NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

PEDRO ALEXANDRE GÂNDARA MONTEIRO

Licenciatura em Engenharia Eletrotécnica e de Computadores

GERADOR DE NÚMEROS ALEATÓRIOS INTEGRADO EM TECNOLOGIA CMOS

MESTRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

Universidade NOVA de Lisboa
março, 2022

NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

GERADOR DE NÚMEROS ALEATÓRIOS INTEGRADO EM TECNOLOGIA CMOS

PEDRO ALEXANDRE GÂNDARA MONTEIRO

Licenciatura em Engenharia Eletrotécnica e de Computadores

Orientador: Doutor Luis Augusto Bica Gomes de Oliveira
Professor Associado com Agregação, NOVA University Lisbon

Júri:

Presidente: Doutor André Teixeira Bento Damas Mora
Professor Auxiliar, NOVA University Lisbon

Arguente: Doutor João Carlos Ferreira de Almeida Casaleiro
Professor Adjunto, Instituto Superior de Engenharia de Lisboa

MESTRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

Universidade NOVA de Lisboa
março, 2022

Gerador de números aleatórios integrado em tecnologia CMOS

Às minhas avós.

Agradecimentos

Ao meu orientador Professor Doutor Luís Oliveira pela sua disponibilidade, não só durante esta dissertação, como por todo o percurso académico, alimentando sempre o meu interesse por eletrónica através do seu incansável entusiasmo.

Também, à Universidade Nova de Lisboa e ao Departamento de Engenharia Eletrotécnica por todos os anos de ensino que me foram proporcionados, com todo o rigor e excelência. Ainda, ao projeto ROBUST: EXPL/EEI-EEE/0776/2021 pelo apoio no desenvolvimento desta dissertação.

Aos meus amigos, pelos anos fantásticos que me proporcionaram, com o apoio e ajuda incondicional, tornando este tempo muito mais agradável. Um especial agradecimento aos meus colegas Diogo Emílio e Moisés Catumbira, que sempre me puxaram para cima e me apoiaram de braços abertos sempre que precisei. Ao amor da minha vida, Diana Monteiro, pelo carinho incondicional que me deu, apoio e um abraço sempre que precisei, tornando esta etapa da minha vida memorável, onde crescemos a nível académico e emocional juntos.

À minha madrinha Alice Fernandes e ao meu padrinho Ernesto Fernandes, por serem uns pilares na minha vida e me terem ajudado a tornar-me no que sou hoje, atribuindo sempre uma parte do meu sucesso ao apoio deles.

E por último, aos meus pais, António Monteiro e Célia Monteiro, pela oportunidade de poder frequentar o ensino superior, por nunca desistiram de mim, estando sempre ao meu lado com todo o carinho e preocupação, ajudando a ultrapassar mais uma etapa desafiante.

“Hope for the Best. Expect the worst.” (Mel Brooks)

Resumo

Desde os primórdios da civilização humana, foram inventadas inúmeras formas de comunicação, surgindo, assim, a necessidade de tornar essas formas de comunicação privadas. Desta forma, considera-se que a criptografia existe desde então.

No entanto, com o início da era digital, a quantidade de informação transmitida aumentou exponencialmente. Consequentemente, a forma como a privacidade das comunicações é mantida deixa de ser a única questão abordada, levando-nos à seguinte problemática:

"Como proteger um elevado número de mensagens sensíveis de forma sistemática?"

A solução para esta questão são os Geradores de Números Aleatórios, RNG. Estes sistemas têm a capacidade de gerar chaves que, ao misturar as mensagens, conseguem escondê-las de forma rápida e simples.

Existem duas categorias de geradores de números aleatórios: os verdadeiramente aleatórios e os pseudoaleatórios.

Pretende-se estudar uma fonte de entropia baseada no ruído do oscilador e, para atingir este objetivo, propôs-se um circuito gerador de números aleatórios que disponha de um consumo, custo e área reduzidos e uma elevada aleatoriedade. Através do circuito proposto na presente dissertação, um gerador de números aleatórios híbrido - circuito composto por osciladores e um circuito caótico - os objetivos relativos à área e ao consumo de potência foram cumpridos, tendo o circuito 1,19 mW de potência consumida, 34,5μm^2 de área de transístores e um *throughput* de 26 Mbit/s. No entanto, não foram reunidas as condições necessárias para se testar estatisticamente o circuito quanto à sua aleatoriedade, sendo que, teoricamente, o sistema apresentado deverá comportar-se como um TRNG.

Palavras-chave: Criptografia, Aleatoriedade, Oscilador, Caótico, PRNG, TRNG, bit, RNG,

Abstract

From the beginning of human civilization, several means of communication were invented and, there was a surge in the need to make the communication private, thus it is considered that cryptography exists since then.

Nonetheless, with the beginning of the digital era, the amount of shared information exponentially grew. Consequently, the means of effectively hide the information is not the only concern, due to the amount of information, which brings a very important question:

"How can we systematically hide large amounts of information?"

The solution to this question is random number generators (RNG). These systems have the capacity to generate cryptographic keys which, when mixed with the information, hide it in an efficient and timely manner. There is two categories of RNG, being truly random (TRNG) or pseudorandom (PRNG).

The objective was to study the entropy source based on the noise of an oscillator, and to achieve that, a RNG circuit was designed to have a low power consumption, a high randomness and a low cost and area usage. The chosen architecture for this dissertation is a hybrid RNG, which uses oscillators and a chaotic circuit to generate the random bits.

With the simulation of the circuit, it was found to be at the objectives mark, having 1,19 mW of power, 34,5μm^2 of area of transistors and a throughput of 26 Mbit/s. However, due to limitations with the simulation, it wasn't possible to run all the statistical tests, although all the run testes were passed.

Keywords: RNG, TRNG, PRNG, statistical, bits, chaotic, information, cryptography,

Índice

Índice de Figuras

Índice de Tabelas

Siglas

AC Alternating Current 14, 15
ASIC Application Specific Integrated Circuit 19

CMOS Complementary Metal Oxide Semiconductor 5, 17, 19, 56, 57
CRP Challenge Response Pair 19

DC Direct Current 15

FPGA Field-Programmable Gate Array 19, 25, 30, 35, 57
FRO Free Running Oscillator xi, 25, 30

IC Integrated circuit 4
IoT Internet of Things 1, 4

KCL Kirchhoff's current law 11

MOS Metal Oxide Semiconductor 17

NMOS N-type Metal Oxide Semiconductor 17

PMOS P-type Metal Oxide Semiconductor 17
PRNG Pseudo Random Number Generator ix, 3, 11, 12, 22, 51
PUF Physically Unclonable Function 3, 4, 5, 19, 20, 57, 58

RNG Random Number Generator 2, 5, 11, 13, 22, 24, 27, 28, 30, 31, 36, 37, 38, 41, 50, 51, 53, 54, 55

TRNG True Random Number Generator ix, 3, 4, 11, 12, 22, 52, 57, 58

1

Introdução

1.1 Motivação

Desde os primórdios da civilização que têm vindo a ser desenvolvidas inúmeras formas de comunicação entre dois ou mais interlocutores. Com a elaboração dessas mesmas formas de comunicação, surgiu a necessidade de arranjar maneiras de encapsular a informação partilhada, de modo que o seu conteúdo permaneça, apenas, do conhecimento das partes interessadas, e garantindo, também, a integridade do mesmo. Nasce, assim, o que hoje denominamos de criptografia.

Com o aparecimento da *Internet*, a dependência da sociedade em sistemas tecnológicos aumentou gradualmente, sendo que, atualmente, os sistemas Internet of Things (IoT) estão presentes em praticamente todos os equipamentos tecnológicos usados no quotidiano. Isto porque gerar, aceder, trocar e armazenar dados, relativos à vida pessoal, negócios ou comércio, tornou-se bastante simples e rápido. Desta forma, o nível de conectividade e partilha de informação entre as pessoas aumentou, havendo, portanto, a necessidade de criação de diferentes sistemas criptográficos, uma vez que se levanta a questão de quão segura é esta constante e rápida troca de informação.

Apesar de, ao início, os sistemas criptográficos serem, essencialmente, de domínio governamental e militar, a necessidade de tornar a partilha de dados segura levou ao aumento e uso generalizado destes sistemas. Desta forma, a criptografia começou a ser utilizada na maioria dos equipamentos usados no dia-a-dia, como, por exemplo, os telemóveis.

Para além das possíveis lacunas, ao nível da segurança, associadas aos sistemas IoT, provenientes da rápida e constante troca de informação, por parte de um elevado número de utilizadores, as atuais preocupações relativas à pegada ecológica dos dispositivos que recorrem a estes sistemas, obrigam a que ocorra um racionamento no consumo de energia por partes dos mesmos. Desta forma, a robustez dos sistemas criptográficos torna-se limitada, estando subjacente um aumento significativo dos ciberataques físicos com a expansão dos sistemas IoT. Assim, o objetivo principal, aquando do desenvolvimento

destes sistemas, consiste na projeção de circuitos com elevada robustez, mas com um reduzido consumo de energia [1].

A criptologia, cuja definição é a ciência que estuda a escrita de segredos, pode ser separada em dois campos distintos: um campo responsável pelo estudo da criação e desenvolvimento de sistemas criptológicos, denominado criptografia, e outro, responsável pelo estudo de técnicas para decifrar conteúdo encriptado, denominada criptoanálise, [2], [3], [4].

A criptografia trata, de uma forma prática, de esconder mensagens antes da sua transmissão, através de um meio que pode estar comprometido, com garantias de integridade e sigilo, [3]. A implementação da criptografia consegue ser explicada de forma simples: a mensagem a encriptar é transformada numa nova, através de um método criptográfico, que é parametrizado por uma chave que deve ser única. A mensagem é desencriptada pelo processo inverso, através do uso da chave única, mencionada anteriormente. Na figura 1.1 encontra-se a representação desta implementação.

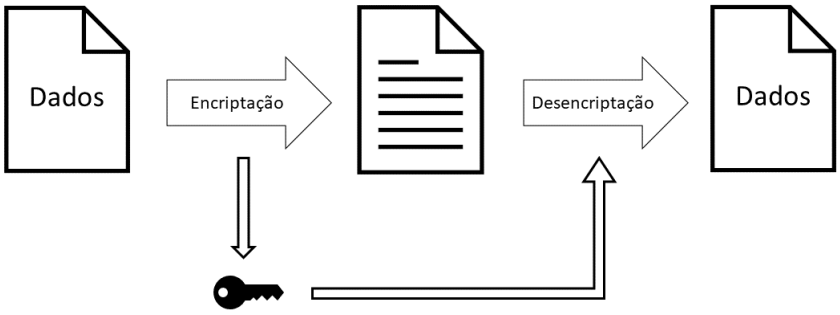

Figura 1.1: Esquema exemplificativo da implementação de Criptografia [4], [5]

Para além do estudo e desenvolvimento de novos métodos criptográficos, é, também, necessário investigar diferentes métodos de criação de chaves criptográficas, ou seja, formas de gerar números aleatórios, Random Number Generator (RNG).

Para uma melhor compreensão das distintas formas de se gerar números aleatórios, proceder-se-á à sua categorização. Desta forma, existem dois tipos de números aleatórios: os números verdadeiramente aleatórios e os pseudo-aleatórios. Um número verdadeiramente aleatório é aquele em que, através de condições iniciais idênticas, é impossível chegar a um mesmo resultado - o número gerado - seguindo um determinado padrão, sendo, por conseguinte, impossível prever o número seguinte. Já um número pseudo-aleatório é aquele que, parecendo aleatório, usando as condições iniciais que lhe deram

origem, volta a gerar o mesmo número.

Assim, também existem dois tipos de geradores de números aleatórios, os geradores pseudo aleatórios e os geradores verdadeiramente aleatórios, [6]. Relativamente aos primeiros geradores enumerados, estes são os mais convencionais dada a sua fácil implementação, podendo ser criados informaticamente, tendo por base algoritmos matemáticos. Porquanto, uma vez criados com base em leis determinísticas, o cálculo da chave pseudo aleatória é complexo, no entanto possível, uma vez conhecido o algoritmo que lhe deu origem.

Uma vez criptoanalisadas todas as principais famílias de PRNGs, o uso desta categoria de geradores revelou-se um risco iminente, tornando-os obsoletos. Não só conhecidos os algoritmos que estão na base das principais famílias de PRNGs, os cálculos complexos que lhes são inerentes, tornaram-se simples de efetuar com a introdução dos computadores, dada a sua elevada capacidade de computação, sendo capazes de resolver milhares de milhões de cálculos por segundo.

Tendo em conta o anteriormente descrito, tornou-se imperativo aumentar o poder de processamento dos sistemas criptográficos para, assim, serem mais seguros. Desta forma, ocorreu uma alteração do paradigma, o que levou ao abandono dos algoritmos complexos, que estão na base dos geradores de números pseudo aleatórios, e na aposta de algoritmos que gerassem números verdadeiramente aleatórios. Dada a sua natureza aleatória, o cálculo das chaves obtidas por estes geradores é impossível, mitigando o problema da segurança na partilha de dados confidenciais entre dois interlocutores.

Apesar das suas mais valias, o desenvolvimento de TRNGs é desafiante, dado a difícil implementação dos mesmos em contexto de limitação de recursos.

Os geradores de números verdadeiramente aleatórios, TRNG, recorrem à entropia dos sistemas para criar aleatoriedade, isto é, estes geradores têm por base circuitos físicos, sendo que, por exemplo, a aleatoriedade pode ser alcançada através do ruído provocado no sinal pelo movimento das cargas nos circuitos. O desafio da criação deste tipo de geradores passa por não influenciar a produção dos números aleatórios através da projeção do circuito, tornando, desta forma, a chave gerada mais segura. Consequentemente, a implementação dos TRNGs torna-se mais complexa.

Existem quatro tipos de geradores de números verdadeiramente aleatórios, sendo estes: baseados em amplificação de ruído, baseados em osciladores, com circuitos caóticos e com partículas quânticas.

Para além dos TRNGs, existe outra categoria de circuitos cuja funcionalidade passa pela criação de chaves aleatórias, sendo estes os Physically Unclonable Function (PUF)s. No entanto, apesar de ambos os circuitos gerarem sequências de *bits* aleatórias, o seu funcionamento difere, isto é, os TRNGs são sempre aleatórios, criando, constantemente, durante o seu funcionamento novas chaves, ao contrário dos PUFs que criam, apenas, uma chave aleatória, mantendo-a durante toda a sua operabilidade.

Assim, os resultados aleatórios mencionados anteriormente, para além de serem utilizados em criptografia, podem, ainda, ser utilizados para proteger a identidade dos circuitos de silício. A ameaça a esta identidade única tem sido uma atual problemática, devido à expansão do mercado de circuitos integrados - Integrated circuit (IC)s - contrafeitos, surgindo graves problemas associados. Uma vez que os fabricantes de circuitos eletrónicos não produzem, de forma geral, todos os ICs comercializados nas suas instalações, comprando-os a granel a outros fabricantes, a verificação da autenticidade de todos os ICs apresenta-se como uma tarefa de elevada dificuldade, podendo estes não ser legítimos e funcionar de acordo com as suas especificações. Assim, nasce o mercado para os PUFs, que, como explicado anteriormente, criam uma chave única e invariável.

Relativamente aos equipamentos IoT, a principal fonte de entropia recorre à autenticação mútua de PUFs e TRNGs. As PUFs geram uma identificação única estável, através das variações físicas introduzidas no fabrico de circuitos integrados, enquanto que os TRNGs criam um fluxo de *bits* estocástico e imparcial, que não tem correlação com os recursos do circuito.

Para que estes equipamentos possam operar durante vários anos em baterias ou através de produção própria de energia, têm de apresentar uma baixa potência e um baixo custo e é imperativo que sejam seguros. Devido aos limites impostos, a sua implementação torna-se, como referido anteriormente, bastante complexa.

1.2 Objetivos

Na presente dissertação propõe-se o desenvolvimento do projeto de um circuito gerador de *bits* aleatórios, partindo do uso de osciladores em anel, com células de atraso diferenciais como *clock* de um *flip-flop* tipo D e um oscilador em anel com uma célula de atraso *single-ended* ligada a um circuito gerador de ruído caótico como dados no flip-flop. As metas do projeto do circuito são:

1. Baixo consumo de potência
2. Reduzida área de circuito
3. Frequência de *clock* entre 20 MHz e 30 MHz
4. Estatisticamente aleatório

Estes requisitos foram definidos de forma a que o circuito seja portátil e possa ser integrado em sistemas que necessitem de um gerador de números aleatórios sem que seja necessário aumentar o tamanho tanto da bateria como do equipamento. Para isso, é imperativo ter a eficiência de potência e de área elevadas, sendo estes dois objetivos importantíssimos.

A frequência de amostragem entre os 20 MHz e 30 MHz é necessária de forma a que o circuito possa ser projetado em Complementary Metal Oxide Semiconductor (CMOS), não necessitando de circuitos auxiliares de radiofrequência e maximizando o ruído, fazendo com que o circuito fique com um preço inferior, sendo economicamente mais apelativo.

Para que o circuito se comporte como um gerador de números aleatórios é necessário que seja estatisticamente aleatório, podendo o seu comportamento ser certificado através de ferramentas de teste, como por exemplo, o NIST 800-22.

1.3 Organização

A presente dissertação encontra-se organizada em seis capítulos, a saber:

- no capítulo 1, encontra-se descrita a motivação para a criação de um Gerador de Números Aleatórios, sendo esta a criptografia e a sua utilidade na atualidade. É apresentada, também, uma breve introdução aos geradores de números aleatórios;
- no capítulo 2 são apresentadas algumas noções teóricas necessárias para uma melhor compreensão do trabalho desenvolvido, tais como o conceito de geradores de números aleatórios, de aleatoriedade, de oscilador, de ruído e de PUF;
- no capítulo 3, é elaborada uma revisão do estado de arte, estando presentes os trabalhos realizados no âmbito da temática Geradores de Números Aleatórios, RNG, que se consideram de maior relevância para a dissertação. São, também, realçadas as diferenças entre os trabalhos selecionados, bem como as suas aplicações práticas;
- no capítulo 4, encontra-se uma descrição do trabalho desenvolvido no contexto desta dissertação, explicando cada constituinte do gerador de números aleatórios desenvolvido;
- no capítulo 5 são apresentados os resultados das simulações e comparações com os trabalhos analisados e apresentados no capítulo anterior, com o resultado dos testes de aleatoriedade;
- por fim, no capítulo 6, são apresentadas as conclusões finais do projeto desenvolvido, bem como as considerações finais para a desenvolução do projeto concebido na dissertação.

2

Enquadramento

2.1 Ruído

O termo ruído é apresentado para descrever diferentes fenómenos. Uma das conotações que esse termo pode ter corresponde às modificações indesejadas que os sinais sofrem, sendo que essas podem ocorrer durante a aquisição, o armazenamento, a transmissão, o processamento e a conversão desses mesmos sinais [7]. Assim, o ruído determinístico pode ser categorizado das seguintes formas: interferência entre fios adjacentes, radiação eletromagnética num caminho sensível do sinal, ruído proveniente dos diferentes níveis do substrato e a sincronização de portas lógicas [8]. A outra designação prende-se à aleatoriedade e imprevisibilidade de alguns sinais [7], podendo este ser categorizado de três formas diferentes: ruído térmico ou de *Johnson*, ruído de *shot* e ruído de *flicker* [8].

Na presente dissertação, o termo ruído será empregue para descrever sinais aleatórios, dado que são esses sinais que apresentam relevância para o trabalho desenvolvido.

As fontes de ruído podem ser externas ao sistema, como, por exemplo, o ruído atmosférico ou o ruído galáctico, ou internas ao sistema, onde se incluem as flutuações espontâneas de corrente e tensão que ocorrem nos circuitos elétricos, como, por exemplo, o ruído térmico ou o ruído *shot* [4].

