Example Subsystem Technical Reference Manual	101370	No
Arm®v8-M Architecture Reference Manual	DDI 0553	No
Arm® Power Control System Architecture Specification Version 2.0	DEN 0050	Yes

Document name	Document ID	Licensee only
Arm® SSE-123 Example Subsystem Configuration and Integration Manual	101372	Yes
Arm® SSE-123 Example Subsystem Release Note	PJDOC-1779577084-12680	Yes
Arm® SSE-123 Example Subsystem Analysis Report	PJDOC-1779577084-28939	Yes
Arm® SSE-123 Example Subsystem Verification Summary Report	PJDOC-1779577084-28938	Yes

Arm tests its PDFs only in Adobe Acrobat and Acrobat Reader. Arm cannot guarantee the quality of its documents when used with any other PDF reader.

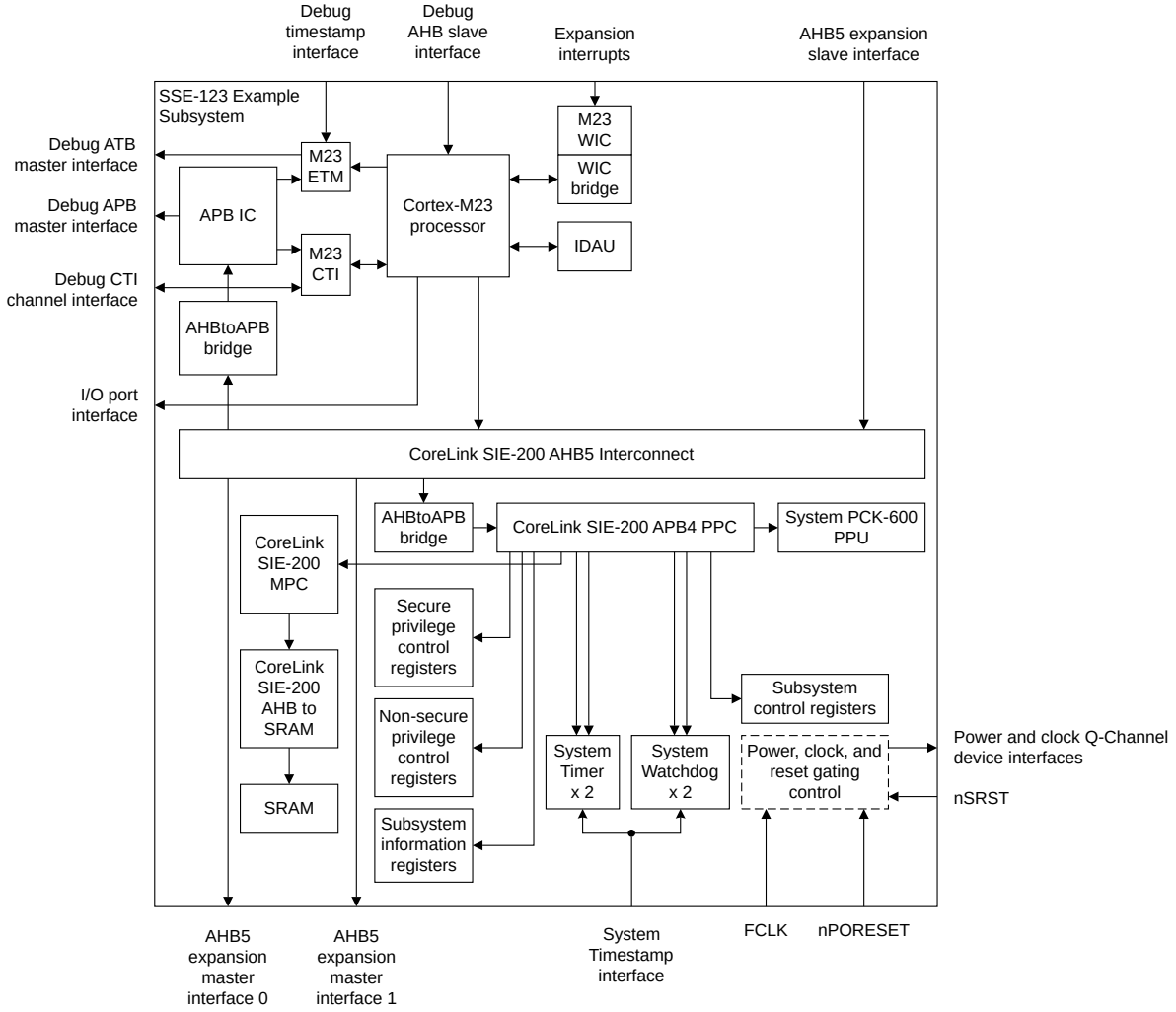
Adobe PDF reader products can be downloaded at http://www.adobe.com

2. SSE-123 Example Subsystem

The SSE-123 Example Subsystem integrates a subsystem of key Arm components that implement core functionality of a system targeting *Internet of Things* (IoT) *System on Chip* (SoC) designs.

