Veri generatori di numeri casuali

Capitolo · Novembre 2014

DOI: 10.1007/978-3-319-10683-0_12

2 autori, Compreso:

Mario StipCeviC
Ru?ehm BoSkoviC Istituto

Veri generatori di numeri casuali

Mario Stipˇčeviˇč e Çetin Kaya Koç

Astratto I numeri casuali sono necessari in molte aree: crittografia, calcolo e simulazione Monte Carlo, test ed etichettatura industriale, giochi di rischio, gioco d'azzardo, ecc. La nostra ipotesi è stata che i numeri casuali non possono essere calcolati; poiché i computer operano in modo deterministico, non possono produrre numeri casuali. Invece, i numeri casuali si ottengono meglio utilizzando generatori di numeri casuali fisici (veri) (TRNG), che funzionano misurando un processo fisico ben controllato e appositamente preparato. La casualità di un TRNG può essere caratterizzata e misurata in modo preciso, scientifico. Particolarmente preziosi sono gli RNG dimostrabili sulla teoria dell'informazione, che, allo stato dell'arte, sembrano essere possibili solo sfruttando la casualità inerente a certi sistemi quantistici. D'altra parte, L'attuale standard di settore impone l'uso di RNG basati su oscillatori a corsa libera (FRO) la cui casualità è derivata dal rumore elettronico presente nei circuiti logici e che non può essere rigorosamente dimostrato come casualmente non informato, ma offre una più facile realizzazione tecnologica. L'approccio FRO è attualmente utilizzato nell'hardware FPGA e ASIC di 3a e 4a generazione, inadatto alla realizzazione di RNG quantistici. In questo capitolo confrontiamo gli aspetti deboli e quelli forti dei due approcci. Infine, discutiamo diversi esempi in cui l'uso di un vero RNG è fondamentale e mostriamo come può migliorare significativamente la sicurezza dei sistemi crittografici e L'approccio FRO è attualmente utilizzato nell'hardware FPGA e ASIC di 3a e 4a generazione, inadatto alla realizzazione di RNG quantistici. In questo capitolo confrontiamo gli aspetti deboli e quelli forti dei due approcci. Infine, discutiamo diversi esempi in cui l'uso di un vero RNG è fondamentale e mostriamo come può migliorare significativamente la sicurezza dei sistemi crittografici e L'approccio FRO è attualmente utilizzato nell'hardware FPGA e ASIC di 3a e 4a generazione, inadatto alla realizzazione di RNG quantistici. In questo capitolo confrontiamo gli aspetti deboli e quelli forti dei due approcci. Infine, discutiamo diversi esempi in cui l'uso di un vero RNG è fondamentale e mostriamo come può migliorare significativamente la sicurezza dei sistemi crittografici e

---

Mario StipˇčeviC̆
Rudjer Boˇškovi´ć Istituto
Zagabria, Croazia
&
Università della California Santa Barbara
Santa Barbara, California 93106, USA e-
mail: stipcevi@gmail.com

etin Kaya Ko¸ç
Università della California Santa Barbara
Santa Barbara, California 93106, USA e-
mail: koc@cs.ucsb.edu

discutere le sfide industriali e di ricerca che impediscono l'uso diffuso dei TRNG.

1. Introduzione

I veri numeri casuali e i generatori di numeri casuali fisici non deterministici (RNG) sembrano avere un'importanza sempre maggiore. I numeri casuali sono essenziali nella crittografia (matematica, stocastica e quantistica), nei calcoli Monte Carlo, nelle simulazioni numeriche, nella ricerca statistica, negli algoritmi randomizzati, nella lotteria, ecc. Oggi, i veri numeri casuali sono richiesti in modo più critico nella crittografia e nelle sue numerose applicazioni nella nostra vita quotidiana: comunicazioni mobili, accesso e-mail, pagamenti online, pagamenti senza contanti, bancomat, e-banking, commercio su Internet, punti vendita, carte prepagate, chiavi wireless, sicurezza informatica generale, sicurezza della rete elettrica distribuita (SCADA) ecc.

Senza perdita di generalità nel resto dell'articolo assumeremo che i generatori producano bit casuali.

Nelle applicazioni in cui la dimostrabilità è essenziale, anche le fonti di casualità (se coinvolte) devono essere casuali in modo dimostrabile, altrimenti l'intera catena di prove crolla. Nella crittografia, dove per il principio di Kerhoff tutte le parti dei protocolli sono pubblicamente note tranne alcuni segreti (la chiave o altre informazioni) noti solo al mittente e al destinatario, è chiaro che il segreto non deve essere calcolabile da un intercettatore, cioè deve Essere casuale. Ad esempio il noto protocollo di distribuzione della chiave quantistica BB84 [4] (descritto nella Sezione 3.4) sarebbe completamente insicuro se solo un intercettatore potesse calcolare (o prevedere) i numeri casuali di Alice o i numeri casuali di Bob o entrambi. Dall'analisi del tasso di chiave segreto presentato in esso è ovvio che qualsiasi prevedibilità di numeri casuali da parte dell'intercettatore gli farebbe trapelare informazioni rilevanti, diminuendo così il tasso di chiave effettivo. È intrigante [94] che nel caso in cui l'intercettatore potesse calcolare esattamente i numeri; il potenziale crittografico del protocollo BB84 sarebbe nullo. Infatti uno dei recenti attacchi di successo alla crittografia quantistica sfrutta la possibilità di controllare i QRNG locali sfruttando un difetto di progettazione di due sistemi crittografici quantistici commerciali e un sistema scientifico pratico. Questo esempio, discusso di seguito,

La lotteria è un'altra faccenda seria in cui i numeri casuali sono essenziali. A causa della grande somma di denaro coinvolta (stimata 6 miliardi di dollari all'anno solo online e solo negli Stati Uniti [40]), alcuni paesi hanno stabilito requisiti espliciti per i generatori di numeri casuali da utilizzare nelle macchine per il gioco d'azzardo e le lotterie online e hanno stabilito le autorità per il rilascio dei certificati. Ad esempio, la Lotteries and Gaming Authority (LGA) di Malta ha prescritto un elenco di requisiti per gli RNG, stipulato nella legge sui regolamenti sul gioco a distanza

[51]. Un RNG non conforme a questa legge non può essere legalmente utilizzato per attività di gioco d'azzardo. Queste regole sono state proposte per garantire un gioco leale da parte dei fornitori e per prevenire la possibilità che i giocatori manipolino il sistema prevedendo i risultati.

I generatori di numeri casuali sono stati un'occupazione di scienziati e inventori per molto tempo. Interi rami della matematica sono stati inventati per la necessità di comprendere i numeri casuali e il modo per ottenerli. All'inizio degli anni settanta, all'alba dell'era dell'informatica moderna, John von Neumann fu uno dei primi a notare che i computer di Turing deterministici non sono in grado di produrre veri numeri casuali e lo mise nella sua nota affermazione che "Chiunque consideri metodi aritmetici di produrre cifre casuali è, ovviamente, in uno stato di peccato".

I generatori di numeri casuali sono uno dei temi di ricerca più caldi dell'ultimo decennio. Ci sono stati circa 83 brevetti all'anno nell'ultimo decennio, 1418 in totale dal 1970 e innumerevoli articoli scientifici pubblicati sui veri generatori di numeri casuali. Tuttavia, una netta discrepanza tra il numero di pubblicazioni e il numero molto modesto di prodotti (solo 4 RNG quantistici e una manciata di RNG basati sul rumore Zener per lo più fuori produzione) che sono mai arrivati sul mercato [37, 38, 71, 68] chiaramente indica l'immaturità della maggior parte dell'arte. A nostro avviso i problemi principali sono la mancanza di prove di casualità e la scarsa riproducibilità della maggior parte delle soluzioni presentate finora. La ricerca della vera casualità continua.

2 generatori di numeri pseudocasuali

Storicamente, ci sono stati due approcci alla generazione di numeri casuali: algoritmico (pseudocasuale) e mediante un processo fisico (non deterministico).

I generatori di numeri pseudocasuali (PRNG) sono ben noti nell'arte e non li affronteremo qui in modo molto dettagliato. Indagini ed esempi individuali di PRNG possono essere trovati altrove [45, 109, 36, 55, 57]. In poche parole, un PRNG non è altro che una formula matematica, che produce una sequenza numerica deterministica e periodica, che è completamente determinata dallo stato iniziale chiamato seme. Per definizione, tali generatori non sono dimostrabili casuali. In pratica, i PRNG presentano un perfetto equilibrio tra 0 e 1 (zero bias) ma anche forti correlazioni a lungo raggio, che minano la forza crittografica e possono presentarsi come errori imprevisti nei calcoli e nei modelli Monte Carlo.

Sebbene la maggior parte dei PRNG moderni superi tutti i test statistici conosciuti, ci sono miti su alcuni PRNG che sono molto migliori degli altri. La verità è che ogni PRNG mostra la sua debolezza in qualche particolare applicazione, infatti i PRNG si trovano spesso ad essere la causa di simulazioni e calcoli stocastici errati [66, 69, 11, 45, 51, 12, 58, 23, 32, 85, 104]. Per quanto riguarda gli scopi crittografici, tutte le principali famiglie di PRNG sono state crittoanalizzato quindi

lontano [45, 89, 72] e l'uso del PRNG laddove dovrebbe essere utilizzato un RNG presenterà quindi un grosso rischio per la sicurezza per il protocollo in questione. Riprenderemo questo punto più in dettaglio nella Sezione 6.

In ogni caso, a causa del rigoroso determinismo degli algoritmi PRNG, nessun PRNG è casuale secondo una ragionevole definizione di casualità. Illustriamolo con un aneddoto fittizio. Alice ha voluto impressionare Bob, con una versione particolare di Mersenne Twister PRNG [57] per la quale ha affermato che produce veri numeri casuali, chiedendogli di testarli. Bob ha accettato ma ha chiesto di inviargli via e-mail un minimo di 1 Giga byte di dati casuali. Alice ha prodotto il file enorme ma il suo programma di posta si è rifiutato di inviare un file così grande. Tagliare un file in piccoli pezzi e inviare più e-mail ecc. era un'opzione, ma una seccatura troppo grande per entrambi. Infine, Bob ha ricevuto da Alice un'e-mail di 1 kilobyte contenente il seguente breve avviso: “Caro Bob, in allegato trovi un programma in C++. Compilalo, usa il seguente seme: 12345678 e fermare il programma dopo aver prodotto 1 Giga byte di dati. Questo è quello che volevo inviarti”. Invece di riprodurre il file ed eseguire sul suo computer test che richiedono molto tempo, Bob ha risposto in breve: "Cara Alice, se pensi che 1 Giga byte di dati veramente casuali può, in qualsiasi circostanza, essere compresso senza perdita a soli 1000 byte, che non ho più niente da dirti!”

I vantaggi dei PRNG sono il loro basso costo, la facilità di implementazione e la facilità d'uso, specialmente in un ambiente disponibile con CPU come un computer PC, ma bisogna essere cauti quando si tratta di usare i numeri PR per simulazioni, crittografia e di fatto qualsiasi uso .

3 veri generatori di numeri casuali

Per il principio di Kerchoff, la definizione di un generatore di numeri casuali adatto alla crittografia deve includere che anche se ogni dettaglio è noto sul generatore (schema, algoritmi ecc.) deve comunque produrre bit totalmente imprevedibili. A differenza dei PRNG, i generatori di numeri casuali fisici (veri, hardware) estraggono la casualità da processi fisici che si comportano in modo fondamentalmente non deterministico, il che li rende candidati migliori per la vera generazione di numeri casuali. Un RNG fisico è un pezzo di hardware separato dal computer, solitamente connesso ad esso tramite USB o bus PCI. L'importazione di numeri casuali in un programma utente è complicata e richiede driver originali. I prezzi vanno da 1k USD a 30k USD per velocità di produzione di bit da 4 a 150 Mega bit al secondo [37, 38, 71].

13, 105, 42, 39, 26]. A differenza dei PRNG, i generatori fisici di numeri casuali soffrono di probabilità dispari di zero e uno, cioè bias (b), definito come

la differenza di probabilità di 1 e 0:

$$b = \frac{P(1) - P(0)}{2} \tag{1}$$

e correlazioni a corto raggio che sono meglio catturate dai coefficienti di autocorrelazione seriale (ak), definiti ad esempio in [45]:

$$un_K = \frac{?_{io=1}^{N-k} (B_{io} - B)(B_{io+k} - B)}{?_{io=1}^{N-k} (B_{io} - B)^2} \tag{2}$$

dove $\{B_1, B_2, \ldots, B_n\}$ in come n bit lunga stringa casuale e K è il ritardo o "ordine" del coefficiente. EntrambiB e un_K sono normalizzati in modo tale che possano assumere valori nell'intervallo [-1, 1] e che un RNG ideale mostri b = 0 e $un_K = 0$. I veri RNG sono generalmente costruiti in modo tale che la correlazione tra i bit sia piccola, ovvero l'idea di casualità. In alcuni casi il sistema fisico misurato viene "ripristinato" a una condizione iniziale dopo la produzione di ciascun bit al fine di ridurre l'autocorrelazione. Pertanto nella maggior parte dei casi sono significativi solo pochi coefficienti di autocorrelazione di ordine più basso, idealmente solo il primo, che prende il nome di autocorrelazione e denotato daun.

Ci sono moltissime costruzioni o veri RNG e la ricerca sta ancora prendendo slancio, ma a nostro avviso si può grosso modo classificare l'arte attuale in quattro famiglie:

- RNG basati sul rumore;
- RNG oscillatori a corsa libera;
- RNG caos;
- RNG quantistici.

L'albero degli RNG è illustrato nella Figura 1. I generatori matematici e pseudocasuali possono anche essere suddivisi in diverse categorie a seconda del tipo di algoritmo utilizzato.

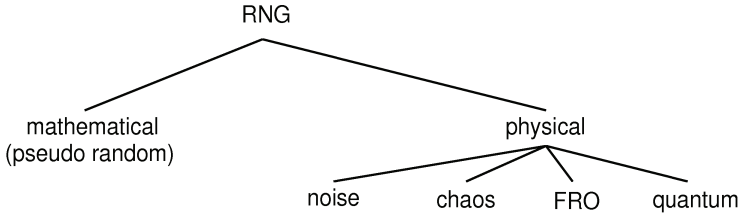


Figura 1: Classificazione dei generatori di numeri casuali.

Si noti che la nostra definizione di vero RNG non deve essere confusa con un generatore di numeri pseudocasuali implementato in logica CMOS o hardware simile; tale generatore è ancora un PRNG, poiché è solo un'implementazione hardware

di un metodo matematico. Successivamente, affronteremo in dettaglio ciascuna delle famiglie di cui sopra.

3.1 RNG basati sul rumore

L'effetto di Johnson [64] crea una tensione casuale sui terminali di qualsiasi materiale resistivo che viene mantenuto a una temperatura superiore allo zero assoluto. Il rumore di Johnson è dovuto al movimento termico casuale della carica elettrica quantizzata (cioè i portatori). Tuttavia, le correlazioni portanti a lungo raggio nei conduttori causano correlazioni nei movimenti di cariche elettriche e quindi la tensione risultante non è completamente casuale [3].

Il rumore Zener (nei diodi Zener a semiconduttore) è causato dal tunneling di portanti attraverso una barriera quantistica di altezza e larghezza idealmente costanti. Se la corrente è sufficientemente bassa, i singoli "salti" delle portanti attraverso la barriera saranno visti come picchi di tensione attraverso il diodo che formano un rumore rosa di perfetta casualità. Una proprietà interessante di questo tipo di rumore è che a una tensione inversa sufficientemente alta il diodo mostra un elevato guadagno a valanga interno. Tale meccanismo di guadagno porta a una grande ampiezza del rumore ed è altamente insensibile alle radiazioni elettromagnetiche dall'ambiente. Tuttavia, l'effetto Zener non si trova mai ben isolato nei dispositivi fisici da altri effetti né la barriera quantistica è costante. La maggior parte dei processi sopra menzionati nei resistori e nei diodi Zener hanno un effetto memoria.

Altre fonti popolari di rumore includono: rottura inversa dell'emettitore di base nei transistor bipolari, rumore di fase laser [33], rumore di caos [50] ecc. Il problema più grande con tutti i tipi di rumore è che la casualità delle sorgenti di rumore non può essere ben caratterizzata, misurata o addirittura controllato durante la fabbricazione del dispositivo. Inoltre, alcuni meccanismi di rumore (in particolare il rumore di Johnson) producono tensioni piuttosto piccole che devono essere fortemente amplificate prima della conversione in forma digitale. La forte amplificazione introduce ulteriori deviazioni dalla casualità dovute alla limitata larghezza di banda dell'amplificatore e alla non linearità del guadagno. Inoltre, la rapida commutazione elettrica della logica binaria utilizzata nei circuiti RNG produce una forte interferenza elettromagnetica in modo che più RNG vicini (specialmente se su chip) tendano a sincronizzarsi reciprocamente causando il drammatico calo dell'entropia complessiva.

L'idea generale del vero RNG basato sul rumore è la seguente. La tensione analogica random viene campionata periodicamente e confrontata ad una certa soglia predefinita: se maggiore di “1” viene generato, altrimenti viene generato “0” (Figura 2). È ovvio che la soglia può essere impostata in modo che le probabilità di 1 e 0 siano più o meno le stesse. Tuttavia, la messa a punto della soglia pone

un problema insormontabile che richiede tempo e non può mai essere fatto correttamente. Ad esempio, se l'ottimizzazione del bias sul valore di 0,1 richiede 10 secondi, l'ottimizzazione su un valore 10 volte inferiore (0,01) richiederebbe 100 volte più tempo (il tempo richiesto viene calcolato come rapporto del quadrato del miglioramento). E poi c'è un problema di stabilità: anche la più piccola deriva del valore medio (per esempio a causa della variazione della temperatura o della tensione di alimentazione) creerà una grande polarizzazione. La verificabilità di qualsiasi RNG di rumore è complicata e alla fine resa impossibile da tre motivi:

1. dimostrabilità della casualità della sorgente di rumore sfruttata;
2. effetto della procedura di campionamento/digitalizzazione;
3. eventuale utilizzo di post-processing deterministici.

Partendo da questo circuito di base, i ricercatori hanno proposto molti circuiti il cui scopo è migliorare la casualità, in particolare il bias.