Independentemente do fenómeno inerente ao termo ou da fonte que o origina, o ruído altera consideravelmente o espetro de frequência e o intervalo de tempo de transição [8]

2.1.1 Ruído Térmico ou de *Johnson*

O ruído térmico, ou ruído de *Johnson*, é gerado a partir da agitação térmica das cargas presentes nos transístores, que induzem a um movimento ordenado das mesmas, ou seja, uma pequena corrente elétrica. Essa corrente elétrica, ao circular pelos transístores, cria uma pequena tensão elétrica, que, somada às tensões de saída, dá origem a perturbações no sinal de saída dos transístores.

A equação 2.1, apresentada por Perepelitsa em [9], define esta categoria de ruído:

$$V^2 = 4RkT \int_0^\infty \left[\frac{[g(f)^2]}{1 + (2\pi f CR)^2} \right] df \tag{2.1}$$

onde R é a resistência que a corrente sofre, k a constante de *Boltzmann*, T a temperatura do sistema, C a capacidade e f a frequência de trabalho do sistema. Nesta equação encontra-se, também, o parâmetro $g(f)$ que representa a amplificação que o circuito provoca no sinal de entrada mediante a frequência de trabalho.

Como seria de prever, pela equação 2.1 verifica-se que a temperatura, T, corresponde ao fator com maior impacto na criação de ruído de *Johnson*.

Este tipo de ruído diferencia-se do *jitter* porque afetam diferentes parâmetros dos osciladores. O ruído de Johnson, como se observa na equação 2.1, afeta a amplitude do sinal, enquanto que o *jitter* afeta a frequência ou o período.

2.1.2 Ruído *Shot*

O ruído *shot*, ou ruído de corrente, está presente em componentes eletrónicos, tais como díodos e transístores, dada a natureza discreta do fluxo de eletrões nos mesmos [4]. Segundo Perepelitsa, o ruído *shot* corresponde à quantificação das cargas em movimento, ou seja, esta categoria de ruído é devida ao movimento de cada eletrão e da sua carga, uma vez que estes provocam uma corrente muito reduzida, que perturba o sinal de forma aleatória [9].

Segundo o autor Simon Haykin, [4], num circuito fotodetetor, a corrente, mencionada anteriormente, é gerada sempre que um eletrão é emitido a partir de um cátodo. Estes eletrões são emitidos de forma natural e aleatória, estando estes tempos de emissão definidos por τ_k, onde $-\infty < k < +\infty$. Desta forma, a corrente total que circula no fotodetetor pode ser descrita através da soma infinita de todas as correntes pulsadas, isto é:

$$X(t) = \sum_{-\infty}^{+\infty} h(t - \tau_k) \tag{2.2}$$

sendo $h(t - \tau_k)$ a corrente pulsada gerada no em $t = \tau_k$.

O processo $X(t)$, definido pela equação 2.2, representa o ruído *shot*.

De acordo com Perepelitsa [9], a equação 2.3 caracteriza este tipo de ruído:

$$V_0^2 = 2eI_{av}R_F^2 \int_0^\infty [g(f)]^2 \, df + V_A^2 \tag{2.3}$$

onde I_{avg} corresponde à corrente média provocada pelo movimento dos eletrões, e reflete a quantidade de eletrões, R_F corresponde ao valor da resistência por onde os eletrões passam, $g(f)$ diz respeito ao efeito de amplificação do sistema e V_A representa a soma das outras categorias de ruído.

Esta categoria de ruído é independente da temperatura e da resistência, sendo dependente da corrente, como se pode aferir pela análise da equação 2.3, o que não é verificado no ruído de Johnson, dado que o valor de corrente, neste caso, se mantém constante, partindo da omissão que a temperatura e a resistência do sistema estejam, também eles, constantes [9].

2.1.3 *Jitter*

Define-se o *jitter* como sendo a variação no período de um sinal em relação ao seu valor nominal, pelo que estas variações podem ser observadas na fase, no período, na largura e no *duty cicle* do mesmo. Assim, o *jitter* difere das outras formas de ruído, uma vez que este último consiste na variação da amplitude de um sinal em relação ao seu valor nominal.

Considera-se que o ruído eletrónico dos dispositivos gera *jitter*. Consequentemente, será feito um enquadramento desta categoria de ruído nos osciladores, dado serem o enfoque da presente dissertação.

O evento temporal correspondente à transição entre 0 e 1, num oscilador, e é definido por t_n, sendo que, idealmente, a diferença entre dois eventos temporais conseguintes é constante - $T_n = t_{n+1} - t_n = k$. No entanto, para osciladores reais, o facto enumerado não se verifica. Isto é, o valor de T_n varia com n devido ao ruído eletrónico presente no oscilador, resultando num desvio dado pela equação 2.4:

$$\Delta T_n = T_n - \overline{T} \tag{2.4}$$

sendo T_n o período e o $\overline{T}$ o período mediano.

A esta incerteza dá-se o nome de *jitter* do *clock*, sendo que o seu valor sofre incrementos em função do intervalo de medição ΔT [10], [11].

A variabilidade acumulada (ou *jitter* acumulado) ocorre quando uma incerteza, numa transição anterior, afeta todas as transições seguintes e o seu efeito persiste indefinidamente. A estatística do *jitter* de tempo depende das correlações das fontes de ruído envolvidas; então, a incerteza de tempo quando ΔT segundos passaram é a soma das incertezas associada a cada transição [10].

Mediante as suas características, o *jitter* pode ser classificado como: *jitter* de longo curso, *jitter* de ciclo para ciclo e *jitter* de ciclo a ciclo [11].

O *jitter* de longo curso, ou absoluto, é dado pela equação 2.5

$$\Delta T_{abs} = \sum n = 1 N \Delta T_n \tag{2.5}$$

Sendo este valor dependente de n, esta categoria de *jitter* é temporalmente divergente e, por isso, o seu uso não é indicado para a avaliação de osciladores [11].

O *jitter* de ciclo descreve a magnitude das variações de período, não fornecendo nenhuma informação sobre a dinâmica do sistema. Esta categoria de *jitter* é definida pelo valor quadrático médio do erro temporal, ΔT_n, dado pela equação 2.6:

$$\Delta T_c = \lim_{n \to +\infty} \sqrt{\frac{1}{N} \sum n = 1 N \Delta T_N^2} \tag{2.6}$$

Desta forma, para proceder à avaliação dos osciladores, deve-se recorrer a esta categoria [11].

O *jitter* de ciclo a ciclo é definido pelo valor quadrático médio entre dois períodos consecutivos e determinado pela equação 2.7, [11]:

$$\Delta T_{cc} = \lim_{n \to +\infty} \sqrt{\frac{1}{N} \sum_{n=1}^{N} (t_{n+1} - T_n)^2} \tag{2.7}$$

Salienta-se que o *jitter* de ciclo para ciclo compara o período de oscilação com o período mediano, enquanto que o *jitter* de ciclo a ciclo compara o período de oscilação com o período de oscilação anterior.

2.2 Sistemas Caóticos

Os sistemas caóticos são considerados sistemas complexos regidos por leis determinísticas. No entanto, atendendo que estes sistemas são bastante sensíveis às condições iniciais, tornam-se imprevisíveis a médio longo prazo. Este atributo advém do facto de, tendo em conta que qualquer medida está sujeita a erros, pequenas diferenças nas condições iniciais induzem a resultados amplamente divergentes nestes sistemas [12].

Desta forma, os sistemas caóticos têm um comportamento, à primeira vista, semelhante ao dos fenómenos estocásticos - ruído -, uma vez que os sinais que lhes dão origem têm uma aparência aleatória. Contudo, os fenómenos estocásticos têm um número infinito de graus de liberdade, contrariamente aos sistemas caóticos, cujo número é finito e, habitualmente, pequeno [12].

Para que se possa analisar o comportamento de um sistema caótico e, assim, distinguilo de um fenómeno estocástico, recorre-se ao espaço de fases. Se estes sistemas possuírem n variáveis, vistas como graus de liberdade, então um estado desse sistema pode ser definido por um ponto num espaço n-dimensional, cujas componentes do mesmo correspondem aos valores das n variáveis, num determinado instante [12].

Os valores tomados pelos fenómenos estocásticos ou pelos sistemas caóticos apresentam-se como sendo outra característica diferenciadora de ambos. Isto é, as variáveis dos sistemas caóticos não tomam todos os valores possíveis, sendo que os fenómenos estocásticos não possuem qualquer restrição em relação aos seus valores. Assim, uma vez que as variáveis dos sistemas caóticos não tomam todos os valores possíveis, as trajetórias apresentadas no espaço de fases ocupam uma região delimitada do mesmo. O objeto criado pela sobreposição de todas as trajetórias tomadas pelo sistema denomina-se de atrator, sendo que o número de variáveis independentes que descrevem corretamente o sistema corresponde ao menor número inteiro conseguinte do tamanho do atrator [12].

Apesar da sua imprevisibilidade a médio e longo prazo, estes sistemas possuem uma auto-organização, isto é, a evolução do sistema depende dos seus estados anteriores [12].

Outra particularidade destes sistemas reside na existência de pontos de evolução do sistema - pontos críticos - onde a sua dinâmica se altera, passando a exibir um padrão de evolução distinto. Estes pontos críticos levam à existência de bifurcações na trajetória evolutiva do sistema [12].

Assim, os sistemas caóticos são caracterizados pelas seguintes propriedades:

1. são regulados por leis determinísticas;
2. apresentam auto-organização;
3. apresentam uma enorme sensibilidade às condições iniciais;
4. são imprevisíveis a médio e longo prazo;
5. têm implícitas equações diferenciais;
6. as variáveis que os descrevem não tomam todos os valores possíveis;
7. apresentam pontos críticos no seu comportamento.

Apesar de os sistemas caóticos serem caracterizados por ruído determinístico, estes servem um propósito importante nos geradores de número aleatórios, nomeadamente na amplificação de ruído dos osciladores, por exemplo. Isto é, no caso de um sistema possuir ruído com uma base aleatória, mas com um impacto reduzido, ao adicionar-se ruído determinístico, que se apresente com maior impacto, aumentará a amplitude de ruído do sistema, não modificando o seu grau de aleatoriedade.

Um exemplo físico da implementação de ruído determinístico encontra-se esquematizado na figura 5.6, [13], [14]. Associando este circuito a dois osciladores em anel, o sinal ruidoso dos osciladores será amplificado.

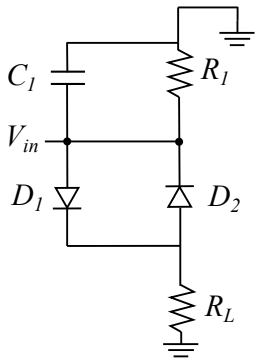

Figura 2.1: Esquemático de um gerador de sinal caótico ([13], [14]), sendo este circuito composto por dois díodos, duas resistências e um condensador.

Aplicando a Kirchhoff's current law (KCL) ao circuito previamente descrito, pode-se caracterizar, matematicamente, este sistema através das equações presentes em 2.8, [15].

$$\begin{cases} \frac{d_{v_{a1}}}{dt} = -\frac{1}{RC}v_{a1} - \frac{G_m}{C}v_{a3} \\ \frac{d_{v_{a2}}}{dt} = -\frac{1}{RC}v_{a2} - \frac{G_m}{C}v_{a1} \\ \frac{d_{v_{a3}}}{dt} = -\frac{(R+R_1)}{(C+C_1)*R*R_1}v_{a3} - \frac{G_m}{C+C_1}v_{a2} - \frac{i_d}{C+C_1} \\ \frac{d_{v_{b1}}}{dt} = -\frac{1}{RC}v_{a1} - \frac{G_m}{C}v_{a3} \\ \frac{d_{v_{b2}}}{dt} = -\frac{1}{RC}v_{a1} - \frac{G_m}{C}v_{a3} \\ \frac{d_{v_{b3}}}{dt} = -\frac{1}{RC}v_{a1} - \frac{G_m}{C}v_{a3} \end{cases} \tag{2.8}$$

2.3 Geradores de Números Aleatórios

Os geradores de números aleatórios são algoritmos que geram um sinal aleatório à saída, correspondendo esse sinal a uma chave criptográfica. Os RNG são categorizados em dois tipos: os TRNG - geradores de números verdadeiramente aleatórios - e PRNG - geradores de números pseudo aleatórios.

2.3.1 Geradores de Números Pseudo Aleatórios (PRNG)

Define-se gerador de números pseudo aleatórios como sendo uma fórmula matemática que produz uma sequência periódica e determinística de números. Essa sequência de *bits* é estabelecida por determinadas condições iniciais, como, por exemplo, o minuto e o

segundo correspondentes ao instante em que o *bit* foi gerado, sendo estas condições-chave denominadas de semente [6]. Desta forma, a sequência numeral gerada é aparentemente aleatória, deixando de o ser aquando da descoberta das condições-chave que lhe deram origem.

A aleatoriedade dos *bits* gerados é tanto maior quanto maior a complexidade do algoritmo que lhes deu origem.

Estes geradores apresentam um balanço perfeito entre 0 e 1 mas também correlações longínquas [6], que, aliados ao facto de o número ser gerado a partir de condições específicas, podendo estas ser determinadas, diminuem a segurança dos sistemas criptográficos, tornando-os suscetíveis a ataques.

Um exemplo de PRNG é apresentado no artigo [16], de 1969, em que Lewis et al. utilizaram os registos de 32 *bits* de um *IBM SYSTEM/360* para gerar números aleatórios, isto é, o sistema recebe os dados presentes nesses registos e aplica um algoritmo por forma a criar *bits* aleatórios.

A criação de PRNG tornou-se mais simples aquando da utilização em massa de computadores. Consequentemente, todas as categorias notáveis de PRNG encontram-se, neste momento, criptoanalisadas. Este facto contribui, igualmente, para a insegurança deste tipo de geradores criptográficos [6].

Apesar das suas desvantagens, este tipo de geradores de números aleatórios é vastamente utilizado devido ao seu baixo custo, facilidade de implementação e facilidade de utilização, principalmente quando operados em computadores [6].

2.3.2 Geradores de Números Verdadeiramente Aleatórios (TRNG)

Os TRNG, tal como o nome indica, geram os seus números a partir de fontes aleatórias e, consequentemente, produzem sinais totalmente aleatórios. Essas fontes de aleatoriedade, correspondem a processos físicos, que podem ser, por exemplo, o ruído térmico, o ruído de Zener, o decaimento radioativo, o tempo de chegada de fotões ou a seleção do caminho que um fotão percorre [6].

Ao contrário dos PRNG, estes geradores não apresentam um balanço perfeito entre 0s e 1s, sendo que a diferença do número de vezes que os dois *bits* são repetidos é dada pela equação 2.9:

$$b = \frac{p(1) - p(0)}{2} \tag{2.9}$$

A *bit rate* destes geradores encontra-se entre os 4 e os 150 megabits por segundo [6].

Devido à dificuldade inerente à projeção de fontes de ruído aleatórias que, partindo do princípio que são efetivamente aleatórias, a sua previsão torna-se impossível e a sua simulação é pouco fidedigna, a arquitetura dos geradores de número verdadeiramente

aleatórios torna-se muito mais difícil. Desta forma, a sua avaliação ou previsão é feita, apenas, depois da sua produção. Assim, dado as chaves geradas serem completamente aleatórias, mesmo conhecendo a arquitetura do RNG - esquemático ou algoritmo - este tipo de geradores são os melhores candidatos para uso criptográfico, uma vez ser impossível prever a sequência de *bits* gerada [6].

Existem quatro categorias de geradores de números verdadeiramente aleatórios: geradores de ruído de Johnson, geradores baseados em caos, *free running oscillators* e osciladores de ruído quântico [6].

Para um melhor enquadramento do trabalho proceder-se-á à explicação da categoria de geradores de ruído de Johnson. Apesar de se recorrer a geradores baseados em caos, para a elaboração do circuito proposto, esta categoria não será descrita nesta subsecção dado não se considerar, como referido na secção 2.2, que o ruído caótico apresente algum nível de aleatoriedade.

2.3.2.1 Gerador de Ruído de Johnson

Os geradores de ruído de Johnson, também conhecidos como geradores de ruído térmico, tal como o nome indica, utilizam o ruído térmico (ou $1/f^2$) para gerar o sinal aleatório. Este ruído é mais proeminente em componentes que apresentam alguma resistência à passagem de corrente, como, por exemplo, resistências e transístores, isto porque, estes componentes ao oferecerem uma maior resistência à passagem de corrente elétrica, induzem uma maior vibração nos eletrões, aumentando, assim, a temperatura. É a vibração dos eletrões que provoca uma distorção na corrente, apresentando-se, desta forma, ruidosa.

Os circuitos destes geradores são compostos por uma fonte de ruído, um amplificador, um oscilador controlado por tensão e um comparador. A fonte de ruído é onde se gera a aleatoriedade, o amplificador serve para que o ruído tenha um impacto significativo, o oscilador gera o sinal ruidoso e o comparador para se retirar os *bits*.

Estes geradores destacam-se por fazerem uma amostragem de um sinal ruidoso em amplitude, ou seja, o *jitter* é desprezável, visto a amostragem ser feita pela amplitude, através do comparador.

2.4 Osciladores

Primeiramente, introduzindo o conceito físico de osciladores, considera-se que uma oscilação é forçada quando, ao sistema, se encontra aplicada uma força motriz periódica, sendo que a oscilação do mesmo tem, sempre, uma frequência coincidente com a frequência

da força aplicada, excetuando no transitório inicial. Relativamente à frequência mencionada, esta característica tem um efeito dramático na amplitude do sinal, sendo que, no caso de a força motriz ser aplicada à frequência de ressonância, ou seja, a frequência da força aplicada ser coincidente com a frequência natural do sistema, pode induzir efeitos catastróficos no sistema, levando à sua rutura [17].

Aplicando a descrição efetuada anteriormente a circuitos eletrónicos, e considerando este como sendo um circuito ressonante, composto pelos componentes bobine e condensador, ligados em série e alimentados por uma tensão Alternating Current (AC), dada por $V(t) = V_0 \cos(\omega t)$, a referida tensão corresponde à energia adicionada para o arranque do sistema, sendo que a oscilação é mantida pela troca de cargas entre os componentes reativos, ultrapassando, assim, as perdas que induzem ao cessar da mesma. No caso de ω - frequência de oscilação - corresponder à frequência natural do sistema, dada por, $\omega_0 = \sqrt{1/LC}$, a amplitude do sinal tenderá para infinito, teoricamente, o que, na prática, se traduz na saturação do mesmo. Desta forma, atenua-se a oscilação através da introdução de um componente resistivo no circuito. Afere-se, assim, que as oscilações provocadas por um circuito ressonante com resistência correspondem a oscilações forçadas com atenuação. [17]

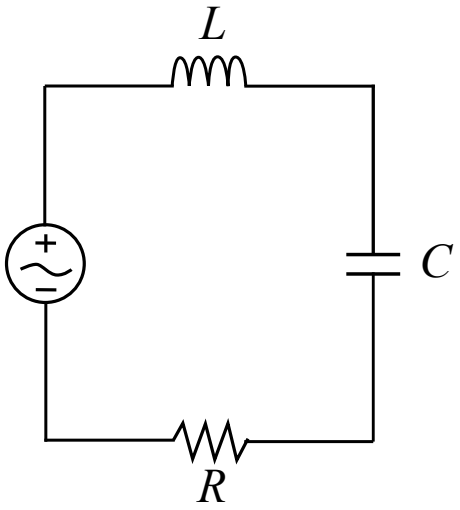

Figura 2.2: Circuito ressonante, composto por uma bobine (L), uma resistência (R) e um condensador (C), ao qual está aplicado uma força eletromotriz, tensão AC ($V(t)$), presente no livro *Vibrations and Waves* [17]

Define-se, assim, oscilador como sendo um circuito eletrónico que ultrapassa as perdas associadas a um circuito ressonante através da injeção de energia, à frequência de ressonância. Através do transitório inicial e da injeção de energia anteriormente mencionada, dá-se início à oscilação, sendo a sua amplitude limitada pelas não linearidades dos componentes ativos, como, por exemplo, os transístores. A resistência, sendo o seu valor negativo, a incluir no circuito, para compensar as perdas e manter a oscilação, é

determinada a partir da equação 2.10, [18].

$$R_n = -\frac{R_e(S_{21})}{50 \times \omega^2 \times C_1 \times C_2} \tag{2.10}$$

O oscilador de Colpitts, esquematizado na figura 2.3, é um exemplo de oscilador com componentes ressonantes e um componente ativo, sendo este composto por dois condensadores, um transístor NPN e uma resistência [18].

