



Universidade Federal do Amazonas

Faculdade de Tecnologia

ANÁLISE E TESTE DE UM CONVERSOR A/D  
INTEGRADOR COM FAIXA DE ENTRADA E  
RESOLUÇÃO PROGRAMÁVEL A CAPACITORES  
CHAVEADOS

VINICIUS MATHEUS BATISTA PEREIRA

Manaus - AM

2022

VINICIUS MATHEUS BATISTA PEREIRA

ANÁLISE E TESTE DE UM CONVERSOR A/D  
INTEGRADOR COM FAIXA DE ENTRADA E  
RESOLUÇÃO PROGRAMÁVEL A CAPACITORES  
CHAVEADOS

Monografia apresentada à  
Coordenação do Curso de Engenharia  
Elétrica da Universidade Federal do  
Amazonas, como parte dos requisitos  
necessários à obtenção do título de  
Engenheiro Eletricista.

**Orientador: Prof. Dr. Thiago Brito Bezerra**

Manaus - AM

2022

## Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

P436a Pereira, Vinicius Matheus Batista  
Análise e teste de um conversor A/D integrador com faixa de  
entrada e resolução programável a capacitores chaveados /  
Vinicius Matheus Batista Pereira . 2022  
77 f.: il. color; 31 cm.

Orientador: Thiago Brito Bezerra  
TCC de Graduação (Engenharia Elétrica - Eletrônica) -  
Universidade Federal do Amazonas.

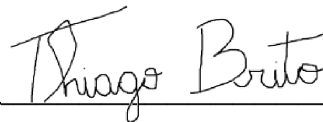
1. Conversor analógico-digital. 2. Parâmetros estáticos. 3.  
Parâmetros dinâmicos. 4. Conversor integrador. I. Bezerra, Thiago  
Brito. II. Universidade Federal do Amazonas III. Título

VINICIUS MATHEUS BATISTA PEREIRA

ANÁLISE E TESTE DE UM CONVERSOR A/D INTEGRADOR COM  
FAIXA DE ENTRADA E RESOLUÇÃO PROGRAMÁVEL A  
CAPACITORES CHAVEADOS

Monografia apresentada à Coordenação do Curso de Engenharia Elétrica da Universidade Federal do Amazonas, como parte dos requisitos necessários à obtenção do título de Engenheiro Eletricista.

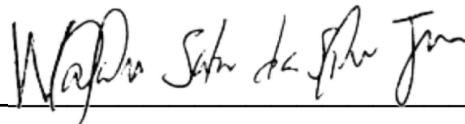
Trabalho aprovado em, 12 de setembro de 2022:



---

**Prof. Dr Thiago Brito Bezerra - Presidente e Orientador**

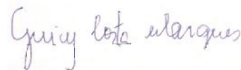
Departamento de Eletrônica e Computação - UFAM



---

**Prof. Dr. Waldir Sabino da Silva Júnior - Membro**

Departamento de Eletrônica e Computação - UFAM



---

**Prof<sup>a</sup>. Dr<sup>a</sup>. Greicy Costa Marques - Membro**

Departamento de Eletrônica e Computação - UFAM

Manaus - AM

2022

# Dedicatória

*Aos meus pais, Manoel e Ozenilda.*

# Agradecimentos

Agradeço a Deus, por me dar forças e esperança em meio as dificuldades.

A meus pais, Manoel e Ozenilda, por todo apoio, suporte e incentivo nos meus estudos ao longo da graduação

Ao meu orientador, o professor Dr. Thiago Brito pela orientação, apoio e ensinamentos transferidos durante a execução deste trabalho.

Aos amigos que fiz durante a graduação, Otávio, Alice, Gustavo, Gabriel, Paulo, Matheus e Luiz, que me acompanharam nessa caminhada desde 2017.

Aos professores que aceitaram participar da banca de avaliação, professor Dr. Waldir Sabino e professora Dr<sup>a</sup>. Greicy Costa.

# Resumo

Conversores analógico-digitais permitem que sistemas digitais tenham acesso as variáveis físicas analógicas presentes no meio, mas para assegurar a qualidade dos dados prestados é necessário conhecer os parâmetros de desempenho do conversor. Neste trabalho de conclusão de curso é desenvolvido um circuito de testes para um conversor analógico-digital integrador com faixa de entrada e resolução programável a capacitor chaveado. O conversor foi projetado em nível de transistores e fabricado com tecnologia AMS 0,35  $\mu$  CMOS, mas ainda é necessário comprovar experimentalmente seu funcionamento e desempenho. Seu método de funcionamento é estudado para aplicação de testes com sinais de rampa e senoidais, por meio de um conversor digital-analógico e uma placa com microcontrolador ARM. A partir dos resultados obtidos, são extraídos os parâmetros estáticos e dinâmicos do conversor para faixa de resolução de 5 e 8 bits, com ganho unitário e de 1,5 vezes. Ao fim dos testes, foi possível validar o desempenho do conversor para a faixa de 5 bits completa, com resultados satisfatórios, no entanto melhorias precisam ser feitas na metodologia de testes para a resolução de 8 bits.

**Palavras-chave:** Conversor analógico-digital. Parâmetros estáticos. Parâmetros dinâmicos. Conversor integrador.

# Abstract

Analog-to-digital converters allow digital systems to access the physical analog variables present in the world, but to ensure the quality of the data provided it is necessary to know the performance parameters of the converter. In this course conclusion work, a test circuit for an analog-to-digital converter with input range and programmable resolution using a switched capacitor is developed. The converter was designed at the transistor level and manufactured using AMS 0.35  $\mu$  CMOS technology, but its operation and performance still need to be tested experimentally. Its method of operation is studied for application of tests with ramp and sinusoidal signals, through a digital-analog converter and a board with ARM microcontroller. From the results obtained, the static and dynamic parameters of the converter are extracted for a resolution range of 5 and 8 bits, with unity gain and 1.5 times. At the end of the tests, it was possible to validate the performance of the converter for the full 5-bit range, with satisfactory results, however improvements need to be made in the testing methodology for the 8-bit resolution.

**Palavras-chave:** Digital to analog converter. Static parameters. Dynamic parameters. Integrating converter.



# Sumário

1	INTRODUÇÃO	1
1.1	Objetivo geral . . . . .	2
1.2	Objetivos específicos . . . . .	2
1.3	Metodologia . . . . .	3
1.4	Organização do trabalho . . . . .	3
2	CONVERSORES ANALÓGICO-DIGITAIS	4
2.1	Princípio de funcionamento . . . . .	4
2.2	Características gerais . . . . .	5
2.2.1	Tipos de sinais analógicos . . . . .	5
2.2.2	Resolução . . . . .	6
2.2.3	Erro de quantização . . . . .	6
2.2.4	Faixa dinâmica . . . . .	7
2.2.5	Valores máximos . . . . .	7
2.3	Especificações estáticas . . . . .	8
2.3.1	Erro de desvio . . . . .	8
2.3.2	Erro de ganho . . . . .	9
2.3.3	Erro de não-linearidade diferencial . . . . .	10
2.3.4	Erro de não-linearidade integral . . . . .	10
2.4	Especificações dinâmicas . . . . .	11
2.4.1	Relação sinal-ruído . . . . .	11
2.4.2	Relação sinal ruído e distorção . . . . .	12
2.4.3	Número efetivo de bits . . . . .	12
2.4.4	Faixa dinâmica livre de espúrios . . . . .	13
2.5	Arquiteturas comuns . . . . .	13
2.5.1	Paralelo . . . . .	14
2.5.2	Pipeline . . . . .	15
2.5.3	Aproximação sucessivas . . . . .	15

2.5.4	Sigma-delta . . . . .	16
2.5.5	Integrador . . . . .	17
3	CONVERTOR A/D INTEGRADOR COM FAIXA DE ENTRADA E RESOLUÇÃO PROGRAMÁVEL A CAPACITORES CHAVEADOS . . . . .	18
3.1	Características do conversor . . . . .	18
3.1.1	Conversor A/D integrador de rampa múltipla <i>RunUp</i> . . . . .	18
3.1.2	Circuitos a Capacitores chaveados . . . . .	20
3.1.3	Faixa de entrada ajustável . . . . .	22
3.1.4	Resolução programável . . . . .	24
3.2	Arquitetura e funcionamento do conversor . . . . .	24
3.2.1	Arquitetura . . . . .	25
3.2.2	Funcionamento . . . . .	26
3.2.3	Especificações técnicas . . . . .	27
4	TESTES EM CONVERTORES ANALÓGICO-DIGITAIS . . . . .	29
4.1	Configurações de testes . . . . .	30
4.1.1	Configuração de teste de onda senoidal . . . . .	30
4.1.2	Configuração de teste de sinal arbitrário . . . . .	31
4.1.3	Configuração de teste de degrau . . . . .	31
4.2	Método de testes . . . . .	31
4.2.1	Método de histograma de rampa . . . . .	32
4.2.2	Teste de FFT . . . . .	33
5	CIRCUITO DE TESTES . . . . .	35
5.1	Circuito de alimentação . . . . .	35
5.1.1	Regulador de tensão . . . . .	37
5.2	Correntes de polarização . . . . .	39
5.3	Chaves . . . . .	40
5.4	Circuitos digitais . . . . .	40
5.4.1	Freescale Freedom FRDM-KL25Z . . . . .	41
5.4.2	Circuito de testes completo . . . . .	42
6	ALGORITMOS DE TESTE . . . . .	44
6.1	Algoritmo de sinal de rampa . . . . .	44

6.2	Algoritmo de sinal senoidal . . . . .	45
7	RESULTADOS . . . . .	47
7.1	Resolução de 5 bits . . . . .	47
7.1.1	Ganho unitário - 0 à 2,56V . . . . .	48
7.1.2	Ganho 1.5x - 0 à 1,70V . . . . .	51
7.2	Resolução de 8 bits . . . . .	54
7.2.1	Ganho unitário - 0 à 2,56V . . . . .	54
7.2.2	Ganho 1.5x - 0 à 1,70V . . . . .	57
7.3	Análise dos resultados . . . . .	60
8	CONCLUSÃO . . . . .	62
8.1	Trabalhos futuros . . . . .	63
A	CODIGOS UTILIZADOS PARA TESTES . . . . .	64
A.1	Teste de rampa . . . . .	64
A.2	Teste de senoide . . . . .	67
B	CODIGOS UTILIZADOS PARA PROCESSAMENTO DE DADOS NO MATLAB . . . . .	71
B.1	Algoritmo para análise de teste de rampa . . . . .	71
B.2	Algoritmo para análise de teste senoidal . . . . .	73
	BIBLIOGRAFIA . . . . .	76

# Lista de Figuras

Figura 1	Diagrama em bloco da conversão analógico-digital. . . . .	4
Figura 2	Aplicação de sinal diferencial. Adaptado de [9]. . . . .	6
Figura 3	Representações de (a) meia-faixa e (b) meia-transição. Adaptado de [12] . . . . .	7
Figura 4	Função de transferência ideal de um conversor de 8 bits. Adaptado de [6] . . . . .	8
Figura 5	Erro de desvio. Adaptado de [8] . . . . .	9
Figura 6	Erro de ganho. Adaptado de [8] . . . . .	9
Figura 7	Curva com INL e DNL não nulos. Adaptado de [8] . . . . .	11
Figura 8	Resposta em frequência de um conversor. Adaptado de [15] . . . .	13
Figura 9	Conversor paralelo. Adaptado de [5] . . . . .	14
Figura 10	Conversor pipeline. Adaptado de [5] . . . . .	15
Figura 11	Conversor por aproximação sucessiva. Adaptado de [5] . . . . .	16
Figura 12	Conversor integrador. Adaptado de [5] . . . . .	17
Figura 13	Conversor rampa múltipla <i>RunUp</i> . Adaptado de [12] . . . . .	19
Figura 14	Equivalência do resistor em (a), com o uso do capacitor chaveado em (b). Adaptado de [12] . . . . .	20
Figura 15	Outro sistema de capacitores chaveados. Adaptado de [12] . . . . .	21
Figura 16	Operação sem inversão de tensão. Adaptado de [12] . . . . .	22
Figura 17	Operação com inversão de tensão. Adaptado de [12] . . . . .	22
Figura 18	Capacitância variável na entrada do integrador. Adaptado de [12] .	23
Figura 19	Arquitetura do conversor A/D rampa múltipla a capacitor chaveado. Adaptado de [7] . . . . .	25
Figura 20	Setup para sinais senoidais. Adaptado de [11] . . . . .	30
Figura 21	Setup para sinais arbitrários. Adaptado de [11] . . . . .	31
Figura 22	Setup para degrau. Adaptado de [11] . . . . .	31
Figura 23	Fontes de alimentação. . . . .	36

Figura 24	LM317 como regulador ajustável de tensão. . . . .	37
Figura 25	Circuito para tensões abaixo de 1,25V. . . . .	38
Figura 26	Circuito para fonte de corrente. . . . .	39
Figura 27	DAC MAX541. . . . .	41
Figura 28	Freescale Freedom FRDM-KL25Z. . . . .	42
Figura 29	Circuito de teste. . . . .	43
Figura 30	Algoritmo simplificado de rampa. . . . .	45
Figura 31	Senoide com 512 níveis lógicos. . . . .	46
Figura 32	Algoritmo simplificado de senoide. . . . .	46
Figura 33	Teste de rampa com 5 bits para ganho unitário. . . . .	48
Figura 34	DNL para a primeira configuração de testes. . . . .	49
Figura 35	INL para a primeira configuração de teste. . . . .	49
Figura 36	Resposta em frequência para o teste senoidal. . . . .	50
Figura 37	Resposta no tempo para o teste senoidal. . . . .	50
Figura 38	Teste de rampa com 5 bits para ganho 1,5x. . . . .	51
Figura 39	DNL para a segunda configuração de testes. . . . .	52
Figura 40	INL para a segunda configuração de teste. . . . .	52
Figura 41	Resposta em frequência para o segundo teste senoidal. . . . .	53
Figura 42	Resposta no tempo para o segundo teste senoidal. . . . .	53
Figura 43	Teste de rampa com 8 bits para ganho unitário. . . . .	54
Figura 44	DNL para a terceira configuração de testes. . . . .	55
Figura 45	INL para a terceira configuração de teste. . . . .	56
Figura 46	Resposta em frequência para o teste senoidal. . . . .	56
Figura 47	Resposta no tempo para o teste senoidal. . . . .	57
Figura 48	Teste de rampa com 8 bits para ganho 1,5x. . . . .	58
Figura 49	DNL para a quarta configuração de testes. . . . .	58
Figura 50	INL para a quarta configuração de teste. . . . .	59
Figura 51	Resposta em frequência para o quarto teste senoidal. . . . .	59
Figura 52	Resposta no tempo para o segundo teste senoidal. . . . .	60

# Lista de Tabelas

Tabela 1	Relação entre capacitor de entrada, ganho de sinal e faixa de entrada.	23
Tabela 2	Especificações do conversor. . . . .	27
Tabela 3	Modos de operação. . . . .	28
Tabela 4	Tensões de alimentação para o ADC. . . . .	36
Tabela 5	Correntes de polarização para o ADC. . . . .	36
Tabela 6	Resistências utilizadas para os reguladores. . . . .	38
Tabela 7	Resistências utilizadas para os reguladores. . . . .	38
Tabela 8	Parâmetros para fontes de corrente. . . . .	39
Tabela 9	Parâmetros DAC. . . . .	41

# Lista de abreviações e siglas

A/D	Analog to Digital
ADC	Analog to Digital Converter
CC	Constant Current
dB	Decibéis
DAC	Digital to Analog Converter
DC	Direct Current
DNL	Differential Non Linearity
DR	Dynamic Range
ENOB	Effective Number of Bits
FFT	Fast Fourier Transform
INL	Integral Non Linearity
LSB	Least Significant Bit
MSB	Most Significant Bit
RMS	Root Mean Square
SFDR	Spurious Free Dynamic Range
SINAD	Signal to Noise and Distortion
SNR	Signal to Noise Ratio

# 1 Introdução

O mundo, tal como entendemos pela física, é repleto de grandezas físicas diversas para representar os diferentes fenômenos e estados observados pela matéria. Por vezes é de nosso interesse obter o valor de alguma grandeza física específica, como temperatura, pressão, velocidade ou fluxo por exemplo. A obtenção e o controle de tais grandezas é essencial para o funcionamento de muitos processos industriais, sendo possível encontrar inúmeros exemplos de aplicações como: [1] com o controle da temperatura em um forno, [2] com metodologias para o controle de velocidade em motores de combustão ou [3] com o controle do nível da água em usinas hidrelétricas.

Antes de se processar as grandezas físicas, no entanto, elas precisam ser convertidas para um domínio no qual seja mais fácil a sua manipulação e a determinação de seus valores. Esta tarefa é executada por transdutores, dispositivos capazes de converter os diversos tipos de grandezas físicas em sinais elétricos [4] capazes de serem tratados mais facilmente em circuitos. Os dispositivos sensores, que são uma forma de transdutor, no geral convertem a grandeza para uma faixa de valores de tensão ou corrente elétrica de forma linear [4], dessa forma produzindo sinais analógicos que devem ser processados posteriormente.

Atualmente, com a disseminação dos computadores e sistemas digitais, estes tem sido cada vez mais utilizados e embarcados para processar as variáveis fornecidas pelo meio, afim de gerar as informações desejadas para o usuário. A motivação para o emprego desses sistemas são suas vantagens inerentes na velocidade de operação, exatidão, precisão dos dados processados, facilidade na programação e baixo custo [5]. Contudo, sua natureza de operação digital difere dos sinais analógicos fornecidos por sensores, com isso, faz-se necessário que haja uma conversão nos tipos de dados para permitir essa interface.

A conversão de um sinal elétrico analógico para a forma digital é feita por um conversor analógico-digital, do inglês Analog to Digital Converter (**ADC**), sendo este um componente essencial para os sistemas de aquisição de dados [6]. O **ADC** é capaz de amostrar, quantizar e codificar um sinal inserido como entrada, os passos necessários para conversão. O processo de conversão se inicia com a amostragem do sinal em um único



instante, seguida de sua quantização, ou transformação em uma quantia finita de valores, finalizando com sua codificação para um código binário desejado. Em geral os ADCs conseguem realizar todas essas etapas internamente, necessitando apenas de um circuito condicionador de sinais para ajustar a sua faixa de operação com as dos sinais amostrados.

Existem diversos tipos de conversores analógico-digitais, com técnicas de conversão que apresentam ganhos seja na velocidade de conversão, na resolução da resposta ou na exatidão da resposta [5]. Cada topologia de conversor apresenta benefícios e desvantagens associados a sua metodologia, tendo em geral que sacrificar desempenho em uma característica para obter maior performance em outra. Uma dessas técnicas é o uso de conversores integradores, que utilizam a integral do sinal no tempo durante o processo de conversão. O objeto de estudo deste trabalho é um conversor que utiliza uma variação dessa técnica, um integrador de rampa múltipla.

Neste trabalho serão executados testes com o conversor proposto em [7], com o intuito de determinar suas características de desempenho para conversão de sinais estáticos e dinâmicos em variadas faixas de operação. Este conversor é capaz de ajustar sua faixa de entrada para se igualar ao nível do sinal amostrado, além de permitir variar sua resolução e com isso obter ganhos no tempo de conversão. Será feita uma comparação entre os resultados obtidos experimentalmente e os obtidos pelas simulações em [7].

## 1.1 OBJETIVO GERAL

Extrair as figuras de mérito de um conversor analógico-digital integrador com faixa de entrada e resolução programável a capacitores chaveados.