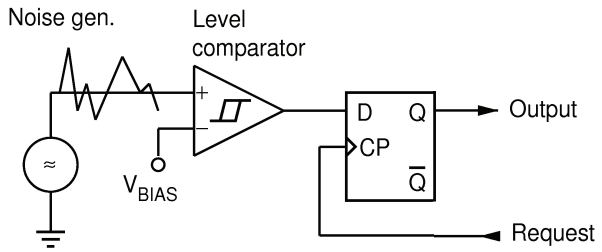


Figura 2: RNG basato sul rumore. Il rumore viene inviato a un comparatore di livello la cui uscita è 0 o 1 a seconda che il suo ingresso di ingresso positivo sia inferiore o superiore al valore di soglia VBIAS. Su richiesta, un nuovo bit casuale verrà posizionato sull'output.

Innanzitutto, il miglioramento più ovvio sarebbe quello di de-polarizzare in qualche modo il flusso di rumore grezzo nell'hardware senza la necessità di alcuna regolazione della tensione di soglia. Una soluzione interessante è stata scoperta da C. H. Vincent nel 1970 [110], generalizzato da Chevalier & Menard nel 1974 [9] e successivamente riscoperto indipendentemente da Bagini e Bucci [1] e Stipcevic [92].

Nel generatore Bagini-Bucci [1] mostrato in Figura 3, la tensione analogica dalla sorgente di rumore in movimento libero viene periodicamente campionata alla frequenza F_{Ck1} e confrontato con un valore di soglia al Comparatore. Ogni volta che il comparatore produce un "1" logico, il flip-flop di tipo T TFF cambia il suo stato. Se il processo campionato è casuale e stazionario, a causa della simmetria temporale di questo processo, l'uscita del TFF trascorre metà del tempo nello stato basso e l'altra metà nello stato alto. Ci sono un paio di problemi con quel design. Innanzitutto, il condensatore di mantenimento funge da memoria che ricorda la tensione analogica precedente. A causa delle impedenze finite nel circuito quando viene caricato con il livello di tensione successivo, la tensione dipenderà in una certa misura dal

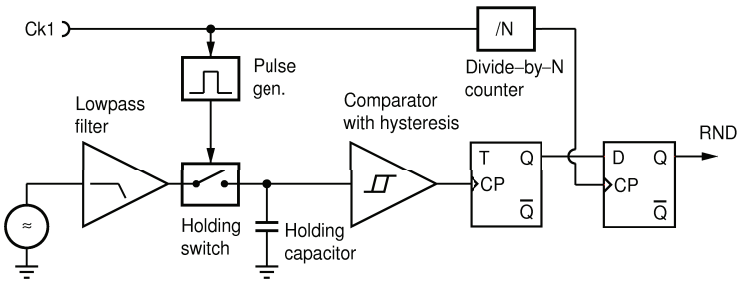


Figura 3: Un RNG basato sul rumore zero di Bagini e Bucci. L'uscita polarizzata prodotta dal principio della soglia imperfetta è divisa per 2 da un flipflop T. L'uscita del flip-flop T trascorre esattamente il 50% del tempo nello stato 1 ed è campionata periodicamente da un generatore di impulsi. L'idea è che quando viene campionato (dal flip-flop D) darà 0 o 1 con probabilità perfettamente uguali. Tuttavia, in pratica si verificherà una deviazione non trascurabile dalla distorsione perfetta e esisteranno correlazioni.

precedente creando così l'autocorrelazione. Il secondo problema è che se il TFF viene interrogato a una velocità troppo elevata, tenderà a dare la stessa risposta più volte nella riga producendo così un output autocorrelato positivamente, anche quando il processo casuale di base è veramente casuale! L'unico modo per aggirare questo problema è usare la frequenza di campionamento dei bitF_{Ck2} molto inferiore alla frequenza di campionamento del rumore F_{Ck1}, Per esempio $F_{Ck2} = F_{Ck1}/N$ arrivando così ad una sequenza asintoticamente casuale di bit nel limite di $n \to$.

Nella variazione di questo principio denominata "sommatoria temporale di un segnale casuale" [92, 91] mostrata ib Figura 4, gli impulsi casuali temporali all'uscita del comparatore COMP sono contati dal contatore modulo 2 (TFF) la cui uscita viene campionata su richiesta inviare l'input della richiesta. I risultati sono simili al circuito Bagini-Bucci eccetto che i bit possono essere generati più velocemente perché non sono necessari né il filtro passa basso né i circuiti di campionamento. Inoltre, è dotato di un anello di polarizzazione zero automatizzato naturalmente incorporato costituito dal comparatore COMP, filtro passa-basso con costante di tempo molto più grande della frequenza di campionamento del bit e amplificatore OPA. Il ciclo imposta la soglia per il comparatore in modo tale che il comparatore passi metà del tempo nello stato "1", che è importante per ridurre al minimo l'autocorrelazione. Il TFF si occupa poi della completa cancellazione del bias.N) va all'infinito. In pratica peròn deve solo essere sufficientemente grande per mantenere le correlazioni al livello desiderato.

Il lato negativo di questo principio di "campionamento" illustrato nelle figure 3 e 4 è che richiedeva n eventi casuali per produrre un bit casuale (bassa efficienza). Il lato positivo è che lasciandon essere sufficientemente grande, si può ottenere qualsiasi livello desiderato di qualità di casualità, almeno in teoria. In pratica, tuttavia, piccole imperfezioni nei circuiti logici alla fine limiteranno il realizzabile

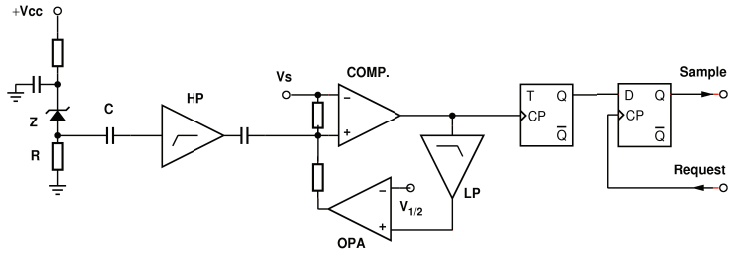


Figura 4: Un RNG basato sul rumore zero di Stipcevic. Gli eventi casuali temporali che compaiono a COMP vengono sommati all'ingresso del flip-flop TFF e quando la somma diventa maggiore del periodo di campionamento del bit dell'intervallo di tempo predefinito T l'uscita casuale è uguale al numero di impulsi casuali in quell'intervallo mod 2. Questo è simile al generatore Bagini-Bucci tranne per il fatto che non necessita di filtro passa basso e circuito di campionamento. Non è necessario che i bit vengano campionati periodicamente. Oltre a ciò c'è un ciclo automatizzato a zero bias.

casualità. Per quanto riguarda la dimostrabilità della casualità di tale principio, imperfezioni tecniche dei singoli componenti, teoria poco chiara di funzionamento della “sorgente di rumore”, nonché complessità complessiva del circuito rendono impossibile arrivare ad una prova credibile di casualità.

Il prossimo esempio di classe di rumore degli RNG è l'Intel RNG [41] implementato in una serie limitata di processori per computer (Figura 5). Utilizza il rumore termico amplificato di un resistore per disturbare un oscillatore controllato in tensione arrivando così a un generatore di impulsi casuali "lento" che viene utilizzato per campionare un oscillatore periodico "ad alta velocità". Questa dicotomia veloce-lento è simile agli RNG di campionamento sopra descritti ed è noto che non genera casualità teoricamente perfetta a meno che il rapporto tra veloce e lento non tenda all'infinito. Una particolarità di questa costruzione è che un oscillatore controllato in tensione (l'oscillatore stazionario ha entropia zero) è disturbato da una tensione rumorosa e quindi molto probabilmente produce un'entropia minore di quella disponibile dalla sorgente di rumore. L'importante proprietà di tale costruzione è che la sua frequenza non può superare certi limiti garantendo così un rapporto sufficientemente alto tra le suddette alte e basse frequenze. È quindi chiaro che i bit generati al flipflop latch (Super Latch) non sono molto casuali e richiedono una post-elaborazione che consiste in un metodo di efficienza von Neumann modificato (e brevettato) 1/4
[100].

C'è ancora un altro Intel RNG apparso nel 2011 dopo "10 anni di ricerca" che è apparentemente estremamente semplice [100] (Figura 6a). L'idea è di ottenere un circuito che non abbia parti analogiche (apparente) e sia quindi compatibile con i chip logici. Il circuito è costituito da due inverter collegati Yin-Yang e due "transistor collegati in modo strano". Gli autori spiegano che questo circuito ha due stati stabili: 0 e 1. Se tutto è perfettamente simmetrico, quando i transistor vengono pilotati in alto, l'uscita finirà in uno stato basso o alto. Gli autori spiegano ulteriormente che anche se idealmente l'output

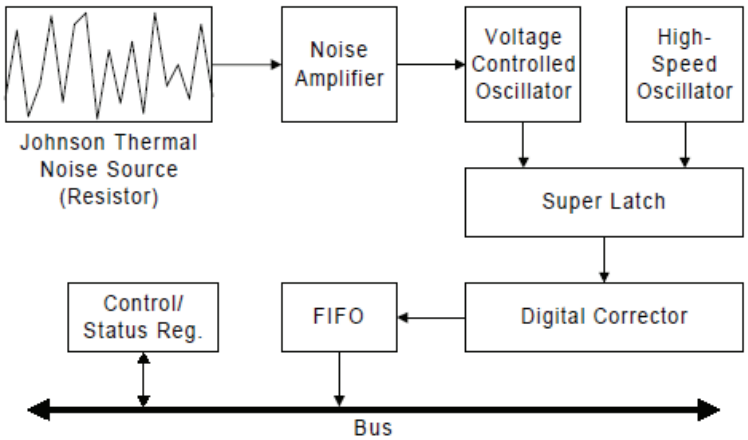


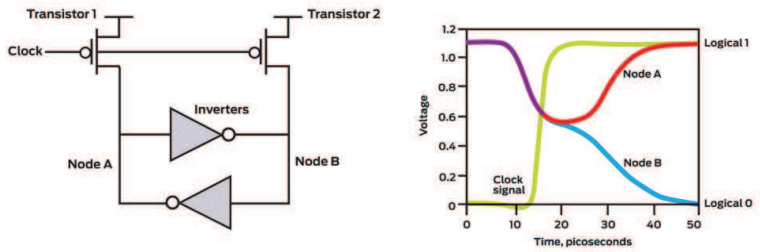
Figura 5: Un RNG Johnson basato sul rumore di Intel. Un oscillatore digitale periodico ad alta velocità viene campionato a tempi approssimativamente casuali definiti da un segnale di rumore Johnson. Gli eventi casuali temporali che compaiono in COMP vengono sommati all'ingresso del flip-flop TFF a levetta.

valore dovrebbe essere casuale, anche la più piccola differenza di velocità o forza degli inverter porterebbe a un elevato squilibrio tra zero e uno (aggiungeremmo: e possibilmente per completare il blocco). Pertanto Intel ha inserito un ulteriore meccanismo di iniezione di corrente che rende gli inverter sufficientemente controllabili da poter essere resi "uguali". La qualità dei numeri casuali deve essere molto bassa, perché Intel utilizza la post-elaborazione in 2 fasi per rimuovere bias e correlazioni (Figura 6b). Il primo stadio è un correttore di casualità non specificato dopo il quale i bit "grezzi" diventano "semi casuali di alta qualità". La seconda fase è un PRNG seminato da questi semi di alta qualità. Non è chiaro perché i veri numeri casuali di alta qualità passino attraverso un PRNG, ma potrebbero esserci solo due ragioni.

Tutti gli esempi precedenti utilizzano il rumore elettronico: una risorsa che sta diventando sempre meno disponibile perché i produttori di componenti e chip elettronici fanno ogni sforzo possibile per ridurla sempre di più. Pertanto le ricerche si sono rivolte a sorgenti in grado di produrre tensioni fluttuanti simili al rumore elettronico ma la cui origine è più fondamentale e quindi meno sensibile ai progressi tecnologici. Ad esempio, i laser possono ottenere un rumore molto veloce. I laser mostrano fluttuazioni molto veloci che possono essere rilevate da PIN veloci o fotodiodi a valanga, producendo così rumore elettrico a banda larga.

Un esempio è il rumore di fase di un singolo laser (Figura 7) inventato dal gruppo CREAM [33].

(un)



(B)

Figura 6: Il generatore di numeri casuali "quantistico" di Intel. (a) Circuito RNG digitale di base. Ad ogni impulso l'uscita si stabilizza nello stato binario casuale. Questo è in effetti ancora un altro generatore di rumore basato su un flip-flop di tipo RS tagliato appositamente preparato i cui ingressi sia set che reset sono legati insieme e guidati allo stesso tempo. La specialità di questo flip-flop è che le sue porte interne possono essere regolate in corrente in modo tale da rendere possibile che l'uscita possa stabilizzarsi in uno stato basso o alto. (Normalmente sarebbe bloccato su uno stato o produrrebbe un bias elevato a causa della più piccola asimmetria delle sue porte interne). (b) Lo schema di post-elaborazione.

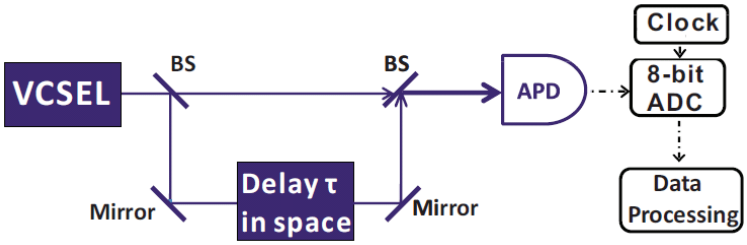


Figura 7: RNG reale basato sul rumore di fase laser. L'intensità del rumore è determinata dall'incertezza fondamentale di fase mentre il suo candore, cioè la distribuzione gaussiana dell'ampiezza istantanea, è dovuto al Teorema del Limite Centrale.

Questo è un esempio di generatore basato sul rumore bianco in cui la distribuzione di forma gaussiana delle ampiezze elettriche analogiche è stata ottenuta con mezzi ottici anziché elettrici (ad esempio come discusso nel generatore Bagini-Bucci [1] e in alcuni altri descritti sopra). La sorgente di rumore, mostrata in Figura 7 è realizzata mediante l'uso di un laser VCSEL single mode dove il segnale

e la sua copia ritardata è stata portata all'interferenza su un rivelatore di APD tramite un interferometro di tipo Michelson-Morley, sistema noto anche come rivelazione "omodina". Il rumore elettrico prodotto da un fotodiodo con larghezza di banda molto elevata è il risultato del jitter di fase del laser. La tensione di rumore prodotta dall'APD viene quindi digitalizzata da un convertitore analogico-digitale (ADC) veloce (40MHz) con risoluzione di 8 bit ei numeri così ottenuti vengono ulteriormente elaborati per ottenere bit casuali a 20 Mbit/sec. Gli autori mostrano che se il ritardo è molto più lungo del tempo di coerenza del laser di 1,6 ns, allora il jitter di fase è dominato da effetti quantistici che sono separati da qualsiasi dettaglio costruttivo e dipendono solo dalle leggi della fisica. In quel regime, l'aggiunta di un jitter sufficiente porta a una distribuzione gaussiana quasi perfetta tramite il teorema del limite centrale, simile al principio utilizzato in [92]. Gli autori misurano ulteriormente la funzione di autocorrelazione del rumore analogico e mostrano che dopo circa 10 ns tutte le correlazioni si estinguono. Per sicurezza, il campionamento del rumore viene effettuato ogni 25 ns e dopo un'ulteriore semplice post-elaborazione si ottengono 20 Mbit/sec di dati casuali che hanno superato tutti i test statistici pertinenti (citati nella sezione E). Un principio di auto-interferenza di fase simile è sfruttato in [70]. Il vantaggio del rumore di fase quantistico rispetto al rumore elettronico è che la sua ampiezza è determinata da leggi fondamentali ed è quindi (nel caso ideale) indipendente dai dettagli tecnologici del laser. Nella nostra scoperta, tuttavia, gli autori qui non hanno preso in considerazione due punti importanti. Primo, il ritardo di tempo introduce un effetto memoria “rolling” che porta necessariamente all'autocorrelazione della tensione di rumore generata dall'APD e quindi i bit da esso ottenuti non sarebbero casuali anche se lo stesso jitter di fase fosse casuale. In secondo luogo, l'algoritmo di generazione dei bit, che include in modo più critico la digitalizzazione di un effetto quantistico casuale analogico, è solo approssimativo e occorre prestare molta attenzione per mantenere la casualità al livello desiderato in ogni momento. Anche così, questo è uno dei rarissimi generatori basati sul rumore che sono caratterizzati da una sequenza pulita di processi fisici e algoritmici in-principio dimostrabili e ben compresi. che include in modo più critico la digitalizzazione di un effetto quantistico casuale analogico, è solo approssimativo e deve essere esercitata una buona cura per mantenere la casualità al livello desiderato in ogni momento. Anche così, questo è uno dei rarissimi generatori basati sul rumore che sono caratterizzati da una sequenza pulita di processi fisici e algoritmici in-principio dimostrabili e ben compresi. che include in modo più critico la digitalizzazione di un effetto quantistico casuale analogico, è solo approssimativo e deve essere esercitata una buona cura per mantenere la casualità al livello desiderato in ogni momento. Anche così, questo è uno dei rarissimi generatori basati sul rumore che sono caratterizzati da una sequenza pulita di processi fisici e algoritmici in-principio dimostrabili e ben compresi.

Ulteriori esempi di veri generatori di numeri casuali basati sul rumore possono essere trovati nella letteratura scientifica e nel database mondiale dei brevetti ESPACENET [22].

Per tutti i generatori di rumore è necessaria una sorta di post-elaborazione. Una semplice post-elaborazione ad hoc come XORing diversi bit successivi, il de-biasing di von Neumann [63] può essere sufficiente. Ma se i bit grezzi mostrano forti correlazioni, le procedure semplici potrebbero non essere sufficienti per eliminare le correlazioni tra i bit che possono anche essere migliorate da semplici procedure di de-polarizzazione o modificate da quelle a corto raggio a quelle a lungo raggio. Un approccio migliore si trova nella post-elaborazione complessa, spesso offline, che tuttavia comporta i propri problemi (vedere la sezione 3.4).

