Universidade Federal do Pampa

Processos Estocásticos

Professor Fabiano Tondello Castoldi

Validação do Gerador de Números Aleatórios do Microcontrolador ESP32

Jean Senger da Silva
Magno Costa Maia

8 de dezembro de 2020

Sumário

1 Introdução

Eventos como o lançar de uma moeda ou a escolha de uma carta de um baralho possuem em comum a característica da aleatoriedade de seus resultados. Aleatoriedade é uma propriedade utilizada em muitas aplicações relacionadas a criptografia [1], geração de números sequência em pacotes e atrasos na retransmissão de pacotes [2].

O Gerador de Números Aleatórios (RNG) pode produzir números com a característica verdadeiramente aleatória ou não, chamados de pseudoaleatórios, dependendo do processo utilizado na sua geração. Existem na literatura vários métodos e métricas para medir a real aleatoriedade da sequência de números gerados [1].

Pesquisas relacionadas à geração de números aleatórios utilizando diversos métodos, e qualidade dos mesmos, podem ser facilmente encontradas [3][4], porém a maioria dos dispositivos atuais já possuem em si mecanismos RNG, abrindo a oportunidade para a investigação dos atributos destes.

Dito isto, este trabalho tem como objetivo analisar o RNG de um dispositivo embarcado, levando em consideração alguns aspectos como o tipo de RNG utilizado, distribuição de probabilidades que foi gerada e autocorrelação. O restante do trabalho é organizado da seguinte maneira: primeiro será feita uma pequena introdução quanto a plataforma escolhida, o microcontrolador ESP32, seguindo por uma explicação quanto à RNG, seus tipos e aplicações e uma comparação teórica será feita quanto as características do RNG descritas na documentação referente ao ESP32.

Para finalizar, sequências de números aleatórios serão geradas utilizando o microcontrolador para que seja possível estimar as características discutidas anteriormente.

2 Fundamentação Teórica

2.1 Variável Aleatória

Uma variável aleatória pode ser descrita como uma associação de um processo aleatório a cada elemento de um espaço amostral. Ela pode ser descrita como uma característica numérica de um experimento [5]. Ou seja, associa um número real a cada ponto do espaço amostral. Uma variável aleatória pode ser discreta ou contínua. Ela será discreta quando possuir um valor enumerável dentro de um espaço finito ou infinito, e será contínua quando estiver dentro de um intervalo de valores contínuos [6].

As variáveis aleatórias podem ser associadas de acordo com suas famílias. Isto é feito devido ao processo a ser utilizado. As famílias são: Bernoulli, Geométrica, Binomial, Discreta Uniforme, Poisson, etc. Para este trabalho a família da Variável Aleatória Discreta Uniforme (k,l) será abordada. Na Equação (1) a Função Massa Probabilidade (PMF) desta família é vista [7].

$$P_X(x) = \begin{cases} \frac{1}{l-k+1} & x = k, k+1, ..., l \\ 0 & \text{caso contrário} \end{cases} \tag{1}$$

Outros parâmetros podem ser analisados para esta família, como por exemplo, média, variância e o desvio padrão. A média pode ser vista na Equação (2) e é utilizada para estimar a média dos valores de um espaço amostral. A variância pode ser vista na Equação (3) e é utilizada para estimar a diferença entre a variável aleatória e o seu valor esperado.

O desvio padrão pode ser visto na Equação (4) e é utilizado para verificar o grau de dispersão das variáveis aleatórias [8].

$$E[X] = \frac{k+l}{2} \tag{2}$$

$$Var[X] = \frac{(l-k)(l-k+2)}{12} \tag{3}$$

$$\sigma_X = \sqrt{Var[X]} \tag{4}$$

A covariância demonstra o quanto uma variável é independente de outra variável aleatória. A covariância é dada pela Equação (5).

$$cov(A,B) = \begin{bmatrix} cov(A,A) & cov(A,B) \\ cov(B,A) & cov(B,B) \end{bmatrix} \tag{5}$$

