Computação quântica aplicada à hardware reconfigurável com foco em segurança da informação

Gustavo Inácio Arraes Fernandes[1], **Gabriel Tauchen Filgueiras**[1], **Otávio de Souza Martins Gomes**[1]

[1]Universidade Federal de Itajubá
Itajubá – MG – Brasil

arraes.fernandes@hotmail.com, otavio.gomes@unifei.edu.br

Abstract. *This work presents the implementation of a quantum circuit emulator on FPGA, focusing on the application of Grover's search algorithm. The system transmits the amplitude coefficients of quantum states via UART to a microcontroller, enabling practical validation through comparison with simulations performed in Qiskit. The architecture, based on fixed-point arithmetic, achieved results consistent with theoretical models while requiring minimal logical resources. The relevance of Grover's algorithm to post-quantum cybersecurity is discussed, as well as the potential of the proposed prototype as a tool to support the analysis and mitigation of cryptographic vulnerabilities.*
Keywords: *quantum computing; FPGA; Grover's algorithm; post-quantum cryptography; hardware emulation.*

Resumo. *Este trabalho apresenta a implementação de um emulador de circuitos quânticos em FPGA, com foco na aplicação do algoritmo de busca de Grover. O sistema transmite os coeficientes probabilísticos dos estados quânticos via UART para um microcontrolador, o que possibilita a validação prática por meio da comparação com simulações realizadas no Qiskit. A arquitetura, baseada em aritmética de ponto fixo, obteve resultados compatíveis com modelos teóricos, utilizando poucos recursos lógicos. São discutidas a relevância do algoritmo para a cibersegurança pós-quântica e o potencial do protótipo como ferramenta de apoio à análise e mitigação de vulnerabilidades criptográficas.*
Palavras-chave: *computação quântica; FPGA; algoritmo de Grover; criptografia pós-quântica; emulação em hardware.*

1. Introdução

A segurança da informação enfrenta mudanças significativas devido à computação quântica, que representa uma grave ameaça aos sistemas criptográficos clássicos. Ao mesmo tempo, esse cenário impulsiona o desenvolvimento de novos protocolos de segurança baseados em princípios quânticos, como entrelaçamento e superposição [1]. Diante desse cenário, torna-se urgente a avaliação e validação prática de algoritmos quânticos, tanto ofensivos quanto defensivos, no contexto da cibersegurança. Contudo, o alto custo e a complexidade do hardware quântico real tornam essa abordagem limitada.

Para contornar esse desafio, sistemas clássicos têm sido amplamente utilizados na emulação de circuitos quânticos. Simuladores baseados em software, como o Qiskit

da IBM, oferecem boa fidelidade matemática [2], mas enfrentam limitações práticas, especialmente quanto ao paralelismo intrínseco à computação quântica e limitações em aplicações embarcadas ou de tempo real. Nesse contexto, FPGAs (Field Programmable Gate Arrays) surgem como alternativa eficiente. FPGAs oferecem paralelismo nativo, previsibilidade temporal e flexibilidade arquitetural. Consequentemente, são adequadas à prototipação e execução de experimentos, tanto em contextos educacionais quanto em aplicações práticas [3].

Este trabalho propõe a implementação de uma arquitetura modular em FPGA para emulação de portas quânticas, utilizando aritmética de ponto fixo `Q1.14`. A proposta é validada pela comparação dos resultados obtidos com simulações no Qiskit, focando na execução do algoritmo de busca de Grover. A implementação do algoritmo permite discutir seu impacto na cibersegurança pós-quântica e explorar sua viabilidade como ferramenta de ensino, prototipação e análise de ataques.

A estrutura deste artigo é organizada da seguinte forma: primeiramente se discute os trabalhos relacionados, depois, na Seção 3, apresenta-se o algoritmo de Grover, na Seção 4 há um detalhamento da metodologia implementada no projeto, e após isso, uma apresentação dos resultados obtidos. E por fim, a Seção 6 conclui o trabalho e propõe trabalhos futuros.

