SP 800-90B Non-Proprietary Public Use Document
CDI CPU Time Jitter Based Non-Physical Entropy Software v1.0.0

Communication Devices Inc.
85 Fulton St # 2
Boonton, NJ 07005

Document Version 1.1
October 23rd 2023

Table of Contents

Description

The CDI CPU Time Jitter Based Non-Physical Entropy Source is a non-physical entropy source. The entropy source was tested under the assumption that the output is non-IID on the operational environment listed in Table 1.

Platform	Operating System	Processor
PA111- SA	Linux 4.14	i.MX6 Ultralite (NXP, IMX6, ARM32)

Table 1: Operational environment

Security Boundary

The security boundary and design of the entropy source are shown in Figure 1. The security boundary of the entropy source is a shared library containing API functions that are called by the Port Authority Series Cryptographic Modules to request entropy from the entropy source to seed an approved SP 800-90A DRBG which is outside the boundary of the entropy source.

The entropy source gets its entropy from the time deltas between repeated hash and memory collection operations. The entropy source uses an internal timer, provided by the shared library, supports an external CPU high resolution timer or the internal timer provided by the shared library.

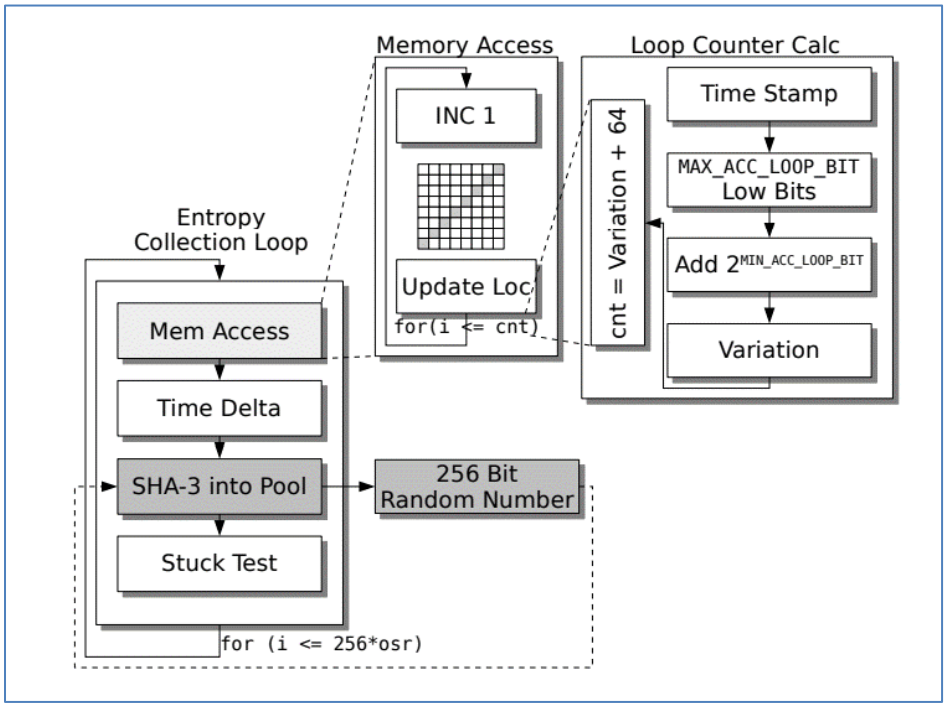

Figure 1: Security Boundary of the Jitter Entropy Source

Operating Conditions

The following table summarizes the operating conditions for the tested platform.

Parameter	Value	Description
Temperature	-40 to 85C	industrial operating temperature range
Voltage	3.1V to 5.5V	After boot should stabilize at 5V
Clock speed	198MHz-528MHz	When performing actions should be closer to 528MHz

Table 2: Operating Conditions

Configuration Settings

The following table summarizes the non-default configuration parameters explicitly set during initialization of the entropy source.