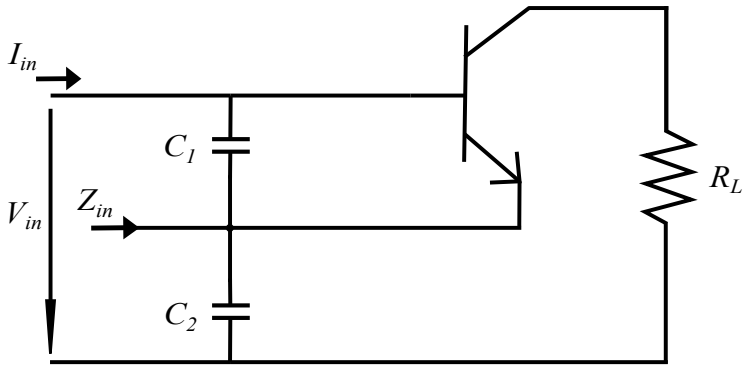

Figura 2.3: Esquemático exemplificativo de um oscilador de Colpitts, retirado do artigo "Oscillator basics and low-noise techniques for microwave oscillators and VCOs", [18]. Este oscilador é composto por dois condensadores (C_1 e C_2), um transistor NPN e uma resistência (R_L), pelo que, se considera que o oscilador tem componentes ressonantes, bem como um componente ativo.

Outra forma de definir o conceito de oscilador é explicando-o como sendo o circuito eletrónico que gera um sinal, [19], a partir de uma tensão de alimentação Direct Current (DC). Assim, pode-se explicar um oscilador como sendo um conversor de DC para AC [20].

Consequentemente, afere-se que a especificação mais relevante, nestes circuitos, é a sua frequência, podendo esta ser fixa ou variável. Esta característica é determinada pelos elementos presentes no circuito, como, por exemplo, as bobines, os condensadores ou as resistências - estes componentes formam redes LC ou RC, que, por sua vez, formam osciladores pelas suas propriedades reativas, como referido anteriormente. No entanto, estes constituintes possuem propriedades variáveis mediante a temperatura, tornando-os menos estáveis [19].

Existem vários tipos de osciladores, como por exemplo, os osciladores de Wien Bridge, de Hartley, de Colpitts, de Armstrong, de cristais, os osciladores LC, e os os osciladores em anel. Cada tipo de oscilador tem as suas vantagens e desvantagens.

Os osciladores em anel correspondem a uma combinação de células de atraso em cascata, ligadas entre si num ciclo fechado [8]. Desta forma, estes osciladores são criados com o objetivo de induzirem, a uma condição inicial, sucessivos atrasos de fase, por

forma a atingir uma oscilação estável, sendo esses atrasos provocados pelas células que o constituem [20].

Os osciladores lineares devem cumprir os critérios de Barkhausen por forma a torná-los viáveis. Estes critérios devem ser seguidos de modo a atingir-se uma oscilação estável, estando definidos através de duas equações, [20]:

- a equação do ganho em malha aberta,

$$|H(j\omega_0)\beta(j\omega_0)| = 1 \tag{2.11}$$

- a equação da margem de fase em malha fechada,

$$arg[H(j\omega_0))\beta(j\omega_0)] = 2k\pi \tag{2.12}$$

Apesar destes critérios garantirem a estabilidade da oscilação, não garantem que esta se dê. Desta forma, no início da referida oscilação, é necessário cumprir um terceiro critério, estando este definido pela seguinte equação [20]:

$$|H(j\omega_0)\beta(j\omega_0)| > 1 \tag{2.13}$$

Tal como mencionado anteriormente, os osciladores atrasam a fase do sinal sucessivamente, e, no caso dos osciladores em anel, a fase é atrasada π radianos a cada passagem pela célula. A frequência de oscilação dependerá do atraso de cada célula e do número de células, sendo esta dependência visível a partir da equação 2.14, [20], [8]:

$$f_{osc} = \frac{1}{t_d \times n} \tag{2.14}$$

Explicando o conceito anterior através de um exemplo concreto, no caso de uma célula de atraso demorar 1ns a inverter o sinal e o oscilador ser composto por cinco andares, pode-se afirmar que os valores teóricos do seu período e frequência são, respetivamente, de 5ns e 200MHz. No entanto, devido ao ruído presente no oscilador, o valor da frequência torna-se variável em cada ciclo, podendo tomar, para diferentes ciclos, os valores 200MHz e 201MHz, por exemplo.

Apesar das semelhanças inerentes a este tipo de osciladores, estes podem ser categorizados tendo em conta o número de entradas e saídas que possuem. Assim, destacam-se os osciladores diferenciais em anel - com duas entradas e duas saídas em oposição de fase, figura 2.4 - e os osciladores *single-ended* - com apenas uma entrada e uma saída, figura 2.5.

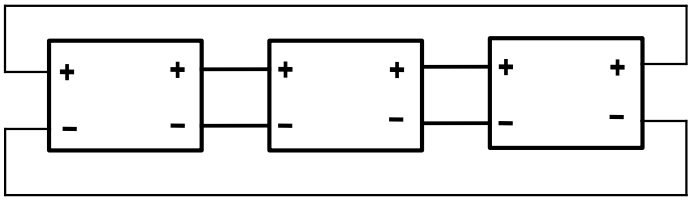

Figura 2.4: Esquemático do circuito de um oscilador diferencial em anel, composto por três células de atraso, ligadas diretamente entre si. Considera-se que cada célula de atraso é composta por duas entradas e duas saídas em oposição de fase.

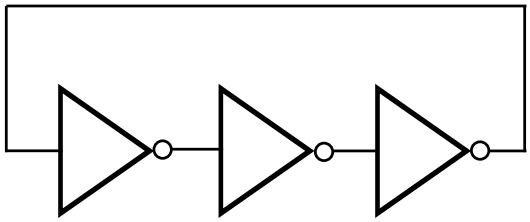

Figura 2.5: Esquemático do circuito de um oscilador em anel *single-ended*, composto por três células de atraso. Considera-se que cada célula de atraso é composta por uma entrada e uma saída.

O atraso de propagação de um inversor CMOS é caracterizado por *Docking* e *Sach-dev*. Um oscilador *single-ended* é geralmente composto por um P-type Metal Oxide Semiconductor (PMOS) e um N-type Metal Oxide Semiconductor (NMOS) ligados entre si. Quando existe a mudança entre *on* e *off*, os transístores Metal Oxide Semiconductor (MOS) funcionam na zona de saturação, linear e tríodo [8].

No caso dos osciladores diferenciais, estes apresentam outras características que os distinguem, para além do número de entradas e saídas presentes no circuito, no caso, o número mínimo de células presentes neste de forma a ser considerado um oscilador. Assim, este tipo de circuitos consegue oscilar quando composto por duas células, apenas no caso das entradas e saídas se encontrarem invertidas. Isto é, a saída positiva de uma célula encontrar-se ligada à entrada negativa da célula seguinte, como se observa na figura 2.6.

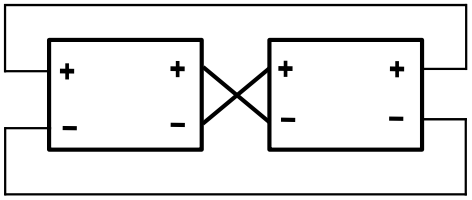

Figura 2.6: Esquemático do circuito de um oscilador em anel diferencial, composto por duas células de atraso, ligadas em cruz. Considera-se que as células de atraso têm duas entradas e duas saídas em oposição de fase..

O tipo de montagem referido anteriormente e representado na figura 2.6, é aplicável a todos os osciladores diferenciais com um número de células par. No caso do circuito mencionado ser composto por um número de células ímpar, o oscilador não funciona.

A tabela 2.1 resume este efeito de cruzamento, evidenciando passo a passo o efeito no sinal à passagem por duas células diferenciais e comparando com o não cruzamento.

Tabela 2.1: Tabela de comparação das transições de um sinal entre as células de atraso de um oscilador diferencial de dois andares, com as células ligadas diretamente e em cruz.

		Entrada 1ª Célula	Saída 1ª Célula	Entrada 2ª Célula	Saída 2ª Célula
Ligação Direta	+	1	0	0	1
	-	0	1	1	0
Ligação em Cruz	+	1	0	1	0
	-	0	1	0	1

Analisando a tabela 2.1, verifica-se que:

- Para um oscilador com as últimas duas células ligadas diretamente, no caso da injeção do sinal (1,0) na entrada da primeira célula, o valor presente na saída da última será, igualmente, (1,0), uma vez que este sinal sofre duas inversões. Isto é, o sinal é invertido na primeira célula e, por isso, à saída desta encontrar-se-á o valor (0,1), sendo este diretamente injetado na entrada da segunda célula, onde ocorrerá nova inversão do mesmo.

- No caso de um oscilador com as últimas duas células ligadas em cruz, no caso da injeção de um sinal (1,0) na entrada da primeira célula, o valor à saída será (0,1). Isto deve-se ao facto de o sinal ser invertido na primeira célula e, por isso, à saída desta o valor corresponderá a (0,1), sendo este injetado em cruz na entrada da segunda célula (o valor presente na saída positiva da primeira célula é injetado na entrada negativa da segunda célula e vice-versa), estando, então, aí presente o sinal (1,0), que será novamente invertido.

Assim, afere-se que ao cruzarmos as saídas e entradas, obtemos uma oscilação à saída do oscilador, o que não acontece na ligação direta dos terminais.

Os osciladores em anel apresentam-se com uma série de vantagens, sendo que se dá ênfase às seguintes, [8]:

1. estes osciladores são facilmente projetados a partir de tecnologias em circuito integrado (CMOS, BiCMOS);

2. as oscilações podem ser obtidas a tensões baixas;

3. as oscilações a altas frequências podem ser obtidas com uma baixa dissipação de potência;

4. estes podem estar sintonizados eletricamente;
5. devido à sua arquitetura, podem ter várias fases de sinais de saída;
6. podem ter uma gama de sintonizações alta.

2.5 Physically Unclonable Function

As PUF são circuitos com a função de armazenar chaves criptográficas de forma segura e não variável [21]. Isto é, estes circuitos recebem uma sequência de *bits* - os desafios - sendo, posteriormente, responsáveis pela produção de uma sequência de *bits* – a resposta - que integrarão o sinal de saída, denominando-se, à combinação do desafio e da sua resposta, par desafio-resposta, Challenge Response Pair (CRP). Assim, nenhuma PUF gera uma resposta idêntica a outra, para um mesmo desafio [22].

Os circuitos PUF são implementados recorrendo-se a diferentes tecnologias CMOS, enfatizando-se os *chips* de memória, a Application Specific Integrated Circuit (ASIC) ou a Field-Programmable Gate Array (FPGA), e, por isso, a sua construção pode ter por base o composto silício [23], [22]. Esta escolha prende-se com o facto de as tecnologias CMOS terem uma grande variabilidade, causada por imperfeições nos processos de fabrico, o que leva a transmutações intrínsecas e aleatórias nas características físicas e elétricas destes circuitos, sendo a resistividade metálica e o comprimento efetivo do canal dos transístores exemplos das variações mencionadas. Desta forma, estas propriedades mutáveis dos circuitos são aproveitadas para gerar uma identidade única, conhecida como assinatura única ou impressão digital, para cada equipamento de *hardware* [23], [22], [24].

De todas as tecnologias CMOS enumeradas, a implementação deste tipo de circuitos é feita, preferencialmente, recorrendo a FPGA, uma vez que a sua integração se torna mais fácil e configurável [21].

Os circuitos PUF podem ser caracterizados de duas formas distintas [22]:

- PUF forte – nesta categoria, o aumento do tamanho do circuito induz a um crescimento exponencial do número de CRPs.
- PUF fraca – nesta categoria, o aumento do tamanho do circuito representa um crescimento linear.

Os circuitos PUF têm de ser robustos uma vez serem empregados para a conservação de informação sensível e, por isso, estarem sujeitos a ataques exteriores, com alguma regularidade [21]. Desta forma, estes circuitos são avaliados pela capacidade de manter a sua chave inalterada, sendo que esta avaliação é feita recorrendo ao valor médio da chave, à correlação entre *bits* e ao consumo de potência e energia por *bit*. Considera-se

uma PUF como ideal, se esta apresentar os seguintes valores: valor mediano $\mu = 0,5$, com $\sigma = \frac{\sqrt{N}}{2}$; rácio de erro (ou $HD_{intra120C}$) de 0%; correlação entre *bits* (ou R_{xx}) igual a 0, com $\sigma = \frac{1}{2*\sqrt{N}}$; correlação entre *chips* (ou HD_{inter}) de 50%, com $\sigma = 100\%\frac{\sqrt{N}}{2}$; e consumo de potência (ou E/bit) igual a $0pJ/bit$.

Os *bits* devem apresentar um valor médio de 0,5, isto porque, para que estes sejam considerados aleatórios não se devem apresentar tendenciosos em relação a um dos valores, sendo imperativo que o número de "0" e "1" seja igual. Desta forma, para se efetuar a avaliação do circuito deve-se recorrer à equação 2.16, de onde se retira do valor médio, estando este, idealmente, próximo de 0,5, como já referido.

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2.15}$$

O rácio de erro é diretamente proporcional aos fatores aleatórios dos circuitos, como é exemplo destes o ruído térmico, sendo que o seu valor deve aproximar-se de 0%, para que o *bit* à saída da célula se mantenha [25–27].

A correlação entre *bits* ocorre quando existe uma interferência dos *bits* atuais com os *bits* futuros, sendo que a correlação entre dois dados induz numa menor combinação destes, levando à diminuição da aleatoriedade. A existência de correlação entre dois dados é medida através da função de autocorrelação 2.16:

$$R_{xx}(j) = \sum_{n} x_n x_n - j \tag{2.16}$$

no qual j representa o atraso do sinal.

Para os casos em que se obtém os valores $R_{xx}(j) = 1$ e $R_{xx}(j) = -1$, afere-se que existe uma correlação entre os *bits* e, portanto, há perda de aleatoriedade no sistema. Para o caso em que se chega ao valor de $R_{xx}(j) = 0$, estamos perante o caso ideal, em que não existe qualquer correlação entre os dados. Desta forma, pretende-se que o valor de R_{xx} seja aproximadamente 0.

O consumo destes circuitos é medido em W/bit e a energia em J/bit. Conforme a literatura consultada ([23], [25], [26]), os valores, considerados ideais, de potência rondam as unidades do mW e a energia as unidade do pJ/bit.

Uma exemplificação do dimensionamento de uma PUF é descrita por Mansouri et al., no artigo "*Ring Oscillator Physical Unclonable Function with Multi Level Supply Voltages*". A PUF, representado na figura 2.7, é composta por dois osciladores ligados a *multiplexers* e a uma célula de memória [28].

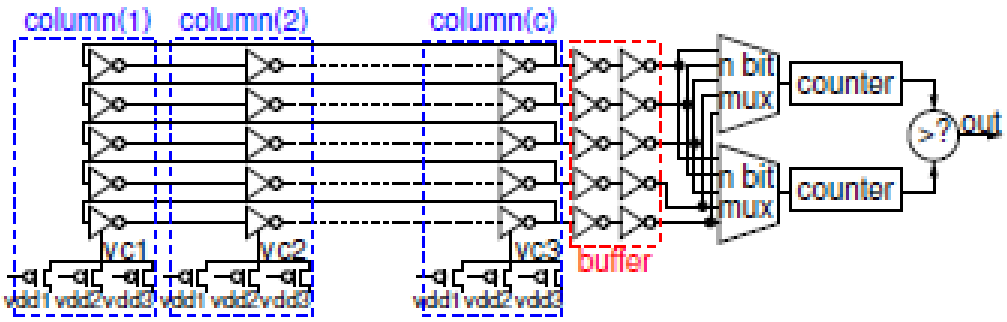

Figura 2.7: PUF apresentada no artigo "*Ring Oscillator Physical Unclonable Function with Multi Level Supply Voltages*"[28]. Esta PUF é composta por dois osciladores ligados a *multiplexers* e uma célula de memória.

2.6 Testes de Aleatoriedade

Existem diversas formas de proceder aos testes de aleatoriedade de um gerador de números aleatórios, apresentando-se de seguida algumas destas.

A ferramenta NIST SP 800-22 consiste em 15 testes estatísticos, sendo estes:

1. Teste de Frequência
2. Teste de Frequencia dentro de uma *bitstream*
3. Teste de *Runs*
4. Teste da maior sequência de "1"numa *bitstream*
5. Teste de classificação de matriz binária
6. Teste da transformada de Fourier discreta (espectral)
7. Teste de correspondência de modelos não sobrepostos
8. Teste "Estatístico Universal"de Maurer
9. Teste de Complexidade Linear
10. Teste de Série
11. Teste de Entropia Aproximada
12. Teste de Somas Cumulativas
13. Teste de Excursões Aleatórias

14. Teste Variante de Excursões Aleatórias.

Desta forma, para se elaborar os testes de aleatoriedade, através da ferramenta NIST SP 800-22, os *bits*, inseridos nesta, são divididos de forma sequencial em conjuntos de n (vários) elementos, de forma a obter-se o número de *bitstreams* pretendido. Seguidamente, a ferramenta gera dados estatísticos e um p-value, para cada um dos testes anteriormente enumerados, e compara o p-value com o nível de significância (α), sendo este valor, em criptografia, de, tipicamente, 0,01 [29], sendo que se:

- p-value $\leq 0,01$, o teste é chumbado.
- p-value $> 0,01$, o teste é passado.

Embora o conjunto de testes NIST SP 800-22 seja utilizado na literatura para testar todo o tipo de RNG, tal como mencionado anteriormente, Stipčević et al. [6] afirmam que este conjunto de testes apenas se aplicam a PRNG.

O Gabinete Federal da Segurança Informática alemão possui um conjunto de testes de aleatoriedade, denominado AIS 20/31, que apresenta dois conjuntos de testes: o AIS 20, para PRNG e o AIS 31, para TRNG.

Por forma a avaliar a robustez dos RNG, esta ferramenta apresenta três classes de funcionalidade para os geradores, sendo estas a PTG.1, a PTG.2 e a PTG.3, sendo esta última a classe ideal.

Para um gerador ser classificado como um TRNG, através da ferramenta enumerada, este tem de estar em conformidade com os critérios da segunda ou terceira classes.

Os critérios para classificar um gerador como sendo da segunda classe são:

- este não ter pós-processamento e/ou pós-processamento criptográfico
- este não pode criar, de forma direta, chaves criptográficas
- obter validação nos testes de aleatoriedade

e os critérios para classificar um gerador como sendo da terceira classe são:

- este ser um gerador de segunda classe com pós-processamento criptográfico.
- obter validação nos testes de aleatoriedade [1]

Tal como a ferramenta NIST SP 800-22, o AIS 31 também consiste em alguns testes estatísticos, de forma a avaliar a aleatoriedade dos geradores, sendo estes:

- T0 - Teste de Desarticulação
- T1 - Teste do *Monobit*

[1] A ferramenta ainda não apresenta os testes necessários para se proceder à validação da terceira classe.

- T2 - Teste de *Poker*
- T3 - Teste *Runs*
- T4 - Teste de Funcionamento Longo
- T5 - Teste de Auto correlação
- T6 - Teste de Distribuição Multinomial
- T7 - Teste de Entropia.

3

Estado de Arte

Os osciladores, como geradores de sinal periódico, não apresentam utilidade em criptografia, dado que são previsíveis e estáveis na maioria das suas utilizações. No entanto, se recorrermos a uma propriedade específica dos transístores, nomeadamente o seu ruído inerente, os osciladores podem ser utilizados como geradores de números aleatórios. Esta aleatoriedade deve-se às fontes de entropia, ou seja, o ruído dos transístores, que afetam o sinal de saída dos osciladores, tornando a sequência de *bits*, à saída destes, imprevisível.