2. Trabalhos Relacionados

Diversas abordagens têm sido propostas na literatura para a emulação de circuitos quânticos utilizando arquiteturas reconfiguráveis. Para validação e comparação da nossa arquitetura, o trabalho de Khalid et al. [3] foi escolhido como *benchmark* primário por sua transparência metodológica e dos resultados apresentados. Sua implementação, focada em baixo custo, redução de recursos e portabilidade, alinha-se aos objetivos deste trabalho. A emulação de referência foi desenvolvida em VHDL com aritmética de ponto fixo (8 e 16 bits) e implementada em um FPGA Altera Stratix. Contudo, o estado da arte em emulação quântica em FPGAs evoluiu para diferentes objetivos de otimização, focadas em simulação acelerada por meio de otimizações nos cálculos tensoriais e trigonométricos [10] e utilização de técnicas de multiplicação complexas com massivo consumo de memória, escalabilidade e paralelismo [11]. A abordagem de Khalid et al. foi escolhida por permitir uma avaliação clara dos trade-offs de desempenho e recursos. Isso permitiu comparação com nossa proposta, voltada à emulação precisa e com aplicação prática em sistemas embarcados de baixo custo.

Os autores relatam o desempenho das portas lógicas com base no consumo de Células Lógicas (LCs), bem como o tempo de execução. Para avaliação da precisão, comparam os coeficientes probabilísticos resultantes da emulação em FPGA com os obtidos via simulação em software Libquantum, executada em um processador Pentium IV de 2 GHz. A comparação dos coeficientes probabilísticos mencionada foi realizada utilizando o erro absoluto entre os coeficientes ideais e os discretizados, conforme a Equação 1:

$$E = \sqrt{\alpha_e^2 + \beta_e^2} \tag{1}$$

onde α_e e β_e representam os erros nos coeficientes do vetor de estado do qubit.

3. Algoritmo de Grover

O Algoritmo de Grover, implementado neste trabalho como caso de uso prático, é uma técnica de busca quântica de chaves criptográficas simétricas, como o AES. Este algoritmo permite encontrar um item em um banco de dados não ordenado de N elementos com complexidade temporal $O(\sqrt{N})$, representando uma aceleração quadrática em relação a buscas clássicas.

Essa aceleração se faz possível através de um processo iterativo que utiliza dois componentes fundamentais: o Oráculo e o Difusor. O Oráculo atua como uma operação "caixa-preta"que marca o estado correspondente à busca esperada, aplicando-lhe uma inversão de fase. O difusor serve como um amplificador de amplitude, atuando após o Oráculo, aumentando a amplitude do coeficiente de probabilidade do estado marcado e diminuindo a de todos os outros [4].

Para a cibersegurança, essa aceleração representa uma ameaça concreta. A segurança de uma chave AES-128, que depende de uma busca exaustiva de 2^{128} chaves, seria reduzida para o nível de uma busca de 2^{64} elementos, tornando um ataque de força bruta computacionalmente viável com um computador quântico eficiente. Funções hash também estão vulneráveis a ataques de pré-imagem com a mesma redução quadrática citada [5] [6].

Portanto, a emulação do algoritmo em plataformas reconfiguráveis, conforme proposto neste trabalho, torna-se uma contribuição relevante para a experimentação, ensino e validação de estratégias de segurança em um cenário próximo ao pós-quântico.

4. Metodologia

O sistema foi desenvolvido no ambiente Altera Quartus Prime Lite Edition 18.1, utilizado para a síntese dos circuitos lógicos e integração com o simulador ModelSim. Os circuitos foram executados em uma FPGA Altera MAX 10 (10M50DAF484C7G), escolhida pelo seu bom balanço entre recursos e custo para prototipação.

Os circuitos foram descritos em Verilog HDL, focando em modularização, escalabilidade e compatibilidade com pipeline. Cada módulo lógico foi projetado para representar uma operação unitária específica, sendo interconectados no módulo de topo (`emulator.v`) de acordo com a lógica do circuito a ser emulado.

O estado quântico de um qubit, um vetor da forma $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, é descrito por dois coeficientes complexos, α e β. Estes foram representados em aritmética de ponto fixo no formato `Q1.14`, totalizando 16 bits por componente (8 bytes por qubit, considerando as partes real e imaginária de ambos os coeficientes). O formato `Q1.14` consiste em 1 bit de sinal, 1 bit para a parte inteira e 14 bits para a parte fracionária, permitindo representar números no intervalo aproximado de -2 a 1.999 com alta resolução decimal. Essa escolha visa garantir precisão suficiente para representar operações unitárias, ao mesmo tempo em que mantém a complexidade de hardware reduzida.

