



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
DEPARTAMENTO DE ENGENHARIAS
CURSO BACHARELADO EM ENGENHARIA ELÉTRICA

CARLOS HENRIQUE DANTAS DA COSTA

**GERADOR DE FORMAS DE ONDA ARBITRÁRIAS UTILIZANDO A TÉCNICA DE
SÍNTESE DIGITAL DIRETA PARA APLICAÇÃO EM COMPUTAÇÃO QUÂNTICA**

CARAÚBAS

2025

CARLOS HENRIQUE DANTAS DA COSTA

**GERADOR DE FORMAS DE ONDA ARBITRÁRIAS UTILIZANDO A TÉCNICA DE
SÍNTESE DIGITAL DIRETA PARA APLICAÇÃO EM COMPUTAÇÃO QUÂNTICA**

Monografia apresentada a Universidade
Federal Rural do Semi-Árido como requisito
para obtenção do título de Bacharel em
Engenharia Elétrica.

Orientador: Prof. Dr. Francisco de Assis Brito
Filho

CARAÚBAS

2025

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

C837g Costa, Carlos Henrique Dantas da.
Gerador de formas de onda arbitrárias
utilizando a técnica de síntese digital direta
para aplicação em computação quântica / Carlos
Henrique Dantas da Costa. - 2025.
57 f. : il.

Orientador: Francisco de Assis Brito Filho .
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de , 2025.

1. Computação quântica. 2. Síntese digital
direta. 3. Gerador de formas de onda arbitrárias
. 4. FPGA. I. Brito Filho , Francisco de Assis ,
orient. II. Título.

Ficha catalográfica elaborada por sistema gerador automático em conformidade
com AACR2 e os dados fornecidos pelo(a) autor(a).

Biblioteca Campus Caraúbas
Bibliotecária: Sarah Freire Bezerra
CRB: 15/1052

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

CARLOS HENRIQUE DANTAS DA COSTA

**GERADOR DE FORMAS DE ONDA ARBITRÁRIAS UTILIZANDO A TÉCNICA DE
SÍNTESE DIGITAL DIRETA PARA APLICAÇÃO EM COMPUTAÇÃO QUÂNTICA**

Monografia apresentada a Universidade
Federal Rural do Semi-Árido como requisito
para obtenção do título de Bacharel em
Engenharia Elétrica.

Defendida em: 17/12/2025.

BANCA EXAMINADORA

Prof. Dr. Francisco de Assis Brito Filho (UFERSA)
Presidente

Prof. Dr. José Garibaldi Duarte Júnior (UFERSA)
Membro Examinador

Eng. Daniel da Silva Santos
Membro Examinador

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus por sempre me iluminar, por derramar bençãos em minha vida, por ser o meu refúgio e a minha fortaleza em todos os momentos de angústia e aflição. Toda honra e toda glória a Ti, Senhor!

Aos meus pais, Francimarques de Almeida Costa e Rita Dantas dos Santos, por todo amor, carinho, cuidado e inúmeras abdições para a concretização de um sonho. Amo vocês!

Aos meus familiares, por todo apoio e ajuda ao longo dessa jornada.

A minha namorada Lorena Lúcio, por todo amor, companheirismo, compreensão e apoio.

Aos inúmeros amigos que pude fazer ao longo desta caminhada, pessoas com as quais eu pude trocar conhecimento, compartilhar momentos de descontração, bem como de frustrações. Sigo na torcida por cada um de vocês!

Ao meu professor e orientador, Francisco de Assis Brito Filho, pela oportunidade e auxílio no desenvolvimento deste trabalho.

A banca examinadora pela disponibilidade para avaliação deste trabalho e a Universidade Federal Rural do Semi-árido – UFRSA seus professores, técnicos e demais servidores por todas as oportunidades e conhecimentos ofertados.

RESUMO

A computação quântica exige sistemas eletrônicos de controle de alta precisão para a manipulação de qubits, nos quais os geradores de formas de onda arbitrárias (AWGs) desempenham um papel fundamental. Este trabalho apresenta o desenvolvimento e implementação digital de um gerador de formas de onda arbitrárias baseado na técnica de síntese digital direta (DDS), com capacidade de gerar ondas senoidais com controle independente de fase e amplitude. A arquitetura do AWG foi desenvolvida por meio da linguagem de descrição de *hardware* Verilog, utilizando o *software* Quartus II. A arquitetura foi validada por simulação funcional no ModelSim e implementada em um FPGA Cyclone IV EP4CE22F17C6 com clock de 50 MHz. Para a validação experimental, utilizou-se um *setup* composto pelo kit DE0-Nano integrado a um conversor digital-analógico R-2R montado em protoboard e ao dispositivo Analog Discovery 2 para análise em tempo real e controle de parâmetros. Os resultados evidenciam que a implementação digital do AWG foi capaz de gerar ondas senoidais com frequência de 1 MHz com resolução de 14 bits e controle preciso dos parâmetros de fase e amplitude. A técnica de *dithering* fase foi validada experimentalmente, mostrando-se eficaz na mitigação de espúrios, embora a pureza espectral tenha sido limitada pela não linearidade do DAC R-2R e pela ausência de filtragem de reconstrução.

Palavras-chave: Computação Quântica; Síntese Digital Direta; Gerador de Formas de Onda Arbitrárias; FPGA.

ABSTRACT

Quantum computing requires high-precision electronic control systems for qubit manipulation, in which Arbitrary Waveform Generators (AWGs) play a fundamental role. This work presents the development and digital implementation of an A WG based on the Direct Digital Synthesis (DDS) technique, capable of generating sine waves with independent phase and amplitude control. The A WG architecture was developed using the Verilog hardware description language within the Quartus II software environment. The architecture was validated through functional simulation in ModelSim and implemented on a Cyclone IV EP4CE22F17C6 FPGA with a 50 MHz clock. For experimental validation, a setup consisting of the DE0-Nano kit integrated with a breadboard-mounted R-2R digital-to-analog converter and an Analog Discovery 2 device was used for real-time analysis and parameter control. The results demonstrate that the digital A WG implementation was capable of generating 1 MHz sine waves with 14-bit resolution and precise control of phase and amplitude parameters. The phase dithering technique was experimentally validated, proving effective in mitigating spurs, although spectral purity was limited by the non-linearity of the R-2R DAC and the absence of a reconstruction filter.

Keywords: Quantum computing; Direct Digital Synthesis; Arbitrary Waveform Generator; FPGA.

LISTA DE FIGURAS

Figura 1 – O estado do gato de Schrödinger expresso na notação Dirac.	16
Figura 2 – Representação do qubit na esfera de Bloch.	17
Figura 3 – Uma moeda lançada tem 50% de chance de cair em cara ou coroa.....	18
Figura 4 - Diferentes estados de um qubit representados na esfera de Bloch.	18
Figura 5 – Dois qubits em estados quânticos não emaranhados (a) e emaranhados (b).	19
Figura 6 – Natureza ondulatória de um qubit.	20
Figura 7 – Fenômeno da interferência quântica em duas funções de onda quântica.	21
Figura 8 – Exemplo de um circuito quântico.	22
Figura 9 – Configuração atual de um computador quântico.	24
Figura 10 – Diagrama de blocos de um sistema de processador quântico supercondutor.	25
Figura 11 – AWG modelo AWG70000B, da Tektronix.	27
Figura 12 – Diagrama de blocos da arquitetura DDS e fluxo completo da geração de sinal.	28
Figura 13 – Processo de endereçamento feito pela LUT.....	28
Figura 14 – Geração de uma senoide através da técnica DDS.	29
Figura 15 – Arquitetura DDS com capacidade de controle de frequência, fase e amplitude.	31
Figura 16 – Diagrama de blocos do processo de conversão D/A.....	32
Figura 17 – Sinal $X_S[n]$	32
Figura 18 – Sinal $X_A[t]$	33
Figura 19 – Sinal $X_A[t]$	33
Figura 20 – Conversor DAC R-2R Ladder no modo de tensão.	34
Figura 21 – Arquitetura básica de um FPGA.	36
Figura 22 – Fluxograma do desenvolvimento do projeto.....	38
Figura 23 – Kit de desenvolvimento DE0-Nano.	39
Figura 24 – Esquemático do conversor DAC R-2R.	40
Figura 25 – Dispositivo Analog Discovery 2.	40
Figura 26 – Diagrama de montagem do setup usado na validação experimental: (a) conexões GPIO DE0-Nano; (b) conexões Analog Discovery 2; (c) conexões conversor DAC R-2R.	41
Figura 27 – Diagrama de blocos para o AWG.	42
Figura 28 – Diagrama de blocos para o acumulador de fase.....	43
Figura 29 – Gerador de números pseudoaleatórios.	44
Figura 30 – Diagrama de blocos para o AWG produzido pelo Quartus II.....	46
Figura 31 – Formas de onda geradas durante a simulação: (a) controle de amplitude; (b) controle de fase; (c) controle de amplitude e fase.	47
Figura 32 – Senoide gerada com amplitude máxima e frequência de 1 MHz.....	48
Figura 33 – Senoide gerada com 75% de amplitude e frequência de 1 MHz.	49
Figura 34 – Senoide gerada com 50% de amplitude e frequência de 1 MHz.	49
Figura 35 – Senoide gerada com 25% de amplitude e frequência de 1 MHz.	50
Figura 36 – Senoides geradas com: fase 0° (laranja); fase 90° (verde); fase 180° (azul).	50
Figura 37 – Senoides geradas com: amplitude 50% e fase 0° (laranja); amplitude 75% e fase 180°; amplitude 25% e fase 90°.	51

Figura 38 – FFT da senoide de 1 MHz: (a) sem dithering de fase; (b) com dithering de fase.	
.....	53

SUMÁRIO

1	INTRODUÇÃO.....	11
1.1	OBJETIVOS	12
2	REFERENCIAL TEÓRICO.....	14
2.1	Computação quântica	14
2.1.1	Fundamentos da computação quântica	14
2.2	A interface de controle e leitura de qubits	22
2.2.1	Blocos de construção lógicos: portas quânticas e medição.....	22
2.2.2	A implementação física de sistemas quânticos	23
2.2.3	A interface clássico-quântica: eletrônica de controle e leitura	24
2.3	Geradores de formas de onda arbitrárias	26
2.4	Síntese digital direta	27
2.4.1	Arquitetura DDS básica	27
2.4.2	Arquitetura DDS com controle de fase, amplitude e frequência	31
2.5	Conversão digital para analógico	32
2.5.1	Teoria de funcionamento	32
2.5.2	Conversores digital para analógico	33
2.6	FPGAs.....	35
2.6.1	Linguagem de descrição de <i>hardware</i>	36
3	METODOLOGIA	38
4	DESENVOLVIMENTO DO AWG	42
4.1	Acumulador de fase (AF)	43
4.2	Gerador de números pseudoaleatórios	43
4.3	LUT	45
5	RESULTADOS	46
5.1	Simulação funcional	46
5.2	Testes experimentais	48
6	CONSIDERAÇÕES FINAIS	54
	REFERÊNCIAS	55

1 INTRODUÇÃO

A computação quântica é uma área interdisciplinar que une aspectos da mecânica quântica e da ciência da computação, e que tem despertado significativo interesse em virtude do seu potencial de revolucionar inúmeros ramos da sociedade. Através de princípios da teoria quântica e fenômenos quânticos como o emaranhamento e a superposição, é possível realizar certas tarefas de maneira exponencialmente mais rápida do que a computação clássica. Por conseguinte, a computação quântica tem o potencial de transformar inúmeras áreas, incluindo a aeroespacial, farmacêutica, astrofísica, segurança cibernética, dentre outras (SHAFIQUE; MUNIR; LATIF, 2024).

Os computadores clássicos realizam o processamento de dados por meio de lógica binária e armazenam informações em bits, ao passo que os computadores quânticos utilizam qubits como suas unidades fundamentais. Segundo Nielsen e Chuang (2011), enquanto os bits estão restritos a zero ou um, os qubits, em virtude da superposição, são capazes de existir em estados de zero, um ou em uma combinação linear de ambos simultaneamente. Essa característica exclusiva concede aos computadores quânticos a capacidade de seguir, de maneira simultânea, múltiplos caminhos computacionais em uma única operação, algo inexequível em computadores convencionais, sem que haja a repetição de iterações (SHAFIQUE; MUNIR; LATIF, 2024).

Os atuais sistemas de computação quântica são compostos por um ou mais chips de qubit (posicionados dentro de um refrigerador de diluição com temperatura base típica de 10 mK), eletrônica de controle e um computador clássico responsável por executar uma pilha de *software*. O controle dos qubits é feito através de sinais analógicos de radiofrequência (RF), tais como pulsos de tensão CC e pulsos de micro-ondas, produzidos por instrumentos de controle situados fora do refrigerador (PARK *et al.*, 2021).

No que tange à execução de operações sobre os qubits, é necessário o uso de sistemas eletrônicos de controle responsáveis pela manipulação e leitura dos estados quânticos dos qubits. Para que o desempenho dos qubits não seja prejudicado, esses sistemas devem fornecer sinais de controle de alta precisão e de baixo ruído, além de realizarem uma leitura altamente sensível, sem causar alterações indesejadas nos estados quânticos dos qubits (VAN DIJK *et al.*, 2019).

Os geradores de formas de onda arbitrárias (do inglês, arbitrary waveform generator ou AWG) são amplamente utilizados em sistemas eletrônicos de controle quântico, pois possibilitam a síntese de pulsos de micro-ondas precisos e ajustáveis em fase, amplitude e

frequência. Além disso, o AWG possibilita ao usuário maior configurabilidade, precisão e estabilidade na geração do sinal.

Atualmente, inúmeros grupos de pesquisa em computação quântica utilizam AWGs comerciais, que atendem, até certo ponto, às necessidades operacionais para um número reduzido de qubits. Contudo, o avanço em direção à computação quântica tolerante a falhas estabelece novas exigências para a eletrônica de controle. Tais requisitos incluem a necessidade de menor latência de comunicação, um processamento de sinal mais rápido e uma maior densidade de integração que garanta a escalabilidade do sistema (YANG *et al.*, 2021).

Para atender a esses requisitos, propõe-se o desenvolvimento de um AWG customizado, utilizando uma arquitetura baseada em FPGA (field-programmable gate array). O uso do FPGA é fundamental, pois propicia baixa latência de comunicação e o alto paralelismo necessários ao sistema.

Aliada à capacidade dos FPGAs, a técnica de síntese digital direta (DDS) desponta como uma abordagem eficiente e moderna para a geração de sinais. De acordo com Junhao *et al.* (2022), AWGs baseados em DDS têm ocupado um amplo espaço no mercado da eletrônica digital por apresentarem alta estabilidade de frequência, fase continuamente variável, ampla faixa de frequência de trabalho e capacidade de geração de formas de onda arbitrárias.

Diante deste cenário, o presente trabalho se debruça na implementação digital de um gerador de formas de onda arbitrárias, utilizando a técnica de síntese digital direta, voltado à aplicação em sistemas de controle e medição de qubits, visando contribuir para o avanço da pesquisa em computação quântica no país.

1.1 OBJETIVOS

O principal objetivo deste trabalho é realizar a implementação digital de um gerador de formas de onda arbitrárias (AWG) baseado na técnica de síntese digital direta, capaz de gerar ondas senoidais com controle independente de fase e amplitude.

