ARM® TrustZone® True Random Number Generator

Revision: r0p0

Technical Reference Manual

ARM 100976_0000_00_en

ARM® TrustZone® True Random Number Generator

Technical Reference Manual

Release Information

Document History

Issue	Date	Confidentiality	Change
0000-00	05 May 2017	Non-Confidential	First release for r0p0

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 Final, that is for a developed product.

Web Address

http://www.arm.com

Contents

ARM® TrustZone® True Random Number Generator Technical Reference Manual

Preface

This preface introduces the *ARM® TrustZone® True Random Number Generator Technical Reference Manual*.

It contains the following:

About this book

This book is for the ARM® TrustZone® TRNG True Random Number Generator.

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 TRNG.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction

This chapter provides an introduction to the ARM TrustZone TRNG True Random Number Generator.

Chapter 2 Functional description

This chapter describes the functions of the ARM TrustZone TRNG True Random Number Generator.

Chapter 3 Programmers model

This chapter describes the TRNG register addresses and functionality for integration.

Chapter 4 Signal descriptions

This appendix describes the top-level signals of the TrustZone TRNG True Random Number Generator.

Appendix A Revisions

Read this for a description of 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.

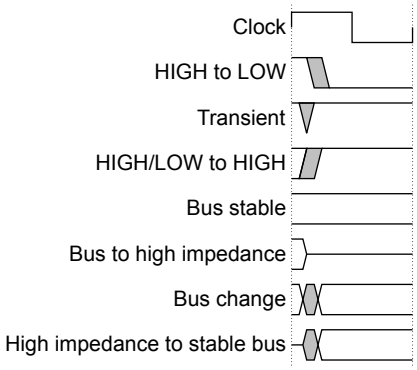

Figure 1 Key to timing diagram conventions

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 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® AMBA® Specification (Rev 2.0)* (ARM IHI 0011).

The following confidential books are only available to licensees:

- *ARM® TrustZone® TRNG True Random Number Generator Configuration and Integration Manual* (ARM 100977).

Other publications

The following relevant documents are published by third parties:

- Bundesamt für Sicherheit in der Informationstechnik (BSI), *Functionality Classes and Evaluation Methodology for True Random Number Generators*, AIS-31, Version 3.1, September 2001.
- Federal Information Processing Standard, *Security Requirements for Cryptographic Modules*, FIPS 140-2, May 2001. See *National Institute of Standards and Technology* website.
- NIST, *Recommendation for the Entropy Sources Used for Random Bit Generation*, SP 800-90B, January 2016.

Feedback

Feedback on this product

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

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

Feedback on content

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

- The title *ARM TrustZone True Random Number Generator Technical Reference Manual*.
- The number ARM 100976_0000_00_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 provides an introduction to the ARM TrustZone TRNG True Random Number Generator.

It contains the following sections:

1.1 About the TrustZone TRNG

The TrustZone TRNG True Random Number Generator enables generation and collection of a truly random bit stream from a digital logic. The TRNG is designed for simple SoC integration.

The typical usage of a TRNG is key generation or for seeding approved deterministic random numbers.

1.2 Compliance

The TRNG True Random Number Generator complies with, or implements, the following specifications:

- Complies with all applicable true random number generator requirements of the *Security Requirements for Cryptographic Modules*, FIPS 140-2.
- Complies with the AIS recommendations in the *Functionality Classes and Evaluation Methodology for True Random Number Generators*.
- Complies with the *Recommendation for the Entropy Sources Used for Random Bit Generation*, SP 800-90B.
- Complies with the APB2 protocol. See the *ARM® AMBA® Specification (Rev 2.0)*.

1.3 Features

The TRNG core has the following key features:

- Produces 10K bits/second of entropy when core is running at 200MHz.
- Includes an internal entropy source that is based on a chain of digital inverters.
  - Odd number of inverters, leading to continuous oscillation (while active).
  - Inverter cells that are taken from a standard cells library.
- Built-in hardware tests for auto correlation and *Continuous Random Number Generation Testing* (CRNGT) as required by the following standards:
  - FIPS 140-2, *Security Requirements for Cryptographic Modules*.
  - AIS-31, *Functionality Classes and Evaluation Methodology for True Random Number Generators*.
- AMBA APB2 slave interface.

1.4 Product revisions

The differences in functionality between the TRNG product revisions are:

r0p0 First release.