As portas lógicas quânticas foram emuladas com base nas suas representações matriciais, conforme formalismo da mecânica quântica [1], sendo adaptadas para operações aritméticas compatíveis com sistemas digitais. A multiplicação por constantes, como $1/\sqrt{2}$ na porta Hadamard, foi implementada diretamente em ponto fixo.

Para garantir o paralelismo e a sincronização das informações durante a evolução do circuito, foi descrito um *Registrador de Estados Quânticos Entrelaçados*. Módulo de extrema relevância para a implementação do Algoritmo de Grover, pois além de assegurar a propagação correta das informações, permite a emulação do entrelaçamento, condição essencial quando os qubits deixam de ser representados individualmente.

Ao final da execução, os coeficientes α e β atualizados são enviados ao módulo `uart_sender.v`, que transmite os bytes resultantes via UART (Universal Asynchronous Receiver-Transmitter), um protocolo de comunicação serial assíncrono amplamente utilizado devido à sua boa relação entre simplicidade e velocidade. Essa transmissão ocorre para um microcontrolador ESP32. O ESP32 atua como interface de monitoramento e análise, possibilitando comparação com os resultados teóricos simulados via Qiskit.

O Algoritmo de Grover foi implementado utilizando exclusivamente as portas lógicas básicas mencionadas (Hadamard, CNOT, X e Toffoli), além do uso de Registradores de Estados Quânticos Entrelaçados. Essa configuração viabilizou a execução iterativa necessária ao funcionamento do algoritmo. Um qubit auxiliar (*ancilla*) foi incluído para auxiliar na correta realização das operações quânticas controladas. Para validação prática, foi implementada a busca por um elemento dentro de um espaço de busca de quatro estados ($|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$), com o estado alvo definido em $|01\rangle$. A seguir, é apresentada a figura RTL do componente Oráculo, essencial para o funcionamento do algoritmo.

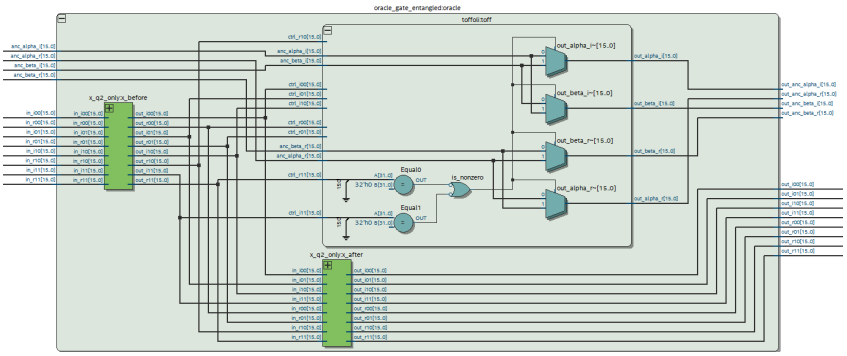

Figura 1. Diagrama RTL do componente Oráculo (Fonte: Elaborado pelos autores, 2025).

5. Resultados

A análise de erro absoluto, conforme discutida na Equação 1, permite estimar o desvio entre os coeficientes ideais e os obtidos fisicamente no sistema embarcado. Dentre as portas lógicas quânticas implementadas, destaca-se a porta Hadamard como a mais sensível a erros acumulados, visto que realiza operações matemáticas não triviais com constantes irracionais. A porta X, por outro lado, é implementada com inversão direta de coeficientes e não adiciona erro significativo. Já para portas de múltiplos qubits, como a CNOT e a Toffoli, o erro depende da forma de propagação das amplitudes no circuito.

A Tabela 1 apresenta um comparativo direto entre o erro absoluto reportado na literatura e o obtido na implementação deste trabalho.

O erro absoluto obtido, de $2,05 \times 10^{-5}$, é da mesma ordem de grandeza do reportado no trabalho de referência ($3,05 \times 10^{-5}$), indicando a compatibilidade da nossa abordagem de ponto fixo, mesmo utilizando menos recursos lógicos.

Tabela 1. Erro absoluto na porta Hadamard (Fonte: Elaborado pelos autores, 2025).