Para alcançar esse objetivo, foram traçados os objetivos específicos a seguir:

- i. Estudar os princípios teóricos da computação quântica e da técnica de síntese digital direta, analisando as principais arquiteturas e aplicações desta técnica em geradores de formas de onda arbitrárias;
- ii. Projetar e descrever a arquitetura digital do AWG por meio de uma linguagem de descrição de *hardware*;

- iii. Realizar a validação lógica da arquitetura proposta através de simulação funcional;
- iv. Implementar o sistema em um dispositivo FPGA e validar experimentalmente os sinais gerados por meio de um *setup* com conversão digital-analógica.

2 REFERENCIAL TEÓRICO

Este capítulo aborda a fundamentação teórica necessária para o desenvolvimento do presente trabalho. Nesse sentido, o conteúdo foi segmentado nas seguintes seções: computação quântica, geradores de formas de onda arbitrárias, síntese digital direta, conversor digital-analógico e FPGAs.

2.1 Computação quântica

Essencialmente, o desempenho dos sistemas computacionais está estritamente ligado à sua capacidade de armazenamento e manipulação de informações. Na computação quântica, são explorados fenômenos da mecânica quântica, como a superposição e o emaranhamento, para potencializar a representação e manipulação dos dados. As unidades de informação quântica, bits quânticos ou qubits, não se limitam aos valores binários de 0 ou 1 característicos dos bits da computação clássica (HUGHES *et al.*, 2021). Em invés disso, um qubit pode existir em uma superposição de ambos os estados simultaneamente. Em virtude dessa propriedade, um conjunto de qubits é capaz de representar um número exponencialmente maior de valores, o que torna a computação quântica uma ferramenta promissora para a resolução de problemas computacionais atualmente considerados intratáveis.

Nesta seção, serão abordados os fundamentos da computação quântica, as diferentes representações de qubits e as propriedades quânticas por eles utilizadas, bem como os processos de controle e leitura de seus estados.

2.1.1 Fundamentos da computação quântica

Conforme visto anteriormente, a unidade básica de informação da computação quântica é o qubit, que possui a capacidade de existir em múltiplos estados simultaneamente, fenômeno conhecido como superposição. Outro fenômeno intrinsicamente ligado à computação quântica é o emaranhamento, uma conexão única entre qubits. Existe também a interferência quântica, capaz de alterar os resultados destes. Além disso, os computadores quânticos enfrentam a problemática do ruído, que pode ocasionar a perda das propriedades supracitadas, afetando o resultado de um sistema. Esta seção visa prover uma compreensão concisa, porém ampla, acerca dos fundamentos da computação quântica (HUGHES *et al.*, 2021).

2.1.1.1 Qubit

Na computação clássica, o bit opera de forma binária, podendo assumir somente dois estados discretos: 0 ou 1. Em contrapartida, o bit quântico (ou qubit) utiliza o princípio da superposição, o que lhe permite existir não apenas nos estados 0 ou 1, mas em uma combinação linear de ambos simultaneamente. Esta superposição é descrita por amplitudes de probabilidade, que determinam a probabilidade de se obter o estado 0 ou 1 no momento da medição (SHAFIQUE; MUNIR; LATIF, 2024).

Na mecânica quântica e, consequentemente, na computação quântica, os estados costumam ser representados na notação Dirac ou “bra-ket” (NIELSEN; CHUANG, 2011). Nesta notação, o estado de um qubit é descrito pela combinação linear dos estados da base computacional, expressos como $|0\rangle$ e $|1\rangle$. O estado de um qubit, expresso como $|\Psi\rangle$, pode ser escrito como:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1)$$

Onde α e β são denominados de amplitudes de probabilidade. Essas amplitudes são cruciais, pois seus módulos ao quadrado definem a probabilidade de se obter um determinado estado do qubit no momento da medição (VASANTRAO; SAXENA, 2025).

De acordo com Nielsen e Chuang (2011), uma vez que a soma das probabilidades de observar todos os estados do sistema quântico deve ser igual a 1, as amplitudes devem seguir a regra de normalização determinada pela Equação 2.

$$|\alpha|^2 + |\beta|^2 = 1 \quad (2)$$

A Figura 1 a seguir ilustra este conceito através da famosa analogia do paradoxo do gato de Schrödinger. Nesta analogia, o gato existe em uma superposição quântica dos estados “vivo” (associado ao $|0\rangle$) e “morto” (associado ao $|1\rangle$), até que a medição (a observação) force o sistema a colapsar para um desses estados.

Figura 1 – O estado do gato de Schrödinger expresso na notação Dirac.

$$|\text{cat}\rangle = \alpha \left| \text{gato acordado} \right\rangle + \beta \left| \text{gato adormecido} \right\rangle$$

Fonte: Hughes *et al.* (2021).

É possível também representar o estado de um qubit em superposição (conforme a Equação 1) utilizando a álgebra matricial. Na representação matricial, o qubit é escrito como um vetor bidimensional, onde as amplitudes equivalem aos componentes do vetor (HUGHES *et al.*, 2021):

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (3)$$

Os estados $|0\rangle$ e $|1\rangle$ costumam ser representados como:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \quad (4)$$

Na representação matricial acima, o elemento superior da matriz representa a probabilidade do qubit colapsar para o estado 0, enquanto o elemento inferior representa a probabilidade de colapso para o estado 1 (BHAT *et al.*, 2022).

Conforme o número de qubits de um sistema quântico é aumentado, sua representação matricial torna-se mais complexa. Por exemplo, para n qubits, têm-se n^2 elementos dentro da matriz. Além disso, estarão presentes na representação matricial muitos elementos zero, uma vez que a matriz ficaria incompleta sem a presença de todos os seus elementos. A notação Dirac permite que esses elementos nulos sejam omitidos, tornando-a uma representação significativamente mais compacta que a matricial (BHAT *et al.*, 2022).

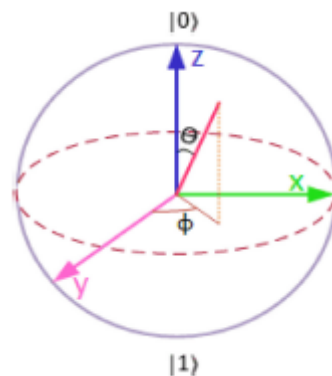
A seguir, as Equações 5 e 6 exibem a representação de dois qubits na forma de notação Dirac e matricial, respectivamente.

$$|\Psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle = \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{bmatrix} \quad (5)$$

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}; |01\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}; |10\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}; |11\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad (6)$$

Além das representações vistas anteriormente, o qubit também pode ser visualizado através da esfera de Bloch – uma representação geométrica do espaço de estados puros de um sistema de mecânica quântica de dois níveis (ver Figura 2).

Figura 2 – Representação do qubit na esfera de Bloch.



Fonte: Bhat *et al.* (2022).

De acordo com Hughes *et al.* (2021), cada ponto na superfície da esfera de Bloch corresponde a um estado puro (uma superposição distinta) para um único qubit, onde o polo norte e o polo sul equivalem aos estados da base computacional, $|0\rangle$ e $|1\rangle$, respectivamente. Ou seja, quando o vetor de estado não estiver apontando para um dos polos, o qubit encontra-se em superposição.

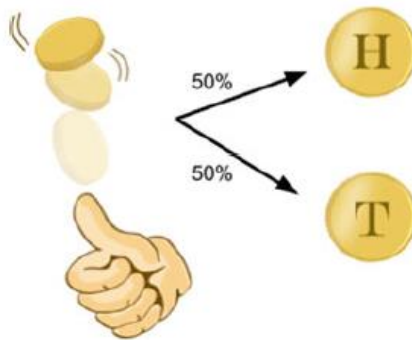
Dessa forma, a esfera de Bloch atua como um auxílio visual, facilitando o entendimento de que um qubit possui um número infinito de estados quânticos possíveis. Contudo, é necessário frisar que é possível representar através dela apenas um único qubit, não sendo possível utilizá-la em sistemas de dois ou mais qubits.

Por fim, vale ressaltar que a medição de um qubit resulta apenas em 0 ou 1. Além disso, a medição altera o estado do qubit, causando o colapso de superposição (combinação de $|0\rangle$ e $|1\rangle$) para o estado específico ($|0\rangle$ ou $|1\rangle$) consistente com o resultado medido (NIELSEN; CHUANG, 2011).

2.1.1.2 Superposição quântica

A superposição quântica pode ser definida como a capacidade que um sistema quântico possui de estar em múltiplos estados simultaneamente. Uma analogia comum para ilustrar esse fenômeno é a do lançamento de uma moeda. Ao lançarmos uma moeda no ar, pode-se afirmar que a probabilidade de ela cair como “cara” ou “coroa” é de 50%, conforme mostrado na Figura 3.

Figura 3 – Uma moeda lançada tem 50% de chance de cair em cara ou coroa.

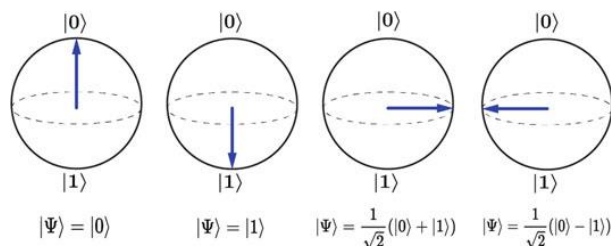


Fonte: Hughes *et al.* (2021).

Nesse caso, pode-se traçar um paralelo onde, enquanto a moeda está no ar, ela estaria em uma "superposição" de ambos os estados (cara e coroa) e, ao pousar, seu estado é definido (cara ou coroa). Este é um comportamento análogo ao observado em um qubit. Contudo, é importante notar que a incerteza da moeda é clássica, enquanto a do qubit é fundamental: ele realmente existe em ambos os estados.

Enquanto na computação clássica um bit possui natureza binária, isto é, seus estados são apenas 0 e 1, na computação quântica o qubit pode existir no estado 0, 1 ou em qualquer combinação linear desses estados (superposição), conforme mostrado na Figura 4.

Figura 4 - Diferentes estados de um qubit representados na esfera de Bloch.



Fonte: Hughes *et al.* (2021).

Dado o exposto, a superposição quântica é a capacidade que um qubit possui de estar em múltiplos estados simultaneamente até a realização da sua medição (SHAFIQUE; MUNIR; LATIF, 2024). Essa capacidade possibilita a execução do paralelismo quântico – isto é, de maneira simultânea, um único qubit em superposição representa uma combinação linear dos dois estados-base. Dessa forma, n qubits podem representar uma superposição de 2^n estados simultaneamente, permitindo que o sistema processe essa superposição de 2^n entradas de uma única vez. Esse processamento paralelo é o que oferece potencial para solucionar problemas que um computador clássico necessitaria de 2^n etapas (BHAT *et al.*, 2022).

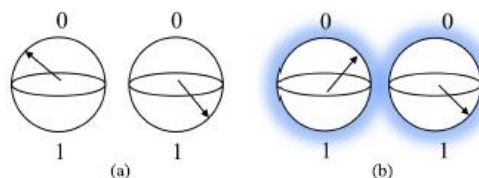
2.1.1.3 Emaranhamento quântico

Na computação clássica, o estado de cada bit é independente, e uma operação em um bit não causa qualquer influência no estado de outro. Na computação quântica, porém, os qubits podem estar correlacionados de forma que seus estados não possam ser descritos individualmente. O resultado da medição de um qubit estará intrinsecamente ligado ao resultado da medição do outro, independentemente da distância entre eles.

Esse fenômeno é conhecido como emaranhamento quântico, que se dá quando duas ou mais partículas estão correlacionadas de tal forma que o estado de um qubit está intrinsecamente dependente do estado de outro, apesar da distância existente entre eles (SHAFIQUE; MUNIR; LATIF, 2024). Assim, em um sistema emaranhado, a medição do estado de um qubit colapsa instantaneamente o estado dos demais qubits para um resultado correlacionado.

Dado um sistema de dois qubits, conforme a Figura 5a, cada qubit encontra-se em um estado de superposição individual, e seus estados são independentes. Logo, as probabilidades de medição de cada qubit são independentes entre si. Em situações em que os qubits estão emaranhados (Figura 5b), eles formam um único sistema quântico não-separável, e o resultado da medição de um está correlacionado ao do outro. A cor azul presente na Figura 5b indica que os qubits são partículas não-independentes, ou seja, formam um único sistema emaranhado.

Figura 5 – Dois qubits em estados quânticos não emaranhados (a) e emaranhados (b).



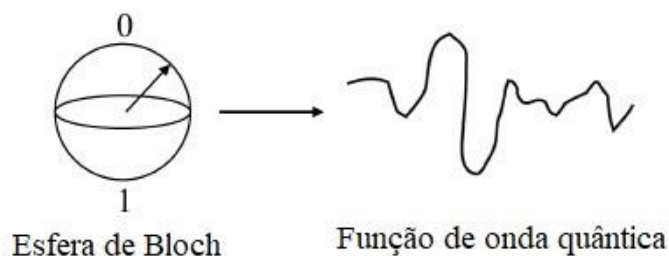
Fonte: Shafique, Munir e Latif (2024).

O emaranhamento quântico introduz correlações não locais entre partículas, violando o princípio do realismo local da física clássica. Essas correlações não clássicas representam um recurso fundamental que permite o processamento de informação quântica, proporcionando vantagens computacionais (TAO, 2024).

2.1.1.4 Interferência quântica

Conforme exposto acima, um qubit pode ser representado através da notação de bra-ket ou pela esfera de Bloch, sendo estas representações matemáticas do estado do qubit. O qubit, na verdade, possui uma natureza ondulatória descrita pela função de onda quântica, a qual satisfaz a equação de Schrödinger, conforme ilustra a Figura 6. De acordo com Shafique, Munir e Latif (2024), uma função de onda pode ser definida como uma descrição matemática do estado quântico, que contém as amplitudes de probabilidade complexas, as quais determinam as probabilidades correspondentes dos estados do sistema quântico.

Figura 6 – Natureza ondulatória de um qubit.

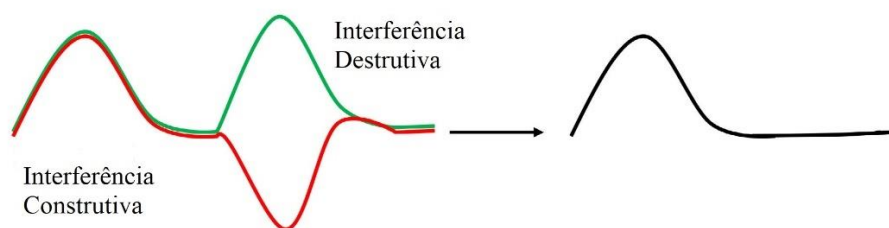


Fonte: Adaptado de Shafique, Munir e Latif (2024).

A natureza ondulatória do qubit é a base do fenômeno da interferência. Como as amplitudes de probabilidade são números complexos (possuindo amplitude e fase), elas não se somam como probabilidades clássicas. Em vez disso, elas podem se sobrepor quando diferentes caminhos quânticos levam a um mesmo estado.

Quando ocorre a sobreposição de duas ou mais amplitudes, elas interagem de maneira construtiva (amplitudes de mesma fase se somam) ou destrutiva (amplitudes de fases opostas se cancelam). Como consequência, a amplitude de probabilidade final é alterada, modificando a distribuição de probabilidade dos estados quânticos do sistema (ver Figura 7).

Figura 7 – Fenômeno da interferência quântica em duas funções de onda quântica.