Aquando do design de osciladores integrados, devem ser cumpridas algumas metas, tais como: minimizar o ruído, o consumo e a área no *chip*. No caso dos geradores de números aleatórios, a redução do seu consumo e da sua área deve ser tida em conta, no momento do seu design, devido à necessidade da sua portabilidade e integração. Em relação às fontes de ruído, estas devem ser maximizadas, de forma a potenciar a aleatoriedade dos osciladores em anel, ao contrário do verificado nos osciladores utilizados em comunicações, por exemplo. Assim, os osciladores, quando aplicados em criptografia, devem ter as suas fontes de ruído amplificadas, sem que isto comprometa o seu tamanho e consumo.

Este tema tem ganho relevo devido à crescente preocupação com a segurança digital. Consequentemente, foram escritos vários artigos que visam o desenvolvimento ou a testagem e avaliação estatística de diferentes RNG. Desta forma, selecionaram-se alguns artigos com diferentes arquiteturas e princípios de funcionamento, nomeadamente os geradores com osciladores de corrida livre, osciladores baseados no caos, osciladores baseados em ruído de Johnson e geradores quânticos, considerados relevantes para o desenvolvimento desta dissertação, estando estes brevemente descritos em seguida.

3.1 *Free Running Oscillator*

Os geradores de números aleatórios FRO são circuitos produtores de números aleatórios provenientes do ruído de fase de osciladores. A existência de dois osciladores no circuito permite a criação de um sinal com frequência alta e um sinal de *clock* com uma frequência mais lenta, sendo que ambos sofrem distorção, devido à presença de ruído, maioritariamente térmico ou $1/f^2$, tornando, desta forma, os sinais um pouco aleatórios mas, no entanto, apenas estes dois osciladores não produzem um sinal aleatório o suficiente. O oscilador responsável pela criação do sinal de *clock* caracteriza-se por ser um oscilador lento, cujo ruído térmico tem uma maior influência, sendo que o outro circuito se caracteriza por ser um oscilador rápido, sendo este desenhado para que a sua frequência não corresponda a um valor múltiplo da frequência do oscilador lento, isto porque ambos os osciladores devem ter frequências distintas.

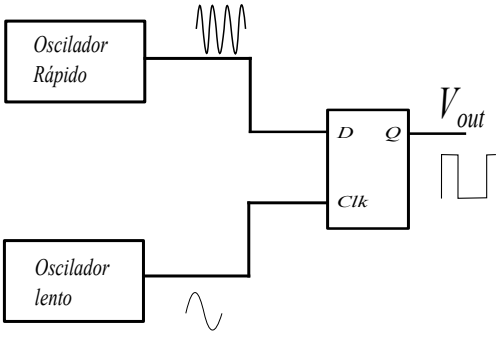

Figura 3.1: Circuito resumido de um gerador de números aleatórios baseado em FRO.

Esta categoria de geradores de números aleatórios são, normalmente, desenhados e implementados em FPGA, como consequência da sua fácil execução e testagem, sendo, também, mais configuráveis e escaláveis. No entanto, este tipo de geradores comprometem a área e a potência consumida, não estando o seu *design* otimizado para o efeito.

No âmbito deste tópico, Petrie et al. [30] desenvolveram um gerador de números aleatórios FRO, visando responder às seguintes questões, durante o processo de modelação:

- Quanto ruído de fase é necessário de forma a produzir aleatoriedade suficiente?
- Que relação entre frequências de oscilação terá um comportamento melhor?
- Os osciladores são influenciados por causas determinísticas externas?
- Este método é sensível ao ruído de *flicker* (ou 1/f)?

3.2 Circuitos geradores de sinais caóticos

Os geradores de sinais caóticos apresentam fontes caóticas que, à primeira vista, são aleatórias mas, fazendo uma análise matemática a estes geradores, verifica-se que são baseados em fontes determinísticas. No entanto, pode-se tornar uma fonte determinística numa fonte aleatória mais robusta se a combinarmos com uma fonte aleatória como o ruído.

Apresenta-se então duas soluções para geradores de números aleatórios baseados em fontes caóticas de sinal, apresentados nos artigos de Tastan et al. [14] e Ergün [13].

Ambos os artigos apresentam um circuito gerador de sinal caótico contínuo composto por 2 díodos, duas resistências e um condensador que, ligado a dois osciladores apresentam um sinal ruidoso à saída. No entanto, os artigos diferem no bloco em que colocam o circuito, Tastan et al. aplica o circuito no bloco rápido do gerador, como evidenciado na figura 3.4, enquanto que Ergün o aplica no *clock*, como está representado na figura 3.3.

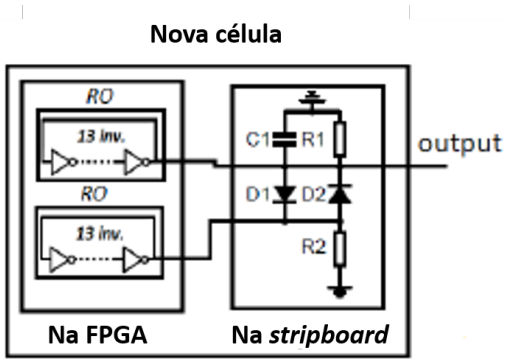

Figura 3.2: Gerador de números aleatórios desenvolvido no artigo "*A Robust Random Number Generator Based on Chaotic Ring Oscillators*"[14].

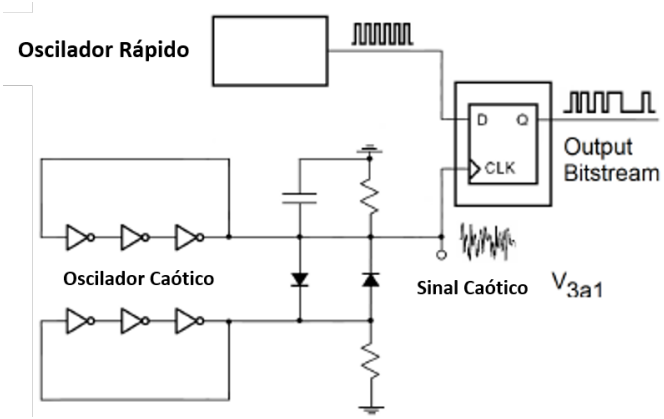

Figura 3.3: Gerador de números aleatórios apresentado no artigo "*Cryptanalysis of a Chaotic Ring Oscillator Based Random Number Generator*"[13]

Tastan et al. refere que a aplicação do circuito no bloco de dados é mais importante

porque este apresenta mais ciclos para sofrer o efeito caótico enquanto que Ergün afirma que, como a frequência de *clock* é mais reduzida, o efeito do ruído é mais impactante. No seu artigo, Tastan et al. refere que, para aumentar a aleatoriedade do sistema, é possível aumentar o número de circuitos caóticos, juntando-os com uma XOR 3.4, evidenciando que são necessários dez circuitos da figura 3.2 para que o sistema passe nos testes de aleatoriedade. Já Ergün apenas apresenta o seu circuito de forma mais rígida, não demonstrando a escalabilidade deste.

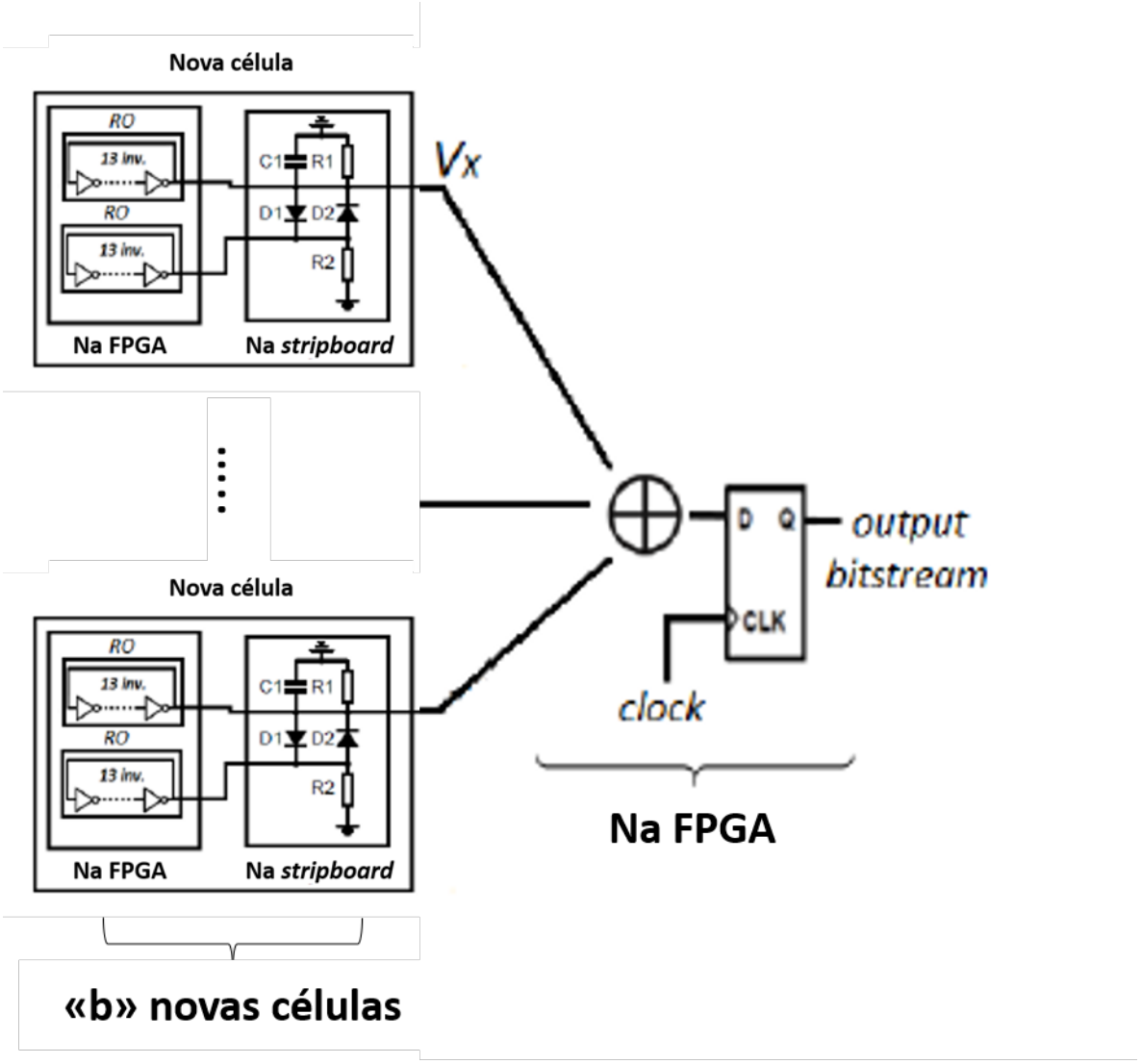

Figura 3.4: Gerador de números aleatórios desenvolvido no artigo "*A Robust Random Number Generator Based on Chaotic Ring Oscillators*"[14].

Tastan et al. tiveram em consideração a redução da área do mesmo, sendo este, tal como supramencionado, um aspeto importante no *design* de osciladores. Esta redução foi possível dado não ser necessário *flipflops* do tipo D em cada célula, de modo a ser possível atingir a aleatoriedade pretendida, estando estes presentes, apenas, no final do RNG com

o objetivo de amostrar o sinal. Para além da redução da área do RNG, alcançada através do *design* proposto por Tastan e Ergün, os autores poderam concluir que, para passar nos testes de aleatoriedade, o RNG só necessita de dez células, enquanto que no gerador clássico são necessárias vinte e cinco células. Para isso, os autores efetuaram os testes do NIST 800-22, utilizando quarenta *megabytes* de dados.

3.3 Ruído de Johnson

Estes geradores utilizam o ruído como fonte de entropia para gerar o sinal aleatório. O ruído de Johnson (ou térmico) é produzido pela passagem de corrente em componentes resistivos, por exemplo, resistências e transístores (3.5).

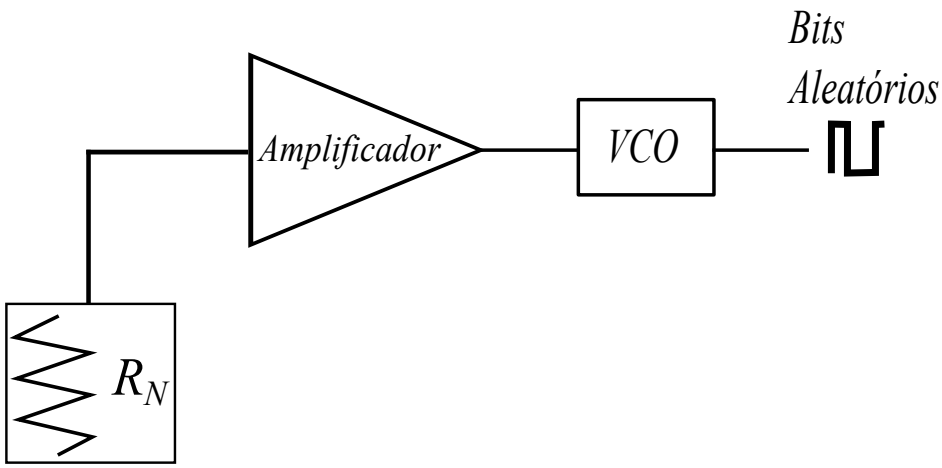

Figura 3.5: Blocos constituintes do circuito RNG baseado em ruído térmico em [31]

O sistema pode ser resumido pela figura 3.5, que mostra, por ordem da esquerda para a direita:

1. Fonte de ruído térmico (neste exemplo uma resistência);
2. Amplificador de sinal;
3. Oscilador controlado por tensão (VCO).

A fonte de ruído é constituída por um componente que produza ruído térmico em funcionamento, como por exemplo uma resistência ou um transístor. Este ruído é gerado porque estes componentes oferecem resistência à passagem da corrente, provocando um aumento da temperatura. Este aumento da temperatura provoca um aumento da excitação das cargas, que vibram mais provocando assim um aumento do ruído no sinal.

O amplificador de sinal é um bloco muito importante neste tipo de geradores de sinais aleatórios porque a sua função é aumentar o impacto do ruído produzido no bloco anterior, visto que a sua variação de tensão é muito baixa comparado com a tensão do sinal.

O bloco de oscilador tem a função de criar o sinal a distorcer, visto que os dois blocos anteriores apenas produzem ruído. O sinal ruidoso é utilizado para controlar a frequência do oscilador, tornando-a aleatória. Este oscilador não precisa de ter uma forma de onda concreta, podendo ter qualquer forma, sendo que o mais importante são as transições positivas, isto é, quando a oscilação passa de '0' para '1'.

3.4 Geradores quânticos

A física quântica é a ciência que estuda o universo através das partículas finitas, isto é, na física clássica, a luz é vista como uma onda eletromagnética, enquanto que na física quântica, a luz é vista como uma quantidade finita de fotões em movimento. Quando esta escala é atingida, perde-se as certezas que se tinha com a física clássica. Um exemplo desta indefinição pode ser evidenciada pelo princípio da incerteza de Heisenberg [32], evidenciado na equação 3.1 que, resumidamente, nos indica que não há forma de saber a posição e o momento da partícula ao mesmo tempo, ou seja, ou sabemos onde a partícula está, mas não sabemos para onde se dirige, ou sabe-se para que direção vai mas não se sabe em que posição se encontra.

$$\Delta x \times \Delta p \geq \frac{\hbar}{2} \tag{3.1}$$

em que x é a posição, p é o momento e $\hbar$ é a constante de Planck.

Posto isto, pode-se afirmar que a física quântica pode ser muito útil em termos de aleatoriedade, visto que os próprios princípios desta escala são incertezas.

4

Circuito gerador de números aleatórios

Apresentados, no capítulo 2, os conceitos determinantes para o desenvolvimento da presente dissertação, bem como os trabalhos desenvolvidos nesta área, que se consideram de maior relevância, no capítulo 3, no presente capítulo encontram-se descritas as diferentes etapas concretizadas que permitiram alcançar o circuito final.

Para este efeito, tiveram-se em consideração os artigos [30], [14] e [13]. Porquanto, o circuito *Free Running Oscillator* apresenta-se com constituintes simples e com um bom aproveitamento do ruído. Relativamente aos circuitos geradores de sinais caóticos, estes melhoram o comportamento do sinal ao nível do seu ruído, não incrementando o seu consumo. Assim, partiu-se dos princípios inerentes às arquiteturas mencionadas para o desenvolvimento do RNG. No entanto, o FRO, presente no artigo [30], encontra-se implementado numa FPGA, comprometendo, desta forma, a necessidade de um consumo e área reduzidos. Relativamente aos circuitos geradores de sinais caóticos, apresentados nos artigos [14] e [13], estes não estão, de forma recorrente, implementados em circuitos integrados. Assim, tendo em consideração os últimos pontos apresentados, o circuito desenvolvido na dissertação revela-se como inovador na área onde se insere.

4.1 Projeto do circuito

Com a presente dissertação criou-se um gerador de números aleatórios através de um circuito composto por um oscilador e um circuito gerador de sinal caótico Desta forma, o circuito que compõe o gerador de números aleatórios integra dois blocos geradores de sinais caóticos, dois osciladores de dados rápidos, uma XOR, um oscilador diferencial de baixo consumo e um *flip-flop*.

O funcionamento do circuito pode ser resumido pela figura 3.1, o qual se pode dividir o circuito em três partes:

1. Circuito oscilador de dados rápidos;
2. Circuito oscilador de *clock*;
3. Circuito de amostragem digital.

O circuito proposto apresenta ainda um bloco extra, sendo este um circuito gerador de sinal caótico. Este circuito, mesmo sendo uma fonte determinística, como está conectado a uma fonte de ruído aleatória (o oscilador), esta torna-o, de igual forma, aleatório.

A escolha destes circuitos teve em conta as características pretendidas para o RNG final e os resultados obtidos em cada uma das etapas da sua construção, sendo estes:

- Área ocupada pelo circuito;
- Consumo de potência;
- Consumo de transmissão - energia gasta pelo RNG para enviar um *bit*;
- Velocidade de transmissão - número de *bits* gerados por segundo;
- Frequência de *clock* e de dados.

Para se obter os resultados de área ocupada dos diferentes circuitos recorreu-se às equações 4.1, 4.2 e 4.3, correspondendo estas, respetivamente, à área ocupada por cada célula de atraso, por cada oscilador e pelo RNG, sendo que as áreas das resistências e dos condensadores foram estimadas.

$$A_{cel} = L_p \times W_p + L_n \times W_n \quad (\mu m^2) \tag{4.1}$$

sendo que L_p e W_p correspondem à soma dos comprimentos e larguras de todos os transístores PMOS, L_n e W_n a soma dos comprimentos e larguras de todos os transístores NMOS.

$$A_{osc} = A_{celula} \times N_{celulas} \quad (\mu m^2) \tag{4.2}$$

$$A = A_{dados} + A_{clock} + A_{digital} + A_{caos} \quad (\mu m^2) \tag{4.3}$$

Os valores de consumo de transmissão foram calculados segundo a equação 4.4.

$$E_t(pJ/bit) = \frac{P(mW)}{v_{trans}(Mbits/s)} \tag{4.4}$$

Já os valores de consumo de potência, número de *bits* gerados em $1\mu s$, frequência de *clock* e frequência de dados obtiveram-se por meio de simulações. A partir da equação 4.5 e do número de *bits* por $1\mu s$, valor esse retirado da simulação, obteve-se o valor de velocidade de transmissão.

$$v_{trans} = \frac{bits}{1\mu s} \times 10^6 \quad (bits/s) \tag{4.5}$$

Nos subcapítulos 4.1.1, 4.1.2 e 4.1.3 encontram-se descritos cada um dos blocos que integram o gerador de números aleatórios, sendo estes o oscilador diferencial de *clock*, o oscilador *single-ended* de dados e o circuito gerador de sinal caótico.