Sendo, $cov(A,B)$ dada pela Equação (6) e $r_{A,B}$ dada pela Equação (7).

$$cov[A,B] = r_{A,B} - \mu_A \mu_B \tag{6}$$

$$r_{A,B} = E[AB] \tag{7}$$

2.2 Microcontroldor ESP32

O microcontrolador é da família do ESP8266 e possui conectividade wi-fi e bluetooth já integrado na placa, o que facilita projetos de Internet das Coisas (IoT). Possui três núcleos em seu processador, um destes sendo Ultra Low Power - ULP, e também um sensor de temperatura embutido, utilizado para controlar a temperatura da placa. Pode ser programado como o Arduíno, com o CodeBlocks, Netbeans e outros meios [9].

Este microcontrolador pode ser utilizado na automação industrial, controle de câmeras, portões, televisores, rádios, câmera de segurança, alarmes, etc. Ou seja, a partir dele, a casa inteira pode ser conectada e controlada a partir de um celular conectado ao mesmo [9].

O microcontrolador possui um timer interno que é acionado desde o momento que é ligado. Este contador está na escala de microsegundos. Cada vez que é extraído um valor aleatório, na verdade é um valor baseado na escala de tempo do microcontrolador. Porém, como é uma escala de tempo muito pequena, pode-se dizer que o processo é dado como aleatório, apesar de ser pseudoaleatório. O processo se denomina aleatório pelo fato de não saber o valor que o *timer* será atribuído. Quando se apresenta um espaço amostral muito alto, o processo tende a tender a um padrão de amostras [10]. Como este controlador possui 32 bits, os valores aleatórios estarão entre 0 e 2^{32}.

2.3 Test Suite - NIST

Os testes [11] foram desenvolvidos com o intuito de ser um primeiro passo promissor para determinar se um gerador de números aleatórios e é adequado para uma aplicação. Esta conclusão é feita a partir de vários testes independentes, sendo que alguns destes serão apresentados nessa seção para depois serem utilizados em cima do gerador de números aleatórios do ESP32.

2.3.1 Teste de Frequência (Monobit)

De acordo com [11], o propósito deste teste é determinar se o número de "zeros" e "ums" em uma sequência são aproximadamente o mesmo que seria esperado para uma sequência verdadeiramente aleatória.

2.3.2 Teste de Frequência Dentro de um Bloco

De acordo com [11], o foco deste teste é a proporção de "ums"dentro de blocos com M bits. Tendo que a frequência esperada de valores "um" é aproximadamente $\frac{M}{2}$.

2.3.3 Teste de Sequência Ininterrupta

De acordo com [11], o foco deste teste é o número total de sequências ininterrupta de bits idênticos dentro de um bloco de bits.

2.3.4 Teste para a Maior Sequência Ininterrupta de "ums" Dentro de um Bloco

De acordo com [11], o foco deste teste é determinar se o comprimento da maior sequência ininterrupta de "ums"dentro de um bloco é a esperada para um bloco de bits aleatórios.

2.3.5 Teste do Rank de uma Matriz Binária

De acordo com [11], o foco deste teste é checar por dependências lineares de subsequências de comprimento fixo dentro da sequência original.

2.3.6 Teste Utilizando a Transformada Discreta de Fourier

De acordo com [11], o foco deste teste é detectar características periódicas na sequência testada que indicariam um desvio da aleatoriedade assumida.

2.3.7 Teste da Complexidade Linear

De acordo com [11], o foco deste teste é determinar se uma sequência de bits é complexa o suficiente ou não para ser considerada aleatória. O foco deste teste é o comprimento de um *linear feedback shift register* (LFSR), pois sequências aleatórias são caracterizadas por LFSRs longos.

3 Estratégia para validação

A metodologia implementada neste trabalho segue os princípios propostos em [11], sendo sugeridos cinco etapas essenciais para a analise estatística de um RNG.

A primeira etapa consiste da escolha do gerador, no caso o microcontrolador ESP32. Ele possui uma função especifica para a geração de números aleatórios e esta possui dois modos de operação: quando o módulo wi-fi esta ativo, a geração é feita utilizando o próprio hardware, se tornando assim um número verdadeiramente aleatório. Sem este módulo ativo, a geração é feita utilizando software e é considerado um número pseudoaleatório [12].