## 1.2 OBJETIVOS ESPECÍFICOS

- Compreender o funcionamento do conversor A/D utilizado;
- Projetar o circuito com as fontes de tensão e corrente necessárias para o conversor;

- Desenvolver um sistema microcontrolado capaz de testar o conversor em mensurar sinais estáticos e dinâmicos;
- Avaliar o desempenho do conversor para diferentes faixas de entrada e resoluções;

### 1.3 METODOLOGIA

O presente trabalho consiste de uma continuação do trabalho descrito em [7], que teve como objetivo projetar um conversor analógico-digital com faixa de entrada e resolução programáveis utilizando a técnica de capacitores chaveados. O trabalho consiste da validação experimental do conversor proposto, para isso será desenvolvido um circuito de alimentação e teste para o conversor, bem como um algoritmo capaz de receber os dados do sensor em um sistema microcontrolado e processá-los.

O sensor terá seu comportamento experimental testado com pelo menos duas faixas de entrada e duas resoluções diferentes, a fim de comparar seu desempenho real com o obtido durante as simulações em [7].

### 1.4 ORGANIZAÇÃO DO TRABALHO

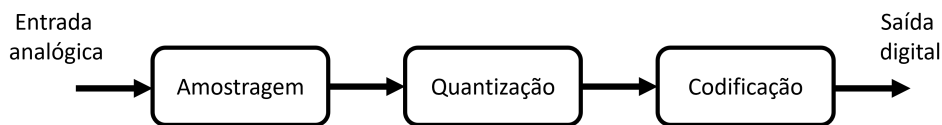
O presente trabalho está organizado da seguinte forma: no capítulo 2 serão apresentadas as características e especificações de um conversor analógico-digital, bem como os tipos mais comuns. No capítulo 3 será descrito o conversor utilizado e seu funcionamento. No capítulo 4 serão apresentado as principais metodologias de testes em ADCs. No capítulo 5 será apresentado circuito de teste utilizado para alimentação e teste do conversor. No capítulo 6 serão apresentados os algoritmos utilizados para gerar os sinais de teste, extrair os dados do conversor e aferir seu desempenho. Por fim, no capítulo 7 serão apresentados os resultados obtidos para cada caso de teste.

## 2 Conversores Analógico-Digitais

Este capítulo apresenta uma revisão das principais características de interesse em um conversor analógico-digital. Explicando seu funcionamento, suas especificações de desempenho e arquiteturas comuns de conversores.

### 2.1 PRINCÍPIO DE FUNCIONAMENTO

O funcionamento de um conversor analógico-digital depende da realização de 3 operações fundamentais: Amostragem, quantização e codificação, exemplificados na Figura 1.



**Figura 1:** Diagrama em bloco da conversão analógico-digital.

A amostragem é o processo que realiza a ponte entre dois domínios de tempo, permitindo converter um sinal de tempo contínuo em um de tempo discreto. Ela é uma etapa fundamental já que os sistemas digitais operam em tempo discreto, inclusive os conversores analógico-digitais. No processo de conversão, a amostragem tem como objetivo reter o valor do sinal de entrada para a realização da conversão durante um período de tempo, denominado de período de amostragem [8]. O período de amostragem pode ser variado de acordo com o uso do conversor, mas é limitado pelo tempo de conversão, que representa o tempo necessário desde a entrada do sinal analógico até que o sinal digital seja produzido.

A quantização, por sua vez, é a classificação da amplitude do sinal de entrada amostrado em níveis finitos e discretos, denominados de níveis de quantização [5]. Os conversores geralmente utilizam a quantização uniforme, na qual a distância entre níveis consecutivos de amplitude é sempre igual. Na quantização a faixa de entrada possível

é dividida nos níveis de quantização possíveis, sendo o nível que mais se aproximar da amplitude do sinal de entrada escolhido para representá-lo.

A codificação é o passo final da conversão analógico-digital e consistem em transformar o nível de quantização obtido anteriormente em um código binário digital. Essa fase depende da arquitetura do conversor utilizado para a escolha da codificação utilizada, os mais utilizados são código binário, código Gray e o código complemento de dois [6].

## 2.2 CARACTERÍSTICAS GERAIS

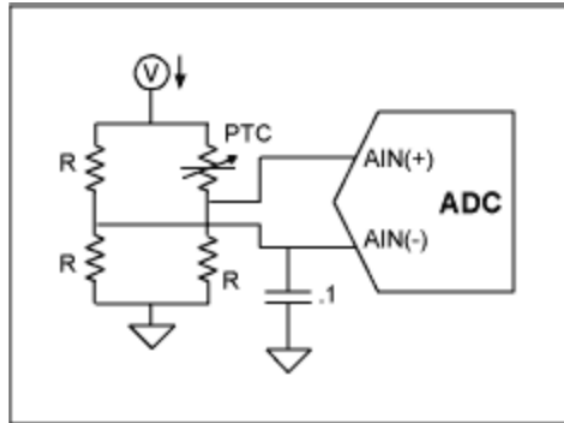
Nesta seção são apresentados os conceitos das características gerais de um conversor Analog to Digital (A/D), que podem ser encontradas em seus *datasheets*.

### 2.2.1 Tipos de sinais analógicos

Um conversor A/D naturalmente tem uma entrada analógica, mas a forma como esse sinal analógico é fornecido ao conversor pode ser variada. A principal característica nesses casos é a relação do sinal e a tensão de referência a ser adotada. Dentre os tipos de sinais temos: sinais de saída única, pseudo-diferencial e diferencial.

Sinais de saída única apresentam apenas uma entrada e tem como referência o mesmo terra do conversor. Sinais pseudo-diferenciais possuem duas entradas, sendo uma o próprio sinal e a outra uma tensão de referência que pode ser diferente do terra do conversor. Sinais diferenciais possuem duas entradas, sendo o sinal desejado a diferença entre ambas, dessa forma ambas as entradas atuam como portadoras do sinal e não há uma referência estática [9].

A Figura 2 apresenta um exemplo de aplicação de ADC utilizando um sinal diferencial. Nessa situação, o objetivo é obter a tensão entre os terminais da ponte de Wheatstone, estando ambos os terminais com uma tensão diferente da referência. O conversor utilizado nesse trabalho opera com essa configuração.



**Figura 2:** *Aplicação de sinal diferencial. Adaptado de [9].*

### 2.2.2 Resolução

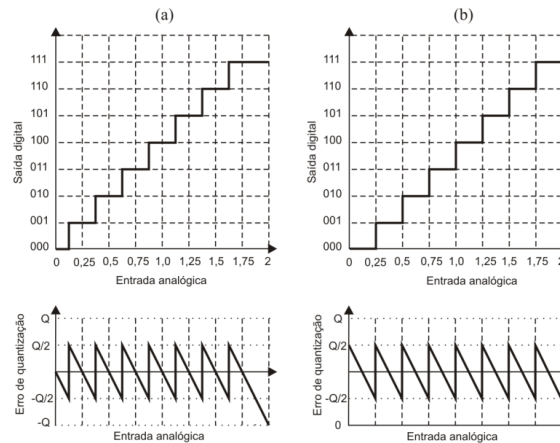
A resolução está relacionada ao menor incremento de tensão necessário para modificar a saída de um conversor. Ela é expressa pela quantidade de bits que o conversor é capaz de produzir em sua resposta.

Dessa forma um conversor com resolução de 10 bits produz uma resposta composta por um sinal de 10 bits, que corresponde a 1024 possíveis níveis lógicos diferente. Se o sinal de entrada desse conversor fosse de  $10V$ , o menor valor capaz de causar um incremento seria  $9,76mV$  [10].

### 2.2.3 Erro de quantização

O erro de quantização está relacionado a diferença entre o valor analógico e o valor digital a ele associado. Todo conversor apresentará um erro de quantização, uma vez que as infinitas possibilidades de valores analógicos devem ser convertidas em um número finito de valores digitais. O erro de quantização está associado também a resolução do conversor, um conversor com maior resolução é mais sensível a pequenas variações na entrada analógica (seu Least Significant Bit (**LSB**) representa uma menor quantia) e, com isso, o erro de quantização associado a cada medida é menor [5].

A norma IEEE 1241-2010 [11] preve duas convenções para definir os intervalos de quantização e caracterizar os erros de quantização de um conversor A/D, a meia-faixa e a meia-transição. A transição de meia-faixa é caracterizada pelo deslocamento das transições em 0,5 LSB para a esquerda, enquanto a denominação meia-transição é dada quando a transição só ocorre após 1 LSB. A Figura 3 representa graficamente essas formas de quantização e seus erros.



**Figura 3:** Representações de (a) meia-faixa e (b) meia-transição. Adaptado de [12]

#### 2.2.4 Faixa dinâmica

A faixa dinâmica, ou Dynamic Range (DR), de um conversor A/D é a razão entre o maior e o menor nível de sinal aplicável à entrada. Ela é expressa em Decibéis (dB) e depende apenas do número de bits do ADC ou resolução [10].

Um conversor de 12 bits apresentará uma razão entre o maior sinal e o menor possível de  $2^{12}$ , o equivalente a 72 dB. Esse valor também é a máxima relação sinal-ruído do conversor.

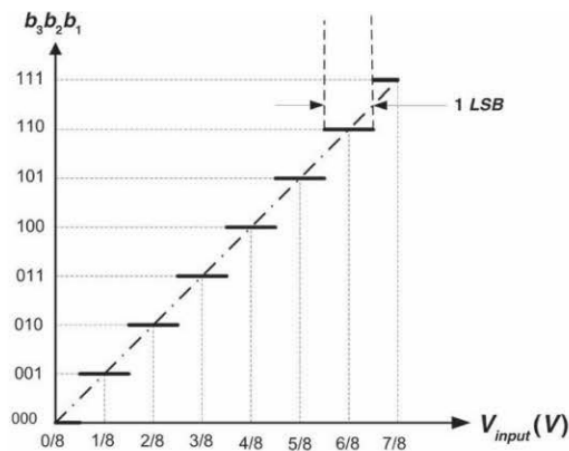
#### 2.2.5 Valores máximos

Os valores máximos absolutos são os limites de tensão utilizáveis que não causam danos ao conversor. Como qualquer circuito, os conversores são projetados para operar a um nível de tensão de alimentação para o funcionamento de seus componentes internos,

ultrapassar esses limites pode resultar em perda de confiabilidade ou até mesmo destruição do conversor.

## 2.3 ESPECIFICAÇÕES ESTÁTICAS

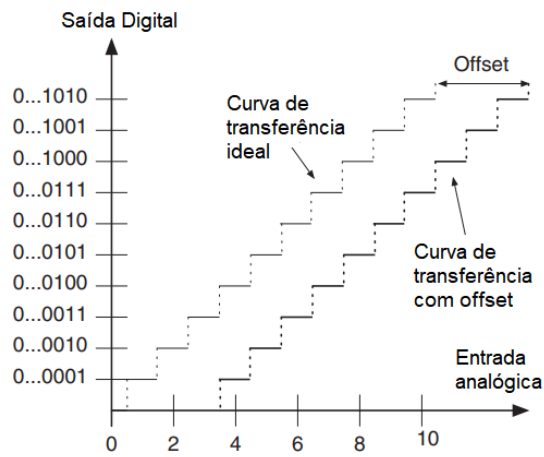
As especificações estáticas de um conversor analógico-digital são os parâmetros obtidos ao se aplicar um sinal contínuo em sua entrada. Essas características são obtidas aplicando um sinal de rampa, variando da tensão de  $0V$  até o valor máximo suportado pelo ADC. A utilização da rampa permite avaliar o comportamento do conversor por toda sua faixa de entrada, sendo esperado em sua saída uma escada com degraus equivalentes ao menor bit significativo, apresentando apenas o erro de quantização devido a limitação digital. Com essa resposta da saída do conversor, ao se traçar uma reta no gráfico unindo o ponto inicial, o final e passando por todos os pontos médios de cada degrau é possível se extrair a função de transferência do ADC, a exemplo da Figura 4.



**Figura 4:** Função de transferência ideal de um conversor de 8 bits. Adaptado de [6]

### 2.3.1 Erro de desvio

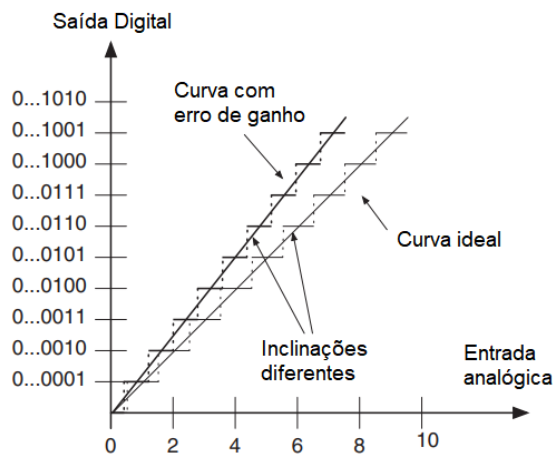
O erro de desvio (*offset*) representa o deslocamento do zero da função de transferência, o que desloca toda a função de transferência consigo, conforme a Figura 5. Esse erro pode ser medido e expresso por **LSB**, pelo valor absoluto da tensão de offset, ou pela porcentagem que ela representa na escala total.



**Figura 5:** Erro de desvio. Adaptado de [8]

### 2.3.2 Erro de ganho

O erro de ganho se refere a diferença de inclinação entre a reta ideal para a função de transferência do ADC e a reta real obtida com os testes. O erro de desvio e ganho juntos representam o erro de escala completa. A Figura 6 apresenta a curva de um ADC com erro de ganho comparado a uma curva ideal.



**Figura 6:** Erro de ganho. Adaptado de [8]



### 2.3.3 Erro de não-linearidade diferencial

O erro de não-linearidade diferencial, ou Differential Non Linearity (**DNL**), é a variação dos intervalos de quantização produzidos pelo **ADC**. Normalmente, um **ADC** ideal irá crescer ou decrescer sua resposta em 1 **LSB** a medida que o sinal de entrada aumenta ou diminui, sem pular nenhum nível e sem manter o mesmo valor na saída por mais de 1 **LSB**, nesses casos o **DNL** é nulo. No entanto em um **ADC** real podem ocorrer variações e uma transição acontecer antes ou depois do esperado gerando perda de código [13].

O **DNL** pode ser calculado como:

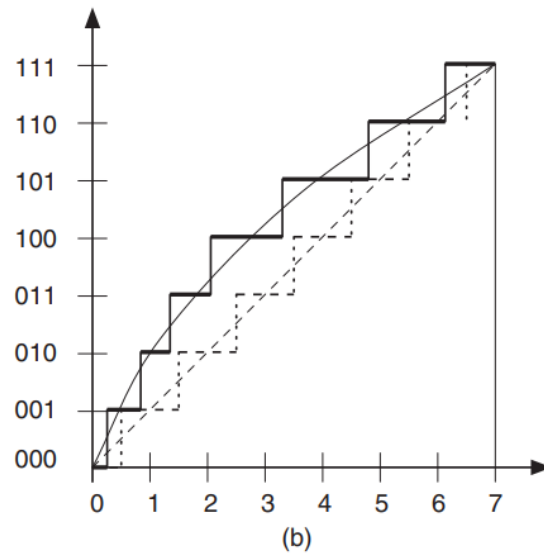
$$DNL(i) = \frac{A(i+1) - A(i)}{A_{LSB}} - 1, i = 0 \dots (2^N - 2) \quad (1)$$

Em que  $A$  representa o valor de transição para um dado nível lógico.

### 2.3.4 Erro de não-linearidade integral

O erro de não linearidade integral, ou Integral Non Linearity (**INL**), é a diferença máxima obtida entre a função de transferência de um conversor e a curva ideal. O **INL** está diretamente ligado ao **DNL**, já que o segundo irá desviar a função de transferência e causar o primeiro. O **INL** pode ser calculado obtendo-se a curva da função de transferência e medindo o maior desvio que ela apresenta em relação ao ideal em função de **LSBs**. Em um conversor ideal a função de transferência e a curva ideal estariam juntas e o **INL** seria 0, caso ela esteja abaixo o **INL** será negativo e se estiver acima será positivo [13].

A Figura 7 apresenta como exemplo a curva de transferência de um **ADC** com **DNL** positivo em todo espectro, e conseqüentemente, **INL** positivo.



**Figura 7:** Curva com INL e DNL não nulos. Adaptado de [8]

## 2.4 ESPECIFICAÇÕES DINÂMICAS

As especificações dinâmicas de um conversor analógico-digital são parâmetros que caracterizam sua resposta em frequência e a velocidade de seus componentes. Essas especificações são obtidas a partir da resposta em frequência da saída produzida pelo conversor, comparada ao sinal de entrada inserido. As principais especificações dinâmicas são apresentadas abaixo.

### 2.4.1 Relação sinal-ruído

A relação sinal-ruído, ou Signal to Noise Ratio (**SNR**), pode ser definida pela razão entre o sinal aplicado no conversor e os ruídos gerados pelo circuito de conversão, ambos medidos por Root Mean Square (**RMS**). No caso em que há apenas ruídos de quantização no sinal, a **SNR** pode ser simplificada como:

$$SNR_{dB} = 6,02 \times N + 1,76[dB] \quad (2)$$

Em que  $N$  é a resolução do conversor ideal.

Essa equação descreve a **SNR** ideal para um conversor, dependendo apenas da sua resolução. No entanto, para mensurar a SNR de um conversor real deve-se utilizar a equação:

$$SINAD_{dB} = 20 \times \log \frac{V_{in(RMS)}}{V_{noise(RMS)}} \quad (3)$$

A comparação do resultado obtido das duas formas permite avaliar o desempenho do conversor [14].

#### 2.4.2 Relação sinal ruído e distorção

A relação sinal ruído e distorção, ou Signal to Noise and Distortion (**SINAD**), possui uma definição similar a **SNR**. Ela é a relação entre o valor RMS do sinal de entrada e o valor **RMS** dos ruídos e distorções presentes, isso permite incluir harmônicas e componentes não lineares presentes no sinal de entrada. Pode ser expressa pela equação:

$$SNR_{dB} = 20 \times \log \frac{V_{in(RMS)}}{V_{noise(RMS)} + V_{distortion(RMS)}} \quad (4)$$

A **SINAD**, por depender de mais parâmetros da entrada, é diretamente influenciada pela amplitude e frequência do sinal a ser amostrado [15].

#### 2.4.3 Número efetivo de bits

O número efetivo de bits, ou Effective Number of Bits (**ENOB**), expressa a resolução de um conversor ideal que apresentaria a mesma **SINAD** do conversor testado [15]. Com isso o **ENOB** pode ser expresso em função da **SINAD** pela equação:

$$ENOB = \frac{SINAD_{dB} - 1,76}{6,02} \quad (5)$$

Como um ADC real irá apresentar mais erros, distorções e falhas que um ADC ideal, seu ENOB sempre apresentará uma resolução inferior a real.

#### 2.4.4 Faixa dinâmica livre de espúrios

A faixa dinâmica livre de espúrios, ou Spurious Free Dynamic Range (SFDR), é definida como a razão em dB entre a amplitude da frequência fundamental do sinal aplicado e a amplitude do maior sinal de espúrio, isso é, um sinal com frequência não múltipla da entrada. A Figura 8 apresenta a resposta em frequência de um ADC, permitindo visualizar a SFDR como a diferença entre o nível do sinal e o da maior perturbação presente no espectro.

