Generación de números aleatorios en arquitecturas de procesadores

Agustín Bermúdez, Mateo Gonzalez Pautaso, Tiago André Calderón

[1] ***Abstract—*** Este informe examina las principales técnicas usadas en generadores de números, tanto pseudoaleatorios como realmente aleatorios, centrándose en sus implementaciones sobre el hardware de los procesadores.
Se discuten las limitaciones actuales que presentan, principalmente para la aleatoriedad real y las posibles modificaciones futuras para optimizar los generadores en arquitecturas futuras.
Además, se analizan sus ventajas, desventajas y aplicaciones, destacando la importancia de la calidad y la impredecibilidad de los números generados.

Index Terms— criptografía, entropía, números realmente aleatorios, números pseudoaleatorios, PRN, PRNG, TRN, TRNG.

1. INTRODUCCIÓN

En los últimos años la importancia de la generación de números aleatorios fue creciendo a medida que su aplicación en distintos campos fue volviéndose más esencial. Uno de los más importantes es la seguridad informática, donde se usan ampliamente en criptografía. Otros ejemplos de campos donde son necesarios los números aleatorios son las simulaciones que necesitan impredecibilidad o variabilidad controlada, también son usados con fines recreativos para representar el azar.

Dada su amplia utilización, se han desarrollado generadores de números aleatorios implementados en hardware para garantizar mayor fiabilidad y eficiencia. Sin embargo, los procesadores modernos, por su naturaleza determinística, enfrentan desafíos para lograr una verdadera aleatoriedad. Esta limitación ha llevado a la clasificación de los números aleatorios en dos categorías principales:

- Pseudoaleatorios (PRN): Se basan en una semilla inicial donde se utilizan algoritmos matemáticos para obtenerlos. Siguen un patrón si se conoce la semilla. Son determinísticos.
- Realmente aleatorios (TRN): Se obtienen aprovechando fenómenos físicos, como ruido térmico en circuitos electrónicos para garantizar la impredecibilidad. No son determinísticos.

[1] Bermúdez, Agustín, Ingeniería informática, Universidad de Buenos Aires (mail: abermudez@fi.uba.ar).
Gonzalez Pautaso, Mateo, Ingeniería informática, Universidad de Buenos Aires (mail: magonzalezp@fi.uba.ar).
Calderón, Tiago André, Universidad de Buenos Aires (mail: tcalderon@fi.uba.ar).

2. IMPLEMENTACIONES EN ARQUITECTURAS

Actualmente la mayoría de las arquitecturas modernas cuentan con hardware especializado para la generación de números PRN y TRN.

En este informe se va a profundizar en uno de los más conocidos para explicar el funcionamiento de los generadores pseudoaleatorios y generadores realmente aleatorios. Este es el módulo DRNG [1] presente en los procesadores Intel desde 2012 con las versiones Ivy Bridge.

2.1 COMPONENTES DRNG

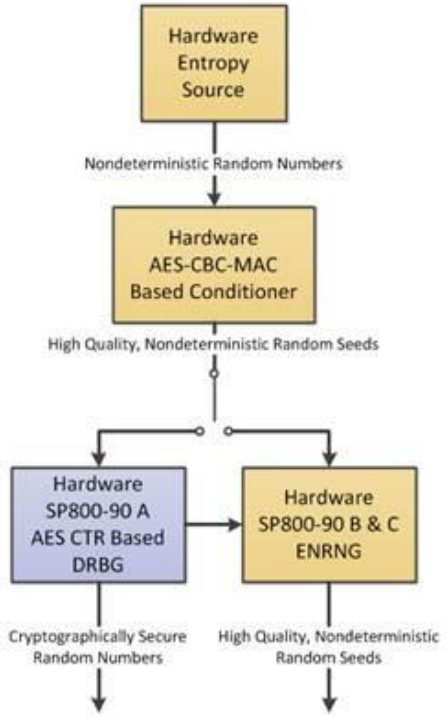

Fig. 1. Flujo generador números aleatorios [2]

2.1.1 Entropy Source (ES)

La fuente de entropía es el componente encargado de producir los datos aleatorios. El fenómeno físico en el que se basa es en el ruido térmico presente dentro del silicio del procesador, que se transforma en bits aleatorios a una velocidad de 3 GHz.