A segunda etapa, a geração propriamente dita foi feita no microcontrolador com o modulo sem fio ligado e desligado a fim de gerar amostras aleatórias e pseudo aleatórias.

Os dados gerados pelo ESP32 são então tratados utilizando o software MATLAB com o proposito de avaliar certas propriedades como Valor Esperado e qual família de variáveis aleatórias estes valores gerados pertencem.

Os valores encontrados, como valor esperado, variância, etc, também foram comparados aos valores definidos na literatura.

A etapa três consiste da execução dos testes estatísticos da NIST, porém estes testes são feitos em sequências binárias. Para adequar a este critério, as amostras de 32 bits geradas pelo microcontrolador foram transformadas em uma sequência binária única.

A quarta etapa consistiu de utilizar os bits utilizados na etapa anterior nos testes propostos na sub-seção 2.3. Estes testes resultam em valores ρ e com estes é possível avaliar se uma sequência pode ser considerada verdadeiramente aleatória ou não.

A quinta etapa trata-se da interpretação destes valores ρ.

4 Resultados

Para que seja possível utilizar as aproximações para o cálculo de parâmetros como Valor Esperado e Variância é necessário encontrar que tipo de família de variáveis aleatórias as amostras pertencem.

Para isto as amostras foram organizadas em um histograma. Como não é possível observar a PMF destas amostras, observar o comportamento do histograma de um número grande de amostras pode indicar a família na qual esta pertence.

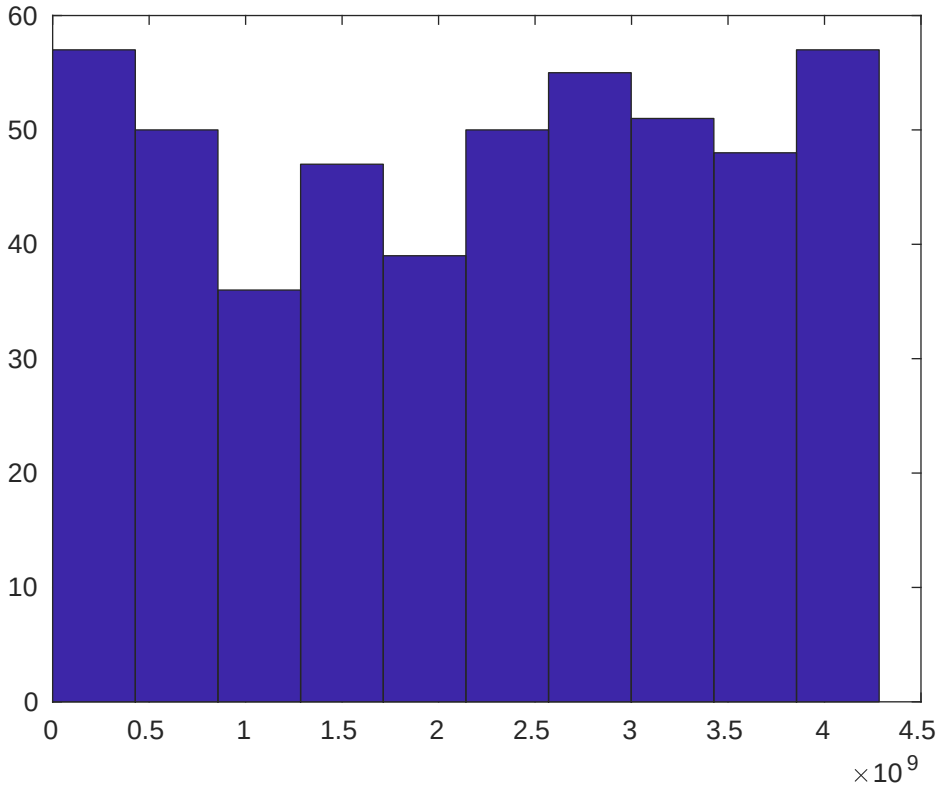

Figura 1: Histograma de uma Amostra.