C'è una forte tendenza tra i ricercatori a chiamare gli RNG basati sul rumore "RNG quantistico" perché il rumore è in definitiva causato da piccole particelle governate dalle leggi della meccanica quantistica. Ma il rumore è anche un effetto collettivo, una sommatoria di molti moti individuali e quindi la sua quantita è “sfocata” da un comportamento collettivo che sta a metà tra quanto e

mondi classici. Inoltre, il moto delle particelle che generano rumore (ad esempio gli elettroni in un filo) è solitamente intercorrelato per azione di forze tra di loro in misura tale che il rumore può non essere del tutto casuale [3]. Si noti che un'autocorrelazione dell'ordine di una percentuale può non essere importante quando si considera il movimento degli elettroni, ma se nelle simulazioni numeriche vengono utilizzati numeri casuali generati con l'autocorrelazione seriale dello stesso ordine (0,01), i risultati potrebbero essere completamente errati. Infine il rumore non può essere “riavviato” per interrompere le correlazioni tra successive misurazioni/produzione di bit.

In conclusione, una prova decente della casualità per gli attuali generatori di numeri casuali basati sul rumore sembra impossibile perché i processi fisici sottostanti non sono ben isolati e non si basano su casualità ovvia o scientificamente dimostrabile.

3.2 Caos RNG

Probabilmente il principio più discutibile per la generazione fisica di numeri casuali è ottenerli da misurazioni ripetute di un sistema fisico nel caos. Il problema filosofico qui è che caos significa trovare ordine in ciò che è apparentemente casuale. Allora perché qualcuno dovrebbe usare consapevolmente un sistema non casuale per generare numeri casuali? Non siamo a conoscenza di nessuno finora che abbia chiesto o risposto a questa domanda. A nostro avviso gli autori ricorrono spesso a questo tipo di generatori per tre ragioni:

1. mescolanza concettuale di caos e casualità;
2. (erronea) convinzione che i sistemi difficili da descrivere si comportino necessariamente in modo casuale;
3. robustezza di alcuni sistemi caotici a produrre livelli macroscopici di “rumore” facilmente utilizzabili per generare numeri casuali essenzialmente tramite metodi RNG di rumore (come descritto nella Sezione A).

Allo stato attuale dell'arte, i sistemi caotici più convenienti per la generazione rapida di numeri casuali sono ottici, elettrici o opt-elettrici, sebbene siano state dimostrate anche costruzioni meccaniche, ad esempio in [61]. In questa sezione presentiamo diversi modelli tipici.

I laser possono essere portati a fluttuazioni caotiche di potenza da molti meccanismi diversi. Sono ben note le costruzioni caotiche che coinvolgono laser a feedback distribuito [50]. Un sistema laser caotico autofeedback molto semplice ma estremamente veloce [42] è mostrato nella Figura 8. Anche in questo caso, la luce di ampiezza caoticamente fluttuante viene rilevata da un fotodiodo veloce (PD) la cui ampiezza viene campionata da un ADC veloce a 8 bit e ulteriormente elaborato eseguendo un'elevata differenziazione degli ordini, per ottenere una velocità di produzione bit record mondiale di 300 GigaBit/sec.

I laser offrono mezzi per la realizzazione di sistemi caotici molto veloci e sono frequentemente utilizzati per la generazione di numeri casuali. A causa della possibilità di costruire

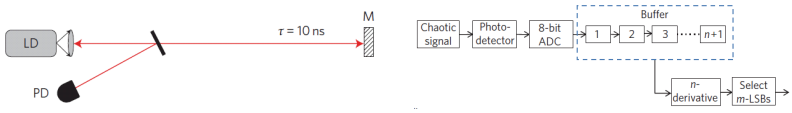


Figura 8: Un'intensità caoticamente comportata da un laser di auto-feedback viene letta da un fotodiodo (PD) la cui ampiezza viene campionata da un ADC veloce e ulteriormente elaborata eseguendo un'elevata differenziazione di ordine, per produrre una velocità di produzione di bit record mondiale di 300 GigaBit/sec .

minuscoli laser, risonatori e vari elementi ottici passivi e attivi su un chip, tali generatori potrebbero essere completamente integrati e potrebbero presentare un basso consumo energetico.

Un generatore di numeri casuali mostrato in Figura 9 [50] è costituito da un laser caotico a banda ultralarga (UWB) (a), un campionatore di ampiezza (b) e un comparatore (C). Il suo principio di funzionamento è un copia-incolla del generatore di rumore Bagini-Bucci descritto in precedenza (Figura 3) con la differenza che qui invece del rumore elettrico viene utilizzata un'intensità luminosa di un laser caotico come fonte di casualità. L'interessante caratteristica distintiva di questo RNG è che è "tutto ottico", nel senso che tutti i segnali e l'elaborazione dei segnali vengono eseguiti a livello ottico, anche i numeri di uscita sono infatti livelli digitali di intensità luminosa: bassa intensità luminosa significa "0" mentre alta intensità significa "1". Questo è interessante per l'uso in chip di elaborazione completamente ottici e inoltre, se necessario, l'uscita può essere facilmente convertita in segnale elettrico mediante l'uso di un fotodiodo veloce e di un amplificatore adatto.

Il laser caotico UWB è costituito da due laser di feedback distribuiti, "master" e "slave" (Figura 9a) con il master che disturba il circuito di feedback dello slave in modo tale da aumentarne la larghezza di banda in regime caotico [114]. L'intensità di uscita viene estratta dal circuito di retroazione mediante un divisore di fascio e campionata da un campionatore ottico a una frequenza di campionamento costante determinata dal laser mode-locked (Figura 9b). Ciascun valore campionato dell'intensità della luce viene quindi confrontato con un valore di soglia mediante un comparatore completamente ottico (Figura 9c) risultante in un'elevata intensità di uscita ("1") o in una bassa intensità ("0"). I bit casuali vengono prodotti al ritmo del laser a modalità bloccata.

I bit così ottenuti sono distorti e in qualche modo autocorrelati. Poiché vengono prodotti in tempi periodici, gli autori ricorrono a una conveniente procedura di riduzione della distorsione e delle correlazioni eseguendo l'XOR sui bit di output simultanei di due RNG identici e indipendenti, come mostrato nella Figura 10. I bit casuali risultanti superano test statistici rilevanti [50]. Il comportamento caotico del laser UWB master-slave è stato teoricamente modellato e la larghezza di banda del modello è risultata in accordo con i dati sperimentali [115], nel tentativo di supportare l'affermazione di casualità del suddetto RNG. Tuttavia, la modellazione o la dimostrazione della forma e dell'ampiezza dello spettro del rumore di una sorgente non dimostra nulla della sua casualità.

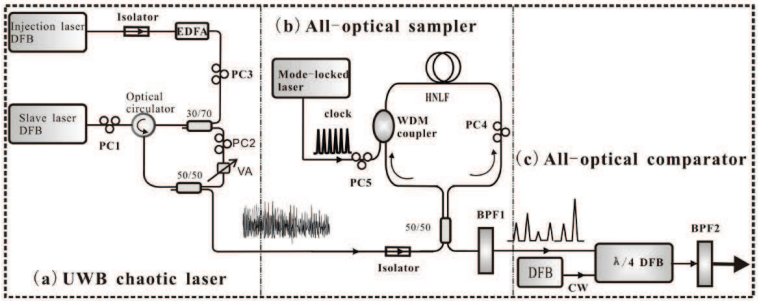


Figura 9: Laser tutto-ottico costituito da: a) laser caotico a banda ultra larga (UWB); b) campionatore completamente ottico e c) comparatore completamente ottico.

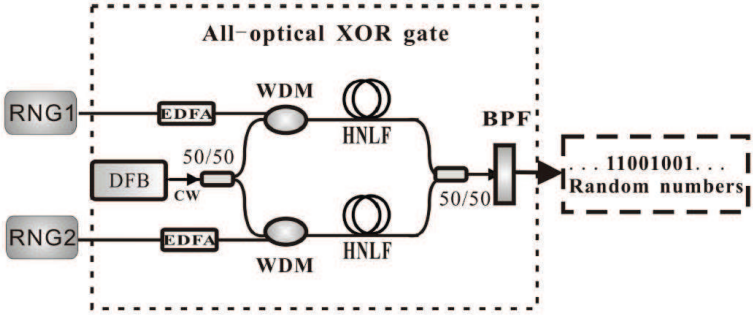


Figura 10: L'XORing completamente ottico di due RNG indipendenti riduce la distorsione e le correlazioni tra i bit.

Nel corpo di ricerche relative agli RNG caotici, alcuni autori affermano di utilizzare sistemi nel caos senza effettivamente fornire alcuna prova diretta che il sistema in uso per la generazione di numeri casuali sia effettivamente nel caos [102], alcuni sono in grado di dimostrare un comportamento caotico ad esempio studiando mappe balistiche o esponente di Lyapunov [73] e alcuni arrivano addirittura a modellare il comportamento caotico del sistema e lo confermano sperimentalmente [50, 114, 115]. Ma comunque sia, gli RNG caotici hanno una base teorica comune a quei PRNG che operano simulando un sistema caotico deterministico (ad esempio) e quindi alla lunga hanno avuto il fiato corto nel produrre nuova entropia, finendo inevitabilmente nel produrre non più di un piccola frazione di 1 bit di entropia per ogni nuovo bit casuale generato.

L'obiezione generale all'idea stessa di generazione di numeri casuali da parte del caos è che il comportamento caotico è definito come un tipo specifico di soluzione dell'equazione differenziale che, integrata da condizioni iniziali, descrive il sistema. Poiché tali equazioni e dati contengono solo una quantità limitata (piccola) di informazioni, una volta che tale quantità di informazioni viene estratta dal sistema mediante misurazioni non vi sono nuove informazioni che possono essere estratte.

tratto da esso e di conseguenza tutte le ulteriori misurazioni contengono (asintoticamente) zero nuove informazioni. In particolare significa che un sistema caotico, in teoria, può produrre solo un insieme limitato di bit casuali e che tutto il resto deve essere perfettamente o quasi perfettamente correlato a quell'insieme. Detto questo, comprendiamo che un sistema caotico realistico non si comporta mai esattamente come farebbe obbedendo all'"equazione del moto" che lo modella a causa di effetti quantistici o statistici casuali che randomizzano continuamente la traiettoria nello spazio delle fasi del sistema. Tuttavia, questi effetti aggiuntivi non sono la base per un RNG caotico (e quindi non sono considerati nella sua definizione) e di solito sono anche troppo piccoli o inefficaci per fare una differenza significativa in un sistema il cui comportamento è per lo più determinato da un caos osservabile macroscopicamente.

3.3 RNG dell'oscillatore a corsa libera

Quando l'uscita di un circuito inverter logico viene alimentata al suo ingresso, il circuito si trasforma in un oscillatore, il cosiddetto oscillatore a corsa libera (FRO), Figura 11.

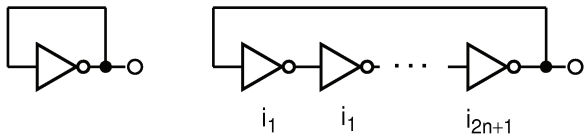


Figura 11: Diagramma schematico di FRO veloci (sinistra) e lenti (destra). La frequenza di oscillazione è determinata da ritardi interni e capacità parassite.

Un gate invertente è in pratica un amplificatore invertente ad altissimo guadagno. Collegando il suo output all'input si crea il paradosso di Zeno: se l'output è nello stato logico ALTO, anche l'input lo sarà e l'azione NOT porterà l'output a diventare LOW. Una volta che l'output diventa LOW, l'azione NOT lo porterà a HIGH e così via. L'analisi della logica booleana teorica darà che l'uscita è indeterminata ma in pratica a causa del ritardo di propagazione finito dell'elemento NOT il circuito oscillerà. La particolarità di questa oscillazione è che appare in un circuito con feedback negativo (sfasamento di 180 gradi) mentre nella teoria dell'elettronica il feedback negativo porta alla "stabilizzazione" piuttosto che all'oscillazione. La ragione di ciò è che analizzando gli stati logici abbiamo ipotizzato un guadagno infinito. Tuttavia, poiché in pratica il guadagno non è mai infinito, può accadere che il circuito si blocchi (si stabilizzi) in uno stato di tensione compreso tra zero e uno senza o con oscillazioni di ampiezza molto piccole che non sono in grado di pilotare ulteriori circuiti logici. Per aiutare le oscillazioni

si può aggiungere intenzionalmente qualche reattanza nel circuito di retroazione in modo da produrre uno sfasamento diverso da ±180 gradi. La stessa funzione può essere fornita con reattanze parassite. In tal caso il criterio di Barkhausen può essere soddisfatto per alcuni poli ad alta frequenza e appariranno delle oscillazioni. A causa del complesso meccanismo delle oscillazioni libere, la loro frequenza è tipicamente abbastanza sensibile alla variazione della tensione di alimentazione e della temperatura ma queste variazioni sono lente rispetto alla frequenza di oscillazione. D'altra parte il rumore elettronico presente in ingresso si addice al segnale retroazionato dall'uscita e dopo essere stato fortemente amplificato provoca jitter molto veloci e casuali di frequenza e fase delle oscillazioni. In tal senso, FRO RNG può essere considerato un caso speciale di generatore di rumore. Poiché il rumore di ciascuno di questi circuiti è individuale, è ragionevole presumere che i molteplici oscillatori anche quando si trovano sullo stesso chip abbiano frequenze diverse e che le loro fasi reciproche si allontanino casualmente nel tempo. Ma quando più oscillatori di questo tipo sono vicini l'uno all'altro (ad esempio su un singolo chip) tendono a sincronizzarsi attraverso l'interazione elettromagnetica facilitata dall'alto guadagno degli amplificatori FRO. In effetti, l'immenso guadagno delle porte NOT necessarie per amplificare il minuscolo rumore elettronico a un livello notevole aiuta anche a rilevare qualsiasi altra interferenza vicina. Questo effetto noto come "interblocco a fasi" [64] può influire negativamente sulle prestazioni del progetto ed è un grave problema inerente agli FRO. Gli anelli interbloccati hanno forme d'onda che condividono (quasi) la stessa fase e questo porterà a un funzionamento (quasi) pseudo-casuale.

Il principio di base della generazione di numeri casuali con FRO è che l'output di un FRO veloce (che può essere 0 logico o 1) viene campionato da un FRO lento. Questo è l'equivalente dell'arresto improvviso di una ruota della fortuna che gira rapidamente. Poiché la ruota gira così "velocemente" sembra ferma in una posizione "casuale". Nel caso di due FRO è importante che il relativo jitter di fase tra FRO veloce e lento sia casuale e sufficientemente grande. Chiaramente, se non c'è jitter di fase relativo, l'uscita fornirà uno schema binario ripetitivo. Se il jitter è casuale ma piccolo, anche la deviazione dal modello ripetitivo sarà piccola portando a un comportamento quasi pseudo-casuale. Se i FRO si sincronizzano o almeno si sincronizzano parzialmente, apparirà un pattern con escursione stocastica (rumore). A parte quello, un altro problema molto importante con gli RNG FRO è che l'ampiezza di uscita di un FRO dipende dai dettagli delle reattanze parassite e dai ritardi nel circuito. Come spiegato sopra, per un particolare circuito può succedere che l'ampiezza di uscita di un FRO sia troppo piccola per pilotare il circuito logico o che FRO si blocchi in qualche stato e smetta di oscillare. L'azione di Schmidt all'ingresso può aiutare a ridurre al minimo questo problema, ma a scapito della riduzione della frequenza di oscillazione e della complicazione della fabbricazione.

Nonostante tutti questi problemi, gli attuali standard di sicurezza [76] impongono praticamente l'uso di RNG basati su FRO gratuiti. Lo standard NIST FIPS140-2 [21] dice: “Non ci sono generatori di numeri casuali non deterministici approvati FIPS”. Di conseguenza, l'approccio FRO, attualmente è utilizzato in 3a e

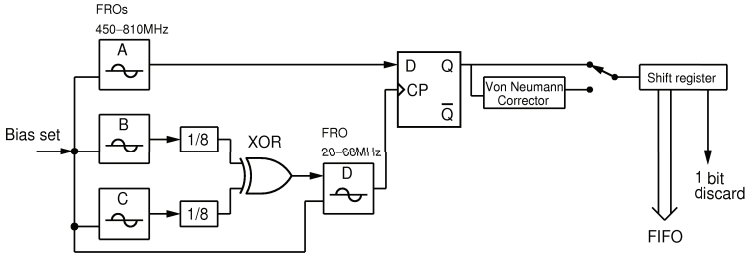


Figura 12: VIA C3 PadLock generatore di numeri casuali campiona velocemente FRO (A) da lento FRO (D).

Hardware FPGA, CPLD e ASIC di quarta generazione per vari scopi crittografici. Un esempio di vita reale che illustra bene la cucina combinatoria tipicamente necessaria per ottenere un RNG decente è la fonte di entropia per l'RNG "quantistico" PadLock implementato nei processori VIA C3 [105, 106, 107, 108]. Consiste di quattro FRO, 3 veloci (450-810MHz) e 1 lento (20-68MHz). L'ampia tolleranza sulle frequenze mostra già i problemi che abbiamo menzionato prima: è molto difficile controllare i parametri degli FRO durante la fabbricazione. In questa topologia un FRO veloce (A) è campionato da un FRO lento (D) come scoperto nella domanda di brevetto [93]. Almeno uno dei due FRO deve essere di buona casualità e poiché è più facile da ottenere con uno più lento VIA ha optato per questa opzione. Il generatore lento è composto da FRO B, C e D. Primo, B e C sono rallentati di 1/8 divisori e le loro uscite XORed sono utilizzate per disturbare FRO D lento (che è l'unico dotato di ingresso digitale). I bit risultanti appaiono all'uscita Q del flip-flop di tipo D in sincronizzazione con gli impulsi del FRO D. Facoltativamente, l'uscita viene filtrata attraverso il correttore di von Neumann [63] che riduce la velocità di produzione dei bit all'incirca di un fattore 4 (vedi descrizione nella sezione E). Guardando questo schema è chiaro che è impossibile arrivare a una prova della sua casualità. Secondo VIA [105], la tensione di polarizzazione analogica iniettata a questo circuito altrimenti digitale "può (o non potrebbe!) migliorare le caratteristiche statistiche dei bit casuali".