Fonte: Adaptado de Shafique, Munir e Latif (2024).

Esse fenômeno é indispensável no desenvolvimento de algoritmos quânticos, pois, em conjunto com o paralelismo quântico (oriundo da superposição), os algoritmos são projetados para assegurar que as amplitudes de probabilidade correspondentes à resposta almejada sejam amplificadas através da interferência construtiva, enquanto as demais amplitudes sejam reduzidas por interferência destrutiva.

Entretanto, em virtude do fenômeno da decoerência, a manutenção da interferência transforma-se também em um desafio para a computação quântica. Os qubits são extremamente sensíveis a erros, e interações com o ambiente, por menores que sejam, podem ocasionar a perda de suas propriedades quânticas, como superposição e emaranhamento – processo conhecido como decoerência (WONG, 2022).

A decoerência, ou perda de coerência, provoca também uma quebra nos efeitos da interferência, o que torna a computação quântica propensa a erros. Neste contexto, Wong (2022) afirma que a decoerência é considerada o empecilho no desenvolvimento de computadores quânticos de larga escala, visto que é extremamente difícil isolar o qubit do ambiente externo e, ao mesmo tempo, mantê-lo acessível para a aplicação de portas quânticas e medições. Atualmente, são aplicadas inúmeras técnicas para correção de erros quânticos, visando a diminuição do impacto ocasionado pelo ambiente externo e preservando a interferência quântica, fundamental para a computação quântica.

Em suma, a interferência é um conceito primordial para a computação quântica, pois possibilita que os sistemas quânticos realizem operações computacionais por meio da atualização das distribuições de probabilidade dos estados quânticos (SHAFIQUE; MUNIR; LATIF, 2024). Isso viabiliza a resolução de determinados problemas de formas inalcançáveis por métodos da computação clássica.

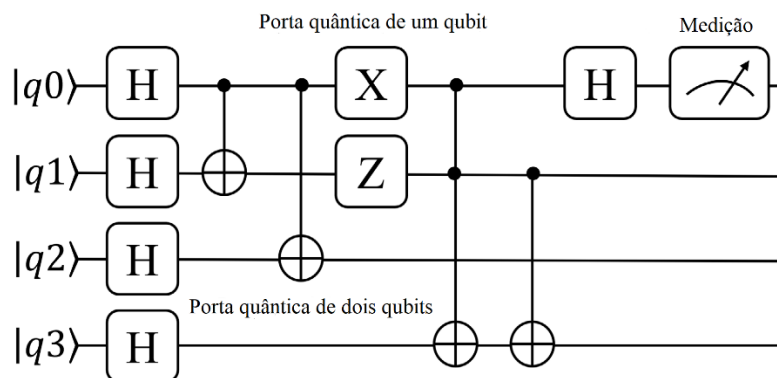
2.2 A interface de controle e leitura de qubits

2.2.1 Blocos de construção lógicos: portas quânticas e medição

As portas quânticas e a medição desempenham papéis fundamentais nos circuitos quânticos. As portas quânticas são responsáveis por realizar a manipulação dos qubits, transformando a informação quântica. Por outro lado, o processo de medição tem por objetivo extrair informações sobre o estado quântico do qubit.

A maioria dos computadores clássicos funciona através de um conjunto de bits, realizando o armazenamento de informações na memória em formato binário. Além disso, é possível alterar o estado desses bits por meio de portas lógicas como OR, AND, NOT, dentre outras. A computação quântica utiliza um modelo análogo ao da computação clássica, denominado de modelo de circuito ou modelo de portas. Nesse modelo, o computador quântico é criado a partir de um circuito quântico composto por fios e portas quânticas (Figura 8), possibilitando o transporte e a manipulação da informação quântica.

Figura 8 – Exemplo de um circuito quântico.



Fonte: Adaptado de Shafique, Munir e Latif (2024).

Segundo Wong (2022), as portas quânticas manipulam o estado de um qubit em circuitos quânticos. Assim, elas realizam um papel semelhante ao das portas lógicas clássicas anteriormente citadas, diferindo pelo uso dos princípios da mecânica quântica no processamento da informação quântica.

Dentre as portas quânticas fundamentais para operações de um, dois e três qubits têm-se, respectivamente: H , X , Y , Z , T , I e a porta de fase (por exemplo, a porta S); $CNOT$, CZ e $Swap$; Toffoli e Fredkin (BHAT *et al.*, 2022). Outrossim, portas lógicas tradicionais como OR,

AND, NOR, NAND e XOR podem ser implementadas através de portas quânticas de um e dois qubits. O funcionamento detalhado das portas quânticas pode ser consultado em Nielsen e Chuang (2011) e Shafique, Munir e Latif (2024).

Conforme supracitado, a Figura 8 apresenta um exemplo de um circuito quântico. Nele, um conjunto de qubits está sendo utilizados ($|q1\rangle$, $|q2\rangle$, $|q3\rangle$ e $|q4\rangle$), cujos estados podem estar superpostos e emaranhados. Além disso, algumas portas quânticas (H, X e Z) encontram-se disponíveis neste circuito, permitindo a realização de operações nos qubits e, conseqüentemente, a alteração da distribuição das probabilidades dos estados para este circuito quântico.

Ao final do circuito, é realizada a medição. A medição obtém a informação acerca do estado quântico do qubit no instante em que está sendo medido, convertendo os múltiplos estados probabilísticos (estados superpostos) em um estado absoluto (0 ou 1), bem como colapsando os demais estados e superposições (SHAFIQUE; MUNIR; LATIF, 2024).

Por exemplo, se um qubit está em uma superposição geral, como a definida na Equação 1 ($|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$), após a realização da sua medição, obtém-se (com probabilidade $|\alpha|^2$) o resultado clássico 0, e o estado pós-medição do qubit passa a ser $|0\rangle$. Esse comportamento é um dos postulados fundamentais da mecânica quântica. Para o circuito quântico analisado, a medição irá colapsar os estados de superposição e revelar o estado final do circuito, baseado na distribuição de probabilidade dos estados.

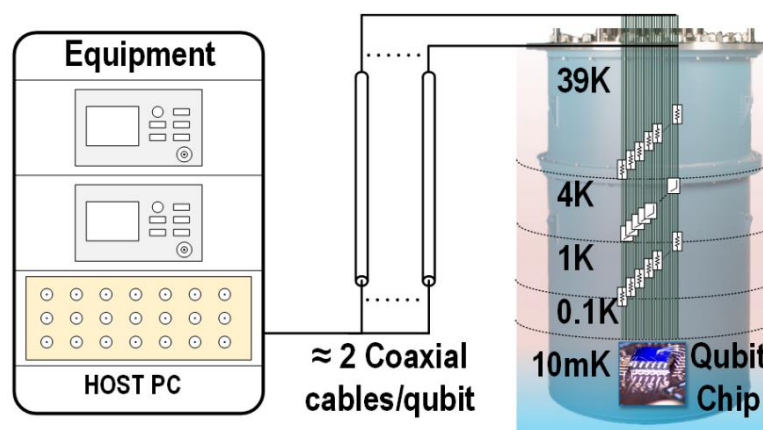
2.2.2 A implementação física de sistemas quânticos

A forma de interagir com um qubit varia de acordo com o tipo de implementação física do computador quântico. Para qubits de spin, por exemplo, a inicialização é feita aplicando um campo magnético estático de aproximadamente 1 Tesla, posicionando todos os spins para cima (isto é, representando o estado 0). Para alterar o estado do qubit, é aplicado um pulso eletromagnético em cada qubit individualmente – onde o estado final do qubit é dependente da amplitude e da duração do pulso eletromagnético que está sendo aplicado (BHAT *et al.*, 2022).

DiVincenzo (2000) estabeleceu cinco requisitos necessários para a implementação da computação quântica: sistema físico escalável, com qubits bem definidos; capacidade de inicializar os qubits em um estado fiducial simples; longos tempos de coerência; conjunto universal de portas quânticas e medição do estado do qubit.

Com base nestes requisitos, inúmeras tecnologias estão sendo aplicadas para demonstrar a computação quântica, com destaque para a computação quântica supercondutora e a computação quântica baseada em armadilhas de íons. Os sistemas de computação quântica utilizados atualmente costumam ser constituídos por um ou mais chips de qubit, eletrônica de controle e um computador clássico (que executa uma pilha de *softwares*) – conforme a Figura 9.

Figura 9 – Configuração atual de um computador quântico.



Fonte: Park *et al.* (2021).

A arquitetura descrita por Park *et al.* (2021), detalha que os chips com os qubits de estado sólido são alocados dentro de um refrigerador de diluição, com temperatura base típica de 10 mK. Segundo os autores, os qubits são controlados através de sinais analógicos de radiofrequência (RF), tais como pulsos de tensão CC e pulsos de micro-ondas, produzidos por instrumentos de controle situados fora do refrigerador, denominados de eletrônica de controle. Por fim, os sinais de controle são enviados ao chip com os qubits por meio de cabos coaxiais, utilizando aproximadamente dois cabos por qubit.

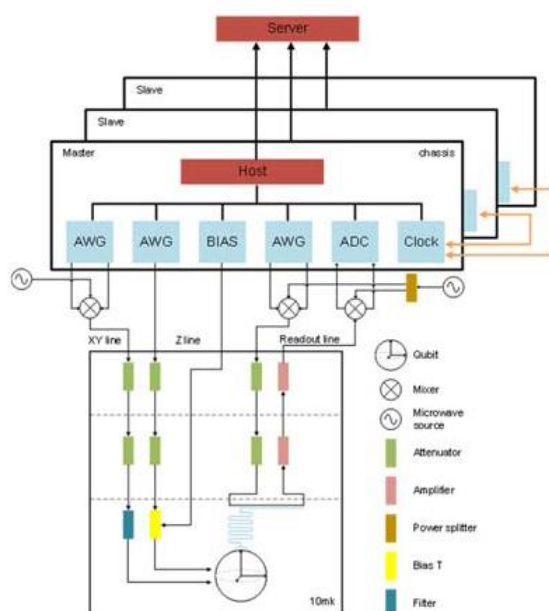
2.2.3 A interface clássico-quântica: eletrônica de controle e leitura

De acordo com Wiseman e Milburn (2010), a eletrônica clássica desempenha um papel imprescindível em qualquer arquitetura da computação quântica. Por exemplo, em um sistema de computação quântica supercondutora, a eletrônica atua como uma ponte, interligando os

comandos de controle clássicos e os processadores quânticos (YANG *et al.*, 2021). Além disso, segundo Yang *et al.* (2021), a imprecisão no controle e na medição pode prejudicar a acurácia das operações realizadas pelas portas quânticas, bem como a eficiência da medição dos estados dos qubits.

A Figura 10 apresenta o diagrama de blocos de um sistema de computação quântica supercondutor. O sistema é formado por diversos elementos, contendo um servidor, subsistema de controle, subsistema de medição e circuitos supercondutores em temperatura criogênica.

Figura 10 – Diagrama de blocos de um sistema de processador quântico supercondutor.



Fonte: Yang *et al.* (2021).

No subsistema de controle, o AWG é responsável pela geração de uma sequência de formas de onda de controle, que posteriormente é alterada para uma frequência mais alta por meio de um misturador (mixer), visando atingir uma radiofrequência próxima (ou ressoante) à frequência intrínseca do qubit. O pulso chega ao qubit através de cabos coaxiais de micro-ondas conectados ao criostato e, por meio da excitação com fótons de micro-ondas, é possível levar o qubit a um estado quântico arbitrário (YANG *et al.*, 2021).

Ainda segundo os autores, no subsistema de medição, o AWG cria um sinal de sonda, que em seguida é transformado para um sinal de micro-ondas de radiofrequência através de um misturador e enviado para os ressoadores supercondutores no processador por cabos coaxiais similares. Em seguida, o sinal transmitido pelo ressoador, que contém a informação do estado

do qubit, é amplificado e posteriormente enviado ao módulo de aquisição de dados (DAQ) após a demodulação IQ. Desta forma, o estado do qubit pode ser obtido através da análise de dados.

Diante do exposto, constata-se que o AWG é uma parte indispensável do sistema eletrônico na computação quântica, influenciando diretamente nas operações com portas quânticas e no processo de medição. A alta performance exigida para este dispositivo – em termos de precisão, velocidade e flexibilidade na geração de sinais complexos – torna o seu projeto um desafio central. Por essa razão, o funcionamento e as arquiteturas dos geradores de formas de onda arbitrárias (AWGs) serão analisados em detalhe a seguir.

2.3 Geradores de formas de onda arbitrárias

Segundo Zhengjun e Fuping (2020), os geradores de sinais são equipamentos eletrônicos capazes de produzir sinais de tensão em diferentes formatos, frequências e ciclos de trabalho – desempenhando um papel fundamental em etapas de teste e verificação comportamental de circuitos eletrônicos. Todavia, os modelos clássicos, como os geradores de função (function generator ou FG), apresentam limitações relevantes no que concerne às formas de onda que podem gerar. Além disso, esses equipamentos carecem de flexibilidade quando se trata de aplicações que demandam sinais mais complexos e personalizados.

Visando superar essas limitações, surge o gerador de formas de onda arbitrárias (do inglês *arbitrary waveform generator* ou AWG), uma solução moderna capaz de criar sinais sob medida – não apenas as formas de onda habituais, como senoidal, triangular e quadrada, fornecidas pelos geradores de sinais convencionais. Além disso, o AWG se diferencia por oferecer ao usuário maior configurabilidade, precisão e estabilidade na geração do sinal.

Comercialmente são encontrados inúmeros modelos de AWGs, tal como o AWG70000B da Tektronix (Figura 11). O equipamento possui taxa de amostragem de até 50 GS/s, resolução de 10 bits e largura de banda de até 18 GHz, permitindo a geração de sinais de alta fidelidade e complexidade.

Figura 11 – AWG modelo AWG70000B, da Tektronix.



Fonte: Tektronix (2025).

Entretanto, apesar da alta performance de instrumentos de ponta como o AWG70000B, a escalabilidade da computação quântica impõe um conjunto diferente de desafios. A crescente demanda é por AWGs que não sejam apenas programáveis, mas também compactos, de baixo custo e extensíveis para controlar centenas ou milhares de qubits. Estes requisitos de integração e escala não são plenamente atendidos pelos atuais instrumentos de bancada, o que limita o avanço da eletrônica de controle e medição e, conseqüentemente, dos computadores quânticos em larga escala (LIN *et al.*, 2019).

2.4 Síntese digital direta

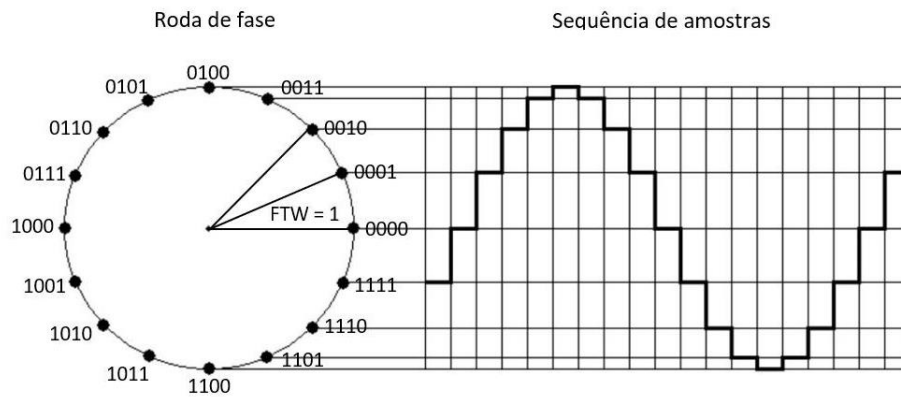
2.4.1 Arquitetura DDS básica

A síntese digital direta (do inglês, direct digital synthesis ou DDS) é uma técnica utilizada para a produção de formas de onda analógicas através da geração de um sinal digital com variação temporal, seguido por uma conversão digital-para-analógica. Dentre as principais vantagens apresentadas por dispositivos que empregam a técnica DDS na geração de sinais, destacam-se: alta resolução de frequência, rápida comutação de frequência, operação em um amplo espectro de frequências e baixo consumo de energia (MURPHY; SLATTERY, 2004).