Chapter 2
Functional description

This chapter describes the functions of the ARM TrustZone TRNG True Random Number Generator.

It contains the following section:

2.1 About the functions

The following figure illustrates a top view of the TrustZone TRNG and its various interfaces.

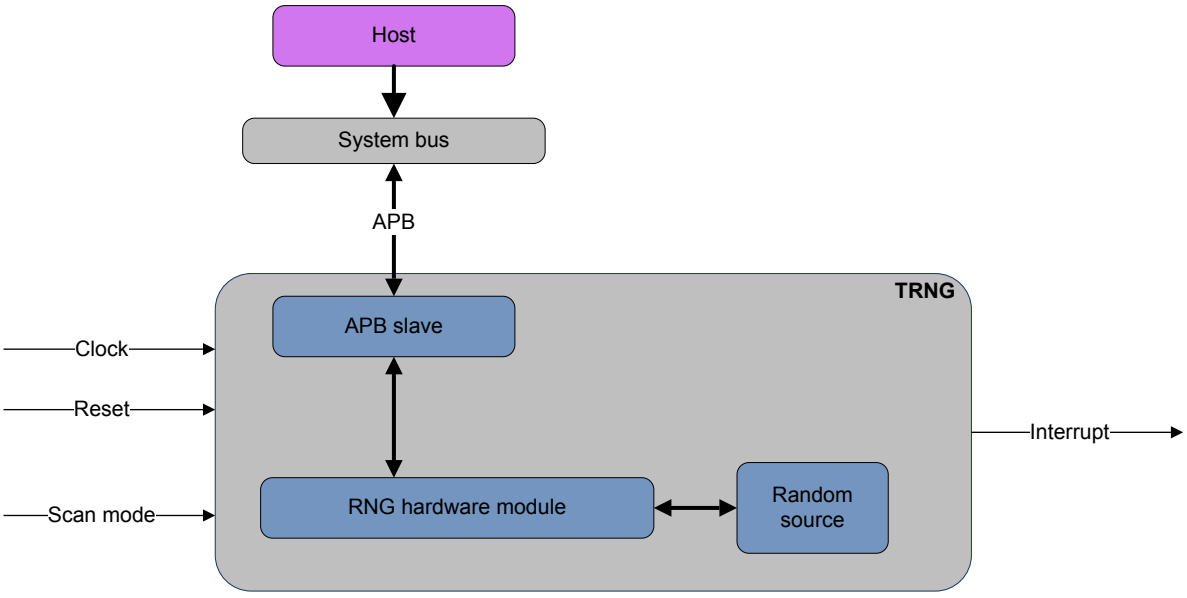

Figure 2-1 TRNG hardware overview

The interfaces of the TRNG are:

APB slave interface The TRNG connects as an APB slave on the SoC system bus. This interface enables a host processor to access the TRNG.

Clock and reset The TRNG clock and reset inputs.

Interrupt A TRNG output that connects to an interrupt on the host processor.

Scan Scan interface.

Chapter 3
Programmers model

This chapter describes the TRNG register addresses and functionality for integration.

ARM provides firmware with the TRNG product, to simplify the use of these registers.

——— **Note** ———

The TRNG product bundle includes header files for TRNG register offsets (`rng_hw_defs.h`).

———————————

It contains the following sections:

3.1 About the programmers model

The following information applies to the TRNG registers:

- The base address is not fixed, and can be different for any particular system implementation. The offset of each register from the base address is fixed.
- All TRNG registers are 32-bit.

The following table lists the access types that *3.2 Register summary* on page 3-20 shows.

Table 3-1 Access permissions

Access type	Description
RO	Read only.
RW	Read and write.
RWs	Read and write, but the register changes according to internal state.
WO	Write only.

3.2 Register summary

The following table lists the registers in the TRNG.