Fonte	Erro absoluto E
Khalid et al. [3]	$3{,}05 \times 10^{-5}$
Este trabalho	$2{,}05 \times 10^{-5}$

Outro fator relevante para análise da eficiência do sistema é o uso de recursos lógicos (Logic Cells - LCs) e registradores. Para o algoritmo de Grover implementado (2 qubits + ancilla), foram utilizados apenas 57 LCs (0,11% do total disponível) e 38 registradores. Por comparação, a implementação de Grover no trabalho de Khalid et al. ocupava 12.636 LCs na FPGA Altera Stratix EP1S80. A baixa utilização de LCs permite uma adaptação dos módulos Oráculo e Difusor para escalonamento com mais qubits.

Tabela 2. Comparação do uso de recursos lógicos (Fonte: Elaborado pelos autores, 2025).

Fonte	LCs Utilizados	Registradores
Khalid et al. [3]	12.636	Não especificado
Este trabalho	57 (0,11%)	38 (0,07%)

O tempo de execução do algoritmo também foi avaliado de forma empírica, utilizando simulação no ModelSim com monitoramento do número de ciclos de clock. A execução completa do Grover requereu 12 ciclos com clock de 50 MHz, o que corresponde a um tempo total de 240 ns. A Tabela 3 apresenta a comparação entre os tempos reportados no trabalho referenciado, na presente implementação e em uma execução simulada via Qiskit [7] (em um Intel Core i7 8ª geração), incluindo os tempos de transpilação e simulação.

Tabela 3. Comparação dos tempos de execução para o algoritmo de Grover (Fonte: Elaborado pelos autores, 2025).

Plataforma	Tempo de Execução
FPGA (Khalid et al.)	84 ns
FPGA (este trabalho, 12 ciclos a 50 MHz)	240 ns
Qiskit (simulação + transpilação)	87,25 ms

A diferença entre os tempos de execução na FPGA e na simulação em software é da ordem de 360 mil vezes, reforçando a adequação do uso de FPGAs em sistemas embarcados para aplicações em segurança da informação, especialmente quando se busca respostas em tempo real e operação com restrições de consumo e área. A Tabela 3 evidencia um trade-off presente na arquitetura desenvolvida. Embora o tempo de execução de 240 ns, seja superior aos 84 ns reportados por Khalid et al., essa diferença é uma consequência da abordagem adotada em uma solução otimizada para baixo custo e mínima utilização de recursos lógicos. Priorizou-se uma arquitetura mais simples e serializada, em detrimento do paralelismo máximo que FPGAs mais robustas podem proporcionar. Essa escolha reforça a adequação para sistemas embarcados e em tempo real.

5.1. A Plataforma como ferramenta de Análise e Escalabilidade

A arquitetura proposta serve, além da emulação efetuada, como uma ferramenta de investigação e quantificação dos desafios práticos da computação quântica. A meto-

dologia escalonável implementada permite uma análise de vulnerabilidades de cifras reais ao medir o custo de implementação do oráculo, principal alvo crítico da viabilidade do algoritmo [12], em duas dimensões:

- **Custo de Recursos Lógicos:** A implementação de referência consumiu apenas 57 LCs. Graças a metodologia modular adotada, essa eficiência libera o uso dos recursos da FPGA para construção de oráculos mais complexos. Os módulos individuais de portas lógicas podem ser facilmente adaptados para operar sobre múltiplos estados simultaneamente, permitindo a manipulação de estados específicos dentro do produto tensorial total. Isso torna possível projetar oráculos para problemas reais (como uma S-Box da cifra AES) e medir seu custo precisamente, fornecendo um limite inferior prático para a complexidade de um ataque.
- **Custo de Tempo:** Graças ao paralelismo nativo da execução em hardware e a simulação em nível de ciclo de clock do ModelSim, têm-se uma análise precisa do tempo de execução de qualquer oráculo implementado. Assim, pode-se mover uma análise de complexidade teórica $O(\sqrt{N})$ para uma previsão de tempo concreta em nanossegundos.

Essa metodologia de quantificação estabelece um roteiro para a análise de cifras reais. Para superar as limitações de pinos I/O da FPGA, a proposta usa sinais internos (`wires`) empacotados via UART, o que reduz a complexidade física e mantém a flexibilidade. Protocolos de alta velocidade também podem ser integrados para ataques práticos e particionamento distribuído.