Poiché l'infrastruttura del chip logico digitale non è adatta alla realizzazione di un RNG quantistico (sottosezione D), un approccio FRO sembra essere un'alternativa ragionevole. Tuttavia, l'avvertenza con gli FRO è che l'industria dei semiconduttori sta facendo uno sforzo enorme per ridurre il più possibile il rumore dell'elettronica e generalmente si riduce con le versioni più recenti di un chip. Di conseguenza, l'effetto del jitter può essere molto piccolo e far sì che l'RNG basato su FRO operi in regime quasi PRNG. Pertanto, i dettagli di implementazione di un RNG basato su FRO molto spesso devono essere personalizzati per ogni tipo specifico o

la generazione e la tecnologia di un chip programmabile/riconfigurabile o ASIC e l'uniformità di funzionamento non possono essere garantite da lotto a lotto.

Una soluzione parziale ai problemi sopra menzionati è stata recentemente trovata in una nuova combinazione sinergica di un registro a scorrimento a retroazione lineare (LSFR) [30] e FRO, denominati Fibonacci Ring Oscillator (FIRO) e Galois Ring Oscillator (GARO) [18]. L'idea è di avere un PRNG simile a LSFR con seeding che è realizzato come un FRO con clock. Tali veri generatori di numeri casuali ricevono uno stato iniziale (seme) ma sebbene il seme imposti lo stato iniziale, due generatori identici con seme identico divergerebbero nel tempo poiché sono sotto l'influenza di (almeno parzialmente) rumori individuali. La Figura 13 mostra lo schema di GARO e FIRO.

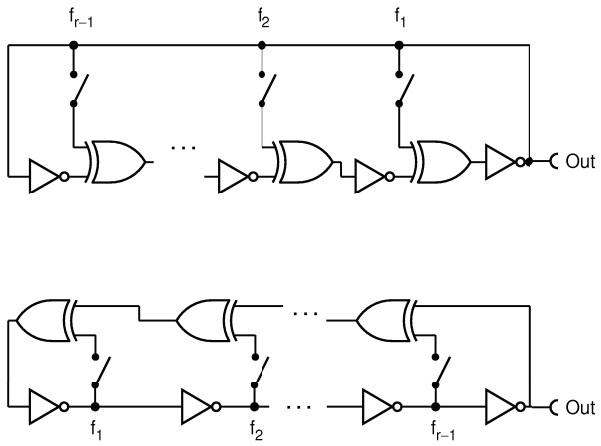


Figura 13: Oscillatore ad anello di Galois (su) e oscillatore ad anello di Fibonacci (in basso). Il numero di stadi definisce l'ordine (r) mentre commuta F_{io} definire i coefficienti del polinomio di retroazione.

Tuttavia, anche con questo principio interessante e innovativo, il problema è la non portabilità multipiattaforma del design e il requisito di un rumore sufficientemente grande affinché lo schema funzioni in un regime ragionevolmente casuale (lontano dallo pseudocasuale). Inoltre, gli autori avvertono che la progettazione deve essere eseguita con la massima attenzione al fine di ridurre al minimo l'interblocco con l'orologio di sistema e altri circuiti logici nel chip, compresi gli FRO vicini. Pertanto hanno sperimentato il posizionamento spaziale degli FRO nel chip. Concludono anche che la casualità di nessuna delle due famiglie di generatori di per sé non è perfetta e potrebbe essere "migliorata" dall'XORing di due generatori indipendenti, più favorevolmente uno GARO e uno FIRO.

Ulteriori esempi sulla ginnastica pre e post elaborazione FRO, inclusi XORing multipli generatori, combinazioni con LSFR ecc. possono essere trovati in [96]. La complessità delle procedure di post-elaborazione necessarie per superare il

i test statistici con RNG basati su FRO sono spesso tali da rendere impossibile qualsiasi prova di casualità, ma anche gli autori più interessanti non sembrano quasi mai essere consapevoli della necessità di una prova. Una rara esenzione in questo senso è il lavoro di Sunar et. al. [98] dove un modello teorico di un RNG basato su FRO è stato presentato, analizzato e dimostrato ma successivamente criticato in [118] come non realistico. Tuttavia riteniamo insoddisfacente l'intera dimostrazione perché si basa sul modello di FRO di McNeill che postula semplicemente che le oscillazioni libere si verificano come un processo casuale non stazionario senza effettivamente collegare il postulato alla realtà, ad esempio mediante leggi della fisica. Un'ottima lettura e sintesi dei problemi e della cucina utilizzata per minimizzarli si trova in [118]. Ulteriori letture sugli RNG basati su FRO sono fornite in [96].

In conclusione, gli RNG basati su FRO sono soluzioni a basso costo e a bassa entropia il cui unico lato positivo è il fatto che possono essere facilmente implementati in chip logici convenzionali programmabili o riconfigurabili che vengono utilizzati in varie soluzioni di sicurezza informatica, ma non offrono casualità molto buona o dimostrabile.

3.4 RNG quantistici

Che cos'è un generatore di numeri casuali quantistici? Poiché viviamo nel mondo governato dalle leggi della fisica quantistica, qualsiasi vero generatore di numeri casuali (ad esempio un dado che lancia o una moneta che lancia) può essere chiamato "quantistico". Tuttavia, vogliamo riservare questo nome solo a quei generatori che utilizzano un singolo effetto quantistico intrinsecamente casuale (realizzato il più vicino possibile alla sua idealizzazione teorica) misurato più e più volte al fine di produrre bit casuali in modo tale che tra due qualsiasi set di misurazioni utilizzate per dedurre bit casuali, il sistema viene ripristinato alle stesse condizioni iniziali. Può sembrare strano che un simile assetto fisico (generatore) sia addirittura possibile, cioè che partendo esattamente dalle stesse condizioni iniziali e misurato esattamente nello stesso modo dia risultati diversi, ma la fisica quantistica lo consente.

Si scopre che alcune cose in Natura arrivano nelle quantità più piccole conosciute come quanti. Ad esempio l'elettrone porta la più piccola quantità di carica, e. Allo stesso modo, c'è la più piccola quantità di informazioni, chiamata qubit. Un singolo quanto di luce (fotone) può essere usato come portatore di un qubit, ma ci sono molti altri esempi e non sono limitati solo alle particelle elementari. Qubit può essere pensato come una combinazione lineare di due valori di bit: 0 e 1. Quando un certo tipo di misurazione viene eseguito su un qubit, "proietterà" allo stato 0 puro o 1 puro nella base in cui la misurazione è stata eseguito. Molto spesso i fotoni vengono utilizzati nei QRNG perché sono facili da creare, manipolare e rilevare. Per illustrare questo, consideriamo un fotone polarizzato circolarmente che entra in un divisore di fascio polarizzante

(PBS), Figura 14. Il PBS scompone la polarizzazione della luce incidente e invia la componente orizzontale lineare su una porta di uscita e la componente verticale lineare sull'altra porta.

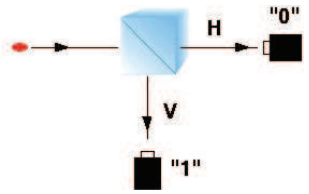


Figura 14: Principio spaziale QRBG. Il fotone a polarizzazione circolare si divide su un analizzatore lineare orizzontale/verticale con il 50% di possibilità di finire in una delle due porte di uscita.

Pertanto, il fotone polarizzato circolarmente ha lo stesso contenuto di entrambe le polarizzazioni lineari, ma poiché non può essere diviso a metà, ha esattamente il 50% di possibilità di uscire da entrambe le porte. Se ora etichettiamo una delle porte come “0” e l'altra come “1” otteniamo subito un RNG teoricamente perfetto la cui casualità è garantita dalle leggi della fisica quantistica. Nota che il sistema che viene "misurato" è sempre lo stesso, ma dà sempre un nuovo risultato casuale. Questo è completamente diverso dai generatori caotici e rumorosi dove per ottenere un risultato diverso devono cambiare i sistemi.

Gli RNG quantistici basati su questo (o altri principi) possono essere realizzati abbastanza bene e le imperfezioni di qualsiasi tipo (emissione multi-fotone, polarizzazione circolare non perfetta, disallineamento dell'asse della porta del divisore del fascio, tempo morto del rivelatore, effetti di post-pulsazione e memoria, ecc.) possono essere misurati indipendentemente dal processo di generazione dei bit in modo che il loro effetto sui numeri casuali possa essere stimato con precisione e trattato con la post-elaborazione (vedere la Sezione 3.5). Questo metodo è una base per un generatore commerciale [37].

Il problema principale nella realizzazione pratica dell'RNG a divisione di fascio è che richiede due rivelatori. Le loro differenze iniziali e il successivo allontanamento nel tempo a causa dell'invecchiamento e/o degli effetti della temperatura avranno un impatto immediato sulla qualità dei numeri casuali. Ad esempio, se le efficienze di rilevamento dei fotoni dei rivelatori non sono perfettamente uguali, o se il divisore di fascio non è perfettamente del 50/50 percento, allora la probabilità di uno non sarà uguale alla probabilità di zero. Questo problema può essere minimizzato utilizzando uno schema di suddivisione del fascio che utilizza un solo rivelatore di fotoni [90] mostrato in Figura 15, ma il rapporto di divisione del raggio deve essere regolato meccanicamente con precisione. I problemi residui derivano dal tempo morto del rivelatore e dalla post-pulsazione che portano a correlazioni impossibili da eliminare completamente, ma che possono essere ridotte al di sotto di qualsiasi livello desiderato mediante post-elaborazione mirata.

Il divisore di fascio RNG è un esempio di "principio spaziale" in cui il valore del bit casuale, 0 o 1, è determinato dal punto in cui finisce il fotone

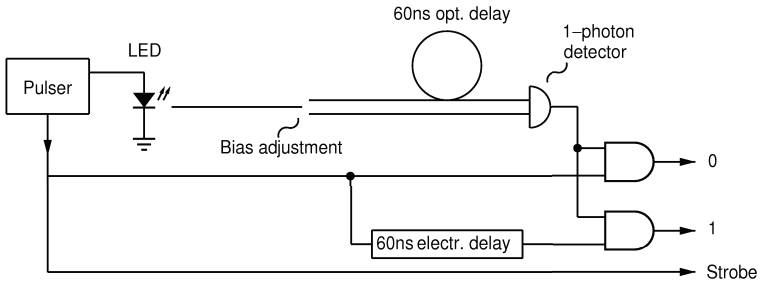


Figura 15: Generatore ottico quantistico di numeri casuali basato sulla suddivisione del fascio che utilizza un solo rivelatore di fotoni per evitare fluttuazioni di polarizzazione con invecchiamento e tolleranze iniziali.

su. Un "principio temporale" complementare utilizza le informazioni temporali dell'emissione di fotoni casuali, ad esempio nel rilassamento atomico diretto (o punto quantico), da laser ben saturati, ecc.

Un semplice metodo di intervallo di tempo mostrato in Figura 16, che è particolarmente immune alle imperfezioni hardware, è stato proposto in [95]. Utilizza le informazioni temporali piuttosto che spaziali contenute nel generatore di eventi casuali (REG). In [95] vengono utilizzati per la prima volta processi di emissione e rivelazione di fotoni invece di un decadimento radioattivo molto più lento (e più pericoloso!) [24, 29]. Il principio di produzione di bit è il seguente. Intervalli di tempoT_1 e T_2 attraversato da tre rilevamenti di fotoni successivi vengono confrontati: if $T_1 > T_2$ quindi produrre "0", se $T_2 > T_1$ quindi produrre "1", se $T_1 = T_2$ quindi non produrre nulla (salta).

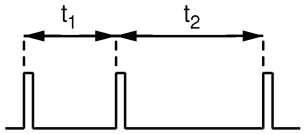


Figura 16: Principio di temporizzazione QRBG. I fotoni di una sorgente Poissoniana a singolo fotone cadono su un singolo rivelatore di fotoni. Intervalli di tempoT_1 e T_2 attraversato da tre rilevamenti di fotoni successivi vengono confrontati: if $T_1 > T_2$ quindi produrre "0", se $T_2 > T_1$ quindi produrre "1", se $T_1 = T_2$ quindi non produrre nulla (salta).

Lo schema della configurazione fisica è mostrato nella Figura 17. Poiché viene utilizzato un solo rivelatore di fotoni, sia la polarizzazione che le correlazioni vengono soppresse a livelli quasi non rilevabili, ma non c'è nulla da regolare (a differenza del principio di divisione del fascio).

Il problema con questo metodo è come vengono misurati gli intervalli di tempo. Il miglioramento cruciale apportato in [95] è la nozione che gli intervalli di tempo di misurazione dell'orologio (ti) devono essere avviati in sincronia con l'inizio di ogni in-

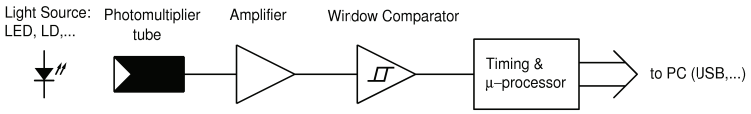


Figura 17: Uno schema generale di elaborazione del principio temporale QRBG. I fotoni casuali cadono sul singolo rivelatore di fotoni costituito da un fotomoltiplicatore, un amplificatore e un comparatore, in modo tale che ciascun fotone rilevato generi un impulso logico. Gli impulsi vengono quindi elaborati secondo il principio di estrazione dei bit desiderato e trasmessi a un computer.

terval, altrimenti il metodo produrrebbe bit correlati anche se alimentato da eventi perfettamente casuali. Questo non era compreso in lavori e brevetti precedenti [24] che di conseguenza dovevano aver prodotto un output correlato ma questo non era stato rilevato al momento perché la frequenza di clock (? 10 MHz) era molto più alta della frequenza media della sorgente (? 10kHz) nel qual caso le correlazioni sono piccole. Si può dimostrare che questo metodo non solo funziona bene a basso rapporto tra frequenze di clock e RPG, ma cancella anche quasi tutte le imperfezioni: cambio di intensità della sorgente, cambio di efficienza, tempo morto e postpulsazione del rivelatore di fotoni. è anche altamente immune alla distribuzione effettiva dei tempi di intervallo casuali, purché gli eventi siano indipendenti l'uno dall'altro. Inoltre, la produzione di bit casuali è autosincronizzata, quindi se la sorgente o il rilevatore muoiono non ci saranno bit in uscita. Questo generatore è stato il primo a superare tutti i test noti inclusi i "soliti" test di statistica t [54, 112, 78, 79] e alcuni test di casualità algoritmica non divulgati [35].

Una combinazione di divisore di fascio e principio temporale è descritta in [39]. Il flusso di fotoni non polarizzati dalla sorgente luminosa (diodo LED) viene fatto passare attraverso un polarizzatore raggiungendo un divisore di fascio polarizzante (NPBS), molto simile al suddetto divisore di fascio RNG (Figura 14). Con un'attenta regolazione dell'angolo relativo tra il polarizzatore e l'asse NPBS (idealmente 45o) i rivelatori D1 e D2 dovrebbero produrre impulsi casuali, reciprocamente non correlati di uguale frequenza (tuttavia la regolazione dell'angolo del polarizzatore è un compito insormontabile, come spiegato per RNG nella Figura 3 Mentre gli impulsi da D1 si azzerano (inout R) il flip-flop di tipo RS impostando l'uscita allo stato LOW, il set D2 (ingresso S) il flip-flop allo stato HIGH. L'uscita di detto flip-flop viene campionata a tempi periodici in modo da generare un bit casuale.

Essendo una combinazione di suddivisione del fascio e principi di campionamento, questa costruzione eredita il peggio di entrambi:

1. il bias è instabile (sensibile alle variazioni di temperatura) e regolabile solo meccanicamente;
2. correlazioni dovute al periodo di campionamento finito come discusso nei generatori di rumore; e tutto questo anche supponendo una sorgente di fotoni perfettamente casuale.

Un QRNG commerciale di Weinfurter et al. [26, 71] che utilizza solo il principio temporale è mostrato nella Figura 19. Lo schema di acquisizione dei dati è equivalente a

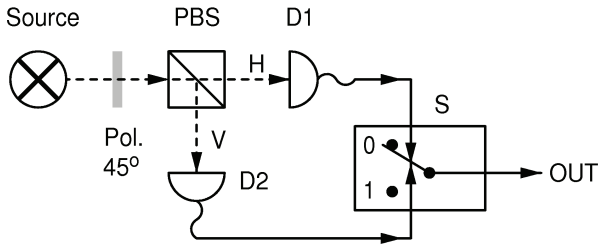


Figura 18: Generatore ottico quantistico di numeri casuali basato sulla suddivisione del fascio e sui principi del campionamento periodico.

lo schema generale riportato in Figura 17 con sorgente luminosa a LED a bassa intensità debolmente accoppiata ad un tubo fotomoltiplicatore. Il basso accoppiamento assicurava una bassa velocità di campionamento dei fotoni dell'ordine di 10^{-8} che sopprime ogni eventuale correlazione di fotoni ben oltre il livello rilevabile. Il metodo di estrazione dei bit è implementato nel chip riconfigurabile FPGA ed è il seguente. Il numero di fotoni rilevati viene contato in intervalli di tempo costante producendo una statistica Poissoniana. Il numero pari di eventi all'interno di un intervallo viene interpretato come "1" e dispari come "0". Gli autori notano che a causa della forma non simmetrica della distribuzione Poissoniana, la probabilità di uno non è uguale alla probabilità di zero. Tuttavia, a causa di due imperfezioni nel rivelatore di fotoni (tempo morto diverso da zero e dipendenza del tempo morto con la frequenza di rilevamento) la distribuzione risultante non è Poissoniana ma più a campana, portando così favorevolmente a un bias che tende rapidamente a zero quando il conteggio la lunghezza dell'intervallo aumenta. Gli autori mostrano e confrontano i risultati sperimentali e teorici per il bias modellato, tuttavia non modellano o provano nulla sulle correlazioni. Invece, le correlazioni vengono semplicemente valutate dai bit generati utilizzando il coefficiente di autocorrelazione lineare. In teoria, il bias tende a zero quando la frequenza di rilevamento va all'infinito. Empiricamente, la condizione operativa preferita è vicina alla frequenza di rilevamento più alta possibile ma un po' più piccola a causa dei problemi crescenti nel rivelatore di fotoni. Ma allo stesso limite, c'è da aspettarsi che il bias fluttuante produca un livello crescente di complesse correlazioni a corto raggio tra i bit, che tuttavia non è stato modellato matematicamente e/o messo in relazione con le imperfezioni del setup. Il problema con questo approccio è che non riesce a descrivere un modello teorico di un RNG che fornisce numeri casuali perfetti basati su un effetto quantistico (quasi) idealmente casuale (ad esempio emissione a bassa intensità da LED) e assumendo un apparato ideale. Di conseguenza, se non riesce a dimostrare chiaramente la casualità e a modellare la deviazione dalla perfetta casualità introdotta da imperfezioni legate all'implementazione. Tuttavia, questo generatore ha un valore pratico perché apparentemente supera tutti i test statistici rilevanti. È comunque da capire questo generatore ha un valore pratico perché apparentemente supera tutti i test statistici rilevanti. È comunque da capire questo generatore ha un valore pratico perché apparentemente supera tutti i test statistici rilevanti. È comunque da capire

che una prova di casualità accettabile non può essere ottenuta superando un numero qualsiasi di test di casualità (come verrà discusso nella Sezione 3.5)

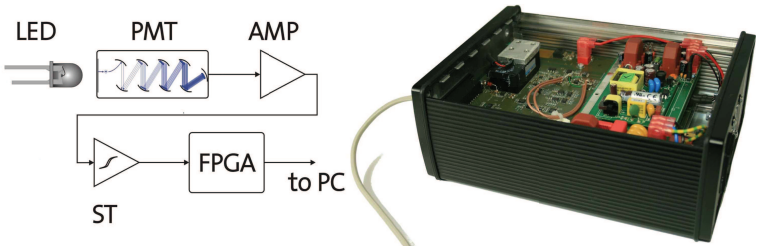


Figura 19: Generatore ottico di numeri casuali quantistici basato su statistiche quasi esponenziali del rilevamento del tempo dei fotoni. È stato scoperto che le principali imperfezioni del rivelatore, il tempo morto e il pile-up, funzionano a favore di bias più piccoli e autocorrelazione seriale che sono risultati essere piccoli come $2 \cdot 10^{-5}$ senza post-elaborazione.