Para garantizar su robustez, está completamente implementado en hardware(consta de componentes lógicos y electrónicos delimitados por un rectángulo de silicio); no depende de firmware ni software. Además está preparado para garantizar su

buen funcionamiento bajo un amplio rango de temperaturas y voltajes excediendo el rango de operación del procesador [Fig 2].

Otra característica que aumenta la impredecibilidad es que la Entropy Source no depende del clock del procesador. En su lugar, utiliza un circuito auto-cronometrado que genera bits de manera independiente del resto del sistema.[2][3]

Parameter	Minimum	Maximum
Temperature	-25C	90C
Voltage	0.8V – 5%	0.8V + 5%
Clock frequency	~1.6GHz	~1.6GHz

Fig. 2. Condiciones de operación [3]

2.1.2 Based Conditioner

El acondicionador es el encargado de estabilizar los bits sin procesar provenientes de la fuente de entropía y reducirlos a muestras de 256 bits con el fin de que sean estadísticamente más robustos, cumpliendo los los estándares de calidad AES [4]

2.1.3 DRBG (Deterministic Random Number Generator)

La función del generador determinístico de bits random es propagar muestras de entropía acondicionadas en un set grande de valores aleatorios, así aumentando la cantidad de números disponibles para el módulo de hardware. Esto se logra utilizando el generador llamado CTR_DRBG que cumple con estándares [5] y cambiando la semilla periódicamente con las muestras mencionadas anteriormente.

Los valores producidos responden las solicitudes de números pseudoaleatorios (PRN) mediante la instrucción RDRAND.

El DRBG no puede generar más de 1022 números aleatorios secuenciales a partir de una misma semilla. [2]

2.1.4 ENRNG (Enhanced Non-deterministic Random Number Generator)

El rol del generador de números aleatorios no determinísticos es el de proveer al usuario de números realmente aleatorios (TRN) a los que puede acceder desde la instrucción RDSEED.

Los TRN que libera el generador son las muestras acondicionadas provenientes del Based Conditioner.

Además, el ENRNG utiliza valores generados por el DRBG para reforzar la calidad de la entropía, asegurando un nivel aún mayor de aleatoriedad.

2.2 FUNCIONAMIENTO DRNG

Dentro del módulo se encuentra un sub-sector llamado “Security Boundary” que a su vez contiene otro sub-sector que representa la fuente de entropía, representados con un borde en negrita y un borde rojo respectivamente [Fig 3].

El sector correspondiente a la fuente de entropía contiene el componente físico llamado “Digital Noise Source” que funciona a modo de ES [Fig 1]. Este digitalizador garantiza mantener el 70% de la entropía original superando el requisito mínimo del 29% necesario para el acondicionador, establecido por los estándares [5].

La señal proveniente del ruido térmico dentro del silicio es enviada a los siguientes componentes:

- CHT (Continuous Health Tests) [Fig 3]: encargado de realizar pruebas automatizadas para asegurar la calidad y fiabilidad de los datos, encendiendo una flag “healthy”, representando el estado del sistema.
- Acondicionador: recibe las muestras de entropía y la flag “healthy”,
    - Aplica el algoritmo AES-CBC-MAC, tomando bloques consecutivos de 256 bits como entrada.
    - Genera un único bloque con mayor uniformidad y calidad estadística.
    - Elimina correlaciones y sesgos presentes en los datos brutos.

Luego, la salida del acondicionador se dirige a uno de los dos destinos siguientes (la misma muestra no puede ir a ambos componentes):

1. CTR_DRBG: cuando se solicita un PRN la muestra funciona como semilla para el DRBG, que luego aplicará el algoritmo matemático correspondiente [5] para generar el conjunto de PRNs.
2. Compuerta XOR de 128 bits: cuando se solicita un TRN, la muestra es combinada con un PRN generado por el DRBG, siendo la salida de la compuerta el TRN generado.