A Figura 1 mostra o histograma plotado utilizando uma da sequência de valores gerados no ESP32 e a partir dele é possível concluir que a variável aleatória na qual as amostras foram obtidas se trata de uma da família de Variáveis Aleatórias Discreta Uniforme.

4.1 Simulados

O microcontrolador ESP32 foi utilizado para gerar vetores de números aleatórios. Estes vetores consistem de 490 valores inteiros sem sinal de 32 bits e a representação da distribuição dos valores em um destes vetores é representada na Figura 2.

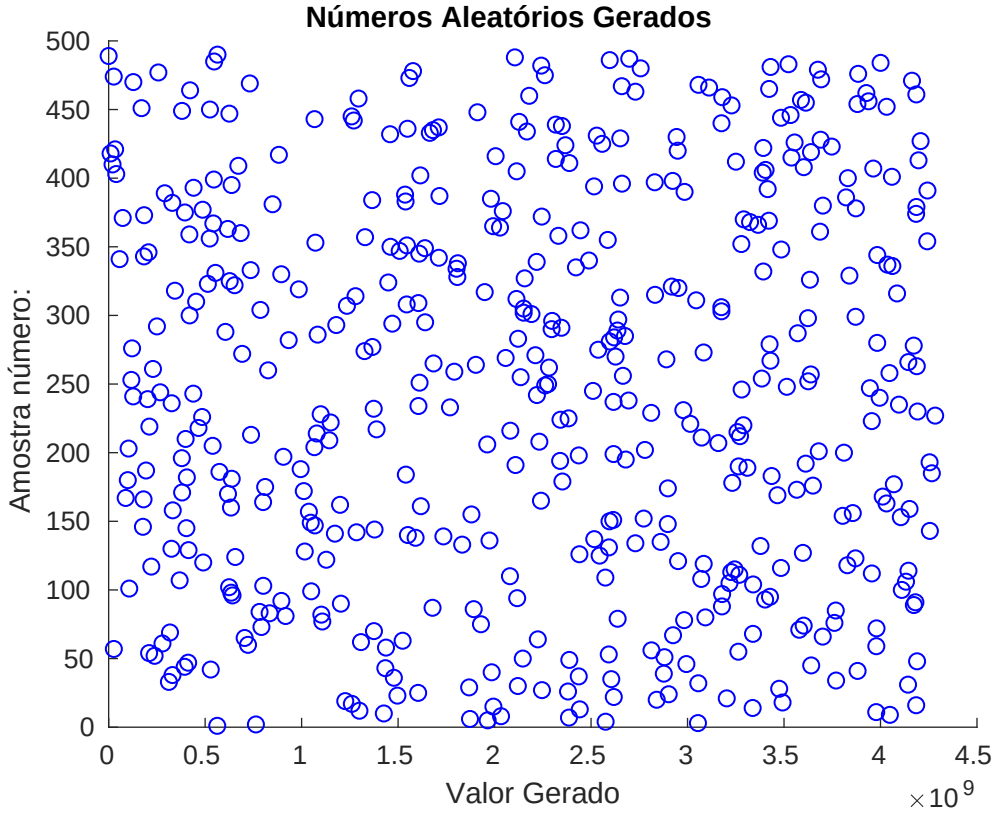

Figura 2: Distribuição dos valores

Os vetores X, Y e Z foram geradas, contendo dados dos números aleatórios gerados a partir do *wi-fi* ligado e os vetores x_0, y_0 e z_0 foram gerados com o *wi-fi* desligado.