Un altro RNG quantistico commerciale che utilizza le informazioni sul tempo di arrivo dei fotoni è stato presentato da Picoquant [111, 68]. Qui l'intera catena di ragionamento richiesta per una convincente dimostrazione di casualità è stata almeno tentata e, secondo gli autori, stabilita con successo. Come nell'esempio precedente, una sorgente di eventi casuali del tipo mostrato in Figura 17 è realizzata utilizzando essenzialmente la stessa tecnica di [26] (LED + tubo fotomoltiplicatore). Differenza specifica di questa costruzione rispetto a quelle precedentemente descritte che utilizzano il rilevamento di fotoni ad alta velocità e producono? 1 bit per rilevamento [90, 95, 39, 26], è che i rilevamenti casuali vengono effettuati a una frequenza media relativamente bassa di 12,5 MHz, operando così in un regime lontano da tempi morti ed effetti di accumulo producendo una distribuzione esponenziale altamente precisa di intervalli di tempo (Figura 20 a sinistra). Gli intervalli di tempoT_1, T_2, T_3, . . .
sono misurati con una precisione al nanosecondo e i numeri interi distribuiti quasi esponenzialmente così ottenuti sono usati per generare in media ? 14 bit casuali per ogni fotone rilevato che produce ? 160 milioni di bit casuali grezzi al secondo. Le imperfezioni sia nel metodo di estrazione che nell'hardware (timer, rilevatori, sorgente di luce) sono modellate risultando in un convincente limite inferiore dell'entropia media per bit dei bit grezzi. L'entropia media viene quindi migliorata dalla compressione del flusso grezzo mediante funzioni resilienti (vedere la Sezione 3.5) al livello teoricamente indistinguibile dalla vera casualità anche per stringhe di bit di lunghezza irrealistica. L'anello più debole, a nostro avviso, è quest'ultima fase di post-elaborazione perché non è chiaramente dimostrato che le funzioni resilienti siano efficaci contro il tipo specifico di imperfezioni presenti nei bit grezzi, ovvero che i limiti sui bit post-elaborati reggano. Però, i bit grezzi sono già molto vicini alla casualità e l'ulteriore post-elaborazione da parte di funzioni resilienti migliora chiaramente la velocità di superamento dei test statistici indicando che i bit risultanti sono molto vicini alla vera casualità. Infatti, post elaborato

bit a una velocità media di 150 Mbit/sec superano tutti i test statistici rilevanti, nonché alcuni test statistici e algoritmici non divulgati eseguiti dal gruppo di ricerca dell'Università di Twente [35]. Tuttavia, potrebbe essere necessaria una cautela quando vengono utilizzate funzioni resilienti perché alcuni ricercatori [96] sottolineano che le funzioni resilienti sembrano essere limitate nella loro capacità di eliminare gli effetti degli avversari attivi sui bit di uscita.

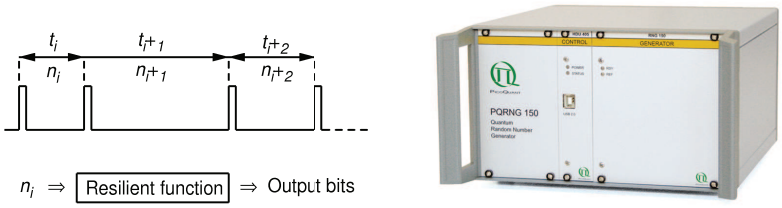


Figura 20: Generatore ottico di numeri casuali quantistici basato su una distribuzione esponenziale altamente precisa dei tempi di rilevamento dei fotoni: foto schematica (a sinistra) del prodotto (a destra). I tempi tra eventi casuali successivi sono misurati da un hardware di temporizzazione molto preciso risultante in numeri interi che rappresentano il tempo. Questi numeri vengono quindi utilizzati per estrarre molto più di 1 bit per fotone rilevato, risultando in una velocità di produzione media complessiva di 150 Mbit/sec ottenuta dopo una post-elaborazione con funzioni resilienti (vedere la Sezione 3.5).

Eppure un esempio di generatore molto veloce (110 Mbit/sec) della costruzione e filosofia simile al precedente è stato presentato in [116, 117], Figura 21. Nel primo articolo, risplende una debole luce continua (proveniente da un LED). su un rivelatore di fotoni che produce eventi casuali (rilevazioni) del tutto simili al sistema generale mostrato in Figura 17. I tempi tra eventi successivi vengono misurati con un orologio ad alta risoluzione per ottenere numeri interi che seguono approssimativamente la distribuzione esponenziale. Questi numeri presentati in forma binaria non producono bit casuali perché sono stati estratti da una distribuzione altamente non uniforme (vale a dire, esponenziale tagliato vicino allo zero al tempo morto). Per ottenere numeri più uniformemente distribuiti, nell'articolo successivo la luce della sorgente (LED) è modellata in impulsi con potenza in forte aumento a partire dall'inizio alla fine di ciascun impulso. L'idea è che utilizzando una forma di impulso accuratamente adattata i tempi tra le successive rilevazioni di fotoni diventerebbero uniformi anziché esponenziali. Ci sono avvertimenti con questo. Innanzitutto, gli intervalli di tempo tra i rilevamenti di fotoni vengono misurati con un orologio a corsa libera che è stato annotato in [95] per portare immediatamente a correlazioni anche se gli eventi casuali in arrivo sono veramente casuali. In secondo luogo, questo schema dipende in modo critico dal fatto che la distribuzione risultante sia esattamente uniforme mentre gli autori ne hanno misurato solo uno approssimativo. Terzo, utilizzando un clock ad altissima velocità, gli autori cercano di "spremere" quanti più bit casuali possono da un singolo evento fotonico (? 20 bit per fotone rilevato) che generalmente porta a una grande amplificazione dell'hardware

imperfezioni che portano quindi a bit casuali grezzi piuttosto scadenti, come in effetti è stato riscontrato. In quarto luogo, i risultati approssimativi relativi alla potenza dell'impulso variabile sono entrambi fondamentali (cioè la potenza dell'impulso dovrebbe tendere all'infinito alla fine dell'impulso proporzionale a $1/(t - t_0)$ dove T_0 è la lunghezza dell'impulso) e pratico (la forma dell'impulso è ottenuta da un circuito analogico, solo parzialmente preciso, quindi non consentendo di condurre correttamente la prova di casualità. Gli autori notano anche che questo circuito produce forti disturbi elettrici nei circuiti vicini che, a nostro avviso, rende non è adatto per la miniaturizzazione a livello di chip.E infine, le basi teoriche per la distribuzione esponenziale dell'arrivo temporale sono tratte da un'ipotesi di campo costante mentre qui la forza del campo elettromagnetico leggero varia notevolmente, quindi anche i motivi teorici per questo generatore non sono puliti .

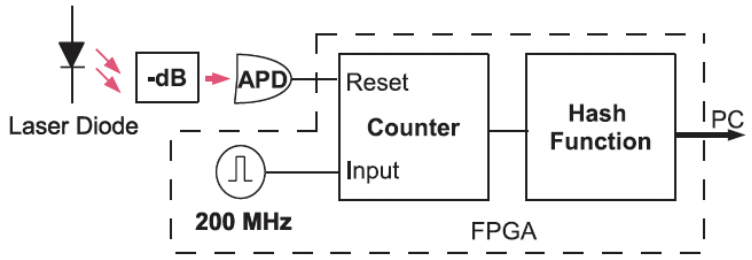


Figura 21: Generatore ottico di numeri casuali quantistici basato su tempi di arrivo di fotoni quasi uniformi da impulsi ottici di forma speciale.

Questo generatore appartiene a un'ampia nicchia di costrutti RNG la cui filosofia generale è quella di produrre dati parzialmente casuali e quindi filtrarli attraverso una funzione hash pseudo-casuale (come SHA256 usata in questo esempio) nella speranza di migliorare la casualità (vedere la Sezione 3.5) . Riteniamo che questo sia un approccio molto problematico ed ecco il nostro ragionamento. La prova della casualità in questo caso si basa sulla stima dell'entropia della sorgente dei bit grezzi e sul processo di amplificazione della casualità mediante hashing. La procedura di hashing non è generalmente infallibile [2] e non consente l'applicazione cieca della funzione hash a un generatore mal costruito. Immaginiamo ad esempio che la sorgente RNG grezza produca alcune sequenze più spesso delle altre (cosa che effettivamente accade se non è casuale). Quindi l'hash di queste sequenze (la funzione hash essendo deterministica) produrrebbe anche alcune sequenze più spesso delle altre, il che significa che anche i bit "corretti" non sarebbero casuali. Una bella conferma di ciò viene proprio da questo esempio: anche dopo l'hashing i bit prodotti non sono completamente casuali e falliscono alcuni test statistici.

Abbiamo visto che le tecniche di emissione e rilevamento di fotoni sono spesso utilizzate nei generatori di numeri casuali quantistici. La velocità di rilevamento dei fotoni degli attuali rilevatori di fotoni singoli è un fattore limitante nella velocità di produzione di bit casuale ottenibile, in particolare per i fotodiodi a valanga (APD) a semiconduttore. Gli APD sono

piccoli e convenienti per il rilevamento di singoli fotoni su scala chip, tuttavia, soffrono di imperfezioni che sono particolarmente dannose per la generazione di numeri casuali e di conseguenza raramente utilizzate per tale scopo. Il problema più grande sono i tempi morti relativamente lunghi (indotti dalla necessità di spegnere la valanga tra i rilevamenti successivi) e l'elevata velocità di postpulsazione (di solito nell'intervallo 1-10%). Per avanzare su questo, Toshiba ha sviluppato uno speciale, cosiddetto approccio di "auto differenziazione" [119] per la lettura dei fotodiodi a valanga a semiconduttore che promette tassi di rilevamento significativamente più elevati (dime morto inferiore) rispetto al solito metodo di spegnimento attivo, sopprimendo gli impulsi di ritorno efficacemente quadratura della probabilità postpulsante. Questa nuova tecnica è stata utilizzata per la generazione di numeri casuali dallo stesso gruppo di autori [20]. Vale a dire,

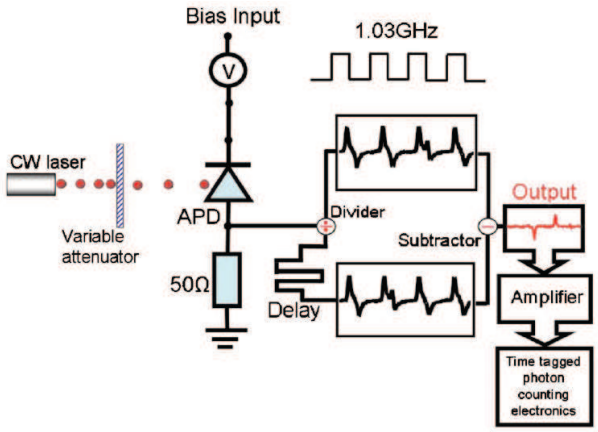


Figura 22: Generatore ottico di numeri casuali quantistici basato su un fotodiodo a valanga (APD) a funzionamento "auto-differenziato" periodicamente. La potenza del laser DFB cw (1550 nm) è regolata (mediante attenuatore variabile) in modo tale che l'intensità del campo elettromagnetico che cade sulla superficie dell'APD provochi circa 0,004 rilevamenti di valanghe per porta, risultando in 4,01 MHz di bit casuali.

Come sorgente luminosa viene utilizzato un laser a feedback distribuito (DFB) in modalità onda continua (cw). La potenza del laser DFB (1550 nm) è regolata (mediante attenuatore variabile) in modo tale che l'intensità del campo elettromagnetico che raggiunge la superficie dell'APD provochi circa 0,004 rilevamenti di valanghe per porta. Quando si verifica il rilevamento, viene generato un nuovo bit casuale e il suo valore è "0" se si è verificato su porta pari o "1" se su porta dispari. Tenere in considerazione

l'efficienza di rilevamento di 0,004 questo metodo produce 4,01 MHz di bit casuali. Questo processo di generazione di bit è intrinsecamente privo di bias (le probabilità di zero e uno sono uguali) ma (ciò che non viene notato dagli autori di questo articolo) esiste un'autocorrelazione negativa intrinseca che aumenta con l'efficienza di rilevamento. Ovvero, nel caso limite di efficienza 1 (una rilevazione per porta) ci sarebbe sempre un “1” dopo lo “0” e viceversa producendo così una sequenza completamente deterministica 01010101· · · che ha autocorrelazione uguale a -1. Anche se gli autori affermano che questo metodo di generazione di numeri casuali potrebbe, in linea di principio, essere esteso a velocità molto più elevate utilizzando una potenza laser più elevata e una velocità di rilevamento fino a 100 MHz (efficienza di 0,100), è chiaro che a quel punto l'autocorrelazione sarebbe quantità approssimativamente -0,1 e i bit non supererebbero alcun test di casualità.

Esistono numerose altre varianti dei principi dello spazio e del tempo che si possono trovare nella letteratura scientifica e brevettuale.

In conclusione, la caratteristica più distintiva di un approccio quantistico alla generazione di numeri casuali è che, finalmente in linea di principio, rende possibile stabilire una semplice relazione tra casualità dei numeri, processo fisico sfruttato e imperfezioni di implementazione, offrendo così una possibilità per la ricerca scientifica prova di casualità. Attente realizzazioni pratiche si avvicinano sufficientemente all'idealizzazione teorica e consentono una valutazione indipendente degli effetti delle imperfezioni dell'implementazione i cui effetti possono, se necessario, essere affrontati dal postprocessing della teoria dell'informazione (vedere la Sezione 3.5). Oltre a ciò, esistono processi di rilevamento casuale quantistico che sono intrinsecamente altamente insensibili alle radiazioni elettromagnetiche (ad esempio, l'amplificazione a valanga nei fotodiodi a semiconduttore) offrendo così l'immunità alla manipolazione del canale laterale da parte di campi esterni. A causa di tutto ciò, gli RNG quantistici sono la scelta migliore per la generazione di veri numeri casuali per la crittografia e altre applicazioni che richiedono in modo critico veri numeri casuali. L'inconveniente più significativo delle soluzioni attuali è che fanno uso di oggetti fisici ingombranti e quindi non possono essere miniaturizzati a livello di chip utilizzando le attuali tecnologie. Inoltre, a causa dell'uso frequente di rilevatori di fotoni, i QRNG sono in genere molto costosi e molto più lenti dei PRNG software. Fortunatamente, la nascente scienza e tecnologia dei chip ottici offre una strada promettente per RNG quantistici veloci, in miniatura e convenienti e si possono prevedere progressi significativi in questo entusiasmante campo nel prossimo futuro. Gli RNGs quantistici sono la scelta migliore per la generazione di veri numeri casuali per la crittografia e altre applicazioni che richiedono in modo critico veri numeri casuali. L'inconveniente più significativo delle soluzioni attuali è che fanno uso di oggetti fisici ingombranti e quindi non possono essere miniaturizzati a livello di chip utilizzando le attuali tecnologie. Inoltre, a causa dell'uso frequente di rilevatori di fotoni, i QRNG sono in genere molto costosi e molto più lenti dei PRNG software. Fortunatamente, la nascente scienza e tecnologia dei chip ottici offre una strada promettente per RNG quantistici veloci, in miniatura e convenienti e si possono prevedere progressi significativi in questo entusiasmante campo nel prossimo futuro. Gli RNGs quantistici sono la scelta migliore per la generazione di veri numeri casuali per la crittografia e altre applicazioni che richiedono in modo critico veri numeri casuali. L'inconveniente più significativo delle soluzioni attuali è che fanno uso di oggetti fisici ingombranti e quindi non possono essere miniaturizzati a livello di chip utilizzando le attuali tecnologie. Inoltre, a causa dell'uso frequente di rilevatori di fotoni, i QRNG sono in genere molto costosi e molto più lenti dei PRNG software. Fortunatamente, la nascente scienza e tecnologia dei chip ottici offre una strada promettente per RNG quantistici veloci, in miniatura e convenienti e si possono prevedere progressi significativi in questo entusiasmante campo nel prossimo futuro. Inoltre, a causa dell'uso frequente di rilevatori di fotoni, i QRNG sono in genere molto costosi e molto più lenti dei PRNG software. Fortunatamente, la nascente scienza e tecnologia dei chip ottici offre una strada promettente per RNG quantistici veloci, in miniatura e convenienti e si possono prevedere progressi significativi in questo entusiasmante campo nel prossimo futuro. Inoltre, a causa dell'uso frequente di rilevatori di fotoni, i QRNG sono in genere molto costosi e molto più lenti dei PRNG software.