Conforme pode ser visto na Figura 12, a estrutura básica para implementação da técnica DDS é composta por: um acumulador de fase (AF); um conversor de fase para amplitude, implementado através de uma tabela de consulta (look-up table ou LUT); um conversor digital-analógico (DAC) e um filtro.

A Figura 14 apresenta a geração de um sinal senoidal utilizando a técnica DDS com as seguintes configurações: acumulador de fase de 4 bits ($j = 4$), isto é, o AF tem capacidade de contar 16 valores ($2^j = 2^4 = 16$); FTW igual a 1, obrigando o AF a utilizar todas as amostras endereçadas, visto que o FTW controla o tamanho dos saltos realizados na roda de fase. Segundo Murphy e Slattery (2004), cada ponto presente na roda de fase corresponde a um ponto equivalente no ciclo de uma onda senoidal. À medida que a roda de fase é percorrida, o seno do ângulo gera uma onda senoidal de saída correspondente. Ou seja, uma rotação completa na roda de fase, em velocidade constante, gera um ciclo completo da onda senoidal de saída.

Figura 14 – Geração de uma senoide através da técnica DDS.



Fonte: Marin e Kaschny (2011).

Por fim, o conversor digital-analógico (DAC) e o filtro são responsáveis por finalizar o processo. O DAC converte as amostras digitais em um sinal analógico em degraus. O filtro (tipicamente um filtro passa-baixas) atua em seguida, suavizando este sinal para torná-lo contínuo e removendo o ruído e as frequências espúrias (VAZ, 2012).

De acordo com Vankka (2000), a frequência do sinal gerado (f_{out}) pode ser obtida através da Equação 7:

$$f_{out} = \frac{FTW \times f_{clk}}{2^j} \quad (7)$$

Onde FTW representa a palavra de ajuste de frequência (ou incremento de fase), j a largura em bits do acumulador de fase (AF) e f_{clk} é o sinal de referência (ou frequência de clock).

Ainda segundo o autor, a resolução de frequência (Δf) pode ser obtida definindo $FTW = 1$, conforme indicado na Equação 8.

$$\Delta f = \frac{f_{clk}}{2^j} \quad (8)$$

Ademais, ao empregar a técnica DDS, deve ser seguido o teorema de Nyquist. Esse teorema estabelece que a frequência de clock (f_{clk}) deve ser pelo menos o dobro da frequência do sinal gerado, evitando assim a ocorrência do fenômeno aliasing (OPPENHEIM; SCHAFER, 2013).

$$f_{clk} \geq 2 \times f_{out} \quad (9)$$

2.4.1.1 Truncamento de fase

O tamanho da LUT está diretamente ligado ao comprimento do AF (j bits). Dessa forma, caso j seja muito grande, consequentemente a LUT também será, o que não é desejável, uma vez que o tamanho da LUT ocasiona diminuição da velocidade do DDS e o aumento do consumo de energia. Visando reduzir o tamanho da LUT, realiza-se a técnica de truncamento de fase (HENRIQUE, 2022).

Conforme visto anteriormente, a fase de uma onda é acumulada progressivamente pelo AF. Entretanto, o valor total da fase geralmente possui mais bits que os endereços da forma de onda armazenada na LUT. Por esse motivo, é realizado o truncamento de fase, descartando os bits menos significativos da fase antes de acessar a tabela de consulta. O truncamento possibilita a redução do tamanho da LUT (consequentemente, economia de *hardware*) e da latência.

De acordo com Henrique (2022), em um sistema DDS prático, apenas os 10 a 15 bits mais significativos provenientes do AF chegam à LUT, ocasionando uma redução no tamanho da LUT, sem afetar a resolução de frequência. Todavia, o truncamento de fase adiciona ruído de truncamento, o que gera harmônicos espúrios (*spurs*) no sinal de saída.

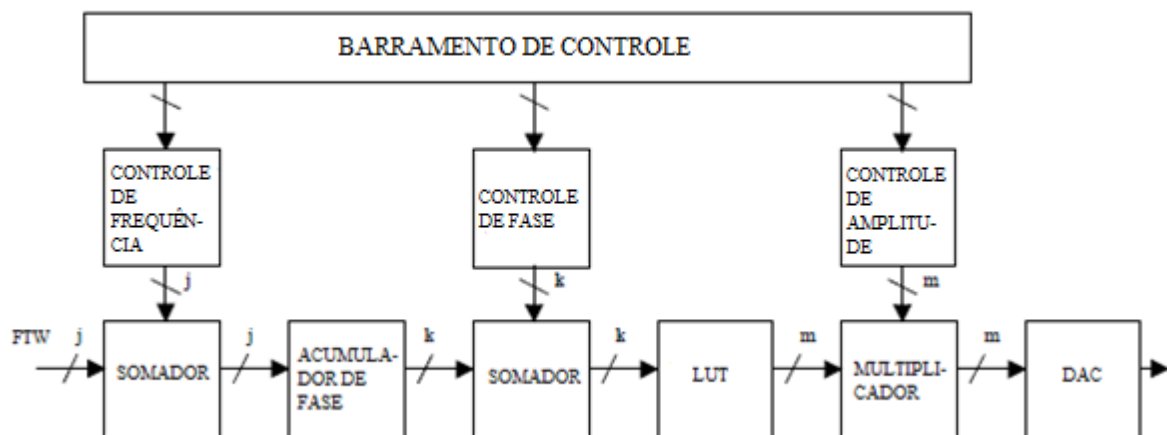
2.4.2 Arquitetura DDS com controle de fase, amplitude e frequência

Segundo Vankka (2000), é factível a adição de recursos de controle ao DDS, por se tratar de um dispositivo de processamento digital. Dessa forma, é possível variar os três parâmetros da forma de onda – isto é, fase, amplitude e frequência.

$$s(n) = A(n)\sin(2\pi(FTW(n) + F(n))) \quad (10)$$

Na Equação 10, $A(n)$ representa o controle de amplitude, $FTW(n)$ o controle de frequência e $F(n)$ o controle de fase. A Figura 15 apresenta um diagrama de blocos de um sistema DDS básico com controles de frequência, amplitude e fase implementadas. A variação de frequência é alcançada colocando um somador antes do acumulador de fase. A de fase também é feita através de um somador, posicionando-o entre o acumulador de fase e a LUT. Por fim, a variação de amplitude é alcançada através da inserção de um multiplicador entre a LUT e o DAC – o multiplicador é responsável por manipular o tamanho da amostra digital que chega ao DAC (VANKKA, 2000).

Figura 15 – Arquitetura DDS com capacidade de controle de frequência, fase e amplitude.



Fonte: Vankka (2000).

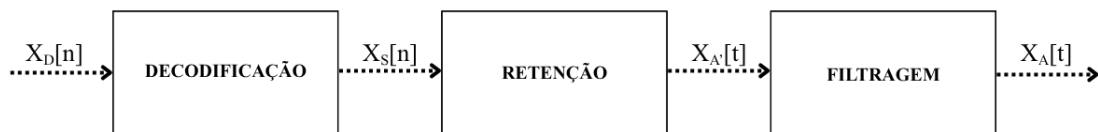
2.5 Conversão digital para analógico

Conforme visto anteriormente, a geração de sinais através da técnica DDS equivale a um processamento digital de sinais. Diante disso, para que seja gerado um sinal analógico pelo AWG, é fundamental que os valores digitais utilizados ao longo do processamento sejam transformados em tensão. O processo responsável por essa conversão é conhecido como conversão digital para analógica (D/A).

2.5.1 Teoria de funcionamento

O processo de conversão D/A é representado através do diagrama de blocos da Figura 16, sendo composto pelas seguintes etapas: decodificação, retenção e filtragem.

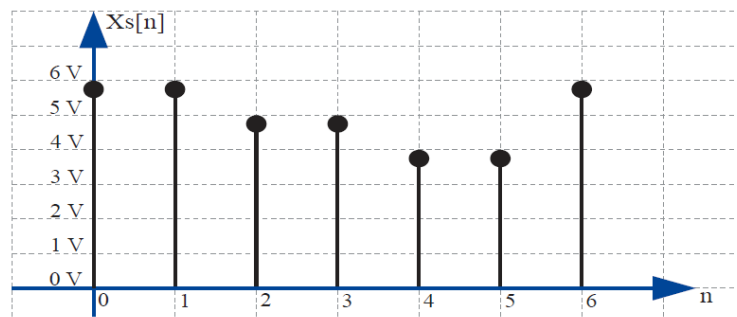
Figura 16 – Diagrama de blocos do processo de conversão D/A.



Fonte: Autoria própria (2025).

No processo de decodificação, atribui-se a cada sinal digital de entrada $X_D[n]$ (de acordo com o código binário por ele representado), determinado nível de corrente ou tensão, onde n representa um número natural. No contexto da técnica DDS, $X_D[n]$ equivale a uma amostra de contida em determinado endereço da LUT. Assim, após a etapa de decodificação, é obtido o sinal de tempo discreto e quantizado em amplitude $X_S[n]$ (conforme a Figura 17).

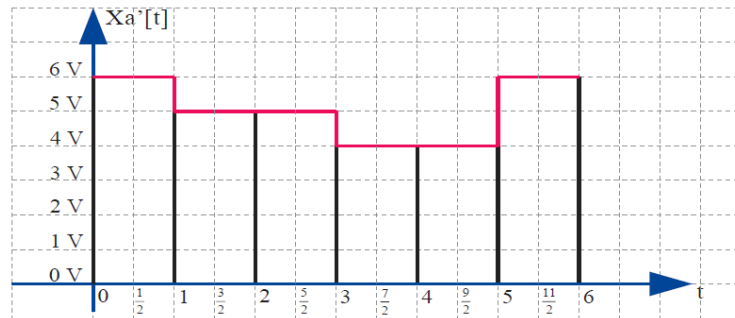
Figura 17 – Sinal $X_S[n]$.



Fonte: Santos e Lampa (2018).

Posteriormente, o $X_s[n]$ passa pelo processo de retenção, comumente implementado por um segurador de ordem zero. Este circuito mantém o valor da última amostra até a chegada de uma nova, conectando as amostras consecutivas e produzindo o sinal em formato de "escada", como pode ser visto na Figura 18.

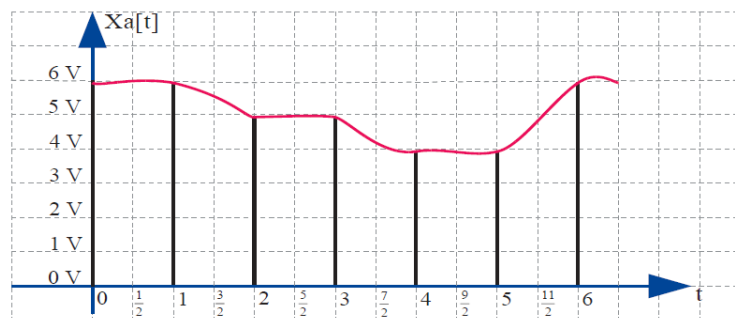
Figura 18 – Sinal $X_A'[t]$.



Fonte: Santos e Lampa (2018).

Após as duas primeiras etapas, é obtido o sinal $X_A'[t]$, que é um sinal de tempo contínuo e amplitude quantizada. Diante disso, na última etapa é aplicado ao sinal um filtro passa-baixas, que suaviza o sinal removendo as componentes de alta frequência introduzidas pelo ZOH, gerando o sinal analógico final $X_A[t]$ (Figura 19).

Figura 19 – Sinal $X_A[t]$.



Fonte: Santos e Lampa (2018).

2.5.2 Conversores digital para analógico

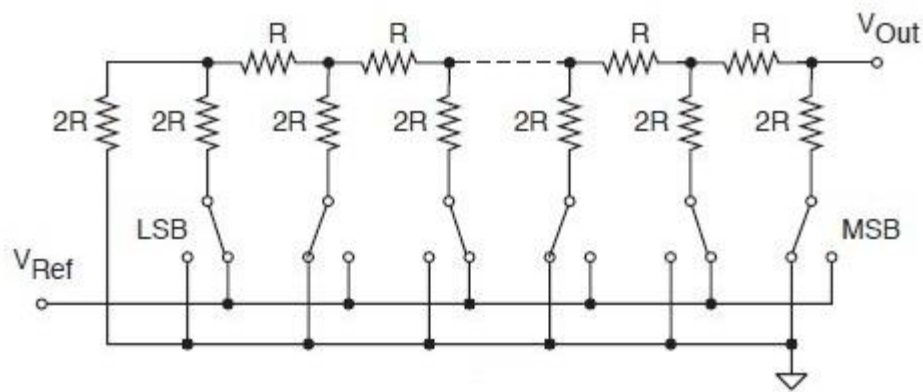
Um conversor digital-analógico (do inglês, digital-to-analog converter ou DAC) é um dispositivo empregado no processo de conversão D/A descrito no tópico anterior. O DAC

associa a palavra digital presente na sua entrada a um nível de tensão ou corrente correspondente obtido entre os limites de sua tensão de alimentação. Assim, a resolução de um DAC é especificada por meio da quantidade de bits do sinal de entrada, enquanto a amplitude máxima do sinal é especificada pela tensão de alimentação (VAZ, 2012).

É indispensável selecionar corretamente o DAC conforme a sua aplicação, garantindo que as características de desempenho deste dispositivo, como resolução e taxa de amostragem, atendam aos requisitos do sistema no qual ele será empregado. Existem inúmeras arquiteturas de DAC, como por exemplo o DAC R-2R Ladder – um dos conversores DAC mais simples. Esse conversor pode ser construído usando apenas dois valores de resistência, sendo escalável para qualquer número de bits. Outrossim, a sua impedância de saída sempre será igual a R , isto é, independe do número de bits, facilitando a filtragem (HENRIQUE, 2022).

O conversor R-2R Ladder é um tipo de DAC que consiste, sobretudo, em uma conversão ponderada de bits digitais para níveis de tensão. Esse processo é feito através de uma cadeia de resistores com valores de resistência R e $2R$. A Figura 20 apresenta uma arquitetura padrão para este modelo de conversor, onde os bits advindos da entrada do DAC são responsáveis por comutar uma chave para a tensão de referência (V_{Ref}) ou para o terra, gerando um valor de tensão (V_{Out}) de acordo com o peso do bit referente a cada braço (MALOBERTI, 2007).

Figura 20 – Conversor DAC R-2R Ladder no modo de tensão.



Fonte: Maloberti (2007).