Table 3-2 TRNG register summary

Offset	Name	Access[a]	Reset value	Description
0x000-0x0FC	-	-	-	Reserved.
0x100	RNG_IMR	RWs	0x0000000F	*3.3 Interrupt Mask Register, RNG_IMR* on page 3-21
0x104	RNG_ISR	RO	0x00000000	*3.4 Interrupt Status Register, RNG_ISR* on page 3-22
0x108	RNG_ICR	WO	0x00000000	*3.5 Interrupt Clear Register, RNG_ICR* on page 3-23
0x10C	TRNG_CONFIG	RW	0x00000000	*3.6 Configuration register, TRNG_CONFIG* on page 3-24
0x110	TRNG_VALID	RO	0x00000000	*3.7 Valid register, TRNG_VALID* on page 3-25
0x114-0x128	EHR_DATA0- EHR_DATA5	RO	0x00000000	*3.8 Entropy Holding Register Data registers, EHR_DATA[0,1,2,...5]* on page 3-26
0x12C	RND_SOURCE_ENABLE	RW	0x00000000	*3.9 Random Source Enable register, RND_SOURCE_ENABLE* on page 3-27
0x130	SAMPLE_CNT1	RW	0x0000FFFF	*3.10 Sample Count register, SAMPLE_CNT1* on page 3-28
0x134	AUTOCORR_STATISTIC	RWs	0x00000000	*3.11 Autocorrelation register, AUTOCORR_STATISTIC* on page 3-29
0x138	TRNG_DEBUG_CONTROL	RO	0x00000000	*3.12 Debug Control register, TRNG_DEBUG_CONTROL* on page 3-30
0x13C	-	-	-	Reserved.
0x140	TRNG_SW_RESET	WO	0x00000000	*3.13 Reset register, TRNG_SW_RESET* on page 3-31
0x144-0x1B4	-	-	-	Reserved.
0x1B8	TRNG_BUSY	RO	0x00000000	*3.14 Busy register, TRNG_BUSY* on page 3-32
0x1BC	RST_BITS_COUNTER	WO	0x00000000	*3.15 Reset Bits Counter register, RST_BITS_COUNTER* on page 3-33
0x1C0-0x1DC	-	-	-	Reserved.
0x1E0-0x1E8	RNG_BIST_CNTR0..2	RO	0x00000000	*3.16 BIST Counter registers, RNG_BIST_CNTR[0, 1, 2]* on page 3-34
0x1EC-0x1FC	-	-	-	Reserved.

[a] See *Table 3-1 Access permissions* on page 3-19 for more information.

3.3 Interrupt Mask Register, RNG_IMR

The RNG_IMR register enables you to mask certain conditions and prevent the assertion of the interrupt output.

The following table lists the RNG_IMR bit assignments.

Table 3-3 RNG_IMR

Bits	Name	Description
[31:4]	-	Reserved.
[3]	VN_ERR_INT_MASK	Set to 1 to mask the Von Neumann error, and prevent the TRNG from setting the **cc_host_int_req** interrupt HIGH for Von Neumann errors.
[2]	CRNGT_ERR_INT_MASK	Set to 1 to mask the CRNGT error, and prevent the TRNG from setting the **cc_host_int_req** interrupt HIGH for CRNGT errors.
[1]	AUTOCORR_ERR_INT_MASK	Set to 1 to mask the Autocorrelation error, and prevent the TRNG from setting the **cc_host_int_req** interrupt HIGH for Autocorrelation errors.
[0]	EHR_VALID_INT_MASK	Set to 1 to mask when the TRNG has collected 192 bits, and prevent the TRNG from setting the **cc_host_int_req** interrupt HIGH when 192 bits are collected.

3.4 Interrupt Status Register, RNG_ISR

The RNG_ISR register returns the status of the interrupts.

If the corresponding RNG_IMR bit is unmasked, then an interrupt is generated.

The following table lists the RNG_ISR bit assignments.

Table 3-4 RNG_ISR

Bits	Name	Description
[31:4]	-	Reserved.
[3]	VN_ERR	When set to 1 it indicates a Von Neumann error. A Von Neumann error occurs if 32 consecutive collected bits are identical, that is, 32 zeros or 32 ones.
[2]	CRNGT_ERR	When set to 1, it indicates a *Continuous Random Number Generation Testing* (CRNGT) error in the TRNG test failed. Failure occurs when two consecutive blocks of 16 collected bits are equal.
[1]	AUTOCORR_ERR	When set to 1, it indicates that the Autocorrelation test failed four times in a row. When set, the TRNG stops functioning until the next reset.
[0]	EHR_VALID	Set to 1 when 192 bits have been collected in the TRNG, and the EHR_DATA[0, 1, 2,…5] registers are ready to be read.

3.5 Interrupt Clear Register, RNG_ICR

The RNG_ICR register enables the host processor to clear the interrupts.

The following table lists the RNG_ICR bit assignments.