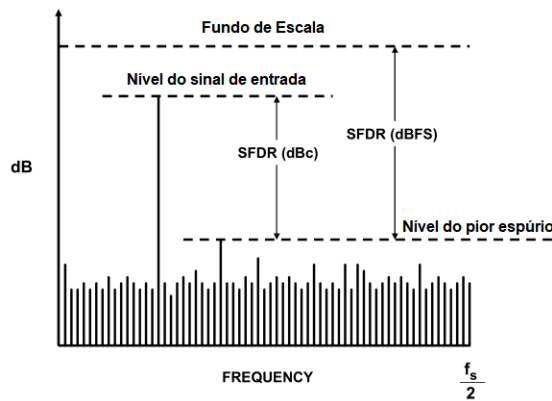


Figura 8: Resposta em frequência de um conversor. Adaptado de [15]

## 2.5 ARQUITETURAS COMUNS

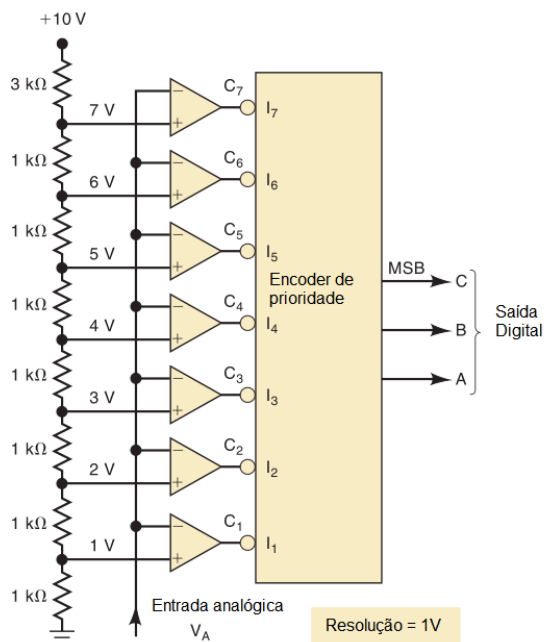
Um conversor analógico-digital pode ser escolhido dentre uma ampla gama de arquiteturas existentes, cabendo ao usuário escolher alguma característica prioritária, como velocidade de conversão, custo ou precisão para guiá-lo em sua decisão. A seção seguinte apresenta alguns dos tipos mais comuns de arquiteturas de conversores existentes: Paralelo, Pipeline, Aproximação sucessivas, Sigma-Delta e Integrador.

### 2.5.1 Paralelo

O conversor paralelo, ou flash, é o ADC disponível com maior velocidade, porém requer um circuito maior, exigindo uma quantidade exponencial de comparadores a medida que a resolução é incrementada.

Esse conversor não necessita de um sinal de clock para seu funcionamento. As conversões são realizadas de forma contínua, uma mudança na entrada gera uma mudança na saída, o tempo de conversão depende apenas dos atrasos de propagação dos comparadores e do circuito de codificação. São ideais para aplicação que utilizam sinais com largura de banda alta, como comunicações, radares e amostragem de osciloscópios. Tendem a ser caros, com baixa resolução e alto consumo de energia.

A Figura 9 apresenta um conversor paralelo, sua estrutura é basicamente composta de diversos comparadores entre o sinal de entrada e uma rede de resistores produzindo uma faixa de tensões conhecida. O valor da entrada codificado depende de quantas comparações produziram nível lógico alto, sendo depois convertidas por um codificador para codificação binária.

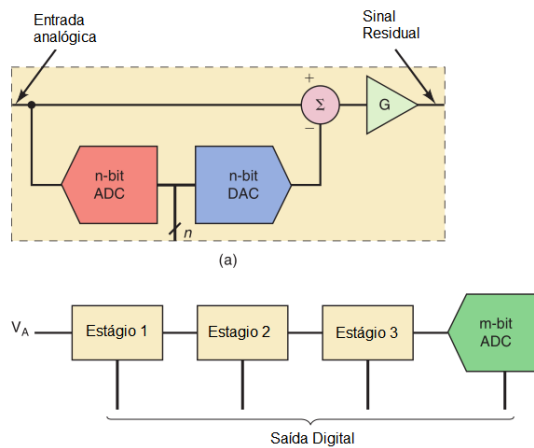


**Figura 9:** Conversor paralelo. Adaptado de [5]

### 2.5.2 Pipeline

Um ADC pipeline, Figura 10, tem esse nome pois é composto por vários estágios de conversão em cascata. Cada estágio é composto por um ADC e um conversor digital-analógico, ou Digital to Analog Converter (DAC), o ADC deve converter a entrada com baixa precisão, enquanto o DAC utilizará o bit mais significativo gerado para produzir um sinal analógico, este sinal será subtraído da entrada, sendo o sinal resultante é amplificado por um ganho residual e enviado ao próximo estágio.

A arquitetura pipeline permite que sejam usados conversores de baixa resolução dentro de cada estágio. Essa característica garante maior velocidade, um custo mais baixo e maior resolução para o conversor completo.



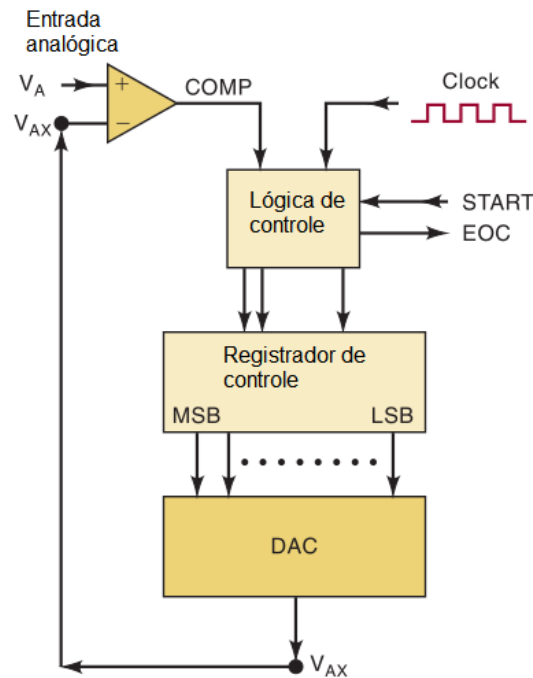
**Figura 10:** Conversor pipeline. Adaptado de [5]

### 2.5.3 Aproximação sucessivas

O conversor por aproximações sucessivas, Figura 11, é um dos mais amplamente utilizados. Ele apresenta um tempo fixo de conversão, que independe do sinal de entrada.

Seu circuito é composto por um DAC, um comparador, um registrador com o valor da conversão e a unidade de controle. Para seu funcionamento, o DAC transforma a saída em um valor analógico e o compara com a entrada, se o valor for superior a entrada o primeiro bit é definido com 0, caso contrário como 1. Isso se repete para cada bit da

resolução do ADC, dessa forma o resultado é gerado do MSB para o LSB em um tempo determinado.



**Figura 11:** *Conversor por aproximação sucessiva. Adaptado de [5]*

#### 2.5.4 *Sigma-delta*

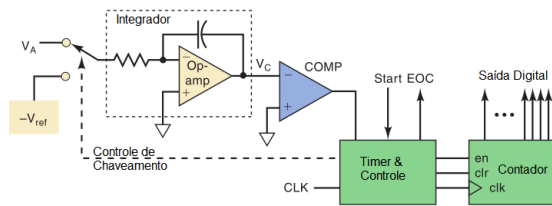
Conversores sigma-delta são baseados na modulação de mesmo nome, combinando altas taxas de amostragem (sobreamostragem) com realimentação negativa e filtragem digital, trocando resolução no tempo por amplitude. São conversores mais robustos a variações nos valores dos componentes, uma vez que empregam apenas um quantizador simples incorporado dentro de um ciclo de realimentação.

Esses conversores são capazes de operar em altas frequências e com alta resolução, sendo muito utilizados no processamento digital de sinais.

### 2.5.5 Integrador

Um conversor integrador, Figura 12, tem como seu componente principal um amplificador operacional capaz de integrar um sinal fornecido. Esse tipo de conversor é menos sensível a ruídos e apresenta menos problemas de não-linearidade, mas seu tempo de conversão é superior ao de outras arquiteturas. Uma de suas principais vantagens é o baixo custo, já que não necessita de muitos componentes.

Em seu funcionamento, a tensão analógica é integrada em um intervalo de tempo fixo, fazendo a tensão no capacitor ser proporcional a entrada. Ao fim desse estágio a entrada do integrado é trocada por um sinal de referência com tensão negativa e este sinal é aplicado até que a saída do comparador seja nula. Um contador é responsável por armazenar o número de ciclos de clock necessário para a descarga do capacitor. Dessa forma o valor do contador será proporcional ao valor da tensão de entrada do conversor.



**Figura 12:** *Conversor integrador. Adaptado de [5]*



# 3 Conversor A/D integrador com faixa de entrada e resolução programável a capacitores chaveados

Este capítulo irá descrever as características do conversor utilizado nesse trabalho. Será feita uma revisão de sua arquitetura, seu funcionamento e suas especificações como desenvolvido no trabalho de [7].

## 3.1 CARACTERÍSTICAS DO CONVERSOR

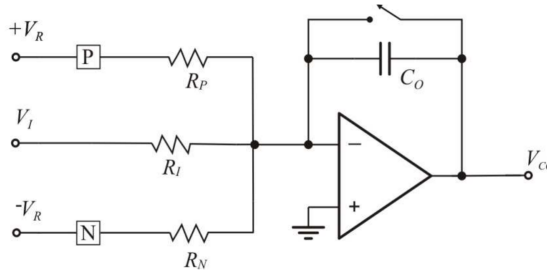
O conversor utilizado é um integrador de rampa múltipla RunUp, utilizando capacitores chaveado ao invés de resistores. Sua faixa de entrada pode ser modificada selecionando o conjunto de ganhos a ser aplicado no sinal de entrada. E sua resolução pode ser definida entre 1 a 8 bits. A seguir cada uma dessas características será explorada.

### 3.1.1 *Conversor A/D integrador de rampa múltipla RunUp*

O conversor integrador de rampa múltipla possui uma resolução superior a de outros conversores integradores, como rampa simples e rampa dupla. Isso é possível pois ele supera as limitações de variação máxima de tensão no integrador e o ruído do circuito em largura de banda.

Nessa técnica é feita a soma da tensão de referência, que pode ser positiva e negativa, no capacitor integrador, gerando múltiplas rampas que mudam de sentido. Esse processo é realizado até que a tensão no integrador seja nula e é chamado de período de *RunUp*.

A arquitetura do conversor é apresentada na Figura 13. Conforme a figura, o conversor possui uma tensão de referência positiva  $+V_r$  associada a chave  $P$ , e uma tensão de referência negativa, associada a chave  $N$ . A tensão de entrada está sempre aplicada ao conversor, enquanto as tensões de referências dependem do acionamento das chaves  $P$  e  $N$ , que é feito a fim de manter a saída do integrador nos limites da faixa de tensão desejada.



**Figura 13:** *Conversor rampa múltipla RunUp. Adaptado de [12]*

As chaves  $P$  e  $N$  operam alternadamente em intervalos de tempo fixos  $\Delta t$ , sendo controladas pelo sinal de um comparador que avalia a tensão na saída do integrador. Caso a tensão na saída seja superior a tensão de comparação a chave  $P$  será acionada para reduzir a tensão no integrador, caso contrário a chave  $N$  será acionada para elevá-la. A conversão estará completa após  $N$  intervalos de tempo, com  $N_p$  sendo o tempo que a chave  $P$  estava ativa e  $N_n$  o da chave  $N$ . A carga total sobre o capacitor ao final da conversão pode ser expressa pela contribuição da entrada e de cada uma das tensões de referência conforme a equação:

$$V_{CO} = -\frac{\frac{NV_I\Delta t}{R_I} + \frac{N_P V_P \Delta t}{R_P} - \frac{N_N V_N \Delta t}{R_N}}{C_O} \quad (6)$$

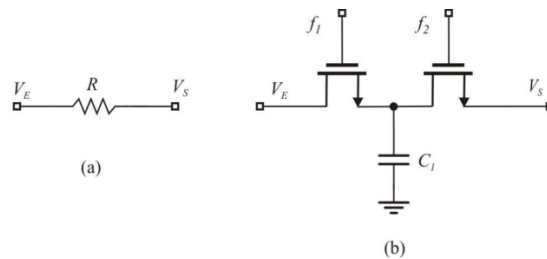
Considerando que ao final da conversão a tensão do integrador será nula, tensão de entrada pode ser expressa pelo número de contagens obtidas conforme:

$$V_I = \left( \frac{N_N}{R_N} - \frac{N_P}{R_P} \right) \frac{R_I}{N} V_R \quad (7)$$

Essa arquitetura permite obter variações de tensão bem superiores ao integrador rampa dupla comum, reduzindo os efeitos das tensões de desvio no amplificador e comparador. No entanto não é possível aumentar a resolução indefinidamente apenas aumentando o tempo de *RunUp*, uma vez que isso pode introduzir erros pelos acionamentos das chaves.

### 3.1.2 Circuitos a Capacitores chaveados

Em circuito integrados não é muito conveniente o uso de resistores, já que estes ocupam uma área muito grande de silício e podem apresentar variações em seu valor com a temperatura. Com isso, uma das formas de substituir resistores é empregar capacitores acionados por chaves, diminuindo a área de circuito e aumentando o controle e a precisão do circuito. Esse circuito pode ser construído com um capacitor e um par de chaves acionadas alternadamente, conforme a Figura 14.



**Figura 14:** Equivalência do resistor em (a), com o uso do capacitor chaveado em (b). Adaptado de [12]

A operação desse circuito consiste de duas fases, na primeira a chave 1 é fechada enquanto a 2 se mantém aberta, na segunda fase o inverso ocorre. No primeiro momento o capacitor é carregado com a tensão  $V_E$ , armazenando a carga  $Q_1 = C \cdot V_E$ , no segundo momento parte da carga será transferida para a saída, sendo o valor final  $Q_2 = C \cdot V_S$ , assim podemos expressar a carga transferida como:

$$\Delta Q = C_I \cdot (V_E - V_S) \quad (8)$$

Expressando em função do tempo, sabendo quantas vezes o chaveamento ocorre em um intervalo definido, obtemos:

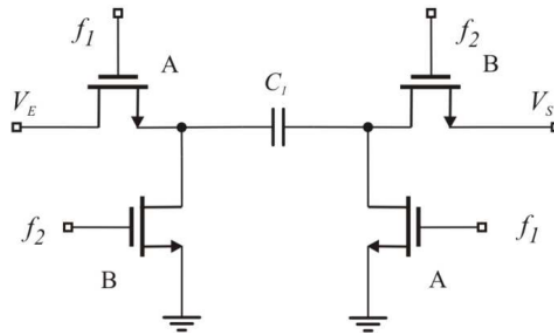
$$i = C_I \cdot (V_E - V_S) \cdot f_{clk} \quad (9)$$

Dessa forma a resistência pode ser descrita como:

$$R = \frac{1}{C_I \cdot f_{clk}} \quad (10)$$

A utilização de capacitores chaveados permite que a resistência seja controlada pela frequência de chaveamento dos capacitores, tornando o valor do resistor mais preciso e flexível.

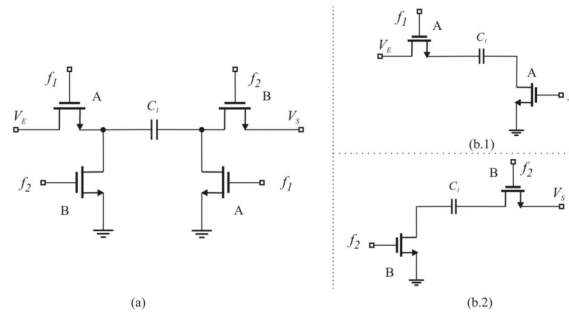
A estrutura mostrada é muito sensível a capacitâncias parasitas e, por isso, não é muito utilizada. A estrutura da Figura 15 é preferível para as aplicações, já que é mais insensível as capacitâncias parasitas presentes no circuito.



**Figura 15:** Outro sistema de capacitores chaveados. Adaptado de [12]

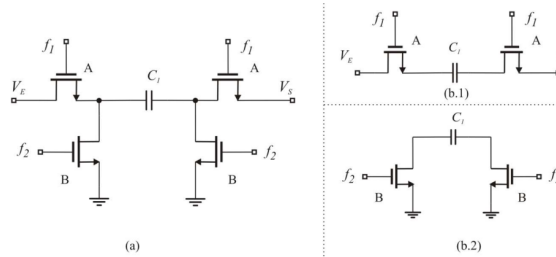
Essa estrutura depende do chaveamento de mais transistores para sua operação, mas apresenta a vantagem de permitir inverter o valor da tensão de entrada

A Figura 16 apresenta o funcionamento do circuito sem inversão da tensão. No primeiro momento, o capacitor é conectado entre a entrada e saída, transferindo uma corrente a corrente para a saída a medida que se carrega. No segundo momento as chave alternam e o capacitor é descarregado.



**Figura 16:** Operação sem inversão de tensão. Adaptado de [12]

Para a tensão invertida é utilizada a configuração da Figura 17. Inicialmente o capacitor é carregado com a tensão de entrada, e em seguida é injetada a tensão com polaridade invertida na saída do circuito.



**Figura 17:** Operação com inversão de tensão. Adaptado de [12]

### 3.1.3 Faixa de entrada ajustável

Um dos diferenciais do conversor utilizado é sua faixa de entrada ajustável, que permite utilizar sinais com faixas menores sem perda de resolução.

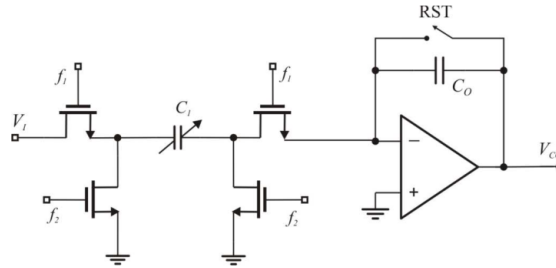
A faixa de entrada de um conversor analógico-digital, supondo a utilização em terminação única, varia de 0 até a tensão de referência do mesmo. Quando a faixa do sinal de entrada é inferior a do ADC ocorre uma perda de resolução, que pode ser quantificada como:

$$N_L = \log_2 \left( \frac{\Delta ADC}{\Delta V_I} \right) \tag{11}$$

Em que  $\Delta ADC$  corresponde à faixa de entrada do conversor e  $\Delta V_I$  à faixa do sinal aplicado.

O conversor utilizado neste trabalho foi projetado para uma perda de resolução máxima de 1 bit, com medição da faixa completa do sinal de entrada por meio do ajuste do ganho aplicado a este no processo de conversão.

O ajuste do ganho é feito modificando o valor da capacitância de entrada do circuito,  $C_i$  apresentado na Figura 18.



**Figura 18:** Capacitância variável na entrada do integrador. Adaptado de [12]

Os possíveis valores de capacitância são apresentados abaixo, juntamente com ganho a elas associado.

**Tabela 1:** Relação entre capacitor de entrada, ganho de sinal e faixa de entrada.

Capacitor de entrada ( $fF$ )	Ganho de sinal ( $C_I/C_R$ )	Faixa de entrada ( $V$ )
200	1,0	0 a 2,56
300	1,5	0 a 1,70
400	2,0	0 a 1,28
600	3,0	0 a 0,85
800	4,0	0 a 0,64
1200	6,0	0 a 0,42
1600	8,0	0 a 0,32
2400	12,0	0 a 0,21
3200	16,0	0 a 0,16

O circuito possui um capacitor de saída de  $600fF$ , e o capacitor de referência de  $200fF$ . Dessa forma, e sabendo que a tensão de referência é de  $2.56V$  enquanto a tensão

de comparação é de  $0.85V$ , as faixas de entrada possíveis podem ser calculados pelas equações:

$$V_I = V_R \cdot \frac{C_R}{C_I} \quad (12)$$

$$V_I = V_{CO} \cdot \frac{C_O}{C_I} \quad (13)$$

Correspondendo novamente à tabela 1, compilando as faixas de entrada possíveis.

#### 3.1.4 Resolução programável

A resolução programável deste conversor é obtida controlando a resolução do contador associado a saída do integrador. Conforme apresentado na estrutura de um conversor integrador, um contador é responsável por acompanhar o número de ciclos de relógio que cada chave foi acionada e apresentar o valor ‘saida, como resultado da conversão.