De acordo com Maloberti (2007), a saída do DAC R-2R Ladder em modo de tensão é dada pela superposição de termos que são a divisão sucessiva da tensão de referência (V_{Ref}) por 2. Dessa forma, para n bits, tem-se:

$$V_{Out} = \frac{V_{Ref}}{2} b_{n-1} + \frac{V_{Ref}}{4} b_{n-2} + \dots + \frac{V_{Ref}}{2^{n-1}} b_1 + \frac{V_{Ref}}{2^n} b_0 \quad (11)$$

A Equação 11 acima corresponde à conversão D/A de uma entrada digital $\{b_{n-1}, b_{n-2}, \dots, b_1, b_0\}$.

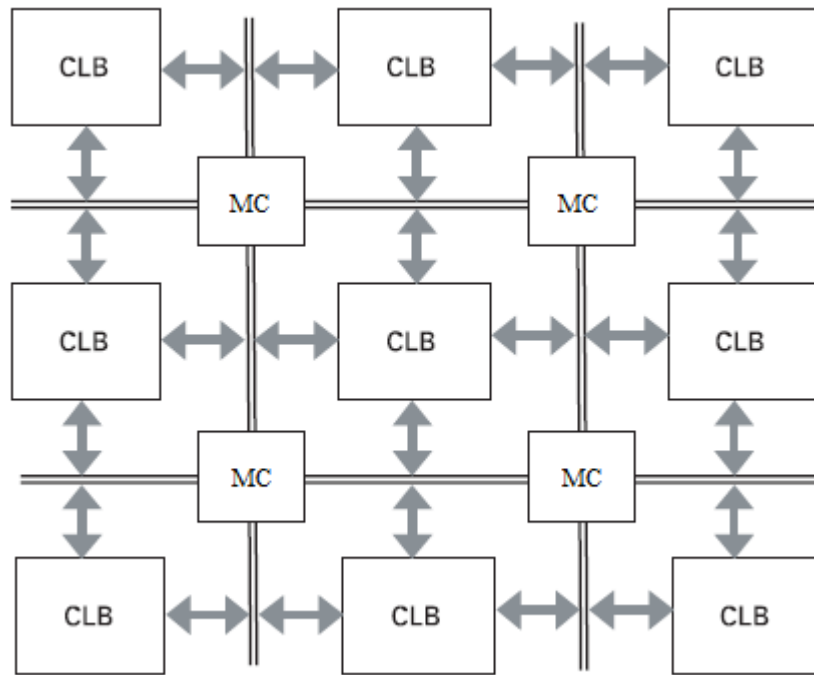
2.6 FPGAs

Visando construir circuitos lógicos combinacionais programáveis, surgiram durante a década de 1970 os PLDs (programmable logic devices ou dispositivos lógicos programáveis) – chips de propósito geral, nos quais o *hardware* pode ser configurável de acordo com as especificações desejadas. Um dos primeiros PLDs foi o PAL (programmable array logic ou lógica de matriz programável), que possuía apenas portas lógicas e possibilitava somente a implementação de circuitos combinacionais. Para superar esse problema, foram criados PDLs com registradores, permitindo a implementação de funções sequenciais simples graças à presença de um flip-flop (PEDRONI, 2004).

Nos anos seguintes, esses dispositivos passaram por inúmeras melhorias e evoluções, que culminaram no surgimento dos FPGAs (field programmable gate arrays ou arranjo de portas programáveis em campo) – um circuito integrado projetado para ser reprogramável, com capacidade de implementar circuitos de grande porte e alto desempenho, sem a necessidade de qualquer modificação física em seu *hardware* (PEDRONI, 2004).

Essa importante característica de reconfigurabilidade foi alcançada através de uma arquitetura básica composta por uma matriz de blocos lógicos configuráveis (CLBs), interconectados por um arranjo de matrizes de chaveamento (MCs), conforme ilustra a Figura 21. Um CLB é composto por LUTs, registradores e multiplexadores configuráveis (HARRIS; HARRIS, 2007), enquanto as matrizes de chaveamento (ou interconexões programáveis) possibilitam a conexão de diferentes CLBs dentro do dispositivo (VAHID, 2008). De acordo com Vahid (2008), os FPGAs disponíveis no mercado podem conter centenas ou até milhares de CLBs e MCs, evidenciando a alta capacidade lógica, flexibilidade e reconfigurabilidade presentes nesses dispositivos.

Figura 21 – Arquitetura básica de um FPGA.



Fonte: Adaptado de Pedroni (2004).

Segundo Sadeghi (2024), os FPGAs possuem um conjunto de características que os transformam em uma tecnologia valiosa em aplicações que exigem alto desempenho. Esses dispositivos propiciam benefícios como configurabilidade e flexibilidade superiores em relação aos ASICs (application-specific integrated circuit ou circuito integrado de aplicação específica), escalabilidade, capacidade de processamento paralelo inerente à sua arquitetura e, por consequência, baixa latência e capacidade de processamento em tempo real.

2.6.1 Linguagem de descrição de *hardware*

Por muito tempo, processos como simplificação de tabelas-verdade ou equações booleanas, bem como a tradução de máquinas de estados finitos (FSMs) em portas, eram feitos manualmente, tornando a busca por um conjunto adequado de portas lógicas para a realização de determinada função extremamente trabalhoso e propenso a erros (HARRIS; HARRIS, 2007).

Diante desta problemática, surgiram durante a década de 1990 as linguagens de descrição de *hardware* (do inglês, hardware description languages ou HDLs). As HDLs possibilitaram trabalhar em um nível de abstração mais alto, permitindo ao projetista especificar

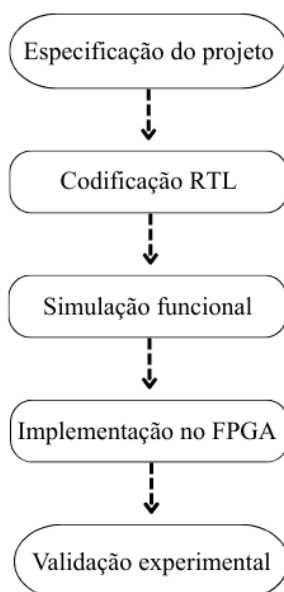
uma função lógica enquanto uma ferramenta de projeto auxiliado ao computador (CAD) produz as portas otimizadas. Isso possibilita o projeto e desenvolvimento de sistemas digitais complexos de maneira mais eficiente. Dentre as principais HDLs, destacam-se o Verilog e o VHDL (HARRIS; HARRIS, 2007).

Portanto, as linguagens de descrição de *hardware* possibilitam não apenas a descrição das interconexões estruturais entre os elementos do sistema, como também a inclusão de métodos que permitem a descrição dos comportamentos destes elementos. Assim, as HDLs se fazem presentes em todos os estágios de um projeto digital moderno, evidenciando a sua importância nessa área (VAHID, 2008).

3 METODOLOGIA

Este trabalho teve como foco a implementação digital de um gerador de formas de onda arbitrárias (AWG), utilizando a técnica de síntese digital direta (DDS) e almejando aplicação na área da computação quântica. Para o desenvolvimento correto deste projeto, seguiram-se as etapas apresentadas no fluxograma da Figura 22.

Figura 22 – Fluxograma do desenvolvimento do projeto.



Fonte: Autoria própria (2025).

Inicialmente, realizou-se uma revisão bibliográfica acerca da técnica de síntese digital direta (DDS), buscando compreender o comportamento dos blocos funcionais que constituem este método, bem como o seu princípio de funcionamento na geração de sinais analógicos. Em seguida, foram analisadas diferentes arquiteturas DDS (incluindo controle de fase, amplitude e frequência) e suas utilizações em AWGs. Esses estudos auxiliaram na definição da arquitetura e das especificações técnicas do projeto.

Definidas as especificações, realizou-se a codificação em nível RTL (register transfer level), descrevendo o comportamento do AWG através de uma linguagem de descrição de *hardware*. A HDL utilizada foi o Verilog, escolhida por sua sintaxe concisa e ampla adoção no setor industrial.

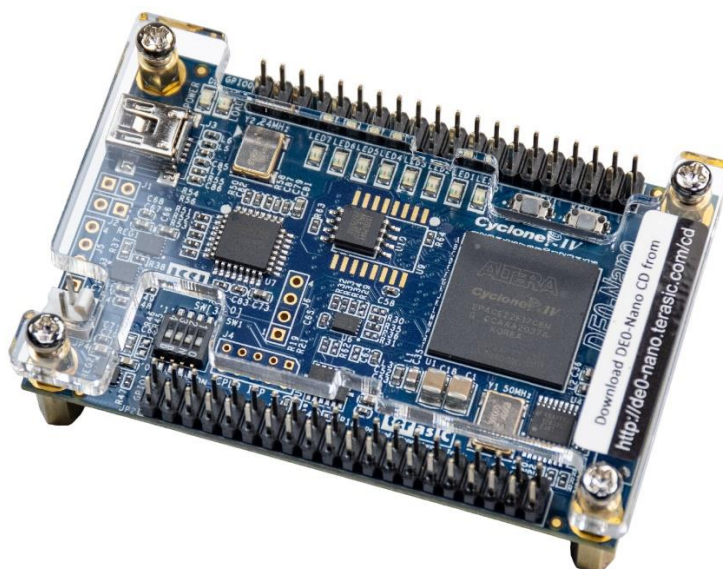
Como ferramenta de desenvolvimento do código, utilizou-se o *software* Quartus II da Intel (anteriormente pertencente à Altera). Este *software* dispõe de um ambiente propício para

projetos baseados em FPGAs da Altera, oferecendo ferramentas para as etapas de criação do projeto, compilação do código, simulação e implementação. Neste ambiente, é possível realizar a descrição de *hardware* utilizando HDLs (como Verilog, VHDL) ou através de diagramas esquemáticos.

Sucessivamente, foi efetuada a etapa de simulação funcional, testando o projeto em um ambiente virtual. Com o auxílio de um *testbench*, é possível comprovar se a lógica e as funcionalidades anteriormente descritas na codificação RTL cumprem as especificações do projeto antes de prosseguir para a implementação física. Para isso, utilizou-se o *software* ModelSim, um ambiente de simulação de hardware amplamente difundido no âmbito acadêmico.

Após o término da simulação funcional, iniciou-se o processo de implementação do projeto no FPGA. Para esta finalidade, utilizou-se o *kit* de desenvolvimento e prototipagem DE0-Nano da Altera (Figura 23) que possui, dentre outros componentes, um FPGA Cyclone IV EP4CE22F17C6 e um oscilador de clock integrado de 50 MHz. Nesta etapa, realizou-se a atribuição dos pinos para os sinais de entrada e saída do AWG a serem utilizados na fase de validação experimental.

Figura 23 – *Kit* de desenvolvimento DE0-Nano.

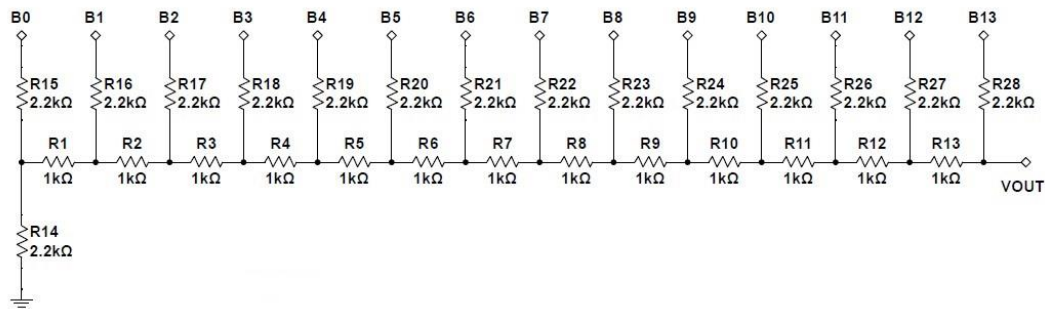


Fonte: Terasic (2025).

Por fim, realizou-se a etapa de validação experimental do projeto. Para isso, montou-se um *setup* de teste integrando o FPGA a um conversor digital-analógico R-2R (Figura 24), conectando a saída analógica desse circuito ao dispositivo Analog Discovery 2 (AD2) para

análise em tempo real. O AD2 (Figura 25) é um instrumento de bancada multifuncional, desenvolvido pela Digilent em parceria com a Analog Devices, que possibilita a medição, visualização e controle de diversos circuitos.

Figura 24 – Esquemático do conversor DAC R-2R.



Fonte: Autoria própria (2025).

Figura 25 – Dispositivo Analog Discovery 2.



Fonte: Digilent (2025).

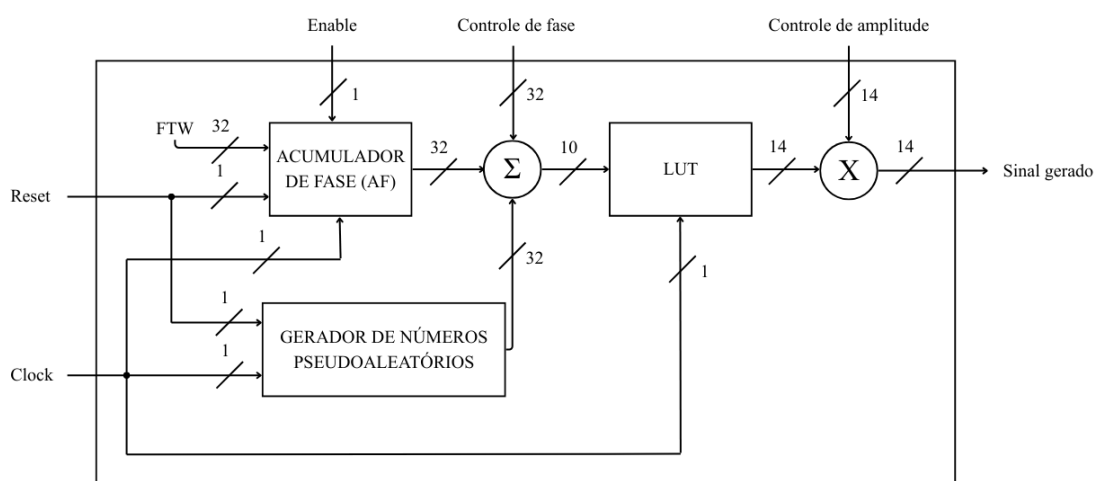
Neste *setup*, o aparelho AD2 foi empregado para enviar ao FPGA sinais de controle responsáveis por modificar os parâmetros de entrada do AWG, bem como para monitorar o sinal gerado. Dessa maneira, o dispositivo atuou também como um osciloscópio. Ressalta-se que o DAC R-2R, montado em protoboard e responsável por realizar a conversão digital-analógico, amostra os sinais em uma faixa de 0 a 3.3 V. A Figura 26 a seguir apresenta o diagrama de montagem do *setup* usado durante a etapa de validação experimental.

4 DESENVOLVIMENTO DO AWG

Após consolidação dos estudos acerca da técnica de síntese digital direta, suas principais arquiteturas e sua aplicação em geradores de formas de onda arbitrárias, iniciou-se a implementação do gerador capaz de produzir ondas senoidais controladas em fase e amplitude. A seguir, será abordado detalhadamente todos os blocos que compõem o gerador, cujo desenvolvimento foi feito no *software* Quartus II utilizando Verilog.

