GERADOR DE NÚMEROS ALEATÓRIOS UTILIZANDO PORTAS LÓGICAS EM PLDS

Otávio Terra Roque

Maximiliam Luppe

Escola de Engenharia de São Carlos - USP

otaviotroque@usp.br

Objetivos

O objetivo deste trabalho é desenvolver um gerador de números verdadeiramente aleatórios(TRNG, do inglês, True Random Number Generator) implementado em um dispositivo reconfigurável.

O TRNG foi desenvolvido utilizando de lógica digital baseando-se no trabalho Analysis of a Circuit Primitive for the Reliable Design of Digital Nonlinear Oscillators [1], desta forma utilizaremos conforme proposto osciladores em anel, portas XOR e Flip-Flops tipo D (FFDs) para obtermos o comportamento desejado pelo circuito

Dispositivos reconfiguráveis a cada ano que passa vem sendo mais utilizados para as mais diversas aplicações, devido a sua praticidade de utilização e a possibilidade de reutilização do hardware quando necessária a modificação de sua aplicação.

O dispositivo reconfigurável que será utilizado neste trabalho é uma FPGA(Field-programmable gate array). As FPGAs foram lançadas para o público no ano de 1985 e é um circuito integrado que tem como objetivo que o consumidor que possui esse dispositivo possa configurá-lo de acordo com suas necessidades.

Métodos e Procedimentos

Para a criação do circuito foi seguida a topologia proposta no artigo High Speed True Random Number Generator Based on FPGA [2] onde uma estrutura com 8 osciladores em anel tem suas saídas ligadas a FFDs e as saídas de cada FFD é conectada em pares a portas XOR, em uma estrutura de semelhante a uma árvore binária, a figura 1 mostra como o circuito foi montado.

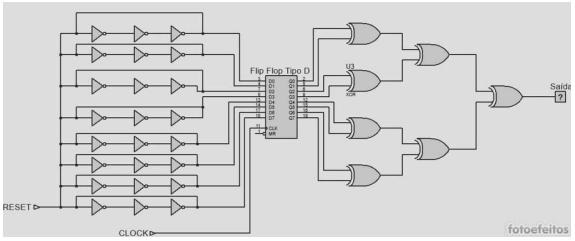

Figura 1: Circuito TRNG

Na figura 1 é representado o circuito do gerador, onde cada entrada Dn representa um FFD e cada saída Qn representa a saída de seu respectivo FFD, ou seja, Q0 é a saída da entrada D0 e assim por diante. Também podemos notar que existem 2 entradas de sinal, a entrada CLOCK e a entrada RESET, a entrada CLOCK é responsável pelo clock dos FFDs e a entrada RESET usada para resetar o circuito. Como saída temos apenas o label Saída e como o nome já diz, representa a saída do circuito.

Após a implementação na FPGA foi necessário testar se os valores gerados são aleatórios ou não, dessa forma, foi necessário a criação de um software que a partir dos dados coletados do gerador pudesse avaliar como o gerador se comporta para que então fosse possível validar o gerador como realmente aleatório ou não. Para isso foi utilizado como referência o artigo do NIST[3] como base, dessa forma utilizando a linguagem de programação Python, foi criado uma série de 14 funções que avaliam a aleatoriedade do circuito.

Resultados

Após a aquisição dos dados foi apresentado para o software de teste os bits gerados e então o software fez a análise desses bits. O resultado encontrado pode ser observado na tabela 1, onde o nome do teste e o valor obtido como resultado é mostrado, além de explicitar se o TRNG passou ou não no teste.

Teste realizado	Valor obtido(p-valor)	Resultado
Teste de frequência do Monobit	0,09128	Aleatório
Teste de frequência em bloco	0,22218	Aleatório
Teste de corrida	0,73067	Aleatório
Mais longa execução de uns em um bloco	0,76129	Aleatório
Teste de classificação de matriz binária	0,09984	Aleatório
Teste da transformada discreta de Fourier	0,04222	Aleatório
Teste de correspondência de modelo sem sobreposição	0,72711	Aleatório
Teste de correspondência de modelo com sobreposição	0,49533	Aleatório
Teste de complexidade linear	0,64445	Aleatório
Teste Serial	0,86796	Aleatório
Teste da entropia aproximada	0,54962	Aleatório
Teste das somas cumulativas(Para frente)	0,15316	Aleatório
Teste das somas cumulativas(Para trás)	0,13048	Aleatório
Teste de excursões aleatórias	0,88044	Aleatório

Tabela 1: Tabela de testes realizados

Como podemos notar o TRNG passou em todos os testes propostos, dessa forma podemos concluir que, assim como esperado o TRNG de fato gera números aleatórios.

Conclusões

Por fim, nesse resumo foi descrito todo o processo de desenvolvimento do TRNG proposto no projeto, desde o trabalho onde foi baseado passando pelo circuito até os testes para verificação de aleatoriedade.
Os resultados obtidos foram assim como esperado e o TRNG gera bits aleatórios de acordo com os padrões do órgão que define esses padrões.
Para trabalhos futuros será desenvolvido outro gerador baseado em outra arquitetura e o sistema de obtenção de dados será aprimorado para que a obtenção desses valores seja feita de forma mais eficaz.

Referências Bibliográficas

[1] ADDABBO A. FORT, R. M. M. M. V. V. T. Analysis of a Circuit Primitive for the Reliable Design of Digital Nonlinear Oscillators. 2019.

[2]XU, Y. W. X. High Speed True Random Number Generator Based on FPGA. 2016.

[3]RUKHIN J. SOTO, J. N. M. S. E. B. S. L. M. L. M. V. D. B. A. H. J. D. S. V.
A. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. [S.l.]: National Institute of Standards and Technology, 2010.