

Emulação do Algoritmo Quântico Deutsch-Josza por Meio de Linguagem de Descrição de Hardware

David M. C. Bezerra¹, José A. N. Silva¹, Francisco H. C. dos Santos Filho¹,
Thiago W. B. da Silva¹, Antônio Joel R. de Castro¹

¹ Engenharia de Computação – Universidade Federal do Ceará (UFC)
Quixadá – CE – Brasil

{davidmachado, jadriannassilva}@alu.ufc.br, joelcastro@fisica.ufc.br
helderhdw@ufc.br, thiagowerlley@ufc.br

Abstract. *This work aims to demonstrate the transformative potential of quantum computing by using the Deutsch-Josza algorithm as a case study to illustrate the superiority of quantum computers over classical ones. Due to current quantum hardware limitations, we resort to emulation in Hardware Description Language (HDL) to simulate quantum behavior on classical platforms. Thus, providing a platform for detailed analysis and deep understanding of quantum operations and their applicability in digital systems.*

Resumo. *Este trabalho trata-se de um projeto da computação quântica, utilizando o algoritmo de Deutsch-Josza como um estudo de caso para ilustrar a superioridade dos computadores quânticos sobre os clássicos. Devido às restrições atuais de hardware quântico, foi recorrido à emulação em Linguagem de Descrição de Hardware (HDL) para simular o comportamento quântico em plataformas clássicas. Assim proporcionando uma plataforma para análise detalhada e compreensão profunda das operações quânticas e sua aplicabilidade em sistemas digitais.*

Palavras-chave: Computador Quântico. Deutsch-Josza. HDL. Algoritmos. Emulação.

1. Introdução

A computação quântica é uma das novas tecnologias que prometem revolucionar o campo da tecnologia por meio de propriedades não presentes em um computador clássico [Terhal 2018]. Um termo foi criado para designar o momento em que um computador quântico superará um clássico em desempenho: "supremacia quântica". Este termo é utilizado quando um computador quântico pode resolver um problema que levaria um tempo exponencial para um computador clássico.

Um dos algoritmos mais renomados, que serve como um marco demonstrativo do potencial da computação quântica, foi introduzido por [Deutsch and Jozsa 1992]. Eles propuseram uma solução quântica que, em contraste com abordagens clássicas de complexidade exponencial, prometia uma resolução de complexidade linear. O algoritmo foi nomeado Deutsch-Jozsa.

A grande questão é como implementar o algoritmo desejado, considerando que a atual era de computação quântica é nomeada de *Noisy Intermediate Scale Quantum*

(NISQ) [Lau et al. 2022]. Nessa fase, os computadores quânticos têm a capacidade de executar certos algoritmos, mas enfrentam limitações, principalmente devido a ruídos. Assim, a utilização de ferramentas como simulações e emulações se torna essencial para a pesquisa neste campo. Neste trabalho, é proposta uma emulação em hardware description language (HDL) do algoritmo Deutsch-Jozsa.

2. Fundamentação Teórica

O algoritmo foi proposto por David Deutsch e Richard Jozsa e lida com uma função booleana f mapeada para $\{0, 1\}$. Esta função pode ser categorizada de duas maneiras: é constante quando todas as suas saídas são iguais ou é balanceada quando tem uma quantidade igual de saídas 1 e 0. O desafio é determinar, dada a função f , sua categoria. Enquanto em um ambiente de computação clássica seriam necessárias duas consultas para tal determinação, a abordagem quântica consegue fornecer a resposta com apenas uma. A Figura 1 ilustra o algoritmo, mostrando como é implementado em um circuito.

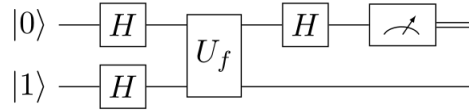


Figura 1. Algoritmo Deutsch-Jozsa.

2.1. Funcionamento do Algoritmo Deutsch-Jozsa

O algoritmo de Deutsch-Jozsa inicia com dois qubits em superposição via portas Hadamard. O segundo qubit, que explora o phase-kickback, também é processado. A caixa preta quântica, ou oráculo, aplica a operação $U_f|x\rangle|y\rangle = |x\rangle|f(x) \oplus y\rangle$. A tabela 1, ilustra as saídas do oráculo. Após a aplicação do oráculo, outro Hadamard é aplicado no primeiro qubit e uma medição é feita para determinar o resultado.