O esquema geral do AWG é formado por um acumulador de fase (AF), uma LUT e um gerador de números pseudoaleatórios (conforme ilustra a Figura 27), tendo como entradas os sinais de *clock*, *reset*, *enable*, controle de fase e controle de amplitude, bem como sinal gerado como saída. O *clock*, fornecido pelo *kit* de desenvolvimento DE0-Nano, é responsável por prover o sincronismo entre todos os blocos que compõem o gerador. O sinal de *reset*, ativo em nível lógico baixo, realiza a reinicialização do AF e do gerador de números pseudoaleatórios, assegurando um estado inicial previsível. O *enable* também auxilia no *reset* do AF, além de impedir a geração de sinal quando desativado. Os sinais de controle de fase e de amplitude permitem variar, em tempo real, as características do sinal produzido. Por fim, o sinal gerado corresponde à saída digital do AWG.

Figura 27 – Diagrama de blocos para o AWG.



Fonte: Autoria própria (2025).

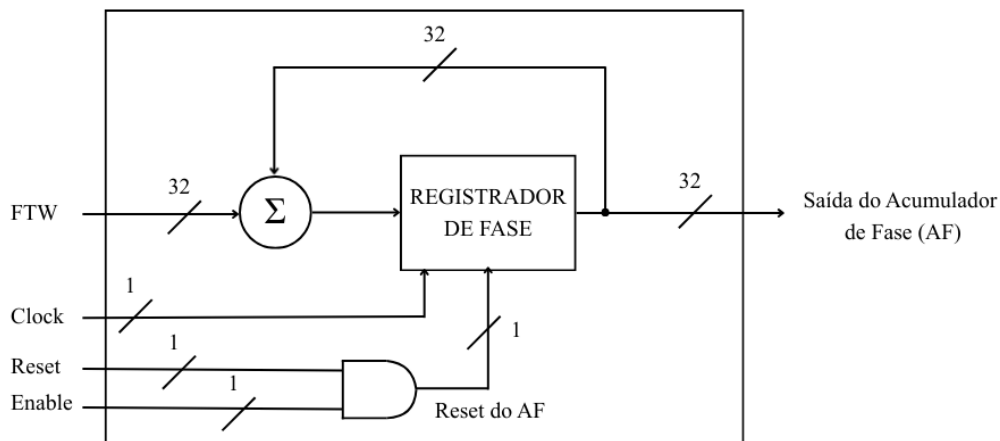
A seguir, serão abordados detalhadamente todos os blocos que compõem o gerador de formas de onda arbitrárias (AWG).

4.1 Acumulador de fase (AF)

O acumulador de fase (AF) é um dos blocos indispensáveis na estrutura AWG. Por meio da palavra de ajuste de frequência (FTW) e do sinal de clock o AF produz, a cada ciclo de *clock*, amostras de fase que serão posteriormente convertidas para amplitude pela LUT. Para este projeto, utilizou-se o *clock* de 50 MHz fornecido pelo oscilador da placa DE0-Nano. Além disso, fixou-se a frequência do sinal gerado em 1 MHz e definiu-se a largura do AF em 32 bits. Estabelecidos esses parâmetros, obteve-se, por meio da Equação 7, a FTW adequada para geração do sinal com essas características ($FTW = 32'd85899346$). Os sinais de *reset* e *enable* controlam a reinicialização do AF – *enable* é ativo em nível lógico alto. Além disso, o *enable* é responsável por sincronizar o início da geração do sinal, garantindo a coerência de fase.

Vale ressaltar que, posteriormente, a frequência do sinal gerado deverá ser alterada por meio de misturadores a fim de atingir a frequência de ressonância do qubit, viabilizando a manipulação ou medição do seu estado quântico.

Figura 28 – Diagrama de blocos para o acumulador de fase.



Fonte: Autoria própria (2025).

4.2 Gerador de números pseudoaleatórios

Conforme visto anteriormente, o tamanho da LUT está diretamente relacionado a largura do AF. Ou seja, quanto maior o AF, maior será a LUT. Contudo, isso pode impactar negativamente no funcionamento do AWG, provocando um aumento no consumo de recursos (memória e *hardware*), de energia e na latência. Dessa forma, visando reduzir o tamanho da

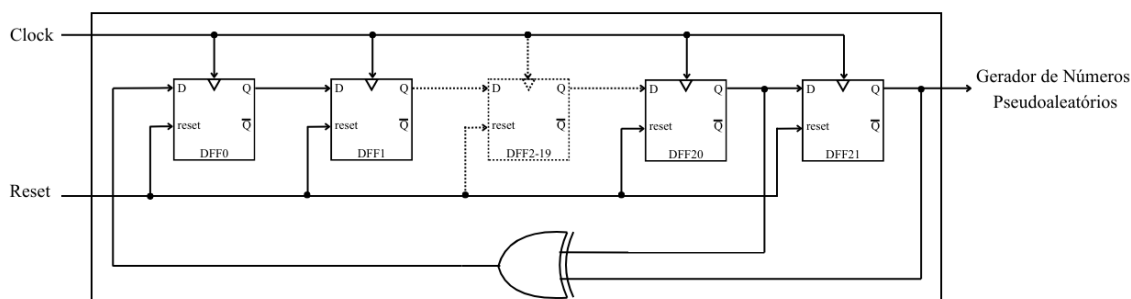
LUT, realizou-se o truncamento de fase, descartando os 22 bits menos significativos do AF e utilizando apenas os 10 bits mais significativos para endereçar a LUT.

Entretanto, de acordo com Torosyan e Willson Junior (2005), o truncamento de fase produz erros determinísticos e periódicos no sinal gerado, que se manifestam por meio de um conjunto de componentes espectrais indesejadas, denominadas frequências espúrias (do inglês, spurs). Esses espúrios são extremamente prejudiciais à pureza espectral do sinal gerado e, no contexto da computação quântica, podem interferir em outras frequências de ressonância, gerando erros durante a medição ou controle dos qubits.

Visando reduzir os erros de quantização de fase provenientes do truncamento e melhorar a pureza espectral do sinal gerado, aplicou-se a técnica de *dithering* de fase. Essa técnica consiste em adicionar uma pequena quantidade de ruído aleatório à saída do acumulador de fase antes do truncamento. Ao realizar essa operação, randomizam-se os bits que seriam descartados, eliminando a periodicidade dos erros responsáveis pela geração dos espúrios e transformando a energia desse erro em ruído de fundo, uma condição espectralmente mais favorável (ANALOG DEVICES, 2004).

O ruído aleatório requisitado pela técnica de *dithering* de fase foi produzido através do gerador de números pseudoaleatórios composto por 22 *flip-flops* tipo D (ver Figura 29). Esse gerador consiste em um registrador de deslocamento com realimentação linear (do inglês, linear feedback shift register ou LFSR). Uma vez que o erro de truncamento é gerado em razão do descarte dos 22 bits menos significativos oriundos do AF, o LFSR é responsável por gerar um valor aleatório de 22 bits a ser somado à saída do AF antes do truncamento.

Figura 29 – Gerador de números pseudoaleatórios.



Fonte: Autoria própria (2025).

4.3 LUT

A LUT (look-up table) é responsável por converter os valores de fase em valores de amplitude. Para isso, foram calculados previamente via MATLAB valores de amplitude da função seno para um período completo da onda. Posteriormente, a LUT foi instanciada no Quartus II como uma memória ROM, por meio da ferramenta *MegaWizard*. Os valores oriundos do AF funcionam como índices (endereços) para a LUT, onde cada endereço contém o valor de seno correspondente. A LUT gerada possui 10 bits de endereçamento ($2^{10} = 1024$ posições) e amostras de 14 bits de resolução.

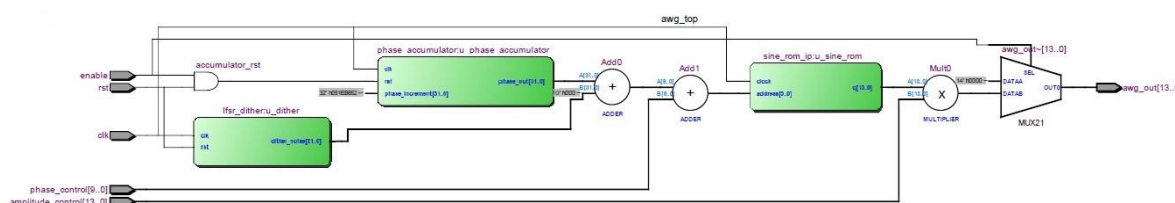
Antes de chegar à LUT, é realizado o controle de fase do sinal gerado, somando ao valor atual do AF o valor presente na entrada de controle de fase. Ao realizar essa soma, altera-se o endereço a ser lido na memória e, conseqüentemente, realiza-se um deslocamento temporal na onda.

Por fim, o controle de amplitude é realizado após a LUT, onde o valor da amostra senoidal é multiplicado pelo coeficiente presente na entrada de controle de amplitude.

5 RESULTADOS

No *software* Quartus II, realizou-se a codificação RTL do projeto em Verilog e, posteriormente, realizou-se o processo de implementação. Uma das subetapas realizadas durante esse processo é a síntese, na qual a plataforma de desenvolvimento produz um diagrama de blocos em nível RTL (uma representação esquemática gráfica do projeto), conforme a Figura 30.

Figura 30 – Diagrama de blocos para o AWG produzido pelo Quartus II.



Fonte: Autoria própria (2025).

Além do diagrama de blocos, gera-se um relatório que registra inúmeras informações acerca do projeto, como total de bits de memória e elementos lógicos do FPGA necessários para a implementação do projeto, análise de tempo e frequência. Foram utilizados 100 elementos lógicos e 14.336 bits de memória, o que representam, respectivamente, menos de 1 e 2% do total disponível no FPGA utilizado (o Cyclone IV – EP4CE22F17C6).

As simulações e testes necessários para validação do projeto foram realizados em duas etapas. Inicialmente, utilizando-se do *software* ModelSim e de um *testbench*, realizou-se a simulação funcional do código implementado e a análise dos resultados obtidos durante esta etapa. Em seguida, através de um *setup* composto por o *kit* de desenvolvimento DE0-Nano, DAC R-2R e Analog Discovery 2, realizou-se a etapa de validação experimental do projeto. Ambas as etapas são detalhadas a seguir através dos tópicos 6.1 e 6.2.

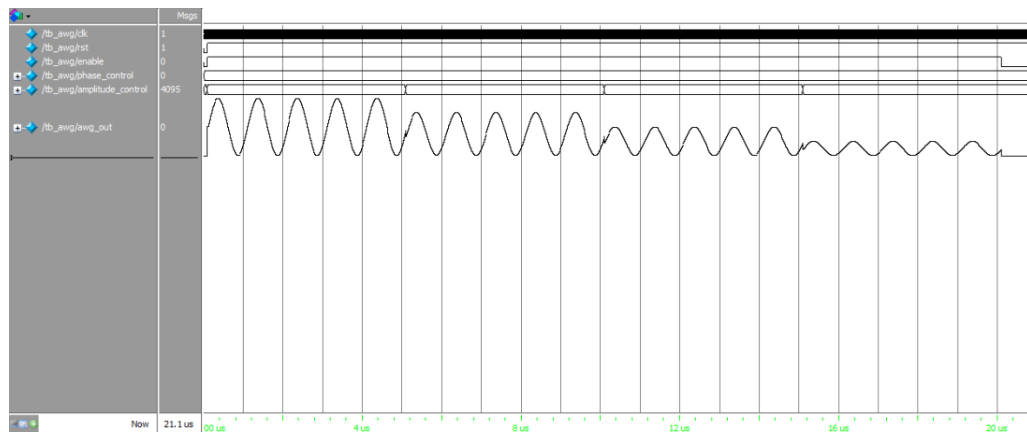
5.1 Simulação funcional

Utilizando-se o *software* ModelSim, geraram-se, por meio de um *testbench*, os sinais necessários para execução da simulação funcional do gerador de formas de onda arbitrárias (AWG) implementado digitalmente em Verilog. Para isso, foram gerados os seguintes sinais: *clock* (50 MHz, o mesmo fornecido pelo DE0-Nano); *rst*; *enable*; *amplitude control* e

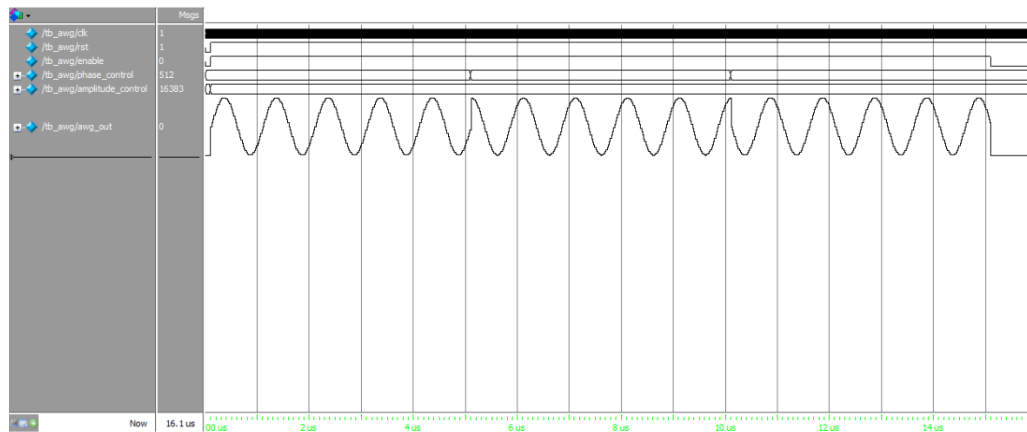
phase_control (responsáveis por controlar, respectivamente, os parâmetros de amplitude e fase da onda gerada); a onda gerada na saída do AWG foi observada por meio do sinal *awg_out*.

A Figura 31 apresenta as formas de onda obtidas durante a simulação.

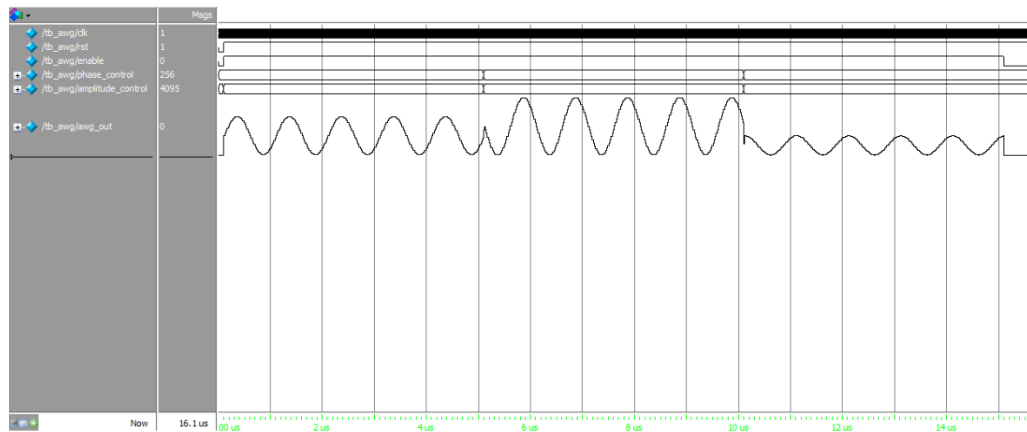
Figura 31 – Formas de onda geradas durante a simulação: (a) controle de amplitude; (b) controle de fase; (c) controle de amplitude e fase.



(a)



(b)



(c)

Fonte: Autoria própria (2025).

Na Figura 31a são realizados testes de controle de amplitude. É possível notar que, a através da entrada *amplitude_control*, é possível alterar a amplitude da senoide que está sendo gerada. No teste em análise, foram geradas senoides com amplitude de 100, 75, 50 e 25%.