Parameter	Value	Description
JENT_FORCE_INTERNAL_TIMER	false	External CPU timer is used
JENT_FORCE_FIPS	true	Extra precaution to ensure parameters are set to FIPS values

Table 3: Configuration Parameters

Physical Security Mechanisms

The entropy source is non-physical. The physical security mechanisms only apply to the hardware component of the operational environment in which the entropy source is installed, and thus the entropy source inherits the physical security mechanisms of the Port Authority Series Cryptographic Module.

Conceptual Interfaces

The entropy source provides the following conceptual interfaces:

GetEntropy

The jent_read_entropy function corresponds to the GetEntropy interface from SP 800-90B. The caller specifies the size of the random value that it wants.

GetNoise

The jent_measure_jitter function corresponds to the GetNoise interface from SP 800-90B. The function calculates time deltas and uses the CPU jitter in the time deltas. The jitter is injected into the entropy pool.

HealthTest

The entropy source does not have a dedicated HealthTest interface, but health tests can be run by instantiating the entropy source via jent_entropy_collector_alloc. This is in compliance with SP 800-90B.

Min-Entropy Rate

In accordance with the Entropy Analysis Report, the vetted conditioned output of this entropy source is full-entropy. This entropy source outputs 256-bit conditioned output block. The Entropy Analysis Report supports the claim of 256 bits of min entropy per 256-bit conditioned output block.

Health Tests

Start-up health tests:

The Jitter RNG conducts the same set of continuous health tests on 1024 samples of noise data. The data is discarded after the start-up health tests have been completed successfully.

Continuous health tests:

The Jitter RNG implements the following health tests:

- Stuck Test
- Repetition Count Test
- Adaptive Proportion Test
- Lag Predictor Test

Those tests are detailed in the following sections.

Stuck Test

This test calculates the first, second and third discrete derivative of the time to be processed by the hash. Only if all three values are non-zero, the received time delta is considered to be non-stuck.

Repetition Count Test

This test conforms to SP 800-90B section 4.4.1. When any health test fails, the API call to generate random numbers jent_read_entropy informs the caller about the failure with error codes. The RCT is applied with $\alpha = 2^{-30}$ compliant to the recommendation of FIPS 140-2 IG 9.8.

Adaptive Proportion Test

This test conforms to SP 800-90B section 4.4.2. Considering that the entropy is present in the least significant bits of the time delta, the APT is applied only to the four least significant bits of the time delta with the cut-off value C = 325 for non-binary data.

Lag Predictor Test

This test is a vendor-defined conditional test that is designed to detect a known failure mode where the result becomes mostly deterministic.

On-demand health tests

On-demand health tests may be performed by rebooting the operational environment, which results in the immediate execution of the start-up tests or by deallocating and reallocating a new Jitter RNG handle.

All health test failures are considered permanent failures. If one is triggered, the current instance of the entropy source will always remain in error state.

Maintenance

There is no maintenance requirements for this software-based entropy source.

Required Testing

Raw noise data samples consisting of at least 1,000,000 bits shall be collected from the operational environment at the normal operating conditions and processed by the SP 800-90B entropy tool that is provided by NIST.

The following test were performed and included in the corresponding entropy report:

1. Raw noise data through the raw noise source interface and processed by the SP800-90B tool to obtain an entropy rate which must be near equal to or the defined min-entropy rate.
2. Obtain the restart noise data through the raw noise source interface and processed by the SP800-90B tool.
    a. the sanity test to apply to the noise restart data of 1000 samples must pass, and
    b. the minimum of the row-wise and column-wise entropy rate shall not be less than half of the entropy rate from 1 above.

Both tests are performed using the test utility (included in the source code) called 'jitterentropy-hashtime' in different modes. The results after performing these tests using an 8-bit symbols length are:

- Raw Testing Result

Processor	H_original	H_bitstring	Min (H_original, 8*H_bitstring)
iMX6ul	6.638399	0.740171	5.921367

- Restart Test Result

Processor	H_r	H_c	H_l	Result
iMX6ul	6.638399	6.629040	0.333000	**Validation Test Passed…**