Input $ x, y\rangle$	Output $ x, y \oplus f(x)\rangle$
$ 0, 0\rangle$	$ 0, 0 \oplus f(0)\rangle$
$ 0, 1\rangle$	$ 0, 1 \oplus f(0)\rangle$
$ 1, 0\rangle$	$ 1, 0 \oplus f(1)\rangle$
$ 1, 1\rangle$	$ 1, 1 \oplus f(1)\rangle$

Tabela 1. Tabela verdade de um oráculo quântico.

2.2. Fundamentos Matemáticos e Intuição

Para compreender a eficácia do algoritmo de Deutsch-Jozsa, é imperativo entender os pilares da mecânica quântica que o sustentam: superposição quântica, interferência e o fenômeno de *phase kickback*.

2.2.1. Superposição Quântica

Ao aplicarmos a porta de Hadamard ao qubit de consulta e ao qubit de verificação (que estava inicialmente no estado $|1\rangle$), obtemos o seguinte estado:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |+\rangle |-\rangle \quad (1)$$

Neste estado de superposição, o qubit de consulta pode explorar ambos os valores de $f(x)$ simultaneamente, possibilitando uma busca paralela eficiente.

2.2.2. Oráculo e Phase Kickback

Ao alimentar o estado $|+\rangle |-\rangle$ no oráculo quântico, obtemos:

$$|\psi'\rangle = \frac{1}{\sqrt{2}}U_f |0\rangle |-\rangle + U_f |1\rangle |-\rangle \quad (2)$$

Graças ao fenômeno de *phase kickback*, se a função $f(x)$ retorna 1 para um determinado x , uma fase negativa é aplicada ao qubit correspondente de consulta. Assim, o estado acima pode ser reescrito como:

$$|\psi'\rangle = \frac{1}{\sqrt{2}}((-1)^{f(0)} |0\rangle + (-1)^{f(1)} |1\rangle) \otimes |-\rangle \quad (3)$$

Neste estado, o qubit de consulta acumula a fase introduzida pela operação do oráculo, o que é uma manifestação do fenômeno *phase kickback*. Esta fase será crucial para a interferência subsequente e para determinar se a função é constante ou balanceada.

2.2.3. Saída do Algoritmo

Com esse estado resultante do oráculo e do *phase-kickback*, é realizado implementado outra porta Hadamard para levar a base padrão de $|0\rangle$ e $|1\rangle$ e com isso garantido o estado final:

$$\frac{1}{2}(((-1)^{f(0)} + (-1)^{f(1)}) |0\rangle + ((-1)^{f(0)} - (-1)^{f(1)}) |1\rangle) \quad (4)$$

Assim, quando medimos o qubit de consulta e encontramos o estado $|0\rangle$, podemos concluir que f é constante. Por outro lado, se medirmos e encontrarmos o estado $|1\rangle$, sabemos que f é balanceada. Esta conclusão é obtida com apenas uma consulta ao oráculo, ilustrando a eficácia superior da abordagem quântica em comparação com métodos clássicos.

3. Metodologia

A emulação do algoritmo utilizando VHDL foi desenvolvida com base na compreensão de que cada elemento do algoritmo atua como uma operação em um espaço vetorial, especificamente no espaço $H \otimes H$, que corresponde ao espaço de Hilbert em 4 dimensões. Cada porta do algoritmo foi representada por matrizes e o estado atual do sistema foi descrito por um vetor de estado.

3.1. Representação de Qubit

Considerando que o algoritmo faz uso de dois qubits, a notação de estado expandido foi adotada para representar o sistema composto por dois qubits, conforme ilustrado pela Equação (2):

$$|\psi\rangle = c_0 |00\rangle + c_1 |01\rangle + c_2 |10\rangle + c_3 |11\rangle \quad (5)$$

As constantes c 's são de natureza complexa. Para representar e operar com os estados quânticos em VHDL, foi introduzido um pacote VHDL personalizado. Este pacote implementa um estado expandido como um vetor desses números complexos.

3.2. Implementação das Portas Quânticas

Diversos portões quânticos são utilizados, cada um como entidades distintas:

- **Porta IX:** Esta porta (`IXGate`) é encarregado de aplicar um portão de identidade no primeiro qubit e um portão Pauli-X no segundo.
- **Porta Hadamard:** Duas entidades distintas, `HadamardXHadamard` e `HadamardXIdentity`, garantem a correta aplicação da operação de Hadamard, seja em ambos os qubits ou individualmente.
- **Oráculo Quântico:** Central para o algoritmo Deutsch-Josza é o oráculo quântico. A entidade `QuantumOracle` emula isso, conduzindo a função oculta que mapeia entradas em saídas.