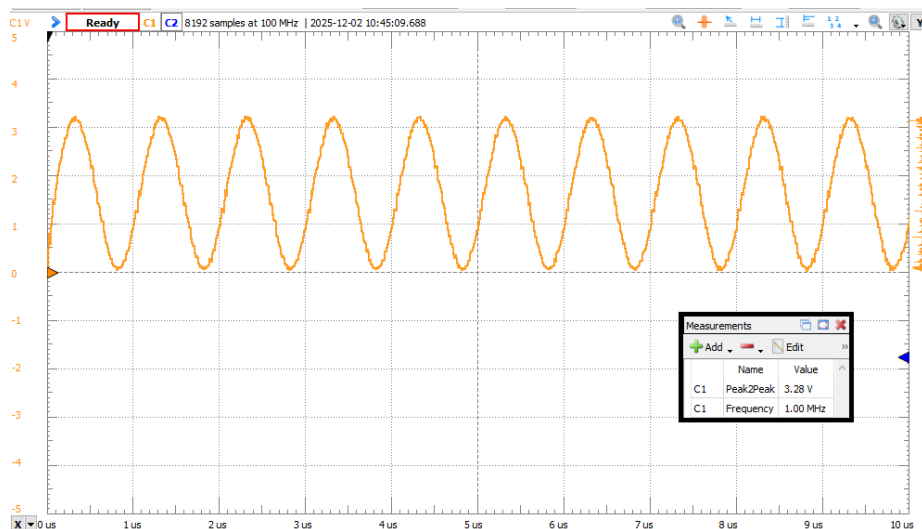
O controle de fase ocorre de maneira semelhante. Por meio da entrada *phase_control*, é possível mudar a fase da senoide que está sendo gerada. Uma senoide com fase 0° inicia-se na origem (valor zero) e imediatamente começa a subir em direção ao seu pico positivo. Uma senoide com fase 90° inicia-se no seu valor máximo ($\sin 90^\circ = 1$) e instantaneamente começa a descer. Por fim, uma senoide com fase 180° inicia-se na origem e começa a descer em direção ao seu pico negativo. Essas mudanças de fase podem ser observadas na Figura 31b.

Além disso, é possível controlar, simultaneamente, os parâmetros de fase e amplitude da senoide (conforme a Figura 31c). Ademais, observa-se que, quando o *enable* é desativado, bloqueia-se a geração de sinal pelo AWG – conforme projetado.

5.2 Testes experimentais

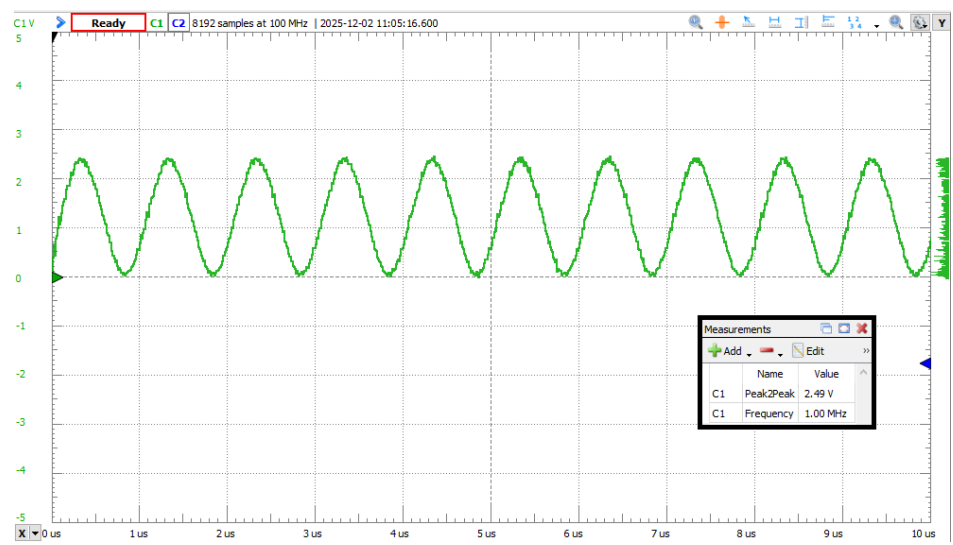
Para a realização dos testes experimentais do AWG, utilizou-se o arranjo experimento descrito na metodologia. As figuras a seguir apresentam senoides geradas com frequência fixa de 1 MHz e amplitudes em 100, 75, 50 e 25%. Para uma amplitude de 100%, obteve-se uma tensão de pico-a-pico (V_{pp}) de 3,28 V (Figura 32); para 75%, V_{pp} de 2,49 V (Figura 33); para 50%, V_{pp} de 1,70 V (Figura 34) e para 25%, V_{pp} de 0,93 V (Figura 34).

Figura 32 – Senoide gerada com amplitude máxima e frequência de 1 MHz.



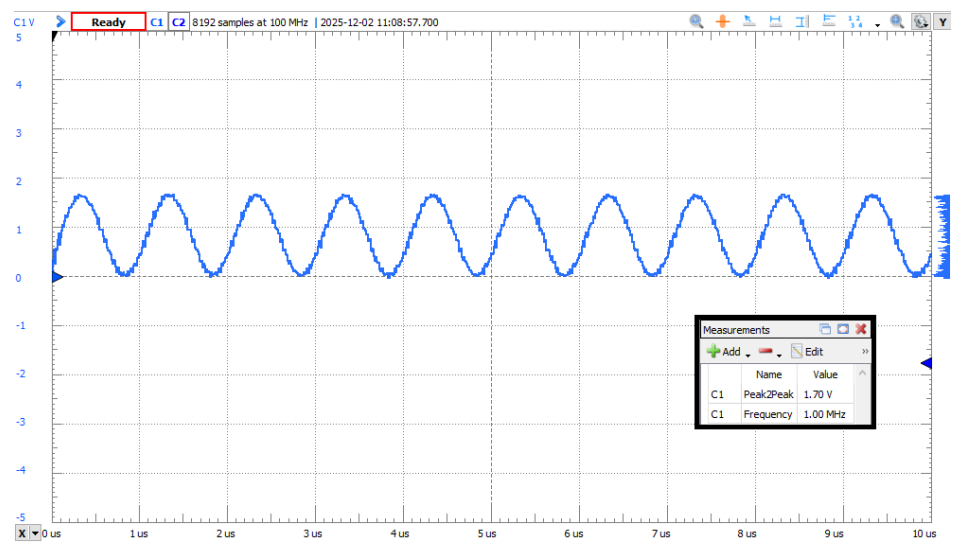
Fonte: Autoria própria (2025).

Figura 33 – Senoide gerada com 75% de amplitude e frequência de 1 MHz.



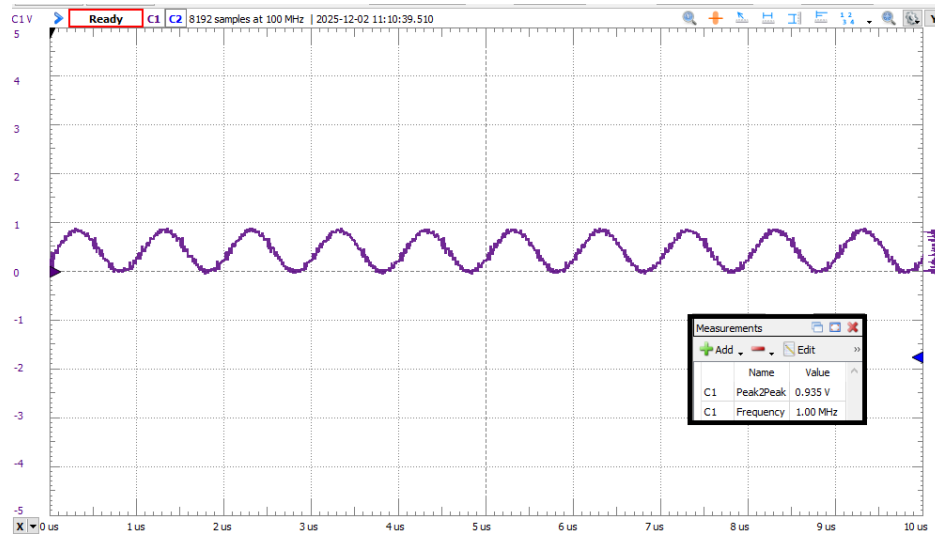
Fonte: Autoria própria (2025).

Figura 34 – Senoide gerada com 50% de amplitude e frequência de 1 MHz.



Fonte: Autoria própria (2025).

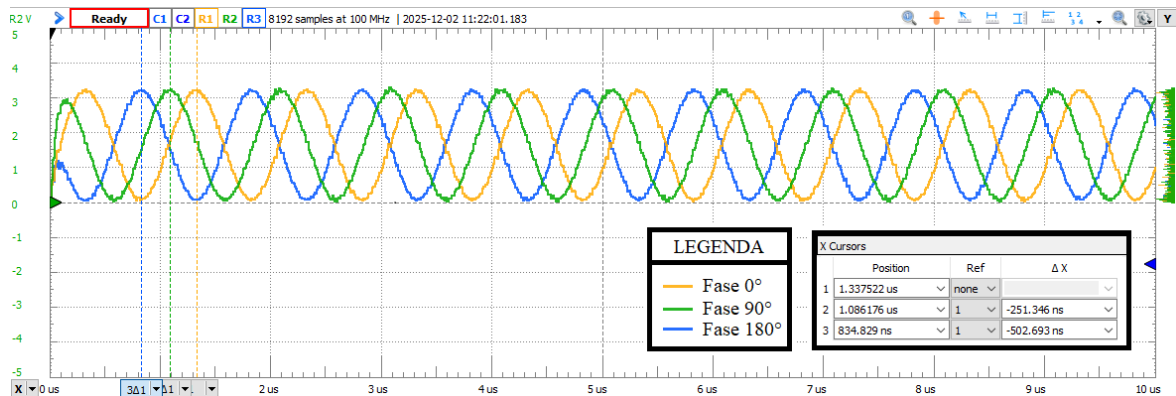
Figura 35 – Senoide gerada com 25% de amplitude e frequência de 1 MHz.



Fonte: Autoria própria (2025).

Em seguida, foram realizados os testes de controle de fase, gerando senoides com fase 0°, 90° e 180°, ambas com máxima amplitude e frequência de 1 MHz. A Figura 36 apresenta as senoides geradas durante este teste.

Figura 36 – Senoides geradas com: fase 0° (laranja); fase 90° (verde); fase 180° (azul).



Fonte: Autoria própria (2025).

Utilizando-se dos cursores presentes no osciloscópio e da Equação 12, é possível verificar defasagem ($\Delta\phi$) entre as senoides e comprovar a fase das ondas geradas.

$$\Delta\phi = \Delta t \times f \times 360^\circ. \quad (12)$$

Onde Δt é o intervalo de tempo de defasagem (obtido utilizando, como referência, o pico máximo das senoides) e f é a frequência da senoide. O Δt obtido entre os picos da senoide laranja (fase 0°) e azul (fase 180°) foi 502,693 ns. Logo:

$$\Delta\varphi = (502,693 \times 10^{-9}) \times (1 \times 10^6) \times 360^\circ \cong 180,97^\circ$$

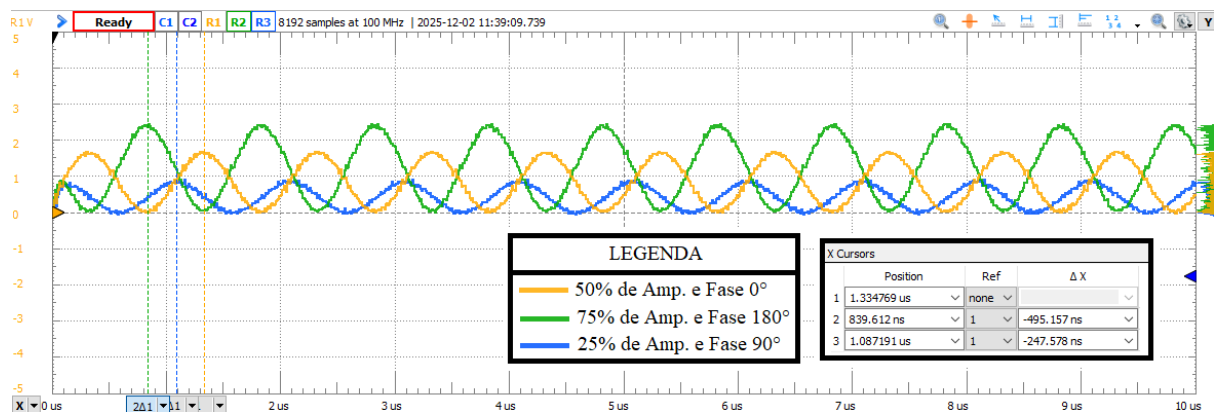
Entre as senoides laranja (fase 0°) e verde (fase 90°), obteve-se um Δt igual a 251,346 ns. Logo:

$$\Delta\varphi = (251,346 \times 10^{-9}) \times (1 \times 10^6) \times 360^\circ \cong 90,48^\circ$$

Os valores obtidos para os ângulos de defasagem entre as senoides geradas asseguram que o controle de fase foi implementado corretamente. A pequena diferença decimal presente nos ângulos de defasagem calculados (antes do arredondamento) são provenientes de erros no posicionamento dos cursores – algo esperado, visto que o posicionamento é feito manualmente.

Prosseguindo com os testes, foram geradas ondas senoidais utilizando o controle de fase e de amplitude de forma simultânea, conforme ilustra a Figura 37 a seguir. Assim, demonstra-se que é possível, simultaneamente, controlar os parâmetros de fase e amplitude da onda senoidal gerada.

Figura 37 – Senoides geradas com: amplitude 50% e fase 0° (laranja); amplitude 75% e fase 180° ; amplitude 25% e fase 90° .



Fonte: Autoria própria (2025).

Conforme mencionado anteriormente, visando diminuir os prejuízos inerentes à uma LUT muito grande – como aumento no consumo de memória, energia e latência – realiza-se o truncamento de fase. Para isso, descarta-se os 22 bits menos significativos oriundos do AF (32 bits) e endereça-se a LUT apenas com os 10 bits restantes. Entretanto, uma vez que esse erro é periódico, são gerados harmônicos indesejados no domínio da frequência conhecidos como espúrios. Esses espúrios prejudicam a faixa dinâmica livre de espúrios (spurious-free dynamic range ou SFDR). Diante disso, foi empregada a técnica de *dithering* de fase.

