Arm® Base Boot Requirements Architecture Compliance

Revision: r0p0

User Guide

arm

102505_0000_01_en

Arm® Base Boot Requirements Architecture Compliance

User Guide

Release Information

Document History

Issue	Date	Confidentiality	Change
0000-01	18 May 2021	Non-Confidential	Initial release

Non-Confidential Proprietary Notice

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349)

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

The information in this document is for a Beta product, that is a product under development.

Web Address

developer.arm.com

Progressive terminology commitment

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

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

If you find offensive terms in this document, please contact *terms@arm.com*.

Contents

Arm® Base Boot Requirements Architecture Compliance User Guide

Preface

This preface introduces the *Arm® Base Boot Requirements Architecture Compliance User Guide*.

It contains the following:

About this book

This book is the user guide for Arm® Base Boot Requirements Architecture Compliance User Guide.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction to BBR

This chapter provides information on BBR and the abbreviations that are used in this document.

Chapter 2 Self-Certification Test

This chapter provides information on SCT and the tests that are required to run it.

Chapter 3 Firmware Test Suite

This chapter provides information on FWTS, steps to run the tests, and to build FWTS.

Chapter 4 Building SCT and FWTS

This chapter provides information on how to build SCT and FWTS tests.

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.

`monospace`

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

`monospace italic`

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

`monospace bold`

Denotes language keywords when used outside example code.

`<and>`

Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example:

```
MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>
```

SMALL CAPITALS

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

Other information

- *Arm® Developer*.
- *Arm® Documentation*.
- *Technical Support*.
- *Arm® Glossary*.

Feedback

Feedback on this product

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

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

Feedback on content

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

- The title *Arm Base Boot Requirements Architecture Compliance User Guide*.
- The number 102505_0000_01_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 to BBR

This chapter provides information on BBR and the abbreviations that are used in this document.

It contains the following sections:

1.1 Abbreviations

This section lists the abbreviations used in this document.

Table 1-1 Abbreviations and expansion

Abbreviation	Expansion
ACS	Architecture Compliance Suite
ACPI	Advanced Configuration and Power Interface
BBR	Base Boot Requirements
BSA	Base System Architecture
EBBR	Embedded Base Boot Requirements
FWTS	Firmware Test Suite
GRUB	GRand Unified Bootloader
OEM	Original Equipment Manufacturer
ODM	Original Device Manufacturer
OS	Operating System
SBSA	Server Base System Architecture
SCT	Self-Certification Tests
SBBR	Server Base Boot Requirements
UEFI	Unified Extensible Firmware Interface

1.2 About the Base Boot Requirements

This section provides information on BBR and the runtime executable environments.

The *Base Boot Requirements* (BBR) for boot and runtime services are based on Arm 64-bit architecture, that system software can rely on. For example, operating systems and hypervisors.

A driver-based model for advanced platform capabilities beyond basic system configuration and boot is required. However, this model is beyond the scope of this document. Fully discoverable and describable peripherals aid the implementation of this type of a driver model. BBR identifies the Arm and industry standard firmware interfaces applicable to the Arm 64-bit architecture. They include the PSCI, SMCCC, UEFI, ACPI, SMBIOS, and DT interfaces. Requirements that are based on these interfaces are specified. In addition, various recipes are created to accommodate the various operating systems and hypervisors.

The BBR test suites check for compliance against the *Server Base Boot Requirements* (SBBR) or *Embedded Base Boot Requirements* (EBBR) specification. Similar to the BSA tests, these tests are also delivered through two runtime executable environments:

1. Self-Certification Tests (SCT)
2. Firmware Test Suite (FWTS)

For more information, see BBR specification *https://developer.arm.com/documentation/den0044/latest*.

Chapter 2
Self-Certification Test

This chapter provides information on SCT and the tests that are required to run it.

It contains the following section:

2.1 Introduction to SCT

SCT examines the UEFI implementation requirements defined by the BBR recipes SBBR or EBBR.

For more information on SCT, see *https://github.com/tianocore/edk2-test*.

This section contains the following subsection:

- *2.1.1 Running SCT* on page 2-13.

2.1.1 Running SCT

This section provides information on how to run SCT.

The UEFI SCTs are built as part of the test suite. Running of UEFI SCTs is now automated. You can choose to skip the automated SCTs by pressing any key when the UEFI shell prompts.

Shell > Press any key to stop the UEFI SCT running.

To run SCT manually, follow these steps:

1. Shell > `FS(X):>cd EFI\BOOT\bbr\SCT`
2. To run SBBR or EBBR tests: `FS(X):EFI\BOOT\bbr\SCT>SCT -s <ebbr.seq/sbbr.seq>`
3. To run all tests: `FS(X):EFI\BOOT\bbr\SCT>SCT -a -v`

Note

Ensure that the system time is correct before starting the SCTs.

You can also select and run tests individually. For more information on running the tests, see SCT User Guide *http://www.uefi.org/testtools*.

See *Chapter 4 Building SCT and FWTS* on page 4-16, for the steps to build SCT for SBBR or EBBR recipes.

Chapter 3
Firmware Test Suite

This chapter provides information on FWTS, steps to run the tests, and to build FWTS.

It contains the following section:

3.1 Introduction to FWTS

Firmware Test Suite (FWTS) is a package that is hosted by Canonical. FWTS provides tests for *Advanced Configuration and Power Interface* (ACPI), SMBIOS, and UEFI. Several SBBR requirements are tested through FWTS.

For more information on FWTS, see *https://wiki.ubuntu.com/FirmwareTestSuite*.

This section contains the following subsection:

- *3.1.1 Running FWTS* on page 3-15.

3.1.1 Running FWTS

To run FWTS manually, the built image must be first copied to the bin directory Linux file system.

To run FWTS on the Linux prompt, follow these steps:

1. For SBBR, use the following command:

   ```
   #/bin/fwts -r stdout -q --sbbr
   ```

2. For EBBR, use the following command:

   ```
   #/bin/fwts uefirtmisc uefirttime uefirtvariable securebootcert uefirtauthvar
   uefivarinfo esrt uefibootpath
   ```

Chapter 4
Building SCT and FWTS

This chapter provides information on how to build SCT and FWTS tests.

It contains the following section:

4.1 Building SCT and FWTS

This section describes how to build SCT and FWTS for EBBR and SBBR recipes.

The SystemReady ACS live image contains all the BBR ACS tests. However, for debug and verification of bug-fixes, the following steps enable the user to build SCT and FWTS images independently.

Following are the steps to build SCT and FWTS:

1. Clone BBR repository:

   ```
   git clone ssh://ap-gerrit-1.ap01.arm.com:29418/avk/syscomp_bbr bbr-acs
   ```

2. Get the source code:

   ```
   cd bbr-acs/<ebbr/sbbr>/scripts
   ```

   ```
   ./build-scripts/get_<ebbr/sbbr>_source.sh
   ```

3. Build BBR ACS from the scripts directory run

   ```
   ./build-scripts/build_<ebbr/sbbr>.sh
   ```

   to build SCT and FWTS. The script applies patches to create EBBR or SBBR to build recipe in the SCT / FWTS build system. It copies over new files to support the new tests and to run or start up a sequence.

The generated image of SCT can be found in `bbr-acs/<ebbr/sbbr>/scripts/edk2-test/uefi-sct/<ARCH>_SCT` which is the `AARCH64_SCT`.

The generated image of FWTS can be found in `bbr-acs/<ebbr/sbbr>/scripts/fwts/fwts_output`.

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-19.

A.1 Revisions

This section consists of all the technical changes between different versions of this document.

Table A-1 Issue 0000-01

Change	Location
First release	-