The following figure shows a block diagram of the SSE-123 Example Subsystem.

Figure 2-1: SSE-123 Example Subsystem block diagram



The block diagram shows all the key integrated components and interfaces.

2.1 About IoT System on Chip implementations

The SSE-123 Example Subsystem must be extended to create an IoT SoC. A complete system typically contains the following components:

Compute subsystem

The compute subsystem consists of a single Cortex®-M23 processor and associated bus, debug, controller, peripherals, and interface logic supplied by Arm.

Reference system memory and peripherals

SRAM is part of the SSE-123 Example Subsystem, but an SoC requires extra memory, control, and peripheral components beyond the minimum subsystem components. Flash memory, for example, is not provided with the SSE-123 Example Subsystem.

Communication interface

The endpoint must have some way of communicating with other nodes or masters in the system. This interface could be WiFi, Bluetooth, or a wired connection.

Sensor or control component

To be useful as an endpoint, the reference design is typically extended by adding sensors or control logic such as temperature input or motor control output.

Software development environment

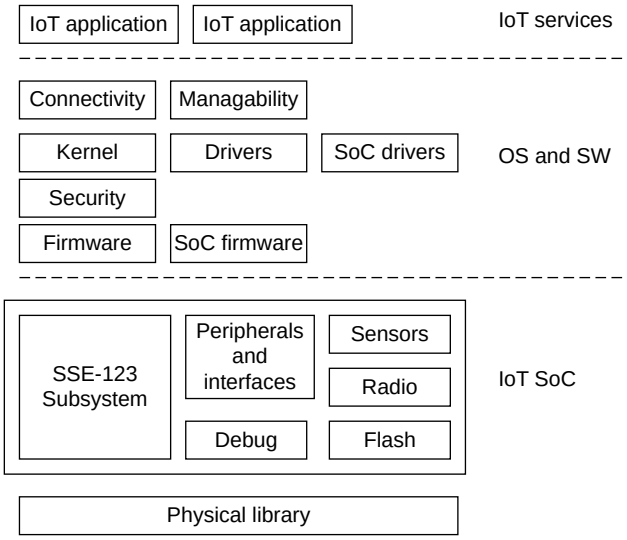
Arm provides a complete software development environment which includes the Arm® Mbed™ operating system, Arm or GNU (GCC) compilers and debuggers, and firmware.

Custom peripherals typically require corresponding third-party firmware that can be integrated into the software stack.

2.1.1 IoT hardware and software

The following figure shows a block diagram of the hardware and software in an IoT system.

Figure 2-2: Hardware and software solution



2.2 Compliance

The SSE-123 Example Subsystem complies with, or implements, the specifications that this section describes. This document complements architecture reference manuals, architecture specifications, protocol specifications, and relevant external standards. It does not duplicate information from these sources.

2.2.1 Arm architecture

The Cortex®-M23 processor in the subsystem implements the Armv8-M Baseline Architecture with Security Extension.

See the *Arm®v8-M Architecture Reference Manual*.

2.2.2 Security architecture

The SSE-123 is designed to facilitate implementation of a TBSA-M compliant system.

See the *Arm® Platform Security Architecture - Trusted Base System Architecture for Armv8-M*.

2.2.3 Interrupt controller architecture

The SSE-123 implements Arm® *Nested Vector Interrupt Controller* (NVIC) and Arm® *Wakeup Interrupt Controller* (WIC).

See the *Arm® Cortex®-M23 Processor Technical Reference Manual*.

2.2.4 Advanced Microcontroller Bus Architecture (AMBA®)

The SSE-123 implements the following interface protocol architectures:

- *Advanced High-Performance Bus 5* (AHB5). See the *AMBA® 5 AHB Protocol Specification*.
- *Advanced Peripheral Bus 4* (APB4). See the *AMBA® APB Protocol Specification Version: 2.0*.
- *Low-Power Interface* (LPI), Q-Channel, and- P-Channel. See the *AMBA® Low Power Interface Specification*.