4.1.1 Oscilador diferencial

A escolha de um oscilador diferencial, como gerador de sinais, teve em conta as características que se consideraram de maior relevância para o projeto, a saber: o baixo consumo, a baixa frequência, característica do oscilador de *clock*, e a fonte de ruído com origem no oscilador de dados rápido. Desta forma, apesar deste oscilador ter um ruído relativamente baixo, esta característica não é determinante no circuito do gerador de números aleatórios.

O oscilador diferencial foi desenhado para ter uma frequência de trabalho entre 20 e 30 MHz (admitindo uma margem de erro de 10%), recorrendo à tecnologia de 130 *nm* e a uma tensão de alimentação de $1,2V$. Foi também tido em conta o consumo do mesmo, tentando minimizar a corrente de funcionamento. A potência que se espera obter é na ordem das centenas de μW para o oscilador, com mais de quatro células juntas, para maximizar o ruído.

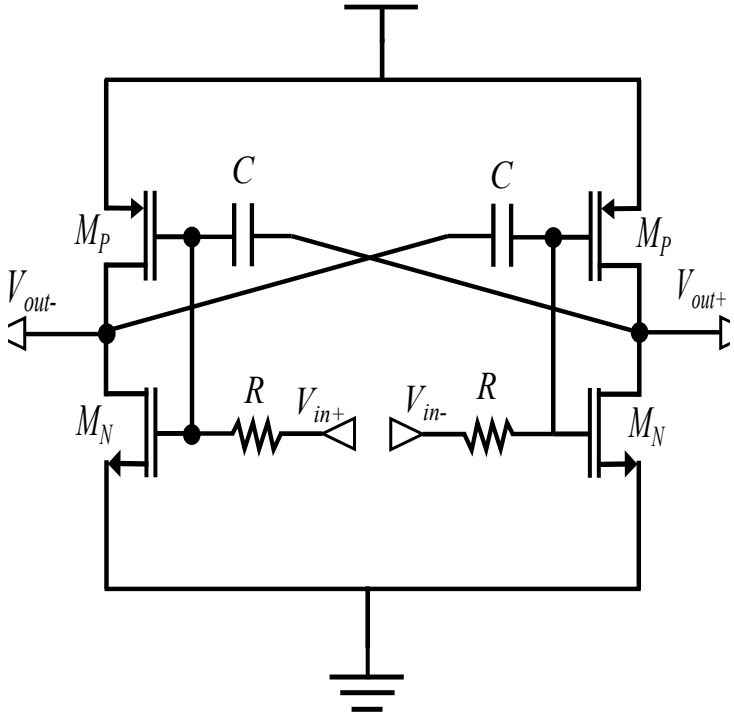

Figura 4.1: Circuito da célula de atraso diferencial com *latch*, através de uma malha RC, presente no oscilador arquitetado na dissertação.

Projetaram-se dois osciladores diferenciais com arquiteturas diferentes, um com *latch* através de uma malha RC, representado na figura 4.1, e um comum com *latch*, através de transístores, representado na figura 4.2, sendo a arquitetura escolhida para a presente dissertação a primeira enumerada. Desta forma, foi possível proceder à comparação entre os dois osciladores, averiguando-se, assim, se a solução escolhida constitui a melhor opção para o circuito pretendido. Note-se que a comparação referida encontra-se descrita no capítulo 5.

A arquitetura do oscilador diferencial com *latch*, através de uma malha RC — não contém ligação entre os ramos da célula — reduz, teoricamente, o consumo da mesma, figura 4.1. Além disso, a célula apresenta uma resistência e um condensador à entrada que, ao encontrarem-se ligados à *gate* dos transístores, não consomem corrente, mantendo-se, assim, o efeito pretendido.

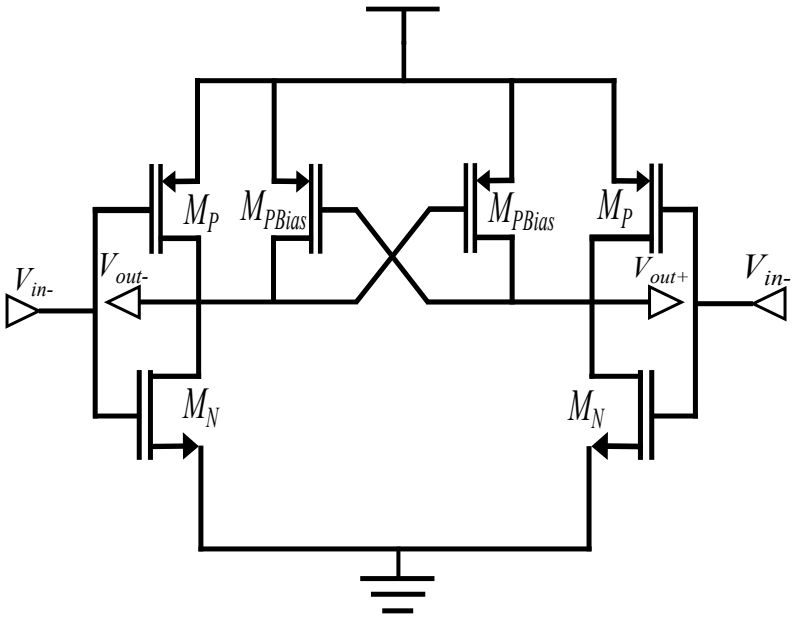

Figura 4.2: Célula de atraso diferencial com *latch* através de transístores.

Na tabela 4.1 são apresentados os valores de comprimento (L) e largura (W) dos transístores em par diferencial de ambos os osciladores. Estes valores permitem calcular a área ocupada pelo oscilador diferencial, de acordo com as equações 4.1 e 4.2.

Tabela 4.1: Valores de comprimento (L) e largura (W), em μm, dos transístores PMOS e NMOS, presentes em ambos os osciladores diferenciais apresentados (oscilador diferencial com *latch*, através de uma malha RC - célula proposta - oscilador diferencial com *latch*, através de transístores - célula alternativa).

Célula	PMOS-L (μm)	PMOS-W (μm)	NMOS-L (μm)	NMOS-W (μm)
Proposta	1	1,5	1	0,5
Alternativa	2	6	2	2

4.1.2 Oscilador de dados rápido

No presente trabalho escolheu-se, como oscilador de dados rápido, um oscilador *single-ended*, cuja frequência apresenta-se elevada de forma a alimentar o *flipflop*, para que seja possível amostrar o sinal. Esta decisão teve em conta a necessidade de se incorporar no circuito um oscilador mais ruidoso e fácil de implementar.

A célula de atraso selecionada é um inversor CMOS, sendo característico deste tipo de circuitos a presença de um transistor PMOS e um transistor NMOS, estando o esquemático desta célula representado na figura 5.3.

Apesar de se ter elaborado o projeto do oscilador, é possível, em alternativa, obter-se o sinal a partir de uma FPGA, por forma a tornar o circuito mais configurável.

Este oscilador deve apresentar uma frequência não múltipla da frequência do oscilador de amostragem, para que estes valores não se combinem, evitando, assim, a destruição da aleatoriedade do sistema. Como referido em 4.1.1, a frequência do oscilador de amostragem é de 26,28 MHz, portanto projetou-se o oscilador de dados com uma frequência de 500 MHz, uma vez que a divisão de ambos os valores corresponde a um valor decimal não periódico.

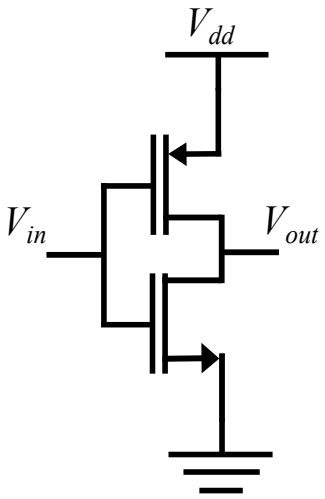

Figura 4.3: Circuito de um inversor CMOS.

Na tabela 4.2 encontram-se os valores de comprimento (L) e largura (W) dos transístores do inversor. Estes valores permitem calcular a área ocupada pelo oscilador de dados, de acordo com as equações 4.1 e 4.2.

Tabela 4.2: Valores de comprimento (L) e largura (W), em μm, dos transístores PMOS e NMOS, presentes na célula de atraso do oscilador de dados.

Célula	PMOS - W (μm)	PMOS - L (μm)	NMOS - W (μm)	NMOS - L (μm)
Inversor	3	0,5	1	0,5

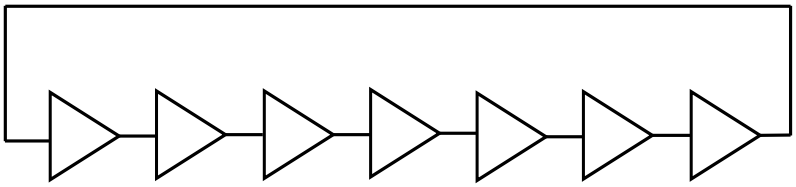

Figura 4.4: Circuito do oscilador em anel *single-ended*, composto por sete células de atraso, presente no RNG projetado nesta dissertação.

4.1.3 Circuito gerador de sinal caótico

Conforme já referido, ao unir o circuito gerador de caos aos osciladores de dados, a aleatoriedade do circuito proposto é ampliada.

O circuito gerador de caos escolhido para o efeito, encontra-se no artigo [13], sendo este composto por duas resistências, dois díodos e um condensador, como é possível observar na figura 4.5. Na tabela 4.3, encontram-se discriminados os valores, que se consideram ótimos, das resistências e do condensador, previamente mencionados.

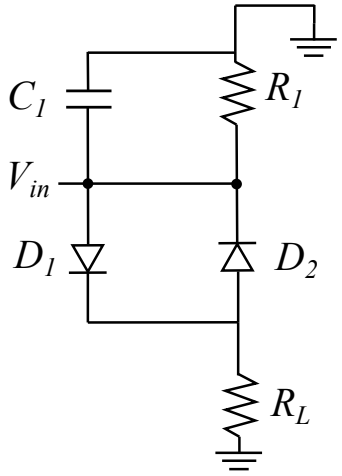

Figura 4.5: Célula de caos presente no artigo 4.5, sendo parte integrante do RNG criado na dissertação. Este circuito é composto por duas resistências (R_1, R_L), dois díodos (D_1, D_2) e um condensador (C_1).

Tabela 4.3: Valores ótimos para as resistências $R1$ e $R2$ e para o condensador C, presentes no circuito gerador de sinal caótico utilizado no RNG desenvolvido na dissertação.

Componente	R1	R2	C
Valor	$100k\Omega$	$10k\Omega$	$1pF$

4.1.4 Circuito gerador de números aleatórios desenvolvido na dissertação

Por forma a cumprir os requisitos inicialmente estabelecidos para o circuito, foram desenvolvidas três variantes do mesmo. Assim, elaborado o projeto de todos os blocos que compõem o circuito do gerador de números aleatórios (RNG), procedeu-se à montagem da primeira versão do mesmo, sendo esta a versão base. Desta forma:

1. ligou-se o oscilador de *clock* à entrada de *clock* de um *flip-flop* do tipo "D";
2. ligaram-se dois osciladores de dados rápidos a uma célula geradora de caos, criando-se dois conjuntos deste tipo;
3. ligaram-se as duas células de caos a uma XOR;
4. ligou-se a saída da XOR à entrada de dados do *flip-flop* do tipo "D".

O circuito com todos os blocos ligados apresenta-se na figura 4.6, onde se encontram evidenciados os seus constituintes.

Por forma a resumir os resultados obtidos na otimização do circuito base, apresenta-se a tabela 4.4, que contém os valores de comprimento (L) e largura (W) dos transístores presentes em cada um dos blocos do circuito proposto, bem como os valores das resistências e condensadores presentes no oscilador de *clock* e no circuito gerador de caos. Tal como anteriormente mencionado, estes valores permitem calcular a área ocupada pelo RNG, de acordo com as equações 4.1 e 4.2.

Tabela 4.4: Tabela com os valores dos componentes de todas as partes constituintes do circuito RNG base.

	PMOS W(μm)	PMOS L(μm)	NMOS W(μm)	NMOS L(μm)	R1 ($k\Omega$)	C1 (pF)	R2 ($k\Omega$)
Oscilador Rápido	3	0,5	1	0,5	-	-	-
Oscilador *clock*	1	1,5	1	0,5	5	0,5	-
Circuito gerador de caos	-	-	-	-	100	1	10

Uma vez que o circuito RNG base, esquematizado na figura 4.6, se caracteriza pela sua escalabilidade, é possível adicionar-se, a este, mais pares de osciladores de dados rápidos. Desta forma, procedeu-se à arquitetura da segunda versão do circuito gerador de números aleatórios (RNG). Para isso, adicionou-se, ao esquemático do circuito base, mais

dois pares de osciladores de dados rápidos e dois circuitos geradores de caos, conectados por uma *XOR*, como é observável na figura 4.7, ficando, desta forma, o circuito com oito osciladores rápidos, do tipo *single ended*, quatro circuitos geradores de caos, estando estes ligados por uma *XOR*, um oscilador de *clock* e um *flip-flop*.

A terceira versão do circuito RNG, tem por base o circuito desenvolvido na segunda versão, utilizando uma tensão de alimentação diferente - $[0,8;1,1]V$ - em cada um dos pares de osciladores rápidos, presentes na figura 4.7.

Devido à dificuldade de simulação, inerente aos circuitos de elevada dimensão, as simulações efetuadas, e analisadas no capítulo 5, tiveram por base o circuito representado na figura 4.6. Já os testes de aleatoriedade, realizados a partir da ferramenta NIST, também apresentados no capítulo 5, tiveram em conta as três versões do circuito gerador de números aleatórios, desenvolvidas na presente dissertação.

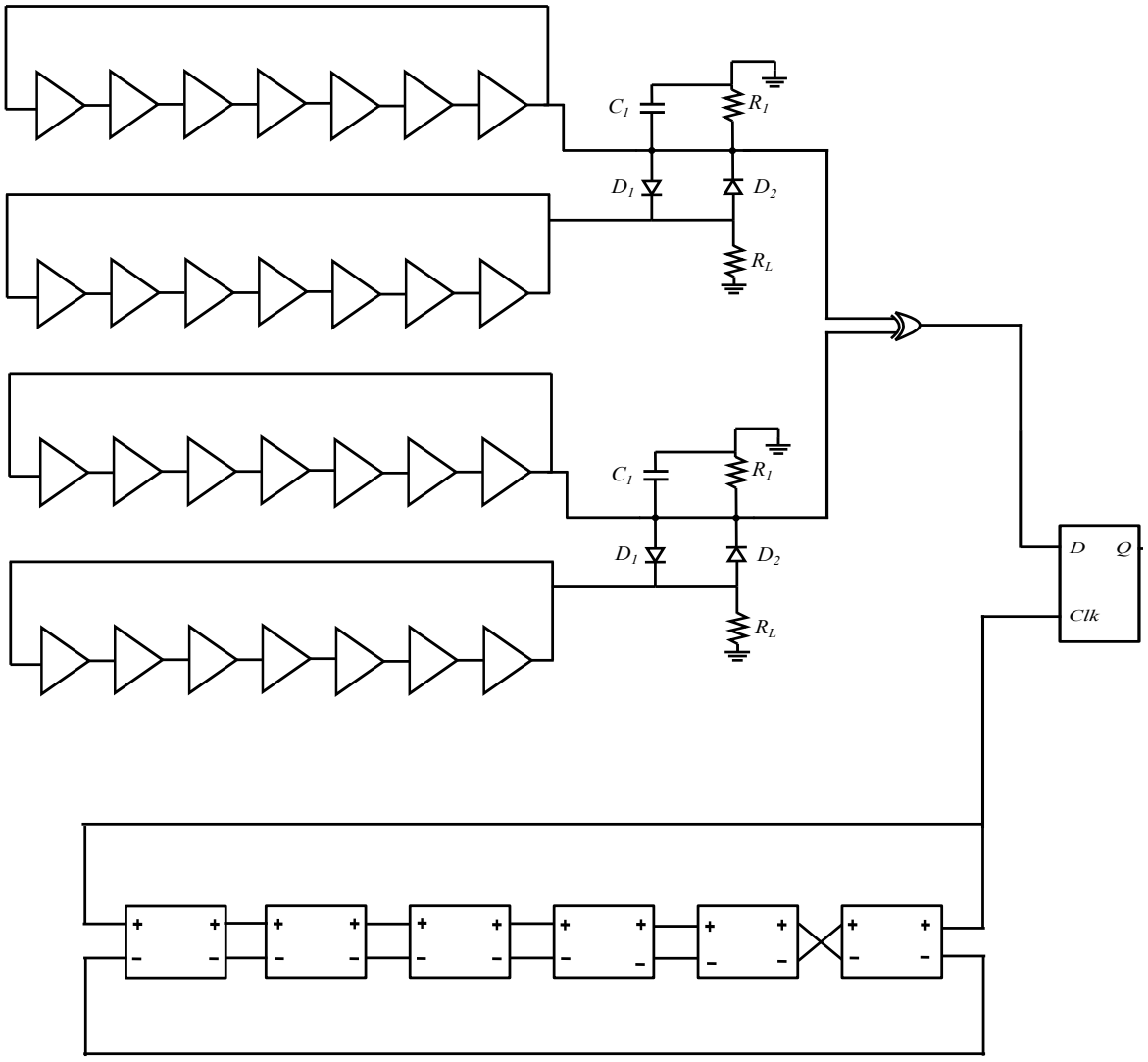

Figura 4.6: Circuito gerador de números aleatórios desenvolvido na presente dissertação.

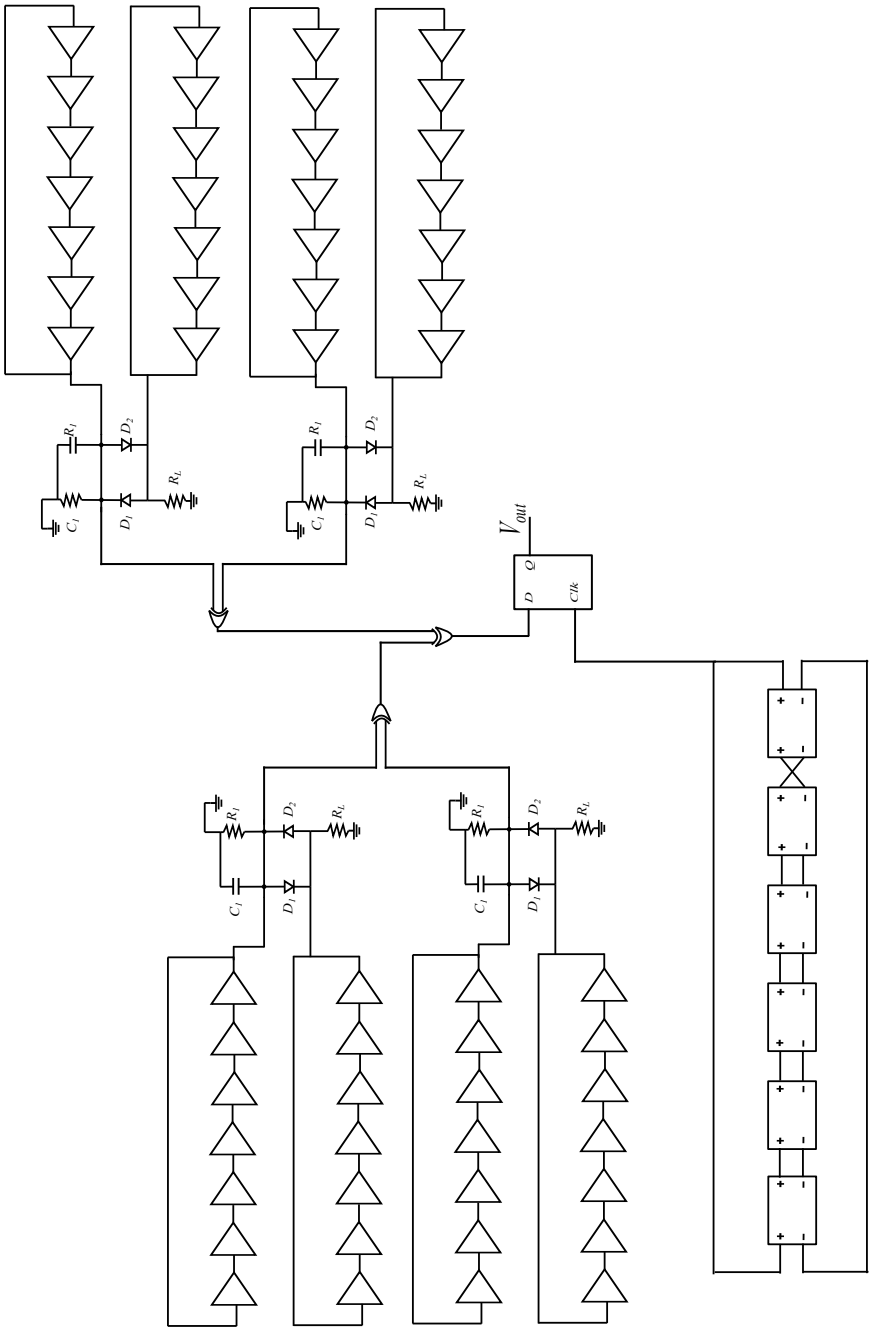

Figura 4.7: Circuito gerador de números aleatórios, com duplicação dos osciladores rápidos, desenvolvido na dissertação.