3.5 Post-elaborazione

I veri generatori di numeri casuali non possono mai essere resi perfetti e quindi di solito è richiesta una post-elaborazione. Esistono numerosi algoritmi di post elaborazione il cui scopo è eliminare le imperfezioni presenti nei numeri casuali "grezzi" prodotti da generatori fisici. Una buona recensione di

i metodi di post-elaborazione sono forniti in [96]. Qui ci limiteremo a categorizzare e descrivere brevemente i principi principali

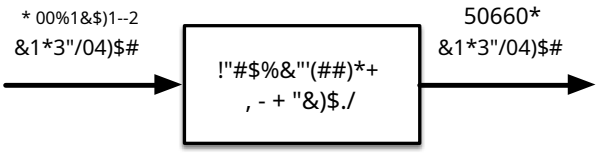


Figura 23: Schema generale della post-elaborazione dei numeri casuali.

L'idea generale della post-elaborazione (Figura 23) è sacrificare una certa percentuale di bit per arrivare a un insieme più piccolo ma più casuale. Esistono fondamentalmente quattro tecniche:

1. correttori semplici ad hoc;
2. sbiancamento con funzioni di hash crittografico;
3. algoritmi di estrazione [86, 87]; e
4. funzioni resilienti [10, 80, 48, 49].

Sebbene vi sia una "zona grigia" di quale parte della produzione di numeri casuali appartiene al metodo di estrazione dei bit e quale alla post-elaborazione, l'estrazione dei bit è solitamente un primo e molto semplice passaggio che converte la misurazione fisica di un segnale analogico o digitale in " numero binario casuale digitale grezzo (come la digitalizzazione del rumore analogico tramite un comparatore di soglia mostrato nella Figura 2), mentre la post-elaborazione è un processo più complesso progettato per ridurre o spostare completamente le imperfezioni che sono necessariamente presenti dopo il primo passaggio. Sebbene l'estrazione dei bit venga sempre eseguita in hardware, gli algoritmi di post-elaborazione sono generalmente così complicati che possono essere eseguiti solo da un computer (o un microcontrollore o FPGA), sebbene le tecniche di post-elaborazione più preziose siano quelle abbastanza semplici da essere adatte per l'implementazione diretta in hardware.

In generale, la post-elaborazione richiede molte risorse e offusca. A nostro avviso, un buon vero RNG dovrebbe essere privo di post-elaborazione o utilizzare una post-elaborazione ad hoc minima. Le tecniche di post-elaborazione più diffuse possono essere classificate in quattro famiglie come descritto di seguito.

3.5.1 Correttori semplici ad hoc

Esempi di correttori ad hoc sono: XORing due o più bit vicini da uno stesso RNG [87], omettendo bit (decimatore), alimentando un LSFR con numeri casuali imperfetti [101], rimescolamento latino di bit quadrati [54], von Neumann [63] e Peres [67] de-biasing, XORing di due o più RNG che funzionano in parallelo [50, 15] ecc.

È importante notare che un'elaborazione ad hoc e ingenua può portare a problemi imprevisti. Ad esempio, di solito è considerata una buona idea applicare lo schema di de-polarizzazione di von Neumann [63] al fine di rimuovere completamente qualsiasi distorsione dalla sequenza di bit. Lo schema funziona come segue. La sequenza di bit distorta viene tagliata in una sequenza di coppie di bit non sovrapposte. Le coppie 11 e 00 vengono scartate, 01 viene convertito in "0" e 10 viene convertito in "1". Mentre si è tentati di pensare che la probabilità di occorrenza di 10 sia uguale alla probabilità di 01 (e quindi la sequenza risultante non ha pregiudizi), spesso si trascura che questo è vero solo se i bit sono completamente indipendenti (nessuna correlazione). Il seguente esempio estremo illustra come miseramente la procedura di von Neumann possa fallire. Consideriamo la sequenza: 101010101010· · ·. Ovviamente non ha pregiudizi. Dopo l'applicazione del de-polarizzazione di von Neumann, la sequenza recita: 111111· · · che è una sequenza massimamente distorta. La ragione di questo risultato inaspettato è che la sequenza originale è massimamente (anti-correlata) e quindi abbastanza lontana dall'assunzione di una completa indipendenza statistica. Generalmente, se la stringa grezza è correlata, la procedura di de-biasing della nave può persino aumentare la distorsione o creare altre carenze statistiche impreviste. D'altra parte, i correttori ad hoc semplici e di facile comprensione hanno il vantaggio, rispetto a procedure più complesse, di essere più facili da includere in una prova di casualità.

3.5.2 Funzioni crittografiche unidirezionali (hash)

La funzione di hash unidirezionale è una funzione matematica il cui dominio è un intero insieme di numeri interi e il cui output è un numero binario di esattamente n bit, dove n solitamente è compreso tra 128 e 512. Le funzioni hash sono caratterizzate da due requisiti:

- Dato un valore di output non esiste un modo più rapido per trovare un input corrispondente rispetto a un'ipotesi casuale (ovvero una funzione hash è "unidirezionale");
- La probabilità che due input diversi diano lo stesso output è minore o uguale a 1/2N.

Una delle tecniche di post-elaborazione più popolari è lo "sbiancamento" dell'output di un TRNG mediante una funzione di hash crittografico. come MD5, SHA-1, SHA-2, SHA-256, SHA-512. Molti autori credono che un cattivo RNG che non superi i test statistici, eseguito attraverso una procedura di compressione hash "crittografica" diventerebbe magicamente molto buono, senza effettivamente dimostrare alcuna comprensione teorica sul perché questo dovrebbe essere il caso. In effetti, un esempio molto interessante fornito in [117] dimostra che l'hashing di un generatore difettoso può non aumentare la casualità abbastanza da superare i test statistici.

Dal punto di vista delle prestazioni, l'implementazione di una funzione di hash nei chip hardware richiede molte risorse, quindi nella maggior parte dei casi l'hashing viene eseguito su un computer; due esenzioni a questa regola sono il suddetto RNG di Intel in

Figura e VIA C3 in Figura 12, che utilizzano SHA-1 cablato proprio accanto all'RNG sullo stesso chip. Per quanto riguarda la dimostrabilità della casualità dell'output hash, anche se risultati interessanti sull'amplificazione della privacy (e della casualità) sono stati teoricamente esercitati per le Universal Hash Function(s) di Wegman [6], nel caso di funzioni hash black-box della vita reale (che probabilmente contengono debolezze statistiche o di sicurezza sconosciute) è difficile eseguire una prova convincente di casualità. Ad esempio una funzione hash può contenere problemi statistici come ad esempio alcune stringhe di output sono più probabili di altre che verrebbero poi ereditate dai bit di output anche se la funzione è alimentata da numeri casuali perfetti.

3.5.3 Funzioni dell'estrattore

Un approccio più approfondito alla guarigione della casualità è offerto dalla giovane teoria degli estrattori [86]. Un estrattore di casualità è un algoritmo che converte una lunga sequenza debolmente casuale in una sequenza più breve con casualità quasi perfetta. Per alcune fonti di casualità esistono estrattori dimostrabili IT, ma attualmente non esiste un singolo estrattore di casualità che abbia dimostrato di funzionare se applicato alla cieca a qualsiasi tipo di fonte ad alta entropia. Il problema con gli algoritmi di estrazione è che richiedono un buffer di memoria e molta CPU che rallenta il bit rate di output complessivo.

Le funzioni di estrazione per la post-elaborazione di veri generatori di numeri casuali sono state proposte da Barak, Shaltiel e Tomer [2]. Lo scopo iniziale era ottenere progetti resistenti ai cambiamenti nei generatori fisici dovuti, ad esempio, all'invecchiamento, alle variazioni di temperatura o agli attacchi. Le funzioni di estrazione sono funzioni senza stato con proprietà quantificabili originariamente sviluppate come strumento per la teoria della complessità. Il suddetto gruppo di autori ha sviluppato un modello matematico per catturare l'influenza di un avversario sulla sorgente di casualità e fornire una costruzione esplicita basata su funzioni hash universali che è dimostrata per le sue proprietà di output anche se esistono correlazioni non locali nella sorgente di input.

Maggiori informazioni sulla teoria e la pratica degli estrattori possono essere trovate in [87].

3.5.4 Funzioni resilienti

Ancora un altro approccio per migliorare la casualità filtrando attraverso un processo deterministico è l'uso di funzioni resilienti che sono state introdotte da Sunar, Martin e Stinson in [98] come fase di post-elaborazione per un progetto FRO RNG. L'idea è, secondo gli autori, di "filtrare eventuali bit deterministici"

dalla stringa grezza nell'ambiente in cui alcuni bit potrebbero essere sotto il controllo di un utente malintenzionato e quei bit sono quindi considerati "deterministi". Gli autori di [98] studiano il grado di resilienza della procedura contro gli avversari attivi (da qui il nome di queste funzioni). In breve, un (n, m, k)-la funzione resiliente è una funzione $f : F_n \rightarrow F_m$ tale che ogni possibile uscita m-è ugualmente probabile che la tupla si verifichi quando i valori di K i bit di input sono fissi e i rimanenti $n - k$ i bit sono scelti a caso. Gli elementi di F sono i valori binari 0 e 1. L'importante caratteristica distintiva delle funzioni resilienti è che sono state costruite specificamente per annullare gli attacchi su (certe percentuali di) bit casuali - un punto di grande importanza nelle applicazioni crittografiche di numeri casuali (vedi Sezioni 4- 6).

Maggiori informazioni sulla teoria e sulla pratica delle funzioni resilienti possono essere trovate in [10, 98, 80, 48, 49].

4 Valutazione della casualità (test)

La nozione più importante sui test statistici è la seguente: se un generatore supera tutti i test statistici conosciuti questo non prova che sia casuale: significa solo che supera tutti i test di casualità attualmente conosciuti. Domani può fallire qualche nuovo test o fallisce già nel modo noto solo ai suoi costruttori.

La maggior parte dei test di casualità controlla una o più proprietà statistiche di lunghe sequenze di numeri casuali, ad esempio bias, autocorrelazione seriale ecc. Alcune compilazioni di test sono più orientate verso problemi nei PRNG (es. DIEHARD [54]) altre sui veri RNG (es. , ENT [112]) mentre alcuni sono di natura generale (es. Universal Test [59], NIST STS [79]). Il fatto spiacevole è che esiste un numero infinito di proprietà statistiche che i numeri veramente casuali devono soddisfare. I test stessi non sono perfetti: alcuni contengono errori scoperti in seguito [50, 76] o costanti di precisione discutibile ottenute mediante simulazione utilizzando generatori di numeri casuali "fidati" come la combinazione di rumore bianco e "rumore nero" [54].

L'esecuzione di un set completo di test richiede molte ore di CPU: per testare 1E9 bit con NIST STS ci vogliono circa 6 ore sulla CPU single core più veloce mentre per produrre così tanti bit con un QRNG commerciale ci vogliono tra 7 e 250 secondi.

I test di casualità richiedono molto tempo: ci vuole molto meno tempo per generare numeri che per testarli. Tuttavia, il test di casualità è importante per i costruttori di RNG. Pertanto in alcuni casi in cui ci si può ragionevolmente aspettare solo un certo tipo di imperfezioni (soprattutto per gli RNG quantistici) si tenderà a utilizzare solo test speciali sensibili a queste particolari imperfezioni per arrivare a test più efficienti.

5 numeri casuali nella crittografia quantistica

La crittografia quantistica è un protocollo di accordo pubblico di una chiave crittografica simmetrica, il che significa che se due parti A e B possiedono una piccola chiave segreta comune, utilizzando questo protocollo saranno in grado di stabilire una chiave segreta comune di qualsiasi lunghezza. Questa funzione crittografica è anche nota come "crescita della chiave segreta". L'obiettivo finale di stabilire una chiave segreta lunga è utilizzarla come un one-time pad e ottenere così il trasferimento dei dati in assoluta segretezza. Esistono diversi protocolli QC matematicamente identici. Il primo, dal nome dei suoi creatori come BB84, è apparso nel 1984 ed è stato realizzato sperimentalmente nel 1991 [4].

Nello scenario BB84, Alice e Bob sono collegati tramite due canali diversi: il canale quantistico (solitamente fibra ottica ben schermata) in grado di condurre singoli fotoni di luce e un canale “classico” non protetto come una linea telefonica, un ponte radio o Internet.

Ecco lo schema semplificato di come funziona il protocollo: Alice può preparare fotoni in diversi stati di polarizzazione. Per stabilire una chiave segreta, Alice invia a Bob una sequenza di numeri casuali codificati nelle polarizzazioni dei fotoni come segue: "1" è codificato in modo equiprobabile come polarizzazione lineare-verticale (LV) o circolare sinistra (LC), mentre "0 ” codificato in modo equiprobabile come lineare-orizzontale (LH) o circolare a destra (RC). Bob, dispone di due analizzatori di polarizzazione: uno in grado di misurare correttamente le polarizzazioni lineari (L) e l'altro in grado di misurare correttamente le polarizzazioni circolari (C). Alice sceglie una polarizzazione a caso, prepara il fotone e lo invia a Bob. Bob sceglie a caso uno dei due analizzatori e con esso misura il fotone ricevuto da Alice. Se per caso Bob ha scelto il polarizzatore giusto e riceverà 0 o 1 come inviato da Alice. Se Bob ha scelto il polarizzatore sbagliato riceverà 0 o 1 con uguale probabilità indipendentemente da ciò che Alice ha inviato. Quindi, dopo aver ricevuto un fotone da Alice, Bob annuncia (su un canale pubblico autenticato ma nitido) quale polarizzatore ha appena usato (L o C). Nota che questo non dice nulla al potenziale intercettatore sul valore del bit che Bob ha. Alice risponde con "Keep it" o "Cestino". Quindi, poco a poco, i due stanno costruendo la loro chiave segreta. Le leggi della meccanica quantistica impediscono che i qubit vengano copiati fedelmente in modo che un intercettatore possa ottenere solo un'informazione limitata sulla stringa di Alice e Bob e inoltre l'intercettazione possa essere rilevata da Alice e Bob. Se Bob ha scelto il polarizzatore sbagliato riceverà 0 o 1 con uguale probabilità indipendentemente da ciò che Alice ha inviato. Quindi, dopo aver ricevuto un fotone da Alice Bob annuncia (su un canale pubblico autenticato ma nitido) quale polarizzatore ha appena usato (L o C). Nota che questo non dice nulla al potenziale intercettatore sul valore del bit che Bob ha. Alice risponde con "Keep it" o "Cestino". Quindi, poco a poco, i due stanno costruendo la loro chiave segreta. Le leggi della meccanica quantistica impediscono che i qubit vengano copiati fedelmente in modo che un intercettatore possa ottenere solo un'informazione limitata sulla stringa di Alice e Bob e inoltre l'intercettazione possa essere rilevata da Alice e Bob. Se Bob ha scelto il polarizzatore sbagliato riceverà 0 o 1 con uguale probabilità indipendentemente da ciò che Alice ha inviato. Quindi, dopo aver ricevuto un fotone da Alice Bob annuncia (su un canale pubblico autenticato ma nitido) quale polarizzatore ha appena usato (L o C). Nota che questo non dice nulla al potenziale intercettatore sul valore del bit che Bob ha. Alice risponde con "Keep it" o "Cestino". Quindi, poco a poco, i due stanno costruendo la loro chiave segreta. Le leggi della meccanica quantistica impediscono che i qubit vengano copiati fedelmente in modo che un intercettatore possa ottenere solo un'informazione limitata sulla stringa di Alice e Bob e inoltre l'intercettazione possa essere rilevata da Alice e Bob. Nota che questo non dice nulla al potenziale intercettatore sul valore del bit che Bob ha. Alice risponde con "Keep it" o "Cestino". Quindi, poco a poco, i due stanno costruendo la loro chiave segreta. Le leggi della meccanica quantistica impediscono che i qubit vengano copiati fedelmente in modo che un intercettatore possa ottenere solo un'informazione limitata sulla stringa di Alice e Bob e inoltre l'intercettazione possa essere rilevata da Alice e Bob. Nota che questo non dice nulla al potenziale intercettatore sul valore del bit che Bob ha. Alice risponde con "Keep it" o "Cestino". Quindi, poco a poco, i due stanno costruendo la loro chiave segreta. Le l

È semplice vedere che l'intero protocollo sarebbe completamente insicuro se solo l'intercettatore potesse calcolare (o prevedere) i numeri casuali di Alice o i numeri casuali di Bob o entrambi. Dall'analisi del tasso di chiave segreto presentato in [5] è ovvio che qualsiasi prevedibilità di numeri casuali da parte dell'intercettatore gli farebbe trapelare informazioni rilevanti, diminuendo così il tasso di chiave effettivo. È intrigante (e ovvio) che nel caso in cui l'intercettatore potesse calcolare esattamente i numeri, il potenziale crittografico del protocollo BB84 sarebbe zero. Questo esempio mostra

che i generatori di numeri casuali locali assunti in BB84 sono essenziali per la sua sicurezza e non dovrebbero essere dati per scontati.

Oltre a quanto descritto sopra, il protocollo BB84 ha altri due sottoprotocolli. Vale a dire, a causa dell'incoerenza quantistica, le perdite nel canale quantistico o le intercettazioni di Alice e Bob non avranno esattamente le stesse stringhe di bit dopo la prima fase, sebbene le due stringhe avranno molte informazioni comuni. Pertanto il secondo sottoprotocollo, la "riconciliazione dati", viene utilizzato per equalizzare le due stringhe, anche se a costo di far trapelare alcune piccole informazioni a un intercettatore. Fortunatamente, Alice e Bob possono calcolare un limite inferiore delle loro informazioni reciproche dopo le due fasi iniziali e quindi eseguire la fase di amplificazione della privacy per arrivare a una chiave più breve ma molto più privata. Questi due sottoprotocolli richiedono ulteriori numeri casuali.

Il protocollo BB84 è considerato un'informazione teoricamente provata [88, 31], il che significa che un utente malintenzionato semplicemente non ha informazioni sufficienti per calcolare il testo in chiaro anche se dispone di infinite risorse di calcolo. Ciò è in forte contrasto con la "crittografia deterministica" ampiamente utilizzata in cui un utente malintenzionato ha informazioni sufficienti per calcolare la chiave, tranne per il fatto che probabilmente richiederebbe risorse di calcolo e/o tempo insormontabili. L'avvertenza con QC è che la prova di sicurezza vale solo contro la famiglia di attacchi considerati nella prova. Sfortunatamente, con il tempo, è diventato evidente che sono possibili attacchi inaspettati al controllo di qualità che utilizzano vari effetti quantistici, il che rende il controllo di qualità molto meno "intoccabile".