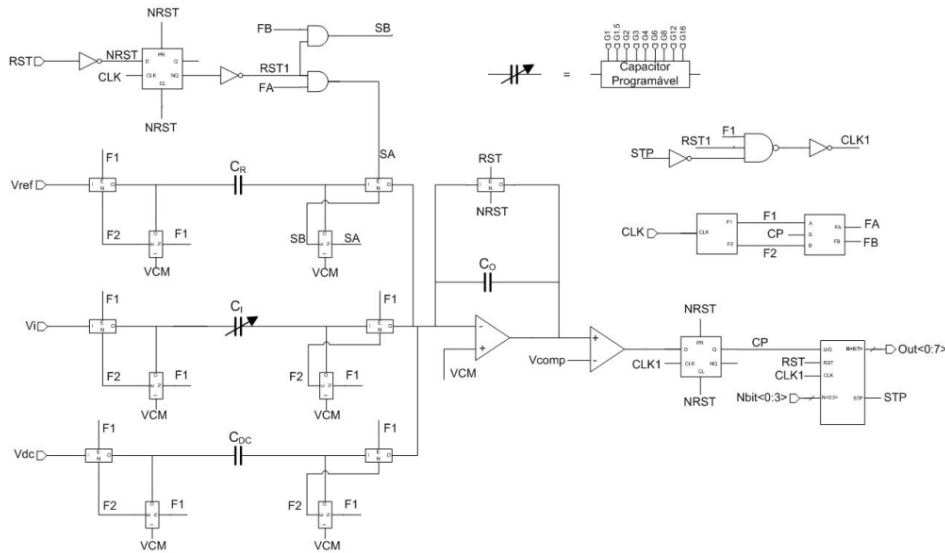
O conversor utilizado nesse trabalho teve seu contador projetado em Verilog, com uma resolução ajustável externamente até 8 bits. Dessa forma é possível obter uma resolução diferente apenas controlando o valor de algumas entradas do conversor, usando uma resolução menor para acelerar a conversão quando for necessário, ou uma resolução maior para um resultado mais preciso.

## 3.2 ARQUITETURA E FUNCIONAMENTO DO CONVERSOR

Nesta seção será descrito o funcionamento interno do conversor, apresentando como a arquitetura rampa múltipla RunUp foi combinada com o uso de capacitores chaveados, e como utilizar o conversor em uma medição.

### 3.2.1 Arquitetura

A arquitetura do conversor utilizado neste trabalho é apresentada na Figura 19. Nela é possível verificar a arquitetura rampa múltipla utilizada no conversor integrador, porém os resistores foram substituídos por capacitores chaveados  $C_I$ ,  $C_R$  e  $C_{DC}$ .



**Figura 19:** Arquitetura do conversor A/D rampa múltipla a capacitor chaveado. Adaptado de [7]

O capacitor  $C_I$  é programável, podendo ter seu valor escolhido a partir de um banco de capacitores, que permite variar a faixa de entrada do conversor.

A tensão  $V_{CM}$  é utilizada como terra virtual do circuito para evitar erros de injeção de carga nos estágios do amplificador operacional. Essa característica provoca um desvio que é corrigido pelo ajuste DC do conversor.

Após o estágio do integrador há a presença de um comparador, responsável por determinar se a tensão de saída está acima ou abaixo da tensão de comparação  $V_{COMP}$ . Esse sinal que irá controlar a alternância do acionamento das chaves  $SA$  e  $SB$  responsáveis por inverter a tensão de referência.

Além disso, o sinal do comparador irá indicar se o contador deve incrementar ou decrementar a contagem de ciclos.

O circuito recebe apenas uma entrada de clock, mas possui um gerador de fases capaz de produzir dois sinais de clock de mesma frequência, porém deslocados em  $180^\circ$  para controlar as chaves de forma alternada,  $F1$  e  $F2$ . As chaves  $SA$  e  $SB$  também operam



a partir desses sinais, porém a ordem irá depender do resultado do comparador, caso a saída do comparado esteja alta  $SA$  e  $SB$  recebem os sinais  $F1$  e  $F2$  respectivamente, mas caso ela esteja em nível baixo receberão os sinais invertidos.

### 3.2.2 Funcionamento

Inicialmente, o sinal de clock de 1 Mhz deve ser fornecido ao conversor e a entrada RST deve receber o nível 1. Isso fará com que o capacitor do integrador seja descarregado, enquanto o contador é zerado e os outros componentes retornam ao estado inicial. Após RST retornar ao nível 0 a conversão é iniciada.

No primeiro ciclo, com a fase  $F1$ , o capacitor  $C_I$  é carregado com a tensão de entrada. Na fase  $F2$  a carga em  $C_I$  será transferida para  $C_O$ , completando um ciclo de relógio. Em cada ciclo ocorre a transferência de carga da entrada para o integrador.

A partir do segundo ciclo, a depender do resultado do comparador, a tensão de referência será aplicada sobre o conversor. Caso o comparador apresente o nível alto, as chaves serão fechadas na sequência:  $SA$  na primeira fase e  $SB$  na segunda, incrementando a tensão sobre o capacitor de saída por meio da transferência da carga em  $C_R$  para  $C_O$ .

No caso de a tensão de saída estar abaixo da de comparação, as chaves serão acionada na sequência:  $SB$  na primeira fase e  $SA$  na segunda, decrementando a tensão na saída.

Ao fim de cada ciclo, o resultado do comparador é utilizado para incrementar ou decrementar o contador, dependendo de seu nível lógico e a conversão prossegue até atingir  $N$  ciclos. O número de ciclos pode ser expresso como  $N = 2^r + 1$ , sendo  $r$  a resolução do conversor, escolhida pelas entradas digitais do mesmo.

Para esta arquitetura, a relação entre a tensão de entrada e as contagens pode ser expresso pela equação:

$$V_I = \left( \frac{N_N - N_P}{N} \right) \cdot \frac{C_R}{C_I} \cdot V_R \quad (14)$$

O valor da diferença  $N_N$  e  $N_P$  é calculado diretamente pelo contador, já que ele é incrementado para comparações negativas e decrementado para comparações positivas. Além

disso, com o valor de  $C_R$  fixo, percebe-se que variando  $C_I$  por meio do banco de capacitores será possível obter ganhos diferentes, modificando a faixa de entrada do conversor.

### 3.2.3 Especificações técnicas

O conversor deste trabalho possui resolução de até 15 bits, podendo ser ajustado através da chave NBIT apresentada na Figura 18, contudo seu desempenho por simulações só foi assegurado até 8 bits.

O ajuste de resolução envolve apenas a modificação no limite do contador, permitindo diminuir a resolução para aplicações que exigem maior velocidade, ou aumentá-la para obter maior precisão.

A Tabela 2 apresenta as especificações do conversor utilizado:

**Tabela 2:** *Especificações do conversor.*

<b>Perda de resolução máxima</b>	1 Bit
<b>Ganho mínimo</b>	1
<b>Ganho máximo</b>	32
<b>Faixa de entrada máxima</b>	0 à 2,56 V
<b>Resolução</b>	8 Bits
<b>Frequência do relógio</b>	1 MHz
<b>Tensão de alimentação</b>	3,3 Volts

É possível utilizar o conversor em 4 modos de operação conforme a tabela 3. Como o conversor opera com um terra virtual  $V_{cm}$ , é sempre necessário a presença de uma tensão de ajuste DC para compensar esse ganho.

**Tabela 3:** *Modos de operação.*

Modos de operação	Entrada $V+$	Entrada $V-$	Entrada $V_{DC}$
Terminação única	$V_{I+}$	$V_{CM}$	$2 * V_{CM}$
Diferencial	$V_{I+}$	$V_{I-}$	$3 * V_{CM}$
Terminação única com ajuste CC	$V_{I+}$	$V_{CM}$	$2 * V_{CM} + V_{DC}$
Calibração	$V_{DC}$	$V_{CM}$	$2 * V_{CM}$

## 4 Testes em conversores analógico-digitais

Neste capítulo serão descritos os testes realizados para validação do conversor analógico-digital em estudo. Embasados na norma IEEE 1241-2010 [11] os testes em ADC's são utilizados para encontrar as especificações estáticas e dinâmicas, bem como analisar possíveis fontes de erros e avaliar os limites de operação.

O processo de teste de conversores deve levar em consideração alguns elementos, entre eles temos: interface com o usuário, condições de teste, ambiente elétrico, equipamentos de teste e a seleção do teste.

O cuidado com a interface com o usuário é de suma importância para assegurar a qualidade dos resultados extraídos. Os conversores analógico-digitais são dispositivos Mixed-signal, que operam com sinais analógico e digitais, cabendo ao usuário o cuidado de separar os dois domínios para evitar interferência e ruídos que comprometam a integridade dos sinais.

Circuitos de alimentação, a interconexão de sinais analógicos, formas de leitura e escrita de sinais digitais, circuitos de clock, efeitos de linha de transmissão estão entre os elementos que devem ser avaliados.

Para as condições de teste, a norma deixa a escolha do fabricante ou do testador o modo de operação do conversor. O usuário deve estar ciente dos intervalos de operação recomendados pelo fabricante ao escolher a faixa de testes desejada. As folhas de especificações, datasheets, fornecem os parâmetros de operação dos ADC como: tensão de operação, temperatura, frequência, tensão de referência entre outros. Ao se realizar um teste, é essencial indicar as condições de operação no momento de informar os resultados, para permitir comparações futuras.

O ambiente elétrico no qual os testes serão executados deve prevenir que distúrbios como ruídos e interferências eletromagnéticas venham a impactar no teste. Os equipamentos e instrumentos devem ser calibrados antes da realização dos testes a fim de evitar erros.

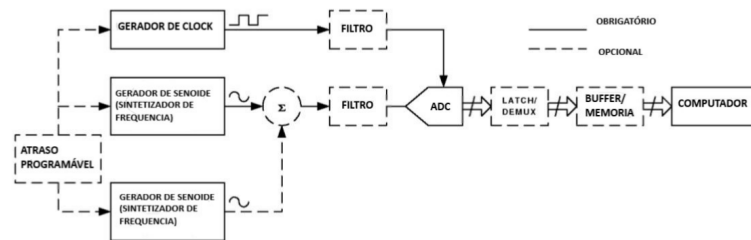
A escolha do teste é determinada pelos parâmetros de interesse do usuário, cabendo a ele decidir o método que melhor captura as características de interesse.

#### 4.1 CONFIGURAÇÕES DE TESTES

Existem 3 configurações comuns para teste de conversores analógico-digitais: onda senoidal, sinal arbitrário e degrau

##### 4.1.1 Configuração de teste de onda senoidal

Este método permite analisar o desempenho do conversor com sinais senoidais, avaliando sua precisão e velocidade. Para isso um gerador de sinais é utilizado para produzir o sinal analógico de interesse. Conforme apresentado na Figura 20.



**Figura 20:** Setup para sinais senoidais. Adaptado de [11]

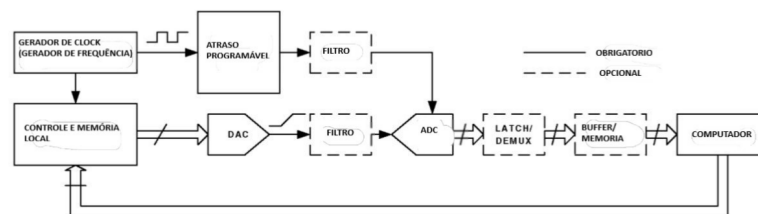
Essa configuração permite combinar dois ou mais sinais de entrada, com intuito de gerar um sinal com distorção intermodulada. Além disso, também é possível inserir ruídos para analisar o comportamento e a rejeição de ruídos pelo ADC.

Alguns elementos são essenciais para execução de testes com essa configuração, como: ADC, computador, gerador de sinais e de clock. No entanto os filtros são opcionais, bem como os latches e buffers de memória.

#### 4.1.2 Configuração de teste de sinal arbitrário

Essa configuração permite o teste com uma ampla gama de sinais, podendo ser utilizado para entradas de tipo rampa, degrau e ondas com frequências variadas.

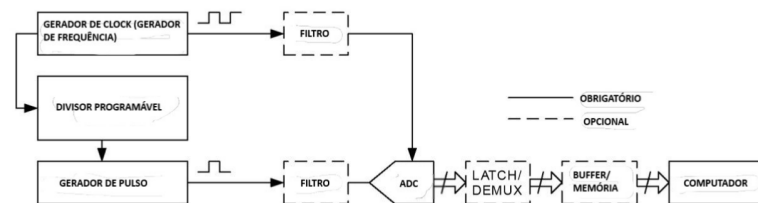
A Figura 21 apresenta essa configuração de testes. Nela é possível verificar que o computador responsável pela leitura dos dados do ADC também é responsável pela geração dos sinais por meio de um DAC de alta resolução. Dessa forma sinais arbitrários podem ser definidos pelo algoritmo desejado e criados pelo DAC.



**Figura 21:** Setup para sinais arbitrários. Adaptado de [11]

#### 4.1.3 Configuração de teste de degrau

Esta configuração, Figura 22 é utilizada para testes de sinais do tipo pulso ou degrau, a fim de obter parâmetros como overshoot, tempo de subida, tempo de transição no domínio do tempo, e amplitude, fase e largura de banda no domínio da frequência.



**Figura 22:** Setup para degrau. Adaptado de [11]

## 4.2 MÉTODO DE TESTES

Existe uma ampla gama de testes disponíveis para análise de conversores analógico-digitais, cabendo a escolha do método de acordo com os parâmetros de interesse. Neste

trabalho serão estudados as especificações estáticas e dinâmicas descritas em 2. Com isso, foram escolhidos os métodos de localização de transição de código, para parâmetros estáticos, e o teste de FFT para os dinâmicos.

O método de localização de transição de código consiste em representar a curva de transferência do conversor e avaliar seus desvios em relação a curva ideal. De acordo com a norma, na análise de parâmetros estáticos são recomendados os métodos de loop de realimentação, determinação da curva de transferência estática, método de histograma de rampa senoidal e o de rampa, sendo o ultimo o escolhido neste trabalho. Para os parâmetros dinâmicos será utilizado o teste de FFT para obter a resposta em frequência.

#### 4.2.1 Método de histograma de rampa

Nesse método de testes é aplicado um sinal de rampa variando de 0 Volts até a tensão de fundo de escala do conversor, cobrindo todo espectro contínuo. Com o grande número de amostras geradas é possível montar um histograma no qual as classes são os códigos binários na saída.

A configuração de sinal arbitrário é a utilizada para esse tipo de teste, com o sinal sendo produzido por um DAC de resolução superior ao ADC em teste. O grande número de amostras permite obter a medida precisa do DNL e conseqüentemente do INL. Os outros parâmetros estáticos, como erro de ganho e offset, podem ser calculados com os dados amostrados. Os níveis de transição dos código podem ser obtidos pelas equações:

$$T[K] = C + A \cdot H_C[K - 1] \quad (15)$$

$$H_C[j] = \sum_{i=0}^j H[i] \quad (16)$$

$$A = \frac{T[2^N - 1] - T[1]}{S} \quad (17)$$

$$C = T[1] \quad (18)$$

Sendo:  $A$  o fator de ganho;  $C$  o fator de *offset*;  $H[i]$  é igual ao número de amostras para um código binário;  $S$  é o número total de amostras.

A precisão dos valores obtidos para os níveis de transição de código depende do número de amostras. Desse modo, quanto maior for o número de amostras obtidas, menor a incerteza em relação ao sinal de entrada em rampa.

#### 4.2.2 *Teste de FFT*

A análise de parâmetros dinâmicos é feita no domínio da frequência, dessa forma uma das ferramentas utilizadas para esse processo é a transformada rápida de Fourier (FFT), que é uma forma de realizar a transformada discreta de Fourier de forma rápida e eficiente.

A FFT mapeia um sinal de entrada e o decompõe no domínio da frequência, representando suas componentes fundamentais, as harmônicas, a frequência de amostragem e os ruídos captados. A partir dos níveis de amplitude de cada um desses sinais é possível calcular os parâmetros dinâmicos.

Os sinais produzidos no domínio da frequência são divididos proporcionalmente ao número de pontos da FFT. Por conta da periodicidade e simetria da FFT a frequência de amostragem deve ser no mínimo 2 vezes superior ao sinal de entrada, como explicado pelo teorema de Nyquist-Shannon.

A frequência do sinal de entrada pode ser determinada por:

$$f_i = \frac{f_S}{N_{FFT}}k \quad (19)$$

No qual:  $f_i$  é a frequência de entrada;  $f_S$  é a frequência de amostragem;  $N_{FFT}$  é o número de pontos para FFT;

O valor de  $k$  representa as possíveis frequências do sinal de entrada e deve ser escolhido com cautela. Ele não pode ser zero, senão apenas o valor DC seria selecionado, e não pode ser muito grande para que erros dinâmicos e de aliasing não se apresentem. Ou seja:

$$0 < k < \frac{N_{FFT}}{2} \quad (20)$$



Dessa forma, o número de pontos para a FFT fica a escolha do usuário, o único requisito que seja uma potência de 2.

Esse método será utilizado para observar o piso de ruído obtido na resposta do conversor. O piso de ruído é a faixa média, em dB, de todos os sinais indesejados obtidos na resposta em frequência.

## 5 Circuito de testes

Neste capítulo será descrito o ambiente físico desenvolvido para o teste do conversor. Serão apresentados os circuitos de alimentação para o conversor, responsáveis por fornecer as tensões e correntes necessárias para sua operação, o circuito de um conversor digital-analógico, responsável por prover os sinais de referência, e a interface com o microcontrolador utilizado.

### 5.1 CIRCUITO DE ALIMENTAÇÃO

O conversor analógico-digital utilizado neste trabalho não possui reguladores de tensão internos, com isso cada um dos valores de tensão e corrente necessários para o funcionamento de seus componentes deve ser fornecido externamente. Isso o difere de um conversor comercial, que necessita apenas de uma alimentação positiva, mas permite que os limites técnicos do conversor sejam testados mais rigorosamente.

O fornecimento de potência é feito para o circuito de alimentação primordialmente por uma fonte de 5V, constituída por um carregador de celular comum. Esse valor foi escolhido pois reduz o risco de queima de componentes em caso de conexões erradas, além de reduzir a potência a ser dissipada nos reguladores de tensão utilizados posteriormente.

Além disso, utilizou-se uma fonte de 12V para computadores portáteis como uma fonte simétrica, a fim de fornecer as tensões de +6V e -6V para o uso nos amplificadores operacionais do circuito de alimentação. Foi utilizado um divisor de tensão simples nos terminais de alimentação da fonte, com o terra virtual sendo conectado ao terra da fonte de 5V. A Figura 23 apresenta essa configuração.

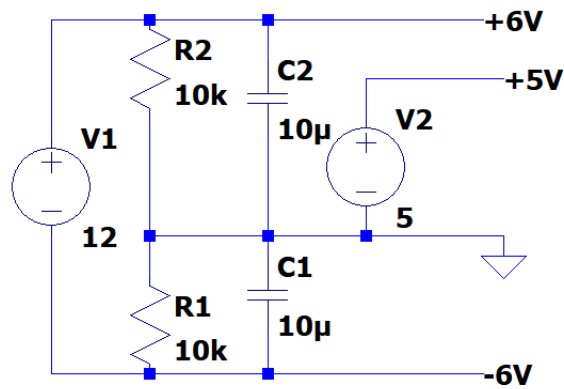


Figura 23: Fontes de alimentação.

O conversor utilizado neste trabalho precisa das tensões apresentadas na Tabela 4 para seu funcionamento.

Tabela 4: Tensões de alimentação para o ADC.