6. Conclusão e Trabalhos Futuros

Este trabalho demonstra a viabilidade do uso de hardware reconfigurável (FPGA) não só como plataforma para executar algoritmos quânticos, mas também para quantificar seus custos práticos em hardware. A implementação do algoritmo de Grover, embora não represente uma ameaça em sua escala atual, demonstra uma metodologia robusta de análise, o que valida a arquitetura como uma plataforma ágil e eficiente para a experimentação, ensino e prototipação de conceitos de cibersegurança quântica.

Através de uma abordagem modular baseada em aritmética de ponto fixo `Q1.14` replicou-se satisfatoriamente os postulados matemáticos da mecânica quântica através de um baixo uso de recursos lógicos (57 LCs e 38 Registradores) e desempenho otimizado, com apenas 12 ciclos de clock a 50 MHz para execução completa do algoritmo. Resultados obtidos graças à capacidade de emulação nativa de paralelismo dessa arquitetura. Esse ganho foi obtido ao custo de um tempo de execução superior, validando a arquitetura como solução leve e embarcada. A implementação demonstrou potencial significativo para aplicações embarcadas, sobretudo em sistemas de cibersegurança quântica, onde restrições de energia, latência e custo são determinantes. A arquitetura proposta viabiliza prototipações e testes reais de algoritmos quânticos.

Como trabalhos futuros, planeja-se implementar algoritmos como os de Shor e Simon, avaliando o custo de oráculos complexos como os da cifra AES. Adicionalmente, planeja-se a integração do emulador com sistemas reais de distribuição de chaves quânticas (QKD), a exemplo do modelo de Zhang et al. [8] [**?**], que também utiliza FPGAs para controle em tempo real. Isso possibilitaria validar empiricamente a resistência de aplicações QKD frente a ameaças pós-quânticas, como um ataque baseado no algoritmo de Gro-

ver. A plataforma proposta integra a análise de vulnerabilidades com a prototipação de defesas. Essa abordagem permite tanto identificar fraquezas em sistemas atuais quanto formular e validar contramedidas, como novas técnicas de criptografia híbrida resistentes à computação quântica.

Agradecimentos

Os autores expressam sua gratidão à Universidade Federal de Itajubá (UNIFEI), ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), à Financiadora de Estudos e Projetos (FINEP) e ao projeto Clavis PlatCiber pelo apoio que tornaram este projeto possível.

Referências

[1] Nielsen, M. A.; Chuang, I. L. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[2] IBM Quantum. *Qiskit Documentation*, 2023. Disponível em: `https://www.ibm.com/quantum/qiskit`.

[3] Khalid, A. U.; Zilic, Z.; Radecka, K. FPGA Emulation of Quantum Circuits. *Proceedings of the IEEE International Conference on Computer Design (ICCD'04)*, 2004.

[4] Grover, L. K. A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, p. 212-219, 1996.

[5] Bernstein, D. J.; Lange, T. Post-quantum cryptography. *Nature*, v. 549, n. 7671, p. 188-194, 2017. DOI: 10.1038/nature23461.

[6] Alagic, G. et al. *Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process*. NIST IR 8413, July 2022. Disponível em: `https://doi.org/10.6028/NIST.IR.8413`.

[7] IBM Quantum Learning. *Grover's Algorithm Tutorial*, 2023. Disponível em: `https://learning.quantum.ibm.com/tutorial/grovers-algorithm`.

[8] Zhang, H. et al. A real-time QKD system based on FPGA. *Journal of Lightwave Technology*, v. 30, n. 18, p. 3026-3030, 2012. DOI: 10.1109/JLT.2012.2217145.

[9] Bennett, C. H.; Brassard, G. Quantum cryptography: Public key distribution and coin tossing. *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing*, p. 175-179, 1984.

[10] Jungjarassub, Y.; Piromsopa, K. *A Performance Optimization of Quantum Computing Simulation using FPGA*. 2022 19th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2022. DOI: 10.1109/ECTI-CON54298.2022.9795495.

[11] Belfore II, L. A. *A Scalable FPGA Architecture for Quantum Computing Simulation*. Old Dominion University, 2024. Disponível em: `https://doi.org/10.48550/arXiv.2407.06415`.

[12] Viamontes, G. F.; Markov, I. L.; Hayes, J. P. *Is Quantum Search Practical?* Computing in Science Engineering, vol. 7, no. 3, pp. 22-30, 2005. DOI: 10.1109/MCSE.2005.53.