5

Análise e simulação do circuito gerador de números aleatórios

Apresentados, no capítulo 4, os blocos que integram o circuito RNG concebido na presente dissertação, bem como as suas características, proceder-se-á à análise do trabalho desenvolvido. Desta forma, o presente capítulo está organizado do mesmo modo que o anterior, para uma melhor ponderação de cada um dos constituintes do circuito.

Para ser possível a avaliação do RNG, recorreu-se à ferramenta CADENCE para elaborar as simulações dos circuitos que o constituem, à ferramenta NIST SP 800-22 para se proceder aos testes de aleatoriedade e às equações 4.4 e 4.5, apresentadas no capítulo 4, para o cálculo do consumo e velocidade de transmissão, respetivamente.

5.1 Oscilador diferencial

No âmbito da análise do oscilador diferencial, realizaram-se duas simulações, DC e transiente, para cada um dos osciladores projetados, de forma a obter-se os valores de frequência, de potência consumida, de ruído e formas de onda. As simulações foram efetuadas isoladamente para cada oscilador, utilizando-se seis células de atraso ligadas a uma fonte de alimentação.

Na tabela 5.1 apresentam-se os valores de frequência, potência consumida e área ocupada, obtidos a partir das simulações efetuadas com os valores de L e W presentes na tabela 4.1, podendo-se, desta forma, efetuar uma comparação entre os osciladores.

Tabela 5.1: Valores de potência consumida, em μW, de frequência, em MHz, e área ocupada, em μm^2, obtidos através das simulações transientes dos osciladores diferencias proposto e alternativo.

Oscilador	Consumo (μW)	Frequência (MHz)	Área (μm^2)
Proposto	115,55	26,28	24
Alternativo	439,656	26,3	336

Analisando a tabela 5.1, verificamos que o valor de consumo do oscilador proposto é quatro vezes inferior ao oscilador alternativo para a mesma frequência, provando que o oscilador projetado é melhor visto um dos objetivos ser o baixo consumo. Nas figuras 5.1 e 5.2 pode observar-se o ruído de fase do sinal, em *ps*, dos osciladores diferenciais proposto e alternativo, respetivamente.

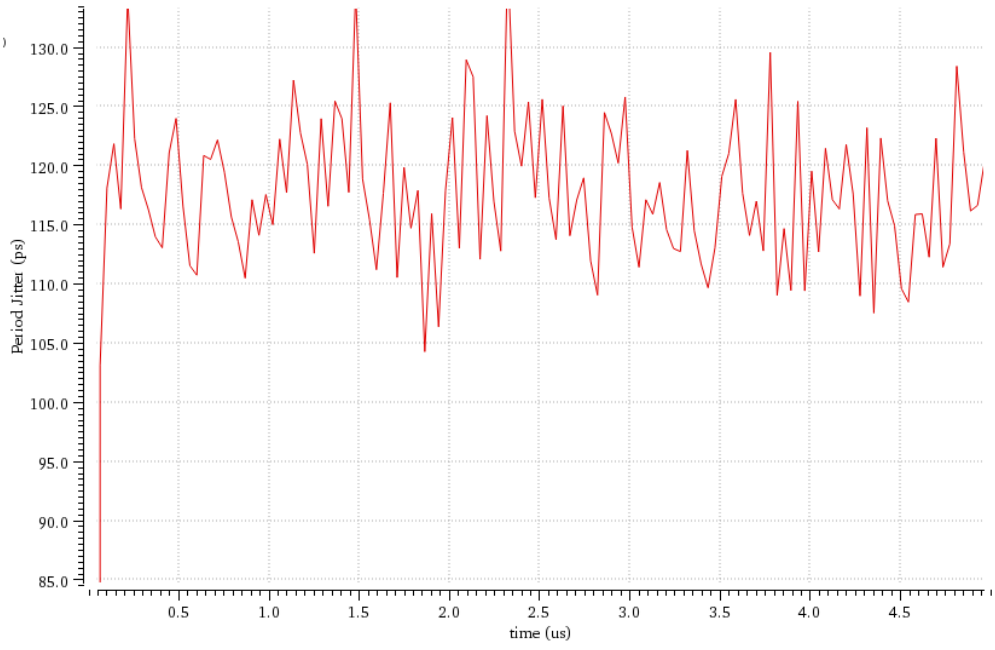

Figura 5.1: *Jitter* do circuito correspondente ao oscilador diferencial proposto

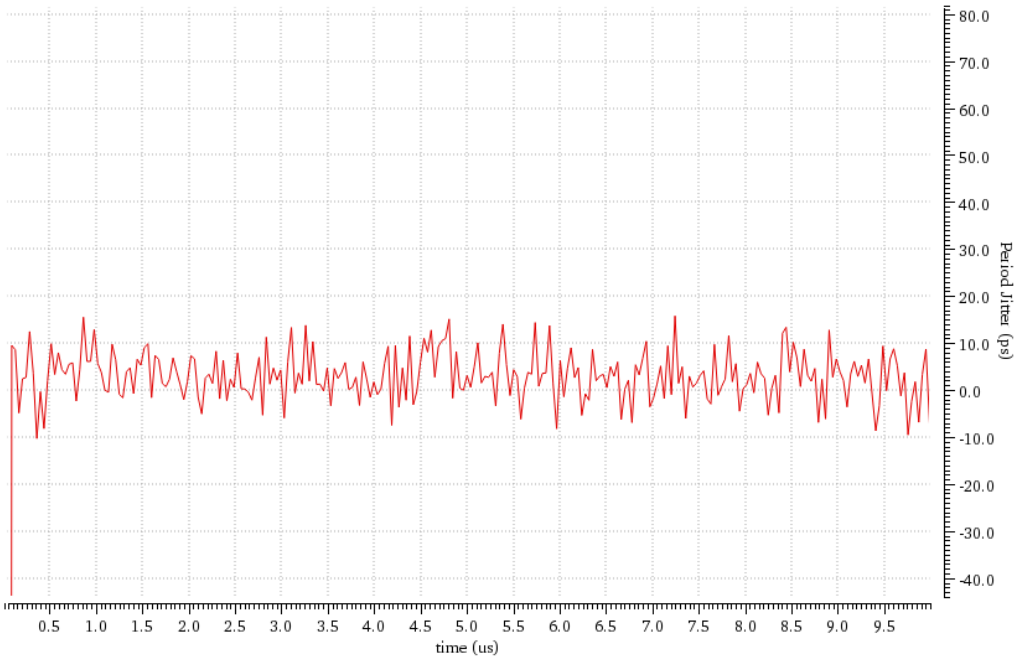

Figura 5.2: *Jitter* do circuito correspondente ao oscilador diferencial alternativo.

Explorando os gráficos acima mencionados, afere-se que o ruído de fase no circuito proposto varia entre os $105ps$ e os $135ps$, valores esses retirados do eixo das ordenadas do gráfico da figura 5.1, e o ruído de fase no circuito alternativo varia entre os $15ps$ e os $-10ps$, valores esses retirados do eixo das ordenadas do gráfico da figura 5.2. Desta forma, verifica-se que o sinal gerado pelo oscilador diferencial proposto nesta dissertação apresenta-se mais ruidoso que o sinal gerado pelo outro oscilador. Analisando os resultados obtidos a partir das simulações, depreende-se que a solução proposta na presente dissertação tem um comportamento próximo ao comportamento ideal, dado que apresenta um valor de potência consumida três vezes menor, permanecendo a frequência dentro dos valores pretendidos, e um valor de ruído superior, quando comparada com o oscilador diferencial alternativo.

5.2 Oscilador de dados rápido

Realizou-se uma simulação na base do tempo para o oscilador de dados projetado, obtendo-se, assim, o valor de frequência, de ruído e formas de onda e uma simulação DC, de forma a obter-se o valor de potência consumida.

Na tabela 5.2 estão presentes os valores de frequência, potência consumida e área ocupada, obtidos a partir das simulações efetuadas com os valores de L e W presentes na

tabela 4.2.

Tabela 5.2: Valores de potência consumida, em μW, de frequência, em MHz, e área ocupada, em μm^2, obtidos através da simulação transiente do oscilador de dados.

Oscilador	Potência Consumida (μW)	Frequência (MHz)	Área prevista (μm^2)
7 células	126,157	502,1	5,25

Analisando os resultados obtidos na tabela 5.2, verificou-se que o oscilador de dados apresenta um baixo valor de potência consumida, permanecendo a frequência dentro dos valores pretendidos, bem como uma área ocupada pelo circuito reduzida.

Nas figuras 5.3 e 5.4 observou-se o sinal de saída e o ruído de fase do sinal, em *ps*, do oscilador de dados *single-ended*, respetivamente.

Examinando o gráfico presente na figura 5.4, verificou-se que o ruído de fase do oscilador de dados varia entre o $1,5ps$ e o $-1ps$, valores esses retirados do eixo das ordenadas do gráfico da figura 5.4. Comparando a gama de valores de ruído do oscilador em discussão com a gama de valores de ruído dos osciladores apresentados na subsecção 4.1.1, constatou-se que a primeira gama mencionada é 30 vezes inferior, não sendo esta situação a ideal, por este oscilador ser o responsável por gerar o ruído do sistema. Desta forma, adicionou-se uma célula de caos a este oscilador, por forma a aumentar o seu ruído de fase.

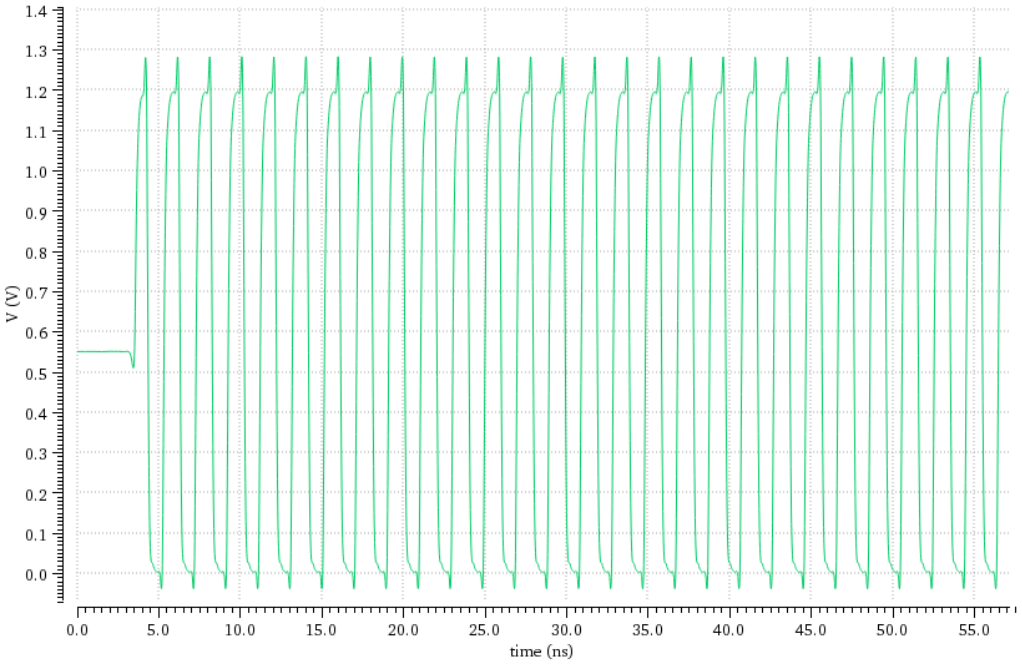

Figura 5.3: Sinal de saída do circuito correspondente ao oscilador de dados *single-ended*.

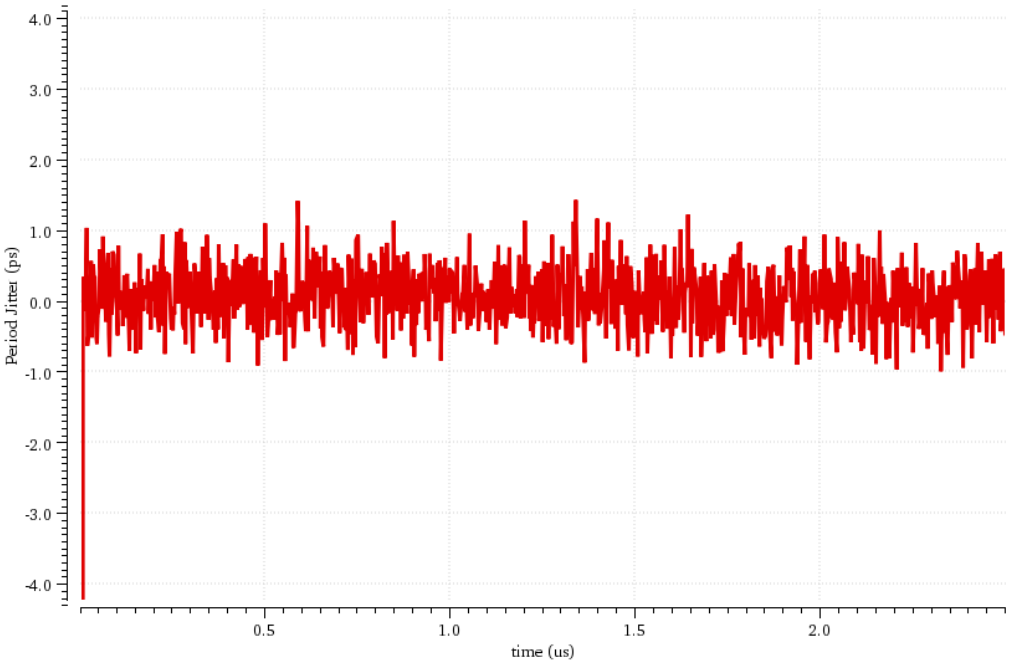

Figura 5.4: Jitter do oscilador em anel *single-ended*.

5.3 Circuito gerador de sinal caótico

Como referido na secção 4.1.3, ao unir a célula de caos aos osciladores de dados, a aleatoriedade do circuito proposto é ampliada, sendo este facto constatado pelo aumento do valor do ruído do sinal à saída dos osciladores enumerados. O mencionado apresenta-se evidenciado no gráfico da figura 5.6, uma vez que a tensão, representada no eixo das ordenadas, apresenta-se mais irregular do que na figura 5.5.

Relativamente à frequência do circuito resultante da ligação da célula de caos aos osciladores previamente mencionados, verificou-se que esta se apresentava reduzida para, aproximadamente, metade.

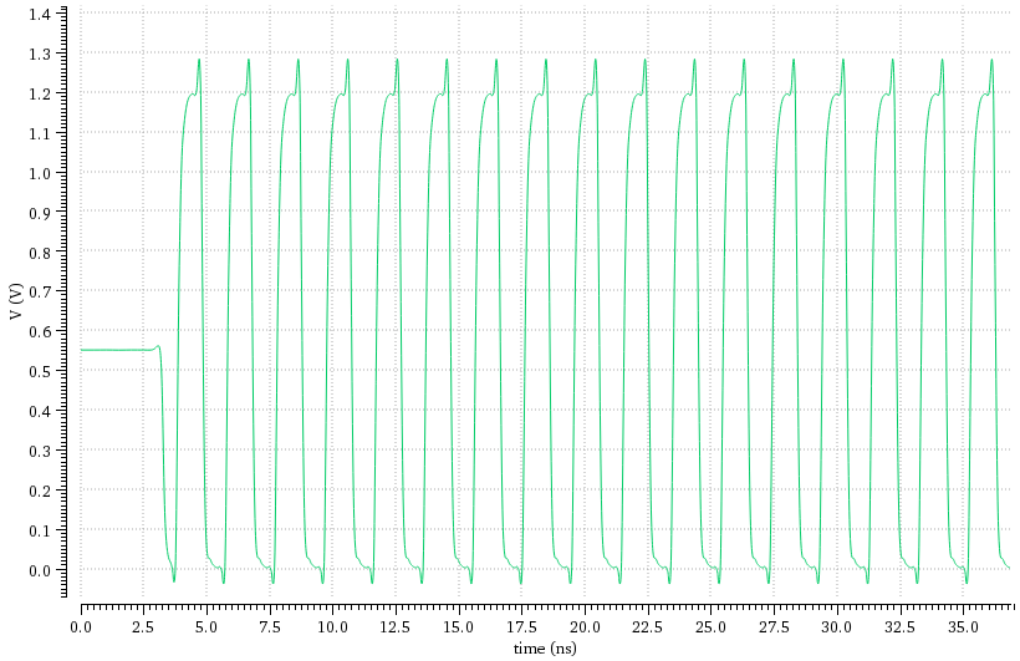

Figura 5.5: Sinal à saída do oscilador de dados, sem a inclusão da célula de caos no mesmo.

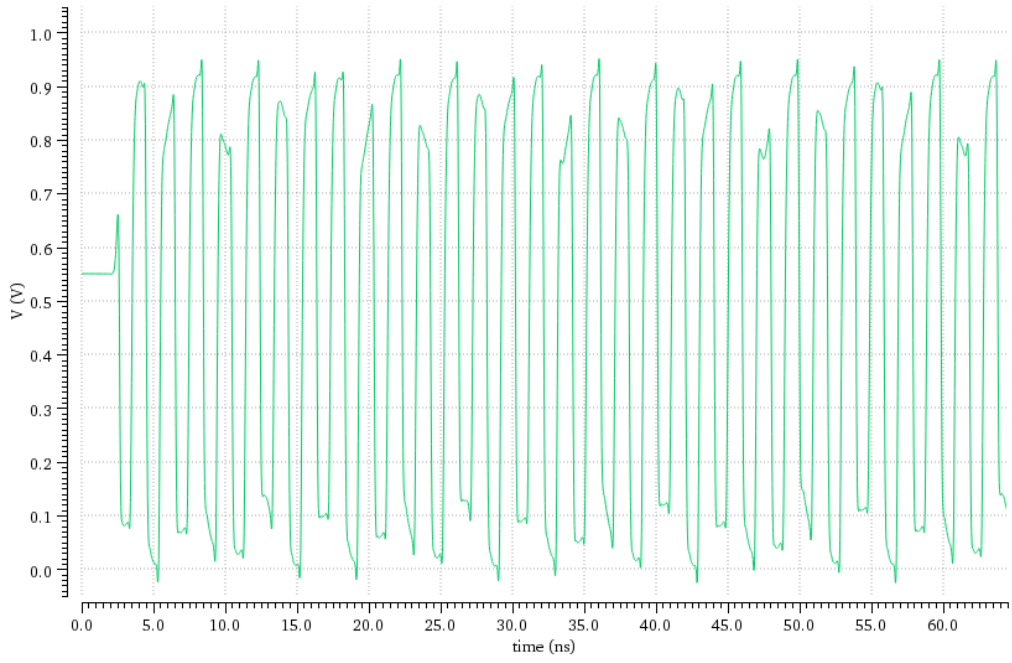

Figura 5.6: Sinal à saída do oscilador de dados, após a inclusão da célula de caos.