Table 3-5 RNG_ICR

Bits	Name	Description
[31:4]	-	Reserved.
[3]	VN_ERR	Set to 1, to clear a Von Neumann error.
[2]	CRNGT_ERR	Set to 1, to clear a *Continuous Random Number Generation Testing* (CRNGT) error.
[1]	AUTOCORR_ERR	Software cannot clear this bit. Only a TRNG reset can clear this bit.
[0]	EHR_VALID	Set to 1 after the EHR_DATA[0,1,2,…5] registers have been read.

3.6 Configuration register, TRNG_CONFIG

The TRNG_CONFIG register controls the length of the inverter chain in the ring oscillator.

The following table lists the TRNG_CONFIG bit assignments.

Table 3-6 TRNG_CONFIG

Bits	Name	Description
[31:2]	-	Reserved.
[1:0]	RND_SRC_SEL	Selects the number of inverters (out of four possible selections) in the ring oscillator (the entropy source): `0b00` = Selects the shortest inverter chain length. This is the reset value. `0b01` = Selects the short inverter chain length. `0b10` = Selects the long inverter chain length. `0b11` = Selects the longest inverter chain length.

3.7 Valid register, TRNG_VALID

The TRNG_VALID register indicates when the EHR_DATA[0-5] registers contain 192 bits of valid data.

The following table lists the TRNG_VALID bit assignments.

Table 3-7 TRNG_VALID

Bits	Name	Description
[31:1]	-	Reserved.
[0]	EHR_VALID	When set to 1, it indicates that the collection of bits in the TRNG is complete, and data can be read from the EHR_DATA[0-5] registers.

3.8 Entropy Holding Register Data registers, EHR_DATA[0,1,2,…5]

Each of the EHR_DATA registers returns 32 bits from the 192-bit *Entropy Holding Register* (EHR). The EHR contains the generated random number.

The following table lists the EHR_DATA[0-5] bit assignments.

Table 3-8 EHR_DATA

Bits	Name	Description	
[31:0]	EHR_DATA	Returns 32 bits from the 192-bit EHR:	
		EHR_DATA0.EHR_DATA	Returns bits[31:0] of Entropy Holding Register.
		EHR_DATA1.EHR_DATA	Returns bits[63:32] of Entropy Holding Register.
		EHR_DATA2.EHR_DATA	Returns bits[95:64] of Entropy Holding Register.
		EHR_DATA3.EHR_DATA	Returns bits[127:96] of Entropy Holding Register.
		EHR_DATA4.EHR_DATA	Returns bits[159:128] of Entropy Holding Register.
		EHR_DATA5.EHR_DATA	Returns bits[191:160] of Entropy Holding Register.

3.9 Random Source Enable register, RND_SOURCE_ENABLE

The RND_SOURCE_ENABLE registers controls whether the entropy source, ring oscillator, is enabled.

The following table lists the RND_SOURCE_ENABLE bit assignments.

Table 3-9 RND_SOURCE_ENABLE

Bits	Name	Description
[31:1]	-	Reserved.
[0]	RND_SRC_EN	1 = The entropy source, ring oscillator, is enabled. 0 = The entropy source is disabled. This is the reset value.

3.10 Sample Count register, SAMPLE_CNT1

The SAMPLE_CNT1 register controls how often the TRNG samples the single output bit of the ring oscillator.

The following table lists the SAMPLE_CNT1 bit assignments.

Table 3-10 SAMPLE_CNT1

Bits	Name	Description
[31:0]	SAMPLE_CNTR1	Sets the number of **rng_clk** cycles between two consecutive ring oscillator samples. ——— **Note** ——— If the Von Neumann balancer is bypassed, the minimum value for sample counter must not be less than 17. Due to the autocorrelation test, the data bus from the collector must stay steady for 17 **rng_clk** clocks. ———

3.11 Autocorrelation register, AUTOCORR_STATISTIC

The AUTOCORR_STATISTIC register returns statistics about the autocorrelation test activations.

The following table lists the AUTOCORR_STATISTIC bit assignments.

Table 3-11 AUTOCORR_STATISTIC

Bits	Name	Description
[31:22]	-	Reserved.
[21:14]	AUTOCORR_FAILS	Count each time an autocorrelation test fails. Any write to the register resets the counter. Stops collecting statistic if one of the counters reaches the limit.
[13:0]	AUTOCORR_TRYS	Count each time an autocorrelation test starts. Any write to the register resets the counter. Stops collecting statistic if one of the counters reaches the limit.