Ambas operaciones para generar PRNs o TRNs son de 128 bits de tamaño, sin embargo el tamaño de los registros presentes en los CPU que integran el módulo DRNG son de 64 bits. Para poder usar todo el tamaño del número generado se emplea un registro de 128 bits con política FIFO que libera el número separado en dos salidas de 64 bits correspondiendo a los registros.

Los componentes que están fuera del sub-sector “Security Boundary” son los que están conectados al clock, BUS y alimentación del CPU, no están aislados. [3]

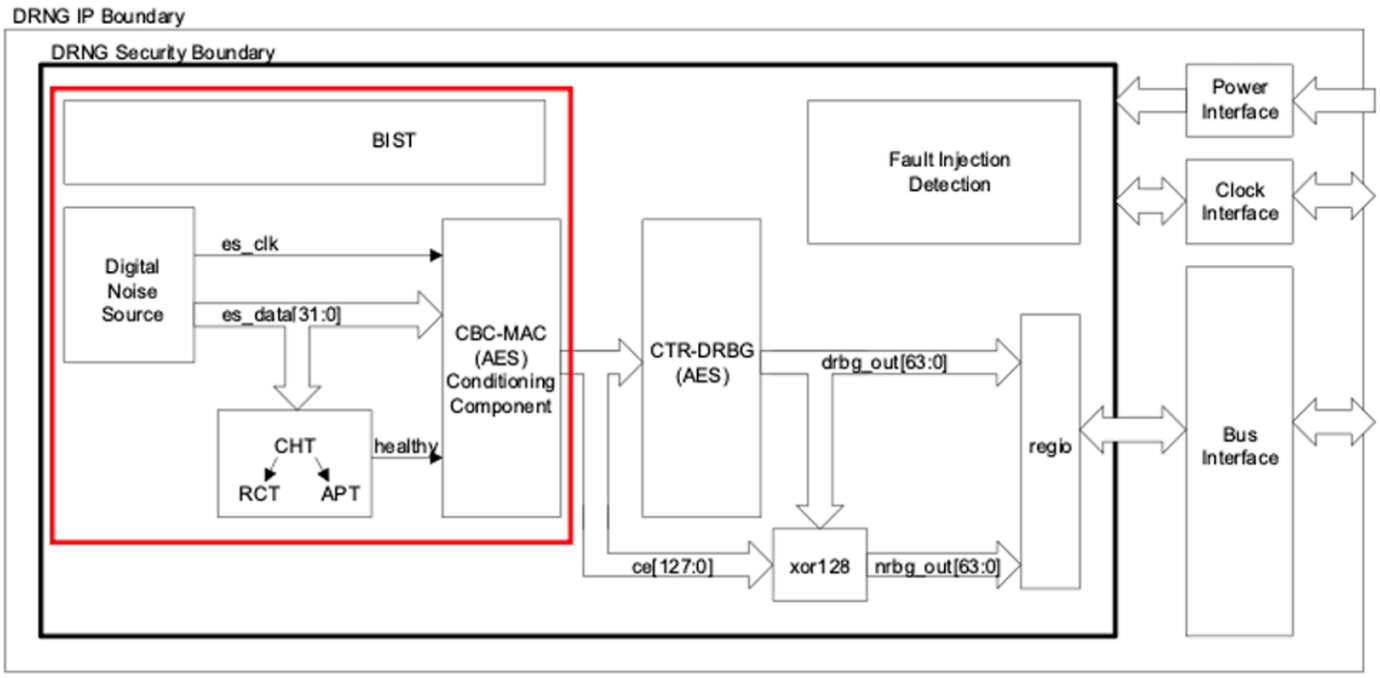

Fig. 3. Módulo DRNG [3]

3. SEGURIDAD Y FIABILIDAD

Al generar números aleatorios, es fundamental maximizar su seguridad y fiabilidad para garantizar una mayor imprevisibilidad. Esto evita que las aplicaciones que los utilizan desarrollen patrones predecibles o susceptibles a ser explotados. Por lo tanto, al incrementar la calidad de los números aleatorios, fruto de estas dos variables, se mejora tanto la robustez como la precisión de dichas aplicaciones.

