Hello, and welcome to this presentation of the STM32 Random Number Generator. The features of this peripheral, which is widely used to provide random numbers, will be covered in this presentation.

Overview

- Provides random numbers
  - Used when producing an unpredictable result is desirable.

Application benefits

- Increase the randomness of numbers
- Decrease the possibility of guessing values

life.augmented

The random number generator (RNG) integrated inside STM32 products provides random numbers which are used when producing an unpredictable result is desirable. Applications can benefit from the RNG to increase the randomness of numbers or to decrease the possibility of guessing certain values.

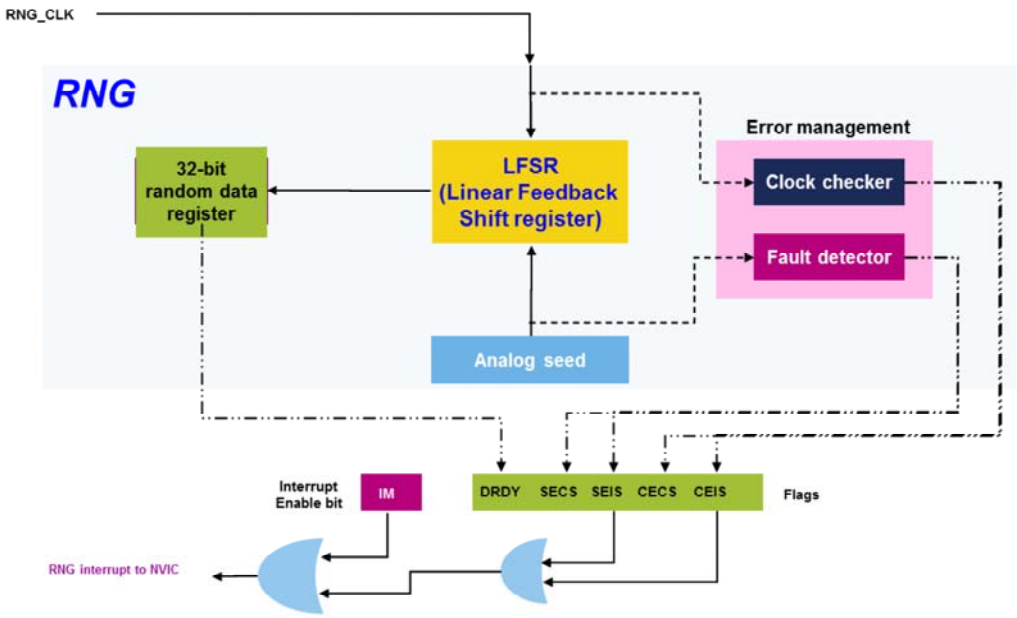
Key features

- 32-bit Random Number Generator based on a noise source.
  - A 32-bit random number can be generated at an average frequency of AHB/54.
  - Can be disabled to reduce power consumption.
- 3 Flags can be triggered when
  - valid random data is ready.
  - an abnormal sequence occurs on the seed (more than 64 consecutive bits having the same value or 32 consecutive alternating 0s and 1s) .
  - a frequency error is detected when using a PLL48 RNG clock source
- 1 interrupt
  - To indicate an error (an abnormal seed sequence or a frequency error).

life.augmented

The RNG peripheral is based on continuous analog noise that provides a random 32-bit value which will be explained in detail later on. The RNG is able to generate a 32-bit random number at an average frequency of AHB/54. A flag is set in the Data register when new random data is ready and validated. The RNG verifies the randomness of the provided data; if more than 64 consecutive bits have the same value (0 or 1) or there are more than 32 consecutive alternating 0s and 1s, a seed error current status flag is set. When a PLL48 RNG clock source is used, a clock error current status flag is set if the PLL48 clock frequency is less than divided by 32.
An interrupt source can also be enabled to indicate an abnormal seed sequence or frequency error.

Block diagram



life.augmented

This simplified block diagram of the RNG shows its basic functional and control modules.

The random number generator is based on an analog circuit made of several ring oscillators whose outputs are XORed to generate the seeds that feed a linear feedback shift register in order to produce 32-bit random numbers.

The linear feedback shift register is clocked by a dedicated RNG clock signal so that the quality of the random number is independent of the HCLK frequency. The contents of the linear feedback shift register is transferred into the data register when a significant number of seeds have been introduced into the LFS register.

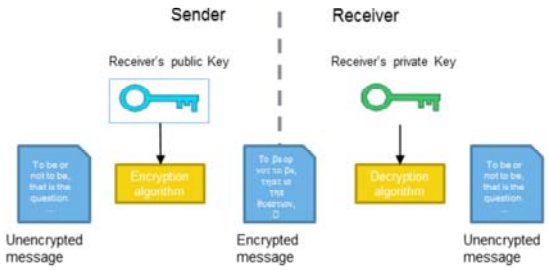
In parallel, an Error Management block verifies the correct seed behavior and the frequency of the RNG source clock if a PLL48 source is used.

Status bits are set and an interrupt is triggered if an abnormal sequence is detected in the seed or if the frequency of the

PLL48 RNG clock is too low.

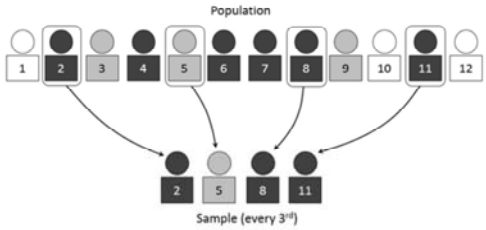
Application examples

- Cryptography



- Games

- Statistical sampling



life.augmented

The RNG can be used for a wide range of applications including cryptography, games, and statistical sampling. For example, all the security of cryptography algorithms are connected to the impossibility of guessing the key. So the key has to be a random number, otherwise the attacker can guess it.

Related peripherals

- Peripherals linked to the RNG
  - RCC (RNG clock control, RNG enable/reset)
  - Interrupts (RNG interrupt mapping)

life.augmented

This is a list of peripherals related to the random number generator. Please refer to these peripheral trainings for more information if needed.

References

- AN4230: STM32 microcontrollers random number generation validation using NIST statistical test suite.
  - AN4230 provides guidelines to verify the randomness of numbers generated by an STM32 embedded random number generator. This verification is based on the National Institute of Standards and Technology (NIST) Statistical Test Suite (STS) SP 800-22, which was published and updated recently as SP800-22rev1a (April 2010).
  - The NIST test suite was run on both the STM3220G-EVAL, rev B and the STM3240G-EVAL, rev B boards. The results are provided in the firmware folder 'NIST_Test_Suite_OutputExample'.

life.augmented

For more details, please refer to application note AN4230 about using the NIST statistical test suite to validate the random numbers generated by STM32 MCUs.