5.4 Circuito gerador de números aleatórios desenvolvido na dissertação

Efetuou-se uma simulação na base do tempo de forma a ver o comportamento do circuito, bem como uma simulação DC de forma a avaliar o seu consumo de potência.

Na tabela 5.3 estão presentes os valores de potência consumida, em mW, área ocupada, em μm^2, velocidade de transmissão, em $Mbit/s$, consumo de transmissão, em pJ/bit, frequência do oscilador de dados, em MHz, e frequência do oscilador de *clock*, em MHz, do circuito desenvolvido na dissertação, e esquematizado na figura 4.6, obtidos a partir das simulações mencionadas. Na figura 5.7 encontram-se representadas as formas de onda, obtidas durante $10\mu s$, do circuito anteriormente mencionado, sendo que o gráfico apresenta, no eixo das abcissas, o período de tempo da simulação e, no eixo das ordenadas, a tensão à saída do circuito no ponto temporal respetivo.

Tabela 5.3: Valores de potência consumida, em μW, de frequência, em MHz, e área ocupada, em μm^2, obtidos através da simulação transiente do oscilador de dados.

	Consumo (mW)	Área (μm)	Velocidade de transmissão ($Mbits/s$)	Consumo de transmissão (pJ/bit)	Freq. de dados (MHz)	Freq. de *Clock* (MHz)
Circuito Desenvolvido	1,19	34,5	26	45,77	214,31	26,31

x

Analisando a tabela 5.3, verifica-se um valor reduzido de consumo, podendo também ser observado pelo valor de energia por bit, que se encontra nos 45,77 pJ/bit. O *throughput* encontra-se nos 26 Mbits/s, estando ligado à velocidade do oscilador de amostragem mais lento.

Para se poder analisar o circuito desenvolvido na dissertação, relativamente aos objetivos a alcançar, em comparação com o circuito oscilador com *latch*, por meio de transístores, e com o circuito oscilador sem célula de caos, elaborou-se uma tabela - tabela 5.4 - à semelhança da 5.3 para os valores dos osciladores mencionados.

Tabela 5.4: Valores de potência consumida, área, velocidade de transmissão, consumo de transmissão, frequência de dados e frequência de *clock* obtidos através das simulações efetuadas aos circuitos: circuito gerador de números aleatórios desenvolvido na dissertação; gerador de números aleatórios com oscilador diferencial com *latch* através de transístores; gerador de números aleatórios sem célula de caos.

	Desenvolvido	C/Latch	S/ Célula de Caos
Consumo (mW)	1,19	1,48	1,13
Área (μm^2)	34,5	346,5	34,5
Vel. de transmissão (Mbit/s)	26	25	25
Consumo de transmissão (pJ/bit)	45,77	67,14	45,20
Frequência dados (MHz)	214,31	214,31	509,18
Frequência clock (MHz)	26,31	24,64	26,16

Analisando a tabela 5.4, verificou-se que o circuito arquitetado na dissertação apresenta um consumo, bem como um consumo de transmissão e uma área reduzidos, em comparação com o circuito oscilador com *latch*, por meio de transístores, e uma velocidade de transmissão superior, comparativamente aos outros dois circuitos, tornando-o, desta forma, otimizado relativamente ao pretendido.

Nas figuras 5.8 e 5.9 encontram-se representadas as formas de onda obtidas durante $10\mu s$ dos circuitos oscilador com *latch*, por meio de transístores, e oscilador sem célula de caos. À semelhança da figura 5.7, estes gráficos apresentam no eixo das abcissas o período da simulação e no eixo das ordenadas está representado a tensão à saída do circuito no ponto temporal respetivo.

Observando os três gráficos, verificaram-se algumas diferenças na sua forma de onda. Tanto na figura 5.7 como na figura 5.8 verificou-se uma frequência do sinal relativamente parecida. No entanto, no terceiro gráfico, verificou-se uma frequência de saída cerca de 4 vezes superior. É possível afirmar que a segunda e terceira figura apresentam padrões na representação do sinal, o que vai contra o pretendido.

Podemos então afirmar que o circuito desenvolvido comporta-se melhor do que as variantes apresentadas, sendo visível (à vista desarmada) uma menor quantidade de correlação entre *bits*, tornando-o, desta forma, mais aleatório.

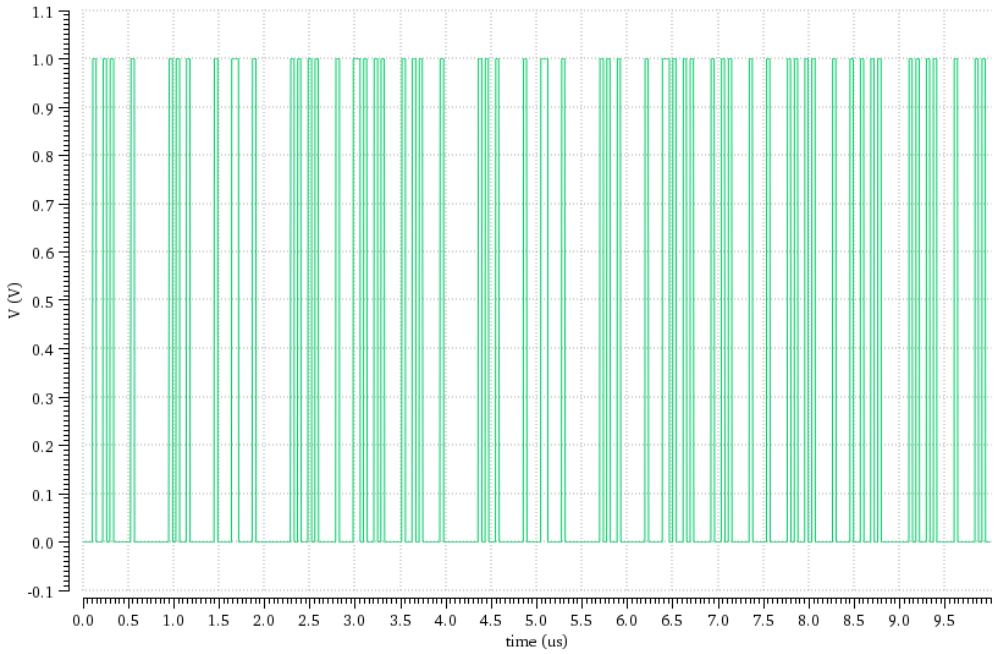

Figura 5.7: Sinal de saída correspondente ao circuito gerador de números aleatórios desenvolvido na dissertação (figura 4.6).

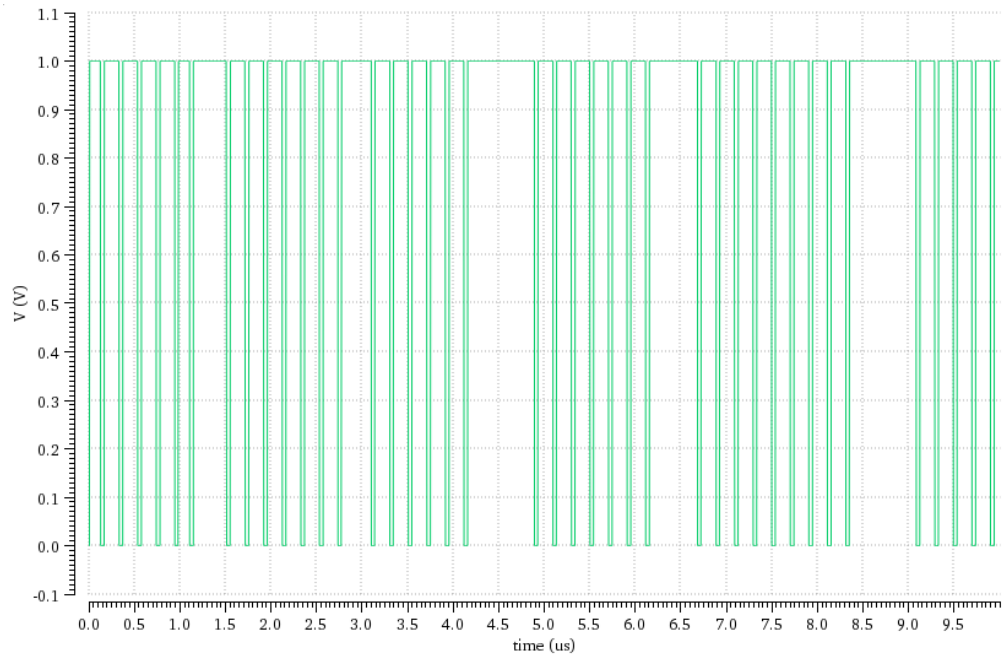

Figura 5.8: Sinal de saída do circuito com clock com latch.

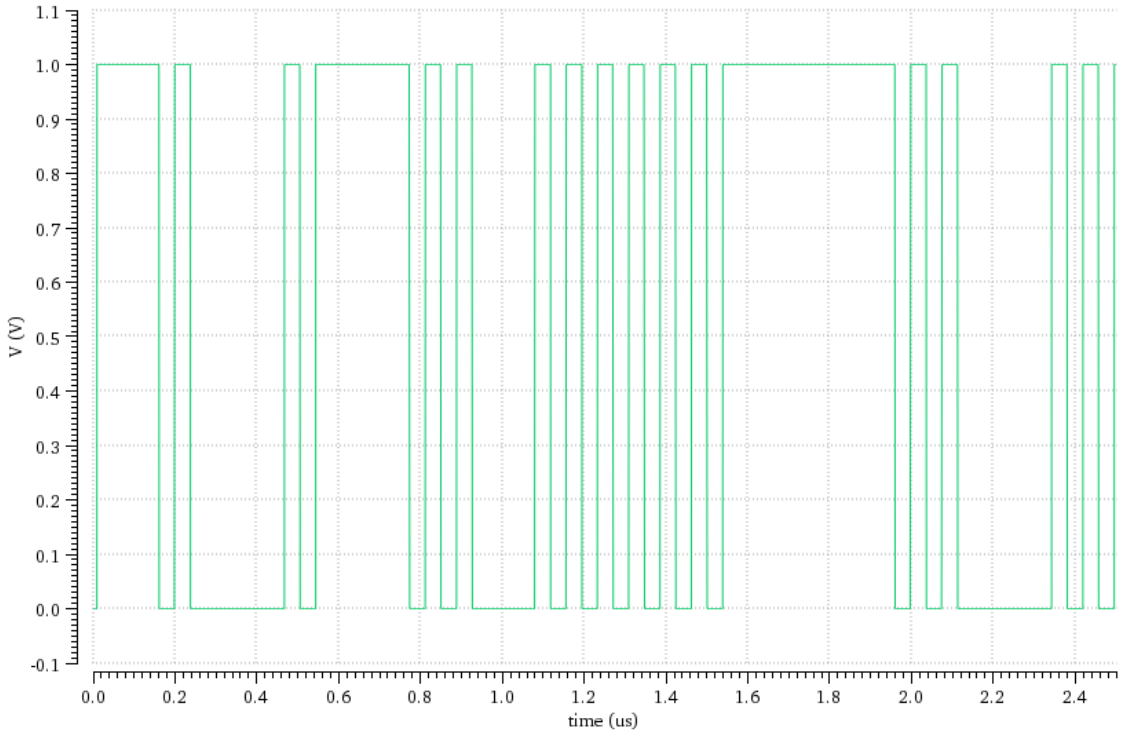

Figura 5.9: Sinal de saída do circuito gerador de números aleatórios sem célula de caos.

5.5 Pós processamento e testes de aleatoridade

Primeiramente, efetuou-se uma simulação, com um período longo, isto é, que contemple um número elevado de ciclos, para se obter múltiplos *bits*. Posto isto, retiraram-se os valores de tempo e amplitude do sinal, presentes, por exemplo, no gráfico 5.7, para um ficheiro do tipo *Comma Separated Values* (.csv) (anexo I), de forma a serem posteriormente processados no programa MATLAB.

Neste programa, para o processamento dos dados, criou-se um algoritmo (anexo II) que selecionasse apenas bits representativos, isto é, gerou-se, através deste, um sinal com uma amostragem inferior à amostragem da simulação, de forma a ter um *bit* por cada ciclo de *clock*. Finalmente, o algoritmo retorna um ficheiro do tipo *Text* (.txt), com os valores a serem analisados pela ferramenta de teste escolhida.

Os testes estatísticos de aleatoriedade dos RNG, selecionados pelos autores consultados para a elaboração da dissertação, estão presentes nas ferramentas de teste NIST, german e DIEHARD. Consequentemente, na presente dissertação, utilizou-se o NIST SP 800-22 para testar o circuito proposto, dado esta ferramenta ser recorrentemente utilizada na literatura.

Para se proceder aos testes de aleatoriedade do circuito proposto na presente dissertação, recorreu-se a uma simulação transitória de 150μs, tendo-se obtido cerca de 20 milhões de pontos. De seguida, construiu-se um *script* em MATLAB para fazer a amostragem dos pontos de acordo com a frequência. A partir desses 20 milhões de pontos retirou-se um ficheiro .txt com 3000 *bits* de modo a ser possível elaborar os testes.

Visto o número de *bits* retirados serem insuficientes para a elaboração dos testes de aleatoriedade a partir da ferramenta AIS 31, apesar da ferramenta NIST SP 800-22 ser, apenas, adequada para PRNG, recorreu-se a esta, dado ser utilizada na literatura.

As simulações que constituem os testes de aleatoriedade, para o circuito desenvolvido na dissertação, recorreram a 3000 pontos.

Tabela 5.5: Valores obtidos através do teste NIST, para o circuito gerador de números aleatórios base - figura 4.6 - desenvolvido na dissertação.

	M	B	m	Mode	pvalue	Result
1.Freq_test	-	-	-	-	2,0656E-06	Fail
2.FreqBlock_test	40	-	-	-	0,5011	Pass
3.Runs_test*	-	-	-	-	-	Fail
4.LongOnesBlock_test	-	-	-	-	6,6020E-01	Pass
7.Non_Overlapping_test	40	000111010	-	-	0,9848	Pass
10.Serial_ test	-	-	3	-	4,244E-45	Fail
	-	-	-	-	4,2720E-01	Pass
11.Approx_Entropy_test	-	-	3	-	0	Fail
12.CuSum_test	-	-	-	Forward	2,6159E-06	Fail
	-	-	-	Backward	2,6159E-06	Fail

Analisando a tabela 5.5, verificou-se que o circuito RNG base (figura 4.6) apresenta um $p-value$ superior a 0,01 nos testes: Teste de Frequência dentro de uma *bitstream*; Teste da maior sequência de "1"numa *bitstream*; Teste de correspondência de modelos sobrepostos e Teste de Somas Cumulativas, sendo que este último apresenta-se parcialmente superior ao valor pretendido. Desta forma, considerou-se que o circuito RNG base passou em três testes de aleatoriedade da ferramenta enumerada e passou parcialmente a um.

Para demonstrar a robustez do circuito proposto, procedeu-se à comparação dos seus testes de aleatoriedade relativamente a um circuito com um *clock* convencional - por forma a verificar-se o impacto do comportamento dos diferentes *clock's* no desempenho do oscilador - e um circuito sem gerador caótico - no sentido de se avaliar o impacto do ruído caótico na aleatoriedade do circuito.

As simulações que constituem os testes de aleatoriedade recorreram a 3000 pontos, no caso do circuito com um *clock* convencional, e 1600 pontos, no caso do circuito sem gerador caótico.

Tabela 5.6: Valores obtidos através do teste NIST, para o circuito com *clock* convencional.

	M	B	m	Mode	pvalue	Result
1.Freq_test	-	-	-	-	2.3502E-310	Fail
2.FreqBlock_test	40	-	-	-	0	Fail
3.Runs_test*	-	-	-	-	-	Fail
4.LongOnesBlock_test	-	-	-	-	8,0509E-07	Fail
7.Non_Overlapping_test	40	000111010	-	-	1	Pass
10.Serial_test	-	-	3	-	0	Fail
	-	-	-	-	2,0000E-03	Fail
11.Approx_Entropy_test	-	-	3	-	0	Fail
12.CuSum_test	-	-	-	Forward	4.7005E-310	Fail
	-	-	-	Backward	2.3622E-310	Fail

Tabela 5.7: Valores obtidos através do teste NIST, para o circuito sem circuito caótico

	M	B	m	Mode	pvalue	Result
1.Freq_test	-	-	-	-	8,7976E-108	Fail
2.FreqBlock_test	30	-	-	-	0	Fail
3.Runs_test*	-	-	-	-	-	Fail
4.LongOnesBlock_test	-	-	-	-	1,2000E-03	Fail
7.Non_Overlapping_test	30	000111010	-	-	1	Pass
10.Serial_test	-	-	3	-	0	Fail
	-	-	-	-	1,8309E-109	Fail
11.Approx_Entropy_test	-	-	3	-	0	Fail
12.CuSum_test	-	-	-	Forward	1,7595E-107	Fail
	-	-	-	Backward	1,7595E-107	Fail

Analisando a tabela 5.6, verificou-se que o circuito com um *clock* convencional apresenta um $p - value$ superior a 0,01 no teste: Teste de correspondência de modelos sobrepostos. Desta forma, considera-se que este circuito passou em, apenas, um teste de aleatoriedade da ferramenta NIST SP 800-22. Assim, constatou-se uma melhoria significativa da aleatoriedade do oscilador quando se recorreu ao *clock* proposto na dissertação.

Examinando a tabela 5.7, aferiu-se que o circuito sem a célula de caos apresenta o mesmo comportamento que o circuito com um *clock* convencional, passando, unicamente, no Teste de correspondência de modelos sobrepostos. Desta forma, apurou-se que a presença de um gerador caótico tem um impacto significativo na aleatoriedade do circuito.

Por forma a provar-se que existe um número ótimo de pares oscilador rápido/circuito caótico, com o objetivo de tornar o circuito proposto num TRNG, efetuou-se o teste de aleatoriedade recorrendo à ferramenta previamente enumerada, utilizando 3000 pontos, para o circuito gerador de números aleatórios com dois pares oscilador rápido/circuito caótico, representado na figura 4.7.

Tabela 5.8: Valores obtidos através do teste NIST, para o circuito com dois osciladores rápidos extra (figura 4.7.)

	M	B	m	Mode	pvalue	Result
1.Freq_test	-	-	-	-	0,7175	Pass
2.FreqBlock_test	40	-	-	-	2,356E-05	Fail
3.Runs_test*	-	-	-	-	-	Fail
4.LongOnesBlock_test	-	-	-	-	0,02	Pass
7.Non_Overlapping_test	40	000111010	-	-	0,9995	Pass
10.Serial_test	-	-	3	-	0,0094	Fail
	-	-	-	-	0,6749	Fail
11.Approx_Entropy_test	-	-	3	-	0,0109	Pass
12.CuSum_test	-	-	-	Forward	0,0046	Fail
	-	-	-	Backward	0,0013	Fail

Observando a tabela 5.8, constatou-se que o circuito anteriormente mencionado apresenta um $p-value$ superior a 0,01 nos testes: Teste de Frequência; Teste de maior sequência de "1"numa *bitstream*; Teste de correspondência de modelos não sobrepostos; Teste de Entropia Aproximada. Desta forma, considerou-se que o circuito passa nos testes 1, 4, 7 e 11 e, parcialmente, no teste 10. Com estes dados, pode-se aferir que existirá um número ótimo de pares oscilador rápido/circuito caótico, por forma a tornar o circuito desenvolvido na dissertação num gerador de números verdadeiramente aleatórios.