Tensão de alimentação ( $V_{DD}$ )	3,3 V
Tensão de modo comum ( $V_{CM}$ )	440mV
Tensão de ajuste DC ( $V_{DC}$ )	880mV
Tensão de comparação ( $V_{comp}$ )	1,29V
Tensão de polarização do amplificador ( $V_{BN}$ e $V_{BP}$ )	1,5V
Tensão de referência ( $V_{ref}$ )	3V

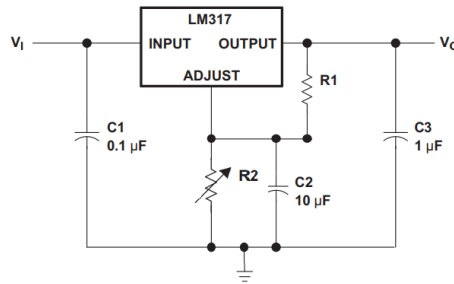
Além disso, duas corrente de polarização são necessárias, Tabela 5 para o comparador e o amplificador presentes.

Tabela 5: Correntes de polarização para o ADC.

Para o amplificador ( $I_{BIASA}$ )	$8\mu A$
Para o comparador ( $I_{BIASC}$ )	$50\mu A$

## 5.1.1 Regulador de tensão

O CI LM317 foi escolhido como regulador para produzir as tensões necessárias. Sua faixa de operação é entre 1.25V e 37V, permitindo ser utilizado para a maior parte dos valores necessário. O circuito utilizado para regulação de tensão é apresentado na Figura 24.



**Figura 24:** LM317 como regulador ajustável de tensão.

Nessa configuração, a tensão  $V_{out}$  pode ser expressa a partir do valor das resistências  $R1$  e  $R2$ , conforme a equação:

$$V_O = 1,25V \times \left(1 + \frac{R2}{R1}\right) + (I_{ADJ} \cdot R2) \quad (21)$$

A corrente  $I_{ADJ}$  possui o valor de  $50\mu A$ , fornecida pelo pino adjust, sendo comum desprezá-la quando as resistências utilizadas possuem valores baixos. Os capacitores apresentados servem para filtrar perturbações e estabilizar a tensão sobre os terminais do regulador.

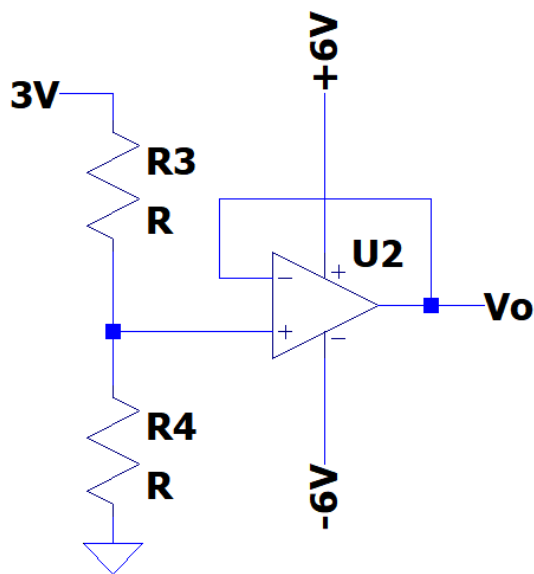
Para os reguladores foram utilizados trimmers resistivos no local de  $R1$ , afim de permitir o ajuste fino da tensão, já que permite utilizar valores não comerciais de resistência.

A Tabela 6 apresenta os valores de tensão necessários, bem como os valores de  $R2$  escolhidos e o valor de  $R1$  necessário para o regulador.

**Tabela 6:** Resistências utilizadas para os reguladores.

Tensão	R1 $\Omega$	R2 $\Omega$
3,3V	330	541,2
3V	330	462
1,5V	1k	200
1,29V	10k	320

As tensões  $V_{CM}$  e  $V_{DC}$  ocupam valores abaixo do limite inferior de operação do LM317, com isso utilizou-se um divisor de tensão e um amplificador como buffer para produzi-las, Figura 25. A tensão de alimentação de 3V foi escolhida como alimentação nesse caso.

**Figura 25:** Circuito para tensões abaixo de 1,25V.

A Tabela 7 abaixo apresenta os valores das resistências utilizadas.

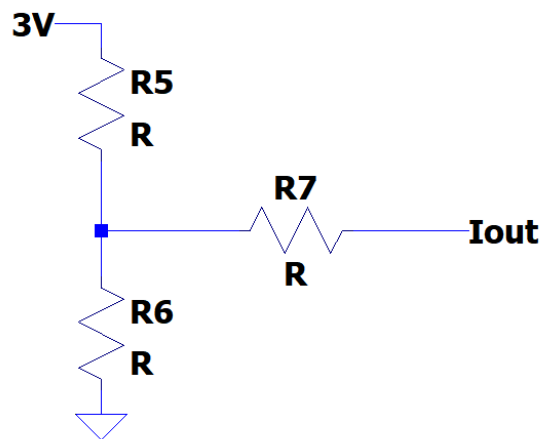
**Tabela 7:** Resistências utilizadas para os reguladores.

Tensão	R3 $\Omega$	R3 $\Omega$
880mV	4,7k	1,95k
440mV	8,53k	1,46k

## 5.2 CORRENTES DE POLARIZAÇÃO

O comparador e o integrador presentes no conversor analógico-digital necessitam de uma corrente de polarização para sua operação. Essas correntes, IBIASC e IBIASA devem ser de  $50\mu A$  e  $8\mu A$  respectivamente, a serem inseridas nos pinos do conversor.

A fim de produzir essas correntes, foi desenvolvido o circuito apresentado na Figura 26, composto por um trimmer resistivo ( $R5$  e  $R6$ ) e um resistor de alto valor ( $R7$ ).



**Figura 26:** Circuito para fonte de corrente.

É importante salientar que as correntes serão aplicadas a um transistor interno ao circuito do conversor, com isso há uma pequena queda de tensão a ser considerada. Além disso, como os valores de corrente são muito baixos, e os equipamentos de medição disponíveis não possuem resolução nem precisão suficiente para representá-los, a corrente foi aferida indiretamente pela tensão sobre o resistor  $R8$ .

A Tabela 8 apresenta os valores de resistências escolhidos, bem como as tensões desejadas para o circuito.

**Tabela 8:** Parâmetros para fontes de corrente.

Corrente $\mu A$	R7 $\Omega$	Tensão V
8	50k	400m
50	10k	500m

### 5.3 CHAVES

O ajuste da faixa de entrada e da resolução do conversor é feito por meio de chaves digitais. Existem 9 entradas digitais, representando as 9 possíveis faixas de entrada, que são selecionadas em nível alto. Além disso, para o ajuste da resolução existem 4 bits a serem ajustados para o tamanho do contador, caso o objetivo seja uma resolução de 8 bits por exemplo, deve-se aplicar o valor 1000 sobre esses pinos.

Para essas entrada foi utilizado um resistor de pull-down de  $10k\Omega$ , enquanto as entradas escolhidas foram conectadas a tensão de 3.3V.

### 5.4 CIRCUITOS DIGITAIS

O conversor utiliza um clock de 1MHz em sua operação, seja para o processo de conversão como para o envio serial do resultado. Esse clock foi produzido a partir de um sinal de PWM da placa microcontrolada utilizada, gerando um sinal com a frequência desejada e duty cycle de 50%.

A geração dos sinais analógicos para a leitura do conversor é feita por meio de um DAC. O modelo escolhido foi o MAX541 com resolução de 16 bits, para que a resolução dos sinais gerados não seja um fator limitante nos testes. Dessa forma é possível varrer a faixa de entrada do conversor com passos inferiores aos níveis de quantização do mesmo.

Esse DAC utiliza comunicação SPI, permitindo uma interface simples com microcontroladores. Os principais parâmetros do conversor estão descritos na Tabela 9:

Tabela 9: Parâmetros DAC.

Tensão de operação	-0,3 a 6V
Tensão de Referência	2 a 3V
Faixa de Tensão de Saída	0 a $V_{ref}V$
Resolução	16 Bits
Frequência de clock	até 10MHz
Erro de desvio	$\pm 1LSB$
Erro de ganho	$\pm 5LSB$

O DAC foi utilizado na configuração unipolar, Figura 27, isto é, produzindo apenas tensões positiva. A faixa utilizada foi de 0 a 2.56V, a maior que o ADC é capaz de utilizar, sendo a configuração apresentada abaixo.

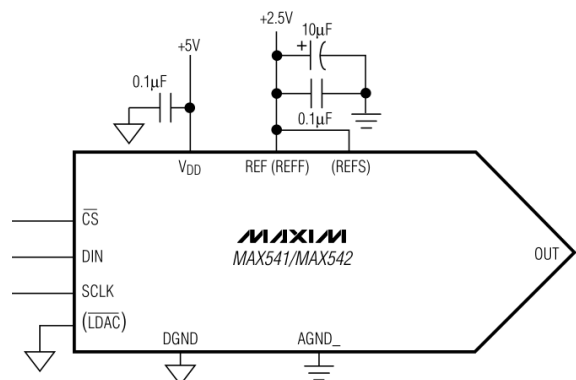
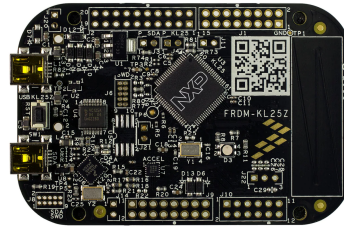


Figura 27: DAC MAX541.

#### 5.4.1 Freescale Freedom FRDM-KL25Z

A placa Freescale Freedom FRDM-KL25Z, Figura 28, foi utilizada para interface com o ADC. Ela é uma plataforma de baixo custo para avaliação dos microcontroladores da série Kinetis L, que são microcontroladores ARM Cortex-M0+. O microcontrolador presente nesta placa é o MKL25Z128VLK4, que possui 128 KB de memória flash, 16 KB de memória SRAM, USB OTG (FS) e pode operar a até 48 MHz.





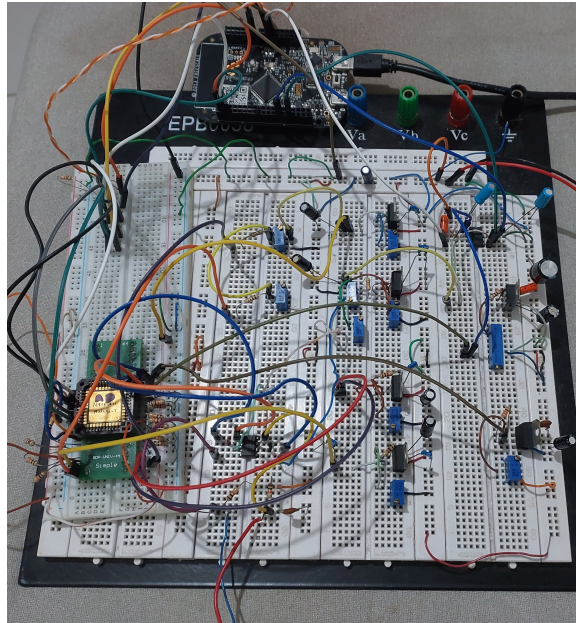
**Figura 28:** *Freescale Freedom FRDM-KL25Z.*

A placa possui diversos recursos que auxiliam no desenvolvimento de aplicações interessantes utilizando os periféricos integrados, tais como diversas opções de alimentação – USB, bateria, Fonte externa, pontos de acesso para medição, facilidade de acesso aos pinos I/O e pinagem compatível com Arduino R3, entre outros.

Para programação/depuração está disponível na placa a interface OpenSDA, um padrão aberto para programação/depuração. Para programação, é possível utilizar diversas plataformas dentre elas o MCUXpresso na própria NXP ou o mbed da ARM. A escolha desta placa ocorreu devido à sua compatibilidade com a tensão de operação do ADC, por possuir o periférico SPI, e sua capacidade de lidar com mais informações em alta velocidade, devido ao clock e arquitetura de 32 bits.

#### 5.4.2 *Circuito de testes completo*

O circuito de teste com todos seus componentes montados em placas protoboard é apresentado na Figura 29.



**Figura 29:** *Circuito de teste.*

## 6 Algoritmos de teste

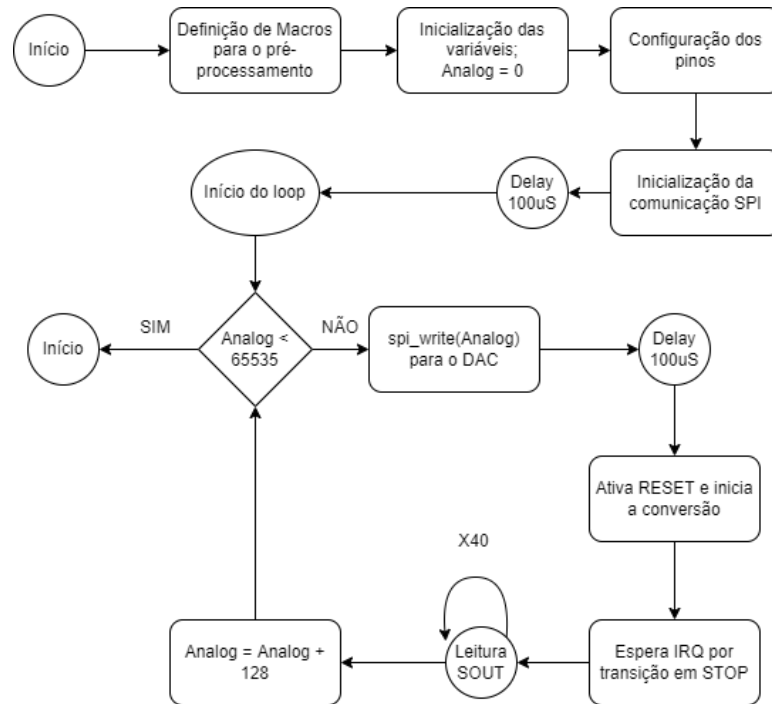
Nesta seção são descritos os algoritmos responsáveis por gerar os sinais analógicos desejados e como é feita a leitura dos dados digitais produzidos pelo ADC.

### 6.1 ALGORITMO DE SINAL DE RAMPA

O primeiro algoritmo consiste em gerar um sinal de rampa, por meio do DAC MAX541 controlado via SPI pela KL25Z. Conforme descrito na seção 2 o sinal de rampa deve variar de 0 Volts até a tensão de fundo de escala do ADC 2,56 Volts. O algoritmo utiliza uma variável de 16 bits representando o dado a ser enviado ao DAC, iniciando em zero.

Após o envio do dado para o DAC, o algoritmo espera alguns microssegundos para a estabilização do sinal. Em seguida o microcontrolador inicia o processo de conversão, levando o pino RESET ao nível alto. Feito isso, o algoritmo espera uma interrupção por mudança de nível no pino STOP, indicando o fim da conversão. Ao reconhecer o nível alto em STOP, o microcontrolador ativa a saída LOAD para que o ADC envie os dados de seu buffer interno de maneira serial pelo pino SOUT.

Ao fim desse processo, o microcontrolador reconhece os níveis lógicos produzidos na saída, os envia ao computador, incrementa a variável enviada ao DAC e o processo se reinicia. O fluxograma na Figura 30 representa esse processo.



**Figura 30:** Algoritmo simplificado de rampa.

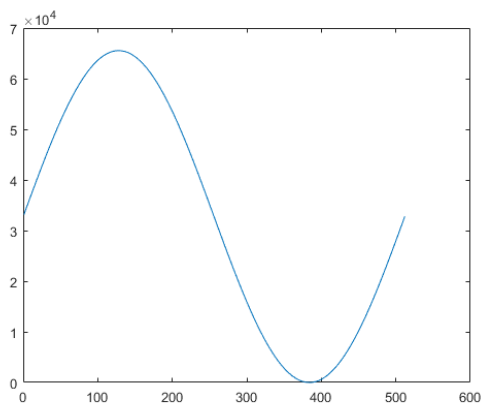
A saída serial do ADC não obedece a um protocolo específico, com isso é necessário cuidado com o sincronismo dos dados. O ADC transfere os dados com uma frequência de 1 MHz, enquanto a KL25Z opera a 48 MHz, dessa forma múltiplas amostragens dos pinos de entrada do microcontrolador são feitas a fim de captar toda e qualquer mudança no pino SOUT.

## 6.2 ALGORITMO DE SINAL SENOIDAL

O segundo algoritmo segue o mesmo princípio básico de funcionamento do primeiro, uma palavra é enviada ao DAC para gerar a tensão analógica e o ADC realiza a conversão em seguida. A diferença está que os sinais enviados ao ADC seguem a forma de uma senoide.

A definição da senoide foi feita subdividindo-a em 512 pontos e obtendo o código de 16 bits equivalente a tensão em cada ponto. Cabe ressaltar que a onda possui um ajuste DC, já que deve estar contida na faixa do ADC, que é puramente positiva.

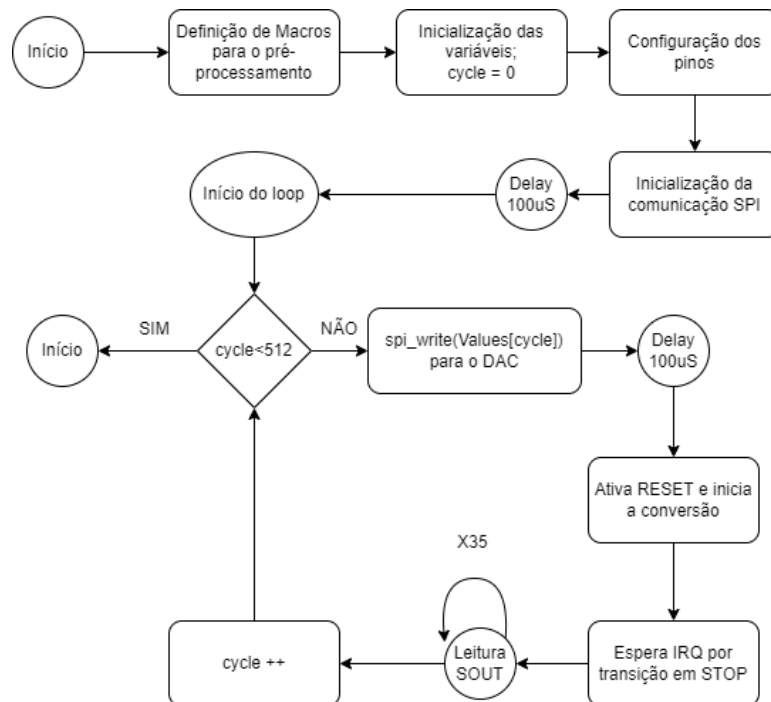
A Figura 31 apresenta o exemplo de senoide construída.



**Figura 31:** *Senoide com 512 níveis lógicos.*

Os 512 valores foram definidos como uma vetor a ser percorrido no código, avançando de posição a cada medição realizada.

O algoritmo resumido de operação é descrito na Figura 32.



**Figura 32:** *Algoritmo simplificado de senoide.*

## 7 Resultados

Nesse capítulo são apresentados os resultados obtidos com a medição do conversor nos cenários propostos. Os dados coletados foram separados de acordo com a resolução utilizada no teste (5 ou 8 bits), a faixa de entrada utilizada (2,56V ou 1,7V) e o tipo de teste (estático ou dinâmico)

Para o ajuste da resolução, os valores de  $NBIT$  foram definidos em 0101 para 5 bits e 1000 para 8 bits. O ajuste da faixa de entrada foi feito ao selecionar o ganho de  $G1$ , para 2,56V, ou de  $G1,5$  para 1,7V.

Em todos os casos, a tensão  $V_I-$  foi mantida em  $440mV$ , sendo compensada pela tensão de ajuste DC mantida em  $880mV$ .