Com o *wi-fi* do microcontrolador desligado foi gerado o vetor x_0 onde contém o espaço amostral com os valores aleatórios, e foram obtidos: uma média $2,1372.10^9$, variância de $1,5292.10^{18}$, desvio padrão de $1,2366.10^9$ e a matriz covariância dada por:

$$cov(X,Y) = \begin{bmatrix} 1,5282.10^{18} & -1,6452^{17} \\ -1,6452.10^{17} & 1,5032.10^{18} \end{bmatrix}$$

Com o *wi-fi* do microcontrolador desligado foi gerado o vetor y_0 onde contém o espaço amostral com os valores aleatórios, e foram obtidos uma média $2,0849.10^9$, variância de $1,5032.10^{18}$, desvio padrão de $1,2261.10^9$ e a matriz covariância dada por:

$$cov(Y,Z) = \begin{bmatrix} 1,5032.10^{18} & -8,3464^{16} \\ -8,3464.10^{16} & 1,5330.10^{18} \end{bmatrix}$$

Com o *wi-fi* do microcontrolador desligado foi gerado o vetor z_0 onde contém o espaço amostral com os valores aleatórios, e foram obtidos: uma média $2,093610^9$, variância de $1,5330.10^{18}$, desvio padrão de $1,2381.10^9$ e a matriz covariância dada por:

$$cov(Z,X) = \begin{bmatrix} 1,5330.10^{18} & -1,2616^{17} \\ -1,2616.10^{17} & 1,5292.10^{18} \end{bmatrix}$$

Agora com o *wi-fi* do microcontrolador ligado foi gerado o vetor X onde contém o espaço amostral com os valores aleatórios, e foram obtidos: uma média $0,5020$, variância de $0,2505$, desvio padrão de $0,5005$ e a matriz covariância dada por:

$$cov(X,Y) = \begin{bmatrix} 0,2505 & 0,0010 \\ 0,0010 & 0,2493 \end{bmatrix}$$

Com o *wi-fi* do microcontrolador ligado foi gerado o vetor Y onde contém o espaço amostral com os valores aleatórios, e foram obtidos: uma média $0,4648$, variância de $0,2493$, desvio padrão de $0,4993$ e a matriz covariância dada por:

$$cov(Y,Z) = \begin{bmatrix} 0,2493 & -0,0175 \\ -0,0175 & 0,2498 \end{bmatrix}$$

Com o *wi-fi* do microcontrolador ligado foi gerado o vetor Z onde contém o espaço amostral com os valores aleatórios, e foram obtidos: uma média $0,4746$, variância de $0,2498$, desvio padrão de $0,4998$ e a matriz covariância dada por:

$$cov(Z,X) = \begin{bmatrix} 0,2498 & -0,0136 \\ -0,0136 & 0,2505 \end{bmatrix}$$

4.2 Calculados

Para o vetor x_0 com o *wi-fi* do microcontrolador desligado foram obtidos: a média $2,1470.10^9$, variância $1,5365.10^{18}$, desvio padrão de $1,2395.10^9$ e a matriz covariância dada por:

$$cov(X,Y) = \begin{bmatrix} 1,5261.10^{18} & -1,6419.10^{17} \\ -1,6419.1018 & 1,5002.10^{18} \end{bmatrix}$$

Para o vetor y_0 com o *wi-fi* do microcontrolador desligado foram obtidos: a média $2,1443.10^9$, variância $1,5326.10^{18}$, desvio padrão de $1,2380.10^9$ e a matriz covariância dada por:

$$cov(Y,Z) = \begin{bmatrix} 1,5002.10^{18} & -8,3294.10^{16} \\ -8,3294.1016 & 1,5299.10^{18} \end{bmatrix}$$

Para o vetor z_0 com o *wi-fi* do microcontrolador desligado foram obtidos: a média $2,1467.10^9$, variância $1,5360.10^{18}$, desvio padrão de $1,2394.10^9$ e a matriz covariância dada por:

$$cov(Z,X) = \begin{bmatrix} 1,5299.10^{18} & -1,2590.10^{17} \\ -1,2590.1017 & 1,5261.10^{18} \end{bmatrix}$$

Para o vetor X com o *wi-fi* do microcontrolador ligado foram obtidos: a média $0,50$, variância $0,25$, desvio padrão de $0,50$ e a matriz covariância dada por:

$$cov(X,Y) = \begin{bmatrix} 0,2500 & 0,0010 \\ 0,0010 & 0,2488 \end{bmatrix}$$

Para o vetor Y com o *wi-fi* do microcontrolador ligado foram obtidos: a média $0,50$, variância $0,25$, desvio padrão de $0,50$.