En este documento se desarrollan las implementaciones en el módulo DRNG de INTEL [1] a modo de ejemplo de las tecnologías que se están usando para garantizar estos aspectos.

3.1 FIABILIDAD DEL DRNG

El DRNG contiene determinados elementos para asegurar su fiabilidad, estos son:

- CHT (Continuous Health Tests): se ejecutan continuamente evaluando la calidad de la fuente de entropía. Consiste en dos pruebas:
  - Corto plazo: valida los bloques individuales de 256 bits de la fuente de ruido térmico. Son verificaciones del tipo éxito/fallo.
  - Largo plazo: usa el historial de los 256 bloques (65536 bits) previamente evaluados por la prueba de corto plazo para determinar un índice de fallo.

  La condición de falla del CHT se da cuando el índice de largo plazo cae debajo de 50%. En esta situación el módulo deja de producir números aleatorios y se resetea a sí mismo hasta recuperar la calidad de la entropía. [3]
- BIST (Built-In Self Tests): se invocan cada vez que el sistema se inicia o reinicia. Estos se completan antes de que el procesador pueda ejecutar las primeras instrucciones, así garantizando el estado operacional del módulo. Incluyen pruebas para verificar la eficiencia de la fuente de entropía (ES-BIST) y pruebas de respuesta conocida (KAT_BIST):
  - Logic Integrity Test: desempeña la prueba sobre la lógica digital mediante la comparación entre un conjunto de PRNs generados y sus resultados esperados.
  - Startup Noise Source Health Test: se ejecuta una prueba de la fuente de ruido térmico utilizando los CHTs en un período de prueba de 65536 bits. Si esta prueba resulta exitosa, el DRNG entra en estado operativo. [3]

3.2 SEGURIDAD DEL DRNG

El DRNG contiene varios mecanismos de seguridad física integrados en la lógica del Security Boundary, entre los cuales se encuentran:

- Sparse Coding (Codificación Dispersa): es un mecanismo que transforma representaciones de valores de n bits en representaciones más

largas de m bits (m > n). Luego se realiza una búsqueda algorítmica que maximiza la distancia de Hamming [6] entre las representaciones codificadas. Cuando un valor recibido no está en el conjunto de representaciones válidas, se activará una alarma. Este mecanismo se utiliza como una defensa contra ataques que intentan alterar el sistema mediante inyección de fallas.

- Arc Integrity (Integridad del arco): en cada etapa de las máquinas de estado dentro del límite de seguridad, se verifica si el estado previo pertenece a la lista de transiciones válidas hacia el estado actual. Si se detecta una violación de esta integridad, se activa una alarma. Esto protege contra ataques de inyección de fallos que intentan forzar a la máquina a un estado no permitido.
- SBOX Masking (Enmascaramiento de SBOX): en el motor AES [4], se implementa un enmascaramiento de SBOX [7] en hardware con máscaras fijas generadas aleatoriamente. Cada SBOX utiliza una máscara distinta como defensa básica contra ataques de canal lateral [8] relacionados con emisiones de SBOX.
- Consistency Checks (Verificaciones de consistencia): en el DRNG existen diversas condiciones, como modos de prueba, modos de operación, estados de buffer y de máquinas de estado. Algunas combinaciones de estos estados no deben coexistir, por ejemplo, un modo de prueba no puede estar activo cuando el DRNG está en modo normal y seguro. Si se da el caso de que alguna verificación falla, se activa una alarma.