3.12 Debug Control register, TRNG_DEBUG_CONTROL

The TRNG_DEBUG_CONTROL register controls the debug behavior of the TRNG.

The following table lists the TRNG_DEBUG_CONTROL bit assignments.

Table 3-12 TRNG_DEBUG_CONTROL

Bits	Name	Description
[31:4]	-	Reserved.
[3]	AUTO_CORRELATE_BYPASS	When set to 1, the autocorrelation test in the TRNG module is bypassed.
[2]	TRNG_CRNGT_BYPASS	When set to 1, the CRNGT test in the TRNG is bypassed.
[1]	VNC_BYPASS	When set to 1, the Von Neumann balancer is bypassed (including the 32 consecutive bits test).
[0]	-	Reserved.

3.13 Reset register, TRNG_SW_RESET

The TRNG_SW_RESET register enables software to reset the TRNG.

The following table lists the TRNG_SW_RESET bit assignments.

Table 3-13 TRNG_SW_RESET

Bits	Name	Description
[31:1]	-	Reserved.
[0]	TRNG_SW_RESET	Writing 1 to this register causes an internal TRNG reset.

3.14 Busy register, TRNG_BUSY

The TRNG_BUSY register indicates when the TRNG is busy.

The following table lists the TRNG_BUSY bit assignments.

Table 3-14 TRNG_BUSY

Bits	Name	Description
[31:1]	-	Reserved.
[0]	TRNG_BUSY	Reflects the status of the **rng_busy** signal.

3.15 Reset Bits Counter register, RST_BITS_COUNTER

The RST_BITS_COUNTER register resets the collected bits counter in the TRNG.

The following table lists the RST_BITS_COUNTER bit assignments.

Table 3-15 RST_BITS_COUNTER

Bits	Name	Description
[31:1]	-	Reserved.
[0]	RST_BITS_COUNTER	Writing any value to this bit resets the bits counter and TRNG valid registers, provided that RND_SOURCE_ENABLE.RND_SRC_EN==0.

3.16 BIST Counter registers, RNG_BIST_CNTR[0, 1, 2]

The RNG_BIST_CNTR[0, 1, 2] registers return the collected BIST results.

The following table lists the RNG_BIST_CNTR[0, 1, 2] bit assignments.

Table 3-16 RNG_BIST_CNTR

Bits	Name	Description
[31:22]	-	Reserved.
[21:0]	ROSC_CNTR_VAL	Returns the results of the TRNG BIST counter.

Chapter 4
Signal descriptions

This appendix describes the top-level signals of the TrustZone TRNG True Random Number Generator.

It contains the following sections:

4.1 Clocks and resets

The following table lists the clock and reset signals.

Table 4-1 Clock and reset signals

Signal	Direction	Description
rst_n	Input	Asynchronous global reset (active LOW).
rng_clk		TRNG engine clock.

4.2 APB slave interface

The APB slave interface enables a host processor to access the TrustZone TRNG.

The following table lists the APB slave interface signals.

Table 4-2 APB slave interface

Signal	Direction	Description
cc_psel	Input	Peripheral select signal.
cc_penable		Indicates that the enable cycle is taking place.
cc_paddr[11:0]		Peripheral address bus (TRNG address space requires 4KBytes).
cc_pwrite		Peripheral write signal.
cc_pwdata[31:0]		Peripheral write data bus.
cc_prdata[31:0]	Output	Peripheral read data bus.

Related information

ARM® AMBA® Specification (Rev 2).

4.3 Interrupts

The following table lists the interrupt signals.

Table 4-3 Interrupts

Signal	Direction	Description
cc_host_int_req	Output	This signal connects to an interrupt on the host processor. Asserted HIGH until the host processor acknowledges the interrupt by writing to the RNG_ICR register.

Related concepts

3.5 Interrupt Clear Register, RNG_ICR on page 3-23.

4.4 Scan signals

The following table lists the scan test signals.

Table 4-4 Scan signals

Signal	Direction	Description
scanmode	Input	When set, the TRNG operates in scan mode.

Appendix A
Revisions

Read this for a description of the technical changes between released issues of this book.

It contains the following section:

- *A.1 Revisions* on page Appx-A-41.

A.1 Revisions

This appendix describes changes between released issues of this book.

Table A-1 Issue 0000-00

Change	Location	Affects
First release	-	-