$$cov(Y,Z) = \begin{bmatrix} 0,2488 & -0,0175 \\ -0,0175 & 0,2494 \end{bmatrix}$$

Para o vetor Z com o *wi-fi* do microcontrolador ligado foram obtidos: a média $0,50$, variância $0,25$, desvio padrão de $0,50$.

$$cov(Z,X) = \begin{bmatrix} 0,2494 & -0,0136 \\ -0,0136 & 0,2500 \end{bmatrix}$$

4.3 Resultados do Test Suite

Para averiguar se o RNG do ESP32 gera números que podem ser considerados verdadeiramente aleatório, os testes descritos na sub-seção 2.3 são então aplicados.

Assim como os métodos aplicados, [11] também disponibiliza um software responsável por fazer os cálculos destes métodos, sendo necessário apenas disponibilizar os dados de entrada de forma binária. Como os valores aleatórios criados no microcontrolador são disponibilizados no formato inteiro sem sinal de 32 bits, o software MATLAB foi utilizado para convertê-los para valores binários.

Os resultados obtidos podem ser observados na tabela abaixo e por questão de formatação, os testes foram enumerados conforme a sequência da sub-seção 2.3.

Tabela 1: Resultados Test Suite

Teste n	x_0	y_0	z_0	x	y	z
Teste 1 ($\rho =$)	0,472304	0,576150	0,749394	0,424525	0,554547	0,005189
Teste 2 ($\rho =$)	0,626409	0,025926	0,170000	0,947775	0,091656	0,834502
Teste 3 ($\rho =$)	0,134319	0,739194	0,774356	0,252238	0.788130	0.986055
Teste 4 ($\rho =$)	0,389946	0,743429	0,466567	0,535652	0,644357	0,365192
Teste 5 ($\rho =$)	0,233583	0,733620	0,291369	0,069927	0,549731	0,389597
Teste 6 ($\rho =$)	0,714051	0,660151	0,240978	0,379179	0,379179	1,000000
Teste 7 ($\rho =$)	0,092748	0,356753	0,488836	0,086413	0,718638	0,320682

De acordo com [11], se o valor de ρ é computado como <0.01, a sequência é considerada como não-aleatória. Caso contrário, a sequência é considerada aleatória. Com isto em

mente, a primeira consideração que pode ser observada, é que todas as seis amostras passam em todos os testes, exceto a primeira variável no teste 1 ($\rho_z = 0.005189 < 0.01$).

Outra característica interessante é o fato de que os valores de ρ geradas com o *wi-fi* ligado, sem o subscrito 0, não são necessariamente maiores que os com o *wi-fi* desligado, hipótese plausível, já que as primeiras são geradas por RNGs em hardware.

Para facilitar a visualização da variação entre os valores de ρ, a seguinte Figura 2 foi construída.