A sincronização entre o microcontrolador e o ADC se mostrou um possível ponto de falha no projeto. Com isso, escolheu-se utilizar uma interrupção por detecção de borda para reconhecer STOP e o fim da conversão, enquanto para assegurar a captura dos dados seriais em SOUT são feitas 40 leituras sequenciais.

Para todos os casos de testes serão obtidos os valores de DNL, INL, SNR e ENOB. Os valores ideais de DNL e INL são 0, com isso será observado quais os valores que mais se afastaram desse ideal, tanto na escala positiva como negativa. Para SNR e ENOB se espera os valores mais próximos possíveis da resolução testada.

Em seguida serão apresentados os resultados obtidos, eles serão divididos seguindo a hierarquia: Resolução, Faixa de entrada, e por fim tipo de teste.

### 7.1 RESOLUÇÃO DE 5 BITS

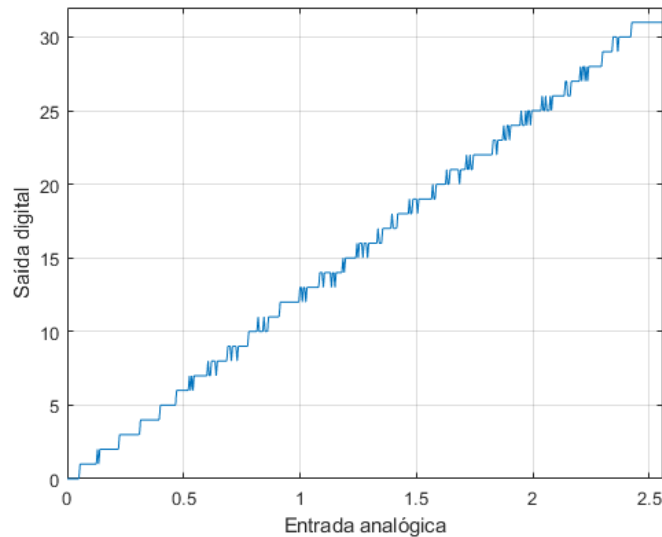
Os primeiros testes foram feitos utilizando a resolução de 5 bits para o ADC. Para isso a configuração de  $NBIT = 0101$  foi selecionada manualmente.

### 7.1.1 Ganho unitário - 0 à 2,56V

A faixa de ganho unitário foi a primeira a ser avaliada, já que não apresenta um ganho extra sobre o sinal de entrada, evitando maiores distorções. Para essa configuração a chave equivalente a capacitância de entrada  $C_I = 200fF$  foi posta em nível alto.

#### 7.1.1.1 Teste de rampa

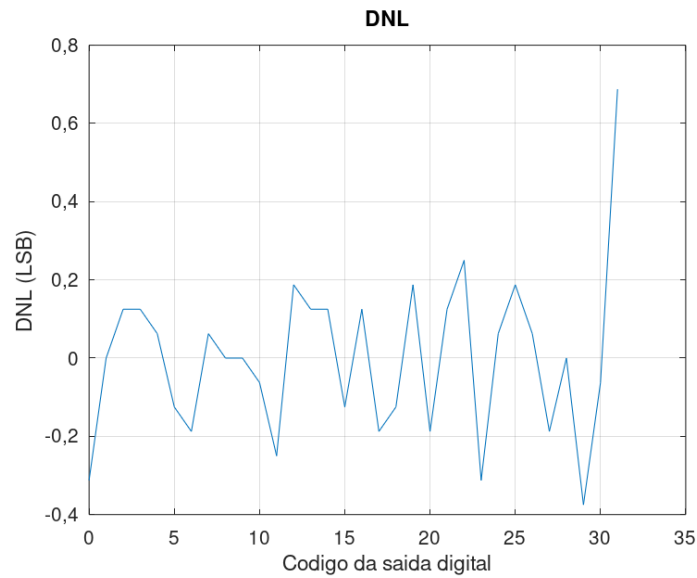
A Figura 33 apresenta o resultado para o teste de rampa nessa configuração. Conforme esperado os 32 níveis estão presentes na resposta e o conversor não apresenta grandes erros de não monotonicidade.



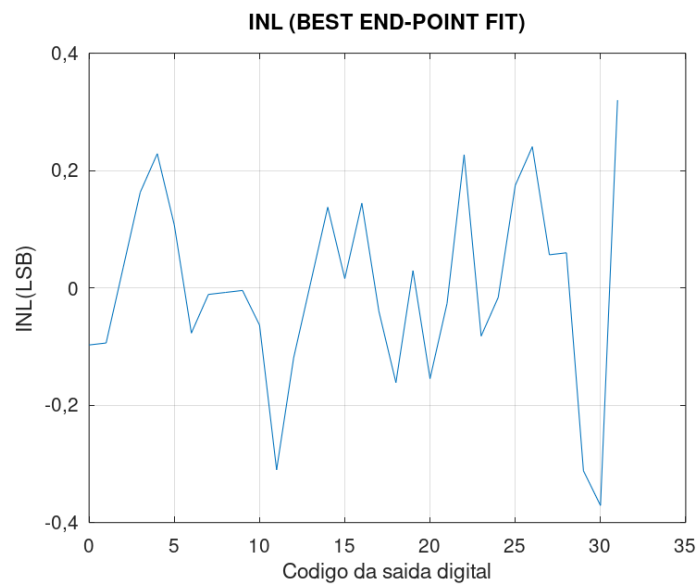
**Figura 33:** Teste de rampa com 5 bits para ganho unitário.

Explorando os resultados obtidos, podemos construir as Figuras 34 e 35, com os erros de DNL e INL respectivamente. Conforme as figuras, os níveis de erro se mantiveram abaixo de 1 LSB, porém com um erro maior próximo ao final da escala.

Em relação aos valores absolutos, o pior DNL negativo foi de  $-0,375$  LSB, enquanto o pior positivo foi de  $0,6875$  LSB. Já em relação ao INL, o pior valor negativo foi de  $-0,371$  LSB, enquanto o pior positivo foi de  $0,320$  LSB.



**Figura 34:** DNL para a primeira configuração de testes.

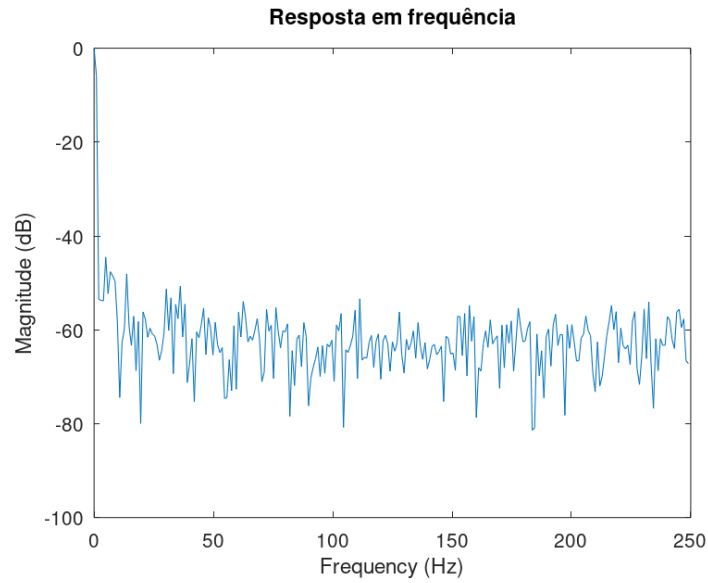


**Figura 35:** INL para a primeira configuração de teste.

#### 7.1.1.2 Teste senoidal

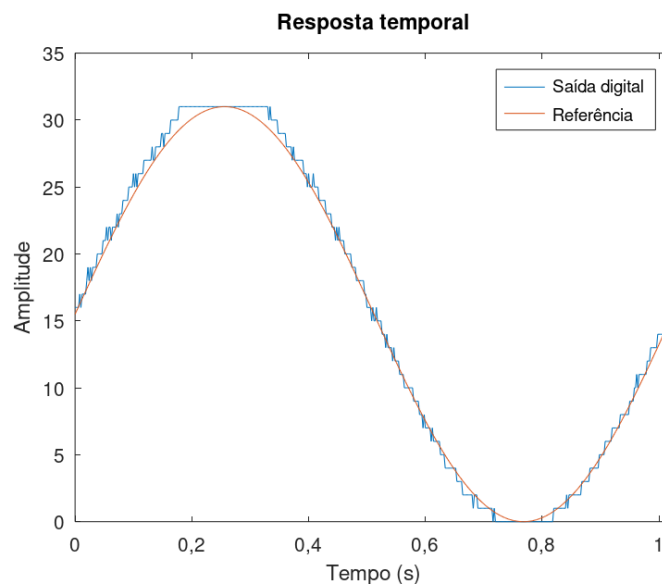
A Figura 36 apresenta o resultado no domínio do tempo para o teste senoidal nessa configuração, enquanto a Figura 37 mostra o resultado no tempo. Para a resposta em frequência, o piso do ruído em frequência foi de -60 dB.





**Figura 36:** Resposta em frequência para o teste senoidal.

Avaliando a magnitude do sinal aplicado e a magnitude dos erros obtidos podemos determinar a SNR e o ENOB nessa configuração, utilizando os valores RMS desses sinais. Avaliando esses resultados obtivemos uma SNR de 30,17 dB, correspondendo a um ENOB de 4,71 bits, ou seja houve uma pequena perda de resolução, mas esta foi inferior a 1 bit.



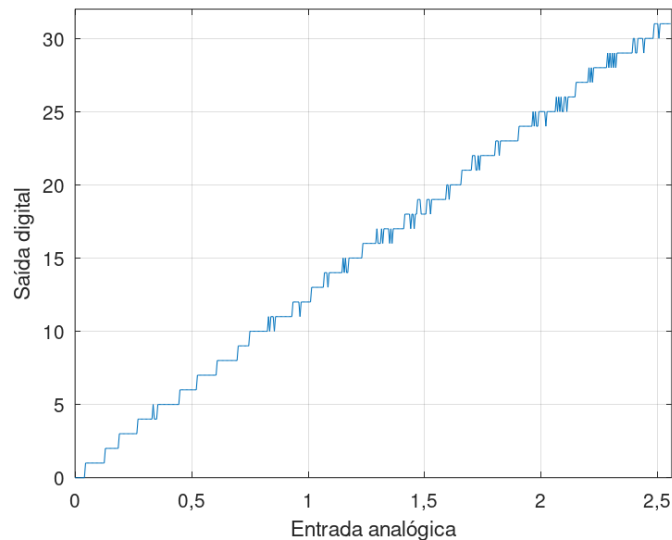
**Figura 37:** Resposta no tempo para o teste senoidal.

### 7.1.2 Ganho $1.5x$ - 0 à $1,70V$

A outra faixa de ganho testada foi a de  $1,5$ , permitindo trabalhar com um sinal de até  $1,70V$ . Esse ajuste é feito selecionando a chave equivalente a capacitância de entrada  $C_I = 300fF$ .

#### 7.1.2.1 Teste de rampa

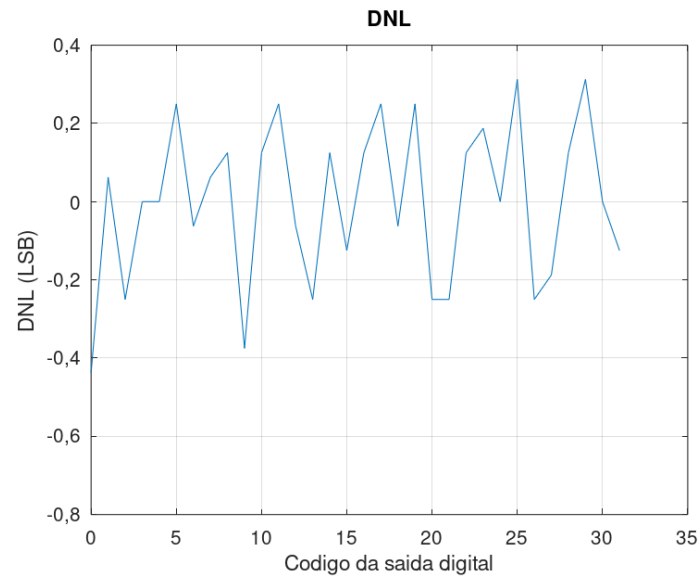
A Figura 38 apresenta o resultado para o teste de rampa nessa configuração. Assim como no primeiro teste, os 32 níveis estão bem representados, com poucos indícios de saturação do sinal.



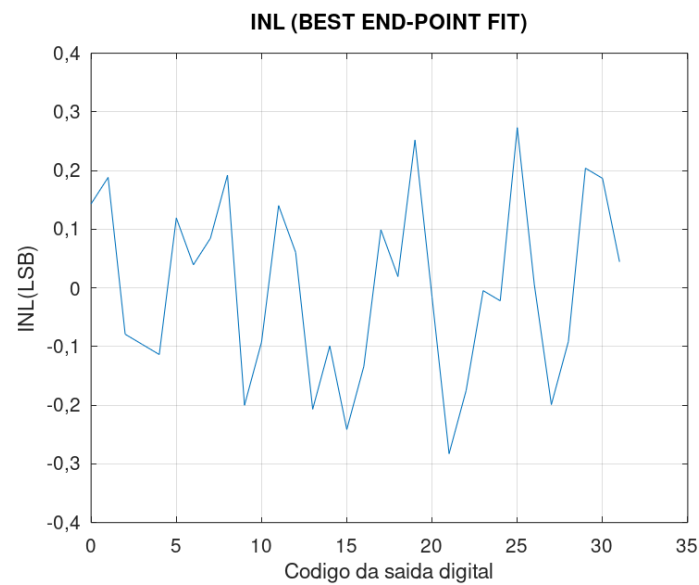
**Figura 38:** Teste de rampa com 5 bits para ganho  $1,5x$ .

Explorando os resultados obtidos, podemos construir as Figuras 39 e 40, com os erros de DNL e INL respectivamente. Assim como no primeiro caso, os níveis de erro se mantiveram abaixo de 1 LSB e com um INL menor, apesar do aumento do ganho.

Em relação aos valores absolutos, o pior DNL negativo foi de  $-0,4375$  LSB, enquanto o pior positivo foi de  $0,3125$  LSB. Já em relação ao INL, o pior valor negativo foi de  $-0,2826$  LSB, enquanto o pior positivo foi de  $0,2731$  LSB.



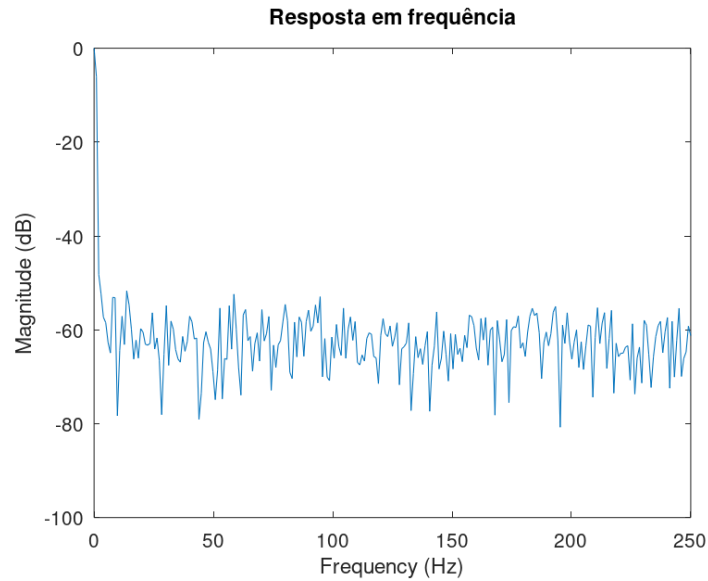
**Figura 39:** DNL para a segunda configuração de testes.



**Figura 40:** INL para a segunda configuração de teste.

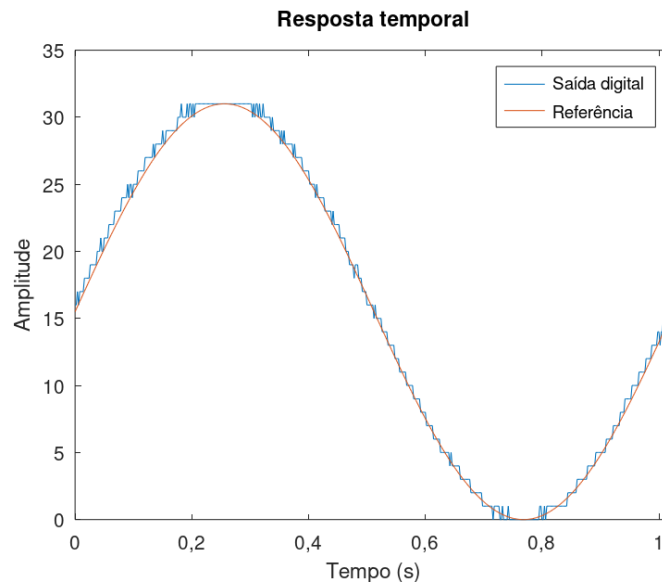
### 7.1.2.2 Teste senoidal

A Figura 41 apresenta o resultado no domínio do tempo para o teste senoidal nessa configuração, enquanto a Figura 42 mostra o resultado no tempo. Para a resposta em frequência, o piso do ruído em frequência também foi em torno de -60 dB nesse caso.



**Figura 41:** Resposta em frequência para o segundo teste senoidal.

Avaliando a magnitude do sinal aplicado e a magnitude dos erros obtidos podemos determinar a SNR e o ENOB nessa configuração, utilizando os valores RMS desses sinais. Avaliando esses resultados obtivemos uma SNR de 31,77 dB, correspondendo a um ENOB de 4,98 bits. Nesse teste o conversor apresentou um comportamento quase igual ao de um conversor de 5 bits ideal.



**Figura 42:** Resposta no tempo para o segundo teste senoidal.

## 7.2 RESOLUÇÃO DE 8 BITS

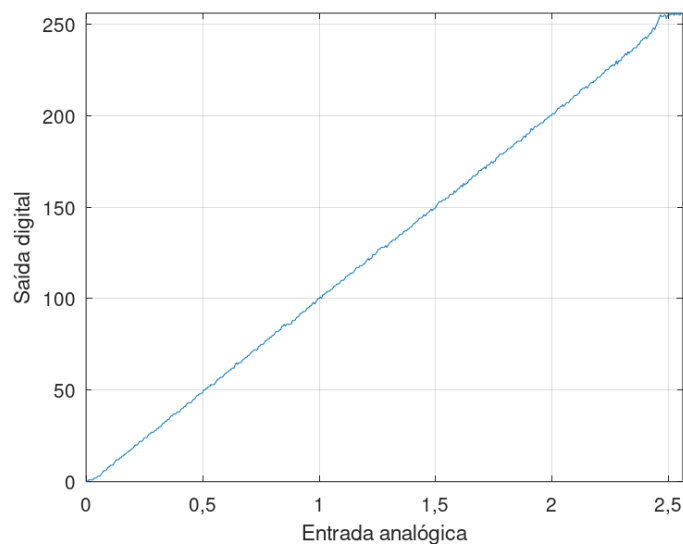
A outra configuração de testes utilizada com relação a resolução foi de 8 bits. Para isso a configuração de  $NBIT = 1000$  foi selecionada manualmente. Com essa mudança o processo de conversão se tornou um pouco mais lento, com mais níveis de quantização. De um lado o ADC se tornou mais preciso em suas medições, porém se tornou mais suscetível a influencia de ruídos e erros.

### 7.2.1 Ganho unitário - 0 à 2,56V

Novamente a faixa de ganho unitário foi a primeira a ser avaliada, utilizando a capacitância de entrada  $C_I = 200fF$ .