3.3. Implementação da Medição

A simulação de medições quânticas é realizada com base na sua natureza probabilística, usando a entidade `LFSR` para gerar números pseudoaleatórios. O `LFSR`, inicializado com um valor de semente, fornece um número inteiro de 13 bits adequado para o *Field-programmable gate array* (FPGA). Esses números alimentam a entidade `Measurement`, que, considerando as probabilidades dos estados quânticos definidas pelas suas amplitudes, simula o processo de medição quântica ao selecionar um estado baseado no número gerado.

4. Resultados

A eficiência do algoritmo Deutsch-Josza foi avaliada usando um *testbench*, que simula o comportamento quântico. Com isso, a medição é simulada levando em conta a geração de números pseudoaleatórios provenientes da entidade `LFSR` e a aplicação da entidade `Measurement` para determinar o estado quântico resultante.

O processo de estímulo `stim_proc` configura o estado quântico inicial e, em seguida, realiza uma série de medições (1.000, neste caso). Cada resultado de medição é escrito em um arquivo de texto chamado `measurement_results.txt`. O resultado é obtido por meio da simulação da execução gerando um *waveform*, como mostrado na Figura 2.

Mostrando o resultado válido da arquitetura montada, pois a função implementada no oráculo tinha sido de uma função f é balanceada e como pode ser visto resultado do primeiro qubit é sempre 1, como previsto no resultado teórico.

Os resultados contidos no arquivo `measurement_results.txt` foram posteriormente processados em Python para gerar um histograma. Esse histograma fornece uma representação visual da frequência de cada resultado de medição, permitindo uma avaliação qualitativa e quantitativa da implementação do algoritmo. Mostrado a seguir na Figura 3.

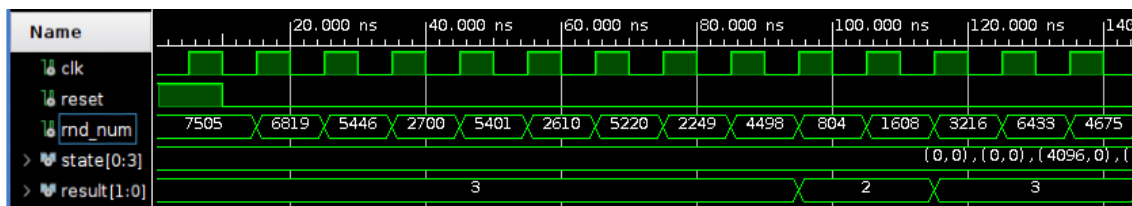


Figura 2. Waveform da emulação.

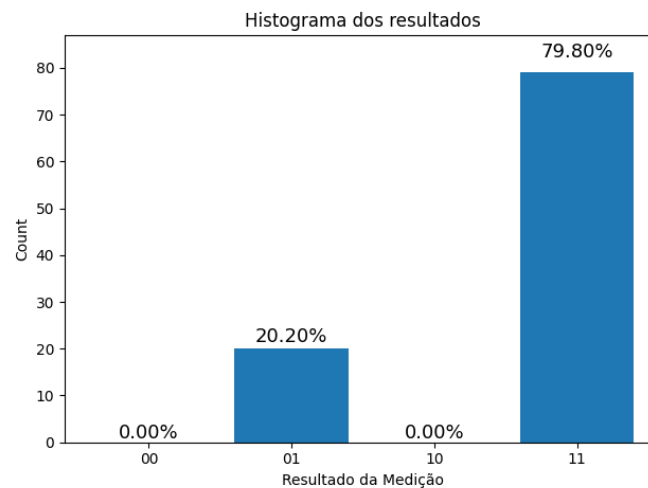


Figura 3. Histograma dos resultados Obtidos.

5. Conclusão

A avaliação do algoritmo Deutsch-Jozsa, mediante simulação quântica, permitiu validar a implementação e comprovar os resultados teóricos esperados. Os resultados obtidos abrem um caminho para explorar algoritmos quânticos mais complexos e seus potenciais aplicações no mundo real.

Referências

- Deutsch, D. and Jozsa, R. (1992). Rapid solution of problems by quantum computation. *Proc. R. Soc. Lond. A*, 439:553–558.
- Lau, J., Lim, K., Shrotriya, H., et al. (2022). Nisq computing: where are we and where do we go? *AAPPS Bull.*, 32:27.
- Terhal, B. M. (2018). Quantum supremacy, here we come. *Nature Physics*, 14(6):530–531.