Ad esempio nel 2007 un gruppo del MIT ha presentato un attacco che ha fornito a Eve fino al 100% delle informazioni sulla chiave, anche se a spese di un BER elevato [44], ma l'attacco è stato classificato in modo rassicurante come "solo simulazione" perché presumeva che Eve ha un'informazione specifica sul ricevitore di Bob che a quanto pare non è riuscita a ottenere.

Come con qualsiasi altra procedura crittografica, alcuni problemi nell'implementazione del protocollo nel mondo reale, in particolare del canale quantistico e dei rilevatori di fotoni reali, potrebbero essere utilizzati per indebolire la sicurezza crittografica del protocollo e aprire percorsi per gli attacchi.

Makarov et al. nel 2010 [52, 27]. La dimostrazione è stata effettuata sui sistemi di controllo qualità commerciale della società svizzera IdQuantique, con sede a Ginevra, in Svizzera, e uno da MagiQ Technologies, con sede a Boston, Massachusetts. Sono possibili e sono stati proposti miglioramenti che renderebbero il QC resistente a tali attacchi [52], ma la lezione appresa da ciò è che anche i protocolli la cui base teorica è dimostrata sicura in alcuni scenari non devono essere automaticamente considerati immuni a tutti gli attacchi pratici. L'attacco è stato reso possibile perché gli autori hanno trovato un modo per manipolare il generatore di numeri casuali presso la stazione ricevente sfruttando i punti deboli dei rilevatori di fotoni singoli. Per peggiorare le cose,

Questo è un altro esempio dell'importanza degli RNG (locali) per la sicurezza di uno schema crittografico.

In conclusione, il protocollo crittografico quantistico BB84 richiede che sia Alice che Bob possiedano i loro generatori di numeri casuali dimostrabili (locali) privati. Questo è un requisito altamente critico. Nota che un server pubblico di numeri casuali non può sostituire i generatori locali perché i numeri casuali dovrebbero essere consegnati ad Alice e Bob in perfetta segretezza, in primo luogo, e il server dovrebbe essere attendibile.

6 numeri casuali nella crittografia statistica

La crittografia statistica è stata inventata da U. Maurer nel 1991. Il cosiddetto protocollo SKAPD [60] è simile alla crittografia quantistica e consiste allo stesso modo di tre sottoprotocolli. Infatti gli ultimi due sottoprotocolli (la riconciliazione dei dati e l'amplificazione della privacy) sono gli stessi del QC. Tuttavia, il primo sottoprotocollo, denominato "Advantage Distillation" (AD) è completamente diverso e non coinvolge il canale quantistico, il che potenzialmente lo rende molto più pratico. Invece, richiede qualcosa chiamato "canale binario con rumore" che è teoricamente un canale di comunicazione classico integrato con un RNG dimostrabile.

La condizione per un accordo chiave di successo è che prima del protocollo AD le informazioni comuni condivise da Alice e Bob siano maggiori delle informazioni comuni condivise da Alice ed Eve o Bob ed Eve.

Il problema pratico con SKAPD è che contiene una "fase zero" non detta in cui Alice e Bob ottengono le loro stringhe iniziali di bit parzialmente correlate che soddisfano la condizione di cui sopra. Non esiste un modo plausibile noto per rendere possibile la fase zero, sebbene siano stati proposti alcuni scenari (scansione della superficie della Luna, ascolto del rumore proveniente da galassie lontane, acquisizione di grandi quantità di dati Internet, ecc.).

7 numeri casuali nella crittografia deterministica

Ciò che chiamiamo "crittografia deterministica" in questo capitolo è ciò che è ampiamente noto come "crittografia". Alcuni autori usano il nome di "crittografia matematica". È la crittografia contemporanea basata sulla difficoltà di calcolare logaritmi discreti in gruppi di Galois e gruppi di curve ellittiche, e anche la fattorizzazione del numero composto in numeri primi. Ha anche bisogno e utilizza dati casuali; un eccellente breve sondaggio è dato in [7]. Poiché tutti questi protocolli di sicurezza sono per definizione deterministici e quindi reversibili, l'unica vera risorsa di sicurezza è quella parte non deterministica: una chiave o una

dati che dovrebbero essere "casuali". La qualità e la dimostrabilità della casualità sono quindi cruciali per la sicurezza dell'intero sistema.

È il fatto che la crittografia deterministica è l'unica nell'uso più ampio e che la maggior parte dei crittografi non è a conoscenza o non si preoccupa dell'esistenza né della crittografia quantistica né della crittografia statistica perché apparentemente non sono ancora pratiche o sufficientemente affidabili. Pertanto è importante esplorare cosa rende sicuri i protocolli di livello commerciale contemporanei e cosa si potrebbe fare per ottenere la massima sicurezza da essi. La nostra ipotesi è che se un protocollo richiede numeri casuali, l'uso di un TRNG ne massimizza la sicurezza. Senza l'ambizione di fare una prova rigorosa o di dare una recensione completa qui, diamo un'occhiata a diversi esempi a sostegno di questa ipotesi.

1. Il protocollo di creazione della chiave Diffie-Hellman [19] abilita la stessa funzionalità dei protocolli BB84 e SKAPD sopra menzionati e viene utilizzato ad esempio nel protocollo "https" per stabilire una chiave di sessione. Il protocollo richiede a entrambe le parti (Alice e Bob) di generare dati casuali privati e, dopo alcune operazioni, di inviarli l'uno all'altro. La versione più resistente di DH richiede ulteriori dati casuali utilizzati per le firme digitali. Una vulnerabilità del PRNG costruita in una prima versione del browser Internet Netscape ha portato alla completa compromissione del successivo protocollo crittografico. Un esempio è l'attacco ai dati e alle chiavi di crittografia di Netscape a 40 bit RC4-40 [75], che è stato in grado di violare il protocollo https in un minuto circa, come descritto in [28].

2. Il protocollo a chiave pubblica RSA si basa sulla generazione di chiavi pubbliche e private separatamente da Alice e Bob. Per creare una coppia di chiavi privata/pubblica è necessario generare due numeri primi univoci e grandi. Già il calcolo dei candidati per numeri primi implica numeri casuali. Successivamente, i candidati devono essere testati per la primalità utilizzando l'algoritmo Miller-Rabin che richiede numeri casuali come basi per testare correttamente la primalità. Ulteriori numeri casuali una tantum possono essere utilizzati nel processo di comunicazione effettiva. Laddove i bit casuali fisici ad alta entropia non sono disponibili o richiedono molto tempo (come in un tipico computer PC) c'è la tendenza a "espandere" una stringa casuale corta in una lunga con metodi pseudo-casuali.

3. Allo stesso modo, una ricerca di attacco crittografico su un generatore di numeri parzialmente pseudo-casuali di un sistema crittografico commerciale basato su AES è descritta in [77].

Per concludere, nella crittografia deterministica i numeri casuali sono l'unica parte del protocollo che differisce da punto a punto e inoltre la loro vera casualità è talvolta una prerogativa per calcoli corretti.

Pertanto, anche se la maggior parte delle primitive di crittografia deterministiche non sono sicure, l'utilizzo di numeri casuali reali garantisce la massima sicurezza ottenibile con questi metodi.

8 Problemi aperti e Outlook

In questo articolo di indagine cerchiamo di mostrare l'importanza dei numeri casuali per la forza dei protocolli crittografici non solo per le crittografie quantistiche e stocastiche in cui i numeri casuali sono una parte essenziale dei dati scambiati tra le parti comunicanti, ma anche per la crittografia deterministica contemporanea in cui l'imprevedibilità e l'entropia massima di i numeri casuali utilizzati in esso massimizzano la forza crittografica complessiva.

I veri generatori di numeri casuali sembrano essere di uso modesto, anche se alcune aziende ne ricavano un buon profitto [37]. Dai dati disponibili, sembra che i TRNG siano venduti principalmente alle società di gioco d'azzardo online, alle agenzie di sicurezza statali e all'industria dell'etichettatura e dei test dei prodotti. Al momento della stesura di questo sondaggio, i principali problemi che impediscono un uso più diffuso di veri generatori di numeri casuali in generale sono:

1. La mancanza di progetti di generatori le cui prove di casualità sarebbero allo stesso tempo corrette, convincenti e dimostrate resistenti alle attese imperfezioni dell'hardware;
2. La (diffusa) mancanza di comprensione del fatto che la pseudocasualità non può essere utilizzata come sostituto della vera casualità in così tante applicazioni, in particolare: crittografia sia classica che quantistica, sicurezza informatica, simulazioni Monte Carlo, lotteria, test di prodotti e loro funzionalità e molti di più.
3. Il prezzo elevato dei veri generatori di numeri casuali;
4. La mancanza di supporto di veri generatori di numeri casuali in vari software popolari che richiedono numeri casuali che li rendono difficili da usare.

9 Commenti e riferimenti aggiuntivi

Una differenza distintiva tra PRNG e TRNG è la dimostrabilità di quest'ultimo. Infatti, l'unica caratteristica dimostrabile di un PRNG è che non è casuale perché tutti i numeri da esso prodotti possono essere calcolati da un unico numero iniziale: il seme. D'altra parte, i TRNG sembrano essere inevitabilmente afflitti da "piccole imperfezioni" nell'hardware che si trasformano in deviazioni misurabili dalla casualità che richiedono la post-elaborazione. Ma la post-elaborazione complessa offusca o indebolisce la convincenza dell'eventuale prova di casualità. Inoltre, una pratica di trattenere le informazioni sul principio di funzionamento

di un TRNG così come una prova scientifica della sua casualità sembrano essere quasi una regola quando si tratta di TRNG commerciali. I produttori giustificano ciò con la necessità di proteggere la loro proprietà intellettuale e la loro tecnologia. Sebbene tale giustificazione vada bene quando riguarda prodotti comuni (ad esempio una lavastoviglie), è esattamente ciò che rovina lo scopo di un TRNG perché senza una chiara visione della tecnologia e della prova di casualità di un TRNG si ricade nella situazione di non dimostrabilità di un PRNG. D'altra parte, nelle pubblicazioni scientifiche le prove di casualità sono offerte anche molto raramente, probabilmente perché la prova è la parte difficile della ricerca mentre non sembra essere richiesta dagli editori delle riviste scientifiche. La maggior parte dei ricercatori quindi ricorre alla strategia dell'azione minima: creare un TRNG, ottenere almeno una sequenza di numeri casuali che supera l'insieme scelto di test di casualità e pubblicare. Tuttavia, senza un'indagine dettagliata della sensibilità della casualità estratta su piccole variazioni nell'hardware e senza prove di casualità, un progetto scientifico non può procedere verso un prodotto. A nostro avviso questa situazione è andata migliorando e continuerà a migliorare molto lentamente nel tempo, garantendo così longevità e freschezza alla ricerca dei TRNG.

Anche se esiste un'ampia raccolta di pubblicazioni che documentano il fatto che i PRNG possono fallire nel loro scopo come generatori di numeri casuali [66, 69, 11, 45, 51, 12, 58, 23, 32, 85, 104, 40, 21, 93] vediamo che i PRNG sono ancora molto più diffusi dei TRNG anche nelle applicazioni più critiche. Tra le ragioni di ciò c'è anche il fatto che i PRNG sono molto più convenienti, semplici ed economici da usare rispetto ai TRNG, ma anche l'onnipresente mancanza di comprensione di cosa sia la casualità e cosa non sia supportata dalla non esistenza di una definizione ampiamente accettata di casualità [45]. Chiaramente, è necessaria un'ulteriore ricerca su questo argomento.

Tutti i TRNG commerciali la cui velocità è di almeno 1 Mbit/sec sono ingombranti e il prezzo è compreso tra $ 5-25k, che è più costoso della maggior parte dei software che utilizzerebbero un TRNG. Pertanto, generalmente non conviene a un produttore di software rendere il suo prodotto molto più costoso richiedendo un TRNG di terze parti per la generazione di numeri casuali. In casi estremamente rari, tuttavia, un software è stato sposato con un TRNG selezionato: ad esempio Mathematica e Quantis (da una terza parte) [62].

I TRNG commerciali in genere vengono forniti con driver che supportano il trasferimento di numeri casuali a linguaggi di programmazione come Pascal o C++ su sistemi operativi selezionati, ad esempio [37], utilizzando una subroutine specifica del prodotto o una funzione di libreria di programma. Questo è probabilmente il massimo che un produttore può ragionevolmente fare per supportare il proprio prodotto. D'altra parte, la maggior parte del software commerciale o gratuito che utilizza o necessita di numeri casuali non supporta alcun TRNG. Questo significa che avendo un software precompilato non c'è praticamente modo di collegarlo a nessun TRNG. L'unica soluzione praticabile per includere un TRNG in un pacchetto software sarebbe scriverlo da zero e includere in esso una specifica chiamata di funzione associata allo specifico TRNG scelto.

o altre periferiche comuni) si potrebbe supportare solo un TRNG specifico per sforzo di programmazione. A nostro avviso è chiaro che fino a quando non ci sarà un modo standardizzato per accedere ai TRNG, o meglio ancora, finché i TRNG non saranno fisicamente integrati nei computer e saranno accessibili nei principali linguaggi di programmazione, la loro popolarità rimarrà minima.

Mentre è chiaro che la vera casualità non può essere generata da operazioni deterministiche e che quindi deve basarsi su fenomeni fisici, il problema di generare una casualità sufficientemente buona e la dimostrabilità della casualità rimangono i principali problemi aperti con gli RNG fisici. Nuove direzioni nello sviluppo di generatori fisici di numeri casuali si concentreranno probabilmente su dispositivi autocalibranti [46, 103] o non calibrabili [95] con fenomeni quantistici fondamentalmente casuali come fonte di casualità.

Riferimenti

1. V. Bagini e M. Bucci. Un progetto di un vero generatore di numeri casuali affidabile per applicazioni crittografiche.*Hardware crittografico e sistemi integrati (CHES)*, C. K. Ko¸ç e C. Paar, Eds., pagine 204-218, Springer 2002.
2. B. Barak, R. Shaltiel e E. Tromer. I veri generatori di numeri casuali sono al sicuro in un ambiente in evoluzione.*Hardware crittografico e sistemi integrati (CHES)*, C. D. Walter, . K. Ko¸ç e C. Paar, Eds., pagine 166-180, Springer 2003.
3. CWJ Beenakker e M. Büttiker. Soppressione del rumore di sparo in conduttori diffusivi metallici.*Fis. Rev. B*, 46:1889-1892, 1992.
4. CH Bennett e G. Brassard. Crittografia quantistica: distribuzione di chiavi pubbliche e lancio di monete. In*Atti della conferenza internazionale IEEE su computer, sistemi ed elaborazione dei segnali*, pagine 175-179, 10-12 dicembre 1984.
5. CH Bennett, F. Bessette, G. Brassard, L. Salvail e J. Smolin. Crittografia quantistica sperimentale.*Giornale di Crittologia*, 5(1):3-28, 1992.
6. CH Bennett, TJ Watson, G. Brassard, C. Crepeau e UM Maurer. Amplificazione generalizzata della privacy.*IEEE Trad. Far sapere. Teoria*, 41(6):1915-1923, novembre 1995.
7. DJ Bernstein, J. Buchmann e E. Dahmen, eds. *Crittografia post-quantistica*. Springer 2009.
8. M. Bucci e R. Luzzi. Progettazione di generatori di bit casuali testabili. In JR Rao e B. Sunar, editori, *Hardware crittografico e sistemi integrati (CHES)*, JR Rao e B. Sunar, Eds., pagine 147-156. Springer, 2005.
9. P.Chevalier et al. Generatore di numeri casuali. Brevetto USA numero 3.790.768, 5 febbraio 1974.
10. B. Chor, O. Goldreich, J. Hasted, J. Freidmann, S. Rudich e R. Smolensky. Il problema dell'estrazione dei bit oT-funzioni resilienti. *26° Simposio annuale sui fondamenti dell'informatica (FOCS)*, pagine 396-407, IEEE 1985.
11. T. Click, A. Liu e G. Kaminski. La qualità dei generatori di numeri casuali influisce in modo significativo sui risultati delle simulazioni Monte Carlo per i sistemi organici e biologici.*J. Comp. chimica.*, 32:513-524, 2011.
12. Polizia di Coddington. Test di generatori di numeri casuali utilizzando simulazioni del modello Ising. *Int. J.Mod. Fis. C*, 295303, 1996.
13. Ricerca sulla crittografia. Riepilogo della valutazione: VIA C3 Neemia generatore di numeri casuali.http://www.via.com.tw/en/downloads/whitepapers/initiatives/ padlock/ evaluation_summary_padlock_rng.pdf, 2003.