#### 7.2.1.1 Teste de rampa

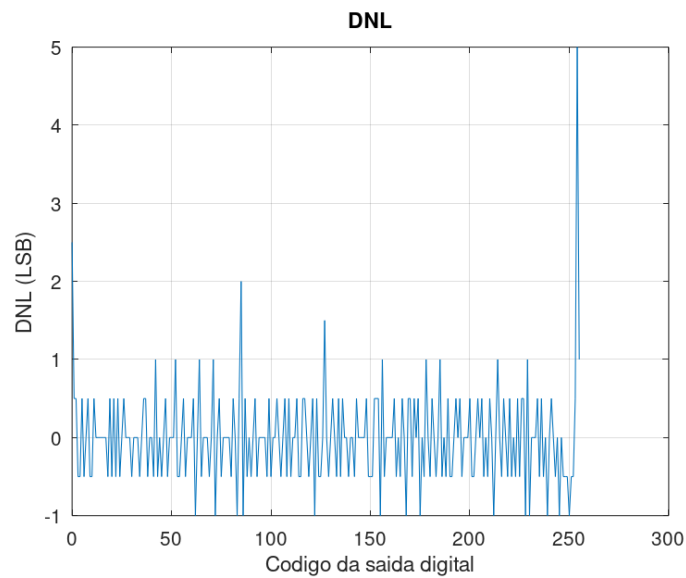
A Figura 43 apresenta o resultado para o teste de rampa nessa configuração. Conforme esperado os 255 níveis estão presentes na resposta. No entanto o conversor apresentou uma saturação do sinal próximo ao fim da escala.



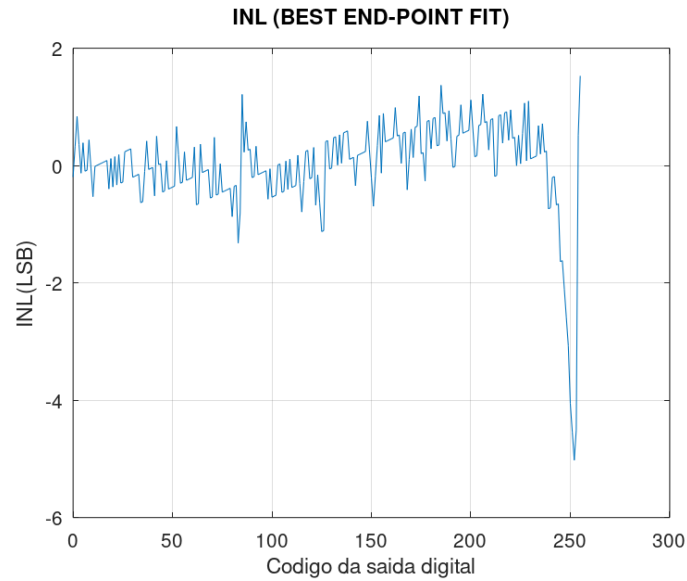
**Figura 43:** Teste de rampa com 8 bits para ganho unitário.

Explorando os resultados obtidos, podemos construir as Figuras 44 e 45, com os erros de DNL e INL respectivamente. Conforme as figuras, os níveis de erro foram bem superiores aos obtidos com a resolução de 5 bits, ultrapassando 1 LSB. Isso se deve a baixa quantidade de amostras obtidas em relação a resolução do ADC, um ponto de melhoria seria dividir a rampa em mais pontos, como 2048 ou 4096.

Em relação aos valores absolutos, o pior DNL negativo foi de  $-1$  LSB, enquanto o pior positivo foi de 5 LSB. Já em relação ao INL, o pior valor negativo foi de  $-5,021$  LSB, enquanto o pior positivo foi de  $1,528$  LSB. Esses níveis de erro refletem mais as limitações da interface de teste do que problemas com o conversor.



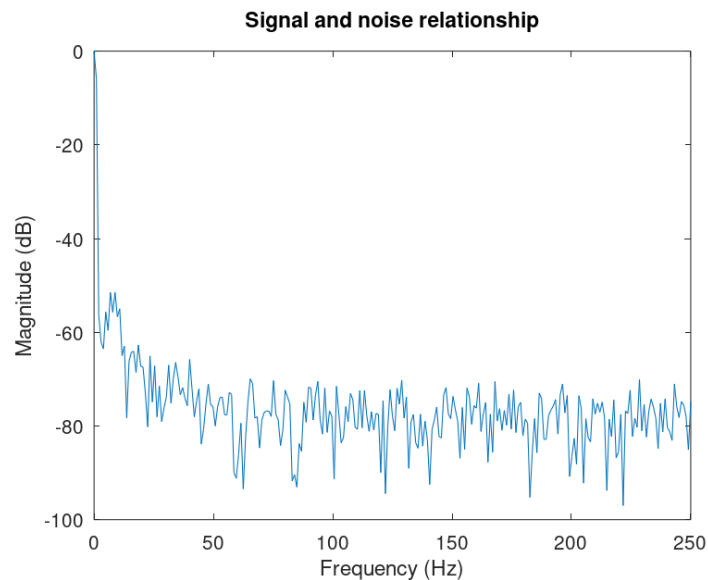
**Figura 44:** DNL para a terceira configuração de testes.



**Figura 45:** *INL para a terceira configuração de teste.*

#### 7.2.1.2 Teste senoidal

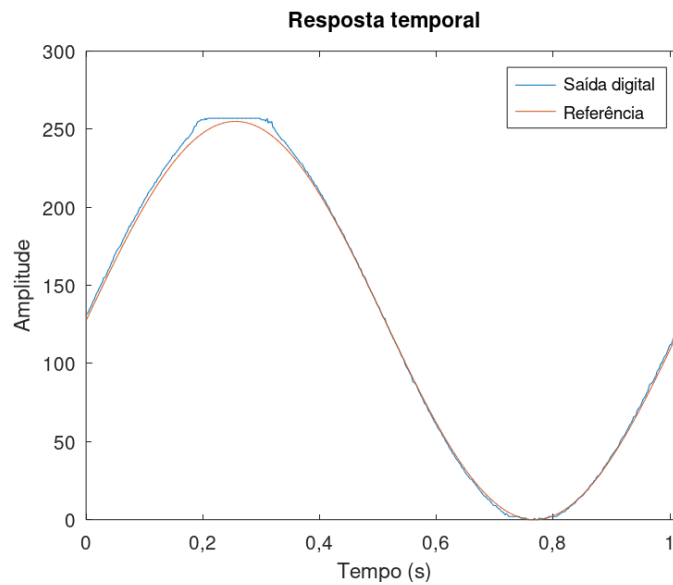
A Figura 46 apresenta o resultado no domínio do tempo para o teste senoidal nessa configuração, enquanto a Figura 47 mostra o resultado no tempo. Para a resposta em frequência nessa configuração o piso do ruído em frequência esteve mais próximo de -80 dB



**Figura 46:** *Resposta em frequência para o teste senoidal.*

Avaliando a magnitude do sinal aplicado e a magnitude dos erros obtidos podemos determinar a SNR e o ENOB nessa configuração, utilizando os valores RMS desses sinais. Avaliando esses resultados obtivemos uma SNR de 34,46 dB, correspondendo a um ENOB de 5,43 bits.

Pela Figura 47 fica claro uma saturação na crista da senoide, que causou uma piora na relação sinal ruído e conseqüentemente perda no ENOB. Situação similar a verificada com o teste estático.



**Figura 47:** Resposta no tempo para o teste senoidal.

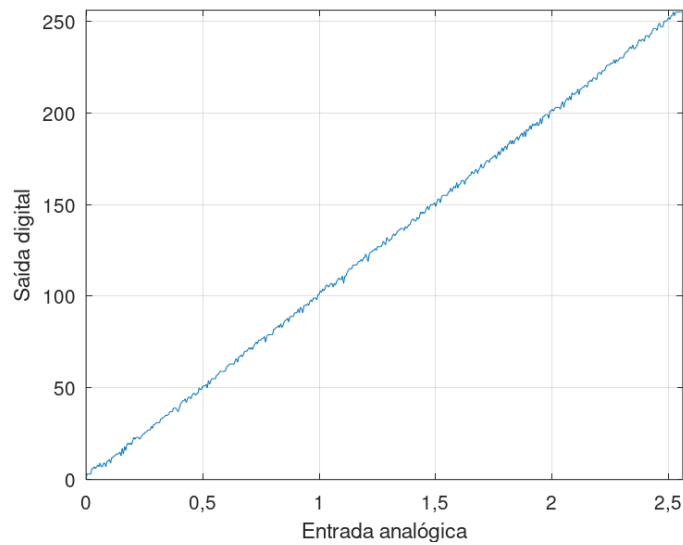
### 7.2.2 Ganho 1.5x - 0 à 1,70V

Igualmente ao teste com 5 bits, também se avaliou a faixa de 1,7 V, utilizando a capacitância de entrada  $C_I = 300\text{fF}$ .

#### 7.2.2.1 Teste de rampa

A Figura 48 apresenta o resultado para o teste de rampa nessa configuração. Nessa situação não ocorreu saturação significativa do sinal nos níveis de quantização superiores.

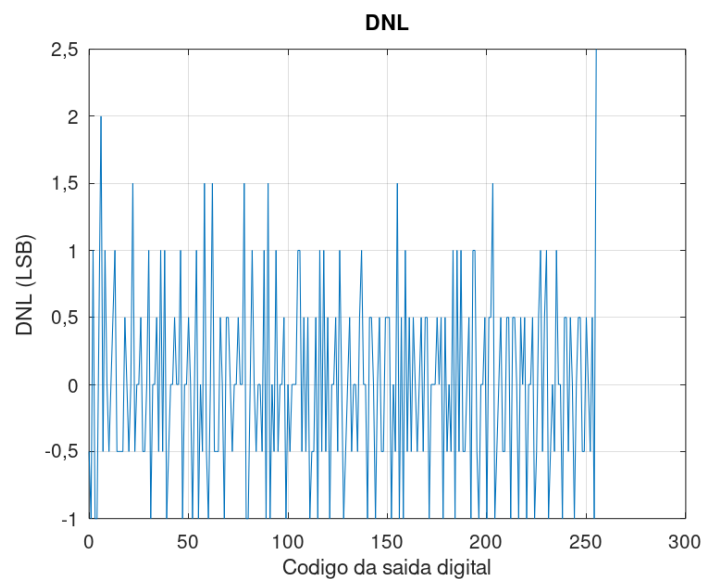




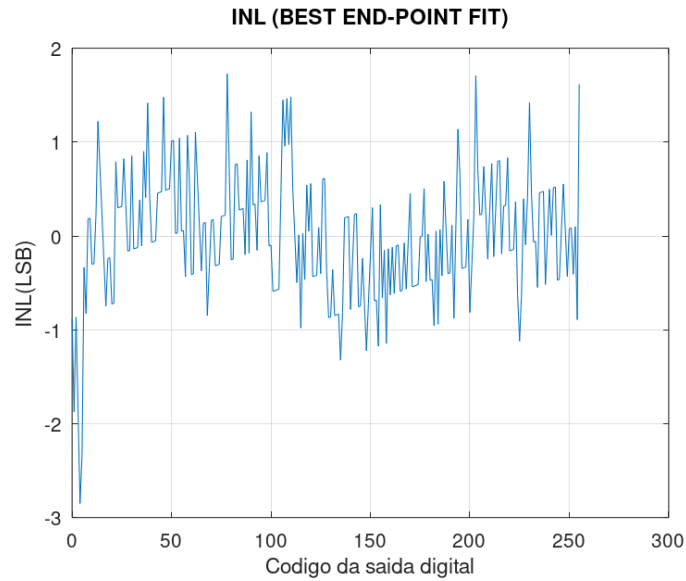
**Figura 48:** *Teste de rampa com 8 bits para ganho 1,5x.*

Explorando os resultados obtidos, podemos construir as Figuras 49 e 50, com os erros de DNL e INL respectivamente. Como no outro teste com resolução de 8 bits, os erros estiveram acima de 1 LSB, devido a baixa quantidade de amostras em relação a resolução do ADC.

Em relação aos valores absolutos, o pior DNL negativo foi de  $-1$  LSB, enquanto o pior positivo foi de  $2,5$  LSB. Já em relação ao INL, o pior valor negativo foi de  $-2,849$  LSB, enquanto o pior positivo foi de  $1,729$  LSB.



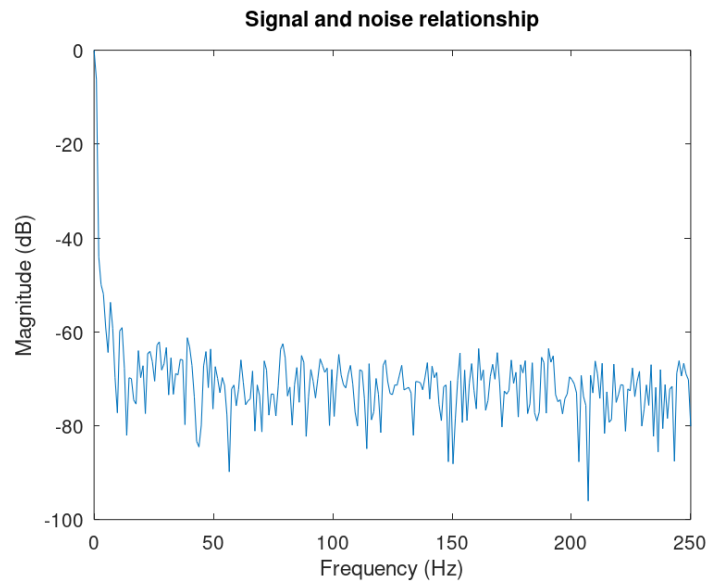
**Figura 49:** *DNL para a quarta configuração de testes.*



**Figura 50:** *INL para a quarta configuração de teste.*

#### 7.2.2.2 Teste senoidal

A Figura 51 apresenta o resultado no domínio do tempo para o teste senoidal nessa configuração, enquanto a Figura 52 mostra o resultado no tempo. Para a resposta em frequência, o piso do ruído em frequência se manteve próximo de -70 dB nesse caso.

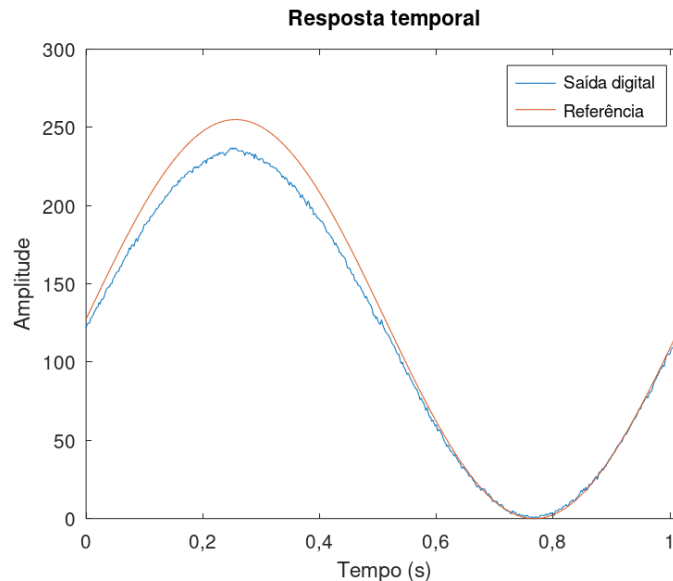


**Figura 51:** *Resposta em frequência para o quarto teste senoidal.*

Avaliando a magnitude do sinal aplicado e a magnitude dos erros obtidos podemos determinar a SNR e o ENOB nessa configuração, utilizando os valores RMS desses sinais.

Avaliando esses resultados obtivemos uma SNR de 22,32 dB, correspondendo a um ENOB de 3,41 bits.

Nesse teste o DAC apresentou problemas em suprir energia no ciclo positivo da senoide, produzindo uma perda de tensão que afetou consideravelmente a resposta do conversor. Com isso, foi esperado que as especificações dinâmicas fossem impactadas pela inserção desse distúrbio não planejado.



**Figura 52:** Resposta no tempo para o segundo teste senoidal.

### 7.3 ANÁLISE DOS RESULTADOS

Nesta seção foram realizados um total de 8 testes com o ADC, avaliando seu comportamento estático e dinâmico. Ficou evidente que, por depender de muitas tensões e correntes externas o conversor está suscetível a distúrbios e perturbações nessas linhas de alimentação. Assegurar a precisão das tensões de teste é essencial para a qualidade dos dados obtidos, principalmente com maiores resoluções.

Avaliando os testes, ao diminuir a resolução do conversor este se apresentou bem mais robusto as incertezas paramétricas inerentes ao ambiente de teste, chegando a obter resultados muito próximos do ideal. Por outro lado, ao aumentar a resolução, o ganho em precisão nas medidas realizadas é contrabalanceado por uma maior influência a ruídos e distúrbios.

Os testes com 8 bits mostraram a importância da escolha adequada do número de amostras e da subdivisão de níveis para teste. Enquanto nos teste de 5 bits haviam 16 amostras para cada nível lógico, para 8 bits só haviam 2, já que se fixou 512 amostras para padronização dos testes. O incremento no número de amostragens realizadas certamente melhoraria os resultados obtidos, e se somado ao ajuste fino das tensões, permitiria obter resultados ainda mais satisfatórios para o desempenho do conversor.

## 8 Conclusão

A execução deste trabalho propiciou a obtenção e o aprofundamentos dos conhecimentos acerca de conversores analógico-digitais, as suas topologias, formas de testes e análise de desempenho. Cada etapa foi essencial para a construção do circuito de teste, desenvolvimento do algoritmo de teste e processamento dos dados obtidos.

A primeira etapa, com o estudo sobre os conversores A/D, suas características e arquiteturas mais utilizadas, solidificou os conhecimentos sobre este tópico. Foi possível perceber a relação entre as características intrínsecas de cada topologia com o desempenho que estas apresentam e seu comportamento. As vantagens e desvantagens de cada modelo puderam ser evidenciadas, servindo de guia para escolhas futuras.

O teste do ADC proposto nesse trabalho demandou o estudo de sua arquitetura, buscando entender suas diferenças em relação a outros modelos. O estudo do modo de operação permitiu compreender as vantagens da técnica de capacitor chaveado em circuitos integrados, a robustez da arquitetura de conversores de rampa múltipla e a flexibilidade proporcionada pelos ganhos programáveis. Os detalhes do modo de operação ajudaram a planejar as especificações do circuito de teste necessário para o conversor.

Os conhecimentos de testes em ADCs conforme a norma IEEE 1241-2010 contribuíram para o desenvolvimento do algoritmo de teste para o circuito. Ficaram evidentes diferentes métodos de obtenção de figuras de mérito para conversores, sendo escolhido aqueles que melhor se adaptavam ao contexto desse trabalho.

Foi realizada a montagem do circuito de teste, em conjunto com algoritmos de teste e de análise de resultados, atingindo o objetivo de extrair as figuras de mérito do conversor. Ficou evidente que o ADC testado possui um bom desempenho para baixas resoluções, tanto em valores estáticos como dinâmicos. No entanto é necessário um refino no sistema de testes para validação de resoluções maiores, os resultados obtidos divergiram muito do ideal, mas foi possível enumerar os pontos de melhoria.

Por fim, o trabalho permitiu averiguar o funcionamento do conversor analógico-digital integrador com faixa de entrada e resolução programável a capacitor chaveado

proposto por [7], validando sua operação e características em alguns cenários de testes. Foi possível testá-lo e operá-lo por meio de uma interface com um microcontrolador ARM, foram obtidos parâmetros bem satisfatórios para a faixa de 5 bits e se melhorou a documentação de testes sobre esse conversor específico.

## 8.1 TRABALHOS FUTUROS

Para a continuidade do trabalho, sugere-se: O uso de componentes com maior precisão no circuito de testes, a fim de evitar variações nos reguladores; O desenvolvimento do circuito de teste em uma placa de circuito impresso, a fim de evitar interferências devido a conexões com fios; A comparação com conversores comerciais disponíveis e a caracterização da temporização do conversor.