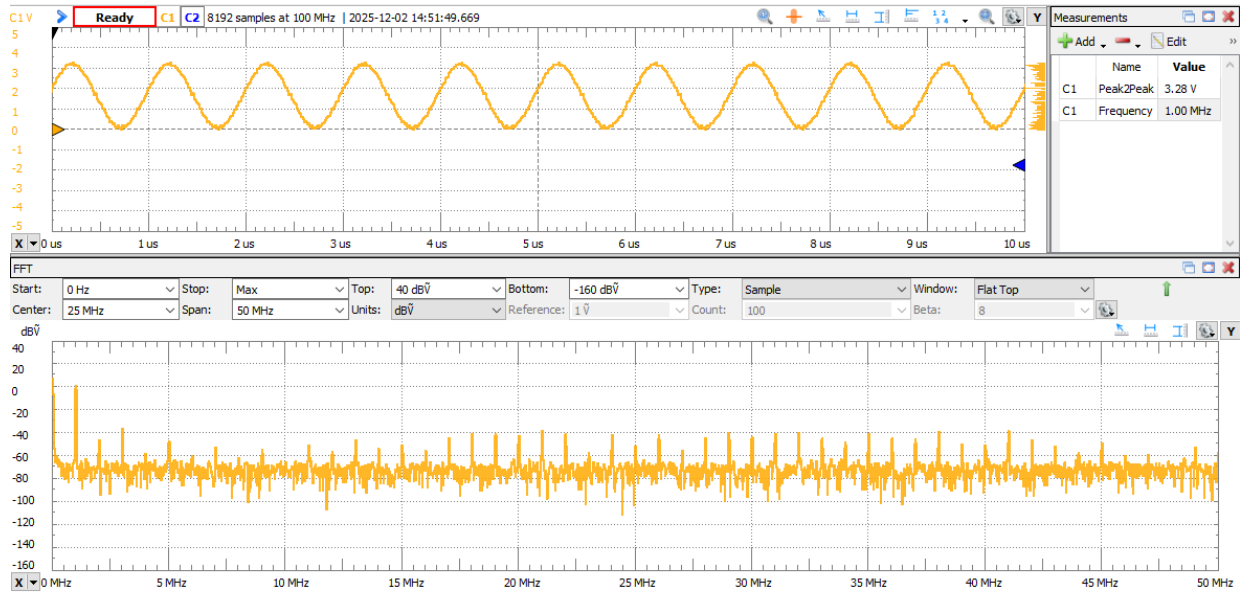
A Figura 38a apresenta o espectro de frequência (FFT) de uma senoide de 1 MHz produzida sem a implementação da técnica supracitada. Nela, nota-se claramente a presença de componentes espúrios, isto é, picos indesejados de energia provenientes do erro gerado pelo truncamento de fase. Para aplicações em computação quântica, a presença desses espúrios é bastante prejudicial, pois podem coincidir com frequências de ressonância de algum qubit, provocando mudanças de estado indesejadas. Para mitigar este problema, implementou-se o gerador de números pseudoaleatórios de 22 bits, somando um ruído à fase antes do truncamento.

A Figura 38b apresenta o espectro de frequência do mesmo sinal, porém com a técnica de *dithering* ativada. Ao compará-la com a Figura 38a, constata-se uma redução significativa na amplitude e ocorrência das componentes espúrias. A energia, antes concentrada nesses picos, é distribuída em todo o espectro, resultando em um piso de ruído ligeiramente mais elevado, porém mais uniforme.

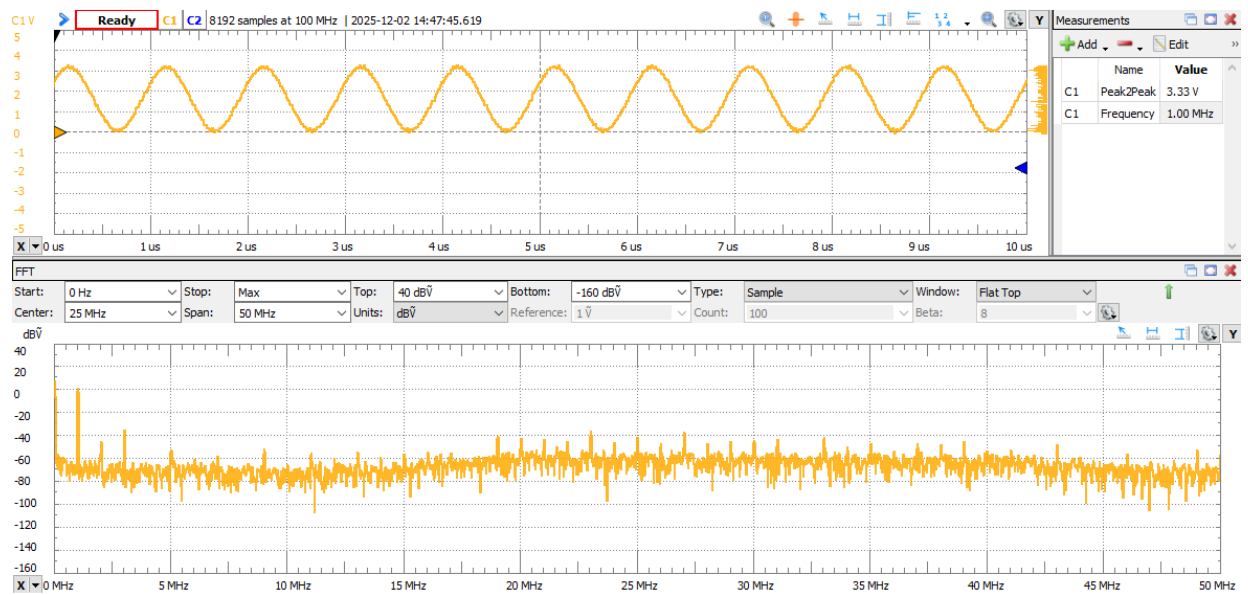
Embora a análise visual da FFT evidencie que a técnica de *dithering* de fase contribui para a redistribuição do erro de quantização proveniente do truncamento de fase, notou-se que não ocorreu uma melhora significativa no valor absoluto do SFDR – aproximadamente 40 dB em ambos os casos.

Assim, o truncamento de fase não parece ser a principal limitação para a pureza espectral do sinal gerado na atual implementação do AWG. Os resultados indicam que os fatores limitantes predominantes são a não linearidade intrínseca do DAC R-2R e a ausência de um filtro passa-baixas (LPF). Uma vez que o sinal é reconstruído por degraus, o espectro apresenta não apenas a fundamental, mas múltiplos harmônicos e distorções, conforme observado na Figura 38.

Figura 38 – FFT da senoide de 1 MHz: (a) sem *dithering* de fase; (b) com *dithering* de fase.



(a)



(b)

Fonte: Autoria própria (2025).

6 CONSIDERAÇÕES FINAIS

O presente trabalho apresentou o desenvolvimento e a implementação digital de um gerador de formas de onda arbitrárias (AWG), baseado na técnica de síntese digital direta (DDS), para aplicação na computação quântica. Para a execução deste projeto, foram realizados, previamente, estudos fundamentais acerca dos seguintes tópicos: computação quântica; técnica de síntese digital direta, suas principais arquiteturas e sua aplicação em geradores de formas de onda arbitrárias. Essa fundamentação teórica permitiu a definição, de maneira precisa, das especificações necessárias para o desenvolvimento do projeto.

A implementação do AWG foi realizada em um FPGA Cyclone IV da fabricante Intel (Altera), utilizando a linguagem de descrição de *hardware* Verilog. Os *softwares* Quartus II e ModelSim foram empregados como ferramentas de prototipagem e simulação, respectivamente. Para a execução dos testes experimentais, utilizou-se um arranjo experimental composto por um *kit* de desenvolvimento DE0-Nano, um DAC R-2R e um Analog Discovery 2. Durante as etapas de validação, que compreenderam a simulação funcional via ModelSim e os testes experimentais no *setup* supracitado, foram geradas ondas senoidais com frequência fixa de 1 MHz, variando-se os parâmetros de fase e amplitude desses sinais.

Diante do exposto, conclui-se que a implementação digital do gerador de formas de onda arbitrária apresentou desempenho satisfatório, cumprindo os objetivos estipulados no início do projeto ao demonstrar a capacidade de gerar ondas senoidais com controle independente de fase e amplitude.

Outrossim, a análise espectral realizada permitiu avaliar o impacto ocasionado pela técnica de *dithering* de fase na pureza do sinal. Constatou-se que a técnica foi bastante eficaz na diminuição dos espúrios oriundos do truncamento de fase, realizando a redistribuição da energia do erro pelo espectro. Entretanto, confirmou-se que o fator limitante para a faixa dinâmica livre de espúrios (SFDR) do projeto atual não foi a etapa de processamento digital, mas sim a etapa analógica, especialmente em virtude da não-linearidade do conversor DAC R-2R, bem como da ausência de um filtro de reconstrução.

REFERÊNCIAS

- ANALOG DEVICES. **The Data Conversion Handbook**. [S. I.]: Newnes, 2004. 976 p.
- BHAT, Hilal Ahmad *et al.* Quantum Computing: fundamentals, implementations and applications. **IEEE Open Journal Of Nanotechnology**. [S. I.], p. 61-77. 27 maio 2022.
- DIGILENT. **Analog Discovery 2 (Legacy)**. Disponível em: <https://digilent.com/reference/test-and-measurement/analog-discovery-2/start>. Acesso em: 03 dez. 2025.
- DIVINCENZO, David P. The Physical Implementation of Quantum Computation. **Fortschritte Der Physik**. [S. I.], p. 771-783. set. 2000.
- HARRIS, David Money; HARRIS, Sarah L. **Digital Design and Computer Architecture**. San Francisco: Morgan Kaufmann Publishers, 2007. 592 p
- HENRIQUE, Rui Tiago Ferro. **Síntese Digital Direta**. 2022. 93 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Instituto Universitário de Lisboa, Lisboa, 2022.
- HUGHES, Ciaran *et al.* **Quantum Computing for the Quantum Curious**. [S. I.]: Springer, 2021. 150 p.
- JUNHAO, Wang; XIAOLING, Zhang; XUESONG, Xie; WANBIN, Wang. Optimal algorithm of waveform generator ROM compression based on DDS technology. **Electronic Measurement Technology**, Pequim, v. 45, n. 7, p. 82-87, jul. 2022.
- LIN, Jin *et al.* Scalable and customizable arbitrary waveform generator for superconducting quantum computing. **Aip Advances**, [S. I.], v. 9, n. 11, p. 115309-115309, nov. 2019. Disponível em: <https://doi.org/10.1063/1.5120299>. Acesso em: 30 out. 2025.
- NIELSEN, Michael A.; CHUANG, Isaac L. **Quantum Computation and Quantum Information**. 10. ed. New York: Cambridge University Press, 2011. 710 p.
- MALOBERTI, Franco. **Data Converters**. [S. I.]: Springer, 2007. 440 p.
- MARIN, J. L.; KASCHNY, J. R. A. Projeto e construção de um gerador de funções microcontrolado usando síntese digital direta. **Congresso de Pesquisa e Inovação da Rede Norte e Nordeste de Educação Tecnológica**, Instituto Federal da Bahia, 2011.
- MURPHY, Eva; SLATTERY, Colm. All About Direct Digital Synthesis. **Analog Dialogue**, [S. I.], v. 38, n. 3, p. 8-12, ago. 2004. Disponível em: <https://www.analog.com/media/en/analog-dialogue/volume-38/number-3/articles/all-about-direct-digital-synthesis.pdf>. Acesso em: 12 set. 2025.
- OPPENHEIM, Alan V.; SCHAFER, Ronald W. **Processamento em tempo discreto de sinais**. 3. ed. [S. I.]: Pearson, 2013. 692 p.
- PARK, Jongseok *et al.* A Fully Integrated Cryo-CMOS SoC for State Manipulation, Readout, and High-Speed Gate Pulsing of Spin Qubits. **IEEE Journal Of Solid-State Circuits**, [S. I.], v. 56, n. 11, p. 3289-3306, nov. 2021

PEDRONI, Volnei A. **Circuit Design with VHDL**. [S. I.]: Mit Press, 2004. 363 p.

REICHERT, Adriano; MORETO, Raphael Francisco. **Desenvolvimento de um gerador de formas de ondas arbitrárias utilizando a técnica de síntese digital direta**. 2017. 113 f. TCC (Graduação) - Curso de Engenharia Eletrônica, Universidade Tecnológica Federal do Paraná, Toledo, 2017.

SADEGHI, Saman. Classifying FPGA Technology in Digital Signal Processing: a review. **International Journal Of Engineering & Technology Sciences**, [S. I.], v. 2024, p. 1-10, jul. 2024.

SANTOS, Juliana Inácio dos; LAMPA, Paulo Henrique Delallo Martins. **Desenvolvimento de um gerador de ondas arbitrárias utilizando a técnica de síntese digital direta e com recursos de modulação**. 2018. 70 f. TCC (Graduação) - Curso de Engenharia Eletrônica, Universidade Tecnológica Federal do Paraná, Toledo, 2018.

SHAFIQUE, Muhammad Ali; MUNIR, Arslan; LATIF, Imran. Quantum Computing: circuits, algorithms, and applications. **IEEE Access**, [S. I.], v. 12, p. 22296-22314, 6 fev. 2024.

TAO, Yunpeng. Quantum entanglement: principles and research progress in quantum information processing. **Theoretical And Natural Science**, [S. I.], v. 30, n. 1, p. 263-274, 24 jan. 2024.

TEKTRONIX. **AWG70000B Arbitrary Waveform Generator**. Disponível em: <https://www.tek.com/en/products/arbitrary-waveform-generators/awg70000#>. Acesso em: 29 out. 2025.

TERASIC. **DE0-Nano FPGA Development and Education Kit**. Disponível em: <https://www.terasic.com.tw/cgi-bin/page/archive.pl?No=593>. Acesso em: 21 nov. 2025.

TOROSYAN, Arthur; WILLSON JUNIOR, Alan N. Exact Analysis of DDS Spurs and SNR due to Phase Truncation and Arbitrary Phase-to-Amplitude Errors. In: IEEE INTERNATIONAL FREQUENCY CONTROL SYMPOSIUM AND EXPOSITION, 2005, Vancouver. **Proceedings [...]**. [S. I.]: IEEE, 2006. p. 50-58.

VAHID, Frank. **Sistemas Digitais: projeto, otimização e hdls**. Porto Alegre: Bookman, 2008. 560 p.

VAN DIJK, J.P.G. *et al.* Impact of Classical Control Electronics on Qubit Fidelity. **Physical Review Applied**, [S. I.], v. 12, n. 4, p. 1-20, maio 2019.

VANKKA, Jouko. **Direct Digital Synthesizers: Theory, Design and Applications**. 2000. 208 f. Tese (Doutorado) - Curso de Ciência e Tecnologia, Departamento de Engenharia Elétrica e de Comunicações, Helsinki University Of Technology, Espoo, 2000.

VASANTRAO, Kendre Saraswati; SAXENA, Amit. Bits to Qubits: an overview of quantum computing. **2025 International Conference On Intelligent Control, Computing And Communications (IC3)**, [Mathura], p. 120-124, 13 fev. 2025.

VAZ, Pablo Ilha. **Síntese Digital Direta**. 2012. 78 f. TCC (Graduação) - Curso de Engenharia Elétrica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012.

WISEMAN, Howard M.; MILBURN, Gerard J. **Quantum Measurement and Control**. New York: Cambridge University Press, 2010. 478 p.

WONG, Thomas G. **Introduction to Classical and Quantum Computing**. [S. I.]: Rooted Grove, 2022. 400 p.

YANG, Yuchen et al. An FPGA-Based Low Latency AWG for Superconducting Quantum Computers. 2021 **IEEE International Instrumentation And Measurement Technology Conference (I2Mtc)**, [S. I.], p. 1-6, 17 maio 2021.

ZHENGIUN, Zhao; FUPING, Wang. The Specific Implementation of a Programmable Arbitrary Digital Signal Generator. **2020 Asia-Pacific Conference On Image Processing, Electronics And Computers (Ipec)**, Dalian, p. 460-465, abr. 2020.