Posteriormente, visto que para avaliar este trabalho ao nível da aleatoriedade é necessário enviar para fábrica, chegou-se à conclusão que, ao colocar os osciladores rápidos no mesmo *chip*, estes poderiam apresentar a sua frequência combinada, utilizando o substrato como meio de correlação. Concluiu-se então que é necessário alterar as frequências entre os osciladores, não as afastando muito para que também não fique correlacionada com o oscilador de relógio. Decidiu-se então, utilizando o mesmo circuito anterior, composto por quatro pares de osciladores rápidos, alimentar cada um com uma tensão diferente. Sendo assim, ficaram quatro pares de osciladores alimentados por 0,8V, 0,9V, 1,0V e 1,1V. Esta diferença na alimentação dos osciladores provocou uma alteração na sua frequência, evitando assim a correlação no *chip*. De seguida, efetuou-se os testes de aleatoriedade apenas com 200 bits, apresentando-se o resultado na tabela seguinte:

Analisando a tabela 5.9, aferiu-se que através, apenas, da alteração das tensões de alimentação dos pares de osciladores rápidos, alcançou-se a aprovação de aleatoriedade nos testes do NIST efetuados. Verificou-se, também, que são, somente, necessários quatro pares de osciladores rápidos para atingir a aleatoriedade do circuito integrado desenvolvido, conservando assim a reduzida área do mesmo.

Por forma a sumarizar o trabalho desenvolvido, apresenta-se, na tabela 5.10, os resultados obtidos para os quinze testes presentes na ferramenta NIST SP 800-22, relativamente aos diferentes circuitos em análise, nomeadamente o circuito do oscilador diferencial com *clock* alternativo, ou seja, com *latch*, a partir de transístores, circuito RNG sem célula

Tabela 5.9: Valores obtidos através do teste NIST, para o circuito representado na figura 4.7, tendo os pares de osciladores rápidos diferentes tensões de alimentação.

	M	B	m	Mode	pvalue	Result
1.Freq_test	-	-	-	-	0,3317	Pass
2.FreqBlock_test	25	-	-	-	0,8114	Pass
3.Runs_test*	-	-	-	-	0,0606	Pass
4.LongOnesBlock_test	-	-	-	-	0,1671	Pass
7.Non_Overlapping_test	15	000111010	-	-	1,0000	Pass
11.Serial_test	-	-	3	-	0,1050	Pass
	-	-	-	-	0,1561	Pass
12.Approx_Entropy_test	-	-	3	-	0,1167	Pass
13.CuSum_test	-	-	-	Forward	0,2907	Pass
	-	-	-	Backward	0,6562	Pass

geradora de caos, circuito gerador de números aleatórios base, desenvolvido na dissertação e presente na figura 4.6, circuito RNG com dois pares de osciladores rápidos extra, desenvolvido na dissertação e presente na figura 4.7, e circuito RNG com tensões de alimentação diferentes para cada par de osciladores de dados rápidos. Representados a verde, encontram-se os testes de aleatoriedade passados pelos circuitos; a vermelho encontram-se os chumbados pelos circuitos e, representados a cinzento encontram-se os testes não efetuados.

Tabela 5.10: Resultados obtidos nos quinze testes presentes na ferramenta NIST SP 800-22, relativamente aos circuitos: circuito do oscilador diferencial com *clock* alternativo, ou seja, com *latch*, a partir de transístores, circuito RNG sem célula geradora de caos, circuito gerador de números aleatórios base, desenvolvido na dissertação e presente na figura 4.6, circuito RNG com dois pares de osciladores rápidos extra, desenvolvido na dissertação e presente na figura 4.7, e circuito RNG com tensões de alimentação diferentes para cada par de osciladores de dados rápidos.

	Clock Alternativo	Sem Circuito Caótico	Proposto Simples	Par de Osciladores Rápidos Extra	Tensões de Alimentação Alteradas
1.Teste de Frequência	2,3500E-310	8,7976E-108	2,0656E-06	0,7175	0,3317
2. Teste de Frequência dentro de uma *bitstream*	0	0	0,5011	2,3560E-06	0,8114
3. Teste de *Runs*	-	-	-	-	0,0606
4. Teste da maior sequência de "1"numa *bitstream*	8,0509E-07	1,2000E-03	6,6020E-01	0,02	0,1671
5. Teste de classificação de matriz binária	-	-	-	-	-
6. Teste da transformada de Fourier discreta (espectral)	-	-	-	-	-
7. Teste de correspondência de modelos não sobrepostos	1,0000	1,0000	0,9848	0,9995	1,0000
8. Teste "Estatístico Universal" de Maurer	-	-	-	-	-
9. Teste de Complexidade Linear	-	-	-	-	-
10. Teste de Série	0 2,0000E-03	0 1,8309E-109	4,2720E-01 4,2440E-45	0,0094 0,6749	0,1050 0,1561
11. Teste de Entropia Aproximada	0	0	0	0,0109	0,1167
12. Teste de Somas Cumulativas	4,7005E-310 2,3622E-310	1,7595E-107 1,7595E-107	2,6159E-06 2,6159E-06	0,0046 0,0013	0,2907 0,6562
13. Teste de Excursões Aleatórias	-	-	-	-	

6

Conclusões e Trabalho Futuro

6.1 Conclusões

Tendo em conta o trabalho desenvolvido na presente dissertação, concluiu-se que o circuito concebido na mesma atingiu, na sua maioria, os objetivos inicialmente estabelecidos. Assim sendo, o circuito apresenta-se com:

- Consumo reduzido, sendo, desta forma, um possível modelo para soluções portáteis;
- Elevado grau de escalabilidade a nível de número de circuitos, podendo-se adicionar alguns dos componentes já existentes no circuito, nomeadamente, conjuntos de osciladores rápidos e circuitos caóticos ligados por uma XOR, aumentado, assim, as suas propriedades aleatórias;
- Alta escalabilidade da tecnologia, podendo funcionar com arquiteturas de semicondutores diferentes, podendo ser reduzido o seu tamanho ao reduzir a tecnologia;
- Área reduzida, permitindo, assim, que o circuito proposto seja produzido num circuito integrado, incrementando, desta forma, o seu grau de portabilidade e de integração em sistemas;
- Reduzido custo, sendo projetado com tecnologia CMOS;
- Aprovação em alguns testes do NIST.

Apesar dos objetivos previamente mencionados terem sido satisfatoriamente alcançados, a aleatoriedade estatística do circuito não foi totalmente comprovada. O sucedido resultou das limitações inerentes às ferramentas utilizadas, bem como da complexidade das simulações, empregando, assim, muitos recursos e tornando-as muito demoradas. Desta forma, foi impossível obter o número de bits necessários para efetuar todos os testes que compõem a ferramenta NIST SP 800-22.

Consequentemente, o circuito projetado na dissertação foi projetado para ser um TRNG, mas, devido às restrições das simulações, a certificação proveniente dos testes de aleatoriedade não foi obtida, podendo-se prever que o sistema tem potencial para ser um TRNG, precisando de ser fisicamente testado para obter a certificação.

No entanto, o circuito proposto passou em todos os testes efetuados e provou-se a possibilidade de aumentar a aleatoriedade do mesmo com a adição de pares de osciladores rápidos a circuitos caóticos, mantendo as frequências dos mesmos distintas.

6.2 Trabalho Futuro

O trabalho desenvolvido na dissertação constitui um estudo prévio para a produção de um TRNG integrado em CMOS. Deste modo, como forma de seguimento da presente dissertação, propõem-se as seguintes tarefas:

- Estudar a possibilidade da integração dos osciladores rápidos numa FPGA;
- Verificar se o número de osciladores rápidos incorporados no circuito proposto na dissertação é suficiente para o circuito ser considerado um TRNG, obtendo os *bits* necessários para correr todos os testes. Caso se verifique que este número não é suficiente, encontrar o número mínimo de osciladores rápidos a integrar de forma a que o circuito passe em todos os testes e obtenha a certificação de aleatoriedade;
- Produção de um circuito integrado, por forma a comparar com o circuito simulado e, consequentemente, verificar se o seu comportamento é mais ou menos aleatório;
- Verificar a fiabilidade do circuito proposto como gerador de sinais para uma PUF.

Os osciladores de dados rápidos projetados nesta dissertação estão preparados para serem incluídos num circuito integrado, mas, para aumentar a sua escalabilidade, é importante que se verifique a possibilidade de se obter estes osciladores através de uma FPGA, podendo separar-se os osciladores de dados do restante circuito, produzindo apenas o circuito gerador de sinal caótico e o oscilador lento, ligando à FPGA para completar o circuito.

A escalabilidade do sistema desenvolvido na presente dissertação permite aumentar a sua aleatoriedade, podendo-se aumentar o número de osciladores rápidos. Portanto é importante que se conheçam os limites mínimos deste trabalho como TRNG e, consequentemente, é imperativo que, depois de se realizarem todos os testes de aleatoriedade e se verificar que apenas com dois circuitos de caos o circuito não passa nos testes, descobrir-se o número mínimo de blocos necessários para os passar.

Os circuitos aleatórios são difíceis de simular devido ao seu comportamento errático e à sua variação de produção, isto é, um circuito integrado quando é produzido, em

diferentes zonas da bolacha de silício vai apresentar variações no seu comportamento e, portanto, é importante que se produza o *chip* para se descobrir verdadeiramente o seu funcionamento físico.

Este circuito poderá ser usado apenas como TRNG mas, no futuro, poderia integrar uma PUF e, portanto, devia-se verificar a possibilidade deste gerador ser integrado num circuito criptográfico acompanhado de uma PUF.

Bibliografia

[1] Y. Cao et al. "A Low-Power Hybrid RO PUF With Improved Thermal Stability for Lightweight Applications". Em: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34.7 (2015), pp. 1143–1147. DOI: 10.1109/TCAD.2015.2424955 (ver p. 2).

[2] J. F. Dooley. *History of Cryptography and Cryptanalysis*. Springer International Publishing, 2018. ISBN: 978-3-319-90442-9. DOI: 10.1007/978-3-319-90443-6 (ver p. 2).

[3] J. M. Kizza. *Computer Network Security and Cyber Ethics*. McFarland e Company, Inc, 2014. ISBN: 978-0-7864-9392-0 (ver p. 2).

[4] S. Haykin. *Communication Systems*. 4th. New York: Wiley, 2000, p. 872. ISBN: 10: 0471178691 (ver pp. 2, 6, 7).

[5] M. Gençoğlu. "Importance of Cryptography in Information Security". Em: (mar. de 2019). DOI: 10.9790/0661-2101026568 (ver p. 2).

[6] M. Stipčević e Ç. K. Koç. *Open Problems in Mathematics and Computational Science*. Ed. por Ç. K. Koç. Cham: Springer International Publishing, 2014, pp. 275–315. ISBN: 978-3-319-10683-0. DOI: 10.1007/978-3-319-10683-0_12. URL: https://doi.org/10.1007/978-3-319-10683-0_12 (ver pp. 3, 12, 13, 22).

[7] V. Tuzlukov. *Signal Processing Noise*. Jan. de 2002, pp. 1–663. ISBN: 9781315220147. DOI: 10.1201/9781315220147 (ver p. 6).

[8] M. Mandal e B. C. Sarkar. "Ring oscillators: Characteristics and applications". Em: (2010) (ver pp. 6, 15–18).

[9] D. V. Perepelitsa. "Johnson Noise and Shot Noise". Em: *Analysis* 2 (2006), pp. 2–5 (ver pp. 7, 8).

[10] A. Hajimiri, S. Limotyrakis e T. Lee. "Jitter and phase noise in ring oscillators". Em: *IEEE Journal of Solid-State Circuits* 34.6 (1999), pp. 790–804. DOI: 10.1109/4.766813 (ver p. 8).

[11] F. Herzel e B. Razavi. "A study of oscillator jitter due to supply and substrate noise". Em: *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 46.1 (1999), pp. 56–62. DOI: 10.1109/82.749085 (ver pp. 8, 9).

[12] C. M. Q. Pereira. "Processamento de Dados Electroencefalográficos - aplicações à epilepsia." Tese de Doutoramento. Universidade de Lisboa, 1998 (ver pp. 9, 10).

[13] S. Ergün. "Cryptanalysis of a Chaotic Ring Oscillator Based Random Number Generator". Em: *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. 2018, pp. 1498–1501. DOI: 10.1109/TrustCom/BigDataSE.2018.00211 (ver pp. 10, 11, 26, 30, 36).

[14] İ. Taştan e S. Ergün. "A Robust Random Number Generator Based on Chaotic Ring Oscillators". Em: *2019 17th IEEE International New Circuits and Systems Conference (NEWCAS)*. 2019, pp. 1–4. DOI: 10.1109/NEWCAS44328.2019.8961295 (ver pp. 10, 11, 26, 27, 30).

[15] İ. Çiçek e G. Dündar. "A chaos based integrated jitter booster circuit for true random number generators". Em: *2013 European Conference on Circuit Theory and Design (ECCTD)*. 2013, pp. 1–4. DOI: 10.1109/ECCTD.2013.6662257 (ver p. 11).

[16] P. A. W. Lewis, A. S. Goodman e J. M. Miller. "A pseudo-random number generator for the System/360". Em: *IBM Systems Journal* 8.2 (1969), pp. 136–146. DOI: 10.1147/sj.82.0136 (ver p. 12).

[17] G. King. *Vibrations and Waves*. Manchester Physics Series. Wiley, 2013. ISBN: 9781118681787. URL: https://books.google.pt/books?id=Biw2lw01RCYC (ver p. 14).

[18] U. L. Rohde. "Oscillator basics and low-noise techniques for microwave oscillators and VCOs". Em: *Proc. European GaAs and Other Semiconductors Application Symp.(now EuMIC)*. 2000 (ver p. 15).

[19] L. E. Frenzel. "Chapter 4 - Electronic Circuits: Linear/Analog: The Building Blocks of Electronic Equipment". Em: *Electronics Explained (Second Edition)*. Ed. por L. E. Frenzel. Second Edition. Newnes, 2018, pp. 63–102. ISBN: 978-0-12-811641-8. DOI: https://doi.org/10.1016/B978-0-12-811641-8.00004-7. URL: https://www.sciencedirect.com/science/article/pii/B9780128116418000047 (ver p. 15).

[20] L. B. Oliveira et al. *Analysis and Design of Quadrature Oscillators*. Dordrecht: Springer Netherlands, 2008. ISBN: 978-1-4020-8515-4. DOI: 10.1007/978-1-4020-8516-1. URL: http://link.springer.com/10.1007/978-1-4020-8516-1 (ver pp. 15, 16).

[21] A. Wild, G. T. Becker e T. Güneysu. "A fair and comprehensive large-scale analysis of oscillation-based PUFs for FPGAs". Em: *2017 27th International Conference on Field Programmable Logic and Applications (FPL)*. 2017, pp. 1–7. DOI: 10.23919/FPL.2017.8056795 (ver p. 19).

[22] A. Babaei e G. Schiele. "Physical unclonable functions in the internet of things: State of the art and open challenges". Em: *Sensors* 19.14 (2019), p. 3208 (ver p. 19).

[23] B. Halak, M. Zwolinski e M. S. Mispan. "Overview of PUF-based hardware security solutions for the internet of things". Em: *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*. 2016, pp. 1–4. DOI: 10.1109/MWSCAS.2016.7870046 (ver pp. 19, 20).

[24] C. Martínez-Gómez e I. Baturone. "Calibration of Ring Oscillator PUF and TRNG". Em: *2020 European Conference on Circuit Theory and Design (ECCTD)*. 2020, pp. 1–4. DOI: 10.1109/ECCTD49232.2020.9218444 (ver p. 19).

[25] S. Stanzione, D. Puntin e G. Iannaccone. "CMOS Silicon Physical Unclonable Functions Based on Intrinsic Process Variability". Em: *IEEE Journal of Solid-State Circuits* 46.6 (2011), pp. 1456–1463. DOI: 10.1109/JSSC.2011.2120650 (ver p. 20).

[26] Y. Su, J. Holleman e B. P. Otis. "A Digital 1.6 pJ/bit Chip Identification Circuit Using Process Variations". Em: *IEEE Journal of Solid-State Circuits* 43.1 (2008), pp. 69–77. DOI: 10.1109/JSSC.2007.910961 (ver p. 20).

[27] C.-E. Yin e G. Qu. "Temperature-aware cooperative ring oscillator PUF". Em: *2009 IEEE International Workshop on Hardware-Oriented Security and Trust*. 2009, pp. 36–42. DOI: 10.1109/HST.2009.5225055 (ver p. 20).

[28] S. S. Mansouri e E. Dubrova. "Ring oscillator physical unclonable function with multi level supply voltages". Em: *2012 IEEE 30th International Conference on Computer Design (ICCD)*. 2012, pp. 520–521. DOI: 10.1109/ICCD.2012.6378703 (ver pp. 20, 21).

[29] L. Bassham et al. *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. en. 2010-09-16 de 2010. URL: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=906762 (ver p. 22).

[30] C. Petrie e J. Connelly. "A noise-based IC random number generator for applications in cryptography". Em: *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 47.5 (2000), pp. 615–621. DOI: 10.1109/81.847868 (ver pp. 25, 30).

[31] C. Petrie e J. Connelly. "Modeling and simulation of oscillator-based random number generators". Em: *1996 IEEE International Symposium on Circuits and Systems. Circuits and Systems Connecting the World. ISCAS 96*. Vol. 4. 1996, 324–327 vol.4. DOI: 10.1109/ISCAS.1996.541967 (ver p. 28).

[32] P. Malik. "A Light-based Interpretation of Schrodinger's Wave Equation and Heisenberg's Uncertainty Principle with Implications on Quantum Computation". Em: *2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*. 2021, pp. 1–6. DOI: 10.1109/IEMTRONICS52119.2021.9422517 (ver p. 29).

I

Anexo 1 - Código MATLAB

```
sig_raw=readtable ('Bits_50u.csv');
sig_treated=table2array(sig_raw);
clock_raw=readtable ('clk_50u.csv');
clock_treated=table2array(clock_raw);
figure (1)
final_bits(:,1) = sig_treated(:,1);
final_bits(:,2)=0;
j=1;
n=1;
f=1;

for i=1:1:size(clock_treated)
    if clock_treated(i,2) >= 0.6
        clock_treated(i,2)=1;
    else
        clock_treated(i,2)=0;
    end
end

for i=1:1:size(sig_treated)
    if sig_treated(i,2) >= 0.6
        sig_treated(i,2)=1;
    else
        sig_treated(i,2)=0;
    end
end
for i=1:1:size(clock_treated)
    if clock_treated(i,2) == 1
        bits(j,1)=sig_treated(i,2);
        j=j+1;
    end
end

n=1;
for i=1:1:size(bits)
```

```
36      if i >= 6523
37          bits_n(n,1)=bits(i,1);
38          n=n+1;
39      end
40  end
41  plot(bits)
42  figure(2)
43  plot(bits_n)
44  n=1;
45  for i=1:1:size(bits_n)
46      if i== n*13046
47          bits_f(n,1)=bits_n(i,1);
48          n=n+1;
49      end
50  end
51  dlmwrite('bits.txt',bits_f','delimiter','');
```

II

Anexo 2 - Código MATLAB

```
clear
clc

signalA = readtable('A_data&clk.csv');
signal=table2array(signalA(:,2));
j=1
for  i = 1:6747:size(signal)-1
    bitsA(j,1) = signal(i,1);
    j=j+1;
end
figure(1)
plot(bitsA)

signalB = readtable('B_data&clk.csv');
signal2=table2array(signalB(:,2));
j=1
for  i = 1:6747:size(signal2)-1
    bitsB(j,1) = signal2(i,1);
    j=j+1;
end
figure(2)
plot(bitsB)

signalC = readtable('C_data&clk.csv');
signal3=table2array(signalC(:,2));
j=1
for  i = 1:6747:size(signal3)-1
    bitsC(j,1) = signal3(i,1);
    j=j+1;
end
figure(3)
plot(bitsC)

signalD = readtable('data_new_4_cells.csv');
signal4=table2array(signalD(:,2));
```

```
36 j=1;
37 for  i = 1:6747:size(signal4)-1
38     bitsD(j,1) = signal4(i,1);
39     j=j+1;
40 end
41 figure(4)
42 plot(bitsD)
43
44
45 dlmwrite('bitsA.txt',bitsA','delimiter','');
46 dlmwrite('bitsB.txt',bitsB','delimiter','');
47 dlmwrite('bitsC.txt',bitsC','delimiter','');
48 dlmwrite('bitsD.txt',bitsD','delimiter',';');
```