# A Codigos utilizados para testes

## A.1 TESTE DE RAMPA

```
1 #include <stdio.h>
#include "MKL25Z4.h"
#include "fsl_debug_console.h"

#include "spi.h"
6 #include "set_clock.h"
#include "pwm.h"
#include "gpio_port.h"
#include "delay.h"

11 volatile uint32_t msTicks; /* counts 1ms
    timeTicks */
uint32_t systemclock_=1;
uint16_t value=0;
uint8_t data[2] = {0,0};
uint32_t contador[101];
16 uint32_t analog = 0;

void SysTick_Handler(void) {
    msTicks++;
}

21 /* Delay for argument number of systicks */
__INLINE static void delay (uint32_t dlyTicks) {
    uint32_t curTicks;
    curTicks = msTicks;
26 while ((msTicks - curTicks) < dlyTicks);
}

void writedac(uint16_t add){
    analog = analog + add;
31 GPIOE->PCOR= (1 << 4);

    delay_us(10);

    data[0] = analog / 0x0100;
36 data[1] = analog % 0x0100;
    SPI_write(data, 2, 0x00);
    delay_us(10);

    GPIOE->PSOR = (1 << 4);
```

```

41     delay_us(100);

        GPIOC->PSOR = (1 << 7);
        delay_us(75);
        GPIOC->PCOR = (1 << 7);
46
        PORTA->ISFR = (uint32_t)(1 << 2);
        NVIC_EnableIRQ(PORTA_IRQn); //Enable interrupt on IRQn
    }

51 void PORTA_IRQHandler() {
        NVIC_DisableIRQ(PORTA_IRQn);
        PORTA->ISFR = (uint32_t)(1 << 2);

        delay_us(5);
56     GPIOC->PSOR = (1 << 0);
        delay_us(10);
        GPIOC->PCOR = (1 << 0);

        for(int cont = 0; cont < 40; cont++){contador[cont] = (GPIOB->
            PDIR);        }
61     for(int i=0; i < 40; i++){PRINTF("%d", contador[i]);}
        PRINTF("\n");
        uint16_t add=128;

        if(analog < (uint16_t)65535){
66         writedac(add);
            return;
        }
        return;
    }
71
int main(void) {
    PRINTF("Hello World\n");
    clock_init();
76     SystemCoreClockUpdate(); // * Get Core
        Clock Frequency */
        systemclock_ = SystemCoreClock;

    __asm volatile("cpsie i"); //Enable Global Interrupts

81     SIM->SCGC5 |= (uint32_t)(1 << 9);
    PORTA->PCR[2] = 0;
    PORTA->PCR[2] = PORT_PCR_MUX(1) | PORT_PCR_IRQC(0x09); //
        Interrupt on rising and falling edge
    GPIOA->PDDR &= ~(uint32_t)(1 << 2);
    GPIOA->PCOR = (uint32_t)(1 << 2);
86     PORTA->ISFR = (uint32_t)(1 << 2);

```



```

SIM->SCGC5 |= SIM_SCGC5_PORTD_MASK;

    gpio_init();
91
GPIOC->PCOR |= (1 << 7);
    GPIOC->PCOR |= (1 << 0);

    pwm_init();
96
    SPI_init();

    //Teste de DAC
    // Inicializa PTE4 - Pino CS
SIM->SCGC5 |= SIM_SCGC5_PORTE_MASK;
101
PORTE->PCR[4] = PORT_PCR_MUX(1);
GPIOE->PDDR = (1 << 4);

    // Set pino CS = 1
GPIOE->PSOR = (1 << 4);
106

    // Clock = 20 MHz - Prescale = 1 - Divisor = 2
    delay_us(10);
    // CS = 0 - Inicia a comunicacao com o MAX 541
GPIOE->PCOR = (1 << 4);
111

    // Aguarda periodo antes de enviar os dados para o MAX 541
    delay_us(10);

    // Envia comando de Escrita para o MAX 541
116
    // Envia dado para configurar o valor do DAC do bit mais
        significativo para o menos significativo
    data[0] = analog / 0x0100;
    data[1] = analog % 0x0100;
    SPI_write(data, 2, 0x00);
    delay_us(10);
121

    // CS = 1 - Encerra a comunicacao com o MAX 541
GPIOE->PSOR = (1 << 4);

    delay_us(100);
126

    GPIOC->PSOR = (1 << 0);
    delay_us(10);
    GPIOC->PCOR = (1 << 0);

131
    delay_us(50);

    GPIOC->PSOR = (1 << 7);
    delay_us(50);

```

```

136     GPIOC->PCOR = (1 << 7);
        PORTA->ISFR = (uint32_t)(1 << 2);
        NVIC_EnableIRQ(PORTA_IRQn); //Enable interrupt on IRQn
        while(1) {}
    }

```

## A.2 TESTE DE SENOIDE

```

#include <stdio.h>
#include "MKL25Z4.h"
#include "fsl_debug_console.h"
#include "spi.h"
5  #include "set_clock.h"
#include "pwm.h"
#include "gpio_port.h"
#include "delay.h"

10 volatile uint32_t msTicks; // counts 1ms
    timeTicks */
uint32_t systemclock_=1;
uint16_t value=0;
uint8_t data[2] = {0,0};
uint32_t contador[101];
15 uint16_t test[523] = {33170, 33572, ..., 32365,32767};
uint16_t analog = 0;
uint16_t count = 0;

void SysTick_Handler(void) {
20     msTicks++;
}

/* Delay for argument number of systicks */
__INLINE static void delay (uint32_t dlyTicks) {
25     uint32_t curTicks;
        curTicks = msTicks;
        while ((msTicks - curTicks) < dlyTicks);
}

30 void writedac(){
    GPIOE->PCOR= (1 << 4);
    delay_us(10);

    data[0] = test[count] / 0x0100;
35    data[1] = test[count] % 0x0100;
    SPI_write(data, 2, 0x00);

    delay_us(10);
}

```

```

GPIOE->PSOR = (1 << 4);
40
    delay_us(100);

GPIOC->PSOR = (1 << 7);
    delay_us(75);
45
GPIOC->PCOR = (1 << 7);

PORTA->ISFR = (uint32_t)(1 << 2);
    NVIC_EnableIRQ(PORTA_IRQn); //Enable interrupt on IRQn
}
50
void PORTA_IRQHandler() {
    NVIC_DisableIRQ(PORTA_IRQn);
    PORTA->ISFR = (uint32_t)(1 << 2);

55
    delay_us(5);
    GPIOC->PSOR = (1 << 0);
    delay_us(10);
    GPIOC->PCOR = (1 << 0);

60
    for(int cont = 0; cont < 35; cont++){
        contador[cont] = (GPIOB->PDIR);
    }

    for(int i=0; i < 35; i++){
65
        PRINTF("%d", contador[i]);
    }
    PRINTF("\n");

    if(count < 512){
70
        count++;
        writedac();
    }
    return;
}
75
int main(void) {
    PRINTF("Hello World\n");
    clock_init();
    SystemCoreClockUpdate(); // Get Core
    // Clock Frequency */
80
    systemclock_ = SystemCoreClock;

    __asm volatile("cpsie i"); //Enable Global Interrupts
    //NVIC_SetPriority(PORTA_IRQn, (1UL << __NVIC_PRIO_BITS) - 1UL);
    SIM->SCGC5 |= (uint32_t)(1 << 9);
85
    PORTA->PCR[2] = 0;

```

```

PORTA->PCR[2] = PORT_PCR_MUX(1) | PORT_PCR_IRQC(0x09); //
    Interrupt on rising and falling edge
GPIOA->PDDR &= ~(uint32_t)(1 << 2);
GPIOA->PCOR = (uint32_t)(1 << 2);
PORTA->ISFR = (uint32_t)(1 << 2);
90
SIM->SCGC5 |= SIM_SCGC5_PORTD_MASK;
    gpio_init();

GPIOC->PCOR |= (1 << 7);
95
    GPIOC->PCOR |= (1 << 0);

    pwm_init();
    SPI_init();
    //Teste de DAC
100
    // Inicializa PTE4 - Pino CS
SIM->SCGC5 |= SIM_SCGC5_PORTE_MASK;
PORTE->PCR[4] = PORT_PCR_MUX(1);
GPIOE->PDDR = (1 << 4);

105
    // Set pino CS = 1
GPIOE->PSOR = (1 << 4);

    // Clock = 20 MHz - Prescale = 1 - Divisor = 2
    delay_us(10);
110
    // CS = 0 - Inicia a comunicacao com o MAX 541
GPIOE->PCOR = (1 << 4);

    // Aguarda periodo antes de enviar os dados para o MAX 541
    delay_us(10);
115

    // Envia comando de Escrita para o MAX 541
    // Envia dado para configurar o valor do DAC do bit mais
        significativo para o menos significativo
data[0] = test[count] / 0x0100;
data[1] = test[count] % 0x0100;
120
SPI_write(data, 2, 0x00);
    delay_us(10);

    // CS = 1 - Encerra a comunicacao com o MAX 541
GPIOE->PSOR = (1 << 4);
125

    delay_us(100);

GPIOC->PSOR = (1 << 0);
    delay_us(10);
130
GPIOC->PCOR = (1 << 0);

    delay_us(50);

```

```
135     GPIOC->PSOR = (1 << 7);
        delay_us(50);
        GPIOC->PCOR = (1 << 7);

        PORTA->ISFR = (uint32_t)(1 << 2);
        NVIC_EnableIRQ(PORTA_IRQn); //Enable interrupt on IRQn
140     while(1) {}
    }
```

# B Codigos utilizados para processamento de dados no MATLAB

## B.1 ALGORITMO PARA ANÁLISE DE TESTE DE RAMPA

```
close all
clear all
clc
4
%%
Y = 2.56;
nbit = 5;

9 flag = 0;
fid = fopen('5_static_170.txt');
tmp = textscan(fid, '%s');

A = uint64(zeros(513,18));
14 %%
for i = 1:1:length(tmp{1,1})
    A(i,:) = uint64(tmp{1,1}{i,1}(8:25)) -48;
end

19 B = uint64(zeros(513,1));
C = zeros(513,1);
%%
for i = 1:1:length(A)
    if(A(i,1) == 1)
24         flag = 1;
        end
        % 8bits
        %B(i,1) = (A(i,18) && A(i,17)) * 1 + (A(i,16) && A(i,15)) * 2 + (
            A(i,14) && A(i,13)) * 4 + (A(i,12) && A(i,11)) * 8 + (A(i,10)
            && A(i,9)) * 16 + (A(i,8) && A(i,7)) * 32 + (A(i,6) && A(i,5))
            * 64 + (A(i,4) && A(i,3)) * 128 + (A(i,2) && A(i,1)) * 256;

29        % 5 Bit
        B(i,1) = (A(i,18) && A(i,17)) * 1 + (A(i,16) && A(i,15)) * 2 + (A
            (i,14) && A(i,13)) * 4 + (A(i,12) && A(i,11)) * 8 + (A(i,10)
            && A(i,9)) * 16 + ((A(i,8) && A(i,7)) || (A(i,6) && A(i,7))) *
            32;

        C(i,1) = B(i,1);
        if flag == 1
```

```

34         C(i,1) = double(B(i,1)) - 64;
           flag = 0;
       end
   end

39 %% Ajuste DC
   C = C + abs(min(C));
   C = C(1:512);
   for i = 1:length(C)
       if(C(i,1) == double(32))
44         C(i,1) = 31;
       end
   end

49 %% Grafico
   t = 0:(Y/512):Y;
   figure
   plot(t(1:512),C)
   axis([0 Y 0 2^nbit])
54 xlabel('Entrada analogica')
   ylabel('Saida digital')
   grid
   %%hold on

59 %% DNL
   DNL = 0;
   aux = hist(C ,2^nbit);
   amostras = 512/2^nbit;
   dnl = (aux./ amostras) - 1;
64 %%dnl = [0.1 dnl(2:31) 0.1];
   figure;
   plot([0:2^nbit-1],dnl);
   grid on;
   title('DNL');
69 xlabel('Codigo da saida digital');
   ylabel('DNL (LSB)');

   DNL = sort(dnl);
   [i, j] = size(DNL);
74 str = sprintf('O pior DNL negativo do conversor eh %d LSB e o pior
       caso positivo eh %d LSB', DNL(i), DNL(j));
   disp(str);
   %% INL

   INL = 0;
79 inl = zeros(1,2^nbit);
   for i=1:2^nbit
       inl(i) = sum(dnl(1:i));
   end

```

```

end

84 %INL with end-points fit , i.e. INL=0 at end-points the straightline
    joining
    %the 2 end points
    %[p,S]=polyfit([1,2^numbit -2],[inl(1),inl(2^numbit -2)],1);
    %the best-fit straight line

89 [p,S]=polyfit([1:2^numbit],inl,1);
    inl=inl-p(1)*[1:2^numbit]-p(2);
    figure;
    plot([0:2^numbit-1],inl);
    grid on;
94 title('INL (BEST END-POINT FIT)');
    xlabel('Codigo da saida digital');
    ylabel('INL(LSB)');

    INL = sort(inl);
99 [i, j] = size(INL);
    str = sprintf('O pior INL negativo do conversor eh %d LSB e opior
        caso positivo eh %d LSB', INL(i), INL(j));
    disp(str)

```

## B.2 ALGORITMO PARA ANÁLISE DE TESTE SENOIDAL

```

close all
clear all
clc

4
Y = 2.56;
flag = 0;
nbit = 5;
fid = fopen('5_dinamic_170.txt');
9 tmp = textscan(fid, '%s');
    ref1 = [33170, 33572, ..., 32365, 32767];
    ref1 = transpose(ref1)/(2^16 -1);
    ref = 0:1:511;
    ref = transpose(sin(ref*2*pi/512)*15.5 + 15.5);
14 A = uint64(zeros(513,18));
    %%
    for i = 1:1:length(tmp{1,1})
        A(i,:) = uint64(tmp{1,1}{i,1}(8:25)) -48;
    end

19
B = uint64(zeros(513,1));
C = zeros(513,1);
%%
for i = 1:1:length(A)

```



```

24     if(A(i,1) == 1)
           flag = 1;
       end
       % 8bits
       %B(i,1) = (A(i,18) && A(i,17)) * 1 + (A(i,16) && A(i,15)) * 2 + (
           A(i,14) && A(i,13)) * 4 + (A(i,12) && A(i,11)) * 8 + (A(i,10)
           && A(i,9)) * 16 + (A(i,8) && A(i,7)) * 32 + (A(i,6) && A(i,5))
           * 64 + (A(i,4) && A(i,3)) * 128 + (A(i,2) && A(i,1)) * 256;
29
       % 5 Bit
       B(i,1) = (A(i,18) && A(i,17)) * 1 + (A(i,16) && A(i,15)) * 2 + (A
           (i,14) && A(i,13)) * 4 + (A(i,12) && A(i,11)) * 8 + (A(i,10)
           && A(i,9)) * 16 + ((A(i,8) && A(i,7)) || (A(i,6) && A(i,7))) *
           32;

       C(i,1) = B(i,1);
34     if flag == 1
           C(i,1) = double(B(i,1)) - 64;
           flag = 0;
       end
       end
39
       %% Ajuste DC
       C = C + abs(min(C));
       C = C(1:512);
       for i = 1:1:length(C)
44         if(C(i,1) == double(32))
               C(i,1) = 31;
           end
       end
       end
49 %%

       Ta = 2e-3;
       fa = 1/Ta;
       t = 0:Ta:(length(C)-1)*Ta;
54 Nb = 2^nbit;
       Tsim = length(C)*Ta;
       NFFT = 2^nextpow2(length(C));
       O = (fft(C,NFFT)/length(C));
       f = fa/2*linspace(0,1,NFFT/2 + 1);
59 dBO = 20*log10(2*abs(O(1:NFFT/2+1)));

       [1, c] = size(dBO);
       dBO = dBO - max(dBO);
       %%
64 figure , plot(f,dBO)
       title('Resposta em frequencia');
       xlabel('Frequency (Hz)')

```

```
ylabel('Magnitude (dB)')

69 %%
figure , plot(t,C)
hold on
plot(t,ref)
title('Resposta temporal');
74 xlabel('Tempo (s)')
ylabel('Amplitude')
legend('Saida digital','Referencia');
hold off

79 diff = ref - C;
SNR = 20*log10((rms(ref)/rms(diff)));
ENOB = (SNR - 1.76)/6.02;
str = sprintf('A SNR eh de %d dB,e o ENOB eh de %d bits', SNR, ENOB);
disp(str);
```

# Bibliografia

- [1] J. CAO, Q. YE e P. LI, «Resistance Furnace Temperature Control System Based on OPC and MATLAB», *Measurement and Control*, vol. 48, n.º 2, pp. 60–64, 2015. DOI: [10.1177/0020294015569257](https://doi.org/10.1177/0020294015569257).
- [2] D. HROVAT e J. SUN, «Models and control methodologies for IC engine idle speed control design», *Control Engineering Practice*, vol. 5, n.º 8, pp. 1093–1100, 1997, ISSN: 0967-0661. DOI: [https://doi.org/10.1016/S0967-0661\(97\)00101-9](https://doi.org/10.1016/S0967-0661(97)00101-9).
- [3] C. JOIN, G. ROBERT e M. FLIESS, «Model-Free Based Water Level Control for Hydroelectric Power Plants», *IFAC Proceedings Volumes*, vol. 43, n.º 1, pp. 134–139, 2010, 1st IFAC Conference on Control Methodologies and Technology for Energy Efficiency, ISSN: 1474-6670. DOI: <https://doi.org/10.3182/20100329-3-PT-3006.00026>.
- [4] I. SINCLAIR, *Sensors and transducers*. Elsevier, 2000.
- [5] G. L. MOSS, R. J. TOCCI e N. S. WIDMER, *Digital systems: principles and applications*, 12. edition. Pearson, 2017, ISBN: 9780134220130; 0134220137; 9781292162003; 1292162007.
- [6] A. BALBINOT e V. J. BRUSAMARELLO, *Instrumentação e Fundamentos de Medidas*, 2ª edição. LTC, 201, ISBN: 9788521618799.
- [7] T. B. BEZERRA, «Desenvolvimento de um Conversor A/D Integrador com Faixa de Entrada e Resolução Programável a Capacitor Chaveado», tese de mestrado, Universidade Federal do Maranhão, 2012, p. 103.
- [8] F. Maloberti), *Data Converters*, 1ª ed. Springer, 2007, ISBN: 9780387324852.
- [9] *Understanding Single-Ended, Pseudo-Differential and Fully-Differential ADC Inputs*, AN1108, Maxim Integrated, 2002.

- [10] *ADC and Resolution*, [https://spectrum-instrumentation.com/support/knowledgebase/hardware\\_features/ADC\\_and\\_Resolution.php](https://spectrum-instrumentation.com/support/knowledgebase/hardware_features/ADC_and_Resolution.php) Acessado em: 04-09-2022, Spectrum Instrumentation.
- [11] *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters*, 1241-2010, IEEE Instrumentation & Measurement Society, 2011.
- [12] R. NUNES, «Conversor Analógico-Digital Integrador a capacitor chaveado com faixa de entrada programável», tese de mestrado, Universidade Federal do Maranhão, 2010, p. 95.
- [13] M. J. M. PELGROM, *Analog-to-Digital Conversion*, 4<sup>a</sup> ed. Springer, 2021.
- [14] *"Taking the Mystery out of the Infamous Formula "SNR = 6.02N + 1.76dB" and Why You Should Care"*, MT-001, Rev. A, ANALOG DEVICES, out. de 2008.
- [15] *Understand SINAD, ENOB, SNR, THD, THD + N, and SFDR so You Don't Get Lost in the Noise Floor*, MT-003, Rev. A, ANALOG DEVICES, out. de 2008.