2.2.5 Debug architecture

The SSE-123 implements the Arm® *Debug Interface Architecture 5* (ADIv5)-compliant debug interfaces.

See the *Arm® Debug Interface Architecture Specification, ADIv5.0 to ADIv5.2*.

2.2.6 Power control architecture

The SSE-123 implements the framework for system power control that the Arm® *Power Control System Architecture* (PCSA) Version 2.0 specification defines.

See the *Arm® Power Control System Architecture Specification Version 2.0*.

2.3 Product documentation

This section describes the SSE-123 product documentation in relation to the design flow.

2.3.1 Documentation

The SSE-123 Example Subsystem documentation is as follows:

Technical Overview

The *Technical Overview* (TO) provides a high-level overview of the SSE-123 Example Subsystem:

- Hardware.
- Software.

Technical Reference Manual

The *Technical Reference Manual* (TRM) describes the functionality and the effects of functional options on the behavior of the SSE-123 Example Subsystem. It is required at all stages of the design flow. The choices that are made in the design flow can mean that some behaviors that are described in the TRM are not relevant. If you are programming the SSE-123, then contact:

- The implementer to determine:
  - The build configuration of the implementation.
  - The integration, if any, that was performed before implementing the SSE-123.
- The integrator to determine the pin configuration of the device that you are using.

Configuration and Integration Manual

The *Configuration and Integration Manual* (CIM) describes:

- The available build configuration options and related issues in selecting them.
- Guidelines on how to integrate the SSE-123 Example Subsystem into an SoC.
- The SSE-123 Integration component, providing examples of integration with Arm® eFlash and debug products.
- The processes to sign off the configuration, integration, and physical implementation of the design.

The CIM is a confidential book that is only available to licensees.

Verification Summary Report

The *Verification Summary Report* (VSR) describes:

- An overview of verification performed on the SSE-123 Example Subsystem.
- The verification quality definition for the subsystem.
- Configurations of the subsystem verified.
- A summary of verification results.

The VSR is a confidential book that is only available to licensees.

Subsystem Analysis Report

The *Subsystem Analysis Report* (SAR) describes:

- Performance characteristics of the SSE-123 Example Subsystem.
- Processor performance analysis and benchmark results.
- Performance analysis of memory system bandwidth and latency.

The SAR is a confidential book that is only available to licensees.

2.4 Product revisions

This section describes the differences in functionality between product revisions:

r0p0 First release.

3. Hardware

The SSE-123 Example Subsystem provides the following features:

- A Cortex®-M23 processor, including Armv8-M Security Extensions
- A single bank of system SRAM
- CoreLink™ SIE-200 System IP for Embedded:
  - AHB5 bus matrix
  - *Memory Protection Controller* (MPC)
  - *Peripheral Protection Controller* (PPC)
  - AHB5 to APB4 bridge
  - AHB5 to SRAM controller
- CoreLink™ PCK-600 Power Control Kit:
  - *Power Policy Unit* (PPU)
  - Clock controller
  - Low-Power Distributor Q-Channel (LPD-Q)
- *Implementation Defined Attribution Unit* (IDAU)
- Cortex®-M23 processor *Wakeup Interrupt Controller* (WIC)
- System Timer and Watchdog
- System Control and Security Control Registers
- Optional Cortex®-M23 processor Debug components:
  - *Embedded Trace Macrocell* (ETM)
  - *Cross Trigger Interface* (CTI)
  - Debug APB interconnect

4. Software

Application processor firmware, which is available separately, consists of the code that is required to boot the up to the point where the OS execution starts. Contact your Arm representative for details on the software and its location.

The SSE-123 firmware contains:

- Trusted Firmware for M-class (TF-M) that separates the Secure and Non-secure execution environment.
- *Cortex Microcontroller Software Interface Standard* (CMSIS) compliant drivers.
- Mbed OS driver support and code for applicable peripherals.


Note

For more information on Mbed™, see mbed.com.

Appendix A Revisions

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

Table A-1: Issue 0000-00

Change	Location
First release	-

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

Change	Location
No technical changes.	-

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

Change	Location
Added Power Control System Architecture Specification to Additional Reading list.	1.4 Additional reading on page 8