Cuando se activa una alarma, el DRNG se resetea a sí mismo y vuelve a ejecutar el BIST. El módulo estará nuevamente operativo una vez concluida la ejecución del BIST sin errores, caso contrario, se repite el proceso. [3]

4. LIMITACIONES ACTUALES Y FUTURO

4.1 LIMITACIONES ACTUALES

A pesar de los grandes avances tecnológicos que se fueron dando durante los últimos años los generadores de números aleatorios, principalmente los TRNG, enfrentan desafíos significativos a la hora de mejorar su calidad. Esto se debe mayoritariamente a las grandes limitaciones con las que las implementaciones modernas cuentan.

Todos los TRNG actuales dependen de hardware especializado que usa fuentes físicas de entropía para respaldar la impredecibilidad de los números. Esta es una de las limitaciones más importantes ya que aunque son teóricamente impredecibles, en la práctica son sensibles a condiciones externas como la temperatura o el envejecimiento del hardware, lo que los puede volver manipulables, perdiendo su propósito.

Además, el hardware necesario para implementar TRNG es costoso debido a la complejidad de capturar la entropía física. A su vez, las fuentes de entropía frecuentemente generan datos con distribuciones no uniformes, lo que requiere etapas adicionales de procesamiento, como el uso del acondicionador presente en el módulos DRNG [3] de INTEL [1]. Este procesamiento extra incrementa la complejidad y el costo.

Algunas implementaciones también necesitan de mantenimiento y calibración. Aunque no todos lo requieren puede agregar un gasto adicional.

Dependiendo de las aplicaciones para las que se utilicen los números, los generadores deben adherirse y cumplir con ciertos estándares internacionales, como por ejemplo el SP800-90A [5]. Como estos necesitan hardware redundante y procesamiento adicional se elevan la complejidad y los costos de implementación.

Consecuentemente las limitaciones descritas también dificultan la escalabilidad de los TRNG, provocando un alto costo para generar grandes cantidades de TRN. Este problema es especialmente relevante en algunos sectores, como el de Blockchain [9], donde la demanda de TRNs es alta y está en rápido crecimiento. Por lo tanto, mejorar la eficiencia y escalabilidad de los TRNG presenta uno de los mayores desafíos para la industria.

4.2 FUTURO DE LA ALEATORIEDAD

Para solucionar las limitaciones mencionadas anteriormente, se están buscando diferentes enfoques mediante la investigación e inversión en el área.

La computación cuántica trae consigo una posible mejora para generar TRNs de gran calidad. Esto gracias a que la naturaleza de estos sistemas aprovecha fenómenos de la física cuántica que son verdaderamente aleatorios. Aunque esta tecnología aún está lejana, su uso podría proporcionar fuentes de entropía que no necesiten acondicionamiento.

Algunos sistemas modernos utilizan nuevas fuentes de entropía no convencionales en la búsqueda de TRNs con la mayor calidad posible. Una de las más llamativas es la combinación de múltiples fuentes de entropía de no tan alta calidad para reemplazar a una única fuente de alta calidad. Esto mejoraría los problemas que tienen dichas fuentes frente a condiciones externas. Generar un promedio de varias fuentes de entropía evita que ocurran periodos de baja entropía en contraste a cuando se usa una única fuente [Fig 4] .

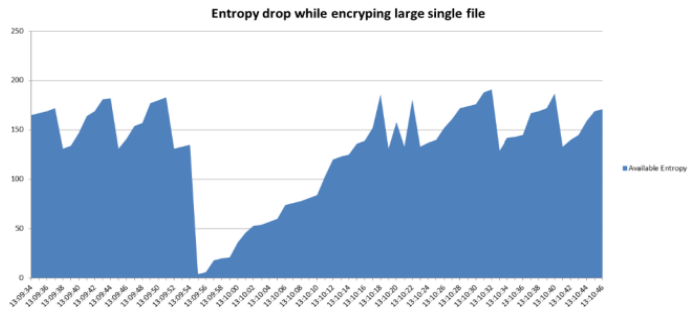

Fig. 4. Entropía disponible durante encriptación (única fuente) [10]

También se están buscando fuentes de entropía que garanticen una aleatoriedad que roce la perfección, por ejemplo:

- Biometría y fenómenos biológicos: como el ritmo cardíaco o los movimientos del ojo. Estos representan patrones imposibles de replicar debido a la variabilidad individual y la naturaleza biológica.
- Fenómenos astrofísicos: son datos capturados de observatorios espaciales como la radiación cósmica.
- Datos extraídos de interacciones humanas: se obtienen mediante redes sociales cómo likes o patrones de tráfico web.