14. Cripta-X. http://roth.cs.kuleuven.be/wiki/Crypt-X.
15. RB Davies. OR esclusivo (XOR) e generatori di numeri casuali hardware.http: // www.robertnz.net/pdf/xor2.pdf. 28 febbraio 2002.
16. D. Davis, R. Ihaka e PP Fenstermacher. Casualità crittografica dalla turbolenza dell'aria nelle unità disco.*Progressi nella crittografia (Crypto)*, Y. Desmedt, Ed., pagine 114-120, Springer, 1994.
17. M. Dichtl. Come prevedere l'output di un generatore di numeri casuali hardware. *Hardware crittografico e sistemi integrati (CHES)*, CD Walter, . K. Ko¸ç e C. Paar, Eds., pagine 181188, Springer 2003.
18. M. Dichtl e JD Golic. Generazione di numeri casuali reali ad alta velocità solo con porte logiche.*Hardware crittografico e sistemi integrati (CHES)*, P. Paillier e I. Verbauwhede, Eds., pagine 45-62. Springer 2007.
19. W. Diffie e ME Hellman. Nuove direzioni nella crittografia.*IEEE Trad. Far sapere. Teoria*, 22:644-654, 1976.
20. JF Dynes, ZL Yuan, AW Sharpe e AJ Shields. Un generatore di numeri casuali quantistici ad alta velocità, senza post-elaborazione.*Appl. Fis. Lett.*, 93:031109, 2008.
21. RJ Easter e C. French. Allegato C: Generatori di numeri casuali approvati per FIPS PUB 140-2.*Requisiti di sicurezza per i moduli crittografici*. NIST, febbraio 2012.
22. ESPACENET. Ufficio europeo dei brevetti.http://www.espacenet.com.
23. AM Ferrenberg, DP Landau e YJ Wong. Simulazioni Monte Carlo: errori nascosti da generatori di numeri casuali "buoni".*Fis. Rev. Let.*, 69:33823384, 1992.
24. A. Figotin et al. Generatore di numeri casuali basato sul decadimento alfa spontaneo. Brevetto USA numero 6.745.217, 1 giugno 2004.
25. V. Fischer e F. Bernard. Veri generatori di numeri casuali negli FPGA. In Benoit Badrignans, Jean Luc Danger, Viktor Fischer, Guy Gogniat e Lionel Torres, editori, *Tendenze di sicurezza per FPGAS*, pagine 101-135. Springer, 2011.
26. M. Fürst, H. Weier, S. Nauerth, DG Marangon, C. Kurtsiefer e H. Weinfurter. Generazione di numeri casuali quantistici ottici ad alta velocità.*Optare. Esprimere*, 18:13029-13037, 2010.
27. I. Gerhardt, Q. Liu, A. Lamas-Linares, J. Skaar, C. Kurtsiefer e V. Makarov. Perfetta intercettazione su un sistema di crittografia quantistica. arXiv:1011.0105v1 [quantph], 18 marzo 2012.
28. I. Goldberg e D. Wagner. Casualità nel browser Netscape.*Diario del dottor Dobb*, gennaio 1996.
29. L. Gollub. Vorrichtung zur gewinnung von zufallszahlen. Numero di brevetto tedesco DE19743856A1, 8 aprile 1999.
30. M. Goresky e A. Klapper. *Sequenze di registro a scorrimento algebrico*. Cambridge University Press, 2012.
31. D. Gottesman, H.-K. Lo, N. Lutkenhaus e J. Preskill. Sicurezza della distribuzione delle chiavi quantistiche con dispositivi imperfetti.*Informazioni e calcoli quantistici*, 4:325-360, 2004.
32. P. Grassberger. Sulle correlazioni in buoni generatori di numeri casuali.*Fis. Lett. UN*, 181:43-46, 1993.
33. H. Guo, W. Tang, Y. Liu e W. Wei. Generazione di numeri veramente casuali basata sulla misurazione del rumore di fase di un laser.*Fis. Rev. E*, 81:051137, 2010.
34. L. Hars. Circuito elettronico per la generazione di numeri casuali. Brevetto USA US7315874(B2), 2008.
35. R. Heinen. Comunicazione privata. Università di Twente, Twente, Paesi Bassi
36. P. Hellekalek. I buoni generatori di numeri casuali sono (non così) facili da trovare.*Matematica e computer in simulazione*, 46:485-505, 1998.
37. IdQuantique. Quantis: vero generatore di numeri casuali che sfrutta i quanti tum fisica. http://www.idquantique.com/random-number-generators/products/ products-overview.html, 2012.

38. Istituto Ruder Boˇškovi´ C. QRBG 121.http://qrbg.irb.hr, 2012.
39. T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter e A. Zeilinger. Un generatore di numeri casuali quantistici veloce e compatto.*Rev. Sci. strumento*, 71:1675-1680, 2000.
40. P. Jonsson. Boom nel gioco d'azzardo su Internet in arrivo? L'inversione della politica statunitense apre la strada. http://tinyurl.com/86b9aaz, 26 dicembre 2011.
41. B. Jun e P. Kocher. Il generatore di numeri casuali di Intel. Cryptography Research Inc., Libro bianco preparato per Intel Corporation, 22 aprile 1999.
42. I. Kanter, Y. Aviad, I. Reidler, E. Cohen e M. Rosenbluh. Un generatore di bit casuale ultraveloce ottico.*Fotonica della natura*, 4(1):58-61, dicembre 2010.
43. J. Kelsey, B. Schneier, D. Wagner e C. Hall. Attacchi cyptanalytic su generatori di numeri pseudocasuali.*Crittografia software veloce*, pagine 168-188, Springer 2005. pagine 53-63, Springer, 2005.
44. T. Kim, IS Wersborg, FNC Wong e JH Shapiro. Simulazione fisica completa dell'attacco della sonda impigliante sul protocollo Bennett-Brassard 1984. *Fis. Rev. A*, 75:042327, 2007.
45. DE Knuth. Rilevamento di singoli fotoni ad alta velocità nel vicino infrarosso.*L'arte della programmazione informatica*, vol. 2, 3a edizione, Addison Wesley, 1997.
46. O. Kwon, generatore di numeri casuali YQuantum che utilizza l'entanglement del percorso del numero di fotoni. *Appl. Ottica*, 48:1774-1778, 2009.
47. P. Li, YC Wang e JZ Zhang. Generatore di numeri casuali veloce completamente ottico.*Optare. Esprimere*, 18:20360-20369, 2010.
48. P. Lacharme. Funzioni di post-elaborazione per un generatore di numeri casuali fisici distorto.*Crittografia software veloce (FSE)*, pagine 334-342, 2008.
49. P. Lacharme. Analisi e costruzione di correttori.*IEEE Trans. Teoria dell'informazione*, 55(10):4742-4748, ottobre 2009.
50. X. Li, AB Cohen, TE Murphy e R. Roy. Generatore di numeri casuali fisici paralleli scalabili basato su un LED superluminescente.*Optare. Lett.*, 36:1020-1022, 2011.
51. Autorità per le lotterie e il gioco d'azzardo. Regolamento del gioco a distanza, Avviso legale 176 del 2004, 110 del 2006, 2760 e 426 del 2007, e 90 del 2011. Malta, 2011.
52. L. Lydersen, V. Makarov e J. Skaar. Schema di rilevamento protetto con gate per la crittografia quantistica. arXiv:1101.5698 [quant-ph], 29 gennaio 2011.
53. L. Lydersen, C. Wiechers, C. Wittmann, D. Elser, J. Skaar e V. Makarov. Hacking di sistemi di crittografia quantistica commerciale mediante illuminazione brillante su misura.*Fotonica della natura* , 4:686689, 2010.
54. G. Marsaglia. Batteria DIEHARD di severi test di casualità.http://stat.fsu. edu/~geo/ diehard.html.
55. G. Marsaglia e WW Tsang. Il metodo ziggurat per la generazione di variabili casuali. *Giornale del software statistico*, 5(8): 1-7, ottobre 2000. http://www. jstatsoft.org/v05/i08.
56. JL Massey. Sintesi shift-register e decodifica BCH.*IEEE Trad. Far sapere. Teoria*, 15(1):122-127, gennaio 1969.
57. M. Matsumoto e T. Nishimura. Mersenne twister: un generatore di numeri pseudo-casuali uniformi equidistribuiti a 623 dimensioni.*Transazioni ACM su modellazione e simulazione al computer*, 8:3-30, 1998. http://www.math.sci.hiroshima-u.ac.jp/~m-mat/ MT/emt.html.
58. A. De Matteis e S. Pagnutti. Correlazioni a lungo raggio in generatori di numeri casuali lineari e non lineari.*Calcolo parallelo.*, 14(2):207210, giugno 1990.
59. UM Maurer. Un test statistico universale per generatori di bit casuali.*Giornale di Crittologia*, 5(2):89-105, 1992.
60. U. Maurer. Accordo chiave segreta mediante discussione pubblica da informazioni comuni. *IEEE Trad. Far sapere. Teoria*, 39:733-742, 1993.
61. T. McNichol. Totalmente casuale.*Cablato*, 11(8), agosto 2003. http://www.wired.com/ wired/archive/11.08/random.html.
62. JA Miszczak. Generazione e utilizzo di stati quantistici veramente casuali in Mathematica. arXiv:1102.4598v2 [quant-ph], 19 ottobre 2011.

63. J. von Neumann. Varie tecniche per l'uso in connessione con cifre casuali.*Opere raccolte di John von Neumann*, vol. 5, pagine 768770, 1963.
64. H.Nyquist. Agitazione termica della carica elettrica nei conduttori.*Fis. rev.*, 32:110-113, 1928.
65. oelermans R. Oelermans e V. Miche Circuito generatore di numeri casuali veri digitali. Domanda di brevetto USA, US2002156819 (A1), 2002.
66. G. Parisi e F. Rapuano. Effetti del generatore di numeri casuali su simulazioni al computer. *Lettere di fisica B*, 157:301-302, 1985.
67. Y. Peres. Iterazione della procedura di von Neumann per l'estrazione di bit casuali.*Anna. Statistica.* , 20:590-597, 1992.
68. PicoQuant. PQRNG 150. http://www.picoquant.com/products/pqrng150/pqrng150.htm, 2012.
69. A. Proikova. Come migliorare un generatore di numeri casuali.*Comp. Fis. Comm.*, 124:125-131, 2000.
70. B. Qi, Y.-M. Chi, H.-K. Lo e L. Qian. Generazione di numeri casuali quantistici ad alta velocità misurando il rumore di fase di un laser a modalità singola.*Optare. Lettere*, 35:312-314, 2010.
71. Qutools GmbH. quRNG.http://www.qutools.com/products/quRNG/, 2012.
72. JA Reeds e NJA Sloane. Sintesi shift-register (Modulo m).*SIAM J. Comput*, 14:505-513, 1985.
73. I. Reidler, Y. Aviad, M. Rosenbluh e I. Kanter. Generazione di numeri casuali ad altissima velocità basata su un laser a semiconduttore caotico.*Fis. Rev. Lett.*, 103(2):024102, 2009.
74. T. Ritter. Macchine a numeri casuali: un'indagine sulla letteratura. http://www.ciphersbyritter.com/RES/RNGMACH.HTM, 4 dicembre 2002.
75. RL Rivest. L'algoritmo di crittografia RC4. RSA Data Security Inc., marzo 1992.
76. F. Rodriguez-Henriquez, NA Saqib, A. Diaz-Perez e Ç. K. Ko¸ç.*Algoritmi crittografici su hardware riconfigurabile*. Springer, 2007
77. CB Roellgen. Visualizzazione della potenziale debolezza delle implementazioni del motore di cifratura esistenti nel software commerciale di cifratura del disco al volo. Global IP Telecommunications, Ltd. & PMC Ciphers, Inc., 15 agosto 2008.
78. A. Ruhkin. Test statistici di casualità: vecchie e nuove procedure.*Casualità attraverso il calcolo*, H Zenil, Ed., World Scientific, 2011.
79. A. Ruhkin et al. Una suite di test statistici per generatori di numeri casuali e pseudocasuali per applicazioni crittografiche. Pubblicazione speciale NIST 800-22rev1a, aprile 2010.
80. D. Schellekens, B. Preneel e I. Verbauwhede. Generatore di numeri casuali reali indipendente dal fornitore di FPGA.http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.86.5319, 2006.
81. W. Schindler. Criteri di valutazione per generatori fisici di numeri casuali. In . K. Ko¸ç, editore, *Ingegneria crittografica*, pagine 25-54. Springer, 2009.
82. W. Schindler. Generatori di numeri casuali per applicazioni crittografiche. In . K. Ko¸ç, editore, *Ingegneria crittografica*, pagine 5-23. Springer, 2009.
83. W. Schindler. Anwendungshinweise und Interpretationen zum Schema (AIS). AIS 32, versione 1, Bundesamt für Sicherheit in der Informationstechnik, 2001.
84. W. Schindler e W. Killmann. Criteri di valutazione per veri generatori di numeri casuali (fisici) utilizzati nelle applicazioni crittografiche.*Hardware crittografico e sistemi integrati (CHES)*, BS Kaliski Jr., . K. Ko¸ç e C. Paar, Eds., pagine 431449, Springer 2002.
85. F. Schmid e NB Wilding. Errori nelle simulazioni Monte Carlo che utilizzano generatori di numeri casuali con registro a scorrimento.*Int. J.Mod. Fis.*, 6:781787, 1995.
86. R. Shaltiel. Recenti sviluppi nelle costruzioni esplicite di estrattori.*Toro. EATCS*, 77:6795, 2002.
87. R. Shaltiel. Come ottenere più chilometri dagli estrattori di casualità.*Struttura casuale. Algoritmi*, 33:157-186, 2008.

88. P. Shor e J. Preskill. Semplice prova di sicurezza del protocollo di distribuzione della chiave quantistica BB84.*Fis. Rev. Lett.*, 85:441-444, 2000.
89. A. Sidorenko e B. Schoenmakers. Attacchi di ripristino dello stato su generatori pseudocasuali. *Workshop dell'Europa occidentale sulla ricerca in crittologia*, pagine 53-63, Springer, 2005.
90. A. Stefanov, N. Gisin, O. Guinnard, L. Guinnard e H. Zbinden. Generatore di numeri casuali quantistici ottici.*J.Mod. Optare.*, 47, 595-598, 2000.
91. M. Stipˇ čevíĆ. Apparato e metodo per generare veri bit casuali basati su integrazione temporale di una sorgente di rumore elettronico. Numero di brevetto WIPO WO03040854, 17 ottobre 2001.
92. M. Stipˇ čevíĆ. Generatore di bit casuale non deterministico veloce basato su eventi fisici debolmente correlati.*Rev. Sci. strumento*, 75:4442-4449, 2004.
93. M. Stipˇ čevíĆ. Generatore quantistico di bit casuali. Numero di brevetto WIPO WO2005106645 (A2), 30 aprile 2004.
94. M. Stipˇ čevíĆ. Prevenzione degli attacchi accecanti del rivelatore e altri attacchi del generatore di numeri casuali alla crittografia quantistica mediante l'uso di un generatore di numeri casuali esplicito. Manoscritto in preparazione, 2012.
95. M. Stipˇ čevíć e BM Rogina. Generatore di numeri casuali quantistici basato su foto emissione tonica nei semiconduttori. *Rev. Sci. strumento*, 78:1-7, 2007.
96. B. Sunar. Veri generatori di numeri casuali per la crittografia. In . K. Ko¸ç, editore, *Ingegneria crittografica*, pagine 55-73. Springer, 2009.
97. B. Sunar e D. Schellekens. Generatori di numeri casuali per circuiti integrati e FPGA. In I. Verbauwhede, editore,*Sistemi e circuiti integrati sicuri*, pagine 107-124. Springer, 2010.
98. B. Sunar, WJ Martin e DR Stinson. Un vero generatore di numeri casuali dimostrabilmente sicuro con tolleranza incorporata agli attacchi attivi.*IEEE Trans. sui computer*, 56(1):109-119, gennaio 2007.
99. S. Takagi. Generatore di dati di numeri casuali. Brevetto USA US2003208517, 2003.
100. G. Taylor e G. Cox. Dietro la nuova generazione di numeri casuali di Intel tore. *Spettro IEEE*, http://spectrum.ieee.org/computing/hardware/dietro-intel-nuovo-generatore-numero-casuale, 24 agosto 2011.
101. TE Tkacik. Un generatore di numeri casuali hardware.*Hardware crittografico e sistemi integrati (CHES)*, BS Kaliski Jr., . K. Ko¸ç, e C. Paar, Eds., pagine 450453, Springer 2002.
102. A. Uchida et. al. Generazione rapida di bit fisici casuali con laser a semiconduttore caotici. *Fotone della natura.*, 2:728-732, 2008.
103. G. Vallone, D. Marangon, M. Tomasin e P. Villoresi Generatore quantistico di numeri casuali autocalibrante basato sul principio di indeterminazione. arXiv:1401.7917 [quantph], 30 gennaio 2014.
104. I. Vattulainen, T. Ala-Nissila e K. Kankaala. Test fisici per numeri casuali nelle simulazioni. *Fis. Rev. Lett.*, 73:25132516, 1994.
105. VIA Inc. Tramite nota applicativa di sicurezza. www.via.com.tw/en/downloads/whitepaper/iniziative/lucchetto/security_application_note.pdf, 2005.
106. Crittografia AES di VIA Inc.. http://www.via.com.tw/en/initiatives/padlock/hardware.jsp, 2012.
107. VIA Inc. Generazione di numeri casuali. http://www.via.com.tw/en/initiatives/ padlock/hardware.jsp, 2012.
108. VIA Inc. Tramite motore di sicurezza con lucchetto. http://www.via.com.tw/en/initiatives/padlock/hardware.jsp, 2012.
109. J. Viega. Generazione pratica di numeri casuali nel software. In*Atti della 19a conferenza annuale sulle applicazioni di sicurezza informatica*, pagine 129-140, 2003.
110. CH Vincenzo. La generazione di numeri binari veramente casuali.*J. Fis. E: Strumenti scientifici*, 3:594-598, 1970.

111. M. Wahl, M. Leifgen, M. Berlin, T. Roehlicke, HJ Rahn e O. Benson. Un generatore di numeri casuali quantistici ultraveloce con bias di uscita dimostrabilmente limitato basato su misurazioni del tempo di arrivo dei fotoni.*Appl. Fis. Lett.*, 98:171105, 2011.
112. J. Walker. Ent: Un programma di test di sequenza numerica pseudocasuale.http://www.fourmilab.ch/random/.
113. CS Wallace. Generatore fisico casuale.*Gior. Comp. Sis. Sci. e l'ing.*, 5(2):82-88, 1990.
114. AB Wang, YC Wang e HC He. Miglioramento della larghezza di banda del segnale ottico caotico generato da un laser a semiconduttore con feedback ottico.*Fotone IEEE. tecnico. Lett.*, 20:1633-1635, 2008.
115. AB Wang, YC Wang e JF Wang. Rotta verso il caos della banda larga in un diodo laser caotico soggetto a iniezione ottica.*Optare. Lett.*, 34:1144-1146, 2009.
116. MA Wayne, ER Jeffrey, GM Akselrod e PG Kwiat. Generazione quantistica di numeri casuali del tempo di arrivo del fotone.*J.Mod. Optare.*, 56:516522, 2009.
117. MA Wayne e PG Kwiat. Generatore di numeri casuali quantistici ad alta velocità e bassa polarizzazione tramite impulsi ottici sagomati.*Optare. Esprimere*, 18:9351-9357, 2010.
118. S.-K. Yoo, D. Karakoyunlu, B. Birand e B. Sunar. Miglioramento della robustezza degli oscillatori ad anello TRNG.*Transazioni ACM su tecnologie e sistemi riconfigurabili*, 3(2):9, maggio 2010.
119. ZL Yuan, BE Kardynal, AW Sharpe e AJ Shields. Rilevamento di singoli fotoni ad alta velocità nel vicino infrarosso.*Appl. Fis. Lett.*, 91:041114, 2007.