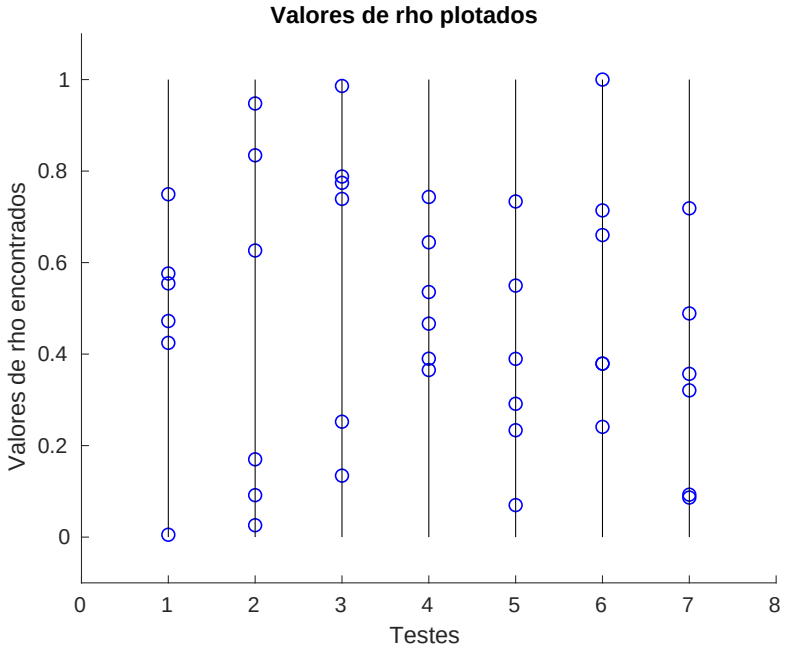

Figura 3: Valores de ρ plotados

5 Conclusão

A partir dos dados calculados com a fundamentação teórica e os dados do experimento, uma analise foi possível ser feita. Quando os valores são coletados com o *wi-fi* desligado, os valores teóricos e práticos possuem uma maior diferença nos valores, o que consegue ser melhor calculado para o *wi-fi* ligado. Para poder melhor estimar o comportamento dos parâmetros com o *wi-fi* desligado deve ser pego mais amostras, afim de uma melhor comparação dos comportamentos. Feito isto, será possível verificar que a mesma possuirá um padrão de distribuição. A diferença entre os valores se dá pois os dados recebido pelo *wi-fi* já são dados normalizados, e os dados recebidos pelo *wi-fi* desligado não.

Apesar da diferença entre os valores para cada um dos vetores amostras, o Teste NIST sugere que ambas as versões do RNG do microcontrolador ESP32 geram números verdadeiramente aleatórios, validando assim o real proposito destes.

Para trabalhos futuros é sugerido a utilização de mais bits, a verificação se é possível obter os valores aleatórios do ESP32 diretamente em valores binários, e se isso altera algo, e por fim realizar todos os 15 testes propostos em [11].

Referências

[1] G. M. Callico, S. Lopez, J. F. Lopez, R. Sarmiento, and A. Nunez, "Low-cost implementation of a super-resolution algorithm for real-time video applications," in *2005 IEEE International Symposium on Circuits and Systems*, pp. 6130–6133 Vol. 6, 2005.

[2] D. Seetharam and Sokwoo Rhee, "An efficient pseudo random number generator for low-power sensor networks [wireless networks]," in *29th Annual IEEE International Conference on Local Computer Networks*, pp. 560–562, 2004.

[3] J. Tsai, I. Chen, and J. Tzeng, "Random number generated from white noise of webcam," in *2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 214–217, 2009.

[4] K. Sathya, J. Premalatha, and V. Rajasekar, "Random number generation based on sensor with decimation method," in *2015 IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions (WCI)*, pp. 1–5, 2015.

[5] *Portal Action*, *Variáveis Aleatórias*. http://www.portalaction.com.br/probabilidades/variaveis-aleatorias, 4 de dezembro de 2020.

[6] U. F. do Amazonas (UFAM), *Variáveis Aleatórias, Notas de estudo de Estatística*. https://www.docsity.com/pt/variaveis-aleatorias-4/4726385/, 4 de dezembro de 2020.

[7] F. T. Castoldi, *Variáveis Aleatórias Discretas*. http://www.moodle.unipampa.edu.br/, 4 de dezembro de 2020.

[8] R. Gouveia, *Desvio Padrão*. https://www.todamateria.com.br/desvio-padrao/, 4 de dezembro de 2020.

[9] A. Electronics, *ESP32 – Especificações e Projetos*. https://athoselectronics.com/esp32/, 4 de dezembro de 2020.

[10] M. Projetado, *Criar gerador de números aleatórios – Arduino*. http://mundoprojetado.com.br/gerador-nos-aleatorios/, 4 de dezembro de 2020.

[11] A. Rukhin, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," *National Institute of Standards and Technology*, 2010.

[12] *Miscellaneous System APIs*. https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/system/system.html, 4 de dezembro de 2020.