Sin embargo, aunque estas fuentes prometen gran impredecibilidad, la tecnología actual sufre dificultades a la hora de recolectarla volviéndola muy costosa y no tan eficiente.

5. CONCLUSION

En la industria, la generación de números aleatorios tiene un rol crucial que cada vez toma más importancia en varios sectores, principalmente en campos como la seguridad y la criptografía. Este informe destaca una de las tecnologías utilizadas en hardware en la actualidad, que combina fuentes de entropía física y algoritmos matemáticos para alcanzar un balance entre seguridad, fiabilidad y rendimiento.

Sin embargo, las limitaciones actuales, como la dependencia de fuentes de entropía físicas y la dificultad para maximizar la impredecibilidad, reflejan la necesidad de seguir en la búsqueda de implementaciones que consigan mejores resultados sin sacrificar rendimiento. Actualmente, se están desarrollando métodos que combinan procesos determinísticos y no determinísticos para mejorar la calidad de los números generados, a la vez que optimizan el hardware utilizado.

En resumen, los resultados de las investigaciones de los generadores están mejorando no solo la calidad de los números generados, sino también a las aplicaciones que los requieren. Esto adquiere una relevancia fundamental en el contexto actual donde la privacidad de los datos es cada vez más importante. A medida que las tecnologías avanzan, los generadores evolucionan en los métodos para generar números realmente aleatorios buscando principalmente nuevas fuentes de entropía que mejoren la eficiencia, a la vez que satisfacen las crecientes demandas de sistemas más complejos y seguros.

AGRADECIMIENTOS

Ing. Marchi Edgardo, Ing. Cervetto Marcos y a todos los colaboradores de la cátedra Organización del Computador, Facultad de Ingeniería, Universidad de Buenos Aires.

REFERENCIAS

[1] INTEL CORPORATION, “DIGITAL RANDOM NUMBER GENERATOR” US 2010/0332574 A1, Dec. 30, 2010.
[2] Intel® Digital Random Number Generator (DRNG) Software Implementation Guide [Online]. Available: https://www.intel.com/content/dam/develop/external/us/en/documents/drng-software-implementation-guide-2-1-185467.pdf
[3] Intel DRNG SP800-90B Non-Proprietary Public Use Document Intel DRNG Entropy Source [Online]. Available: https://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/entropy/E57_PublicUse.pdf
[4] Advanced Encryption Standard (AES) [Online]. Available: https://www.intel.com/content/dam/develop/external/us/en/documents/fips-197.pdf
[5] Recommendation for Random Number Generation Using Deterministic Random Bit Generators (Revised). [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf
[6] A hybrid implementation of Hamming weight. [Online] Available: https://ieeexplore.ieee.org/document/6787256
[7] A Very Compact “Perfectly Masked” S-Box for AES (corrected). [Online]. Available: https://eprint.iacr.org/2009/011.pdf
[8] Introduction to Side-Channel Attacks. [Online] Available: https://perso.uclouvain.be/fstandae/PUBLIS/42.pdf
[9] The importance of Randomness to Blockchains and Web3. [Online] Available: https://orochi.network/blog/the-importance-of-randomness-to-Blockchains-and-Web3
[10] Encuentran una nueva forma de generar números aleatorios de alta calidad y mejorar los cifrados. [Online] Available: https://www.redeszone.net/2016/05/20/encuentran-una-nueva-forma-generar-numeros-aleatorios-alta-calidad-mejorar-los-cifrados/